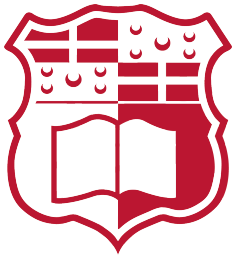

Reading Between Events: Exploring the Role of Machine Understanding in Event Tracking

Nicholas Mamo

Supervised by Dr Joel Azzopardi

Co-supervised by Dr Colin Layfield

Department of Artificial Intelligence

Faculty of Information & Communication Technology

University of Malta

October, 2023

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Artificial Intelligence.

L-Università ta' Malta

University of Malta Library – Electronic Thesis & Dissertations (ETD) Repository

The copyright of this thesis/dissertation belongs to the author. The author's rights in respect of this work are as defined by the Copyright Act (Chapter 415) of the Laws of Malta or as modified by any successive legislation.

Users may access this full-text thesis/dissertation and can make use of the information contained in accordance with the Copyright Act provided that the author must be properly acknowledged. Further distribution or reproduction in any format is prohibited without the prior permission of the copyright holder.

Copyright ©2023 University of Malta

WWW.UM.EDU.MT

First edition, Sunday 8th October, 2023

FACULTY~~/INSTITUTE/CENTRE/SCHOOL~~ of Information & Communication Technology
DECLARATION OF AUTHENTICITY FOR DOCTORAL STUDENTS

Student's Code _____

Student's Name & Surname Nicholas Mamo

Course Doctor of Philosophy

Title of Dissertation/Thesis
Reading Between Events: Exploring the Role of Machine Understanding in Event Tracking

(a) Authenticity of Thesis/Dissertation

I hereby declare that I am the legitimate author of this Thesis/Dissertation and that it is my original work.

No portion of this work has been submitted in support of an application for another degree or qualification of this or any other university or institution of higher education.

I hold the University of Malta harmless against any third party claims with regard to copyright violation, breach of confidentiality, defamation and any other third party right infringement.

(b) Research Code of Practice and Ethics Review Procedure

I declare that I have abided by the University's Research Ethics Review Procedures. Research Ethics & Data Protection form code 7866_23022021_Nicholas Mamo.

☒ As a Ph.D. student, as per Regulation 66 of the Doctor of Philosophy Regulations, I accept that my thesis be made publicly available on the University of Malta Institutional Repository.

□ As a Doctor of Sacred Theology student, as per Regulation 17 (3) of the Doctor of Sacred Theology Regulations, I accept that my thesis be made publicly available on the University of Malta Institutional Repository.

□ As a Doctor of Music student, as per Regulation 26 (2) of the Doctor of Music Regulations, I accept that my dissertation be made publicly available on the University of Malta Institutional Repository.

□ As a Professional Doctorate student, as per Regulation 54 of the Professional Doctorate Regulations, I accept that my dissertation be made publicly available on the University of Malta Institutional Repository.

_____ NICHOLAS MAMO

Signature of Student — Name in Full (in Caps)

8 October, 2023

Date

Acknowledgements

This work marks the end of nine years at the Department of Artificial Intelligence, at the University of Malta. The last three of those years, I spent researching Topic Detection and Tracking, and what I dubbed "the long road to understanding". My gratitude goes out to those who made this long road easier, if not shorter:

To Dr. Joel Azzopardi and Dr. Colin Layfield, my supervisors. They were the first to hear about my ideas and the first to encourage me to pursue them. Without them, this dissertation would have been much different.

To my teachers and lecturers for the tools necessary to start this work, and to my examiners and reviewers for the feedback to complete it.

To Professor Charlie Beckett, who I interviewed, for lending his expertise and perspective. His work gave more meaning to mine.

To DARPA for funding the first Topic Detection and Tracking study, and to TESS for funding mine.

Finally, to the friends and family members who accompanied me.

The research work disclosed in this publication is partially funded by the Tertiary Education Scholarships Scheme (Malta)

Abstract

Humans observe, humans understand, and then, humans act, but machines only act. The Topic Detection and Tracking (TDT) community realised, early on, that to accomplish its task, to detect and track events from the news media, it would not suffice to act without understanding. Yet the TDT community rarely sought to understand. Therefore when Twitter modernised the task, now to detect and track events from social media, researchers had no response to the new challenges: the volume and velocity, the brevity and the noise. Today, we ask more of our TDT algorithms. We demand that they detect events precisely, describe comprehensively and model formally. We demand that they meet our modern needs without answers to the questions posed by understanding. What does it mean to understand events? How can we understand events? How can understanding improve TDT? In this dissertation, we answer the three questions. We debate interpretations of understanding and adopt a structured, semantic definition of events: Who does What, Where and When. We develop DEPICT, a novel algorithm to understand Who participates in events and Where, and EVATE to understand What can happen from past events. And with understanding, we propose SEER, a novel TDT algorithm that tracks events with increased precision, coverage and sensitivity, and which drives a novel and simplified event modeller. In the end, we demonstrate that understanding remains a worthwhile ambition. Machines can observe and understand, and when they do, they act better.

Contents

List of Publications

Mamo, N., Azzopardi, J., and Layfield, C. An Automatic Participant Detection Framework for Event Tracking on Twitter. *Algorithms*, 14(3):92, Mar 2021. doi:10.3390/a14030092[1]

Mamo, N., Azzopardi, J., and Layfield, C. Fine-grained Topic Detection and Tracking on Twitter. In *Proceedings of the 13th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management - (Volume 1)*, page 79–86, Remote, Oct 2021. SciTePress. doi:10.5220/0010639600003064[1]

Mamo, N., Azzopardi, J., and Layfield, C. Who? What? Event Tracking Needs Event Understanding. In *Proceedings of the 13th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management - (Volume 1)*, page 139–146, Remote, Oct 2021. SciTePress. doi:10.5220/0010639600003064

Mamo, N., Layfield, C., and Azzopardi, J. From Event Tracking to Event Modelling: Understanding as a Paradigm Shift. In Fred, A., Aveiro, D., Dietz, J., Bernardino, J., Masciari, E., and Filipe, J., editors, *Knowledge Discovery, Knowledge Engineering and Knowledge Management*, volume 1718, page 21–36. Springer Cham, Jul 2023. URL `https://link.springer.com/chapter/10.1007/978-3-031-35924-8_2`

Mamo, N., Azzopardi, J., and Layfield, C. The Myth of Reproducibility: A Review of Event Tracking Evaluations on Twitter. *Frontiers in Big Data*, 6:1067335, Apr 5, 2023. URL `https://www.frontiersin.org/articles/10.3389/fdata.2023.1067335/full`

[1] This publication extends our previous work [141].

List of Figures

List of Tables

List of Abbreviations

AI Artificial Intelligence 1
AP Average Precision 47
APD Automatic Participant Detection 5
ATE Automatic Term Extraction 5
DEPICT DEtecting Participants by Inferring Common Traits 35
EF Event Frequency 70
ELD Event TimeLine Detection 16
EMM Event Modelling and Mining 3
EVATE EVent-Aware Term Extractor 68
FIRE Finding Important News REports 16
ICF Inverse Corpus Frequency 70
IDF Inverse Document Frequency 71
IR Information Retrieval 4
LLR Log-Likelihood Ratio 83
MAP Mean Average Precision 47
MMR Maximal Marginal Relevance 129
NER Named Entity Recognition 4
NLP Natural Language Processing 19
NLTK Natural Language Toolkit 31
P@k Precision at k 47
PMI Pointwise Mutual Information 83
POS Parts of Speech 19
SEER Stream-Enabled Event Reporter 116

1

Introduction

The year is 1996. In the United States, four teams participate in the pilot study into Topic Detection and Tracking (TDT). TDT's objective: use Artificial Intelligence (AI) to segment, detect and track news stories in the media [7]. The first results leave room for improvement, so a year later, Allan et al. [9] offer a suggestion: understanding Who does What, Where and When, and Why and How in events. Today, applications for TDT abound, but the research area finds itself with more troubles and still, no understanding. Therefore in this dissertation we adopt the suggestion as our research question: can understanding improve TDT?

The suggestion to understand events was tinged with reluctance. Allan et al. [9] hoped that understanding the 'five Ws and one H' (the Who, What, Where, When, Why and How) would lead to "significant advances" but feared "that the gains may not be large." Over the next few years, as the TDT community grew, some followed the suggestion, but never with significant advances. Simple understanding did not solve the research area's troubles [169]. Sometimes, understanding worsened them [35; 120; 125; 139].

In 2006, Twitter launched and the research community cast understanding aside. The social network revolutionised news dissemination, and with it, the TDT task. News spread quickly on Twitter, and soon, it started to break there too [130]. Twitter chronicled the Arab Spring's organisation [110; 134], and famously broke the news of Osama bin Laden's death [98] and the Boston Marathon bombing [220; 245]. And still, Twitter grew and grew. By 2015, Twitter was producing 500 million tweets per day [258]. In 2019, Twitter users described an illness that the world would call, only weeks later, the COVID-19 pandemic [133].

There, on Twitter, TDT found purpose. Event tracking algorithms stopped repeating after the newswire and became the newswire [201]. Around the world, Reuters Tracer detected breaking news within minutes, usually much quicker than the news media [129; 130]. In Japan, the system by Sakaki et al. [227] detected most earthquakes within one minute. In the USA, the system by Zhao et al. [296] detected topics from NFL games within seconds. Tweets came to symbolise the antithesis to news reports.

Yet tweets came to symbolise the antithesis to news reports in other ways too. Twitter replaced the newswire's steady stream with too many tweets arriving too fast for heavy processing [86; 191]. Twitter replaced eloquent narratives with curt tweets no longer than 280 characters. And Twitter replaced the formality of news articles with vulgar informality and noise [102].

Years later, with Twitter's challenges in the backdrop, understanding resurfaced. The TDT community wondered again about understanding. Is noise not a failure to recognise relevance, objectivity and importance in tweets? Understanding could help TDT algorithms make sense of Twitter streams [26]. Are the research area's issues not a sign of ignorance about events? Understanding could drive algorithms [49] and improve their performance [28]. But the research community's solution was rarely understanding [26; 49; 94; 137].

Unable to harness tweets or understand events, progress in TDT stalled. We speak of similar flaws in the algorithm by Zhao et al. [296] from 2011 and in ours from 2019 [141]. Algorithms still only build timelines from popular events [93; 225], still succumb to noise and miss the details [79; 136; 146; 161]. TDT literature never made understanding events the problem, so understanding never became the solution.

The year is no longer 1996, but understanding still matters. Understanding still matters to the performance of event tracking algorithms [9], to their abilities to describe events [49] and, eventually, to comprehend and reason about them [39]. Understanding still matters because we expect machines to perform the job of a human with none of the knowledge.

Understanding matters beyond academic circles as well. The news industry, which inspired the research area of TDT, grows increasingly accepting of AI. In 2022, 70% of the respondents to a survey by Newman [180] felt AI would play "a key part in helping find or investigate stories using data." Journalists envision AI as a disruptor [179], a revolution in news gathering, production and distribution [18]. To the media industry, understanding matters because it determines the utility of TDT algorithms.

Therefore this dissertation answers the question: can understanding improve TDT?

1.1 | To understand events

Can understanding improve TDT? The question appears simple, its answer obvious, but you would not find it in early research. Those who followed the suggestion by Allan et al. [9] sought easy answers that would fit in manuscripts a few pages long, and the results confirmed the initial apprehension [35; 120; 125; 139]. Differently from those early experiments, we dedicate this entire dissertation to answering the research question from three aspects. What does it mean to understand events? How can we understand events automatically? How can understanding improve event tracking?

What does it mean to understand events?

In Chapter 2, we seek interpretations of understanding in past research on TDT. We find few and only implicit definitions. The research community rarely stopped to ask what it means to truly understand events. A philosophical debate on the definition of knowledge is moot, but the matter of what true knowledge entails remains central to our work. For our sake, Plato's widely-accepted [23], three-word definition suffices:

Definition 1 (**Knowledge, or understanding**[1])**.** Justified true belief. — Plato

Had early TDT research pursued knowledge more intently, it would likely have uncovered more difficulties. Events have complex structures and take different forms. We cannot understand what happens in events without understanding what constitutes events, but the research community could not agree on a common definition [225]. In part, researchers afforded not to. A spectator watching a football match can recognise a goal without understanding the game's rules. Likewise, a TDT algorithm can recognise an incident without a formal structure of events, such as by identifying an unexpected influx of tweets. Research thus subsisted without a definition.

Elsewhere, another research area could not begin to exist without a definition of events, nor without understanding them. In Event Modelling and Mining (EMM), detecting events only denotes the first step in the pursuit of true understanding, which requires representing events in a "semantically-meaningful way" [39]. Unlike TDT literature, EMM research could not but agree on a semantic definition. Therefore from EMM literature we adopt the following definition of events:

[1] A slight semantic nuance separates knowledge from understanding. Knowledge represents facts, whereas understanding suggests the observer's perception of knowledge. Ideally, we want algorithms to generate knowledge, but in reality, they will always generate what their designs perceive as knowledge. In short, they generate understanding. For the sake of simplicity, we consider knowledge and understanding as synonyms in this dissertation. Furthermore, throughout this dissertation we limit our scope of understanding to extracting information about Who is doing What, Where and When in events.

Definition 2 (**Event**). "An action, or a series of actions, or a change [What] that happens at [a] specific time [When] due to specific reasons [How/Why], with associated entities such as objects, humans [Who], and locations [Where]." — Chen and Li [39]

EXAMPLE: "At least 120 people [Who] are feared to have been killed [What] in a series of devastating attacks [How/Why] across Paris [Where] on Friday evening [When]." [203]

The definition serves three functions. First, it aligns our problem with the 'five Ws and one H', which Allan et al. [9] proposed and which we follow; for the sake of clarity, we capitalise references to the 'five Ws and one H' throughout this dissertation. Second, the definition links TDT with EMM, giving the former new purpose in the latter, as we argue in Chapter 6. Third, the definition provides a theoretical structure with which to understand events, which leads us to the second question.

How can we understand events automatically?

In Chapters 3 and 4 we understand Who does What and Where in events.[2] Understanding remains a challenging task even with a semantic structure of events. It demands that researchers leave behind a decades-long tradition that neglected knowledge. Understanding, then, demands novel ideas, but novel ideas alone may not suffice either. We develop and apply understanding on Twitter, the de facto modern standard of TDT research [201] that has defied Information Retrieval (IR)'s traditional tools [41]. Therefore understanding also demands novel tools.

In Chapter 3, we develop our understanding of the Who and the Where. Some research describes the Who and the Where as event terms [99; 289], the words that characterise an event and distinguish it from the others, normally persons, organisations or locations—named entities. Instinctively, the TDT community turned to Named Entity Recognition (NER), but in this work we reveal some of the flaws in the use of such tools. In Chapter 3, we show how NER models do not distinguish between named entities mentioned in passing and named entities with an active role in an event, participants. Previously, we defined participants as follows:

Definition 3 (**Event participant, or participant**). A person, location or organisation that affects or is affected by the event. — Mamo et al. [144]

EXAMPLE: The teams, players and venue in a football match; the political parties, candidates, and states and counties in an American election.

[2] Of the remaining 'five Ws and one H', the When follows implicitly from TDT's detection task; the primary function of a TDT algorithm is to detect When something happens. The Why and the How denote reasoning [39], and thus fit better within the scope of event mining.

We understand Who participates in events and Where with DEPICT, a novel algorithm. DEPICT does not discard NER's output altogether but refines it with the Automatic Participant Detection (APD) framework [144]. Unlike traditional NER models, which do not comprehend the nature of participants, DEPICT gleans an understanding about Who or Where they are and what they do. Because it understands, DEPICT can precisely discover the majority of an event's participants, including those that NER tools miss.

In Chapter 4, we develop our understanding of the What. The same research that speaks of event terms often also refers to domain terms, the words that characterise an event domain and distinguish it from the others, like how *vote* characterises the event domain of elections. Domain terms thus belong to event domains, so before we discuss terms any further, we formally define event domains:

Definition 4 (**Event domain, or domain**)**.** A group of events that share a common vocabulary, the domain terms. — Filatova et al. [66]; Hua et al. [99]; Yang et al. [289].

EXAMPLE: The domain of football matches; the domain of national elections.

Domain terms play a particular role in event domains. In general, terms may describe any concept, but in event domains, they describe "an action, or a series of actions, or a change" [39], or What happens in the domain's events. In TDT's simpler jargon, domain terms describe topics: the important key topics, like goals in football matches, and the comparatively unimportant non-key topics, like yellow cards. Formally, we define domain terms as follows:

Definition 5 (**Domain term, or term**[3])**.** Words or phrases that describe What happens during events from a particular event domain. — Hua et al. [99]

EXAMPLE: The terms *goal*, *score* and *corner* in the domain of football matches; *voting*, *recount* and *ballot* in the domain of elections.

We understand What happens in events with EVATE, a novel algorithm. We could find little research on extracting terms from event domains, and even less from Twitter, but we approach the problem as an Automatic Term Extraction (ATE) task: to recognize domain-related terms from domain-related corpora [13]. EVATE adapts both to event domains and to Twitter by observing the outputs of a TDT technique working on tweets. In the end, EVATE reciprocally serves TDT techniques with terms tailored to event tracking on Twitter, around which revolves our third question.

[3]For the sake of clarity, in this dissertation we italicise domain terms to distinguish them from words or concepts. When we refer to a concept, such as a goal in football matches, we do not italicise the word. However, when we mean to refer to a domain term, not a general concept, we italicise it: *goal*, not goal.

How can understanding improve event tracking?

In Chapters 5 and 6, we apply our understanding of Who does What and Where. On most of the occasions when TDT research explored understanding, the community did not distinguish between developing and applying understanding. The two meshed inseparably and left the suggestion by Allan et al. [9] in an answer-less limbo. In this dissertation, we undertake the test and explore whether the application of understanding can improve TDT.

In Chapter 5, we apply our understanding in a novel TDT algorithm, SEER. Historically, event tracking research could reserve only a limited role to understanding because the understanding itself was limited. In contrast, EVATE's understanding from Chapter 4—proper event understanding—tells us, in advance, What can plausibly happen in any event from a domain. Therefore understanding drives our new algorithm. In the end, through SEER we unequivocally answer the question of whether understanding can improve event tracking.

In Chapter 6, we combine the three algorithms in the understanding-driven event modeller. Detecting events no longer suffices: we need algorithms to model Who did What, Where and When [191]. In the understanding-driven event modeller, DEPICT understands Who is participating and Where, EVATE understands What may happen, and SEER understands When events happen. Later, we use the event models to build the event knowledge graph: a visualisation and a tool—a form of storytelling.

We have shared the data and algorithms used in this dissertation in two GitHub repositories. The data and its outputs, the ground truths and results, and the Jupyter Notebooks all reside in the `NicholasMamo/phd-data` GitHub repository. Separately, we have released all the algorithms as an open-source project in the `NicholasMamo/EvenTDT` GitHub repository with a GNU GPLv3 licence. The EvenTDT library, written in Python 3, includes a suite of 15 tools, 14,342 lines of documentation and an additional 8,585 lines of Python code, excluding tests. To the best of our knowledge, EvenTDT represents the largest open-source TDT library. We formalise our aims and objectives next.

1.2 | Aims and objectives

In this dissertation, we present what we believe to be the first study into the role of machine understanding in event tracking. We make none of the simplifying assumptions and adopt none of the simplified interpretations of understanding that pervade TDT's history. On the contrary, we discuss, in depth, what true event knowledge entails, and how the research community can develop and apply understanding. In the

end, we answer three questions. What does it mean to understand events? How can we understand events automatically? How can understanding improve event tracking?

We answer our research questions through five chapters. In the review chapter, Chapter 2, we survey literature to answer what it means to understand. In the understanding chapters, Chapters 3 and 4, we answer how we can understand events automatically. In the application chapters, Chapters 5 and 6, we answer how understanding improves event tracking. Every chapter answers one fundamental question that helps us comprehend the role of understanding in event tracking. Together, these questions form this dissertation's principal aims and objectives:

- What does it mean to understand events? The research community understood haphazardly, never pausing to question what it means to understand events. In Chapter 2, we trawl the research area's past for interpretations of understanding as we present TDT literature's first review dedicated to event knowledge.

- When does a named entity become a participant? Named entities did not improve TDT performance sufficiently [169] because they did not represent understanding, at least not semantic understanding. In Chapter 3, we understand the Who and the Where with DEPICT, an APD algorithm that understands, first and foremost, what makes participants of named entities.

- When does a word become a domain term? To understand the Who and the Where, researchers could use NER, and to understand the When, they could use temporal features, but the What had no direct equivalent. In Chapter 4, we understand the What with EVATE, the first semantic ATE technique designed for event tracking on Twitter.

- How can understanding improve TDT algorithms? Allan et al. [9] perceived understanding as a way to improve accuracy—only accuracy—but understanding can benefit algorithms in many ways. In Chapter 5, a case study on football matches, we apply EVATE's understanding in a novel TDT algorithm, SEER, to give the first definitive answer to the suggestion byAllan et al. [9].

- Where does TDT's next revolution lie? Like Twitter before it, understanding can be a paradigm shift for event tracking. In Chapter 6, a case study on British politics, we apply our understanding of Who does What, Where and When in the understanding-driven event modeller to explore how understanding can give TDT algorithms new purpose.

1.3 | Contributions

In the rest of this dissertation, we address the suggestion that Allan et al. [9] made in the aftermath of the TDT pilot study: understanding events. We argue on the merits of different forms of understanding, develop semantic understanding about Who does What and Where in events, and then apply the same knowledge to prove that understanding can indeed improve event tracking. In the end, we make the following contributions:

- In Chapter 1, we share the `NicholasMamo/EvenTDT` repository, the largest open-source TDT library
- In Chapter 2, we present the first literature review on understanding in TDT and other event-related research areas
- In Chapter 3, we propose DEPICT, a novel APD algorithm that understands Who participates in events and Where by understanding the participants themselves
- In Chapter 4, we propose EVATE, the first ATE algorithm that understands What happens in events from Twitter
- In Chapter 5, we propose SEER, the first TDT algorithm driven by foreknowledge of What can happen in an event, and which proves that understanding can improve event tracking
- In Chapter 6, we propose a novel event modeller simplified and augmented by understanding and understanding-driven TDT

Apart from the core material above, we have included an additional chapter as an appendix. In Appendix A, we review the most notorious of TDT literature's difficulties on Twitter: its evaluations. The appendix contains what we believe to be the most comprehensive survey yet on TDT evaluation methodologies on Twitter. Through the surveyed papers, we discuss the challenges to measure progress accurately and objectively, including the progress due to understanding.

It does not surprise us that the research community has not studied understanding in depth: the road to understanding is a long one. In this work, we avoid repeating the past mistakes that sought to shorten it, namely the hasty interpretations of what it means to understand. Because while the research community never considered the question seriously, it has proposed several implicit interpretations of event knowledge. We embark on the long road to understanding in the next chapter as we debate the virtues and flaws of the many meanings of understanding.

2

Review

The Long Road to Understanding

The long road to understanding should probably have been much shorter. The idea that machine understanding can improve machine performance is anything but revolutionary—certainly not in TDT, whose community of researchers recognised knowledge's potential in its inception. Understanding represents an intuitive idea, and like all intuitive ideas, it reappeared frequently over the years, which makes it all the more perplexing why TDT researchers did not study event knowledge more closely.

Allan et al. [9] were the first to introduce the idea of understanding events to TDT literature. It had been two years since the launch of the pilot project [7]. At the time, the TDT task had three objectives: to segment, detect and track events in news media. However, early algorithms struggled with performance when aggregating news articles, so at the end of the two-year pilot project, Allan et al. [9] reluctantly proposed understanding as a solution:

> Significant advances in Event Tracking accuracy are most likely to be obtained using some limited form of story parsing and "understanding". It is likely to be useful to capture notions of who, what, where, when, why, and how, although the well-known past experience from IR suggests that the gains may not be large.
>
> — Allan et al. [9]

Over the next few years, the TDT community followed the suggestion and explored understanding, but knowledge barely improved performance. In the meantime, not only did algorithms struggle to overcome the early challenges but new ones appeared

when researchers migrated to Twitter. In this chapter, we follow the long road to understanding, from the research area's origins to modern day applications of TDT on Twitter, and ultimately answer the following questions:

- Why does TDT need understanding? Algorithms were limited even before Twitter introduced new challenges, but Allan et al. [9] believed that understanding could solve many of the area's problems. In Section 2.1, we explore how TDT's challenges changed over the years and identify applications for understanding.
- What does it mean to understand events? Since Allan et al. [9] first proposed understanding, the TDT community has adopted different interpretations of what it means to understand events. In Section 2.2, we contrast these different perspectives and their suitability as event understanding.
- How can TDT research understand events? A major obstacle to understanding is characterising the type of information that would be useful [137; 169]. In Section 2.3, we adopt a structured definition of events based on the 'five Ws and one H', and explain how researchers can generate understanding about the Who, Where and What automatically.

Material from this chapter has been published [145] or is in print [148].

2.1 | Twitter: the crowd-sourced newswire

Early on, Allan et al. [9] hypothesised that understanding in its ideal form could trivialise the TDT task. Every event has a unique identity: Who did What, Where and When, the 'four Ws'. Two train accidents share a general vocabulary, but the specifics—the 'four Ws'—distinguish them [139; 289]. A system that understands events recognises that one event shares its identity with the other. Inversely, a system that understands events recognises a new event because it shares its identity with no other. Understanding the event identity symbolised the research area's hope of solving its early problems.

From a certain perspective, TDT research's problems were not of its own making. Back then, algorithms would collect reports from the news media and group them into events, so the research community naturally gravitated towards clustering. In clustering, or document-pivot approaches, literature found simplicity and a well-researched task but also several challenges, namely fragmentation and cumbersome parameter-tuning [5; 72]. If understanding could solve clustering's problems, the research community reasoned, it would solve TDT's own.

Yet Allan et al. [9] feared understanding would never suffice, and the early experiments to improve clustering through understanding vindicated those fears. At its worst, understanding failed. The event profiles by Makkonen et al. [139] mirrored the event identity proposed by Allan et al. [9]—one sub-vector for each of the 'four Ws'—but it performed worse than a traditional baseline, as did the similar effort by Kumaran and Allan [120]. And at its best, understanding barely improved results [169]. Within a few years, what Allan et al. [9] had predicted came true: limited understanding yielded limited improvements.

The failures had, at least, the merit of spurring the creation of feature-pivot techniques. Feature-pivot techniques do not understand the idiosyncrasies of specific events but how events behave in general: when an event happens, news outlets publish more reports or adapt their language [288]. To measure those changes, Fung et al. [72] formalised feature-pivot approaches, inspired by the earlier concept of burst [113], an elegant solution to measure spikes in the number of published articles or in the use of some keywords. Because burst made sense, feature-pivot approaches quickly became the popular alternative to clustering, the document-pivot approach.

Feature-pivot approaches overcame clustering's challenges without understanding, but they created new ones. Analyses on term correlations can be misleading [5; 93], and individual keywords do not tell a story like a news article or even a headline do. To compensate, feature-pivot approaches normally follow term extraction with term clustering to provide context [5], but groups of terms are not always expressive either. Do the terms *defeat*, *Republicans* and *Democrats* mean that the Republicans defeated the Democrats, or the opposite? Therefore by the time Twitter launched, researchers had neither explored understanding nor overcome TDT's challenges.

Twitter's launch in 2006 revolutionised news dissemination. Late in 2010, Twitter played a prominent role in chronicling the Arab Spring [110; 134]. In 2011, Twitter broke the news of Osama Bin Laden's death [98], and in 2013, it broke the news of the Boston Marathon bombing [220; 245]. Even the British Royal Family first announced the death of Queen Elizabeth II in a tweet [248; 256]. Reuters estimates that between 10% and 20% of all news breaks on Twitter first [130]. For many, the social network became a key source of news [257].

Twitter's launch revolutionised TDT too. Before Twitter launched, algorithms would wait for the news media to publish reports and then cluster them to form events; TDT was a mere aggregator. After Twitter launched, it made amateur reporters of regular internet users and transformed the social network into a crowd-sourced newswire. More importantly for the research area, Twitter made most of its content available for free through its API. Now, algorithms could chase the breaking news themselves instead

of repeating after the media, sparking many new applications for event tracking. TDT literature had its new medium of choice [201].

Nothing changed TDT research quite like Twitter did. Literature progressed from unspecified event detection, or identifying breaking news from general streams, to specified event detection, or building timelines of specific, often planned events [62]. Since Zhao et al. [296] first built timelines of American football games, research has followed sports events [162], monitored protests [95] and mapped natural disasters [61], among many other applications.

Yet accepting these new possibilities meant accepting the new challenges [26; 49; 86; 99; 102], of which Twitter created plenty. Twitter mixed news with fake news, quality with noise, and facts with opinions. To someone unfamiliar with Twitter, the social network's challenges may best be verbalised in the blunt words of Paul Doyle, a football writer for the Guardian who we interviewed in our previous work [141]:

> People are ... more extreme on Twitter. It is a forum where people are deliberately seeking attention. If you're looking at accuracy, it's not interesting. I consider Twitter to be the "toilet walls" of the twenty-first century. People would write graffiti on toilet walls to make their opinions known. Now, they use Twitter.
>
> — Paul Doyle for Mamo [141]

Twitter dwarfed earlier concerns among TDT researchers with three new formidable challenges. First, Twitter added volume and velocity. As early as in 2011, Twitter users published more than 50 tweets per second during popular events, like the Superbowl [296]. By 2015, Twitter generated 500 million tweets per day, or almost 6,000 tweets per second [258]. The Twitter API's free-tier limits the stream to 50 tweets per second, but even smaller numbers require prohibitively-heavy processing in real-time systems [62; 191; 225].

Volume and velocity affect document-pivot approaches the worst. Clustering algorithms ceaselessly compare documents with clusters: the more tweets, the more comparisons. In the context of Twitter's voluminous tweets, Panagiotou et al. [191] argued that clustering becomes outright infeasible. Document-pivot approaches did survive Twitter after all but only in the form of on-line methods [142; 146; 158].

Second, Twitter added brevity. At launch, the social network propelled the concept of microblogging into the mainstream. Twitter's distinctively-short 140-character tweets, later expanded to 280 characters, would not even fit this paragraph. Brevity robbed microblogs of the expressiveness of news reports [41; 228]. Donald Trump's in-

Figure 2.1: Twitter's brevity robs tweets of context. Humans, however, need little context. With the heavily-publicised 2020 US presidential election in the backdrop, Donald Trump's message could fit in a three-word tweet.

famous tweet, shown in Figure 2.1, reads like incoherent rambling especially because it omits the controversial context surrounding the 2020 US election. Therefore TDT methods on Twitter rely on groups of tweets for context [64].

Brevity does not only pose challenges, however. It lets users react more quickly and forces them to be more selective with words, to concentrate tweets around a few topical keywords [287]. Donald Trump needed just three words to deliver a clear message to those who understood the context: *stop* and *count* carried the narrative on their own. In TDT, Choudhury and Breslin [42] exploited the same brevity as they trained a classifier for cricket games. The classifier ignores most words in tweets and instead uses only the few that appear in a list of cricket terms and players, a form of understanding compiled manually by the authors.

Yet brevity led to sparsity, and sparsity harmed even established IR methods. The Term Frequency-Inverse Document Frequency (TF-IDF) term-weighting scheme, for example, was developed for longer documents than tweets [146; 153; 228; 260]. Unlike in formal news reports, words rarely repeat in tweets, effectively transforming TF-IDF into Document Frequency-Inverse Document Frequency (DF-IDF). Ironically, TF-IDF on Twitter promotes rare terms, not frequent terms [153]. Later, Saeed et al. [226] argued that the static design of TF-IDF cannot capture the dynamic nature of events.

Third, Twitter added noise. For many Twitter users, the rules of grammar and orthography exist as optional constructs, or so they seem. Researchers could no longer assume that the IR techniques that worked on formal news reports would work on tweets [41; 191; 225; 298]. Mishra and Diesner [166], for example, proposed an NER algorithm tailored to Twitter's orthography; tellingly, the solution hinges on gazetteers of names, organisations and locations, another type of understanding.

Neither could TDT research assume that every tweet was newsworthy [99]. Noise

appears in the subject matter of tweets too. Twitter users talk about their daily life, react to news and share opinions [99; 206; 264], sometimes in the same tweet. Moreover, the noise only seems to be increasing. Meladianos et al. [161] compared tweet datasets from 2014 with other datasets by Nichols et al. [183] from 2010 and noted a greater presence of noise.

But what is noise if not the algorithmic failure to distinguish between what is relevant and irrelevant, what is objective and subjective, and what is important and what is trivial? If humans can recognise noise, spam and advertising effortlessly, why cannot machines do the same? Does the difference not lie in understanding?

Over the years, some researchers endeavoured to understand Twitter streams better. Hasan et al. [93] manually compiled a list of 350 phrases related to spam, like *click here*, to remove noisy tweets, and Kolajo et al. [114] used knowledge bases of slang, acronyms and abbreviations. Some even involved simple types of event understanding. Hossny and Mitchell [95], Hua et al. [99] and Zhou et al. [297] all filtered off-topic tweets with automatically-extracted news keywords.

Such efforts appear scarcely, however. TDT research rarely ever built on understanding [26; 49; 94; 137], preferring simpler alternatives instead. Researchers pummelled Twitter's best virtue, its large volume, by aggressively filtering all retweets [28; 55; 101; 102; 226]. They justified themselves by arguing that retweets introduce bias and redundancy [158; 225], but the effects can be staggering. McMinn and Jose [158] removed all retweets and any tweet without a named entity: 95% of the dataset.

Aggressive filtering appears in another form in document-pivot techniques. Some research retains only the largest clusters [102; 197; 264] or clusters exceeding a certain size because they are more likely to be newsworthy. Our previous algorithms accept clusters with three or more tweets [142; 146], but others adopt far more conservative limits; Hasan et al. [93], Ifrim et al. [102], and McMinn and Jose [158] accept no cluster with fewer than 10 tweets, and McMinn et al. [157] only accept clusters with at least 30 tweets. Ozdikis et al. [189] set the threshold at 250.

Hampered by the new challenges, progress stalled. Evidently, AI witnessed technological advances, which the research community adopted. Researchers applied machine learning [71], text embedding [61] and, more generally, increasingly-convoluted solutions. Yet the performance woes continue to torment TDT, now compounded by Twitter and the complex needs of contemporary applications.

If performance improved, it did not improve substantially. We draw parallels between the algorithm by Zhao et al. [296] from 2011 and in ours from 2021 [146]. The idea of building a timeline of an unpopular event seems delusional [93; 225]. What remains if the system filters 95% of all tweets from an unpopular event? How many top-

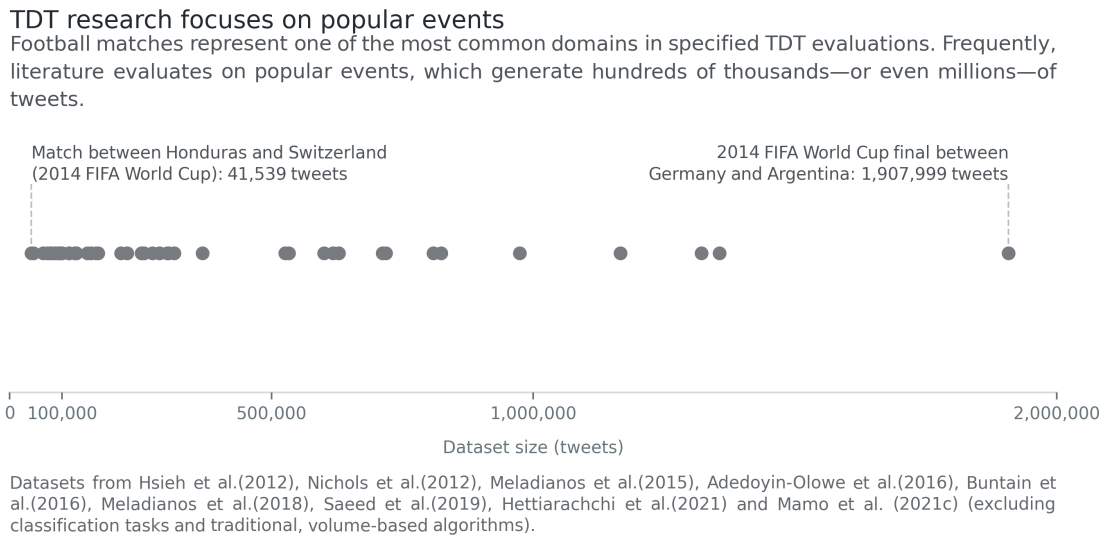

Figure 2.2: TDT publications generally evaluate on large datasets from popular events. Excluding classification tasks and the early volume-based techniques, datasets range from a few tens of thousands of tweets to millions.

ics in unpopular events form clusters with 10 or more tweets? Excluding classification tasks and trivial algorithms, our surveyed research in the domain of football matches required datasets with at least 40,000 tweets.[1] The median, however, lies 6 times higher, at around 240,000 tweets.

To follow unpopular events, TDT research must compromise. van Oorschot et al. [265] trained a classifier limited to detecting a few football topics. Löchtefeld et al. [136] provided a manually-curated knowledge base related to football, which included extraction patterns and a list of team and player names from the German Bundesliga. Otherwise, what remains are trivial algorithms like those by Zhao et al. [296] or Lanagan and Smeaton [121]: inexpressive methods that only identify periods with a high activity.

Yet algorithms struggle even in popular events. In football matches, TDT algorithms complain about capturing key topics, like goals, but missing many more non-key topics, like yellow cards, which evoke less interest [79; 121; 136; 146; 150; 183; 265]. Even Meladianos et al. [162], with their massive datasets, suffer from the same fate: they miss

[1] Figure 2.2 and Figure 5.2 on page 111 exclude classification tasks and simple algorithms. Classification limits the scope of TDT to a few, usually easily-enumerable topics, such as goals in football matches, and excludes difficult-to-enumerate topics, like missed chances and other exceptional developments. Simple algorithms, which characterised early research, have no such limits and operate even on a few hundred tweets [121]. However, as we discuss in Chapter 5, their performance and outputs no longer meet the needs of modern TDT.

39 topics, of which 36 yellow cards.

Lack of interest does not excuse TDT methods' failure to capture non-key topics. Non-key topics occur frequently and possibly with newsworthy consequences [136; 266]. A foul may lead to a free-kick or a penalty, which in turn may lead to a goal. The narrative feels incomplete without the non-key topics filling the spaces among key incidents. We cannot justify the failure of sophisticated algorithms to detect non-key topics even from massive datasets.

In the past, we attempted to overcome some of the challenges. Finding Important News REports (FIRE) [142], a batched algorithm, combined document-pivot and feature-pivot techniques to detect topics. Conceptually, the document-pivot or clustering approach detected candidate topics, while the feature-pivot technique confirmed whether users were discussing them like breaking news. Later, Event TimeLine Detection (ELD) [146] refined FIRE's formula with a real-time process and a novel feature-pivot technique.

The two algorithms achieved mitigated success. FIRE considered clusters with as few as three tweets and often detected breaking news around the world quicker than the news media [142]. ELD stretched the capabilities of clustering to create timelines far more granular than those by Zhao et al. [296] from datasets with as little as 64,000 tweets [146]. Yet a practical limit remains, not much lower than 64,000, as we show in Appendix B. Furthermore, while the combination of document-pivot and feature-pivot techniques improved precision and recall, ELD too fell to old vices, capturing noise and missing many non-key topics.

It is tempting to reflect on past failures with clemency. We could accept the flaws as part of Twitter's character, a worthwhile sacrifice in the name of TDT research's newfound purpose. The failures, however, reveal something far simpler: that basic tweaks to document-pivot and feature-pivot techniques cannot overcome the research area's challenges. Researchers need to accept volume and velocity, brevity and noise as obstacles without justifying the faults in their algorithms, and then examine more fundamental solutions, like understanding. Only then may TDT research progress again.

Recently, some researchers have started to evoke understanding again. This time, understanding does not merely represent an avenue worth exploring, as Allan et al. [9] suggested, but a necessity. Bontcheva and Rout [26] proposed understanding as a way to make sense of social media. De Boom et al. [49] advocated for knowledge to drive event detection, and Panagiotou et al. [191] argued that algorithms should not only detect but also describe Who did What, Where and When. Two decades after Allan et al. [9] first mooted understanding, the research community seems willing, again, to explore event knowledge as a solution to TDT's problems. First, however, it needs to

resolve what it means to understand.

2.2 | What we understand by understanding

Liverpool had been leading Leeds for almost an hour—a fragile, 1-0 lead—when Diego Llorente scored the equaliser. With an air of resignation, Liverpool's community manager only tweeted a few numbers, a hashtag, and the two 1-word sentences in Figure 2.3: "Goal. Leeds." Two 1-word sentences sufficed. Liverpool supporters knew exactly what the community manager and the goal meant.

A little understanding goes a long way for humans. A few well-chosen keywords, in no particular order, give us enough context to grasp the substance of a story [96]. Despite its brevity, Donald Trump's tweet in Figure 2.1 became an internet sensation, and although Liverpool's tweet had no such fortune, it is expressive precisely because the choice of words is so plain. Far from being ambiguous, the two sentences, short even by Twitter's standards, make the messages poignantly incisive.

Machines understand differently. We understand Liverpool's tweet because it is direct and unambiguous but still simple in its use of common football language. We also understand the contextual cues: the tweeting account and the knowledge that Liverpool were playing against Leeds. In contrast, algorithms do not understand semantics like we do, so they do not distinguish between words like *bench* and *goal*, and process them similarly. Machines do not really understand events despite understanding having accompanied TDT research since its early days.

Of course, TDT algorithms generate simple forms of understanding. At minimum, algorithms understand When something happens, and all but the most trivial of techniques understand topics in more detail: document-pivot approaches understand what users are discussing and feature-pivot approaches understand how. Yet neither let understanding drive the process [49], so they generate limited knowledge.

More practically, the generated understanding has limited practical use. None of the knowledge about particular events or topics transfers to other events. A machine may detect a goal in one football match, but it does not understand the meaning or significance of goals, nor of the term *goal*. Therefore the machine cannot apply that knowledge in other football matches. In this dissertation, we do not focus on the expendable understanding that emerges naturally from the TDT process but on the type of transferable understanding that can drive event tracking.

The TDT community does not employ understanding in its algorithms because it has none. The research area still does not really understand what it means to gener-

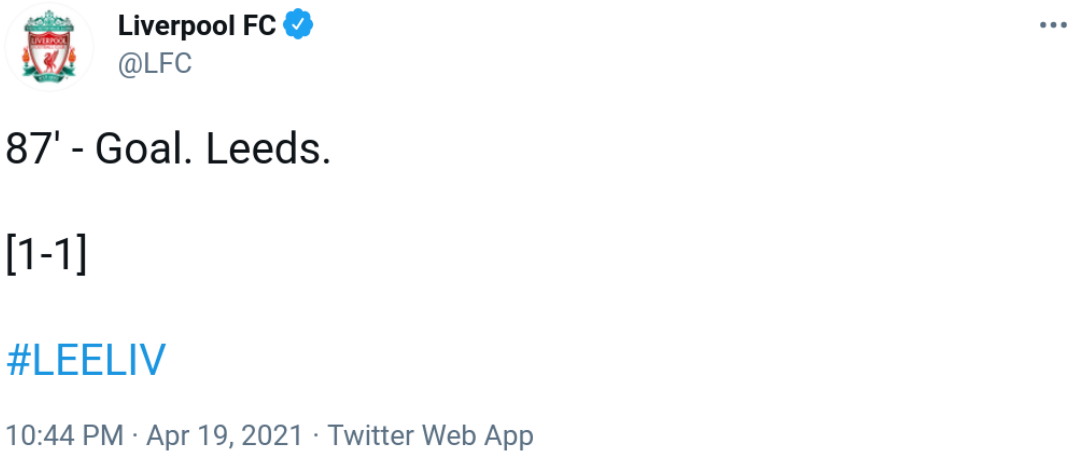

Figure 2.3: Tweets can be expressive even if they are short as long as the reader understands the context. Liverpool's tweet from their match against Leeds included all the necessary information: When the incident happened, What happened and Who made it happen, the scoreline and the event hashtag.

ate knowledge about events, even though interpretations abound. From the theoretical reflection on the event identity by Allan et al. [9] to more practical perspectives, researchers have interpreted understanding differently, but they never reached a consensus. Furthermore, most interpretations do not arise from explicit study but from indirect and casual experiments into understanding.

The earliest type of understanding is also the most deliberate: structured or theoretical understanding. Every event "comprises at the very least what happened, where it happened, when it happened, and who was involved" [169]. The event structure proposed by Allan et al. [9] formalises those components in one framework, 'the five Ws and one H': Who did What, Where and When, and Why and How.[2] Above, Liverpool's community manager afforded such brevity because the tweet describes the Who, What, and When: "87' [When] - Goal [What]. Leeds [Who]."; the Where, unchanging throughout the event, is redundant.

Today, the 'four Ws' have all but disappeared from event tracking, but they are experiencing a quiet renaissance elsewhere. While TDT researchers afforded to track events without a theoretical structure, those in EMM did not. Event modelling research assumed the task of representing "in a semantically meaningful way" [39] the events that TDT research could not understand. In the 'four Ws', the event modelling community

[2]In practice, TDT literature focuses only on the Who, What, Where and When, or the 'four Ws' [139; 169]. The Why and How denote more complex reasoning, a task for event mining [39].

found its semantic structure.

Yet in event tracking like in event modelling, the event structure does not constitute understanding. Framed in Plato's definition of knowledge, Definition 1, the 'four Ws' fail. The structure contains no propositions: no justifications, no truths and no beliefs—only a template for event understanding. Researchers have to interpret for themselves how to fill in the 'four Ws', with difficulty. The When alludes to time, but researchers had to rationalise what it means to understand What happens in an event, with Whose involvement and Where. The TDT community frequently resorted to linguistics to fill in the structure. Less commonly, it prioritised semantics.

Linguistic understanding

First, the TDT community understood with linguistics. In linguistics, researchers found convenience and accessibility. They found that they could trial understanding effortlessly with existing Natural Language Processing (NLP) tools. The When aligned with the publication time, the Who and the Where with NER's named entities, and the What with most of what remained. Thus, linguistics helped fill in some of the gaps in the theory of the 'four Ws'.

Some research modelled events from news articles in this manner. Makkonen et al. [139] built event profiles entirely with linguistics: NER tools to understand the Who and the Where, and Parts of Speech (POS) tagging to understand the What. More contemporary applications augment the old structure with modern metadata. The event models of Li et al. [126], otherwise identical to event profiles [139], consider Twitter mentions as the Who and hashtags as the topics, the What.

Others studied certain aspects of the event structure. Chen and Ku [35] extracted named entities, the Who and the Where, and boosted their weight, whereas McMinn and Jose [158] processed tweets separately depending on the named entities within. Others yet left behind the event structure altogether and gave understanding a more sophisticated but withdrawn role. They interpreted understanding events as discovering the latent subjects in streams with topic modelling [49], or simply comprehending language better with synonymy [114; 138] or text embedding [61; 94].

Linguistic understanding failed frequently, however. The event profiles built by Makkonen et al. [139] performed worse than the baseline, as did those by Kumaran and Allan [120], and for Li et al. [125], performance only improved after they filtered the profiles semantically. Likewise, topic models [49] and named entity boosting [10; 35] only reduced errors when applied selectively. Even when linguistics succeeded, the progress appeared insignificant next to the efforts and complexity of understanding [169].

Linguistics failed because they did not capture true event understanding. They represent the assumption that if TDT algorithms understood language better, they could track events better. To a certain extent, the assumption makes sense. So much of TDT's process depends on language: document-pivot techniques cluster documents that use the same language and feature-pivot techniques monitor changes in language. Nevertheless, the assumption represents only a weak belief, that linguistic understanding could approximate event understanding. Knowledge rests not only on belief but on justification and true propositions too, and linguistics contributed little of either.

Semantic understanding

Second, the TDT community understood with semantics. On occasion, researchers recognised that human understanding of events has a more refined nature than simple parsed language. Like the two words that describe Leeds' goal against Liverpool, TDT research needed to understand, precisely, events. Machines do not require named entities or nouns or verbs; they require the named entities and nouns and verbs that describe events semantically.

Semantic understanding does not have to fall far from linguistic understanding. Nouns like *goal*, *cross* and *foul* comprise the vast majority of terms in the lexicon by Kubo et al. [118], which they use for event summarisation. Likewise, the lexicon by Choudhury and Breslin [42] contains nouns and named entities relevant to cricket games. Differently from linguistics, however, you could pore over every noun and every named entity, and find an inextricable link with the event and its domain.

Thus, semantic understanding refines linguistics. Hua et al. [99] rank a POS tagger's words based on their affinity with the domain, and Hossny and Mitchell [95] extract protest-related words with a method similar to feature selection. Elsewhere, Huang et al. [101], who build separate timelines for each named entity in an event, seek to understand participants better: they discard infrequent named entities and identify coreferences. Each approach uses but does not depend on linguistics.

Notwithstanding such examples, semantic understanding seldom figures in TDT literature. Like Kubo et al. [118] or Choudhury and Breslin [42], researchers could define semantic understanding manually to capture true event knowledge. Buntain et al. [28] mulled the possibility but found the manual process infeasible, unreasonable and unscalable. Moreover, ready-made knowledge bases rarely cater to the specific needs of particular domains either [244]; WordNet lists 16 different senses of the word *cross*, which Kubo et al. [118] included in their lexicon, but not one relates to football.

Buntain et al. [28] never considered generating semantic understanding automati-

cally. Perhaps the task appeared formidable without any guarantees. When automatic semantic understanding appeared, it followed the failures of linguistic understanding, such as when Chen and Ku [35] filtered trivial named entities, or when Li et al. [125] built event profiles using only event clauses. Such research recognised that a few precise terms distinguish one story from another [8]. The difficulty, however, lies in identifying which [139] and rationalising the same logic to extract them algorithmically. So far, the problem remains without a solution.

Event models

Third, the TDT community understood with event models. Event models represent the culmination of semantic understanding. Semantic understanding comprehends only What happens, with Whose involvement or Where; event models combine all of the 'four Ws' in one structure that describes how one relates with the other. Unlike the isolated terms in the lexicon by Kubo et al. [118] or the detached named entities extracted by Huang et al. [101], the event model may link What happens with Who makes it happen. Thus, event models fill the event structure with semantics.

Löchtefeld et al. [136] present the best example of event models in TDT literature. A football knowledge base stores the names of German teams and players, the Who, and a set of hand-crafted extraction patterns that capture Who does What in matches. Whenever a pattern matches a few tweets, the system creates an event model, such as which player scored a goal. Machine-readable understanding begets machine-readable understanding. Yet the process implies not only the difficulties of understanding semantically but also the new difficulties of understanding how events function, and the example by Löchtefeld et al. [136] may well be the only one in TDT literature.

Event models appear predominantly elsewhere, in EMM research. Naturally, event modelling's scope differs. TDT understands retrospective or real-time events, whereas event modelling understands only past events, the outputs of event extraction or tracking algorithms that have concluded. The event models thus arrive far too late to drive the TDT process. In other words, they still leave the TDT community needing to understand events for itself.

The interpretations of event understanding leave an almost-unreconcilable divide. At one end, automatic understanding, like POS tagging or NER [35; 139], is convenient and accessible but ambiguous and inaccurate. At the other end, manually-compiled understanding [118; 136] is laborious but unambiguous and accurate. The challenge lies

Figure 2.4: Journalists rely on the 'five Ws and one H' to describe events. The BBC's tweet describes Who is doing What, Where, and Why and How; the publication time implies the When.

not only in generating understanding automatically but rather in generating the type of reliable, semantic knowledge around which researchers could design TDT algorithms. In the next section, we describe how research can start closing the gap.

2.3 | Semantics as a solution

While the TDT community struggled to understand events, the news media had mastered them. For centuries, journalists and reporters in the news media have had to understand events to report about them to readers and listeners. A skilled journalist can weave a narrative in just one sentence using nothing more than the tools that Allan et al. [9] proposed: the 'five Ws and one H'.

The 'five Ws and one H' represent a fundamental rule of journalism [222]. They orient readers and listeners towards the essence of a story [105]. The BBC's tweet in Figure 2.4 gives the background and an update about a story at once: "Indonesian

Navy [Who] hunting for submarine [What] that has gone missing [Why] in waters north of island of Bali [Where]". Liverpool's tweet in Figure 2.3 achieves the same effect with even more pronounced minimalism.

Because the 'five Ws and one H' make sense, we adopt them as our framework for understanding. Historically, the major obstacle to understand events has been to identify important information, what it means to understand [137; 169], but the event structure serves as a theoretical guideline. The 'five Ws and one H' orient our understanding like they orient the news media's readers and listeners.

Of course, past failures and a low adoption tarnish the 'five Ws and one H'. Nevertheless, history portrays these simple tools as theoretically-sound; only in TDT literature do they appear practically-shallow. Our position in this dissertation is that TDT research failed the 'five Ws and one H', and not the other way round. Research understood language, not events. Differently from prevailing literature, in this dissertation we fill the event structure with semantic understanding.

In practice, our vision of event knowledge resembles the lexicon of terms that Kubo et al. [118] wrote or the knowledge base of participants that Löchtefeld et al. [136] compiled: semantic, precise, human-like. Nevertheless, while we generate different types of understanding, our end-goal represents only an elementary form of event models. We propose to understand, semantically, What may happen in an event, or Who may be involved and Where, but not how one relates to the other. We explore, briefly, event models in Chapter 6.

We impose two conditions on how we must fill the event structure with semantic understanding. First, we must generate understanding automatically, which requires us to formalise what it means to understand What may happen, or Who may perform an action or Where. We refer back to the definition of events, Definition 2 on page 4 [39]. Second, we must generate understanding early enough to drive the TDT process. Understanding retrospectively, like EMM research does, excludes the application of knowledge in real-time events. Therefore we aspire to understand events semantically, automatically and ahead of time.

Of the 'five Ws and one H', we focus on the 'four Ws'. In particular, we look to understand the event participants, the Who and the Where, and the actions or changes, the What; the primary TDT task, to detect new events, implies the When, whereas the Why and the How demand complex reasoning [51], and thus fit better as problems for event mining [39]. Table 2.1 lists our interpretations of the 'four Ws'. In the rest of this section, we discuss the significance of the Who and the Where, and the What, and explain how they can make sense in practice, not just in theory.

Who	"The entities that play a significant role in shaping the event progress" [101; 234]; the persons, locations or organisations who affect or are affected by the event [144]
What	"An action, or a series of actions, or a change" [39]; "the subject, occasion, body or activity that [is] involved in the event" [169]
Where	The places or locations where the event takes place [169]
When	The date and time when the event takes place [169]

Table 2.1: Selected interpretations from literature of the four basic elements of an event: Who did What, Where and When.

The Who and the Where

No form of understanding appears as commonly as the Who and the Where in TDT literature. Early on, some researchers perceived a semantic value in persons, organisations and locations that distinguished one event from the other. Therefore research boosted the weights of named entities [5; 35], or separated them from ordinary words to form event-centric adaptations of the classical, undiscerning Vector Space Model (VSM): event profiles [120; 139; 285].

Others reasoned that events primarily concern the Who and the Where [158] and gave participants a role to match. McMinn and Jose [158] cluster tweets separately, depending on which names appear within. Likewise, the algorithms by Shen et al. [234] and Huang et al. [101] follow participants to build individual timelines for each named entity and its coreferences. Such thinking refined the philosophy of events, gave timelines a new sense. A timeline no longer symbolised a list of topics but a list of topics related to a certain named entity.

The prevalence of the Who and the Where in TDT literature leads to two reflections. First, it reflects the driving role that named entities assume in events [158; 169]; ignore them, research found, and performance drops significantly [10]. Second, and perhaps more consequential to how the research community chose to understand, it reflects the ease with which researchers could extract participants. The Who and the Where symbolise persons, organisations and locations—named entities. Existing NER literature gave researchers a rare relief from having to define the Who and the Where, and the tools with which to identify them.

Regrettably, research never progressed past NER. Had research investigated the Who and the Where more closely, they might have comprehended why only discriminating named entities helped Chen and Ku [35] improve performance. Chen and Ku [35] discovered what Zhou et al. [297] remarked later: named entities do not necessarily

indicate newsworthiness. Although the Who and the Where manifest as named entities, not all named entities represent the Who or the Where. At fault, our assumptions: NER tools understand language, not events.

Because NER models understand language, not events, they cannot discern between named entities and event participants. On 26 July 2020, as Leicester City's Jamie Vardy played against Manchester United, he was simultaneously competing for the Premier League Golden Boot against Arsenal's Pierre-Emerick Aubameyang and Southampton's Danny Ings [237]. Twitter users mentioned Aubameyang and Ings alongside Vardy, despite having no bearing on the match itself. All three were named entities, but only Vardy was a participant. Yet TDT research still assumes, erroneously, that named entities could substitute for participants.

In our previous work, we revised the assumption [144]. We assumed that at least some named entities represent participants, and that they can lead us to the other participants. The new assumption formed the basis of Automatic Participant Detection (APD), a six-step framework to understand Who participates in events, or Where they happen. In the first three steps, APD still relies on NER to identify, score and filter infrequent named entities. In the next steps, however, the framework refines NER's output: it disambiguates named entities to retain only the relevant participants and extrapolates from them the ones it missed.

APD reconciled NER with proper event understanding of the Who and the Where. NER tools, whether designed for Twitter or not, succumbed to the same challenges: they understood language, not events. In contrast, our first iteration of the APD framework correctly filtered many irrelevant named entities and captured many more relevant participants that NER models had missed, even before the event started [144]. APD helped us understand events, not language. We describe DEPICT, a novel algorithm to understand the Who and the Where, in Chapter 3.

The What

The TDT community's biggest failure to fill the event structure with semantics was the What. Like for the Who and the Where, researchers looked for readily-available solutions to understand the What. Algorithms applied topic modelling to discover latent topics [49] and POS tagging to understand What happens in events [35; 120; 139; 169]. This time, however, the obvious flaws of the linguistic assumptions could not escape researchers. The lexicon constructed manually by Kubo et al. [118] contrasts sharply with the crudity of linguistic understanding.

The What should have a significance in the event. It should describe "the subject,

occasion, body or activity that *[is]* involved in the event" [169], or "an action, or a series of actions, or a change" [39]. What understanding do linguistics impart? Makkonen et al. [139] understood What happens in events simply through the "subjects, objects, attributive nominals, prepositional complements and main verbs"—linguistic components. POS tagging only understands the syntax of a language, and synonymy and text embedding only the semantics of language, not events.

In a way, the absence of definitions impelled researchers to rely on linguistics. While the What is intuitive for a human to understand, it proves far more challenging to define [267; 282]. We read Liverpool's tweet and sense the disappointment of a late equaliser, and we read the BBC's tweet and sense tragedy. Machines read words and sense nothing. Nevertheless, understanding What happens in events remains a crucial aspect of event knowledge. No other component of the 'four Ws' describes events as expressively as the What [169].

We again revise the assumption of linguistics. Like Hua et al. [99] and others [66; 289], we assume that a few keywords, normally the Who and the Where, separate events in the same domain; Hua et al. [99] called them "event terms". Fundamentally, events belong to the same domain because they share a general vocabulary [161], the domain terms: a building *collapses* in an earthquake, a political party *wins* an election, a player *scores* in a football match. We assume that domain terms describe What happens in events from a particular domain.

Our assumption aligns the process of understanding What happens in events with Automatic Term Extraction (ATE), the task of identifying domain-specific terms from domain-specific corpora. Like TDT's algorithms, ATE methods generally have a linguistic component and often rest on nouns. Unlike TDT's algorithms, however, ATE methods only rest on the nouns that describe domains. A statistical component, the termhood measure, follows the linguistic component and weighs a word's suitability as a term [140]. Thus, ATE can become the answer for TDT research to understand What happens generally in events from a domain.

The TDT community has experimented with ATE only briefly, never explicitly, in the work we have surveyed. There are no sophisticated termhood measures tailored to events or tweets—only elementary techniques. Hua et al. [99] and Zhou et al. [297, 298], the closest research we could find to ATE in TDT literature, used the most rudimentary of the research area's baselines: TF-IDF and derivations of it. Hossny and Mitchell [95] simply applied feature selection with Jaccard similarity to the problem. None of the surveyed approaches even evaluated the quality of the understanding—only its effects on the system.

Elementary ATE techniques may not satisfy TDT. To understand What happens in

events, event tracking research must adapt, not merely adopt, ATE, for two reasons. First, while ATE literature focusing on general domains abounds, we could find little research on event domains. TDT literature needs domain terms to reflect, accurately, "an action, or a series of actions, or a change" [39]. As our findings in Chapter 5 suggest, ATE's general domain terms, much like linguistics, may not suffice.

Second, we could find even less ATE research on Twitter. Hua et al. [99] tracked events from tweets but understood the What from news articles, and so did Zhou et al. [297, 298]. In fact, among the literature that we reviewed, only Hossny and Mitchell [95] understood What happens in event domains from tweets, and they approached the task as a feature selection problem. If TDT algorithms track events on Twitter, then surely they should understand them like Twitter does.

Thus, ATE's performance and challenges in event domains and on Twitter remain a great unknown. TDT research on Twitter may not understand events as long as ATE algorithms do not understand events from tweets. Like we adapted NER to understand the Who and the Where, event tracking research must also adapt ATE to understand the What. We describe EVATE, a novel algorithm to understand What happens in events from tweets, in Chapter 4.

Recap

Allan et al. [9] first set TDT on the road to understanding in 1998, but more than two decades later, the research community still has barely embarked on the journey. The reluctance does not mean that the community has solved the challenges that motivated the original suggestion. On the contrary, TDT literature faces new challenges for which research has no answer. In this chapter, we revived the proposal to study understanding by answering the following questions:

- Why does TDT need understanding? For years, research has grappled helplessly with the limitations of document-pivot and feature-pivot techniques, and later with Twitter's challenges. In Section 2.1 we explored TDT literature's difficulties and showed how event understanding can overcome some of modern day's most persistent challenges, including Twitter's volume and velocity, brevity and noise.

- What does it mean to understand events? From the event structure to linguistic understanding and more sophisticated, semantic knowledge, TDT research has explored understanding, albeit often indirectly. In Section 2.2, we compared per-

Principal contributions

- The first review dedicated to the development and application of understanding in TDT literature
- A critical discussion of different forms of understanding that links TDT with EMM and the news media
- Practical guidance to understand the Who and the Where with APD, and the What with ATE

spectives of understanding and discussed why linguistics could not match a semantic understanding of events.

- How can TDT literature understand events? Without direction from the 'four Ws', the research community approximated understanding through linguistics, which could never suffice. In Section 2.3, we adopted the 'four Ws' as our definition of events, and we linked the Who and the Where with APD, and the What with ATE to develop semantic understanding automatically.

In the rest of this dissertation, we develop our vision of event understanding. In Chapter 3, we understand the Who and the Where with APD, and in Chapter 4 we understand the What with ATE. Then, in Chapters 5 and 6, we apply our understanding in TDT and EMM. We start by exploring how APD can help us understand, semantically, Who participates in events and Where in the next chapter.

3

Understanding
The Who and the Where

McMinn and Jose [158] hypothesised that events primarily concern participants. More precisely, McMinn and Jose [158] hypothesised that named entities form "the building blocks of events": events happen because named entities make them happen or, by happening, events involve or affect named entities. Others formulated more measured hypotheses. Yang et al. [289] and Popescu et al. [205] recognised the value of participants and hypothesised simply that named entities could substitute for them. Each treated named entities as a window into events: understand named entities and you will have understood events themselves.

From such practical hypotheses flowed the TDT community's first understanding of events, the Who and the Where. The hypotheses by McMinn and Jose [158], Yang et al. [289] and Popescu et al. [205] seem to flow logically, without any apparent flaws. In practice, however, the hypotheses proved incomplete and ingenuous, and ultimately detrimental. Understanding about the Who and the Where only harmed its various applications [10; 35; 120; 125; 139]. The first taste of understanding exposed the dangers of understanding too simply.

In this chapter, we explore literature's first form of understanding, the Who and the Where. We demonstrate how the TDT community erred in conflating named entities with participants, and how the assumption ultimately let down their applications. More importantly, we also demonstrate how the research area could understand participants with Automatic Participant Detection (APD). We demonstrate all by answering the following questions:

- What makes a named entity an event participant? The idea that named entities

could represent event participants has a logical basis, but Named Entity Recognition (NER) tools on their own often worsened performance. In Section 3.1, we demonstrate how named entities failed TDT algorithms primarily because the fundamental assumption failed: few named entities could substitute for participants.

- How can a better understanding of participants improve our understanding of Who participates in events and Where? The TDT community expected named entities to be a quick way of understanding the Who and the Where, and the efforts matched the expectations. In Section 3.2, we present DEPICT, a novel APD algorithm that follows a more circuitous route to understand, not merely detect, participants.

- How does APD refine the NER assumption, that named entities could substitute for participants? The TDT community never studied the suitability of NER models to understand event participants; it only ever studied how named entities affected performance, often in the negative. In Section 3.3, we compare three NER models, including, for the first time, Twitter's own named entity annotator, with two APD algorithms.

3.1 | What makes a named entity an event participant

McMinn and Jose [158] made explicit what many others had previously only implied. Before McMinn and Jose [158], Popescu et al. [205] had similarly argued that events revolve around "a small set of important entities", as a general rule persons, locations or organisations. A decade earlier, Li et al. [125] went even further and posited, with excessive optimism, that knowledge of participants would make event tracking "easy and effective." The hypotheses seemed to flow logically.

Beneath the hypotheses' elegant veneer, however, lay a more practical matter. Understanding the Who and the Where represented an attractive prospect because the two proved the most accessible of the 'five Ws and one H'. TDT researchers could understand the Who and the Where with existing NER models. So began the rocky journey into understanding, with named entities the staple assumption. The research community rarely asked the critical question at the heart of the premise: what makes a named entity an event participant?

Without an answer, the assumption failed frequently and systematically. Few researchers measured the effects of named entities on their TDT algorithms explicitly, but when they did, flaws appeared with a telling consistency. First, named entity boost-

Model	Precision	Recall	MAP	Balance
NLTK	48.33%	31.46%	30.42%	0.3533
TwitterNER	48.00%	28.85%	34.76%	0.2829

Table 3.1: Deployed on six football matches, NLTK and TwitterNER could neither extract event participants comprehensively nor, more surprisingly, precisely. We present a full breakdown of the results in Table F.1.

ing harmed algorithms, which performed better in complete ignorance of named entities [10; 35]. More recently, Phuvipadawat and Murata [204] resurrected the idea of named entity boosting, this time on Twitter, and inspired many others [5; 102; 264]. The trend, however, appears to be a facade built only on an empirical study.

Then, event profiles failed. Early on, the TDT community repeatedly attempted to reconstruct the event structure [9], predominantly from named entities. Event profiles worsened performance as well. The efforts of Kumaran and Allan [120], Makkonen et al. [139] and Li et al. [125] could not exceed even elementary baselines. The event profiles extracted by Li et al. [125], built on named entities, made almost three times as many mistakes as TF-IDF with cosine similarity.

The difficulties to improve performance painted a distorted picture of understanding. We could, perhaps, trace the lack of semantic understanding in modern solutions back to these unsuccessful experiments. Regardless, the TDT community seems to have set aside named entities. Elsewhere, however, EMM research continued to insist on the assumption—on named entities. To Edouard et al. [56], named entities represent such a crucial element that no event could conceivably exist without the involvement of one. Yet others found little guarantee of newsworthiness in the presence of named entities [297].

To understand why the assumption failed TDT algorithms so regularly, we examined the output of two NER models. We used Natural Language Toolkit (NLTK)'s general-purpose NER model [22], and TwitterNER [166] to extract 50 named entities from six football match datasets, which we use in Section 5.3. As we annotated named entities following the same process as in Section 3.3, the causes of NER's failures in event tracking literature became clearer. We identified three problems in the results of Table 3.1.

First, we found that many participants never appeared in NER's rankings. We tracked all six football matches using not only the event hashtag but also the names of the stadium, teams, players and coaches. Still, on average, NLTK and TwitterNER could recall less than a third of participants. Instead, the two NER models played to the

whims of the events: they captured the participants with a prominent role and missed altogether the more numerous but withdrawn actors.

Second, and more surprisingly, we found that most named entities bore little relevance to the event. On average, we could label less than half of the named entities as clear mentions of participants. Instead, NLTK and TwitterNER filled the rankings with ambiguous references to participants, mostly common first names, and many other spurious named entities with a tangential relevance: the names of unrelated teams, players and other football personalities.

Third, the named entities described very selective versions of events, the ones told by Twitter users. The low balance figures indicate how both NER models succumbed to what we previously termed as bias [144], or the tendency for a few named entities to dominate discussion [101]. In the domain of football matches, one team often hogs attention, and the two NER models mirrored the asymmetric behaviour, covering one set of players disproportionately more than the other.

To summarise, TDT research's assumption failed our experiment on every count. The two NER models could neither capture participants precisely nor comprehensively nor symmetrically. And the flaws appeared consistently: no extractor and no event deviated considerably from the trend. Not even TwitterNER's better handling of Twitter's syntax led to statistically-significant gains over NLTK's general-purpose tool. It should come as no surprise, then, that named entities did not improve TDT algorithms. We will examine the difficulties of NER models to understand the Who and the Where in more detail in Section 3.3. For now, our failed experiment can serve as a valuable lesson.

Some inquisitive researchers drew the right conclusions. While many, like Makkonen et al. [139], lamented but accepted the flaws of understanding, others looked for the reasons why understanding failed to improve performance. Allan et al. [10] concluded that several named entities had little relevance to the event and that many others referred to the same participants, and thus created ambiguity. Chen and Ku [35] concluded, from a highly TDT-centric perspective, that many named entities have no discriminating power to distinguish one event from the others. Similarly, Li et al. [125] concluded that few of the named entities captured the essence of an event.

Because the three groups of researchers drew the right conclusions, they could shape NER's output to suit the needs of the TDT problem. Allan et al. [10] created a stopword list of named entities to remove noise and applied co-reference resolution to map references to a common entity, thus reducing ambiguity. Chen and Ku [35] only boosted the weight of the discriminating named entities, and Li et al. [125] constructed event profiles from the few named entities that appeared in important sentences, the event clauses. On these rare occasions, when the TDT community bridged linguistic understanding with

semantics, the methods' results improved.

These results reveal NER models to be useful tools with a limited scope. NER's scope has never been to capture all event participants and filter all irrelevant named entities. Even the ideal NER model could not distinguish between a named entity with a passive presence and a participant with an active role in an event—what Chen and Ku [35] called the discriminating named entities. NER's scope is simply to capture named entities. Therefore the problem of NER reminds us of the broader problem of event understanding: the problem is not so much the use of named entities as it is identifying which named entities can contribute positively [169].

The rare recent efforts to understand tend to be little more discerning of the outputs from NER tools than Allan et al. [10] and the rest. Shen et al. [234] and Huang et al. [101] resolved co-references and filtered infrequent named entities, but a stark dissonance remains between their definition and practical interpretation of participants.[1] Both define participants as "the entities that play a significant role in the event" but reduce significance to a contest of popularity. McMinn and Jose [158], who called participants the "building blocks of events", do not filter named entities at all.

If the Who and the Where carry so much importance in events; if participants constitute the foundation of events, then surely they warrant a more thorough understanding. Surely we must progress past the assumption that NER's linguistic understanding could substitute for semantic understanding.

In our previous work, we revised the assumption [144]. The TDT community assumes that named entities could substitute for participants. We assumed that certain named entities could substitute for participants, and that those named entities could lead us to the ones that the NER model missed. From our new assumption emerged, unwittingly, the present solution to understand the Who and the Where: the six-step Automatic Participant Detection (APD) framework, summarised in Figure 3.1.

Originally, we devised the APD framework to follow the query expansion principle [141]. We describe events with queries that humans could understand but which machines interpret literally: an event representation, normally a hashtag. If machines recognised the participants, however, they could automatically augment the queries, or the event representations. This simple act of expanding the query with a semantic understanding of the event's participants would ultimately improve TDT performance [141]. The crux, of course, lay in overcoming NER's limits: identifying which named entities could substitute for participants.

[1] We refer to Shen et al. [234] and Huang et al. [101] frequently throughout this chapter, and we often refer to them as one interchangeable system. We do so because the article by Huang et al. [101] appears to be a re-print with only minor, cosmetic changes of the earlier paper by Shen et al. [234].

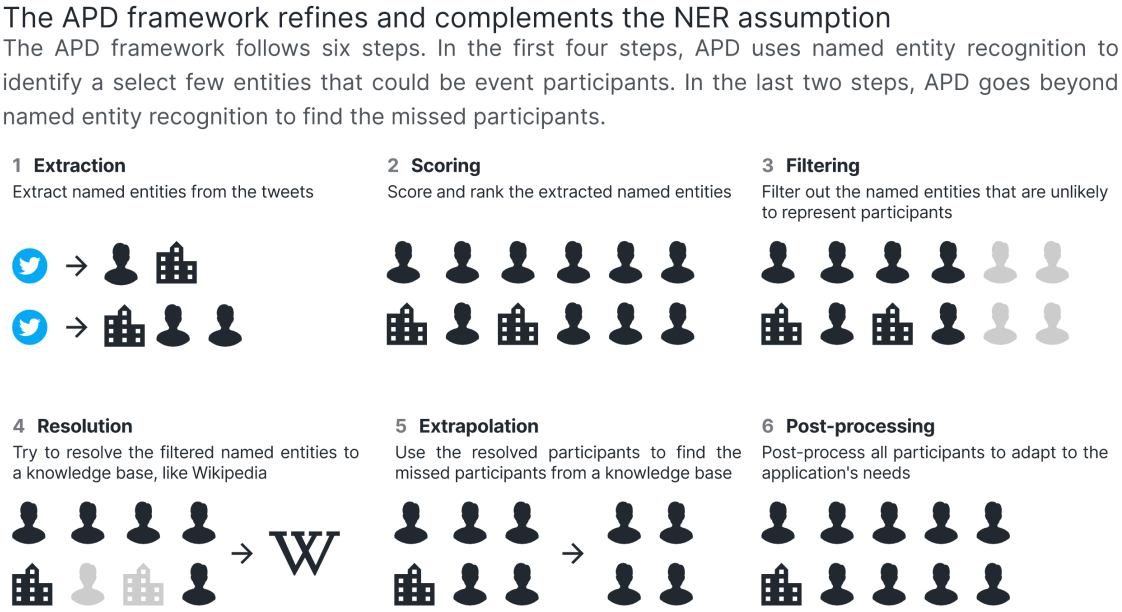

Figure 3.1: The six-step APD framework extends the NER assumption. The first three steps apply an NER tool, whereas the last three refine and complete its output. Figure reproduced from [148].

Moreover, the query expansion principle alone did not suffice. NER's limited scope, in particular recall, hindered the query expansion task and forced us to add the entity set expansion principle to the design. Participants tend to share common traits, we observed, so we could follow the named entities that could substitute for participants to find the ones that the NER tool had missed. These two principles formed our new assumption about participation and guided the APD framework's six steps.

The first four steps reflect the first part of the assumption, that certain named entities could substitute for participants. The first step extracts all named entities, the second step scores and ranks them, and the third step filters the ranking to retain only frequent named entities, like Shen et al. [234] and Huang et al. [101] do. Differently from Shen et al. [234] and Huang et al. [101], however, the fourth step resolves the remaining named entities to a knowledge base, a process of disambiguation and semantic filtering.

The last two steps reflect the second part of the assumption, that the resolved named entities could lead us to the ones that the NER tool had missed. A few named entities hog attention [101], as we also showed above, so the fifth step uses the resolved participants to extrapolate the missed ones from a knowledge base. Finally, the sixth step post-processes the participants, adapting them to the needs of the application.

At a glance, the APD process may appear unnecessarily-complicated. It may seem

like a complex answer to ostensibly-simple questions: Who is participating in an event? Where is the event taking place? Nevertheless, it is the inherent complexity of semantic understanding that demands APD's complex process. Therefore the APD process represents more than a glorified filter of NER's outputs. It incorporates all the lessons of Allan et al. [10], Chen and Ku [35], and Li et al. [125]: it co-references and filters, and understands participants with a more semantic direction than any of them. It represents the transition from linguistic to semantic understanding.

The APD framework represents our answer to the question of what makes a named entity a participant. A named entity becomes a participant when it plays a "significant role in the event" [101; 234]; when we cannot separate the named entity from the event because one affects the other [144]; when we cannot explain the event without the named entity: a "building block" [158]. Detecting participants, then, goes through understanding the participants themselves. In the next section, we present DEPICT, a novel algorithm to understand, not merely detect, participants.

3.2 | DEPICT: DEtecting Participants by Inferring Common Traits

Our first APD implementation showed a way for TDT researchers to overcome the limits of linguistic understanding [144]. We assumed that certain named entities could substitute for participants, and that other participants formed tight-knight communities around them in the Wikipedia graph. The algorithm, part of ELD's architecture, yielded statistically-significant gains in precision and recall over two NER models. Nevertheless, we could not describe the technique as an unmitigated success.

Our first APD implementation also exposed the complexity of semantic understanding [144]. Every step in the framework represents an assumption, and every assumption an intricate process. ELD's assumption required us to re-construct sections of the Wikipedia graph, and still, despite the complexity, the algorithm remained unaware of what made participants out of named entities. It understood relations among participants but not the participants themselves. In some domains, with such a weak grasp of semantics, the algorithm was prone to semantic drift. In other domains, the core assumption lost all meaning and the algorithm failed utterly.

In this section, we introduce DEtecting Participants by Inferring Common Traits (DEPICT), an extension to ELD's APD algorithm in three parts. First, in DEPICT we experiment with Twitter's new-and-untried proprietary NER model; to the best of our knowledge, we become the first to experiment with Twitter's annotations in the context

of TDT. Second, DEPICT extracts attributes to understand the participants themselves, not the associations among them; we give an example in Figure 3.3 on page 44. Third, DEPICT applies the attribute profiles to simplify and refine the APD process.

Our novelties concern only three steps. We experiment with different extractors at step one, and implement a new extrapolator and a new post-processor at steps five and six. Nevertheless, the three other steps—scoring, filtering and resolution—still play a crucial role in DEPICT's design, so for the sake of clarity, we describe all six steps next.

Extraction

The extraction step receives a corpus of tweets and extracts, from each tweet, a list of named entities. We filter no tweets; the next steps render it unnecessary. We do not even filter retweets. Retweets might induce bias [225] and, intuitively, accentuate the imbalance and asymmetry that we observed in the previous section, in Table 3.1. Yet bias takes another form, which we accept more readily: the bias towards well-written, authoritative content, as we demonstrate in Section 4.3.

We only change one thing in tweets. We expand user mentions, replacing account handles such as *@NicholasMamo* with display names like *Nicholas Mamo*. The minor change makes tweets more imitative of natural language, and thus facilitates NER. Then, we extract named entities with one of three NER models, which represent three broad classes of tools: the general-purpose, the bespoke and the proprietary.

First, we extract named entities using NLTK's general-purpose NER model. NLTK's NER model [22] represents the general-purpose solution, a model that balances performance with annotation quality [231; 270], designed to work in any medium and with any subject. It does not, however, adapt to the unconventional. In particular, in Section 3.3 we will reveal how NLTK struggles to adapt to Twitter's foreign syntax and erratic grammar.

Second, we extract named entities using TwitterNER [166]. TwitterNER represents the bespoke solution, a model whose existence revolves around overcoming one well-defined problem: Twitter's syntax. TwitterNER distinguishes between words, hashtags and mentions, and uses a word representation trained on six billion tweets. More distinctively, the NER model also employs a multitude of gazetteers, covering common first and last names, places and companies, and many other types of entities from diverse domains. With the gazetteers, TwitterNER progresses from a mere syntactical understanding to an understanding of what a named entity may be. We had also used TwitterNER in our previous work [144].

Third, we extract named entities using Twitter's own NER model. Twitter's annotator represents the proprietary solution, the social network's answer to its own challenges of syntax and language. Twitter introduced entity annotations as part of the new version of its API in late 2019 [83], revealing very little about how its algorithm annotates named entities. Twitter has implied the use of semantic analysis and the existence of a domain graph, but those seemingly only concern tweets' contextual annotations, not entity annotations. Therefore at the time of writing, we still do not understand Twitter's entity recognition process any better than we did at launch.

To the best of our knowledge, we become the first to apply Twitter's named entity annotations in the context of events. The black-box nature of Twitter's algorithm inhibits research from drawing any meaningful conclusions on its design, and we will not presume to either. Nevertheless, a certain value remains in the study of Twitter's annotator. As we compare Twitter's NER model with NLTK's and TwitterNER in Section 3.3, we come to understand better the benefits and trade-offs of the proprietary algorithm. At the end of the extraction step, NLTK's NER model, TwitterNER and Twitter's annotator return a list of named entities.

Scoring

The scoring step receives a list of named entities, one for each tweet, and assigns a score to every named entity in the corpus. While NER models extract named entities from individual tweets, the APD framework extracts named entities from the entire event discourse. The first part of our assumption requires us to identify the few that could plausibly substitute for participants. Scoring facilitates the task. The second step aggregates the extractor's output into individual scores: one score for every named entity, an indication of the entity's extent of participation in the event.

To score named entities, we adopt the same assumption as Shen et al. [234] and Huang et al. [101]. As we confirm later, in Section 3.3, frequent named entities tend to be participants. Therefore the scoring step simply counts the number of times a named entity appears in our corpus. At the end of the scoring step, the scorer returns a frequency-based ranking of named entities.

Filtering

The filtering step receives a ranking of named entities and retains the ones that could substitute for participants. While we follow the same assumption as Shen et al. [234] and Huang et al. [101] to score participants, the premise, which rests solely on frequency, re-

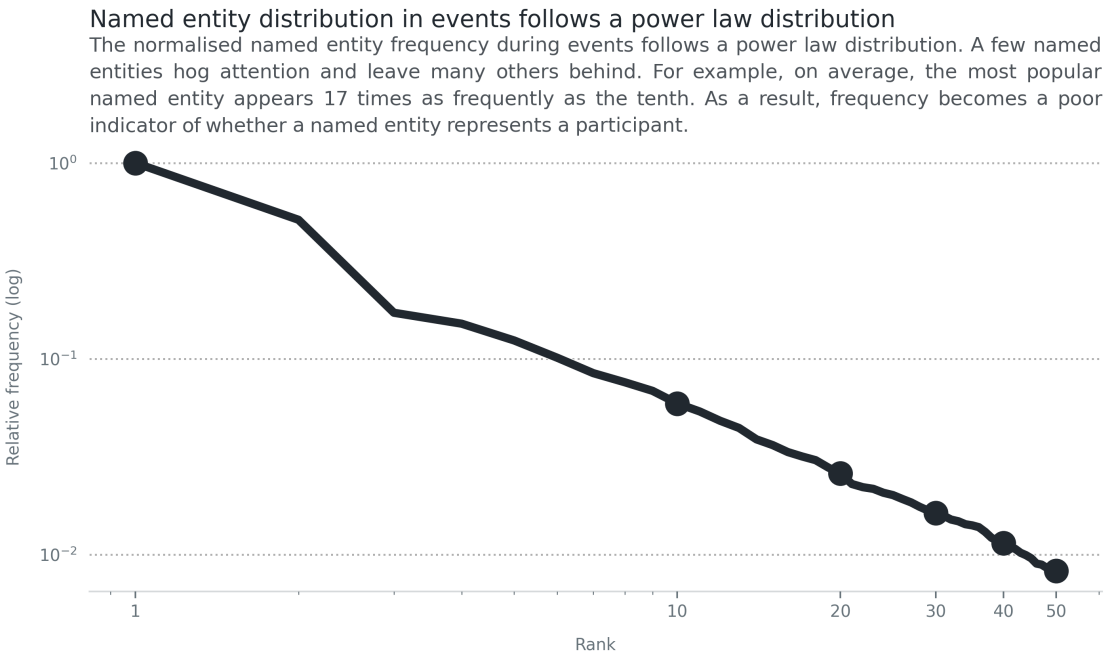

Figure 3.2: Named entity frequency in events follows a power law distribution. A few participants appear with a forceful intensity, while most appear scarcely. The behaviour makes simple, frequency-based linguistic filters unreliable to extract participants.

mains a bold one. The assumption that frequent named entities represent participants has an undeniably-intuitive basis, but it fails often. As we showed in Table 3.1, frequency may reflect relevance, but relevance seldom reflects participation.

The overindulgence in frequency has important ramifications. Ignore, momentarily, NER models' low precision, and frequency remains an unreliable indicator of participation. Rapid declines and sheer drops characterise the frequency plot in Figure 3.2, a power law distribution. The most-frequent named entity appears, on average, seventeen times as much as the tenth. Even if a frequency-based filter sufficed to understand semantically, it would still tread an uncertain boundary that shifted constantly across events: at what frequency do the participants end and the named entities begin?

We relax the assumption. We still assume that frequency indicates participation, but we do not depend on the assumption alone. Instead, like in our previous work [144], we use a rank-based filter; a football match has around 50 participants, so we filter the top 50 named entities. The rank-based filter avoids an arbitrary threshold that, like in Shen et al. [234] and Huang et al. [101], varies with the event's popularity, and it captures as many candidate participants as possible. Evidently, we pay little regard to precision,

but we do so in the safe knowledge that the next step, resolution, filters named entities with a more semantic diligence. At the end of the filtering step, the filter returns a list of the most frequent named entities.

Resolution

The resolution step receives a list of named entities and resolves them to a knowledge base. Our filtering so far, like in Shen et al. [234] and Huang et al. [101], has depended on frequency, a process barely more expressive of participation than linguistic understanding. The fourth step filters again, but it filters differently, with a more explicit semantic intent. The resolution step thus symbolises the threshold between linguistic understanding and semantic understanding.

Resolution filters named entities in a process of disambiguation. By definition, resolution only maps named entities to a knowledge base, but the inverse process, of discarding named entities with little relevance to the event, amounts to a filtering operation. From a more practical perspective, resolution refines the extrapolator's seed set to capture more accurately semantic relevance [209]. In this chapter, this process of refinement matters even more than it did in our previous algorithm. DEPICT understands participants to extrapolate the ones that the NER models missed, and therefore it requires an accurate understanding of the ones that the NER models captured.

DEPICT re-uses ELD's resolver, which could disambiguate named entities to a precision of around 70% [144]. In summary, the resolver disambiguates named entities by assessing how well their Wikipedia articles fit in the event's domain. The algorithm starts by searching for named entities on Wikipedia and retrieves, for each one, the top ten articles. Then, it compares the first sentence of each page with the centroid of all tweets in the event's corpus using cosine similarity. A named entity resolves to an article if the similarity exceeds a threshold, empirically-set to 0.10. At the end of the resolution step, the resolver returns a list of disambiguated participants.

Extrapolation

The extrapolation step receives a list of resolved participants and extrapolates from them the ones that the NER model missed. No other component matches the extrapolator's contribution to recall. In ELD, the extrapolator alone caused recall to soar, almost double, above NLTK's (27% ↑ 53%) and TwitterNER's (32% ↑ 60%) [144]. Yet the recall figures still betray failures in the extrapolator, such as how it tended to drift semanti-

cally. Much of DEPICT's design aims to stifle semantic drift, so before we present the new extrapolator, it is worth exploring why and how ELD's method failed.

Extrapolation, together with resolution, embodies our answer to the question: what makes a named entity a participant? In ELD, we answered the question with a simple approximation. We assumed, like several others in entity set expansion literature [31; 235], that named entities become participants when they form communities with other participants in the Wikipedia graph. The extrapolator ranked articles from the largest communities by comparing them with the domain, like the resolver. It was this lack of semantic guidance, this ignorance about what made participants of the resolved named entities, that led to semantic drift [144].

Our assumption here is that named entities become participants when they resemble resolved participants in what they are and what they do. In the 2020 US presidential election, the community structure did not liken Joe Biden to Donald Trump—their role did: both were politicians contesting the same election. The link between them, like the hundreds of other connections that they shared with non-participants, existed only as proof of an unknown relation: a guide, certainly, but not a solution. Therefore to understand the Who and the Where, we first need to understand the participants themselves, a form of meta-understanding.

We understand participants by extracting their defining traits, their attributes. Expanding entity sets by identifying the common aspects in the seed set seems entirely intuitive, even if attributes seldom feature in practice [21; 293]. We could find no common definition of attributes [12; 251], a recurrent theme in IR, but we regard the task as a generalisation of relation extraction. Every attribute has a type and a value, like a relation, but the value does not necessarily denote another entity. On the contrary, an attribute normally describes an intrinsic property [100]. Therefore, like Huang et al. [100], we consider attributes and relations in the same way.

We extract attributes from Wikipedia. The massive encyclopedia stands as a high-quality [74; 284], consistent [286] and up-to-date [1; 209] record of concepts. In fact, you could find Wikipedia behind derivative knowledge bases like YAGO [241] and DBpedia. Their structures, containing readily-available attributes, may even appear as superior candidates to Wikipedia itself if not for often-incomplete and outdated information [36; 37; 269].

Wikipedia has another, more subtle advantage. Formal knowledge bases do not distinguish clearly between past and present aspects of an entity. At the time of writing, DBpedia affiliated Neymar, a Brazilian footballer, with three `dbp:clubs`: Santos, Barcelona and Paris Saint-Germain. To find Neymar's current club, you would have to look up another attribute, `dbp:currentclub`. Conversely, a Wikipedia article's first sen-

tence, the definition sentence, contains no spurious details [144; 301]. It describes the concept's present state—the essence and nothing more. Therefore we extract attributes from the definition sentence.

To extract attributes, DEPICT uses a custom context-free grammar implemented in NLTK's shallow parser [22]. By design, the grammar assumes that the definition sentence describes, uniquely, the article's concept, which frees it from the burden of identifying the subject of an attribute. At the highest level, attributes comprise names or types, such as `BASED-IN`, and one or more values. At the lowest level, attribute names represent verbs, and attribute values represent named entities, nouns, numbers or dates. In-between, the grammar also recognises modifiers to attributes—adjectives or adverbs—and prepositions. Table 3.2 summarises our context-free grammar.

The actual extrapolation process follows a traditional IR route: generate, score and rank candidates. Our understanding simplifies the first step. We no longer reconstruct the Wikipedia graph to examine the community structure; we only compile a list of broadly-related concepts. DEPICT generates candidate participants by fetching the most frequently-linked concepts from the resolved participants. We ignore concepts with years in their titles, unless the year appears in parentheses, as a disambiguator.

The extrapolator's only parameter, k, controls how many candidates we generate and balances coverage with semantic drift. A larger value of k allows DEPICT to reach otherwise-unreachable candidates but also stretches the scope of the candidate set. In practice, we found a value of 200 to strike a good balance. Then, DEPICT extracts the attributes of every resolved and candidate participant; we refer to a participant's collection of attributes as its attribute profile.

DEPICT builds attribute profiles for the k concepts using the grammar, but we also consider four additional practical aspects of attributes. First, prepositions alter the meaning of attributes. We cannot compare the team a footballer plays *for* with the position which they play *as*. The two attributes hold incomparable values, so DEPICT programmatically separates attributes by preposition: `PLAYS-FOR` and `PLAYS-AS`.

Second, modifiers tend to be needlessly-specific. Max Verstappen's definition sentence describes him as "a Belgian-Dutch racing driver", but nationality has very little to do with Formula 1. Of course, the modifiers might be what link participants together; in US presidential elections, candidates are not merely politicians but *American* politicians. In reality, however, the low value of k limits the semantic drift to a negligible effect. Therefore DEPICT retains only the head of the attribute value, without modifiers: racing driver, not *Belgian-Dutch* racing driver.

Third, not every attribute characterises the prototypical participant. Attributes in parentheses, such as `BORN`, describe personal characteristics, and others describe almost

Description	A date consists of two numbers and a proper noun in various orders, and may be preceded by a weekday.	
Production	`DATE: <CD> <NNP> <CD>`	EXAMPLE: 14/CD May/NNP 2017/CD
	`DATE: <NNP> <CD> <,> <CD>`	EXAMPLE: March/NNP 9/CD ,/, 1996/CD
	`DATE: <NNP> <,> <DATE>`	EXAMPLE: Tuesday/NNP ,/, 6/CD June/NNP 1944/CD
Description	A named entity consists of a sequence of proper nouns, but it may also contain numbers or pronouns.	
Production	`ENT: <CD>? <NNP.*> (<CD|NNP.*|PRP>)*`	EXAMPLE: 1860/CD Munich/NNP EXAMPLE: World/NNP War/NNP I/PRP
Description	A modifier includes a number, or a list of adjectives or adverbs that modify something, usually a noun phrase.	
Production	`MOD: <CD><IN><DT>`	EXAMPLE: one/CD of/IN the/DT
	`MOD: <CD>?<JJ.*|RB.*>+`	EXAMPLE: 2012/CD Finnish/JJ [drama film]
	`MOD: <MOD> (<CC|,|TO>? <MOD>)+`	EXAMPLE: Canadian/JJ ,/, English-language/JJ ,/, conservative/JJ [newspaper]
	`MOD: <ENT|NP> <MOD|POS>`	EXAMPLE: Brazilian/ENT national/MOD [team]
Description	A noun phrase consists of a sequence of nouns, which may be preceded by modifiers or by an entity. A noun phrase may also combine a series of noun phrases separated by modifiers.	
Production	`NP: <MOD|VBG>* <NN.*>+`	EXAMPLE: football/NN team/NN
	`NP: <ENT> <NP>`	EXAMPLE: France/ENT (national team)/NP
	`NP: <NP> <MOD> <NP>`	EXAMPLE: broadsheet/NP daily/MOD newspaper/NP

Description	An attribute name or type consists of one verb in any tense, including past participles.	
Production	`NAME: <VB.*>`	EXAMPLE: plays/`VBZ` [for France]
Description	An attribute value consists of a noun phrase, an entity, a number or a date, or a series of such values.	
Production	`VALUE: <NP|ENT|CD|DATE>+`	EXAMPLE: (Brazilian professional footballer)/`NP` EXAMPLE: [born on] (28 October 1955)/`DATE` EXAMPLE: (Ligue 1)/`ENT` club/`NP` Lyon/`ENT`
Description	An attribute consists of one or more values separated by coordinating conjunctions or commas. Values may have determiners, possessive pronouns or modifiers. The list itself may also have a preposition as a modifier.	
Production	`VALUES: <TO>? <IN>? (<DT|PRP$>? <MOD>* <VALUE><CC|,>*)+`	EXAMPLE: an/`DT` (American businessman)/`VALUE`, developer/`VALUE` and/`CC` investor/`VALUE` EXAMPLE: [adopted the euro] as/`IN` their/`PRP$` (primary currency)/`VALUE`
Description	An attribute consists of a name and a list of values, and each list may have modifiers.	
Production	`ATTR: <NAME> (<MOD>? <VALUES>)+`	EXAMPLE: plays/`NAME` (as an attacking midfielder)/`VALUES` (for Ligue 1 club Lyon and the Brazil national team)/`VALUES` EXAMPLE: [Memphis Depay, also] known/`NAME` simply/`MOD` (as Memphis)/`VALUES`

Table 3.2: DEPICT uses a context-free grammar to extract entity attributes. In the examples, we use parentheses to indicate the POS tag of a group of symbols without annotating each symbol individually. For example, the named entity *Ligue 1* would be annotated as *(Ligue/`NNP` 1/`CD`)/`ENT`*, but we annotate it as *(Ligue 1)/`ENT`* for the sake of clarity.

DEPICT creates attribute profiles from Wikipedia's definition sentences

DEPICT creates attribute profiles using only the first sentences of Wikipedia articles, the definition sentences. DEPICT removes all text in parentheses—secondary information—and uses a shallow parser to extract attributes. The final profiles exclude modifiers and separate attribute types by preposition.

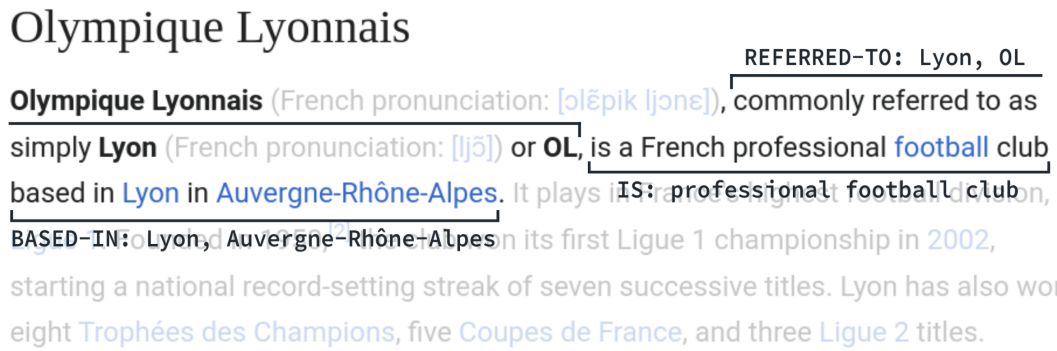

Figure 3.3: DEPICT creates attribute profiles by parsing the first sentences of Wikipedia articles, the definition sentences. The extrapolator extracts attributes using a custom context-free grammar, but it also refines profiles further programmatically.

exclusively one participant—a football team can only have one captain. We cannot generalise such attributes and require them of all the candidates. Therefore DEPICT prunes attributes if they appear in parentheses or describe only one resolved participant.

Fourth, we can only be certain that the participants' attributes relate to the event. However, an attribute may describe a candidate but not any of the participants. We can neither ascertain that the attribute is relevant nor that it is irrelevant, and it would be incorrect to penalise a candidate's score for an attribute that may, in the end, have some relevance. Therefore DEPICT prunes the attributes of candidates that do not describe any of the resolved participants. Figure 3.3 shows a sample profile with three attributes: `REFERRED-TO`, `IS` and `BASED-IN`.

We score and rank candidates by comparing them with the resolved participants. A candidate resembles a resolved participant not only because the two share attributes but, most importantly, because they also share attribute values: two footballers participate in the same event because they `PLAY-FOR` the same team. DEPICT reflects this logic with the Jaccard similarity, $sim_{c,r}$, which computes the overlap in the attribute values of candidate c and resolved participant r:

$$sim_{c,r} = \frac{\sum_{a \in A(r)} \frac{|r_a \cap c_a|}{|r_a \cup c_a|}}{|A(r)|} \tag{3.1}$$

The equation iterates over the participants' attributes, $A(r)$, and compares the values of attribute a in the resolved participant's and candidate's profiles, r_a and c_a. We construct the final ranking similarly. We average candidate c's similarity over all resolved participants, set R, to get the overall score, $score_c$:

$$score_c = \frac{\sum_{r \in R} sim_{c,r}}{|R|} \cdot \frac{\ln w_c}{\ln \max(w_c | c \in C)} \tag{3.2}$$

To the averaged score, we also add a second factor, a logarithm with a dual purpose. First, the logarithm functions as a weighting mechanism. Similarly to ELD's assumption on community structure, we assume that the greater the number of incoming links w_c from resolved participants, the greater the likelihood that candidate c is relevant to the event. While designing DEPICT, we observed that weighting improves the ranking order but reduces balance by increasing bias; we leave further investigation into bias for future work. We normalise the weight by the highest value among all candidates C to maintain the score's range between 0 and 1.

Second, and more subtly, the logarithm acts as a threshold. If just one resolved participant links to a candidate, then the candidate must only have a meagre similarity to the resolved participants. By extension, it must have a meagre relevance to the event. The logarithm assigns a score of 0 to such candidates, regardless of the logarithmic base. DEPICT filters all candidates with a score of zero, whether by weight or similarity, and ranks the rest. At the end of the extrapolation step, the extrapolator returns a ranking of extrapolated participants.

Post-processing

The post-processing step receives a list of resolved and extrapolated participants, and adapts them to the application's needs. DEPICT simply re-constructs the participant profiles, this time with fewer cares. The post-processor extracts all attributes, without removing modifiers and without pruning, to portray participants to the best of the grammar's abilities. Then, it ranks participants by score: first the resolved participants, then the extrapolated ones. At the end of the post-processing step, DEPICT returns a ranking of participants and their profiles: who they represent and what they do.

Understanding adds an intrinsic value to participants, understanding for the sake of understanding. DEPICT identifies not only the associations between participants but what makes them participants in the first place. We understand not only that a named

entity represents a person but also what they do and what nicknames they go by; not only that a named entity represents a location but also where it lies, a hierarchy that spans cities, regions and countries. Yet understanding adds an even greater value to participants, an extrinsic one.

Understanding gives TDT researchers new possibilities around which to design algorithms. With a better understanding of basketball players, perhaps Shen et al. [234] and Huang et al. [101] would have covered less noise and more participants. Perhaps they would have built separate timelines for teams, not only players. At least, they would have had the possibilities with understanding like DEPICT's. They could not have had them with the understanding of NER models. We explore these new possibilities of understanding in Chapter 6. Before, we evaluate DEPICT.

3.3 | Refining linguistic understanding

The disappearance of semantic understanding from TDT literature was logical. It made sense for Shen et al. [234], Huang et al. [101] and others to stray no further than the lessons of Allan et al. [10], Chen and Ku [35], and Li et al. [125]. Understanding semantically represented a lengthy, complex process to replace the enduring tradition of linguistics with event semantics, a solution with no guarantees. After all, research's task was TDT, not understanding. In this section, we cede to the doubts. We start by debating whether linguistic understanding could ever serve as semantics before we explore how semantic understanding pushes the boundaries of event knowledge.

Our evaluation setup follows our previous work [144]. For most of this section, we evaluate NER and APD models on ten football matches, a domain with a rigid structure that has served TDT research well, as we explain in Chapter 5. We collected datasets for an hour, starting 75 minutes before each match and ending an hour later. During this time, teams publish the line-ups, triggering intense discussions about participants. Appendix D.1 includes more details about our datasets.

We compare five models throughout this section: three NER tools and two APD algorithms. The three NER tools, NLTK's [22], TwitterNER [166] and Twitter's annotator, follow the APD framework until step three: extraction, scoring and filtering. The two APD algorithms, DEPICT and our de facto baseline, ELD's participant detection technique, follow the entire process. At the end, all models produce rankings containing not more than 50 named entities or participants.

We compare three aspects of the five models. First, we evaluate the overall quality using IR research's standard metrics: precision and recall, often combined into the sin-

gular F-score. The three metrics can be expressed in terms of the true positives tp, the false positives fp, and the false negatives fn:

$$precision = \frac{tp}{tp + fp} \tag{3.3}$$

$$recall = \frac{tp}{tp + fn} \tag{3.4}$$

$$F\text{-}score = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{3.5}$$

Second, we evaluate the order quality. The rank of a participant reflects the model's confidence in its level of participation, so the order matters. We measure the order quality using Average Precision (AP), itself based on Precision at k (P@k), or the precision of all elements until rank k. AP focuses more narrowly on the values of P@k at ranks with valid participants; rel_k evaluates to 1 if the element at rank k is valid and 0 otherwise. Then, it computes the average over all participants in the ground truth GT. For brevity, we summarise the AP with Mean Average Precision (MAP), or the mean AP across all events, the set E:

$$rel_k = \begin{cases} 1 & \text{if the element at rank } k \text{ is relevant} \\ 0 & \text{otherwise} \end{cases} \tag{3.6}$$

$$\text{AP} = \frac{1}{|GT|} \sum_{k=1} \text{P@k} \cdot rel_k \tag{3.7}$$

$$\text{MAP} = \frac{1}{|E|} \sum_{e \in E} \text{AP}_e \tag{3.8}$$

Third, we evaluate the symmetry. In our previous work, models often captured a skewed perspective of participation. Shaped by Twitter's discussion, rankings frequently extracted nearly all of the participants from one football team but almost none from the other. To measure symmetry, we introduce the balance metric, bound between 0 and 1. Balance reaches the minimum value when the ranking is completely asymmetrical, or when it captures participants exclusively from one team. It reaches the maximum value when the ranking is completely symmetrical, or when the number of participants from the first team, r_1, equals those of the second team, r_2. In other words, the higher the balance, the lower the bias:[2]

[2]Our balance metric is not defined in the case where a model captures no participants at all. Similarly, balance can only be applied in event domains whose participants segregate naturally into two groups, such as in football matches. In the context of football matches, we associate the stadium with the home team.

$$balance = \min(\frac{r_1}{r_2}, \frac{r_2}{r_1}) \tag{3.9}$$

We annotate rankings using widely-available ground truths. Specifically, we use LiveScore.com's and The Guardian's reports. We consider as precise all entities that fulfil our definition of participants, Definition 3, normally the teams, coaches and players as the Who, and the stadium as the Where. The APD models do not produce any redundant participants, but NER tools capture many indirect references: surnames, nicknames and aliases. Therefore in the NER rankings, we also accept as precise indirect but unambiguous references to participants.

In the rest of this section, we follow the procedure above. We start by analysing the quality of the three NER models, symbols of linguistic understanding. We ask, in the first analysis, whether NER tools could ever improve enough to bridge the gap between linguistic and semantic understanding.

The limits of linguistic understanding

Before the TDT community could consider semantic understanding on Twitter, it had to confront the quality of its linguistic understanding. As early as in 2009, Sankaranarayanan et al. [229] observed tell-tale signs that Twitter would challenge all manner of NLP tasks, not least NER. Later, Ritter et al. [218], Panagiotou et al. [191] and many other researchers who looked closely enough observed similar signs and echoed the same calls: NLP on Twitter had to develop independently of general-domain NLP. Therefore for a long time, the TDT community could not even claim that its linguistic understanding was good linguistic understanding.

Recent advances have heeded those early calls. NER models on Twitter have progressed tremendously since Sankaranarayanan et al. [229] and others first noted the inadequacy of traditional NLP tools on Twitter. The summary results in Table 3.3 show how modern solutions like TwitterNER [166] and Twitter's own annotator eclipse the more traditional NER model of NLTK. Spurred by these results, in this analysis we investigate whether a new generation of NER tools could permit research to forego semantic understanding.

NLTK's NER model fared poorly. In its outputs, we observed many of the same signs that Sankaranarayanan et al. [229] did. Twitter's unique syntax—irregular orthography, emojis, full stops replaced with new lines—misled NLTK's model, which could not delimit named entities, identify where one ends and another one starts. Ordinary capitalised hashtags like *#JuventusFC* appeared as names. NLTK's traditional model understood neither Twitter's syntax nor, as a consequence, named entities.

Model	Precision	Recall	MAP	Balance
NLTK	29.80%	21.55%	17.36%	0.4885
TwitterNER	△ 32.80%	▲ 26.67%	20.09%	△ 0.7273
Twitter	▲ 44.40%	▲ 33.98%	▲ 30.60%	0.5786

Table 3.3: TwitterNER and Twitter's annotator out-perform NLTK but still do not suffice, neither in precision nor in recall. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the model in the row above. We present a full breakdown of the results in Table F.2.

TwitterNER fared better. The model was not misguided by Twitter's peculiarities, around which Mishra and Diesner [166] designed their model. TwitterNER understood, through its gazetteers, what a named entity may be. Only the heavy Western influence of the lexicons—European and American names dominate—seemed capable of hindering the model, which captured more confidently participants from Western countries. Still, in these experiments TwitterNER captured significantly more participants and less noise than NLTK.

Twitter's annotator fared even better. No one could understand Twitter's syntax better than Twitter itself. Twitter's proprietary model skilfully navigated hashtags, mentions and all the other distinctive properties of the social network's user-generated writing. The annotator captured named entities with impressive accuracy: 11.60% more precision than TwitterNER and 14.60% more than NLTK's NER model; 7.31% more recall than TwitterNER and 12.43% more than NLTK—all statistically-significant gains at the 99% confidence level.

Of course, we cannot rest too comfortably on Twitter's annotations. We can only speculate about why Twitter's annotator improved precision and recall to such a degree. Do Twitter's entity annotations owe their success to a hidden measure of semantics? Or do the improvements perhaps stem from the topic graph, which Twitter's contextual annotations use? Every named entity is annotated by a function in a black box that we can neither fully comprehend nor, more importantly, improve.

Moreover, Twitter's annotations do not outdo TwitterNER in every aspect. The three extractors generally performed consistently: precision and recall rose and fell in synchrony across events, but not balance. In balance, the three models did not perform consistently. TwitterNER represented teams with more symmetry than either NLTK ($0.4885 \uparrow 0.7273$; one-tailed paired samples t-test: $p = 0.0159$) or Twitter's annotator

(0.5786 ↑ 0.7273), although the improvements over the latter were only significant at the 90% confidence level (one-tailed paired samples t-test: $p = 0.0802$).

Such imbalance has secondary effects, including how it affects algorithms like the ones proposed by Shen et al. [234] and Huang et al. [101]. The two designed an algorithm to democratise TDT's standard monolithic timeline with individual timelines, one for each named entity. Individual timelines would overcome Twitter's bias, they reasoned, unaware that NER tools yield to the same bias. In fact, the way in which Shen et al. [234] and Huang et al. [101] filtered infrequent named entities might have inadvertently exacerbated the imbalance in favour of the few participants that dominate Twitter's discussions.

Imbalance affects APD as well, perhaps more so than any other metric. As we showed in our previous work [144], and as we show again later in this chapter, extrapolation mirrors the input in its output: give an extrapolator a biased seed set and it only accentuates the bias further; but give an extrapolator a symmetric seed set, and it could capture almost every participant with the same symmetry. Even in DEPICT, balance remains a determining factor to performance.

We cannot easily determine why Twitter's annotations outdo the two other methods so convincingly in every metric except balance. Twitter's cryptic annotator may induce bias somewhere in its function. More likely, however, imbalance does not indicate a flaw in Twitter's annotator but the opposite: a virtue of TwitterNER's. In TwitterNER, gazetteers hold a large variety of first and last names, which distribute evenly between teams. Thus, TwitterNER normally favours no side in particular.

Nevertheless, beyond the incremental improvements from NLTK's NER model to TwitterNER to Twitter's own annotator, one sign remains irrefutable: not a single NER model meets our needs. We could combine the best qualities of TwitterNER—its openness and balance—with the best qualities of Twitter's annotator—its precision and recall—and the results would still not match our expectations. Frequent entities tended to be participants, but the rest were scattered haphazardly, participants intermittent among named entities. Even at their highest, precision and recall remain too low.

We could hardly fault TwitterNER or Twitter's annotator either. Rarely did they mistake a noun or a verb, or a hashtag for a named entity. Most annotations correctly identify named entities, including a small selection of participants and a multitude of references to them. Actually, such references hold value. They tells us all the different ways of referring to a participant, formal and informal, which Wikipedia does not. At best, however, references require a good co-reference resolution algorithm to make sense of all the dangling names, like the one Shen et al. [234] and Huang et al. [101] employ. At worst, as we showed in the experiment of Section 3.1, references saturate the ranking,

occupying spaces that other, less popular participants could fill better.

In short, the adaptations to Twitter's syntax and user behaviours refine NLP, but only NLP. The technological advances in TwitterNER and Twitter's own annotator lessen the worries of Sankaranarayanan et al. [229], Ritter et al. [218] and others but do nothing to lessen the TDT community's own worries. Regardless of how much the performance of NER models improves, it cannot seem to improve enough to bridge linguistic understanding with semantic understanding. The event tracking community must develop its own understanding of the Who and the Where.

The new limits of semantic understanding

If linguistic understanding cannot suffice, then semantic understanding must. In this analysis, we try to imagine what kind of understanding Shen et al. [234], Huang et al. [101] and others would have inherited had they sought out semantic understanding that truly captured participants "that play a significant role in the event." As we investigate ELD's APD model and DEPICT, we discover the trade-offs between the convenience and accessibility of linguistic understanding, and the complexity of semantic understanding.

The results in Table 3.4 paint an unmistakeable scene. We tested three versions of ELD's APD method and DEPICT, one for each NER model, and the conclusions did not change. It mattered little whether we used NLTK's model, TwitterNER or Twitter's annotator; and it mattered even less whether the extrapolator implemented ELD's assumption of interconnectedness or DEPICT's deeper understanding of participants: our APD models surpassed the NER tools in precision and recall consistently, and always with statistical significance. The worst APD model out-performed even the best NER tool. Clearly, semantic understanding out-performs linguistic understanding.

Of course, semantic understanding still relies heavily on linguistic understanding, the NER extractors. When the latter erred, the former erred with it, and all three NER models erred, although none as frequently or as gravely as NLTK's. NLTK's NER model captured less than a quarter of participants on average, and the dearth of information re-appeared subtly in our APD methods. ELD's and DEPICT's recall hovered around 50%. For every participant from one team, ELD and DEPICT captured around four from the other, suggesting an almost-perfect asymmetry that confirms our previous findings about bias [144].

In practice, however, our APD methods required very little more than what NLTK's NER model could provide. Generally, TwitterNER only captured three or four more participants than NLTK and with a significantly better balance, but the incremental

Model	Precision	Recall	Precision (lenient)	MAP	Balance
NLTK	29.80%	21.55%	31.20%	17.36%	0.4885
ELD_{NLTK}	▲ 48.68%	▲ 50.55%	▲ 66.52%	▲ 38.56%	▽ 0.2856
$DEPICT_{NLTK}$	△ 54.00%	53.77%	△ 76.04%	40.28%	0.2581

(a) The poor results of NLTK's NER tool hindered ELD's APD method and DEPICT.

Model	Precision	Recall	Precision (lenient)	MAP	Balance
TwitterNER	32.80%	26.67%	34.40%	20.09%	0.7273
$ELD_{TwitterNER}$	▲ 57.40%	▲ 60.93%	▲ 78.00%	▲ 47.66%	▽ 0.5337
$DEPICT_{TwitterNER}$	▲ 66.08%	▲ 70.57%	△ 87.67%	△ 56.05%	0.6161

(b) TwitterNER only improved slightly over NLTK's NER model, but the improvements, notably in balance, translated into large gains in the performance of our two APD algorithms.

Model	Precision	Recall	Precision (lenient)	MAP	Balance
Twitter	44.40%	33.98%	46.40%	30.60%	0.5786
$ELD_{Twitter}$	▲ 60.40%	▲ 64.20%	▲ 80.80%	▲ 51.73%	0.5542
$DEPICT_{Twitter}$	62.13%	66.05%	84.84%	54.80%	0.5431

(c) Twitter's annotator improved over TwitterNER, although this time, the improvements did not reflect quite as strongly in our two APD techniques.

Table 3.4: Our APD techniques out-performed the NER models in most metrics when extracting event participants. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the model in the row above. We present a full breakdown of the results in Table F.3.

gains sufficed. In the match between Juventus and Inter, TwitterNER only captured the stadium, two teams, two players and one coach—merely 12.50% of all participants. Nevertheless, the perfect symmetry between sides gave our APD methods just enough information for DEPICT to recall 85.42% of all participants.

Likewise, TwitterNER's improvements in the other events proliferated throughout our APD methods. For the baseline, TwitterNER's more complete output served to clarify the community structure and performance spiked. For DEPICT, the increased symmetry served to create a more generalisable representation of the event's typical participant, rather than one team's typical participant. DEPICT peaked and captured, on average, more than seven out of every ten participants.

Our APD methods still failed on occasion, however. Neither method could maintain

the NER models' balance, and while ELD did not understand participants well enough to distinguish them from tangential concepts, DEPICT stuck too rigidly to its understanding. The coaches and stadiums, integral aspects of the Who and the Where, did not fit the profile of a typical participant, a footballer. Consequently, the extrapolator rarely extracted the stadium and the coaches as participants.

Sometimes, DEPICT also gleaned a deceptive representation of participants. Among the resolved participants in the match between Real Madrid and Chelsea figured Timo Werner, Antonio Rüdiger and Kai Havertz. All three footballers played for Chelsea and, incidentally, for the German national team too. Therefore after DEPICT had thoroughly filled-in the Chelsea squad, it resorted to the German squad and ignored altogether the Real Madrid one. Tellingly, ELD's method performed only marginally better.

Yet even when ELD's APD method and DEPICT failed, they failed gracefully. ELD's extrapolator first exhausted true participants before wandering through communities that, though relevant, did not host participants, and DEPICT did even better. Our novel method understood what a participant could be and what it could not be, and rejected any concepts that shared no traits with the resolved participants. In both cases, precision rose sharply, and recall and MAP more than doubled over NLTK's and TwitterNER's.

The upward trajectory ended abruptly. When we replaced TwitterNER with Twitter's annotator, the quality of the linguistic understanding improved. Twitter's NER model now gave ELD and DEPICT ample proof about participants and the communities they formed, albeit with less symmetry, but our techniques seemed unmoved by the extra data. While ELD's performance improved slightly, DEPICT registered no improvements at all. On the contrary, its recall worsened (70.57% $\downarrow$ 66.05%) and extinguished the statistical significance over ELD. The two algorithms converged, recalling around two-thirds of participants and nothing more.

To understand why ELD and DEPICT struggled to refine further, we annotated more leniently. The two APD algorithms could not easily understand that a footballer missed a match because of an injury or suspension, or simply because the coach dropped them. Yet by their absence, unavailable players affect managerial decisions and, thus, the event itself.[3] Therefore the lenient variant of the precision metric, shown in Table 3.4, accepts as precise those non-participants who would normally have participated, had they been healthy, eligible or selected for the match by the coach.

The new metric describes the problem more clearly. In two events, every one of DEPICT's participants could have participated in the event, and in three others, fewer than five named entities had no clear association with the event. ELD performed sim-

[3] An anonymous reviewer of our first APD article [144] made this argument and inspired this analysis.

ilarly, if slightly worse. The lenient annotation raised ELD's precision from 60.40% to 80.80% and DEPICT's from 62.13% to 84.84%. In other words, the lenient interpretation of precision practically halved the number of perceived mistakes.

The converging results, then, point to a practical limit in semantic understanding, and not a particularly obstructive one either. If we could forgive the relevant but mistaken named entities as nuisances in the way of the actual participants, then the solution would lie simply in enlarging the ranking. If we could not, the mistakes of semantic understanding still appear in far smaller numbers than those of linguistic understanding. Hence, we re-affirm our conclusion: the event tracking community must develop its own understanding of the Who and the Where.

The generalisability of semantic understanding

NER outdoes APD in one quality: generalisability. NLTK's NER model, TwitterNER and Twitter's annotator might have been designed for different mediums but not for the idiosyncrasies of any particular domain. The core idea of NER generalises and so the models behave predictably: most events concern named entities [56], and without any changes, NER models could always capture named entities, if not participants. Conversely, an APD algorithm's ability to capture participants depends on how accurately its assumption reflects the structure of an event domain.

Consider ELD's model. We speak of ELD's APD model as if our design only assumed one feature, that participants form communities, a practical premise in certain event domains. In fact, however, the design implied a second, subtler assumption: the expectation that only participants may form communities. The second assumption fails in many domains, and when it does, the brittle ELD drifts severely, as we show next.

To study the generalisability of our APD models, we experimented with the domain of Formula 1 Grands Prix. We collected data from seven races from the start of the 2022 season using the same process as before, as we detail in Appendix D.1. We also re-used our previous configurations to extract participants, except in one event; in the British Grand Prix, the resolver could not generate any seeds from which to extrapolate, so we lowered the threshold from 0.10 to 0.05. As we annotated the rankings, seeking the names of drivers and constructors as the Who, and the circuit and location as the Where, ELD's difficulties became apparent.

ELD's APD model failed abjectly. The results in Table 3.5 show how ELD barely even matched NLTK's recall and performed worse than more sophisticated NER models. At best, recall only reached 30.88% and MAP only 21.38%. We could only label 21.14% of ELD's participants as precise, and most, ELD owed to its resolver. The extrap-

Model	Precision	Recall	MAP
NLTK	25.43%	26.73%	11.28%
ELD_{NLTK}	▼ 18.29%	26.73%	△ 15.88%
$DEPICT_{NLTK}$	▲ 36.86%	▲ 58.06%	▲ 35.45%

(a) DEPICT achieved a high recall despite the often-inaccurate input from NLTK's NER tool.

Model	Precision	Recall	MAP
TwitterNER	36.57%	37.33%	33.50%
$ELD_{TwitterNER}$	▼ 20.86%	▽ 30.88%	▽ 21.38%
$DEPICT_{TwitterNER}$	▲ 34.00%	▲ 53.92%	▲ 35.13%

(b) DEPICT ceased to improve, despite the improvements in TwitterNER's output relative to NLTK. It captured most drivers but could not identify constructors as participants.

Model	Precision	Recall	MAP
Twitter	40.86%	38.71%	33.16%
$ELD_{Twitter}$	▼ 21.14%	30.88%	▽ 21.29%
$DEPICT_{Twitter}$	△ 35.43%	▲ 56.68%	▲ 33.75%

(c) Twitter's annotator exceeded TwitterNER's results but without matching those of our APD algorithms, which plateaued.

Table 3.5: DEPICT generalised better than ELD's APD model, even if performance dipped from the domain of football matches. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the model in the row above. We present a full breakdown of the results in Table F.4.

olator roamed unknowingly from communities of drivers and constructors to the more abundant clusters of car models. In almost every case and every metric, ELD performed worse than the NER models, our linguistic understanding.

DEPICT succeeded, but laboriously. Performance dropped from our previous analyses and settled on almost-static values of precision, recall and MAP. Despite the drops, however, our new model always registered statistically-significant gains in recall over the NER models. DEPICT only failed to identify participants once, in the glamorous setting of the Monaco Grand Prix. The footballers, artists and all other celebrities in attendance muddled the definition of participation, and DEPICT only resolved one driver, Charles Leclerc, himself a Monégasque. There, in Monte Carlo, DEPICT recalled just 16.13% of participants.

Model	Candidates	Locations	Governance	Parties	Other
TwitterNER	18.66%	16.33%	1.00%	1.33%	62.67%
$ELD_{TwitterNER}$	14.10%	12.82%	7.69%	4.49%	60.90%
$DEPICT_{TwitterNER}$	85.33%	5.33%	1.00%	1.00%	7.33%

Table 3.6: DEPICT gleaned a more general representation of participants than ELD, which restricted semantic drift. While ELD, like TwitterNER, varied the type of participants, erring often, DEPICT predominantly extracted election candidates.

The unvarying recall values indicate a limit to robustness. Wikipedia articles describe participants such as Lewis Hamilton, Nicholas Latifi and Pierre Gasly consistently as *racing drivers*, and DEPICT extracted drivers almost without fail. Constructors, however, do not follow a fixed template. Ferrari is a *racing division*, AlphaTauri is a *racing team* and *constructor*, and Aston Martin is a *car manufacturer*. Without a uniform writing style, DEPICT could not infer a typical definition of constructors. We hypothesize that the use of linguistic semantics, such as word embeddings, could reduce the reliance on writing styles, but we leave the idea for future work.

Yet even with its flaws, DEPICT remains more generalisable than ELD's model. Our final experiment pitted the two algorithms in the context of Canada's federal election of 2021, where the very concept of participation is ill-defined. Politicians and parties, provinces and ridings, and journalists and governmental institutions could all claim to participate to varying degrees. The domain gave ELD and DEPICT absolute freedom to interpret the matter of participation.

Most of the configurations remained unchanged. This time, we extracted named entities from a random sample of tweets published on election day, a brief exercise in retrospective APD, as we explain in Appendix D.1. We used TwitterNER to extract the top 50 named entities, which gave us sufficient information about participants and their nature. We also reduced the resolution threshold from 0.10 to 0.05, which resolved 27 participants, and increased DEPICT's candidate concepts from 200 to the 500 most-frequent links. In the end, we retained the top 300 concepts as participants, which we annotated using ground truth from The Globe and Mail, and other sources.

The gaps between models grew even wider, as Table 3.6 shows. ELD only identified 156 concepts, although it had the distinction of capturing every province and every major political party in Canada. Yet the ones principally concerned with the event, the hundreds of candidates vying for election, constituted fewer than a fifth of TwitterNER's and ELD's rankings. Of the rest, mostly journalists, retired politicians and other Canadian concepts, we could qualify few as outright irrelevant, but they reminded us of

Principal contributions

- The first in-depth analysis on how Twitter users converse about event participants before and during events
- The first study on how Twitter's NER annotator compares with traditional and targeted NER models on the social network
- DEPICT, a novel APD algorithm to not only identify but also understand participants

linguistic understanding: tangential concepts in the way of more befitting participants.

Undeterred by ambiguity, DEPICT sustained its advantage. Crucially, our novel algorithm learned differently. While ELD compared candidates with the domain, DEPICT compared candidates with the resolved participants. The final ranking reflected the algorithm's general understanding of participation: a slew of election candidates came first, then some provinces, ridings and other locations, and finally a few political parties and governmental institutions. Only 22 of the 300 concepts did not fit in any of our broad categories, and tellingly, 12 were the resolver's concepts: journalists, irrelevant politicians and incorrectly-disambiguated participants.

Inevitably, DEPICT still makes an assumption on the structure of the event domain. The attribute profiles assume that every participant follows a prototypical template. Nevertheless, DEPICT's assumption generalised better than ELD's across events and across domains. More importantly, its assumption did not consign the algorithm to trail behind the NER models. We conclude this chapter next.

Recap

Framed in the structure of events that Allan et al. [9] proposed, the 'five Ws and one H', McMinn and Jose [158] had reason to qualify participants as "the building blocks of events". Precisely because participants carry so much weight, they merit a better understanding than what linguistic understanding offers. Semantic understanding does better. Not only can APD models capture participants more precisely and more comprehensively than NER tools, but they can understand the participants themselves, unlocking new applications in TDT. In this chapter, we answered the following questions:

- What makes a named entity an event participant? A named entity becomes a

participant when what it represents and what it does affects or is affected by the event. In Section 3.1, we explored TDT literature's answers to the question to finally demonstrate that not all named entities are participants.

- How can a better understanding of participants improve our understanding of Who participates in events and Where? Distinguishing between named entities and participants requires us to understand, first and foremost, the participants themselves. In Section 3.2, we presented DEPICT, a novel APD algorithm that understands participants and their roles to glean the Who and the Where.
- How does APD refine the NER assumption, that named entities could substitute for participants? The APD assumption presumes that only certain named entities could substitute for participants and searches for the ones that do. In Section 3.3, we showed how even simple APD models can greatly outdo NER tools, but only a robust assumption can imitate the generalisability of NER models.

Despite our success, understanding the Who and the Where demands a certain unrealistic robustness of APD models. Participants change across most events, requiring an individual understanding of every single event. Conversely, an understanding of the What generalises across events, at least within domains. In fact, no form of understanding describes a domain's events better than the What [169]. Therefore in the next chapter, we develop our understanding of What may happen in events.

4

Understanding The What

A goal is a goal. A football fan has no need for FIFA's 74-word dull dictionary definition to understand what a goal is: they intuitively know that a goal represents an important concept in ways they could not express. A machine does not. An algorithm understands neither that a goal represents an important football concept nor why, and humans can hardly explain why either.

In Chapter 2, we proposed Automatic Term Extraction (ATE) to understand What happens in events. It would only be natural to expect literature on ATE, the task of identifying the words and phrases that characterise a domain, to master, and not simply understand, domain terms. It does not, however, understand them any better than TDT literature. The ATE community contends with the same fundamental question as the TDT community: when and how does a word become a domain term?

Throughout this chapter, we investigate What happens in events. We show that defining a term represents only the first of ATE research's problems. To the best of our knowledge, ATE's research community has neither studied event domains nor tweets, which differ considerably from its conventional domains and mediums. Yet we still conquer the difficulties with our novel method, EVATE, designed to extract terms from event domains on Twitter. In this chapter, we answer the following questions:

- What makes a word a domain term? Before understanding what terms describe an event domain, a machine must understand why a term is a term. In Section 4.1, we explore the attributes that distinguish a word from a term by exploring explicit definitions and implicit design choices in algorithms.

- How can ATE methods extract terms that make sense semantically? Terms represent semantically-important concepts, but a concrete measure of semantics has eluded researchers. In Section 4.2, we present EVATE, a novel algorithm to extract meaningful domain terms.
- How well can ATE techniques extract domain terms from Twitter? To the best of our knowledge, ATE methods have neither been applied to event domains nor to Twitter. In Section 4.3, we study the difficulties facing EVATE and other ATE techniques when extracting domain terms from football matches on Twitter.
- What roles do named entities play in slow-changing domains? To distinguish between domain terms and event terms, ATE algorithms depend on accurate and representative samples, but those can be a luxury. In Section 4.4, we question whether Formula 1's named entities qualify as domain terms or event terms, and address both possibilities with EVATE.
- Do ATE algorithms truly adapt to dynamic domains? ATE literature proclaims its algorithms the solution to minimising the burden of manually-listing the terms in changing domains. In Section 4.5, we debunk the claims in the volatile domain of American politics and propose a solution to create a generalisable and transferable political lexicon.

4.1 | What makes a term a domain term

Kubo et al. [118] filled their lexicon with words like *goal*, *shot* and *substitution*. We recognise these words as undeniably football-related terms, but what makes *goal* a term? It is difficult to justify its presence in the lexicon with more than abstract notions, like importance, but algorithms do not function with abstract notions. We need ways to quantify importance, around which the ATE task revolves.

The solution might seem obvious. If humans understand terms but machines do not, then let humans define lexicons; Kubo et al. [118] did, and so did Olteanu et al. [187] and Temnikova et al. [247]. Buntain et al. [28] disagreed. They imagined a machine that tracked football matches by following certain keywords but feared that a lexicon would inevitably miss terms, like how the lexicon by Kubo et al. [118] omits *penalty*. Buntain et al. [28] considered adding the missing terms but saw it as a never-ending process. When would they stop adding terms?

Ready-made alternatives exist, but they always carry the same caveats. In early research on bootstrapping, or the task of expanding lexicons, Roark and Charniak [219] ex-

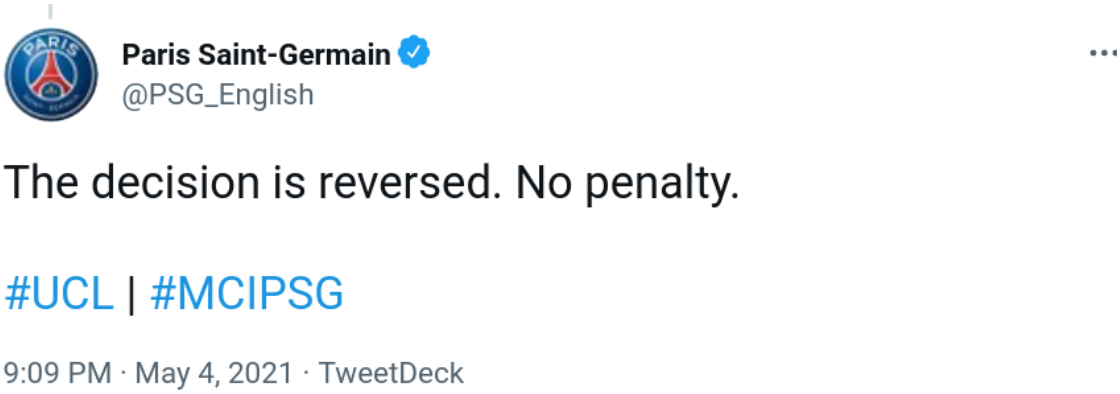

Figure 4.1: ATE research distinguishes between keyphrases and domain terms. In the tweet above, the word *penalty* is not the only keyphrase but the only domain term. The keyphrase extraction task makes no such distinction.

tracted terms from news corpora and compared them with WordNet. WordNet missed three out of every five correct terms. Since then, many others have noted the same insufficiencies in human-crafted resources [103; 214; 215; 241; 249], including domain-specific ones. A dictionary by UEFA [259] contains more than 2,500 football-related terms, dwarfing the 33 phrases that Kubo et al. [118] included in their lexicon, but still misses simple terms like *tackle*.

In part, a resource like WordNet can never be complete, both because it is a general knowledge base and because event domains change [14; 116; 208]. Football's rules have remained relatively stable for a century, but the sport has still innovated with technologies like the *goal-line technology* and the *Video Assistant Referee (VAR)*. Maintaining databases manually for changing domains remains a time-consuming, expensive, and ultimately still error-prone job [104; 135; 159; 202; 300], as Buntain et al. [28] observed.

Evidently, Buntain et al. [28] and the rest of the TDT community overlooked automatic alternatives, of which ATE could be one. Unlike keyphrase extraction, ATE extracts terms that describe an entire domain, not a single document [294; 295]. In Paris Saint-Germain's tweet, displayed in Figure 4.1, *decision*, *reversed* and *penalty* are all keyphrases, but only *penalty* describes What happens in the domain of football matches. ATE makes a similar distinction. Regrettably, however, we found little TDT research that integrated aspects from ATE; the closest works rely on manual annotation [187; 247]. Therefore we focus instead on ATE literature's broader efforts.

Moving from manual solutions to ATE requires defining terms to an algorithm, but even humans disagree on what makes a word a term sometimes [70; 155]. Literature treats terms as embodiments of important concepts [15; 24; 115; 117; 194; 268]: in an elec-

tion, it is not the lexeme *vote* itself that is important but the concept of participating in democracy. At the same time, such human-like concepts remain abstract notions [185], so the research area remains without an agreed-upon definition of terms [13].

Given the absence of a definition, we study more pragmatic interpretations. Research must approximate the notion of importance through some other means, conventionally by answering two questions: what can be a term? What makes a good term? These two questions align with ATE's two broad schools of thought, or the linguistic and statistical methods [111; 160; 176], even though the modern approach combines the two [140]. We discuss linguistic and statistical methods next.

The linguistic importance of terms

Linguistically, terms take many forms. We focus on single-word terms in this dissertation, but terms can also manifest as phrases [111; 115; 159; 232], like the terms *counter attack*, *free kick* and *right foot* that Kubo et al. [118] chose. In fact, Basili et al. [17] hypothesised that phrases constitute the majority of terms, and that simple, single-word terms act as shorthand to more technical multi-word terms, like using *yellow* instead of *yellow card* in football. Nakagawa and Mori [174] argued similarly.

Event tracking literature never quite reached such sophisticated reasoning about terms. Nevertheless, it did approach ATE research on the practical question of what can be a term. Often, TDT research assumes that terms are nouns and verbs [129], two POS tags that are "important to describe an event" [128]. Similarly, ATE research assumes that terms are nouns [13], although adverbs, adjectives [70] and verbs [115] are also considered on occasion.

However, neither TDT nor ATE research justifies why certain POS tags suit terms better than others [111]. ATE research scarcely considers verbs as terms, but is *shot*, a noun, not simply a footballer *shooting*, a verb? Even closed-class words can be terms; in football, it is common to describe players who have been expelled or substituted as being *off*, a preposition, and Kubo et al. [118] even included *far*, an adverb. In fleeting moments of self-introspection, the research community recognises the weak justifications of some of its choices [70]. More commonly, however, the ATE research community accepts the assumptions of linguistic filtering as a necessary compromise [70; 131].

Similarly, researchers skirt the question of whether named entities can or should be domain terms. Generally, named entities characterise the Who and the Where of events, or the event terms, not the What, as we explained in Section 2.3. Nevertheless, certain persons, organisations and places can appear ubiquitously enough to qualify as domain terms, as we argue in Section 4.4. Formula 1's governing body, the Fédération Interna-

tionale de l'Automobile (FIA), has been organising Grands Prix events since 1950. Its constant presence blurs the line between event terms and domain terms.

Some research addresses names explicitly. Park et al. [195] assign the highest possible score to named entities, and Lopes et al. [132] present named entities as essential components of domains. Few others follow. Most exclude named entities altogether from the design of their ATE algorithms. In fact, even as Velardi et al. [267] consider named entities as part of the terminology, their algorithm ultimately rejects them.

The statistical importance of terms

Whatever similarity exists between TDT and ATE extends only as far as linguistics. Event summarisation [223] and tracking [139] research blindly accepts POS tagging's output as unquestionably correct terms. Few question what makes a linguistically-valid word not merely a candidate term but a proper domain term. Conversely in ATE, linguistics only represent a starting point to be complemented by statistical measures [140].

Statistical methods measure termhood, or how well a word fits as a term [111; 127; 159; 294]. Inevitably, however, researchers realised that without a formal definition of terms, they would also struggle to explain what makes a good term [196]. Thus, ATE research returned to abstract notions of importance. We could summarise the efforts as the quest to materialise those abstract notions, to find the features that transform words into terms. In this section, we explore three such common features in termhood metrics: namely relevance, specificity, and consistency.

Relevance

First, a good term is relevant to its domain [99; 127; 131; 282]. Intuitively, *vote* represents an elections domain term because it holds a certain relevance to elections, but like importance, relevance remains too vague a concept. Therefore ATE methods estimate relevance in some other way, such as by using word frequency [196; 290]: the word *vote* holds relevance because it appears frequently in the context of elections.

Nevertheless, word frequency's convenient simplicity conceals naïvety. Word frequency assumes that only common words can postulate as domain terms, a bold assumption at both ends of the frequency spectrum. At one end, we can reasonably expect valid terms to appear frequently, but by definition, stopwords like *above*, *are* and *yourself* appear more commonly than any other class of words. So do abbreviations and profanity on Twitter. In short, frequent words do not necessarily represent terms [16; 127; 131; 135].

At the other end of the spectrum, uncommon words can also be terms [70; 111; 127; 185]. The word *impeachment* is a domain term in politics, but a rare one; before US president Donald Trump was first impeached in 2020, only two other presidents had been impeached in over 200 years. In fact, not only can infrequent words be terms, but Ha and Hyland [89] interpret rarity as a sign of technicality. In short, word frequency renounces semantics in favour of simplicity [127], which is why ATE techniques often also consider specificity.

Specificity

Second, a good term carries a specific meaning in a domain or within a community [24; 25; 175; 176]. Specificity manifests itself clearly in technical terms [25; 283]. What *impeachment* lacks in frequency, it compensates with its specific, legal definition that applies only to politics. In contrast, stopwords form the language's core vocabulary without being specific to any particular domain [4; 173]. Therefore specificity automatically bars them.

Chung [43] represented specificity as a four-point scale. The lower half of the scale includes function words and other words with a tangential relevance to the domain, like the word *flag* in politics. The upper half of the scale includes closely-related words and domain-specific words, which Chung [43] accepts as domain terms. The algorithm places words on this scale using a classic, ratio-based approach; the metric calculates the ratio of times that a word appears in the domain and in general, as formalised in Equation 4.1. Domain-specific terms have a high ratio because they appear disproportionately in the domain, whereas the ratio of function words falls closer to one.

Similar algorithms appear elsewhere in ATE literature. A few years after Chung [43], Park et al. [195] proposed an identical ratio-based algorithm, now called Domain Specificity. Domain Specificity compares the probability of a candidate term t appearing in any document from the domain, $p_{t,D}$, with its probability of appearing in a general corpus, $p_{t,G}$:

$$Specificity_t = \frac{p_{t,D}}{p_{t,G}} \tag{4.1}$$

More broadly, these algorithms form part of a family of approaches that rely extensively or exclusively on specificity: contrastive ATE. Contrastive approaches assume that termhood can be measured by comparing a word's appearance in one domain with its use in other domains or in general [115; 135; 185].

Notwithstanding our two examples, contrastive approaches do not have to rely on ratios. In the same year that Park et al. [195] proposed Domain Specificity, Kit and Liu

[111] proposed Rank Difference, which computes ranks, not ratios. More specifically, Rank Difference compares the rank of a word in a specific domain, $k_{t,D}$, with its rank in general, $k_{t,G}$; the higher the difference, the higher a word's termhood.[1]

$$Difference_t = \frac{k_{t,D}}{|V_D|} - \frac{k_{t,G}}{|V_G|} \tag{4.2}$$

Here, rank is a loosely-defined concept. Kit and Liu [111] calculate the rank using Term Frequency (TF), but any other ATE metric could substitute. Kit and Liu [111] also normalise the ranks using $|V_D|$ and $|V_G|$, the number of unique words in the domain-specific and general corpora. Such approaches sound intuitive, but like word frequency approximating relevance, contrastive approaches can overreach.

Contrastive approaches assume that domains share no overlap. Balachandran and Ranathunga [14], whose ratio-based technique resembles closely those of Chung [43] and Park et al. [195], assume that contrastive approaches can only work in highly-detached domains. Still, even separate domains can share terms. In this chapter we extract terms from football matches, Formula 1 Grands Prix and US politics. The three domains share little, but *red* still appears prominently in each: *red* cards in football matches, *red* flags in Formula 1 Grands Prix and *red* states in US politics. Contrastive approaches miss many valid terms when the assumption of specificity fails [185].

Specificity appears as an even bolder assumption in event domains, which ATE literature neglected. Consider Figure 4.2, which illustrates one way of mapping words related to football matches to the scale conceived by Chung [43]. The placement of certain keywords is admittedly subjective, but the domain-specific terms leave little room for doubt. Chung [43] interpreted domain-specific terms as words that associate with no other domains. However, no terms carry a more specific meaning to the domain of football matches than the names of footballers and clubs; *Liverpool F.C.* only has meaning in the domain of football matches.

More generally, points three and four on the scale delineate domain and event terms. The more specific a term in an event domain, the more likely it represents the Who and Where, or the event terms. Yet as we discussed in Chapter 2, event terms should distinguish between events, not paint a general picture of What happens in a domain. Consequently, the fourth tier in the scale no longer represents domain terms—a first sign of the frailty of ATE's most fundamental of assumptions in event domains.

Even so, specificity appears so much in ATE methods because generally it makes sense—just not in isolation. Techniques commonly apply specificity to compensate for

[1] In Equation 4.2, $k_{t,D}$ and $k_{t,G}$ represent ranks. Conventionally, 1 stands for the highest rank, but Kit and Liu [111] wrote the equation such that 1 stands for the lowest rank to make the concept clearer. We stay faithful to their decision.

Figure 4.2: The specificity scale does not hold in event domains. Chung [43] accepted words in the third and fourth tiers as domain terms, but in TDT, the fourth tier represents event terms.

relevance's simplicity. TF-IDF, a conventional metric, combines relevance, TF, with specificity, IDF [131; 135]. As we show in this chapter, the simple combinations of TF-IDF and its variants remain powerful baselines. In fact, both Hua et al. [99] and Zhou et al. [298] use TF-IDF and offshoots of it in a TDT architecture to extract event-related terms from news reports.

Naturally, TF-IDF remains a general technique borrowed from IR literature, unaware of ATE's sensibilities. Lopes et al. [131] took aspects of TF-IDF and moulded them to suit ATE's needs. The resulting metric, Term Frequency-Disjoint Corpora Frequency (TF-DCF), compares the presence of a candidate term t in a specific domain, the term frequency $TF_{t,D}$, with its presence in several other, contrasting domains, D':

$$TF\text{-}DCF_{t,D} = \frac{TF_{t,D}}{\prod_{\forall d' \in D'} 1 + \log(1 + TF_{t,d'})} \tag{4.3}$$

The slight variation on TF-IDF allows Lopes et al. [131] to interpret specificity more leniently than ratio-based measures. By design, TF-DCF accepts that a term may belong to more than one domain but penalises words that appear in many areas. Nevertheless, while TF-DCF and similar techniques perform well, they miss another important aspect

of termhood: consistency.

Consistency

Third, a good term appears consistently throughout the domain. Consistency has not earned the same prominence as relevance and specificity, but it develops the former: a good term is not merely relevant to a domain but consistently relevant. From a slightly different perspective, a term that appears in one document cannot be relevant to a domain [202]. While not prominent, consistency has long accompanied ATE literature. Presenting Domain Consensus, one of the field's earliest termhood measures, Velardi et al. [267] argued that a domain's community should agree on the term's use and therefore apply it uniformly. Consistency captures that aspect of terms.

Since consistency is so closely-tied with relevance, the two inevitably share similarities. While word frequency captures relevance, document frequency captures consistency [202]. The two also share shortcomings. Despite rarely representing terms, stopwords and function words do not only appear frequently but also consistently. Inversely, inconsistent words can be terms too; few football matches have *penalty shootouts*, but when they do, little else matters.

Still, we consider consistency to be a cornerstone of domain terms. We do not interpret consistency as the consensus within a community but as the property of termhood that distinguishes event terms from domain terms. Event terms, the Who and Where, only describe a subset of events, but domain terms appear consistently throughout the domain. *Liverpool F.C.* only plays a few football matches a year, but *goals* remain a central theme in all football matches even without *Liverpool F.C.*'s involvement. In other words, the difference between event terms and domain terms lies simply in consistency.

In their own unique ways, linguistic and statistical principles try to give a formal structure to terms. In reality, they only get us a little closer to understanding them. Linguistic filtering tells us that terms are likely to represent nouns without justifying why [70]. Statistics tell us that domain terms appear frequently but maybe infrequently too, that terms are specific to a domain but not necessarily either, and that terms appear consistently except for the ones which do not. Linguistic and statistical qualities get us close to but miss altogether the essence that transforms a word into a domain term: semantics.

A goal does not represent an important concept because it appears frequently or consistently in football matches, nor because it is specific to a domain. A goal represents

an important concept because it gives meaning to the game—it describes What happens in events. In other words, a goal's significance lies in an abstract notion of semantic importance that neither linguistics nor statistics adequately capture.

Few methods address semantics. Qureshi et al. [210] construct a graph of Wikipedia categories to elect domain terms, and Meijer et al. [160] follow up ATE with word disambiguation using a taxonomy. More recently, Zhang et al. [294, 295] proposed two general re-ranking algorithms, both based on word similarity, to re-rank a base ATE technique's output. Still, you could not claim that either captures semantics.

Evidently, capturing a notion as abstract as meaning poses a difficult task [155]. In fact, explicit uses of semantics remain rare even in contemporary ATE literature [294]. Yet what ties the above approaches together is the manner in which they try to quantify semantics: with statistics. Numbers cannot easily measure the role of words, distinguish between one that has a mere relevance and one that describes What happens in events. In the next section, we propose EVATE as an alternative that approaches semantics from a novel perspective, linguistics.

4.2 | EVATE: EVent-Aware Term Extractor

A young child sits down to watch a football match for the first time. The child does not understand the significance of a goal yet, but they will soon. And in a few years' time, the young child will have become a football fan who understands goals, the offside rule, and maybe even why VAR ruled against their favourite team. In this section, we present EVent-Aware Term Extractor (EVATE), a novel ATE approach inspired by how humans learn about event domains.

EVATE is, to the best of our knowledge, the first ATE algorithm designed for event domains. Our technique combines principles of linguistics with principles of statistics to learn domain terms by observation, just like a human [66]. While we design EVATE's statistical metric around the structure of event domains, we innovate best in the linguistic component, to which we assign the role of extracting semantically-meaningful candidate terms. The statistical component simply orders words based on their relevance, specificity and consistency in the domain. We describe these components in detail next.

Extracting semantically-meaningful words

A word becomes a term when it carries a semantic significance. Unlike its TDT counterpart, the ATE community never pretended that linguistics could lead to a semantic understanding, and with good reason. POS tagging, which traditionally drives linguis-

tic components, could never aspire to capture the precise semantics of the 33 terms that Kubo et al. [118] hand-picked. Instead, research implements semantics in the statistical component, seemingly by default and without much success. EVATE injects semantics differently, through the linguistic component, but before we present our method, we reflect on what gives meaning to a concept in event domains.

Regular domains differ from event domains. In regular domains, almost anything can be a concept, but in event domains, the most meaningful concepts are the ones that change the state of the event. The earliest works in TDT literature defined events in terms of What happens in them [9], and the automata that Kleinberg [113] conceived, which would eventually inspire feature-pivot techniques, revolve around the transitions from an event at rest to an event in motion. In short, event domain concepts are represented by topics, which we define as follows:

Definition 6 (**Topic**)**.** An event within an event, a sub-event which we could also describe as a function of Who did What, Where and When, and Why and How.

EXAMPLE: Alexandre Lacazette [Who] scores [What] a penalty [How] after 22 minutes [When].

We describe topics, the concepts, through topical keywords, which distil the essence of an incident: the 'five Ws and one H'. Since topics represent the concepts of event domains, topical keywords analogously represent the event terms and domain terms of event domains. We define topical keywords as follows:

Definition 7 (**Topical keyword**)**.** The lexemes or phrases that describe Who did What, Where and When, and Why and How.

EXAMPLE: *Alexandre Lacazette*, *scores*, *penalty*, *22*.

Because topics represent concepts and topical keywords represent terms in event domains, EVATE replaces POS tagging with a TDT algorithm. We task the TDT algorithm with tracking events to extract topics and the associated topical keywords: EVATE's semantically-meaningful candidate terms. To construct a comprehensive lexicon, we require a TDT algorithm with a feature-pivot component, and that extracts topics precisely and comprehensively—both key and non-key topics.

ELD [143; 146], which we described in Chapter 2, fits all criteria. ELD uses a standard document-pivot method to cluster tweets and a feature-pivot technique to identify the topical keywords in each cluster. Combined in such a way, the document-pivot and feature-pivot approaches allow ELD to report about events with a fine granularity without succumbing to excessive noise. Therefore we run ELD on several events, extracting topical keywords from each. Then, we rank them using EVATE's statistical component.

Measuring termhood

The linguistic component generates a list of words that carry a semantic weight in event domains. ELD, and by extension EVATE, does not filter words based on POS tags, but still, it filters out stopwords, function words and other closed-class words as long as they hold no significance. The role of the statistical component changes accordingly. Termhood ceases to be the arbiter of semantics and instead assumes that all words have a semantic bearing—some more than others.

In this work, we present a novel termhood measure tailored to event domains. EVATE's statistical metric combines the three desirable characteristics of domain terms described in Section 4.1: relevance, specificity and consistency. The flaws in the three attributes' assumptions remain, but the linguistic component's semantic basis strengthens the premise. EVATE estimates the termhood of a candidate term t by multiplying three scores together, ensuring that any zero value disqualifies the term:

$$EVATE_t = EF_t \cdot ICF_t \cdot Entropy_t \tag{4.4}$$

The three components play distinct but mutually-reinforcing roles. First, Event Frequency (EF) estimates relevance by counting the number of events in which a word appears as a topical keyword. Second, Inverse Corpus Frequency (ICF) approximates specificity by comparing a word's usage in the domain with its usage in general. Third, Entropy measures a word's consistency in the domain. In the end, EVATE ranks words in descending order of score and accepts the top words as terms. We describe EF, ICF and Entropy in the rest of this section.

Event Frequency (EF)

Event Frequency (EF) measures relevance. EF represents our biggest departure from popular ATE methods. Standard word frequency can be misleading, as we explained in Section 4.1, but EF tailors the measure to event domains and Twitter. As the name implies, EF counts the number of events, E, in which the TDT algorithm identified the candidate term t as one of the event's topical keywords, $T(e)$:

$$EF_t = \log_{10} |\{e \in E | t \in T(e)\}| \tag{4.5}$$

We assign EF the role of subduing the effects of user behaviour. TDT methods miss non-key topics, like yellow cards in football matches, precisely because they are unpopular among Twitter users and thus scarcely feature in tweets [150]. Even certain key

topics, like red cards, appear infrequently despite their importance. By definition, however, EF does not require an intense discussion surrounding a topical keyword—a word either describes a topic as a topical keyword or it does not.

EF also subdues the effects of the event domain's structure. We consider the event frequency, not the topic frequency, which would place an unfair disadvantage on topics with an upper limit on frequency. Consider Formula 1 Grands Prix. A driver can enter the pit lane several times, but a race rarely requires more than one formation lap. In other words, *pit stops* would appear as topical keywords far more frequently than *formation laps* just by virtue of the event domain's rules. By definition, however, EF does not require topics to happen frequently in any one event—only frequently enough throughout the event domain.

Our final design choice in EF is to take its logarithm, not its raw frequency. As Lopes et al. [131] so eloquently put it, "a term t that occurs 10 times is not 10 times more important than a term t' that appears only once." Likewise, humans do not need to renew and reaffirm their belief about a topic's importance in every event. After observing the same topic a few times, humans permanently commit its importance to memory. The logarithmic EF reflects the same behaviour.

Simultaneously, the logarithmic EF helps EVATE overcome one of ATE's major challenges: dynamic domains. When the English FA introduced the brand-new VAR technology to the English Premier League in 2019, football fans quickly grasped that it would change the game. Similarly to human memory, the logarithmic curve grows quickly as the method learns and then slowly as it consolidates knowledge, allowing rare or new terms to catch up. Note that the logarithmic base scales the scores but does not affect the ranking order.

Inverse Corpus Frequency (ICF)

Inverse Corpus Frequency (ICF) measures specificity. Differently from standard ATE approaches, we found it infeasible to apply Inverse Document Frequency (IDF) on tweets without any changes. The brevity of tweets constrains term frequency to not more than one in most tweets [153], and standard IDF arbitrarily promotes uncommon words. In this work, we replace IDF with ICF, the second component of Term Frequency-Inverse Corpus Frequency (TF-ICF) [213], and add Laplace smoothing:

$$ICF_t = \log \frac{|G|}{|\{d \in G | t \in d\}| + 1} \tag{4.6}$$

We calculate ICF in the same way as IDF for a candidate term t, by counting the number of documents d in which the word appears. Differently from IDF, however, ICF

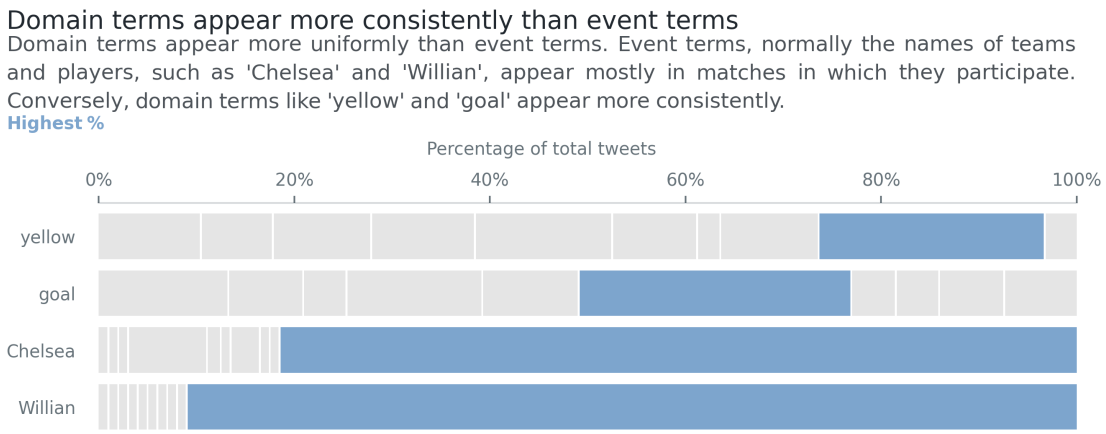

Figure 4.3: Domain terms appear more consistently than event terms. Domain terms appear in most events, whereas event terms appear only in events in which they are participating.

counts the frequency on a different, static corpus, G, not on the domain corpus. Our static corpus contains 457,429 English tweets collected using the Twitter Sample API over 12 hours. This corpus of general tweets effectively turns EVATE into a contrastive ATE approach that compares the frequency of a word in the event domain with its use in general. We exclude retweets when calculating ICF to minimise bias. We describe the general corpus in more detail in the introduction of Appendix D.

Entropy

Entropy measures consistency. A football fan does not learn the names of teams and players first; they change from one event to the other. A football fan first learns the fundamental rules, from the law governing goals to the more technical one regulating offsides, because they occur in most matches. The contrast delineates the separation between event terms and domain terms, which entropy seeks to establish.

Event terms and domain terms distribute differently. As shown in Figure 4.3, the event terms *Chelsea* and *Willian*, then a Chelsea player, appear disproportionately in Chelsea's matches. Conversely, the domain terms *yellow [card]* and *goal* distribute more uniformly across events. Entropy identifies domain terms from such differences. We adapt the standard entropy equation such that $p_{t,e}$ represents the probability a candidate term t appears in event e among all events E. To calculate $p_{t,e}$, we count the term's document frequency in the corpus of event e, $D(e)$, and divide by the term's total document frequency across all events:

$$p_{t,e} = \frac{|\{d \in D(e) | t \in d\}|}{\sum_{e'}^{E} |\{d \in D(e') | t \in d\}|} \tag{4.7}$$

$$Entropy_t = -\sum_{e \in E} p_{t,e} \log p_{t,e} \tag{4.8}$$

In Equation 4.8, entropy assigns a lower score the less regular a term's distribution, or when the term appears with a high probability in a few events and a low probability in the rest. Conversely, entropy assigns the optimal score when a term distributes evenly across all events. To minimise redundancy, we exclude retweets when measuring $p_{t,e}$ in Equation 4.7.

While conceptually similar, EF and entropy complement each other. On the one hand, EF measures relevance by considering the number of events in which a candidate term appears as a topical keyword. On the other hand, entropy considers all events, ensuring that a candidate term remains relevant even when it does not appear as a topical keyword. A football match might end goalless, but Twitter users will still talk about *goals* in their absence—perhaps more so.

By design, EVATE captures relevance, specificity and consistency. Above all, it captures the semantics that have for so long eluded ATE research. Yet the metric's pursuit of meaning in terms does not come without limits or disadvantages. EVATE draws semantics from topical keywords, a concept unique to ATE but which depends on TDT. Without topical keywords, EVATE loses not just its linguistic component's semantics but also the foundation of its statistical measure, EF.

In fact, the fusion of EVATE with a TDT algorithm ties the former's limits closely with the latter's. ELD, like most other feature-pivot techniques, only extracts unigrams, not n-grams—*yellow* and *card* separately, not as one—and some noise inevitably escapes ELD's scrutiny. We could argue similarly about POS tagging, but optimising TDT algorithms proves far costlier than optimising the former. POS tagging requires one efficient pass over documents, whereas TDT algorithms tend to be highly parametric, slow to execute and expensive to evaluate, as we explain in Chapter 5 and Appendix A.

Finally, EVATE requires careful curation of its data. Given a set of events with a singular perspective of the domain, entropy would lose all meaning, and with it, EVATE would lose the ability to distinguish event terms from domain terms. The events should provide a generalisable view of the domain, allow domain terms like *yellow* and *goal* in Figure 4.3 to climb above the rest. In fact, more than just varied data, EVATE requires copious data. The logarithmic EF must observe a word as a topical keyword in at least two

events to assign a non-zero score, but as we show in Section 4.5, EVATE keeps learning from new datasets for a long time. Still, we make all these sacrifices for semantics.

In the rest of this chapter, we study EVATE's behaviour in different types of domains. In Section 4.3, we compare our technique with four common baselines to understand ATE's general behaviour on Twitter and in the domain of football matches. Then, in Section 4.4 we question the fine line between event terms and domain terms in Formula 1 Grands Prix. Lastly, in Section 4.5 we examine the performance of ATE methods in the dynamic domain of American politics.

4.3 | A goal is a goal: the language of the beautiful game

In 1990, the Italian Alps hosted the second World Cup semi-final. Germany knocked out England. After the match, Gary Lineker, whose goal for England was not enough to take his country to the final, reacted with an iconic quote: "Football is a simple game—22 men chase a ball for 90 minutes and at the end, the Germans always win." The conclusion is not true, of course; Lineker revised his quote years later when Germany was unceremoniously knocked out of the 2018 FIFA World Cup in the group stage [3], but he always kept the first part: football is a simple game.

Football is a simple game because it has a rigid structure. Before a match starts, we know that 22 players will spend 90 minutes trying to score goals and avoiding to concede them. As a result, as we explain in Chapter 5, the rigid structure and sheer popularity of football has made the sport a popular domain for evaluations in TDT literature. In this section, however, we focus on football for a different reason: football represents an ideal event domain for ATE.

Describing the ideal setting for ATE, Wong et al. [282, 283] pushed for "balanced, unbiased and randomised" datasets. Similarly, Luo et al. [135] emphasised the need for varied but thematic documents. Football, with its broad popular support and abundance of events [28], embodies those ideals. For the experiments in this section, we collected more than 4.5 million tweets from 24 football matches, each split into an understanding period and an event period. ELD uses the one-hour understanding period before the match to understand the event, and the event period to extract topics and topical keywords. Appendix D.2 includes more details about the datasets.

We compare EVATE with four baselines. The baselines, all of which we introduced in Section 4.1, represent ATE's two broad schools of thought. The first two represent the frequency-based methods, which we refer to as the general methods for reasons that will become clear later. The last two represent the purely contrastive, specific methods:

- TF-ICF compares the frequency of a candidate term in a specific domain with its frequency in a general domain [213]. TF-ICF represents our adaptation of TF-IDF to Twitter's brevity by calculating the IDF component on a static, general corpus. The second component, ICF, also features in EVATE.

- TF-DCF compares the frequency of a candidate term in a specific domain with its frequency in several others [131]. In this work, we only use one other domain, a general one, to compute TF-DCF.

- Domain Specificity contrasts the use of a candidate term in a specific domain with its use in general [195]. Domain Specificity represents the classical, ratio-based termhood measures like the one proposed by Chung [43].[2]

- Rank Difference contrasts the rank of a candidate term in a domain-specific corpus with its rank in a general corpus [111]. Like Kit and Liu [111], we rank words in the domain-specific and general corpora using term frequency, but we initially set a minimum cut-off point of 100 occurrences to eliminate the rare and out-of-dictionary words so rife on Twitter.

We describe the approaches by Kit and Liu [111] and Park et al. [195] as contrastive, but in reality, all baselines and EVATE have a contrastive element. In all cases, we use the sample dataset described in Appendix D as the general domain corpus with which to contrast the domain. The baselines share a common linguistic component, which extracts stemmed nouns, verbs and adjectives using the NLTK library [22].

Our evaluation has two parts. First, in this chapter we evaluate terms directly by comparing them against a gold standard, one of ATE's two predominant evaluation methodologies [13; 111]; the alternative, manual annotation, proves to be both costly and subjective. Second, in Chapters 5 and 6 we evaluate the effects of EVATE's terms on TDT algorithms, a form of indirect evaluation [111].

In this chapter's direct assessment, we compare the lexicons with two ground truth lists. First, we compare the outputs with a dictionary published by UEFA [259], containing over 2,500 technical terms related to football. Second, we counterbalance the technicality of the first dictionary with the more colloquial, crowd-sourced glossary on Wikipedia [281]. We filter both dictionaries to retain only single-word terms.

[2]Terms that never appear in the general corpus, usually named entities or misspellings, receive an infinitely-high score from Equation 4.1 as the denominator equals 0. Park et al. [195], who took a lenient, inclusive view of named entities, accepted such terms and ranked them highly. Differently from Park et al. [195], we reject terms that only appear in the domain corpora, both because named entities tend to represent event terms and because noisy out-of-vocabulary terms abound on Twitter.

Using our ground truths, we measure two aspects of quality. The first aspect is the overall quality of the term lists [13; 117; 159; 196], which we measure using precision, recall and the F-score. We only consider the top 200 terms, after which we observe quality to degrade, with fewer domain terms, their place taken by noise.

The second aspect is the quality of the ranking's order. Given the semantic importance of the term *goal*, we do not expect an algorithm to simply extract it but to extract and rank it highly, above noisier words. While research traditionally uses AP [13] to measure ranking quality, the sheer size of our ground truth relative to our lexicons reduces the metric to very low, incomparable values. Therefore we use AP sparingly in favour of the more expressive P@k. In the rest of this section, we analyse the performance of EVATE and the four baselines, starting by studying ATE's behaviour in the domain of football matches on Twitter.

Twitter is a difficult medium

Like Twitter changes most other IR tasks, it changes the ATE problem. The high precision and recall values of ATE research disappear on Twitter, replaced instead with the poor performance in Table 4.1. On formal documents, Rank Difference boasted a state-of-the-art precision of 97% [111]; on Twitter, Rank Difference's precision only reached 42.50%, and only after heavy filtering. The other methods performed even worse. Twitter is a difficult medium.

Twitter's noise hindered ATE algorithms in all its forms. We observed POS tagging fraught with mistakes, weakening algorithms and producing the poor results in Table 4.1a. The errors in the POS tags, fruits of Twitter's disorderly orthography, pushed the English club *Arsenal* to among the top 10 terms in TF-ICF's and TF-DCF's lexicons. Despite the mistakes, however, the baselines' weak understanding of semantic value made POS tagging an invaluable element, without which all four methods performed worse. Even on Twitter, ATE research does not afford to exclude linguistic filtering.

Yet misspellings, hurried mistakes, only account for a small part of the algorithms' errors. To adapt to brevity, Twitter users developed an informal vernacular. Many users prefer the short *mid* to *midfielder* and refer to skilled footballers as *ballers*. Nowadays, acronyms, abbreviations and informal words form an important and accepted part of how Twitter converses. While language evolves, tradition changes slowly. To UEFA and Wikipedia's formal dictionaries, the new vocabulary represents inferior, informal language to be rejected.

Because of the glossaries' own shortcomings, the poor performance in Table 4.1a looks worse than it actually is. Twitter's vernacular may be informal but not wrong. The

Algorithm	P@50	P@100	Precision	Recall	F-score
EVATE	50.00%	36.00%	29.00%	7.03%	11.32%
TF-ICF	42.00%	29.00%	23.00%	5.58%	8.98%
TF-DCF	34.00%	25.00%	20.50%	4.97%	8.00%
Domain Specificity	16.00%	17.00%	21.50%	5.21%	8.39%
Rank Difference	26.00%	23.00%	20.50%	4.97%	8.00%

(a) The performance of ATE methods, using all tweets except retweets, does not compare with results on more traditional corpora.

Algorithm	P@50	P@100	Precision	Recall	F-score
EVATE	50.00%	36.00%	29.00%	7.03%	11.32%
TF-ICF	46.00%	34.00%	26.00%	6.30%	10.15%
TF-DCF	38.00%	29.00%	24.50%	5.94%	9.56%
Domain Specificity	22.00%	21.00%	19.50%	4.72%	7.61%
Rank Difference	32.00%	32.00%	25.00%	6.06%	9.76%

(b) The performance of the ATE baselines improved when we included retweets, although it remained far below their results on more traditional corpora.

Algorithm	P@50	P@100	Precision	Recall	F-score
EVATE	50.00%	36.00%	29.00%	7.03%	11.32%
TF-ICF	48.00%	43.00%	31.50%	7.64%	12.29%
TF-DCF	46.00%	38.00%	30.00%	7.27%	11.71%
Domain Specificity	32.00%	25.00%	22.50%	5.45%	8.78%
Rank Difference	36.00%	37.00%	42.50%	10.30%	16.59%

(c) The ATE baselines performed best when we used only tweets by verified users. For the first time, all methods except Domain Specificity out-performed EVATE in various metrics.

Table 4.1: On Twitter, ATE algorithms require fine-tuning. Performance improved when we included retweets, and especially when we restricted the input to the authoritative content of verified users. We could not change ELD's tweet filters, so EVATE's results did not change either.

Twitter community widely adopted the colloquial mannerisms to the point that even formal accounts sometimes use acronyms, abbreviations and other informal language. Therefore even though EVATE and the baselines fail to capture the formal terminology of glossaries, they learn, at least, how Twitter users converse about events.

Aside from the informal, glossaries cannot capture every single formal term either. UEFA's dictionary has space for the spectators' *boos* but not *VAR*, and while the list includes five types of tackles, it misses the term *tackle* itself. Furthermore, our algorithm and baselines capture multi-word terms as single-word terms. Instead of the glossaries' *yellow card*, we capture *yellow* and *card*; instead of *free kick*, we capture *free* and *kick*. Our unigrams do not feature individually in the ground truth, to the detriment of precision.

Nevertheless, the study of tweets can improve performance. The gradual improvements in Table 4.1b's results highlight the need for more ATE research on social media content. Consider retweets, which generally improve the quality and order of lexicons. Research, in particular in TDT, avoids retweets on the premise of bias [225], but literature rarely describes what bias it avoids. Logically, bias favours content by authoritative users, like journalists, whose huge following attracts retweets. In short, retweets promote authoritative content and thus improve results.

Results improve further with more thoughtful considerations. Using only tweets by verified authors led to even more impressive gains in performance. Until 2020, verified users comprised a relatively exclusive group of prominent accounts, comprising journalists, companies and other public figures. Twitter opened applications for verification in 2021, but getting the coveted blue verification tick requires users to have a high following, usually reserved for authoritative users.

Our baselines thrived with tweets by verified users. Authoritative users tend to write authoritative content, and authoritative content tends to have the good orthography that characterises formal documents. POS tagging, now in its element, made fewer mistakes. As shown in Table 4.1c, Rank Difference, this time excluding words that appear fewer than 50 times, achieved the highest precision score: 42.50%. TF-ICF and TF-DCF performed worse but still surpassed EVATE, at least in overall quality.

Yet by imitating the more proper language of verified users, the baselines traded away Twitter's casual vernacular. Words like *pen*, short for *penalty*, disappeared from the lexicons, and so did *book* and *gol*, informal ways of referring to yellow cards and goals. The lexicons became slightly less representative of Twitter, and thus, slightly less relevant to TDT on Twitter. EVATE made no such trade-off: it embraced Twitter's language at the cost of performance, and even then, our method obtained a comparable precision and recall to TF-ICF's and TF-DCF's. Only Rank Difference greatly out-performed our method with a considerable gap.

The performance results seem to clash with our assertions. Table 4.1 shows linguistic understanding outperforming our presumed semantic understanding, EVATE. A difference still exists, a subtle presence that you could miss by only observing the figures. There exist matters of performance—how well the five methods learned—and matters

of substance: what the five methods learned. As we demonstrate in the next analysis, the five methods learned vastly different things about football.

Of specific, general and semantic methods

The figures in Table 4.1 imply that some methods simply perform better than others. Domain Specificity generally performs poorly, whereas EVATE, TF-ICF and TF-DCF perform similarly to each other. The assertion holds some truth—certain methods really do perform better than others—but the simplistic view obscures the differences among the lexicons. In these experiments, we examine the effects that the statistical and even linguistic choices have on the lexicons.

In Figure 4.4, EVATE and the four baselines, using only tweets by verified authors, diverge into three groups.[3] Rank Difference shares the most overlap with TF-ICF and TF-DCF, but it shares more meaningful characteristics, notably the types of mistakes, with Domain Specificity. Therefore we consider the first group to include Domain Specificity and Rank Difference, which favour highly specific terms, like *goalkeeper*, event hashtags, and team and player names. The second group includes TF-ICF and TF-DCF, whose similar design based on word frequency, extracts more general terms, like *football*, *team* and *game*. The third group, completely isolated from the others, includes only EVATE, which builds markedly different lexicons than the baselines.

The first group, with Domain Specificity and Rank Difference, produced highly-specific lexicons. As we argued in Section 4.1, few terms have the specificity and distinguishing power of event terms, normally the participants: Who is involved in an event or Where the event is located. Whenever the POS tagger erred, it exposed specificity's weakness to named entities. Rank Difference might have achieved a best-performing precision of 42.50%, but it interspersed domain terms among team and player names. Despite boasting the highest precision of all methods, Rank Difference trailed EVATE by 14.00% (50.00% ↓ 36.00%) in P@50.

The specific methods failed at the other extreme too. When a term did not fulfil the assumption of specificity, Domain Specificity and Rank Difference relegated it to the end of the lexicon. In Rank Difference's list, *goal* only appeared at rank 119. Other common terms, including *yellow* and *card*, appeared even further down among the top 200 terms or not at all, like *red*. Domain Specificity's metric, even more simplistic than Rank Difference, accentuated the problem.

[3]TF-ICF and Rank Difference consistently perform best in their groups. For brevity and clarity, throughout the rest of this chapter we focus extensively on the two algorithms and EVATE. However, unless otherwise specified, our observations of TF-ICF apply to TF-DCF too, and our observations of Rank Difference apply to Domain Specificity too.

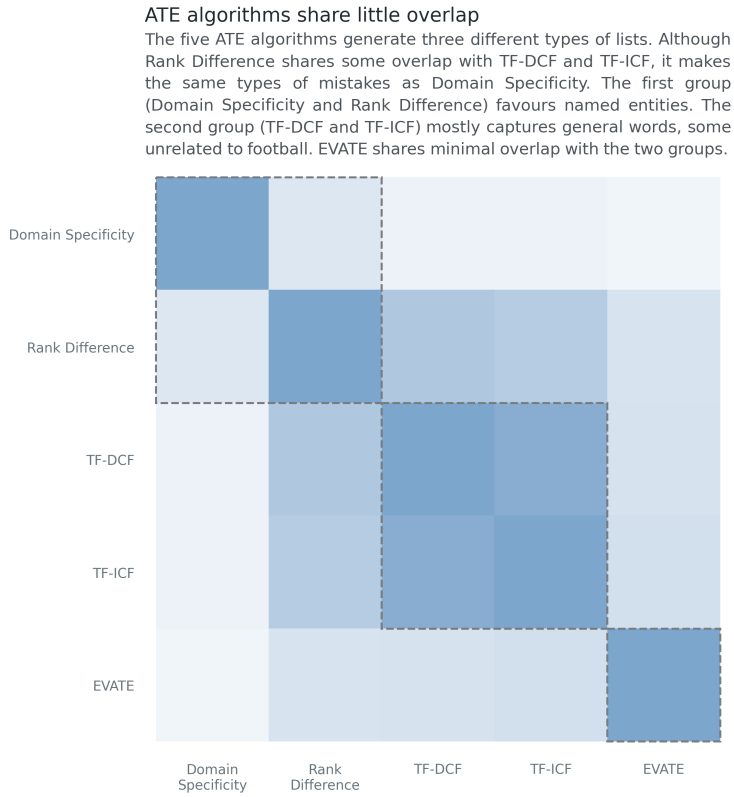

Figure 4.4: The five methods build three types of lexicons. Domain Specificity and Rank Difference construct specific lexicons, and TF-DCF and TF-ICF construct general lexicons. EVATE constructs topical lexicons.

In contrast, the second group, with TF-ICF and TF-DCF, tended to generate overly-general lexicons, to the detriment of specific terms. Because these methods depend heavily on word frequency, general and frequent terms climbed over the more specific but less frequent terms. TF-ICF's top four terms comprised *goal*, *score*, *game* and *player*. Without context, you could easily mistake the domain for hockey.

EVATE lay somewhere in-between the two groups. Our algorithm avoided event terms without gleaning a superficial understanding of the domain. *Kepa*, the name of a Spanish goalkeeper, was the highest-ranked event term in its lexicon, but even it only appeared at rank 42 due to low EF and Entropy scores. Instead, EVATE filled the higher ranks with domain terms, though without the triviality of TF-ICF's and TF-DCF's lists. In fact, out of EVATE's top 10 terms, only 2 appeared among TF-ICF's top 100 terms: *goalkeeper* ($7^{\text{th}} \downarrow 89^{\text{th}}$) and *foul* ($9^{\text{th}} \downarrow 75^{\text{th}}$).

To confirm that EVATE builds a more technical lexicon than TF-ICF, we adapted the definitions of technicality by Ha and Hyland [89] to WordNet. Ha and Hyland [89]

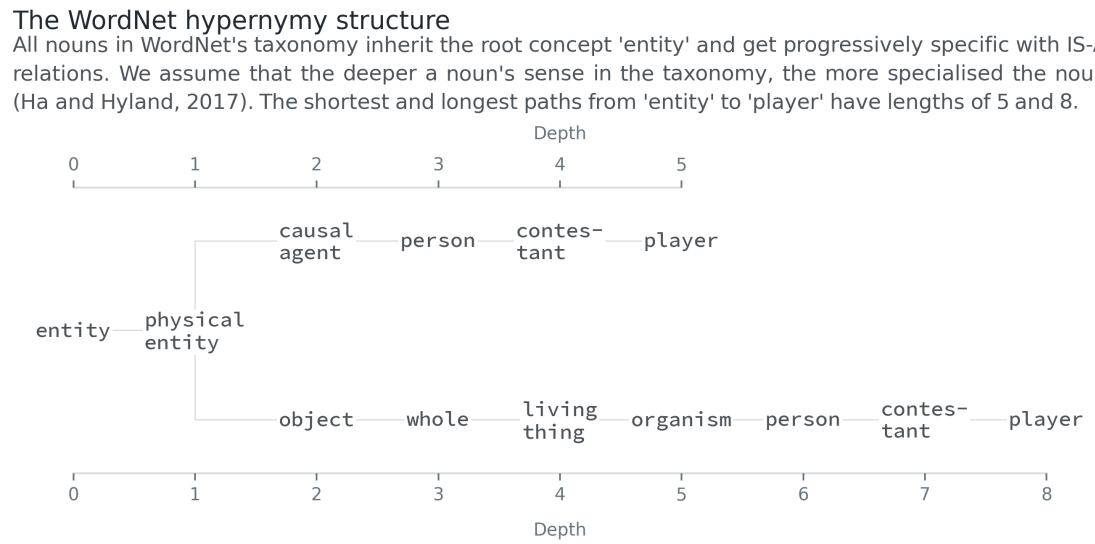

Figure 4.5: WordNet represents concepts in a hypernymy structure. Similarly to Ha and Hyland [89], we assume that the higher a concept appears in the taxonomy, the more general it is, and vice-versa.

described technical words as monosemous, so we calculated the mean number of senses of EVATE's and TF-ICF's top 20 terms from WordNet. We found that the average term in TF-ICF's top 20 terms had 13.06 senses, more than twice as many meanings as EVATE's top 20 terms, which had 5.47 senses on average. The term *score*, for example, has 18 senses, whereas *keeper* or *goalkeeper*, both highly-ranked by EVATE, only have 2 senses.

Ha and Hyland [89] also noted that technical words have specialised senses. To test this property, we manually mapped EVATE's and TF-ICF's top 20 terms to their closest WordNet sense, if available. Then, we measured the minimum and maximum depths of each sense from the root concept in the hypernym taxonomy. For example, as shown in Figure 4.5, the term *player* originates from the root concept *entity*, ancestor to all nouns, and forks into two hypernym paths. Again, TF-ICF's terms had a shallower depth, with the average minimum and maximum depths ranging from 6.57 to 6.93, as opposed to EVATE's ranges, which went from 8.31 to 9.23.

EVATE's performance problems, then, do not lie in substance but in form. As we intuited previously, many of EVATE's terms were not incorrect but missing from the ground truth. Twitter's informal language seemed to worsen performance. While 6 of Rank Difference's top 10 terms were event terms—the names of teams or related hashtags—only 1 of EVATE's top 10 terms was clearly incorrect: *FFS*, an expression of frustration. The remaining four incorrect terms included two informal words, *baller* and *gol*, and two missing ground truth terms, *VAR* and *yellow*. Rank Difference erred with

event terms, while EVATE erred with casual domain terms.

The question follows naturally: which group best describes the domain of football matches? First, the answer depends on the medium. Rank Difference's formality when using tweets by verified authors may have appeased UEFA's dictionary, but on Twitter, informality reigns. The informal word *pen* does not appear in the ground truth, but when throngs of users mention the word, it becomes significant. A TDT algorithm depends on the language of the common Twitter user as much as—or even more than—the language of journalists and clubs to detect topics.

Second, the answer depends on what the application needs. Topics in the same event share a general vocabulary [161], so TDT algorithms rely on the specifics to distinguish among topics. Consider when the German national team thrashed Brazil 7–1 on 8 July 2014, scoring four goals in six minutes. The goals included a rebound, a long shot and two counter-attacks, but they shared a common vocabulary: Germany scored a goal. The common description could have deceived an unknowing machine into believing the four goals to be the same, but the specific vocabulary—the context, the goalscorer, and the way the player scored—set the goals apart.

EVATE's combination of general and specific concepts has a distinct TDT-like quality. While the baselines described the domain broadly with terms such as *quarter-final*, *aggregate* and *season*, EVATE focused specifically on What happens during events. In fact, EVATE captured more terms from the lexicons by Kubo et al. [118] and Zhang et al. [291] than any other baseline. Rank Difference, the best-performing method in recall in Table 4.1, extracted 47.50% of the single-word terms in the two lexicons, compiled specifically to summarise football matches. In contrast, TF-ICF recalled 50.00% and EVATE recalled 52.50% of terms.

EVATE also provides a lexicon of terms with the right tone and specificity for TDT on Twitter. TF-ICF's and TF-DCF's vocabulary remains too general and too noisy to be useful to a machine. Conversely, while Domain Specificity and Rank Difference construct highly-specific term lists, the Who and the Where do not transfer across events. EVATE places itself between the four baselines by balancing general, specific and topical domain terms: it understands, precisely, What happens in events in Twitter's dialect. From these perspectives, EVATE provides a more suitable vocabulary for TDT on Twitter than the baselines.

We will consolidate our findings in Chapters 5 and 6 as we demonstrate the qualities of EVATE's terms in TDT. For now, our analyses teach us two lessons about understanding What happens in event domains. First, they demonstrate the need for ATE ground truths that reflect, accurately, what it means to understand What happens in events on Twitter. Second, the poor performance of TDT's adopted ATE metrics, TF-ICF and

TF-DCF, exemplifies and explains the many failures of the research area's early ventures into understanding.

Bootstrapping to describe the How

Any supporter could tell that Germany had scored seven goals against Brazil, but a journalist could articulate the individuality of each goal. The Guardian's Barry Glendenning [81] described Germany's first goal graphically: "From a corner, a totally unmarked Thomas Muller *[sic]* side-foots home from six yards out". Glendenning did not satisfy with the What and explained the How too. So far, EVATE's understanding resembles that of a casual supporter more than that of a journalist: *goal*, *half* and *referee* rank highly, while *corner*, *deflect* and *midfield* lag far behind. In this section, we study how bootstrapping can help us construct a more descriptive, but still precise, lexicon.

We approach bootstrapping as a re-ordering problem. We focus, in particular, on EVATE; in our experiments, the noisy terms towards the top of TF-ICF's and Rank Difference's rankings led to noisy bootstrapping. In contrast, EVATE's top terms describe the domain precisely and broadly-comprehensively; the top 20 terms alone include references to goals, halves, fouls and substitutions. Therefore we use EVATE's top k terms as the seed set and re-order the remaining $200 - k$ terms.

In this section, we use three statistical metrics as bootstrappers: the chi-square statistic, Log-Likelihood Ratio (LLR) and Pointwise Mutual Information (PMI). The three bootstrappers evaluate the co-occurrence statistics of a candidate term with seed set terms in the same tweets. At each iteration, the bootstrappers add 10 terms to the seed set and continue bootstrapping with the updated set until they have re-ordered all terms. We exclude retweets from the bootstrapping calculation to consider how Twitter converses in general, without the influence of repeated content.

All three methods promote terms that co-occur significantly more than by chance, or had the candidate term and the seed set term been independent. Chi-square was used by McIntosh and Curran [156] for bootstrapping and, more notably, by Yang et al. [289] in TDT to separate words into domain and non-domain terms. The chi-square statistic between a candidate term t and a seed set term s in a corpus of tweets T can be expressed as follows using the contingency table shown in Table 4.2:

$$\chi^2 = \frac{(A+B+C+C)(AD-CB)^2}{(A+C)(B+D)(A+B)(C+D)} \tag{4.9}$$

The LLR and PMI statistics are more clear in their intents. Similarly to Roark and Charniak [219], we formulate LLR based on the log-likelihood statistic, seeking surpris-

	$\lvert\{d \in T \lvert t \in d\}\rvert$	$\lvert\{d \in T \lvert t \notin d\}\rvert$
$\lvert\{d \in T \lvert s \in d\}\rvert$	A	B
$\lvert\{d \in T \lvert s \notin d\}\rvert$	C	D

Table 4.2: In the chi-square contingency table, A represents the number of documents in corpus T that contain both the candidate term t and the seed term s, and D represents the number of documents that contain neither. B and C represent the number of documents that contain only s and only t respectively.

ing co-occurrences between candidate terms and seed set terms. We compute the logarithm of the ratio between the observed co-occurrence probability O and the expected co-occurrence probability in case of independence E [212]. Likewise, we express PMI, a common bootstrapper [103; 244], as the logarithm of the ratio between the observed co-occurrence frequency, p_{t_1,t_2} of two terms, and the expected co-occurrence frequency in case of independence, $p_{t_1}p_{t_2}$ [244].

$$LLR = 2 \cdot O \ln \frac{O}{E} \qquad (4.10) \qquad\qquad PMI = \log \frac{p_{t_1,t_2}}{p_{t_1}p_{t_2}} \qquad (4.11)$$

Apart from the bootstrappers, we vary two other parameters in our experiments. First, we adjust the size of the seed set k to understand how the number and quality of terms affect the algorithms. Second, we experiment with different ways how the bootstrappers score candidate terms: either by taking the candidate term's highest similarity to any seed set term, *MAX*, or the candidate term's average similarity with all seed set terms, *MEAN*.

Table 4.3 lists the best-performing bootstrappers. Only 3 out of 50 setups worsened EVATE's ranking quality, and all drops were marginal. The best setup with chi-square, chosen experimentally, used 16 seed terms and boosted EVATE's AP from 3.19% to 3.75%—a relative improvement of 17.49%. EVATE's P@50, already the highest of all methods, increased by 4.00% (50.00% ↑ 54.00%), and P@100 rose by 9.00% (36.00% ↑ 45.00%)—2.00% higher than TF-ICF's.

Generally, using the *MEAN* score to bootstrap new terms benefited bootstrappers more than the *MAX* score. In fact, only 3 out of 18 models performed better with the *MAX* score than the *MEAN* score. The difference between the two types of scoring was minimal, averaging just 0.06% in AP due to the sheer size of the ground truth, but the *MEAN* strategy improved consistently enough over the *MAX* strategy to be statistically-significant (one-tailed paired-samples t-test: $p = 0.001$).

Bootstrapper	Scoring	Seeds	P@50	P@100	AP
EVATE			50.00%	36.00%	3.19%
Chi-square	*MAX*	30	52.00%	44.00%	3.50%
	MEAN	16	54.00%	45.00%	3.75%
LLR	*MAX*	30	48.00%	40.00%	3.34%
	MEAN	30	48.00%	42.00%	3.38%
PMI	*MAX*	50	50.00%	42.00%	3.29%
	MEAN	30	56.00%	42.00%	3.45%

Table 4.3: Chi-square out-performed the other bootstrappers even with small seed sets. Moreover, bootstrappers consistently performed better with the *MEAN* scoring scheme.

The outcome may seem intuitive, but it clashes with what both Igo and Riloff [103], and Zhang et al. [295] found. The two had found bootstrapping based on the *MAX* similarity with any seed set term to out-perform the *MEAN* strategy. Widdows and Dorow [279] disagreed, warning that such scoring could introduce "infections", or idiomatic terms with little relevance to the broader domain. In football matches, the equivalent of infections would be bootstrapping *Manchester* because it combines with *United* to form *Manchester United*, an event term. Our findings aligned with those of Widdows and Dorow [279]. A more general word than *Manchester*, like *kick*, would be a better candidate because it combines with free-*kick*s, corner *kick*s and *kick*-offs.

Out of the three bootstrappers, chi-square generally performed best and never failed to improve EVATE's list. In large part, the chi-square bootstrapper succeeded because of its ability to choose words with precision, even when starting from small seed sets. As shown in Figure 4.6, chi-square's AP exceeded 3.65% with just 12 seed terms, and rose to 3.75% with 16 terms. PMI and LLR paled in comparison. As chi-square approached an AP of 3.75%, PMI and LLR tarried with an AP of around 3.40%. Results improved with an increasingly-large seed set but never reached chi-square's early heights.

More than just quantity, all bootstrappers require quality. Beyond 25 terms, the initial seed set grew, but the noise grew with it. When the seed set grew larger than 25 terms, PMI's and LLR's performance reached a plateau, and chi-square's AP worsened. With each new noisy term that entered the seed set, one fewer noisy term remained for the bootstrappers to re-order. The bootstrappers lost their purpose and effectiveness. By the end, with a seed set of 60 terms, the bootstrapped lexicons converged, and increasingly resembled each other and EVATE's original lexicon.

A closer look at the best bootstrapper reveals its immediate influence. The chi-square

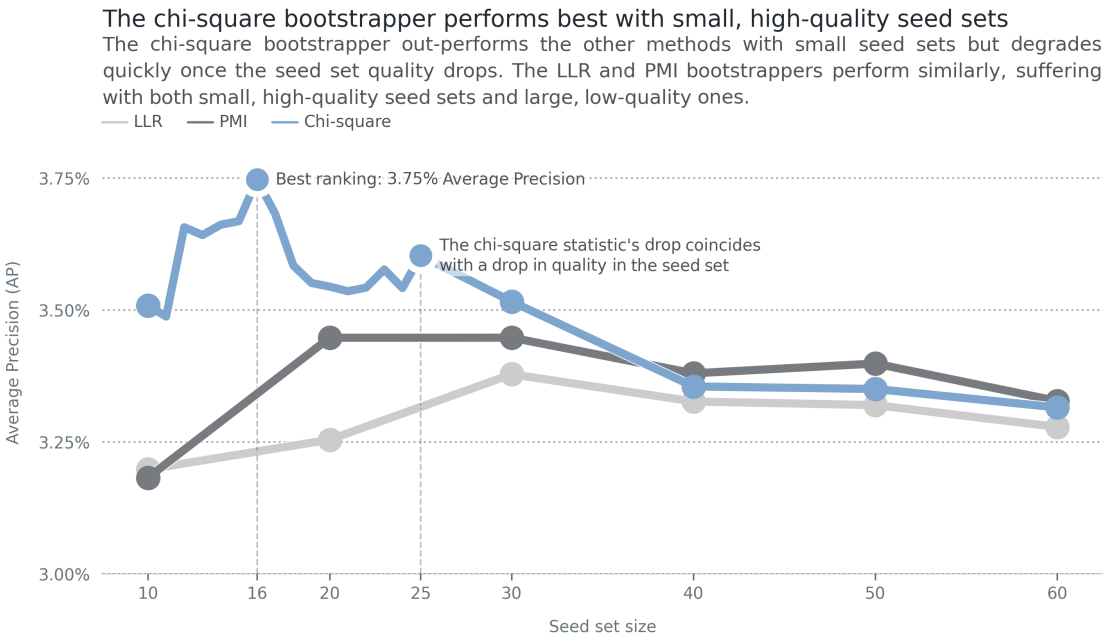

Figure 4.6: The chi-square bootstrapper performs best with small, high-quality seed sets. Conversely, the PMI and LLR bootstrappers require large seed sets, at which point quality dwindles, hindering the two bootstrappers.

bootstrapper with *MEAN* scoring and 16 seed terms gave a prompt and considerable boost to precision. Early on, the bootstrapper boosted words like *corner* ($174^{th} \uparrow 41^{st}$), *net* ($193^{rd} \uparrow 33^{rd}$) and *decision* ($178^{th} \uparrow 22^{nd}$). P@k peaked at 65.38% after the 25^{th} term, but the bootstrapper maintained its improvement over EVATE until the end. Precision remained above 50.00% until the 88^{th} term, at which point EVATE had scored just 40.00%.

At the other end of the ranking, chi-square made few mistakes too. When the bootstrapper's precision started to decrease, it decreased almost monotonically, as if only noisy terms remained to bootstrap. The term *freekick*, a popular misspelling of *free-kick*, dropped heavily ($17^{th} \downarrow 172^{nd}$). However, many other terms dropped along with *freekick*, with the worst-hit words being subjective expressions like *WTF* ($26^{th} \downarrow 166^{th}$), *masterclass* ($27^{th} \downarrow 173^{rd}$) and *wow* ($33^{rd} \downarrow 175^{th}$).

We apply the chi-square bootstrapper's terms again later in this chapter and in Chapter 5, where we apply the top terms in a TDT algorithm. Before, however, we move from football's ideal setting to more unwelcoming domains. We start by analysing EVATE and the baselines in the slow-changing domain of Formula 1 Grands Prix.

4.4 | EVATE as a semantic re-ranker: the language of Formula 1

Between 2007 and 2022, Lewis Hamilton raced in more than 300 Formula 1 Grands Prix. Over 16 years, he broke countless records and became synonymous with Formula 1: Lewis Hamilton was Formula 1, and Formula 1 was Lewis Hamilton. In the previous section, EVATE constructed a lexicon that captured What happens in football matches, but what allowed EVATE to construct such an accurate lexicon was the rigidity, structure and rich variety of the domain. Such variety appears nearly impossible in Formula 1. Had we collected one Grand Prix dataset per season since Twitter launched in 2006, only one corpus would not have featured Lewis Hamilton: the 2006 season.

Event domains like Formula 1 blur the line between domain and event terms. The ubiquity of drivers and constructors, like Hamilton's Mercedes, creates a seeming paradox in which the Who ceases to represent event terms and starts to represent domain terms. In this section, we study the role of named entities in Formula 1 Grands Prix. We do not attempt to answer the question of whether named entities should be domain terms—not even ATE literature seems capable of deciding. Instead, we entertain both possibilities: named entities as domain terms and named entities as event terms.

We base our experiments on the 2020 Formula 1 season. The 2020 season, curtailed by COVID-19 restrictions, included 17 Grands Prix, and we collected data about 15 of them. Similarly to the experiments on football matches, each dataset includes an understanding period, which lasts half an hour, and a longer event period. Differently from football matches, however, the participants—the drivers and the constructors—seldom change. Therefore we collected tweets that mention participants for both periods. Appendix D.5 includes more details.

The baselines and metrics in this section remain unchanged. We re-use TF-ICF and TF-DCF as the general ATE algorithms, and Domain Specificity and Rank Difference as the specific methods. Once again, we focus on TF-ICF and Rank Difference as the best-performing methods in each group. Each baseline produces a lexicon using the best configuration in Section 4.3: by considering all tweets by verified authors.

We compare the outputs from EVATE and the baselines with a set of four ground truth lists. The crowd-sourced glossary on Wikipedia [280] serves as a general motor sport ground truth, and three others, by F1technical.net [60], Formula 1 Dictionary [68] and Formula 1 [67], focus specifically on Formula 1. We also use widely-available lists of drivers, constructors and Grands Prix from the 2020 season. Differently from other ground truths, the Formula 1 glossaries consist of highly-technical, mostly multi-word

terms. Therefore for this experiment only, we split multi-word terms into their single-word constituents.

Using the described setup, we dedicate the rest of this section to study how ATE algorithms handle named entities. We start by considering named entities as domain terms, not event terms, in the first analysis.

Named entities as domain terms

To someone who follows Formula 1, watching the same drivers lining up on the grid and racing around the circuit is an almost-weekly routine during the season. They grew up watching the same constructors; several have been competing for decades. The drivers and constructors still act as the Who, and the actions and changes as the What, but you could see why some event terms could double as domain terms. Therefore in this analysis, we consider drivers and constructors as domain terms.

To consider event terms as domain terms, we loosened the linguistic filters. Now, the four baselines accepted proper nouns as well as nouns, verbs and adjectives. With fewer linguistic constraints, the four baselines filled their lexicons with named entities, as shown in Table 4.4. Because of the consistency of named entities in Formula 1, even TF-ICF and TF-DCF, the general methods, include many named entities in their lexicons; the prevalence of named entities accounts for the 20.00% difference in TF-ICF's precision between Tables 4.4a and 4.4b, a comparable figure to Rank Difference.

The abundance of named entities makes more sense in the lexicons of the specific methods. By design, Domain Specificity and Rank Difference focus on technical terms, but the two methods subverted our expectations. Instead of harnessing POS tag filtering, Domain Specificity and Rank Difference succumbed to noise. In pursuit of highly-technical terms, Rank Difference arbitrarily prioritised driver nicknames, Grand Prix locations and racing positions, like *P14*. Few truly technical terms, like *constructor*, *halo* or *DRS* (Drag Reduction System), appeared in Rank Difference's lexicon.

EVATE's behaviour remained consistent. With a lexicon that balanced the general terms with the more specific terms, EVATE again found a place between the two groups of methods. Our method captured regular domain terms like *gravel*, *wing* and *tire*, but not at the expense of neglecting drivers and constructors. Even without relying on POS tagging, EVATE identified drivers and constructors as domain terms due to their specificity and persistent involvement in topics.

Thus, EVATE out-performed all other methods. Two-thirds of the first 50 terms described either What happens or Who is involved in Grands Prix. Even TF-ICF, which achieved 8.00% higher overall precision than EVATE (43.50% ↑ 51.50%), ranked fewer

Algorithm	P@50	P@100	Precision	Recall	F-score
EVATE	24.00%	28.00%	25.50%	5.26%	8.72%
TF-ICF	28.00%	39.00%	31.50%	6.49%	10.77%
TF-DCF	22.00%	30.00%	29.00%	5.98%	9.91%
Domain Specificity	14.00%	13.00%	14.00%	2.89%	4.79%
Rank Difference	14.00%	17.00%	23.00%	4.74%	7.86%

(a) The ATE baselines performed poorly when we excluded drivers and constructors from the ground truth. All techniques, including the baselines, which benefited from POS tagging, still mistakenly captured many named entities, including the baselines, which benefited from POS tagging.

Algorithm	P@50	P@100	Precision	Recall	F-score
EVATE	66.00%	57.00%	43.50%	8.44%	14.13%
TF-ICF	62.00%	64.00%	51.50%	9.99%	16.73%
TF-DCF	62.00%	64.00%	49.00%	9.51%	15.92%
Domain Specificity	44.00%	33.00%	27.00%	5.24%	8.77%
Rank Difference	60.00%	55.00%	43.50%	8.44%	14.13%

(b) Performance more than doubled when we included drivers and constructors in the ground truth. The increase in performance over Table 4.4a is due to the prominence of names in the five algorithms' lexicons.

Table 4.4: ATE methods captured many named entities in the domain of Formula 1 Grands Prix. EVATE's lexicon stands out for its higher quality due to its ability to filter Grand Prix locations.

precise terms among the first 50 (66.00% ↓ 62.00%). The difference is in EVATE's ability to recognise the drivers and constructors from the actual event terms: the Grand Prix locations, or the Where.

Drivers and constructors are not the only important named entities in Formula 1. Albeit only relevant to one or, rarely, two Grands Prix per season, the circuits' names, symbolic of the Where, appear with intensity. As a result, opening up the baselines to named entities without any other semantic controls degraded the baselines' rankings. EVATE had no trouble filtering the names of Grands Prix, both because ELD rarely captured them and because most hosted only one event. The baselines made no such distinction: TF-ICF identified 18 locations, and Rank Difference extracted 20. EVATE captured none.

The distinction between how the baselines and EVATE handled event terms typi-

fies the distinction between tools built for general domains and those built for event domains. Event domains differ from general domains, and EVATE understood the difference. Even without any additional controls on the linguistic component, it was able to distinguish between the Who and the Where, the domain terms and the event terms. The lack of controls, however, make the next question a tougher ask of EVATE. What if drivers and constructors were event terms, not domain terms?

Named entities as event terms

Named entities as domain terms make sense, but you could also understand why research might frown upon the idea. From a purist perspective, named entities, the Who and the Where, remain the responsibility of APD—ATE should describe uniquely What happens. Moreover, a lexicon without named entities generalises better. Drivers retire and constructors fail, but the fundamental concepts that compose the What remain—the impermanence of named entities cannot be but a negative trait. A general understanding, free of named entities, could transfer across motor sport domains, like Formula 2, Formula E and possibly MotoGP. Therefore in this section, we consider drivers and constructors as event terms, and remove them from the ground truth.

For the baselines, excluding named entities proved as simple as tweaking the POS tagging algorithm to reject proper nouns. Of course, POS tagging fails on occasion, and because POS tagging fails, the baselines experienced the drops in performance from Table 4.4 to Table 4.5a. For instance, POS tagging correctly interpreted the names of Grands Prix, such as the *Austrian* Grand Prix, as adjectives. While the baselines avoided drivers and constructors for the most part, the Grand Prix locations again degraded the overall precision and ranking quality.

EVATE suffered more heavily. For EVATE, rejecting named entities requires tweaking the TDT algorithm, which is inconvenient and adds unnecessary overhead if the code is at all available. We did not change our method, and neither did the lexicon change. Drivers and constructors still occupied 9 out of EVATE's top 10 words, and its performance remained the same as in Table 4.4a, stooping to the lowest P@50 in Table 4.5a.

In the end, neither lexicon adapted sufficiently to consider named entities as event terms. The baselines' linguistic component excluded the Who but not the Where, and while EVATE's statistical component excluded the Where, it was unable to exclude the Who. Therefore in the rest of this analysis, like Zhang et al. [294] before us, we combine the baselines' qualities with those of EVATE to re-rank the baselines' lexicons.

First, we used EVATE as a simple semantic re-ranker. EVATE calculated its own ter-

Algorithm	P@50	P@100	Precision	AP
EVATE	24.00%	28.00%	25.50%	1.36%
TF-ICF	58.00%	47.00%	37.50%	4.07%
TF-DCF	48.00%	47.00%	37.50%	3.78%
Domain Specificity	30.00%	23.00%	18.50%	0.97%
Rank Difference	38.00%	45.00%	39.00%	3.32%

(a) All ATE methods struggled to distinguish between between domain terms and event terms, often promoting named entities. The difficulties persisted even after configuring POS tagging to retain only nouns, verbs and adjectives.

Algorithm	P@50	P@100	Precision	AP
TF-ICF	54.00%	47.00%	37.50%	3.71%
TF-DCF	54.00%	48.00%	37.50%	3.61%
Domain Specificity	42.00%	27.00%	19.00%	1.58%
Rank Difference	56.00%	50.00%	39.00%	3.83%

(b) As a simple re-ranker, EVATE re-ordered the baselines' lexicons independently. EVATE's contributions improved Domain Specificity and Rank Difference, the worst ATE methods, but had minimal effect on the best ones.

Algorithm	P@50	P@100	Precision	AP
TF-ICF	66.00%	48.00%	37.50%	4.36%
TF-DCF	56.00%	48.00%	37.50%	4.13%
Domain Specificity	42.00%	27.00%	18.50%	1.51%
Rank Difference	60.00%	49.00%	39.00%	4.18%

(c) As a combined re-ranker, EVATE multiplied its termhood scores with those of the baselines. This time, every method improved, including TF-ICF and TF-DCF, the best-performing ATE baselines.

Table 4.5: EVATE improved the performance of ATE methods as a semantic re-ranker. The most consistent improvements, however, resulted from combining EVATE's termhood scores with those of the baselines.

mhood scores for the terms in each baseline's lexicon and then used the new scores to re-rank the terms. As shown in Table 4.5b, simple re-ranking achieved mixed results. Domain Specificity's and Rank Difference's AP rose sharply, but the improvements coincided with declines in TF-ICF's and TF-DCF's qualities. The two factors combined allowed Rank Difference to overtake TF-ICF in P@50 (54.00% ↑ 56.00%) and P@100 (47.00% ↑ 50.00%). Simple re-ranking thus brought the four baselines' results closer

to each other, but at the expense of the general methods.

In simple re-ranking, the baselines relinquished too much control to EVATE. The four baselines became the linguistic filters for EVATE, the actual termhood measure. Domain Specificity and Rank Difference could hardly perform worse, and benefited from having our method relegate Grand Prix locations from the top ranks. Conversely, EVATE, which prioritised drivers and constructors, punished any mistake in TF-ICF's and TF-DCF's lexicons. In the former lexicon, *[Lance] Stroll* climbed from 93rd to 8th and *[Max] Verstappen* from 98th to 4th.

Second, we allowed the baselines and EVATE to share control by combining termhood measures. In combined re-ranking, the baseline still primarily acts as a linguistic filter and EVATE still primarily acts as a termhood measure. This time, however, EVATE shares its termhood role with the baseline. Combined re-ranking re-scales the baseline's and EVATE's termhood scores between 0 and 1, and then multiplies them to compute the terms' final termhood scores. In other words, combined re-ranking considers both the baseline's and EVATE's termhood scores, and prioritises terms that the two metrics rank highly.

Combined re-ranking brought the four baselines closer to each other in AP, but this time at no baseline's expense. At the higher end of the lexicons, EVATE and the baselines compromised to prioritise common and high-scoring terms. At the other end, EVATE exercised its degree of control to relegate Grand Prix locations. Domain Specificity and Rank Difference again benefited the most, with P@50 rising by 12.00% (30.00% ↑ 42.00%) and 22.00% (38.00% ↑ 60.00%), as shown in Table 4.5c. The general methods profited less but profited nonetheless. TF-DCF and TF-ICF experienced an 8.00% improvement in P@50, and the latter retained a 66.00% precision after 50 terms.

EVATE's contributions as a re-ranker despite its own struggles highlight the importance of semantics in ATE. We cannot conflate EVATE failing by missing valid domain terms, which was not our method's primary flaw, with it failing because it does not meet our expectations. EVATE's shortcomings in this section reflected not flaws in logic but the datasets' inadequate variety. However, even when it fails to meet our expectations, EVATE can act as a semantic re-ranker. In the last section, we apply its semantic properties in a domain that, unlike Formula 1, changes constantly: politics.

4.5 | ATE in dynamic domains: the language of American politics

If a week is a long time in politics, then three months must feel like an eternity. The three months surrounding the 2020 United States presidential election certainly felt like they would never end. Americans went from a controversial president to a controversial election, from protests on social media to protests at the Capitol, and from the COVID-19 battle to the vaccination battle. They were challenging circumstances for the United States and they are challenging circumstances for ATE.

Politics create a stark contrast with football matches and Formula 1 Grands Prix. Under normal conditions, football matches start with a referee whistle and end with another whistle an hour and a half later, and Grands Prix start on the grid and end at the finish line: fixed rules and clear boundaries. Politics have neither. On one day during the 2020 United States election cycle, America voted, and on the other, America rioted. Election Day stretched into Election Week. To ATE, politics symbolise the polar opposite of our previous domains, the ideal settings for term extraction.

Still, the volatility of politics serves a purpose. ATE research postulates as a solution to evolving domains, where WordNet and other resources do not suffice [14; 140; 208]. In dynamic domains, the community takes its algorithms' suitability as a foregone conclusion, rarely pausing to question the claim, but we do. In this section, we use politics' state of constant change to test whether and how ATE algorithms, including EVATE, adapt to dynamic domains.

The setting of our problem consists of the three months surrounding the 2020 United States presidential election, contested by Donald Trump for the Republican Party and Joe Biden for the Democratic Party. We started collecting data on 20 October 2020, in the final days of the election campaign. Americans voted two weeks later, on 3 November. We continued collecting data until 21 January 2021, the day following President Joe Biden's Inauguration Day. During these 94 days, and as shown in Figure 4.7, we captured mounting legal challenges, riots at the Capitol, and background themes, like the COVID-19 pandemic.

The setting itself poses two challenges to adapt the datasets to ATE algorithms. First, the lack of boundaries forced us into a simplifying assumption: each day represents an individual event structured identically as the others. Second, because political events have no clear start, we could not collect an understanding period, which ELD uses to construct timelines. Instead, ELD always uses the general dataset described in Appendix D as the understanding period. Thus, ELD's term-weighting scheme promotes

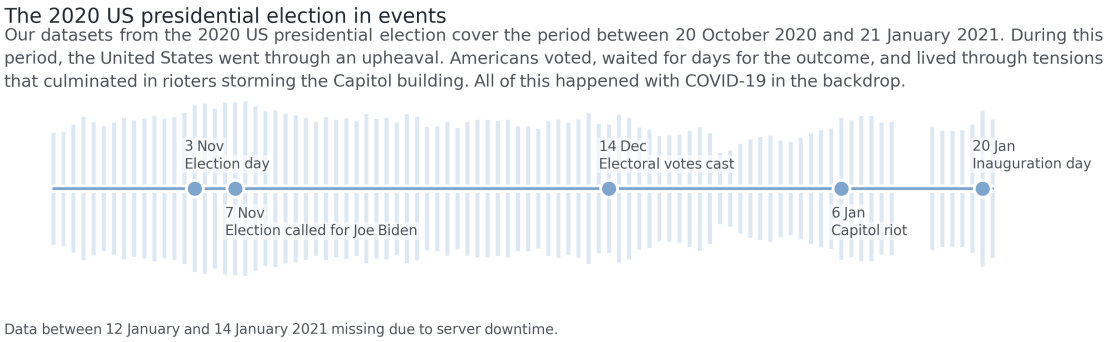

Figure 4.7: A timeline of notable events from the 2020 US presidential election. The data spans three months and notably covers election day, the Capitol riot and inauguration day, among other events.

words that appear more often in politics than in general.

The nature of the data itself barely changes. We missed three days, between 12 January and 14 January 2021, due to server downtime, resulting in a 91-day dataset. On the other days, we excluded retweets due to the large volume and still collected more than 85 million tweets, an average of 939,763 tweets per day. Appendix D.7 describes our data collection process in more detail.

Likewise, the evaluation procedure barely changes. We re-use the same baselines and configurations: tweets by verified users except retweets; Rank Difference again only considers terms that appear at least 50 times. Our ground truth covers unigrams from three subjects. First, we include general political terminology from a glossary by the Dole Institute of Politics [52] and a dictionary by Brown et al. [27]. Second, we use a list of election terms from the U.S. Election Assistance Commission [263]. Third, we consider legal terminology from Justia [107] and the United States Courts [261].

In the rest of this section, we analyse ATE's behaviour over the three months covered by our datasets. We start by studying just how well the four baselines and EVATE adapt and learn from the volatility of American politics. Then, we combine the techniques from this chapter to create a generalisable and transferable lexicon of political language.

The myth of adapting

After three months, all ATE algorithms had surveyed, directly or indirectly, more than 85 million tweets. Those 85 million tweets carried breaking news, jeers and opinions from a polarising election like few others. In the ways of noise, the data from the US election resembled that from football matches and Formula 1 Grands Prix. In almost every other way, it did not. EVATE and the four baselines reacted poorly, and as Ta-

Algorithm	P@50	P@100	Precision	Recall	F-score
EVATE	24.00%	28.00%	26.00%	4.44%	7.59%
TF-ICF	46.00%	38.00%	32.00%	5.47%	9.34%
TF-DCF	44.00%	38.00%	30.50%	5.21%	8.90%
Domain Specificity	30.00%	27.00%	28.00%	4.78%	8.17%
Rank Difference	44.00%	38.00%	35.50%	6.06%	10.36%

Table 4.6: ATE algorithms struggled immensely in the domain of politics. The four baselines and EVATE captured very few terms, even from our massive ground truth lexicons. Not one algorithm and not one assumption adapted to the volatility of politics.

ble 4.6 shows, the five algorithms extracted relatively few terms, even from a domain as limitless as politics.

TF-ICF and TF-DCF hardly adapted at all. TF-ICF provided an overview that must be familiar by now: a general picture of the domain composed of terms like *vote*, *president* and *election*. Yet part of what gave TF-ICF a general outlook seems to have been consistency: *vote*, *president* and *election* were not only relevant but consistently relevant. In a domain as dynamic as politics, however, where most of what constitutes relevance changes from one day to the next, few events appeared consistently. Most appeared as an ephemeral presence: one day of voting and an exceptional few days of vote counting, a day of rioting and its aftermath, and one day to inaugurate the new president.

Without consistency, what TF-ICF and TF-DCF got wrong devalued what they got right. The two still sought frequent and specific terms, but because the specific terms appeared so briefly, TF-ICF and TF-DCF populated their lexicons with frequent words instead. What remains constant and consistent, frequent but not specific in the domain of politics is the language of profanity and opinions, everyday language. The assumption of frequency lost all meaning.

A reliance on frequency produced trivial lexicons. TF-ICF and TF-DCF scattered political terms among the noise, a saturation of trivial words such as *say*, *think* and *year*. Figure 4.8 shows how almost three out of every four terms in TF-ICF's lexicon appeared in the list of the thousand most common English words according to English First [59]. In contrast, less than a quarter of Rank Difference's and EVATE's top 200 terms figured in the list. Of course, not all common words are incorrect—*vote*, *president* and *election* are all common—but most bear little relevance to the domain and thus trivialise the lexicon.

Rank Difference adapted remarkably in comparison. The specific method thrived amid the highly-technical, unambiguous terminology of politics. Terms like *impeach-*

ment, *certification* and *insurrection* soared, and not even named entities proved problematic. The main actors of the 2020 United States election, namely Joe Biden and Donald Trump, already enjoyed global repute (and disrepute) before the election, which lowered specificity. Rank Difference again performed best with 35.50% precision.

Yet Rank Difference's high precision hides a muted defect. The simplistic assumption of contrastive approaches, that a term appears in only one domain, fails too with politics' most popular terms. The terms *vote*, *president* and *election* all appear in the list of most common words [59] because politics seep into everyday life. In other words, they fail the assumption of specificity. Domain Specificity and Rank Difference punished general terms severely, and few broached the top 200 terms. The term *vote* only appeared 552$^{\text{nd}}$ in Domain Specificity's ranking, incredibly still much better than its position in Rank Difference's lexicon: 7,403$^{\text{rd}}$.

EVATE again established itself as the middle-ground between the triviality of the general methods and the specificity of the technical metrics. It captured both *vote* and *impeachment*, both *president* and *certification*. Yet while EVATE's lexicon balanced general terms with more specialised ones, its assumption failed too. Not every day represented an individual event with an identical structure to all others. Only the six days of tireless vote counting, between voting day and when cable networks called the election for Joe Biden, remained faithful to EVATE's assumption. The rest of the days only shared the overarching context of a single election from one political landscape.

The conundrum of named entities returned. Without consistency, named entities became the common themes, and EVATE learned the Who and the Where much more comfortably than the What. Joe Biden and Donald Trump, their running mates and family members inundated the top ranks. And while TF-ICF and Rank Difference captured no American states, they constituted 10 (5.00%) of EVATE's 200 terms.

Most of EVATE's remaining mistakes resembled Rank Difference's. On manual inspection, many errors proved false flags. Several terms were absent from the ground truth: *socialist*, *insurrection* and *overturn*. Many others described the political agenda without being specific to politics—*attorney*, *economy* and *vaccine*—or gained temporary relevance from the context: the controversy of Hunter Biden's *laptop* and Trump's campaign slogan, to drain the *swamp*.

In fact, the mistakes may even appear as welcome signs. Many capture an accurate, if brief, snapshot of American politics, set in 2020. If ATE algorithms truly adapted to domains, as research postures, then we should only expect our lexicons to capture a narrow, general facet of the ground truth alongside newly-relevant terms. The mistakes seem to augur well for adaptive algorithms.

In reality, however, ATE literature's aspirations to adaptive learning seem shallow.

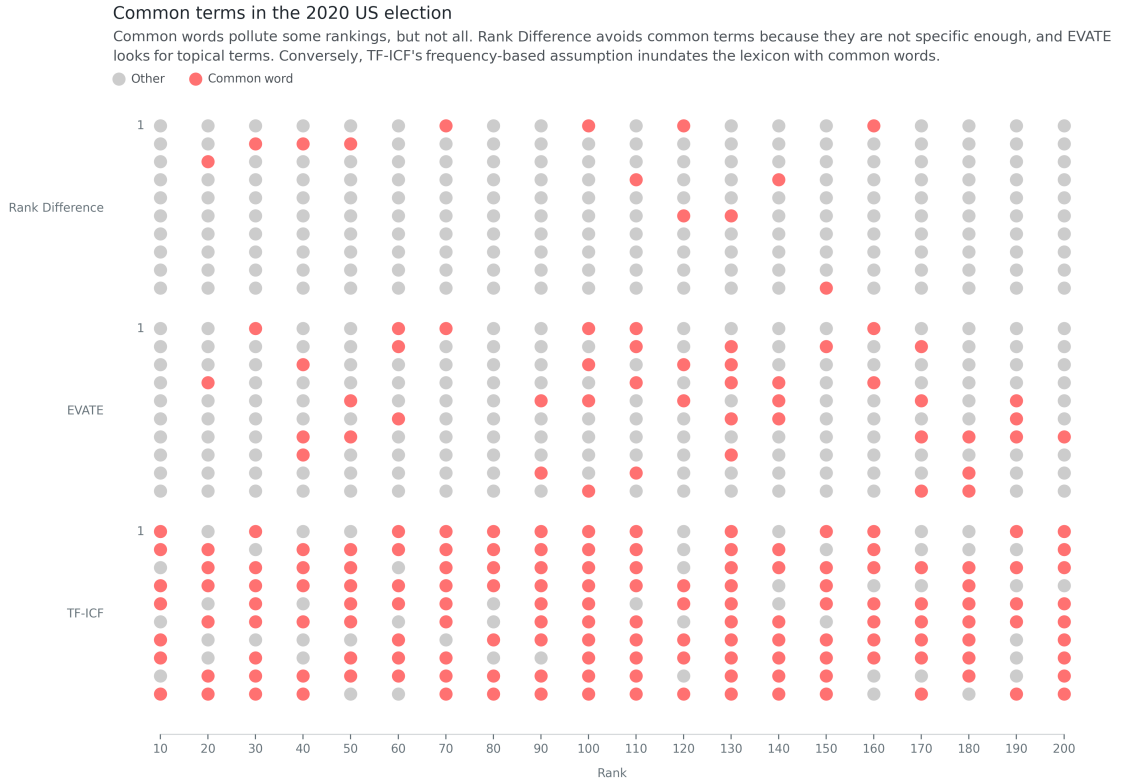

Figure 4.8: TF-ICF fills the political lexicon with common words. Almost three-quarters of TF-ICF's terms appear in the list of the thousand most common English words [59], a far higher rate than Rank Difference and EVATE.

The state of American politics on day 1 differed greatly from its state on day 91; calls to vote quieted, campaign slogans expired, and scandals faded to be replaced by new ones. Adaptive learning should renew understanding as it consumes new data. It should constantly cycle between forgetting and learning. Algorithms must forget quickly—and selectively, only the irrelevant terms—and then learn just as quickly to replace old concepts with newly-topical ones.

Despite pretences, forgetting and learning do not characterise ATE algorithms. In our experiments, TF-ICF hardly learned. Newly-relevant terms strained to catch up with more established terms; even *riot* only finished 247$^{\text{th}}$. Rank Difference learned more easily. The differences in ranks seemed easier to overcome than TF-ICF's raw term frequencies but only as long as terms burst powerfully. Conversely, the baselines forgot too slowly. Both relied, directly or indirectly, on term frequency, and the more tweets the techniques surveyed, the harder it became to forget old concepts and learn new ones.

Forgetting may be EVATE's strongest quality in a domain with scarce qualities. At al-

most every major shift in discourse, EVATE's lexicon worsened slightly as it introduced new terms at the end of the lexicon before finding them a rightful place or discarding them altogether. Precision dipped after news networks called the election for Joe Biden, after rioters stormed the Capitol building, and again after inauguration day. Between the United States Electoral College's vote and the end, TF-ICF replaced 21 terms and Rank Difference changed 31 terms. During the same period, EVATE recycled more than a quarter of its lexicon: 52 terms.

Yet EVATE learned cautiously and slowly, and sometimes not at all. Because our assumption failed—not every day experienced the same events—EVATE had to wait for days until it observed another similar event. The term *riot* first appeared in its lexicon on 10 January 2021—four days after the actual Capitol riots and three days after Rank Difference first added the term to its lexicon. Sometimes, another similar event did not occur; ELD only captured allegations of an attempted *insurrection* once, not enough for EVATE to assign a positive termhood score. In fact, of 1,944 topical terms, ELD only captured 783 (40.28%) at least twice.

Our findings leave a void. Not only do the five algorithms hardly adapt, but they also capture a narrow view of politics stuck in 2020. In TF-ICF and TF-DCF, the generalisable mixes with the banal; distracting knowledge reminiscent of the understanding from TDT's early research—lexicons too trivial to be useful, as we show in Chapter 5. In Domain Specificity and Rank Difference, the specific terms evoke esoteric knowledge: convenient when needed and harmless otherwise, but ultimately rarely useful. Finally, in EVATE, the Who and the Where detract from the generalisability of the What.

We conclude this chapter with one last analysis. Our lexicons so far apply to one brief and parochial window, American politics in late 2020. Even ignoring the mistakes, applying the lexicons in any other time frame or in any other location would erode relevance further. In the last analysis, we remedy the situation by combining techniques from previous experiments to craft a generalisable and transferable political lexicon.

The baseline of politics

The scene is one of abject failure: a failure to learn, a failure to generalise and a failure to adapt. Yet it is not an entirely new scene. We have seen ATE algorithms failing to learn a precise, topical lexicon in football matches, and we have seen ATE algorithms failing to generalise in Formula 1 Grands Prix. We cannot make a lexicon respond more quickly to the dynamics of events—that would require changing the algorithm itself—but we can improve the learned output to make it more topical and more generalisable. This is the challenge that we undertake in this analysis: to combine re-ranking with bootstrapping

Method	P@50	P@100	P@200	P@400	AP
TF-ICF	46.00%	38.00%	32.00%	25.50%	3.39%
TF-ICF with re-ranking	62.00%	45.00%	38.00%	27.25%	4.60%
TF-ICF with re-ranking, bootstrapping	68.00%	57.00%	42.00%	27.25%	5.59%

Table 4.7: The combination of re-ranking with bootstrapping created a generalisable and transferable political lexicon. The final lexicon inherits the best qualities of TF-ICF and EVATE. We considered the top 400 terms in each lexicon to calculate the AP.

and salvage from the lexicons the transferable political knowledge.

Consider where we start. Rank Difference appears immediately contrary to the principles of generalisability and transferability. It paints a relatively precise picture of the domain of politics, but it remains too specific, too American—too incompatible with our goals. Moreover, EVATE shares very little overlap with Rank Difference for re-ranking to have any meaningful influence; of Rank Difference's top 1000 terms, EVATE agrees on just 175 (17.50%). TF-ICF has a very different set of flaws. We described TF-ICF's lexicon as too trivial and too noisy, with words like *[they]'re*, *make* and *today*. Nevertheless, scattered among the noise lie the elements of a generalisable and transferable lexicon.

Therefore to begin with, we applied EVATE as a semantic re-ranker to sort TF-ICF's lexicon. We re-used the best configuration from Section 4.4, combining the termhood scores of TF-ICF and EVATE to re-rank the former's top 1,000 terms.[4] Evidently, the list includes many more words than we would need, but only because it also contains much more noise than we would want. We charge EVATE with diligent refinement, and as Table 4.7 shows, it refined TF-ICF's coarse lexicon masterfully.

TF-ICF and EVATE agreed on just 382 terms (38.20%), but 382 terms sufficed. EVATE transformed TF-ICF's lexicon. Trivial noise like *say* ($4^{th} \downarrow 384^{th}$), *people* ($14^{th} \downarrow 386^{th}$) and *want* ($49^{th} \downarrow 392^{nd}$) dropped hundreds of positions, replaced with more topical terms such as *riot* ($247^{th} \uparrow 96^{th}$), *county* ($323^{rd} \uparrow 92^{nd}$) and *speech* ($199^{th} \uparrow 52^{nd}$). Performance reflected the quality of the movements too. As topicality overcame noise, P@100 rose by 7.00% (38.00% ↑ 45.00%) and P@200 by 6.00% (32.00% ↑ 38.00%). TF-ICF's lexicon became visibly cleaner and more topical.

Bootstrapping improved the lexicon even further. As EVATE moved terms to the dense upper echelons of the lexicon, the first ranks started to describe politics minutely,

[4]We experimented with several other configurations too. We tweaked the number of terms, re-ranked Rank Difference's lexicon, and even attempted to merge the re-ranked lists of Rank Difference and TF-ICF. The results were underwhelming. More than just ineffectual, the low overlap between EVATE and Rank Difference rendered re-ranking detrimental. For brevity, we do not present the results here.

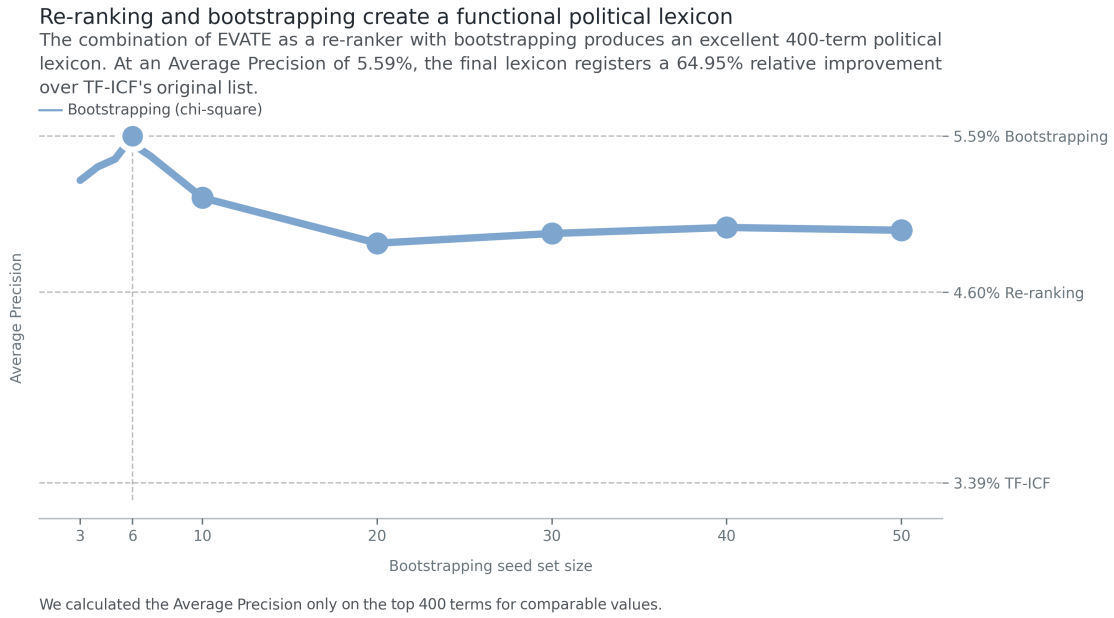

Figure 4.9: Semantic re-ranking and bootstrapping improve TF-ICF's political lexicon. With just six seeds, the chi-square bootstrapper moves valid terms to the forefront of the new lexicon and relegates noise to the end.

which enabled bootstrapping. Now, we could use the first political terms to seek other similarly-political terms. We re-used the same configuration from Section 4.3, the chi-square bootstrapper with the $MEAN$ scoring strategy. This time, chi-square ordered the first 400 terms, ten at a time, covering all terms to which both EVATE and TF-ICF assigned a positive termhood value. We only varied the seed set size, and as Figure 4.9 shows, bootstrapping never failed to improve the lexicon.

Once again, chi-square needed very few examples. The best bootstrapper registered a relative increase of 64.95% in AP (3.39% ↑ 5.59%) over the base lexicon with just six seeds: *election*, *vote*, *president*, *pardon*, *administration* and *campaign*. General, subjective and other nondescript terms in the re-ranked lexicon, like *today* (76^{th} ↓ 293^{rd}), *good* (73^{rd} ↓ 302^{nd}) and *Biden* (57^{th} ↓ 380^{th}), made way for the ephemeral concepts that EVATE scarcely encountered. Terms like *project* [result] (268^{th} ↑ 66^{th}), *mail*[-in ballots] (359^{th} ↑ 23^{rd}) and *cast* [ballot] (216^{th} ↑ 13^{th}), though only relevant briefly, climbed the ranks.

You could not call the final lexicon EVATE's or TF-ICF's, but it bears the best qualities of both. Among the top 200 terms, the share of common English words, so abundant in TF-ICF's lexicon, declined from 74.50% in the original lexicon to 62.50% in the re-ranked one and 51.00% in the bootstrapped one. Simultaneously, P@200 rose steadily, from 26.00% in EVATE's lexicon and 32.00% in TF-ICF's to 38.00% in the re-ranked one

and 42.00% in the bootstrapped one. For the first time, P@100 exceeded 50.00% and remained above the symbolic threshold until the 138th term. We empirically truncated the final lexicon at 250 terms, at which point precision still stood at 36.00%.

Of course, we make sacrifices in the combination. We sacrifice the informality of the average tweet for the authoritative tone of verified users, and Rank Difference's specific terms for TF-ICF's general ones. More importantly, our data left an indelible trace of American politics in the final lexicon. We captured *senate*, *governor* and [the United States Electoral] *College* but missed *parliament*, *minister* and [the House of] *Commons*, the foundation of other systems of governance. Evidently, TF-ICF and EVATE could not compensate for the gaps in data.

Yet even the defects cannot tarnish the overall quality of the final lexicon, which captures everything lastly political and discards anything subjective and ephemeral. It demonstrates that even in a domain as vast as politics, a baseline of terms remains. The baseline describes current affairs, issues of policy—*climate change*, the *military* and the *economy*—and the basic functioning of democracy—*voting*, *debating* and *governance*. All these concepts and others, our lexicon captures thanks to EVATE's semantic understanding. We apply this lexicon in Chapter 6. For now, we conclude our experiments in understanding, and with them, this chapter.

Recap

A goal is a goal. A human understands that a goal is an important concept in football matches but scrambles to explain why. Nevertheless, finding an answer to the question remains important, and we found ours in semantics. In the end, our endeavours let us harness the link between ATE and TDT to propose a semantic termhood measure tailored for event tracking on Twitter, EVATE. In doing so, we answered the following questions:

- What makes a word a domain term? Semantics make a word a domain term, but semantics remain an elusive concept. In Section 4.1, we found in ATE literature the attributes that get us close to—but do not quite capture—the meaning behind words: the POS tags, and the frequency, specificity and consistency of words.

- How can ATE methods extract terms that make sense semantically? The answer depends on the application, but in event domains, the answer is somewhat clearer: topical terms command a powerful semantic presence. In Section 4.2, we designed

Principal contributions

- The first study on the performance of traditional ATE algorithms on Twitter and in different types of event domains
- EVATE, the first ATE method designed for tweets and for TDT
- The first study into how bootstrapping and semantic re-ranking can adapt the outputs of traditional ATE algorithms to different types of event domains

EVATE by replacing traditional linguistic filtering with a TDT algorithm to identify candidate terms.

- How well can ATE techniques extract domain terms from Twitter? Twitter is a difficult medium that devastates performance, but progress remains possible, as our analyses showed in the domain of football matches. In Section 4.3, we demonstrated how ATE research needs to understand user-generated content better, and how EVATE manages to build adequate lexicons for TDT research on Twitter.
- What roles do named entities play in slow-changing domains? EVATE understands event domains and, by extension, the role of named entities better than traditional ATE algorithms. In Section 4.4, we contemplated the roles of named entities in Formula 1 and showed how even when EVATE fails, it can contribute to other algorithms as a semantic re-ranker.
- Do ATE algorithms truly adapt to dynamic domains? Research presents itself as a solution to dynamic domains, but traditional algorithms forget too slowly and learn too slowly to adapt. In Section 4.5, the domain of politics discredited ATE literature's claims, but we also demonstrated how EVATE can help create a generalisable and transferable political lexicon.

In the first half of this dissertation we explored the development of event understanding. In the second half, we apply our understanding in two applications. Our first use-case applies EVATE's knowledge about football matches in a TDT algorithm. In the next chapter, we reflect on what makes the ideal algorithm and design a technique driven by our understanding from this chapter to approach the hypothetical ideal.

5

Application

The Football Case Study

In TDT's early years, what Allan et al. [9] had suggested, that performance may improve with understanding, became a self-fulfilling prophecy. Good intentions motivated the suggestion, but the research community interpreted it too literally. Allan et al. [9] suggested that algorithms might improve with "some limited form of story parsing and understanding", and the community responded with a limited form of linguistic understanding. Allan et al. [9] also suggested that "the gains [in accuracy] may not be large", and the community responded with minimal gains.

TDT literature rarely explored, let alone challenged, the suggestion. It rarely pushed the limits of understanding. Even today, understanding still rarely accompanies algorithms, and only at a distance. Limited understanding drives techniques in few instances [49], like in the participant timelines of Huang et al. [101] and McMinn and Jose [158]. Everywhere else, knowledge appears merely as an accessory, an incremental change applied to established algorithms, or not at all. Incremental changes led only to incremental improvements.

In this chapter, we prove Allan et al. [9] right—and wrong. We prove them right by the gains in performance achieved by a novel algorithm designed for Twitter and driven by understanding. We prove them wrong by demonstrating that the gains in performance can be large, that the gains do not apply only to accuracy, and that understanding can drive and not simply accompany TDT. In this chapter, we study the application of understanding in a classical TDT context, football matches on Twitter, and answer the following questions:

- What makes the ideal TDT algorithm? The TDT task has evolved since Allan

et al. [9] first proposed event understanding as a solution to the area's challenges, but while accuracy represents a primary concern, it no longer remains the only concern. In Section 5.1 we outline eight aspects of the ideal TDT algorithm and describe event understanding's role in the hypothetical standard.

- How can event understanding drive TDT? The research community satisfied itself with cursory applications of event understanding, rarely applying it to drive event detection [49]. In Section 5.2, we present SEER, the first TDT algorithm driven by the automatic understanding of What may happen in events.

- In what ways can event understanding improve TDT, and to what extent? From the original suggestion of understanding [9] to participant timelines [101; 158] and everything in-between, the research community alluded to but rarely studied the benefits of understanding. In Section 5.3, we evaluate the effects of understanding on two aspects of the ideal TDT algorithm: precision and comprehensiveness.

Due to space constraints, we present a study on the benefits of understanding to sensitivity, a continuation of Section 5.3, in Appendix B.

5.1 | The ideal TDT algorithm

The TDT community obsesses with accuracy. Even Allan et al. [9] only regarded understanding as a way of improving accuracy. Like precision, recall and other derivative metrics, accuracy made sense in 1998, when metrics measured an algorithm's ability to fulfil the role of a news aggregator, but it makes less sense today. When the research area migrated to Twitter and its role changed from news aggregator to automatic newswire, its metrics needed to change too.

Today's ideal TDT algorithm is more than just precise and comprehensive. The ideal algorithm is also expressive [191] and timely [62; 191; 266] in its output, sensitive and scalable regardless of the event's popularity, lightweight and efficient [62; 191], parameter-free, and portable to all events, domains and languages. In this section, we explore each quality in detail, investigating the motivations behind these eight characteristics, the design choices that empower algorithms and the necessary compromises that inhibit them.

Comprehensive

First, the ideal TDT algorithm is comprehensive. Data is scarce, but already in 2015, Twitter could generate 500 million tweets per day [258]. In the span of two hours, even football matches can produce millions of tweets [161; 162]; the 2014 World Cup final generated 1,907,999 tweets [161]. Twitter's high-volume, high-velocity streams would overwhelm anyone seeking information [6; 26; 129], but the ideal TDT algorithm synthesises all the details in a dataset into a few topics.

Borrowing from IR literature, the TDT community evaluates comprehensiveness using recall. In football matches, annotators calculate recall on easily-enumerable topics: goals, at least, but also yellow and red cards, the starts and ends of halves, and substitutions, as we explain in Appendix A. Few direct comparisons between systems exist in literature, but some patterns still arise across papers and events.

Distant comparisons across papers make immediately clear research's difficulties in designing comprehensive algorithms. TDT systems have a habit of capturing popular key topics with relative ease but of missing non-key topics with equal ease. Meladianos et al. [162], for instance, captured all goals from 17 matches but less than a third of yellow cards. Many others [146; 150; 183], alongside Meladianos et al. [162], captured key topics but missed non-key topics, forming an unmistakeable pattern.

In our previous work [146], we designed ELD precisely to overcome these challenges. ELD captured more non-key topics—yellow cards and substitutions—than the baseline from Zhao et al. [296], and performed with consistency on datasets of varying sizes. While we designed ELD to prioritise comprehensiveness, however, it still missed, with the same consistency, certain topics: non-key topics involving the less popular team or the team with a non-English-speaking fanbase, and other topics rendered inconsequential by the event's scenario, like a last-minute yellow card.

ELD overcame the technical challenges but stumbled at the behavioural ones. Twitter talks about what it finds interesting [150]. When Olympique Lyonnais faced VfL Wolfsburg in the 2020 UEFA Women's Champions League final [238], only 1 tweet from over 11,000 mentioned Dzsenifer Marozsán's yellow card and, incidentally, misspelt her name. Twitter simply did not find Marozsán's yellow card interesting. Nevertheless, newsworthiness, as opposed to interestingness, is not measured by what mobilises the masses [71]; non-key topics can be newsworthy without being interesting [136; 266].

While TDT research can do little to overcome such behavioural challenges, it can, at least, overcome the technical limits. The event shadow [121], or the way seconds-long key topics stretch for minutes in Twitter's discourse [266], presents a technical challenge. The fat-tailed distribution of tweets that follows key topics overshadows non-key top-

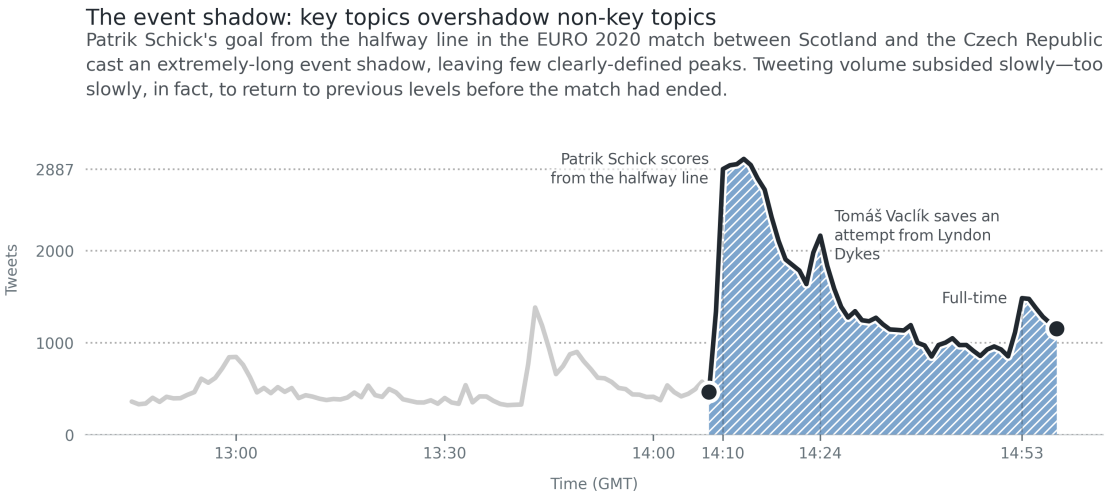

Figure 5.1: Extraordinary topics hide subsequent non-key topics. Twitter users keep discussing key topics for a long time, increasing the likelihood of detecting duplicate topics and decreasing the likelihood of detecting other topics. Lanagan and Smeaton [121] called the phenomenon the "event shadow".

ics [150; 226], as shown in Figure 5.1, but it does not erase their existence. Consequently, identifying the non-key topics hidden in the shadow of key topics becomes a technical challenge.

Different approaches handle the event shadow with varying success. Document-pivot approaches do not rest on tweet volume and so remain unaffected by topic popularity as long as enough content exists to form sufficiently-large clusters. Conversely, feature-pivot techniques that rely on spikes in volume to detect topics miss non-key topics, dwarfed completely by the event shadow [226]. Therefore comprehensiveness leaves no space for the traditional volume-based, feature-pivot approaches, the earliest and simplest methods of early TDT research on Twitter [28; 97].

ELD was more comprehensive than simpler methods, even on datasets with a few tens of thousands of tweets, because it overcame the technical limits. ELD first clustered tweets using an incremental document-pivot approach, which bypassed the event shadow. Then, the feature-pivot technique could extract bursty keywords from each individual cluster, regardless of the topic's popularity. Nevertheless, underneath the event shadow, ELD uncovered not only non-key topics but all forms of noise too, and contended with IR's classical compromise: the precision-recall trade-off.

Precise

Second, the ideal TDT algorithm is precise. A TDT algorithm would be completely comprehensive by classifying every tweet as a topic, but the ideal algorithm cannot presume to minimise information overload without also detecting what is not a topic. In other words, the ideal algorithm synthesises all the information in a dataset into a few but precise topics [129].

Precision assumes various forms, but certain types of topics are clearly imprecise. Precision must entail veracity, although Liu et al. [130] found fake news to be far less prevalent than harmful. Noise, such as spam and advertisements [206], is also noisy by definition: the type of ubiquitously and undesirable content that distinguishes TDT on Twitter from TDT in more formal mediums [191; 275]. By some accounts, noise constitutes up to 95% of datasets [158], as we describe further down. Other types of topics are borderline cases, at least in traditional TDT research.

In Appendix A.3, for example, we argue that redundant topics should be considered equally as imprecise as noise. Over the years, perhaps driven by the early misguided notion, unproven on Twitter, that topic detection represents TDT's more difficult task [289], research ignored the second task, topic tracking. The event shadow poses a challenge because algorithms fail to recognise the same key topic being discussed in marginally-different ways. In short, the event shadow challenges topic tracking as well as topic detection. Literature, however, seems to have tacitly accepted redundancy, rarely acknowledging it and downplaying the actual high rates of topic duplication in its timelines [277].

Literature has offered few solutions to improve precision. Some approaches train models to learn the domain's vocabulary, and thus retain the precise instead of filtering the imprecise [95; 99; 297; 298], but such filtering methods appear uncommonly. More frequently, research filters noise, like how Hasan et al. [93] use a manually-compiled list of 350 spam terms to remove 70% of all tweets, or how McMinn and Jose [158] remove all tweets without named entities—95% of all tweets.

Noise does not necessarily make up 70% of all tweets, much less 95%. Nevertheless, Hasan et al. [93], and McMinn and Jose [158] demonstrate to what extent aggressive filtering became the norm. Like McMinn and Jose [158], many others aggressively filter all retweets [162; 226]. Research justifies retweet filtering by arguing that retweets contribute noise and no additional new information [101; 234], and thus removing them helps algorithms scale to high-volume streams. However, if users and tweets represent sensors and signals [227], then do retweets not represent boosts to the sensors' signals?

Aggressive filtering takes other forms too. In document-pivot approaches, research

retains only large clusters since many forms of noise scatter across clusters and do not form sizeable groups. Ozdikis et al. [189] boldly rejected clusters with fewer than 250 tweets. Others set a relatively more lenient threshold of 10 tweets [93; 102; 158], or retained only the largest clusters [102; 197; 264]. Yet even lenient thresholds become stringent in unpopular events.

Despite research's best intentions, aggressive filtering only reinforces the precision-recall trade-off. Research ignores the obvious: non-key topics seldom attract enough attention to form large clusters. Retweet filtering further exacerbates the problem; the fewer the tweets, the fewer the clusters that reach the threshold. Moreover, retweets occupy an increasingly-large share of datasets, from around 30% a few years ago [157] to more than 40% in our experiments. And so precision increases, but recall inevitably decreases too.

Expressive

Third, the ideal TDT algorithm is expressive. Earle et al. [55], Kumar et al. [119] and Sakaki et al. [227] all designed systems to detect earthquakes, but the three algorithms express themselves with varying degrees of clarity. The system by Earle et al. [55] can only detect When an earthquake occurs. The system by Kumar et al. [119] goes further and extracts key terms to express, vaguely, What happened. Differently from Earle et al. [55] and Kumar et al. [119], as soon as the early warning system by Sakaki et al. [227] detects an earthquake, it geo-locates Where the earthquake occurred and sends alerts. Even in detecting topics, Sakaki et al. [227] express and describe.

Detecting and tracking topics with accuracy no longer suffices: TDT methods need to describe topics too [191]. We do not mean that an algorithm should assume the role of a summariser or a visualiser, or describe topics in any particular way. We mean that the raw output of an event tracking algorithm rarely suffices in practice; Earle et al. [55] aspired to provide situational information, but detecting merely that an earthquake occurred somewhere on Earth without geo-locating the place aids neither human nor machine. The ideal TDT algorithm guides the reader, the summariser or the visualiser with its output.

The choice of model again determines the potential of an algorithm to be expressive. Document-pivot algorithms identify topics based on what Twitter users discuss. Even without any form of summarisation, document-pivot algorithms provide human-readable summaries through the clusters' tweets [102]. In contrast, feature-pivot algorithms identify topics based on how Twitter users discuss them, and vary more in their expressiveness.

Like in Earle et al. [55], a feature-pivot algorithm may simply detect abnormal tweeting behaviour. Regardless of how precise or comprehensive, such traditional volume-based techniques distil no information from a topic [28; 97]; they detect spikes, but not their subject. Determining the subject requires extra work. Nichols et al. [183], aware that their volume-based approach detected without describing, complemented the technique with a component to seek the tweets responsible for spikes.

Alternatively, like in Kumar et al. [119], a feature-pivot algorithm may identify popular or bursty keywords. Scattered keywords with no context, however, only set a blurry scene [96], an incoherent story that cannot serve as situational information. For example, Kumar et al. [119] extracted five keywords to describe an earthquake that hit Italy's Emilia-Romagna region in 2012. The keywords included *Emilia* and *chies[a]* (Italian for *church*) but did not name any particular church in Emilia-Romagna, nor indicate what happened to or in it.

Being expressive can also result from simple forms of understanding. The participant timelines by Huang et al. [101], and McMinn and Jose [158], otherwise identical to TDT literature's standard monolithic timeline, express something about an event's participants. The two-level system by Yang et al. [289] classifies documents with labels such as *tornadoes* or *bombings* before clustering them, and in the process expresses the broad subjects of topics. Similarly, Farnaghi et al. [61] used location-based clustering to express geographical situational information about Hurricane Florence, succeeding where Earle et al. [55] and Kumar et al. [119] failed.

Responsive

Fourth, the ideal TDT algorithm is responsive. The early warning system by Sakaki et al. [227] relied on users reacting like sensors to early tremors to warn others about the slower but more destructive waves that could follow. The developed system had utility because it could send earthquake warnings quicker than the Japan Meteorological Agency; citizens could turn off stoves and take cover from falling objects [227]. In such cases, a warning that arrives too late has as much value as one that never arrives.

The utility of responsive systems extends beyond emergency situations. A responsive algorithm is more convenient and useful than an unresponsive one. Reuters Tracer provided utility not only through the number of news alerts it raised, magnitudes higher than those raised by its network of more than 2,500 journalists [129; 130]. Reuters Tracer also provided utility because its alerts almost always preceded competing newsrooms [130]. The ideal TDT algorithm exploits Twitter's potential by reacting to news with low latency [62; 191; 266].

A TDT algorithm could still have valid reasons for sacrificing responsiveness. When a topic occurs during an event, it shifts the conversation [6; 161; 162]. Feature-pivot approaches often use fixed or sliding time windows as reference points to observe changes in volume or shifts in discourse. The shorter the time windows, the shorter an algorithm's response time but the more jarring the shifts too. The longer the time windows, the smoother the shifts [150] and consequently, the more reliable the algorithm's output.

Responsiveness, then, becomes a trade-off for reliability, or precision. Zhao et al. [296] engineered ten-second time windows that could detect topics from American football games within around 40 seconds. However, as a baseline applied to quiet streams in our previous work [146], the algorithm misconstrued any minor deviation in volume as a topic. Therefore depending on the systems' requirements of precision, time windows stretch from seconds [296] to minutes [226] and hours [260], and sometimes days [61].

Researchers must carefully manage the trade-off between reliability and the practical value of an algorithm. Algorithms that must amass data before extracting precise information rob the output of its practical value [275]. Consider the situational information generated by Farnaghi et al. [61] and their algorithm, by all accounts expressive; the algorithm's one-day time windows make it ineffective situational information. At best, the system fulfils little more than an archival role when its reporting has more latency than a human journalist's.

Scalable but sensitive

Fifth, the ideal TDT algorithm is scalable but sensitive. A scalable and sensitive algorithm performs in the same way on events that generate millions of tweets as it does on events that generate a few hundreds. In other words, the ideal algorithm is scale-invariant: comprehensive and precise, expressive and responsive in events both popular and unpopular.

Sensitivity, in particular, represents a different utility than comprehensiveness. Most TDT algorithms can boast of being scalable, but few events produce the volumes of tweets that challenge scalability. We collected just 2,774 tweets in English over more than 2 hours from the 2020 Copa Del Rey final, played on 3 April 2021 between Athletic Club and Real Sociedad; a popular event could generate 3,000 tweets in a minute with Twitter's API limit of 50 tweets per second [296]. A sensitive and scalable algorithm would allow us to build quality timelines even for events with low coverage or with a non-English-speaking audience, like the Copa del Rey final.

While most TDT algorithms scale, few can boast being sensitive, for sensitivity symbolises a costly trade-off for precision. Sensitive algorithms have to maintain a metic-

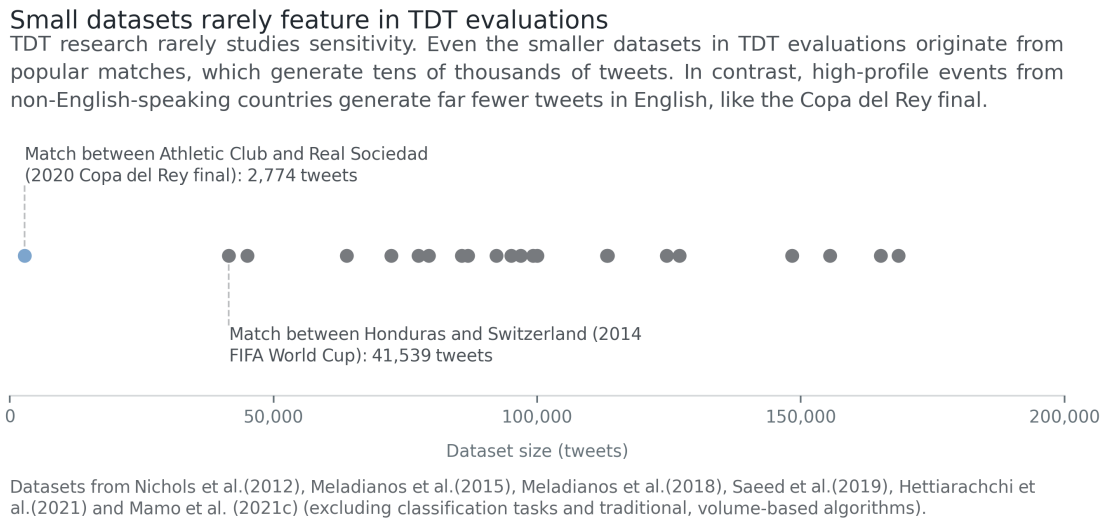

Figure 5.2: Few TDT publications experiment with small datasets from unpopular events. Excluding classification tasks and simple volume-based algorithms, datasets range from a few tens of thousands of tweets to millions.

ulous balance between granularity and precision. Like longer time windows increase reliability, the large communities behind popular events lead algorithms with the wisdom of the crowds. In unpopular events, however, the crowd is smaller and its wisdom proportionate.

Modern literature retains an unrealistic view of small datasets. Studies make corpora with 40,000 [162] and 60,000 [143] tweets seem small, as shown in Figure 5.2, ignoring the truly unpopular events, which produce far smaller datasets. The most sensitive algorithms remain the traditional and "naïve" [28; 97] volume-based methods, which trade sensitivity for precision, comprehensiveness and expressiveness. Others limited the scope of the algorithm by approaching TDT as a classification task [265] or by applying rigid, manually-defined understanding [136].

Scalability and sensitivity demand algorithms designed for and dedicated to them. Scalability excludes the heavy processing of document-pivot algorithms, which must be simultaneously sensitive to unpopular events' quiet streams and scalable to popular events' high-volume, high-velocity streams [62; 93]. In comparison, feature-pivot algorithms face fewer problems; discourse changes in unpopular events as it does in popular events, as we show in this chapter.

Sensitivity makes even steeper demands. Sensitivity excludes the aggressive tweet filtering of Hasan et al. [93], and McMinn and Jose [158], which makes large datasets small, and small datasets even smaller. Sensitivity also excludes the aggressive cluster

filtering of Hasan et al. [93], Ifrim et al. [102] and Ozdikis et al. [189], which spares few clusters in unpopular events, as we show in Appendix B. Finally, sensitivity excludes buffering an impractical amount of tweets that may never arrive. After all, the Copa del Rey final's 2,774 tweets would not even have filled two of the 1,500-tweet bins that Corney et al. [44] used.

Lightweight and efficient

Sixth, the ideal TDT algorithm is lightweight and efficient. Tweets arrive in great volumes and at high velocities, leaving algorithms with little time to process and detect [62; 157; 191]. Being lightweight and efficient is a measure of an algorithm's resource demands [62; 129; 191] and a measure of feasibility [274]: how much processing can an algorithm shed while still meeting performance requirements?

Above all being lightweight and efficient is a pre-condition for other qualities. To be responsive to Twitter's high volumes, the ideal algorithm must process tweets efficiently, without buffering data unnecessarily [191]. To be sensitive and scalable, the ideal algorithm must be efficient to match Twitter in throughput. Finally, the ideal algorithm remains comprehensive and precise despite being lightweight and efficient.

The earliest TDT systems on Twitter represent the epitome of efficiency. Marcus et al. [150], Nichols et al. [183] and others [55; 296] spent no time at all studying individual tweets. Instead, they simply studied Twitter's overall behaviour: how tweeting volume changed over time. However, early research on Twitter quickly recognised that systems could not easily remain comprehensive and precise while being lightweight and efficient.

The complexity increased. Algorithms progressed from short, 10-second time windows [296] to minute-long blocks. To overcome the event shadow, research studied the tweets' keywords [28] and their authors [97]. And to overcome the event shadow without sacrificing precision, in ELD we combined document-pivot and feature-pivot methods into one heavyweight and inefficient method [143; 146].

Document-pivot algorithms symbolise the anti-thesis of efficiency. *K*-means' repeated passes over thousands of tweets and its reliance on batched data led to Panagiotou et al. [191] arguing that Twitter leaves no space for document-pivot algorithms, especially in real-time systems. ETree, a real-time event modeller and miner, required "effective and efficient event modelling", and so Gu et al. [86] avoided complex clustering, "infeasible due to its low efficiency" [86].

Like in ETree, clustering survived Twitter in the form of single-pass, incremental algorithms. In the general recipe of incremental clustering, the algorithm assigns a tweet

to the most similar cluster if one exists, and otherwise creates a new cluster for the tweet. Nevertheless, while admittedly more efficient than traditional clustering, single-pass algorithms remain inefficient: they compare a tweet with all the clusters in memory to finally assign it to just one cluster. Perhaps Panagiotou et al. [191] were right, and document-pivot methods really are infeasible on Twitter after all.

Parameter-free

Seventh, the ideal TDT algorithm is parameter-free [50]. Parameters adapt an algorithm to the data by balancing the trade-offs between qualities: between precision and comprehensiveness, between precision and scalability or sensitivity, and between precision and responsiveness. A measure of human control may appear desirable, but the ideal algorithm simply has no need for parameters. It detects events without human bias, perspectives and interpretations, and in the words of Keogh et al. [109] "let[s] the data itself speak to us".[1]

A parameter-free algorithm is a mark of robustness. An algorithm laden with parameters cannot be portable, neither to different languages and domains, nor to events. TDT's early research on Twitter produced robust algorithms [150; 183; 296]. The technique by Lanagan and Smeaton [121], for example, simply seeks the highest peaks in volume. No parameters shackle such algorithms with unreasonable assumptions about the data, so they adapt effortlessly to all languages, domains and events.

By extension, a parameter-free algorithm is a mark of convenience. The settings that optimise an algorithm for a popular event mislead it in an unpopular one [275]. In evaluations, parameters leave researchers scrambling for the optimal configuration in domains where even small changes can make big differences [276]. Unsurprisingly, as we demonstrate in Appendix A.1, empirically-set parameters prevail in TDT literature.

A parameter-free algorithm is also a mark of elegance. Elegance represents an enviable pursuit, but elegant algorithms do not easily overcome the technical challenges. Buntain et al. [28] and Hsu et al. [97] recognised that the elegance of techniques like the one proposed by Lanagan and Smeaton [121], which the former called "naïve" [28], makes them simplistic for the needs of modern TDT. The existing elegant algorithms do not distinguish between an opinion and a fact, and cannot overcome Twitter's technical

[1] We distinguish between a parameter-free algorithm, and a lightweight and efficient algorithm. A parameter-free algorithm is not necessarily lightweight and efficient, and vice-versa. The *K*-means algorithm, used as a document-pivot technique, has few parameters—the number of clusters *K* and the distance measure— but it is notoriously inefficient. Conversely, while the algorithm we present in Section 5.2 has several parameters, it remains lightweight and efficient.

limits. In short, like lightweight algorithms, parameter-free techniques often create a trade-off between elegance, and comprehensiveness and precision.

The chase for parameter-free algorithms is what led Fung et al. [72] to propose feature-pivot techniques. Before Twitter had even launched, Fung et al. [72] complained that document-pivot algorithms were notoriously parametric: from choosing the smallest acceptable cluster and the similarity measure to deciding how to handle the temporal factor [35] and fragmentation [72]. Twitter only exacerbated the challenges associated with parameters, which nowadays have to be set empirically, as we explain in Appendix A.1. Therefore Fung et al. [72] proposed feature-pivot techniques, loosely-based on automata, which Kleinberg [113] had previously proposed.

Feature-pivot techniques escaped the criticism reserved for document-pivot algorithms but only succeeded in slightly reducing parameters. The length of time windows affects both precision and comprehensiveness [150], so it must be considered a parameter. Burst normally does not accept or reject topics with absolute confidence but takes a range of values—the burst's cut-off point too must be considered a parameter. In fact, few feature-pivot algorithms eliminated parameters completely; even the feature-pivot technique by Lanagan and Smeaton [121] has an arbitrarily-set parameter: a minimum of ten topics in every event.

Nevertheless, parameter-free algorithms are not a quixotic ideal. Algorithms with few or no parameters can stem from deliberate choices. Lappas et al. [122] identified periods of high activity based on an earlier algorithm by Ruzzo and Tompa [224], which requires no parameters. Similarly, on Twitter, Farnaghi et al. [61] and Madani et al. [138] used Hierarchical Dirichlet Processes, not Latent Dirichlet Processes, to reduce parameters.

Portable

Eighth, the ideal TDT algorithm is portable. A portable algorithm should work with different languages, and adapt to every domain and every event. While the researchers' assumptions, reflected in parameters set empirically on sample data, harm portability [275], portability remains the TDT community's unspoken rule. Technical limitations rarely bind an algorithm to a single language, domain or event. Users flare like sensors when they feel an earthquake's tremors [227], and they flare like sensors when a footballer scores a goal [183].

Only understanding harms portability. Very few studies—just Buntain et al. [28], to the best of our knowledge—discuss the effects of understanding explicitly. Nevertheless, if we consider developing understanding to be a separate task from the TDT

process, then event and domain understanding binds an algorithm that depends on it to a single event or a single domain. An algorithm that understands one event cannot be portable to other events, and an algorithm that understands one domain cannot be portable to other domains.

Buntain et al. [28] questioned the utility of understanding when it harms portability. At first, they considered tracking certain keywords, like *goal* in football matches, to detect topics. If the TDT algorithm could not track penalties or missed chances, the lexicon could grow to accommodate the new topic type. Buntain et al. [28] seemed inclined to accept that understanding limits an event tracker to one language, but their consideration ended abruptly when they tried to account for every remote topic.

Buntain et al. [28] were right. Understanding has questionable utility in the face of exceptional topics. We can enumerate the expected key and non-key topics in a football match, but not exceptional topics: a player biting another [28], a terrorist attack happening on the perimeter of a stadium [203], or a parachutist mistaking a football pitch for a landing field [246]. Exceptional topics have an inherent value of newsworthiness by virtue of being exceptional [51], and understanding fails utterly at capturing them.

Zhang et al. [292] sought to design the ultimate generalisable TDT method. They philosophised that a topic can be reduced to tweets, topics and their time intervals. When a topic happens, certain semantic aspects in the tweets change, and those same changes embody the topic. Thus, their design solicited the anomalies in semantic aspects, a simple word embedding, as portents of a topic. Yet Zhang et al. [292] placed excessive value in generalisability, and the algorithm could not consistently out-perform simpler baselines.

Therefore while Buntain et al. [28] were right, we still disagree. The qualities in this section read like trade-offs for precision, but in reality, portability is a trade-off for every other quality. Like Zhang et al. [292], the research community trades portability for comprehensiveness, expressiveness, responsivity, scalability and sensitivity, efficiency and elegance. The community trades portability for precision itself. Buntain et al. [28] questioned the utility of understanding; we question whether research affords to continue ignoring it any longer.

In the end, even Buntain et al. [28] reconsider understanding. They reflect on understanding as a "potential opportunity ... in combining domain knowledge with ... domain-agnostic foundations."

Portability has failed TDT. Faced with a choice between general and portable algorithms that achieve limited results in any domain, and specialised algorithms designed

to exploit knowledge about one domain, research repeatedly chose the former. In this dissertation, we break the prevailing trend and investigate whether a sacrifice to portability can lead to the "significant advances" that Allan et al. [9] envisaged. We demonstrate the benefits of our choice in the rest of this chapter.

5.2 | SEER: Stream-Enabled Event Reporter

In many ways, ELD had succeeded. The final solution proved responsive, portable and comprehensive without incurring harsh penalties to precision. Penalties did appear, but elsewhere. ELD's combination of document-pivot and feature-pivot techniques was so parametric that we were forced to set many of its parameters empirically. Its clustering was so complex, cumbersome and inefficient that we were forced to slow down the streams of popular events. Simultaneously, ELD's clustering restricted the algorithm's ability to scale up to popular events and to be sensitive to unpopular ones. In many ways too, ELD had failed.

In this section, we present Stream-Enabled Event Reporter (SEER) as a solution to the challenges that ELD faced. SEER does not simply contribute a novel TDT algorithm—the research community has proposed many and progressed little. Instead, it contributes a novel model to apply understanding in TDT, one that creates separate timelines for separate types of topics. In our case study on football matches, SEER constructs a timeline about goals, another about yellow cards, and several more timelines, one for each type of topic.

SEER, like ELD, splits processing into two parts, as shown in Figure 5.3: an understanding period before the event, and the actual event period [146]. During the understanding period, SEER establishes a baseline of the event's vocabulary by constructing a term-weighting scheme. During the event period, it separates tweets into topical streams and detects topics in each stream separately.

In this work, we implement SEER's architecture to simplify ELD. We remove ELD's core, the then-novel feature-pivot algorithm, and embed it in SEER's understanding-driven architecture to detect topics in each stream. The improved solution remains as responsive and comprehensive as ELD but becomes more efficient, sensitive and precise. In the end, SEER only sacrifices portability. We describe the architecture in more detail in the rest of this section.

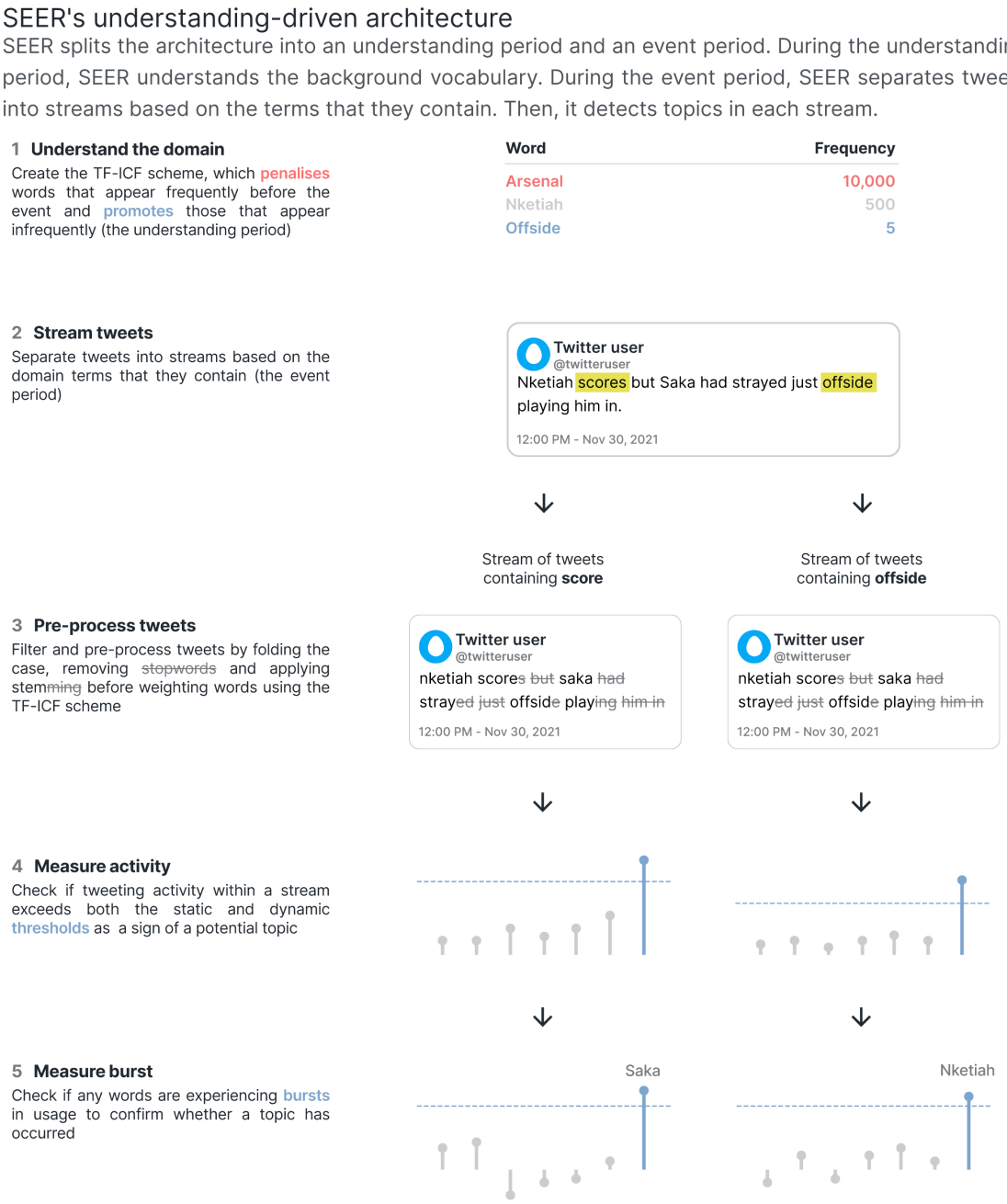

Figure 5.3: SEER's architecture splits processing into an understanding period and the actual event period. During the understanding period, SEER understands the event's background vocabulary as a TF-ICF term-weighting scheme. During the event period, SEER splits tweets into streams and detects topics in each.

The understanding period

The understanding period complements, but does not replace, EVATE's domain understanding from Chapter 4. EVATE understands the domain's vocabulary; SEER understands the event's background vocabulary. The occurrence of topics shifts the event's vocabulary [161; 162], but the background never changes. In a match opposing Leicester City and Manchester United, the two teams exist persistently in the background: every topic involves, directly or indirectly, Leicester City and Manchester United. Therefore during the understanding period, SEER establishes a baseline of the event's vocabulary to accentuate later shifts in the discourse due to topics.

SEER, like ELD, represents the background vocabulary as a term-weighting scheme based on TF-ICF [213]. The TF-ICF scheme adapts the standard TF-IDF to dynamic document streams and, incidentally, to Twitter's brevity by calculating IDF on a separate, static corpus. In this work, the static corpus is a collection of tweets, D_u, collected during the understanding period. Later, during the event period, TF-ICF combines the term frequency of a word t in a tweet d, $TF_{t,d}$, with the ICF scheme:

$$\textit{TF-ICF}_{t,d} = TF_{t,d} \cdot \log \frac{|D_u|}{|\{d' \in D_u | t \in d'\}| + 1} \tag{5.1}$$

We employ TF-ICF during the event period to promote words that appear prominently during the event but not before it. In particular, TF-ICF boosts the topical keywords. Twitter users cannot discuss goals that have not been scored before the event, so TF-ICF boosts any mentions of the words *score* and *goal* during the event. More generally, the less frequently a word appears during the understanding period, the higher its TF-ICF weight during the event period, and vice-versa.

Term clustering

As humans, we perceive semantic concepts in isolated terms. In football, a foul is not the literal *foul* but a well-defined concept depicting an action and its consequences. Similarly, the concepts of fouls, tackles and yellow cards do not exist in isolation but as linked concepts with overlapping information: a mistimed tackle leads to a foul, and a foul possibly leads to a yellow or red card. SEER's topical streams, which process tweets separately based on the terms within, represent such concepts. We define each topical stream as a group of terms by clustering EVATE's domain terms, thus developing the literal *foul*, *tackle* and *card* into the human-like concept of a foul.

In word clustering, the same elusive, abstract notion of ATE re-appears: semantics. Formally, word clusters represent "sets of words that share a significant aspect of

their meaning" [48], but neither aspect nor meaning have clear definitions. Therefore, like in ATE literature, applications of word clustering—from word sense disambiguation [53; 112; 271] to sentiment analysis [272], and from text classification [46] to short text clustering [182]—interpreted semantics in different ways.

Word clustering adopted interpretations of semantics from other areas of IR. From document clustering, word clustering adopted the interpretation that words in a cluster should be similar to each other and dissimilar to those in other clusters [46]. From bootstrapping, word clustering adopted the interpretation that contextual and syntactical cues can gauge semantics [46]. Like in ATE literature, if the research community could not measure the semantics of word clusters quantitatively, it could, at least, estimate them. In SEER, we also borrow interpretations from document clustering and bootstrapping to estimate semantics.

Our word clustering algorithm revolves around the idea of distributional similarity. The idea, traditionally attributed to Harris [92], states that "words that occur in the same contexts tend to be similar" [192]. We consider that two terms share a context if they appear in the same tweet, an assumption enabled by the brevity of tweets. We estimate contextual similarity using chi-square, from Equation 4.9 on page 83; PMI normally overestimates the similarity of rare word pairs [154].

After calculating the contextual similarity, we construct a word graph with terms as nodes, connected with edges weighted to reflect the chi-square score. Out of all edges, we retain only links in the 95^{th} percentile of contextual similarity to eliminate misleading, possibly incidental associations between domain terms. Then, we apply the Girvan-Newman algorithm [80], repeatedly removing the edges with the highest betweenness until we have partitioned the graph into the desired number of components: the human-like concepts of topic types.

Our algorithm only has one parameter: the number of clusters. The larger the number, the smaller the clusters and the more fragmented the concepts [106] but the purer the meaning of each group. The smaller the number, the larger the clusters and the more likely that different concepts merge [106]. At one extreme, singleton clusters describe the precise, unambiguous ideas of single domain terms; at the other extreme, a solitary cluster with all terms describes the entire domain, not its concepts or topic types.

Choosing the number of clusters presents a particularly-difficult challenge in event domains. Meladianos et al. [161] argued that TDT in unplanned events, like breaking news, faces fewer difficulties than TDT in planned events, like football matches, whose topics share a common vocabulary. In football matches, for example, seemingly-disparate concepts intertwine: the referee may consult the VAR to disallow a goal due to a foul or an offside in the build-up. Consequently, the ideal cluster, which collects do-

main terms that refer, unambiguously, to one topic, appears as an unreachable ideal. We could not find prior work on word clustering in event domains, nor a suitable ground truth to help us establish the optimal number of clusters.

Instead, we set the number of clusters empirically. We constructed the graph using the top 70 terms from EVATE's bootstrapped lexicon and clustered domain terms into 15 groups with a modularity of 0.7034. The value lies at the higher end of what Newman and Girvan [178] consider attainable, but it falls just below the optimal value: 10 clusters and a modularity of 0.7514.[2] Nevertheless, we find that the 15 clusters make more sense thematically, as we show in Table 5.3 on page 137: one cluster describes spam words, another refereeing decisions, and yet another goals. Moreover, the finer selection of clusters allows us to understand better the effects of different types of understanding in Section 5.3.

Topical streams

What distinguishes the algorithms by Huang et al. [101], and McMinn and Jose [158] from others are the unconventional participant streams. Both sought topics, like any other TDT approach, but neither knew What might happen in advance—only Who could make it happen. The participant streams served as a proxy to the question: What will the participants do? SEER draws inspiration from Huang et al. [101], and McMinn and Jose [158], but we know What can happen in advance. Thus, SEER requires no proxy and can answer a more direct question: What will happen?

If the intuition that drove EVATE is that topics generate topical domain terms, then the intuition that drives SEER is that topical domain terms generate topics. The way Twitter users discuss events lends credibility to our intuition. A topic, though unexpected, flows predictably; topical conversations revolve around a small set of keywords [8; 42], the domain terms, and a few other event terms. Brevity pushes tweets to revolve even more closely around a few but unambiguous topical domain terms [287].

SEER's intuition, driven by the knowledge about What can happen in a domain's events, eliminates the need for the Who. The Who changes across events, and so an algorithm would have to identify participants for each event using either APD or NER, which incur overhead. In contrast, the What remains fixed across all events in the same domain, and therefore it depends on the one-time output of an ATE technique, which does not impinge on the TDT algorithm's processing.

[2]More generally, modularity favours large clusters. Topics in the same event domain share a common vocabulary [161], which leaves only a murky distinction between concepts.

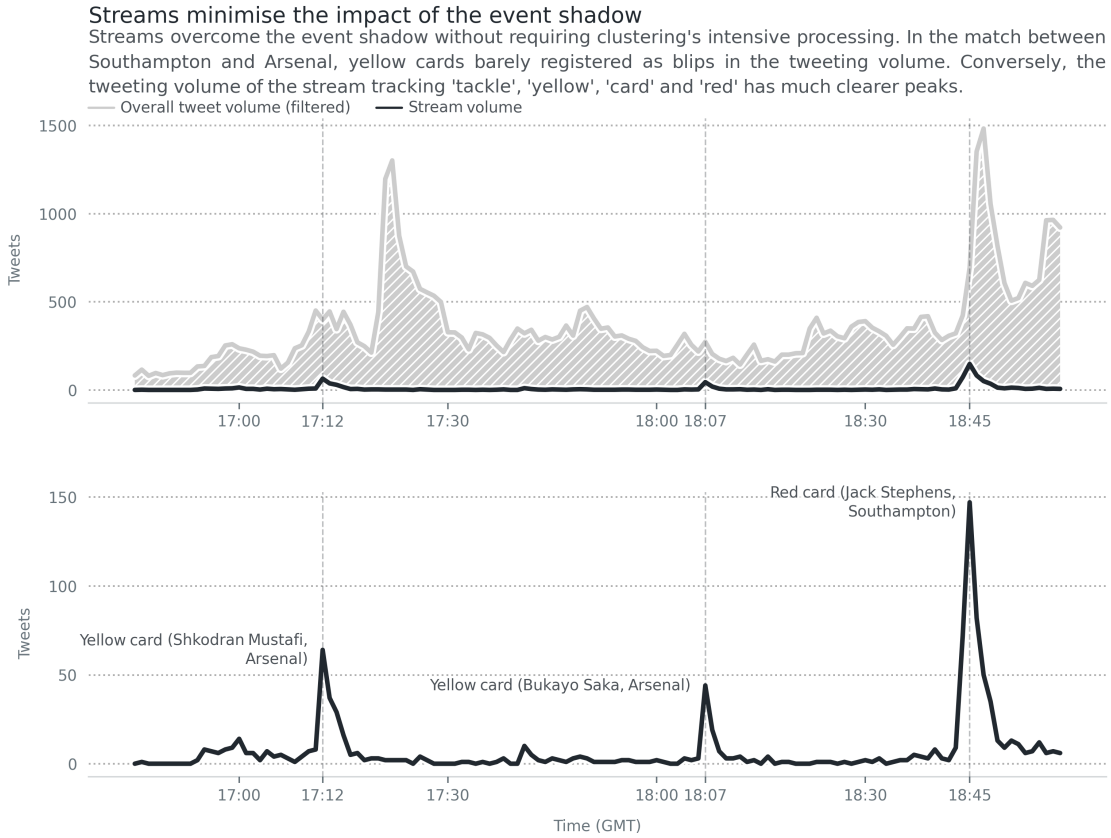

Figure 5.4: Streams isolate different types of topics to minimise the event shadow. Above, tweets mentioning the terms *tackle*, *yellow*, *card* and *red* make a lean impression on the overall tweeting volume. Isolating those tweets, however, makes yellow and red card offences much clearer, without having to resort to clustering.

As tweets arrive, SEER dispatches them to the corresponding streams, which we provide as an input. SEER decides which streams receive the tweets by removing stopwords and stemming any words that remain using NLTK's Porter Stemmer [22]. Each stream handles a different set of tweets: if a tweet mentions the word *goal*, SEER forwards the message to the stream whose cluster includes the term *goal*.

The topical streams shift our perspective of events. In Figure 5.4, the stream's narrow focus on card offences promotes Bukayo Saka's yellow card in Arsenal's match against Southampton from a barely-noticeable blip to a clearly-demarcated spike. The topical streams shift the perspective of algorithms as well as humans. Even to the early and trivial volume-based algorithms of TDT research, Saka's yellow card would appear, unmistakeably, as a topic.

SEER's keyword-based streaming acts as an aggressive, albeit sensible, filter. A

tweet may be processed by one or more streams, depending on the present keywords. If a tweet mentions both *goal* and *offside*, SEER forwards it to the streams whose clusters include *goal* or *offside*. Just as likely, a tweet may be processed by no stream at all; at this stage, our novel algorithm will already have discarded around 50% to 60% of all tweets for not containing any domain terms.

Later, each stream applies even more filters. Some filters derive from our previous works [142; 146] and others from our findings in Appendix A. SEER retains retweets but filters replies and quoted tweets, both normally used to react to topics rather than to proactively report about them. SEER also filters tweets with more than two hashtags or with URLs, noisy behaviour to maximise message reach. Finally, it filters tweets by users with the aspects of career spammers: those who left their profile descriptions empty, who have fewer than one follower per thousand tweets, or who never tweeted or favourited a tweet before.

By now, SEER must be starting to resemble the TDT algorithms that we criticised for filtering aggressively. Indeed, SEER does not hesitate to filter aggressively, but it filters differently, more intelligently and purposefully than other algorithms. It does not filter indiscriminately, on perceived notions of what could be noise, but on well-understood notions of what could be relevant to the event domain based on EVATE's keywords. More importantly, as we demonstrate throughout this chapter, SEER's aggressive filtering inhibits none of the algorithm's qualities.

Topic detection

Internally, the topical streams behave similarly to ELD [146], at least initially. Within each stream, SEER pre-processes tweets by removing stopwords, stemming the remaining words, and replacing account handles with display names: from *@NicholasMamo* to *Nicholas Mamo*. SEER also removes Unicode symbols and normalises characters that repeat more than twice, simplifying the impassioned *gooooaaal* into *goal*. Finally, the technique weighs words using the TF-ICF scheme from the understanding period and normalises documents. The more meaningful changes occur later, when the topical streams eliminate the costly clustering technique.

Clustering has its appeals, like how documents describe topics better than keywords [5]. Clustering's critical appeal, however, lies in its ability to overcome the event shadow: a goal does not interfere with a yellow card because the two topics exist in different clusters. In SEER, streams achieve the same effect without the resource-intensive and inefficient clustering process: a goal does not interfere with a yellow card because the two topics exist in different streams, not clusters. Even if two similar topics, such as

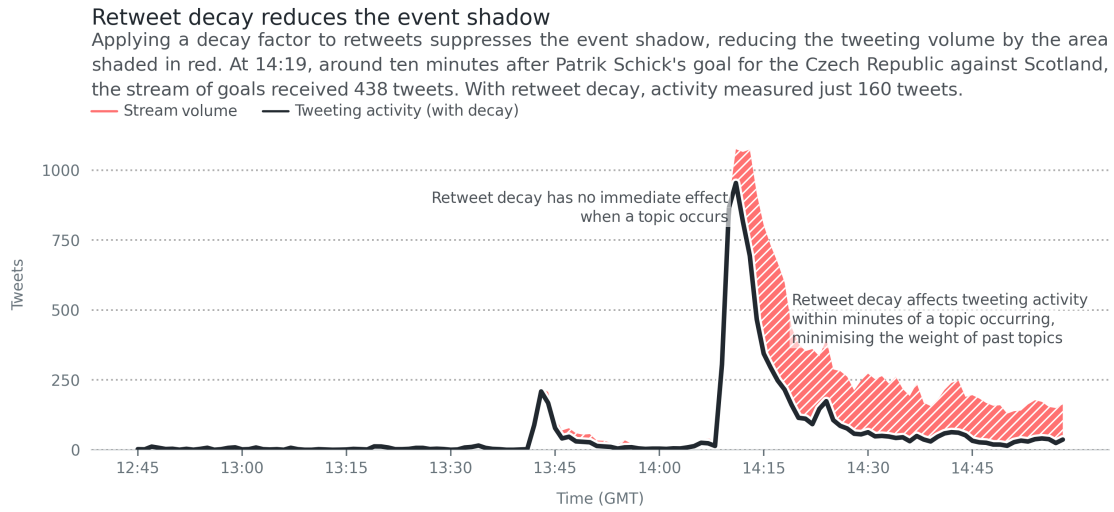

Figure 5.5: Retweet decay minimises the effect of the event shadow by giving a lower weight to old retweets. Visually, the decay makes the spikes in volume better-defined, thus reducing the influence of past topics on the algorithm's perception of the present.

two goals, find themselves competing for attention, it is unlikely for one to dwarf the other like a key topic dwarves a non-key topic: similar topics attract similar attention. SEER may thus retain only ELD's feature-pivot technique.

The reduction of ELD's combination of TDT techniques to a lone feature-pivot algorithm still requires some changes. In ELD's combination, the minimum cluster size acted as an indirect control over tweeting activity: if Twitter users did not discuss a topic intensely enough, they would not form sufficiently-large clusters. Without any control over tweeting activity, ELD's feature-pivot technique could mistake even incidental shifts in the vocabulary of quiet streams as topics. Therefore SEER borrows two assumptions from TDT research on how users react to topics: when a topic occurs, users tweet more, and with a different, specific vocabulary [161; 162].

In SEER, we test the first assumption, the increase in tweeting activity, through volume. We segment the stream into time windows and count the tweets published in a time window, D_s. While we do not remove retweets, we do not want what happened in the past to warp our perception of the present either. Therefore unlike other TDT research, we apply a decay factor to retweets: the more time passes before someone retweets a tweet, the less its influence on the tweeting activity. The decay factor uses the exponential distribution with λ empirically-set to 0.5 to rapidly reduce the weight of retweets. In Equation 5.2, $time_d$ and $time_{d'}$ represent the UNIX timestamps, in seconds, of the original tweet and the retweet.

$$decay_{d'} = \begin{cases} e^{-\lambda \frac{time_{d'} - time_d}{60}} & \text{if } d' \text{ is a retweet} \\ 1 & \text{otherwise} \end{cases} \tag{5.2}$$

$$activity_s = \sum_{d \in D_s} decay_d \tag{5.3}$$

The decay factor suppresses the event shadow. It makes peaks in volume visibly clearer and better-defined. A one-minute delay costs a retweet around 40% of its weight and lightens the heavy tail of the event shadow. In Figure 5.5, decay lightens the event shadow cast by Patrik Schick's goal from the halfway line in the Czech Republic's match against Scotland [170]. It limits Twitter's hubbub to a handful of minutes, making the wonder goal seem more ordinary than it actually was. Five minutes after the goal, the match had moved on, and so had SEER.

We consider that a stream could contain a topic when tweeting activity, after applying the decay factor, rises sharply. Therefore before deploying the feature-pivot algorithm to detect a topic, as we describe further down, SEER looks for a burst in tweeting activity, controlled by two thresholds: a dynamic threshold and a static baseline.

The dynamic threshold adapts to each stream's varying levels of activity. While isolated in separate streams, key and non-key topics still attract different levels of attention. The threshold for what constitutes a sufficiently sharp rise in tweeting activity should be higher in the agitated stream of goals than in the quieter stream of yellow cards. In SEER, the dynamic threshold of a stream simply calculates the arithmetic mean of the activity in all previous time windows:

$$dynamic_{s_n} = \frac{1}{n-1} \sum_{i=1}^{n-1} activity_{s_i} \tag{5.4}$$

We experimented with other types of dynamic thresholds too. As expected, a stricter threshold like the one Hsieh et al. [96] used, one or two standard deviations above the arithmetic mean, improved precision at the expense of recall. We found the trade-off unnecessary. A stricter threshold needlessly micro-manages the actual TDT algorithm, ELD's robust feature-pivot technique, and wastes the merits of SEER's clean and topical streams. Nevertheless, retaining the more lenient arithmetic mean requires the introduction of a second, static threshold.

The static threshold establishes the minimum activity in a time window to consider the presence of a topic plausible. While the dynamic threshold adapts well to voluminous streams, even a slight, anomalous increase in quieter streams could be mistaken

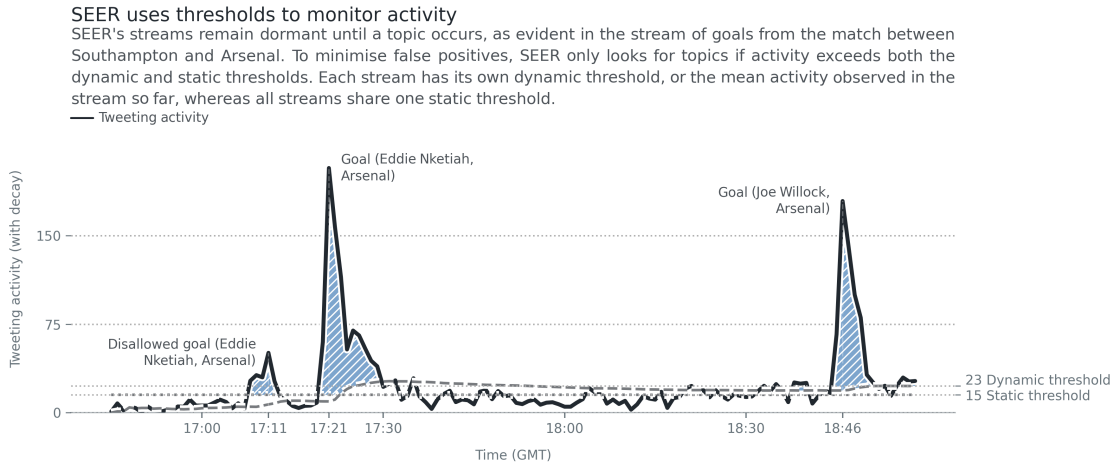

Figure 5.6: SEER's dynamic and static thresholds monitor tweeting activity. The dynamic threshold adapts to the varying levels of activity in each stream, whereas the static threshold remains fixed. The TDT algorithm only checks for topics if activity exceeds both thresholds.

for a topic. The static threshold, a parameter applied equally to all streams but targeted specifically at low-volume streams, sets a minimum activity below which SEER never checks for topics. If the stream passes the first test, its activity climbing above both the static and dynamic thresholds, as shown in Figure 5.6, SEER verifies whether a topic has occurred by deploying the feature-pivot technique.

We test the second assumption, whether the discourse has shifted, with ELD's burst-based measure. ELD's feature-pivot technique changes little in SEER. To accommodate clustering's inefficient processing, ELD placed fixed checkpoints at regular intervals to observe changes in vocabulary. By eliminating clustering, however, SEER may reintroduce the more responsive sliding time windows. Analogously, instead of comparing the popularity of keywords in a cluster with their popularity in past checkpoints, SEER compares the popularity of keywords in the current window with previous sliding time windows.

We also tweak ELD's measure of a word's popularity, or nutrition. In the original sense imagined by Cataldi et al. [32], nutrition combined the usage of a word with the reputation of those who used it. ELD, however, considered all users as equal sensors and calculated the nutrition of a word t based only on its usage: the sum of the word's TF-ICF term weights in the normalised tweets published during the time window, D_s. In SEER, we adopt the same simplified interpretation as ELD, but we add the decay factor for retweets from Equation 5.2:

$$nutrition_{t,s_n} = \sum_{d \in D_s} TF\text{-}ICF_{t,d} \cdot decay_d \tag{5.5}$$

Complementing nutrition is burst. While nutrition measures the instantaneous popularity of a word, burst measures the change in its popularity. Intuitively, a bursty word appears more prominently in the present than in the past. ELD's measure of burst monitors the shift in the event's vocabulary, one word at a time, to detect topical keywords and, by extension, topics.

SEER re-uses ELD's process to measure burst [146]. Before SEER calculates burst, it rescales the nutrition of each sliding time window between 0 and 1, corresponding to the words with the lowest and highest nutrition values. Then, it computes the burst of each word t by comparing its nutrition in time window n, s_n, with its nutrition in previous time windows:

$$burst_{t,s_n} = \frac{\sum_{i=n-\tau}^{n-1} (nutrition_{t,s_n} - nutrition_{t,s_i}) \cdot \frac{1}{\sqrt{e^{n-i}}}}{\sum_{i=1}^{\tau} \frac{1}{\sqrt{e^i}}} \tag{5.6}$$

The first component in the numerator fulfils burst's fundamental role, to compare the nutrition in the present with the nutrition in the past. When we originally designed the formula, however, we intended for burst to have two additional properties [146]. First, we wanted burst to give more weight to recent observations. The second component in the numerator applies a decay factor, controlled by the parameter τ, to past time windows. Incidentally, the decay factor makes old time windows redundant, so we empirically set τ to five.

Second, we wanted burst to be explainable. The denominator normalises burst and, together with the earlier rescaling of the nutrition scores, binds the final value between -1 and 1. At one extreme, a burst of -1 signals a previously-topical word on the decline; at the other extreme, a burst of 1 signals a newly-topical word.

Burst reveals how Twitter users discuss topics and explains better the event shadow. Figure 5.7 shows how the burst of Joe *Willock* evolved over a few minutes after he scored for Arsenal against Southampton. Burst's behaviour mirrors findings on how bursty events progress through four stages in sociological research: a quiet period before the word becomes topical, a rapid increase in popularity, which then settles on a longer plateau, and finally the paracme, or the word's gradual return to normality [128].

Burst's four stages are important because they reveal the predictable pattern of topics and their topical keywords. First, we can expect to capture topical keywords quickly. After Joe Willock's goal, his burst in the stream of goals rose from 0 to above 0.7 within just 17 seconds of his first mention. We can confidently set a relatively-high threshold

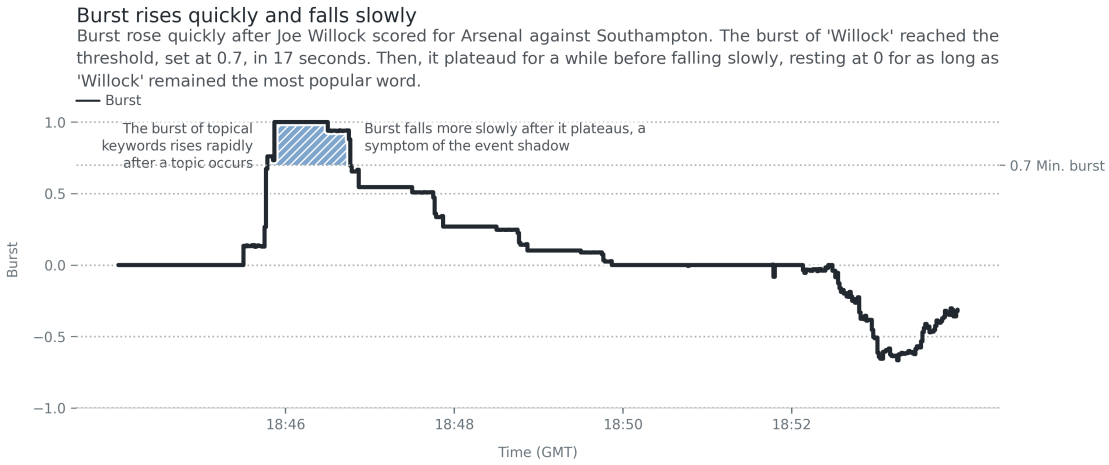

Figure 5.7: Topical keywords become bursty almost immediately after a topic occurs, but they persist longer in the event's vocabulary. SEER accepts words as topical for as long as their burst remains above the threshold, a user-defined parameter.

for burst without harming responsiveness. SEER accepts words as topical—and thus, accepts that a topic really has occurred—for as long as burst remains above the user-defined threshold.

Second, we can expect topical keywords to remain bursting for longer. The long bursts of topical keywords, often outlasting the sliding time window itself, allow SEER to describe topics. In our previous works, the clustering technique's ability to group descriptive tweets in bursty clusters dictated the algorithm's expressiveness. Without clustering, SEER could rely on bursty keywords to describe topics, but it would submit to the limits of keywords, inexpressive in isolation [5]. Instead, SEER describes topics by collecting every tweet that mentions a bursty keyword, starting from the beginning of the time window and stopping when burst dips below the threshold.

Finally, SEER organises the entirety of its output—topics, topical keywords, and tweets—in timelines, one for every stream. A timeline serves the purposes of a chronological ordering of topics and an intuitive representation of events. Above all, a timeline's structure helps SEER serve the purpose of a topic tracker.

SEER tracks topics like ELD [146]. If a new topic bursts within 90 seconds of another in the same stream, SEER adds the new topic to the existing node. Otherwise, it compares the new topical keyword with the keywords of other nodes from the previous 10 minutes; if the cosine similarity exceeds 0.6, SEER again adds the new topic to the existing node. If no similar node exists, SEER appends a new node to the stream's timeline.

The option to collapse the stream timelines into one traditional event timeline re-

mains. In fact, in the next section we assume that if a topic overlaps several streams, such as a *goal* disallowed for an *offside*, it bursts almost simultaneously in each stream. The assumption allows the monolithic timeline to combine topics that occur within 90 seconds of each other.

Understanding transforms ELD in simple ways. Understanding removes the need for clustering, so SEER gains in responsiveness, scalability and sensitivity, as we show in the next section and in Appendix B. Of course, the algorithm requires the understanding of What may happen in an event as an input, which in practice must involve a certain degree of manual supervision. Yet without the document-pivot technique, SEER becomes simpler, more efficient and elegant.

Apart from the understanding, only three user-defined parameters control SEER's performance. The feature-pivot technique adapts to events and to event domains with the length of the sliding time windows, the static threshold and the minimum burst. In practice, SEER could be simplified even further. Of the three parameters, the length of the sliding time windows does not change across events; it only changes across event domains. A stricter dynamic threshold can also make redundant the static one, as we show Chapter 6. In the end, SEER only sacrifices portability.

Buntain et al. [28] had argued on behalf of exceptional topics. An understanding-driven TDT technique could not track the exceptional, such as when health officials invaded the pitch during a match between Brazil and Argentina to deport players who had broken COVID-19 quarantine laws [262]. Buntain et al. [28], however, over-stated the problem. Topics in the same domain share a vocabulary [161; 162]—even outlandish topics: health officials invaded the *pitch* during a *match* between Brazil and Argentina to deport *players* who had broken COVID-19 quarantine laws. In practice, domain understanding only restricts SEER's portability to a domain's events and topics, exceptional or not, as we show in Chapter 6.

In the rest of this chapter, we quantify understanding's contributions to TDT. In the next section, with football in the backdrop, we demonstrate how good understanding, even applied trivially in ELD, can benefit event tracking in limited ways. Later, when we experiment with SEER, we demonstrate how proper understanding handed control to drive TDT algorithms can provide all-encompassing benefits.

5.3 | The benefits of understanding

Football is a simple game, and still, it manages to surprise. Research found in football a predictable domain with a vocal following, comprehensive media coverage and frequent events [28], but even in the predictable, research found the unpredictable. From routine matches to surprise routs, TDT research could trial its algorithms in different scenarios without ever leaving the domain. Unsurprisingly, few domains appear with the consistency of football matches in literature: 18 out of 79 studies in our review in Appendix A. Therefore in this chapter, we too test understanding and its application in the whimsical setting of football matches.

In this section, we perform a manual evaluation following the principles we outline in Appendix A. Evidently, the choice of datasets can sway results; Keogh et al. [109] called the corpora a "meta parameter" of the evaluation. We minimise our bias somewhat counter-intuitively, by curating events to cover various scenarios. Moreover, we analyse six datasets, a high figure in the standards of manual evaluations in TDT literature. We describe the six events, including their defining characteristics, and the number and types of topics in Appendix D.3.

In our evaluation, understanding itself takes a withdrawn role, leaving us to evaluate its contributions to TDT. Naturally, we compare SEER with its predecessor without understanding, ELD, which we designed to optimise precision and recall. Even then, however, we only study the contributions of understanding indirectly, through a summary-based evaluation; as we explain in Appendix A, manual costs render more direct evaluations infeasible. Therefore like the majority of TDT literature, we manually annotate summaries as a way of evaluating the topic tracking performance.

We generate summaries using the Maximal Marginal Relevance (MMR) method [84]. MMR, an extractive technique, constructs summaries from individual documents or sentences, always seeking to balance relevance with non-redundancy. We generate summaries from the 20-longest tweets in each timeline node, thus preserving context and coherency. The topical keywords and their bursts serve as the relevance query. Summaries may include several tweets but may not exceed 280 characters, the length of the longest possible tweet. Unless otherwise stated, we merge SEER's stream timelines into one timeline to simplify our evaluation.

Our annotations of the summaries deviate from literature. As we argue in Appendix A, TDT's metrics, in particular precision, cannot express the range of Twitter's content. Therefore in our analyses, we follow a novel evaluation methodology, adapted and refined from existing literature. While our analyses retain a notion of precision, as we describe further down, we divide the metric into five labels:

- Noisy topics do not describe a real occurrence, or describe it with insufficient clarity. Noise represents a catch-all label for spam, fake news and off-topic tweets, but noisy topics may also lack adequate context to understand what happened. The tweet "Nathan Patterson wouldn't have got [sic] in his way" does not tell us what Patterson did, nor in whose way he got himself.

- Redundant topics describe newsworthy but old occurrences without contributing any novel information. Most redundant topics would already have been detected by the algorithm in the past, but they endured long enough in Twitter's discussion for the TDT algorithm to capture them several times. In fact, redundant topics tend to be key topics with a long event shadow, like Patrik Schick's goal.

- Subjective topics describe an opinion or a desire without offering any justification. A subjective topic provides the reader with too little information to verify a claim. The tweet "[Hector] Bellerin is so so poor" may be a true observation, possibly shared by many users, but without context, it may also be the tinted perception of one Twitter user.

- Non-enumerable topics describe real but difficult-to-enumerate occurrences. A non-enumerable topic might occur too often or have too little influence to be newsworthy, but we cannot deny that it occurred. Among others, non-enumerable topics include injuries, missed chances and more general observations, as in the tweet "Southampton with six shots to Arsenal's one in the second half."

- Enumerable topics describe real, easy-to-enumerate occurrences with considerable newsworthiness. In this work, like in our previous research [146], we consider goals (including disallowed ones), yellow and red cards, the start and end of each half, and substitutions.

We obtained the ground truth for non-enumerable and enumerable topics from reliable sources: The Guardian, LiveScore.com, BeSoccer.com and others.

In our analyses, these five labels depict the algorithms' behaviours, but they also form the basis of IR's traditional metrics: precision, recall and the F-score. We interpret precision with a strictness uncharacteristic of TDT evaluations. Since opinions and desires have a questionable value of newsworthiness, we regard subjective topics as imprecise. Furthermore, as we argue in Appendix A, TDT research should reject redundant topics with the same determination with which it rejects noisy topics. Therefore in this work we only consider enumerable and non-enumerable topics as precise.

Conversely, we only calculate the recall of enumerable topics: goals, cards, halves and substitutions. We provide a full break-down of topics in Appendix D.3.

In the rest of this section, we present and study two applications of understanding. First, we disprove the idea that understanding must drive a TDT algorithm to be effective by showing how even trivial applications can improve precision. Second, we exhibit the benefits of understanding-driven applications to recall in SEER. We present another analysis on the benefits of understanding to sensitivity in Appendix B.

The benefits of understanding to precision

Early TDT experiments did not discover the limits of applied understanding but the limits of linguistic understanding. When researchers followed Allan et al. [9] into exploring limited forms of understanding, the mitigated successes failed to emphasise the potential of event knowledge. Instead, the experiments had the opposite effect. They seemed to confirm the initial fears about understanding and its applications. And so TDT research moved on and away. Few seemed to realise that the gains had not been limited by flawed application but by flawed understanding.

In our first application, we apply event domain understanding trivially. We disregard momentarily SEER to demonstrate how domain understanding does not have to drive TDT methods to be beneficial. In fact, our application of understanding is perhaps the most conceivably trivial: we remove all tweets without domain terms.

Our understanding, like its application, assumes a simple form: the top 70 domain terms from EVATE after bootstrapping. The cut-off point includes many topical terms but excludes the deluge of general, often inaccurate terms that follow, such as *club*, *good* and *poor*. Table 5.3 on page 137 shows the 70 terms split into streams. In our experiment, we fed tweets containing any of the 70 terms to an unchanged ELD, which we refer to as $\text{ELD}_{\text{Filtered}}$, and discarded the rest. To study the benefits of understanding applied trivially, we established ELD's configurations empirically, as we describe in Appendix E.1, and then fixed the same parameters in $\text{ELD}_{\text{Filtered}}$.

The trivial application of understanding, although a passive presence, simplified $\text{ELD}_{\text{Filtered}}$. At a distance, understanding could not make $\text{ELD}_{\text{Filtered}}$'s processing any more efficient, but domain filtering enabled by understanding simplified the task, as shown in Table 5.1. In the match between Southampton and Arsenal, understanding retained 41,207 of 97,874 tweets—42.10% of the data. Even at its most indulgent, in the match between Scotland and the Czech Republic, understanding only kept 51.15% of all tweets. In the end, after $\text{ELD}_{\text{Filtered}}$ filtered tweets with its own internal rules,

Algorithm	Dataset size (tweets)	Domain filtering (%)	Algorithm filtering (%)
ELD	124,479	124,479.00 (100.00%)	100,856.33 (80.37%)
$ELD_{Filtered}$	124,479	59,131.67 (46.97%)	46,909.50 (36.93%)
SEER	124,479	82,632.67 (65.19%)	35,073.67 (26.84%)

Table 5.1: $ELD_{Filtered}$ and SEER filter aggressively but sensibly. The table reports the macro-average number of tweets that pass each stage of filtering across our six datasets. Domain filtering refers to the understanding-based filtering or streaming, whereas the algorithm filtering refers to each technique's specific filters. SEER's figures include duplicate tweets.[4] We present a full breakdown of the dataset filtering in Table D.12.

only between 33.03% and 42.51% of the datasets remained. The trivial application of understanding made $ELD_{Filtered}$ more responsive and scalable.

The perfect understanding, however, is not defined by the aggressiveness of its filters but by the sensibility of its nature. The perfect understanding removes all the irrelevant tweets and retains all the relevant ones, and our understanding removed many of the former. As shown in Table 5.2, $ELD_{Filtered}$ captured, on average, 15 fewer topics than ELD (37.83 ↓ 22.83) but only 4 fewer precise topics per match (20.33 ↓ 16.33). Understanding reduced noisy topics in $ELD_{Filtered}$ to a third of ELD's (19.82% ↓ 6.57%) and subjective topics to almost a half (20.70% ↓ 12.41%). Filtering worked sensibly, judging tweets not on assumed markers of noise but on subject.

Because filtering worked sensibly, $ELD_{Filtered}$ became more precise than ELD. Precision climbed drastically, by almost 18% (53.74% ↑ 71.53%). If we had to consider as precise the newsworthy but redundant topics, signs of flaws in topic tracking, not in understanding, $ELD_{Filtered}$'s precision would climb even higher, to 81.02%, up from ELD's 59.47%. Evidently, the trivial application of understanding made $ELD_{Filtered}$ significantly more precise than ELD (one-tailed paired samples t-test: $p = 0.005$).

Our understanding, however, was not perfect. Filtering removed some relevant tweets too and the statistically-significant gains in $ELD_{Filtered}$'s precision were matched with statistically-significant drops in recall (56.73% ↓ 42.31%; one-tailed paired samples t-test: $p = 0.0274$). Only twice did $ELD_{Filtered}$ recall at least half of the ground truth topics. In the match between Scotland and the Czech Republic, the most testing trial, $ELD_{Filtered}$ only recalled Patrik Schick's two goals; the drop in recall in this match almost single-handedly pushed $ELD_{Filtered}$'s average F-score below ELD's (55.04% ↓ 52.09%).

Sceptically, you might wonder whether the conclusion was predictable. Was it inevitable that when $ELD_{Filtered}$'s clusters became smaller, like its datasets, the algorithm would succumb to IR's classical trade-off between precision and recall? The question has its merits; after all, filtering reduced dataset sizes by between 50% and 60%. As if to

Algorithm	Topics	Precise topics	Precision	Recall	F-score
ELD	37.83	20.33	53.74%	56.73%	55.04%
$ELD_{Filtered}$	▼ 22.83	▼ 16.33	▲ 71.53%	▽ 42.31%	52.09%
SEER	33.83	24.17	△ 71.43%	55.77%	△ 62.89%

(a) SEER improved precision without sacrificing recall, unlike $ELD_{Filtered}$. The table reports the macro-average number of topics and F-score, and the micro-average precision and recall. We present a full breakdown of the results in Table F.5.

Algorithm	Redundant	Noise	Subjective	Non-enumerable	Enumerable
ELD	5.73%	19.82%	20.70%	30.84%	22.91%
$ELD_{Filtered}$	9.49%	6.57%	▽ 12.41%	△ 43.07%	28.47%
SEER	4.43%	▼ 6.90%	17.24%	▲ 45.81%	25.62%

(b) SEER captured little noise, which it re-invested into capturing non-enumerable topics. In our interpretation, precision includes only non-enumerable and enumerable topics. We present a full breakdown of the annotations in Table F.6.

Algorithm	Goals	Cards	Halves	Substitutions
ELD	87.50%	52.94%	37.50%	57.45%
$ELD_{Filtered}$	100.00%	52.94%	29.17%	▽ 25.53%
SEER	93.75%	52.94%	50.00%	46.81%

(c) Our three algorithms struggled to capture cards, halves and substitutions due to Twitter's behavioural challenges. The table reports the micro-average recall for each type of enumerable topic. We present a full breakdown of the results in Table F.7.

Table 5.2: The understanding-driven SEER improved over traditional methods. ELD struggled with both precision and recall, whereas $ELD_{Filtered}$ traded recall for precision. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to ELD.

further prove the trade-off, recall did not drop in the match between Leicester City and Manchester United, where even after filtering, $ELD_{Filtered}$ retained a large dataset with more than 88,000 tweets. The real answer, however, is more nuanced.

There is a lot to be said about the trade-off between precision and recall. For now, we note only that if the dataset size creates the trade-off, it must create it with a measure of consistency. We observed no trade-off in the recall of key topics, namely the goals and the two red cards. Nevertheless, while $ELD_{Filtered}$ captured just as many cards as ELD, it struggled immensely to recall the other non-key topics: halves and, especially, substitutions. In fact, $ELD_{Filtered}$ captured less than half of the substitutions that ELD did (57.45% ↓ 25.53%). Had we excluded halves and substitutions, $ELD_{Filtered}$'s recall of

goals and cards would have overtaken ELD's (69.70% ↑ 75.76%).[3] Clearly, the trade-off affects substitutions disproportionately.

The trade-off, then, must exist in something subtler than the dataset size, such as the understanding that supports its filtering. EVATE understood cards and substitutions differently. It gleaned an exceptional understanding of the former, not only with foundational terms—*book*, *yellow*, and *card*—but also with supporting terms, like *foul*, *referee* and *tackle*. Conversely, substitutions presented a more difficult challenge to EVATE, which it described with far less clarity; only *sub*, short for substitution, explicitly refers to the topic. We elaborate on the trade-off in the next analysis.

The trade-off shows the quality of understanding limiting our application, and not just the application limiting the understanding. The remark evokes our argument in Chapter 2, about how linguistic understanding limited the application by providing a different kind of understanding than what machines required. Our argument here differs on one point: domain understanding provided $\text{ELD}_{\text{Filtered}}$ the right kind of understanding to detect certain topics, like yellow cards, but its perfectible quality failed with substitutions. Still, even the imperfections of understanding applied trivially simplified $\text{ELD}_{\text{Filtered}}$'s task and improved precision at no cost to the recall of goals and cards.

The benefits of understanding to recall

In reality, the application limits the understanding as well. As datasets grew smaller, $\text{ELD}_{\text{Filtered}}$'s clusters grew smaller too, and the three-tweet threshold for clusters transformed into an aggressive filter that limited sensitivity and the utility of understanding. In this section, we experiment with SEER, our novel algorithm designed to harness understanding. We fixed SEER's sliding time windows to one minute, and varied the static threshold and the minimum burst as we describe in Appendix E.1.

SEER, with 70 domain terms grouped into 15 concepts, did not eschew $\text{ELD}_{\text{Filtered}}$'s aggressive filtering. On the contrary, it filtered even more aggressively. Initially, the streams received between 59% and 74% of the dataset volume, but after filtering tweets, SEER only processed between 22.14% and 35.41% of the dataset volume.[4] Nevertheless,

[3] Curiously, substitutions condition recall more than any other type of topic in our experiments, even compared to other TDT research. During the COVID-19 pandemic, football associations reacted to the physical toll of the congested calendars by increasing the number of permitted substitutions from three to five. As a result, substitutions make up almost half of our ground truth topics: 47 out of 104.

[4] We distinguish between tweets and dataset volume. If $\text{ELD}_{\text{Filtered}}$ consumes 50% of tweets, it would process 50% of tweets once, but SEER may process the same tweet in multiple streams. The dataset volume is simply the sum of tweets processed by each stream as a fraction of all tweets in the dataset. If SEER processes 50% of the dataset volume, the 50% could mean 25% of unique tweets processed by two streams.

while SEER captured, on average, four fewer topics than ELD (37.83 ↓ 33.83), it also captured four more precise topics (20.33 ↑ 24.17).

Consequently, SEER retained $\text{ELD}_{\text{Filtered}}$'s precision and significantly improved over ELD's (53.74% ↑ 71.43%; Wilcoxon Signed-Rank test: $p = 0.0310$). SEER's worst precision (66.67%), in the match between Turkey and Italy, exceeded ELD's best performance (65.52%), in the match between Wales and Switzerland. Like $\text{ELD}_{\text{Filtered}}$, SEER significantly increased the proportion of enumerable topics over ELD (30.84% ↑ 45.81%) and significantly lowered the proportion of noisy topics (19.82% ↓ 6.90%).

SEER, however, differed from $\text{ELD}_{\text{Filtered}}$. Although it applied the same understanding as $\text{ELD}_{\text{Filtered}}$, SEER applied knowledge better, more cleverly. Our new algorithm detected more halves than ELD (37.50% ↑ 50.00%) and just as many cards. On the singular occasion when SEER missed a goal, its topical keywords—*Embolo*, *goal*, *corner* and *lead*—required no description; only the summarisation algorithm failed to communicate the idea clearly. Ultimately, recall dropped minimally from ELD—too little to be statistically-significant (56.73% ↓ 55.77%; one-tailed paired samples t-test: $p = 0.3355$). In short, SEER did not trade precision for recall.

Still, SEER exhibited its limits. Neither our understanding nor its application could overcome Twitter's behavioural challenges. TDT approaches will probably always be limited by what triggers Twitter, whose interests hang over recall like an invisible ceiling [146]. SEER could only yield to Twitter's biases. Our algorithm routinely struggled to detect topics related to the less popular team. In the match between Hungary and France, it captured all of France's substitutions but none of Hungary's. The same trend persisted in all other matches, albeit never with such pronounced effects.

SEER struggled with substitutions too, although not to the same extent as $\text{ELD}_{\text{Filtered}}$. With understanding, SEER could eliminate clustering and become more sensitive, as we demonstrate in Appendix B, and the increased sensitivity compensated for the lack of knowledge about substitutions. However, while SEER improved the recall of substitutions relative to $\text{ELD}_{\text{Filtered}}$ (25.53% ↑ 46.81%), it still did not reach ELD's levels (57.45% ↓ 46.81%). Moreover, SEER often detected substitutions indirectly, such as through tweets that discussed the changes, or when the substitution accompanied another topic, as in the tweet "Shane Long coming on [in the] *second half*."

Two factors logically reduced $\text{ELD}_{\text{Filtered}}$'s and SEER's recall of substitutions, but neither fully explain the difficulties nor suggest solutions. First, substitutions draw little attention, but so do yellow cards, and both $\text{ELD}_{\text{Filtered}}$ and SEER recalled more cards than substitutions. Second, and as we explained earlier, our incomplete understanding limits our applications, but we cannot expect domain terms to describe participant-centric topics like substitutions. The football community reports about substitutions using many

common words with no particular attachment to the domain: the coach may *change*, *replace*, *bring on* or *take off* a player, who would *come off* or *leave* the pitch for another to *come on* or *enter*, or a player could simply be *off* or *on*.

In this work, we hypothesise a third explanation for our mishandling of substitutions. Throughout our work, we observed researchers attempting to define events but never topics. Here, we separate topics into three broad classes. First, mixed topics, the majority class, involve a participant performing an action: a participant, Scotland or Patrik Schick, scores a *goal*. Second, event-centric topics describe an action and advance the state of the event without directly affecting participants: the *half* starts or ends. Third, participant-centric topics express an opinion about or advance the state of participants with little immediate effect on the event: one player replaces another.

Our hypothesis would explain why EVATE could not describe substitutions comprehensively and why SEER often recalled substitutions indirectly. Domain terms describe actions and changes, the What, but ignore altogether participants, the Who and the Where. Therefore our filtering and streaming captured mixed and event-centric topics but thwarted the detection of participant-centric developments. Perhaps, then, the participants streams by Huang et al. [101], and McMinn and Jose [158] serve a different purpose than topical streams: they express a different, participant-centric version of events. While we leave further study of our hypothesis for future work, we study how understanding hinders its application in more depth in Chapter 6.

The analysis so far describes SEER's performance as an application but reveals little about the influence that different types of understanding have on TDT algorithms. To evaluate, indirectly, the effects of diverse forms of understanding, we followed the same procedure as before, but we annotated each stream's timeline separately. We present a summary of the results in Table 5.3.

Most streams behaved predictably, as we intuited in Section 5.2: topical domain terms generated topics. Streams whose terms describe enumerable topics tended to capture enumerable topics, and streams whose terms describe non-enumerable topics tended to capture non-enumerable topics. We annotated as enumerable 58.82% of the topics generated by the stream tracking *yellow*, *card* and other terms, and as non-enumerable 70.59% of the topics generated by the stream tracking *touch*, *cross*, *ball* and *pass*. The trend persisted in other types of topics too: 44.44% of topics in the self-explanatory stream tracking *world*, *class* and *striker* expressed opinions.

Most streams behaved predictably, but the few exceptions to the intuition revealed SEER's strengths. In particular, the provenance of noise highlights the virtues of SEER's

Stream	Topics	Precise topics	Precision
champion, final, league, football, win	15.33	7.67	50.00%
take, knee, player	11.83	6.17	52.11%
touch, cross, ball, pass	11.33	10.33	91.18%
goal, score, concede, equalise, offside, assist	10.00	7.50	75.00%
need, half, sub, second, lead, 2nd	10.00	7.00	70.00%
keeper, best, goalkeeper, defend, ~~Kepa~~, save	6.83	4.50	65.85%
foul, referee, book, decision, VAR, given, pen, dive, ref, penalty	5.00	3.50	70.00%
gol, stream, online, free, Reddit, link, ~~Manchester~~, FFS, live	4.67	4.00	85.71%
deflect, kick, corner, shot, net	4.50	4.00	88.89%
world, class, striker	3.00	1.00	33.33%
tackle, dribble, yellow, red, card	2.83	2.50	88.24%
~~man~~, utd	1.17	1.00	85.71%
hit, post	0.67	0.67	100.00%

(a) The more a stream resembled linguistic understanding, the worse it performed. The table reports the macro-average number of topics and the micro-average precision of each stream. We present a full breakdown of the results in Table F.8.

Stream	Redundant	Noise	Subjective	Non-enumerable	Enumerable
champion, final, league, football, win	6.52%	14.13%	29.35%	23.91%	26.09%
take, knee, player	4.23%	11.27%	32.39%	22.54%	29.58%
touch, cross, ball, pass	1.47%	7.35%	0.00%	70.59%	20.59%
goal, score, concede, equalise $_{+2 \text{ terms}}$	11.67%	0.00%	13.33%	43.33%	31.67%
need, half, sub, second, lead, 2nd	11.67%	1.67%	16.67%	20.00%	50.00%
keeper, best, goalkeeper, defend $_{+2 \text{ terms}}$	4.88%	9.76%	19.51%	43.90%	21.95%
foul, referee, book, decision $_{+6 \text{ terms}}$	6.67%	10.00%	13.33%	46.67%	23.33%
gol, stream, online, free, Reddit $_{+4 \text{ terms}}$	0.00%	7.14%	7.14%	25.00%	60.71%
deflect, kick, corner, shot, net	7.41%	0.00%	3.70%	51.85%	37.04%
world, class, striker	11.11%	11.11%	44.44%	16.67%	16.67%
tackle, dribble, yellow, red, card	0.00%	5.88%	5.88%	29.41%	58.82%
~~man~~, utd	0.00%	0.00%	14.29%	28.57%	57.14%
hit, post	0.00%	0.00%	0.00%	75.00%	25.00%

(b) Topical streams normally captured topics, while non-topical streams generally captured different types of noise. The table reports the micro-average distribution of annotations across each stream. In our interpretation, precision includes only non-enumerable and enumerable topics. We present a full breakdown of the annotations in Table F.9.

Table 5.3: Linguistic understanding injected noise into event timelines. The streams that best reflect semantic understanding tended to out-perform the others. For clarity, we manually lemmatised the terms in this table, and struck out terms that we had used as tracking keywords in Chapter 4, which SEER ignores. The streams tracking *baller* and ~~*Arsenal*~~, and *clear* and *handball* generated no topics.

aggressive filtering and topic detection. Our novel algorithm sought shifts in vocabulary, but noise does not shift and does not burst [101; 234], and so noise rarely emerged as a topic. In fact, at a precision of 85.71%, not even the stream tracking noisy words, like *stream*, *online* and *Reddit*, proved noisy. The question, then, follows naturally: where does noise originate from?

Two factors introduced noise and other subjective content to the detriment of precision. First, a weak, negative correlation exists between precision and the average number of topics in a stream (Pearson correlation coefficient: $r = -0.4023$). The number of topics ranged from 0.67 per match in the stream tracking *hit* and *post* to 15.33 in the stream tracking *football*, *win* and others. Generally, as the average number of topics increased, so did the noise and the subjective content: the precision-recall curve.

The negative correlation indicates a systemic failure in the dynamic threshold. In low-activity streams, SEER's static threshold often silenced topic detection, but in high-activity streams, when tweeting activity rose higher than the static baseline, the dynamic threshold continuously invoked and tested the TDT component. A stricter dynamic threshold, one or two standard deviations above the mean [96], might have suppressed noise and opinions better.

Nevertheless, the negative correlation alone remains too weak to serve as an explanation for the provenance of noise and subjective content. In fact, we found almost no correlation between precision and the average number of precise topics in a stream (Pearson correlation coefficient: $r = -0.0793$). The streams tracking *take*, *knee* and *player*, and *touch*, *cross*, *ball* and *pass* averaged around 11 topics per match. However, whereas we only annotated 52.11% of the former's topics as precise, we accepted 91.18% of the latter's. Therefore for our second explanation, we sought an answer in the nature of our understanding, not in our application.

Second, we noticed that the more noisy or subjective a stream, the more general its subject. General streams contradict our intuition. The terms they track rarely describe topics, What happens during events, and instead characterise the broader domain. In fact, the three least precise streams tracked *world*, *class* and *striker*; *take*, *knee* and *player*; and *league*, *football* and *win*, among others. The second observation also explains the first one. General streams generate more topics than specific streams by virtue of being general: all matches concern *football* and *winning*, but players *hit* the *post* infrequently.

Our analyses say something about the fragility of understanding and its application. The conclusions vindicate our decision to use EVATE's topical terms over TF-ICF's and TF-DCF's general lexicons, or Rank Difference's and Domain Specificity's technical but non-topical lexicons. At the same time, our conclusions explain the struggles of early TDT research. If not even EVATE's general but still football-related terms served SEER,

how could early research hope to overcome complex challenges with simple linguistic understanding?

Our analyses also say something about our interpretation of understanding. Proper event understanding, the kind that TDT methods need, cannot and will not result from classical IR research into linguistics. Proper event understanding will result from TDT literature learning about events independently of classical research, like how DEPICT adapted traditional NER or how EVATE adapted traditional ATE. Better understanding, perhaps even honed by manual curation, reserves promise to improve results further.

The benefits of understanding extend beyond precision and comprehensiveness. They reach into sensitivity, as we show in Appendix B, and into less tangible, hardly-quantifiable qualities: the expressiveness of SEER's topical keywords, the responsiveness of a simplified technique, and the elegance and efficiency of a solution without clustering. What we can quantify, however, attests to understanding's virtues. Twitter does not force TDT's sacrifices in the name of precision. Our algorithms do, and our understanding does.

Recap

In the years since TDT's founding, the research community fixated on accuracy. Perhaps the community felt compelled to remain faithful to its traditional evaluations. Or perhaps the community did not dare to hope for improvements in qualities it could not quantify—not without first making satisfactory gains in accuracy. Now, perhaps the time has arrived for TDT research to move past accuracy.

The research community can move past accuracy without abandoning it. Accuracy can improve alongside other qualities through understanding. Because if even raw, unfiltered understanding driving simple algorithms in noisy domains can provide wide-ranging benefits, what challenges in the TDT community's way could possibly seem insurmountable? In this chapter, we demonstrated understanding's potential by answering the following questions:

- What makes the ideal TDT algorithm? Perfect comprehensiveness and perfect precision no longer make the perfect algorithm. In Section 5.1, we explained how understanding can help researchers approach our vision of the ideal algorithm: a comprehensive, precise, expressive, responsive, scalable but sensitive, lightweight and efficient, parameter-free and portable algorithm.

Principal contributions

- Confirmation of the benefits of semantic understanding to precision, even when event knowledge is applied trivially
- SEER, the first event tracking algorithm driven by understanding and a demonstration of the wide-reaching benefits of semantic knowledge in TDT
- The first study on the effects of different types of understanding on the performance of TDT algorithms

- How can event understanding drive TDT? Unlike the research community's early efforts with linguistic understanding, event domain understanding can assume more than a passive role and direct algorithms. In Section 5.2, we proposed SEER, a reinterpretation of ELD [146] greatly simplified by understanding, the prescience of What can happen.
- In what ways can event understanding improve TDT, and to what extent? Even as a passive presence, proper event understanding—not linguistic understanding—can improve event tracking algorithms. In Section 5.3, we demonstrated how understanding makes TDT algorithms more precise without sacrificing comprehensiveness when research lets event knowledge drive its techniques.

We present another analysis on the benefits of understanding to sensitivity in Appendix B. Nevertheless, the analyses in this chapter alone answer our research question by demonstrating that semantic understanding can improve TDT. Understanding, however, can do more than simply improve TDT. Improvements will beget improvements and understanding will give the research area new purposes. In the next chapter, we look to the future, this time with a case-study in politics in which we explore what forms those purposes could take.

6

Application

The Politics Case Study

In 2015, Reuters embarked on a project that would harness computational power to minimise information overload on journalists and gain a competitive edge: Reuters Tracer [129; 130]. By 2017, Reuters Tracer would be scouring 12 million tweets daily for news. It would cluster them with a TDT algorithm, capturing 70% of the newsroom's own stories with around 60% precision, and raising alerts minutes ahead of Reuters' internal ticker and its competitors. Yet Reuters Tracer would also produce more than 6,600 clusters every day, 275 events every hour, 4.5 stories every minute [130]. Reuters had its competitive edge, a still-overloaded competitive edge.

Reuters were not alone in experimenting with AI. In Japan, JX Press automatically mines social networks for news [149], and in Germany, Deutsche Welle uses Dataminr [47], an AI-powered news discovery product. More and more newsrooms around the world are starting to recognise the value in computational journalism, not least to aid in newsgathering efforts [18; 51; 149; 179; 180; 217]. Like Reuters, will they also realise that even as systems process and reduce data, too much remains to digest manually [165]? Will they realise that it no longer suffices for TDT algorithms to detect and track without explaining?

The scientific community concluded the same. So lacking was event understanding, and so great the need to understand events, that a new research area emerged: Event Modelling and Mining (EMM) [39]. TDT algorithms do not understand events; event modellers understand Who did What, Where and When. TDT algorithms detect and track; event miners infer new information from event models. TDT algorithms return events as outputs; event modellers return events as resources. In this chapter, with politics as our backdrop, we envision a future in which event modelling revolutionises

TDT as we answer the following questions:

- What makes event modelling so complex? The process to model events formally, to make of them resources that we could mine automatically [39], represents an elaborate undertaking. In Section 6.1, we argue that event modelling only grows complex when understanding does not drive it.

- How can understanding make resources of events? The difference between events as outputs and events as resources lies in whether and how we understand them. In Section 6.2, we combine the outputs of DEPICT, EVATE and SEER to create a simple event modeller driven by understanding.

- What do we sacrifice with understanding? Buntain et al. [28] feared that understanding could never transfer across events and event domains. In Section 6.3, we investigate the sacrifices of understanding to portability by transferring our understanding of American politics to British politics.

- What role will event tracking play in the modern newsroom? Event tracking and event modelling, improved and augmented by understanding, can find a practical purpose in journalism. In Section 6.4, we talk with Professor Charlie Beckett, founder and director of the JournalismAI initiative at the London School of Economics (LSE), to explore the use-cases of event tracking in newsrooms.

6.1 | The many names of understanding

You could call Reuters Tracer by many names [129; 130]. You could call it an event extractor for the way it extracts Who does What, Where and When from individual tweets. You could call it an event detector and tracker for the way a TDT algorithm uses those 'four Ws' to cluster tweets into events. Or you could simply call it an event modeller because a cluster's 'four Ws' could form a semantic event model.

The many names represent different problems but also different needs and ends. Event modelling represents the modern need of computational journalism [39; 165]. To a journalist, Reuters Tracer's 6,600 daily events might as well have been 12 million tweets: more tractable, certainly, but still intractable data. Journalists and researchers already have data—too much of it, in fact [39; 165]. What they need are new ways to index, query and mine it automatically [39], but neither would be possible without semantic, machine-readable representations of events: event models.

Thus, event modelling emerged, perhaps not quite as a nascent research area as much as a reformulation of existing tasks. Event extractors already glean structured representations of events, but only from individual documents and often with arbitrary structures; TwiCal extracts entities, phrases, timestamps and event types [218]. In contrast, TDT algorithms identify events from whole corpora, but rarely with any structure at all. Event modelling simply combined principles of event extraction with principles of TDT: it broadened event extraction's task to TDT's corpora and imposed a semantic structure on event models, the 'four Ws' [39].

Inevitably, the three research areas share a close resemblance. In fact, for a short while, and driven by the vision of understanding that Allan et al. [9] first outlined, the TDT task shared a much closer resemblance with event modelling than today. Makkonen et al. [139], like Liu et al. [129, 130] in Reuters Tracer, extracted the 'four Ws' from news articles to build event profiles. A present-day reader might have called the event profiles by another name: event models. Both represent events semantically, in terms of the 'four Ws'. Both understand events.

There, in understanding, lies the difference between merely detecting and modelling. The algorithms of Makkonen et al. [139], Liu et al. [129, 130] and a few others detect, track and model because they understand events, even if only with linguistics; the rest, the greater part of modern research, afford to detect and track without understanding. Only event modellers afford no such luxury: they must understand Who does What, Where and When to build event models. Evidently, event modelling's retrospective understanding arrives far too late to drive TDT algorithms, but it is understanding that separates the two research areas.

Event modelling reveals two shortcomings in TDT literature. First, it exposes the most prominent flaw in TDT algorithms, the absence of understanding. The efforts of Makkonen et al. [139] and Liu et al. [129, 130] remain outliers in a scientific tradition that shunned understanding to the detriment not only of performance but also of expressiveness. Event models loom over TDT literature's raw events as reminders of what Panagiotou et al. [191] wrote about, that algorithms need to detect, track and express events semantically, and as reminders that first, algorithms must understand events.

Second, event modelling exposes the crudity of TDT's outputs. Without a structure, TDT algorithms' events represent an output, an informational dead-end, but in event modelling, events represent resources to mine further, a means to an end [39]. From event models, Gottschalk and Demidova [85] build a knowledge graph to reconstruct timelines of events. From their graph, Opdahl and Tessem [188] discover interesting news angles, and from theirs, Abhishek et al. [1] identify fake news, a problem for which linguistics alone cannot suffice [164]. Like Twitter before it, event modelling gave new

meaning to TDT literature's events.

Nevertheless, event modelling research soon discovered the complexity of understanding. Representing events semantically requires solving problems for which a solution still does not exist [75]. It requires algorithms to detect events, to track stories and to extract semantic information from tweets [136] or from events [90]. In short, it requires researchers to follow the long road to understanding that we have followed in this dissertation.

Complexity harms event modelling. The absence of open-source systems forces projects meaning only to harness event models as a resource to design trivial event modellers [90]. Primarily, they trivialise the understanding. Research resorts, again, to named entities to understand the Who and the Where, and to nouns, verbs and adjectives to understand the What [218]. The understanding becomes simplistic and linguistic, and the errors cascade [269]. Event modellers spend a lot of time understanding, and in the end, they understand poorly.

Complexity also trivialises the TDT algorithms. They appear as if only out of necessity, to deduplicate the event extractor's events, and they appear in simple forms. Feature-pivot techniques measure only changes in volume [252], and document-pivot techniques seldom bear more sophistication. Basic clustering algorithms, often graph-based [20; 56] or incremental [77; 78; 123], dominate. Even Reuters Tracer [129; 130] uses a plain clustering algorithm.

Still, systems grow unwieldy. EMBERS AutoGSR uses three sets of models to represent protests semantically [230]. The hybrid event extractor and event modeller by Petroni et al. [198] includes 14 components to detect seven types of events, and the architectures of Reuters Tracer [129; 130], NewsReader [108; 221; 269] and SUMMA [77; 78; 190] grew so immense that their descriptions span multiple publications. Complexity does not imply ornate design. It implies architectures that extract, detect and model without understanding.

Event modelling is an inherently complex task, but literature let it grow needlessly so. The community neither harnessed its proximity with TDT nor understood events properly. If a TDT algorithm detects, tracks and understands; if, like Reuters Tracer [129; 130], it tells us Who participated and Where or, like SEER, it tells us What happened, then the event tracker becomes an event modeller. The alternative, to model without understanding driving the process, might reserve the same fate in event modelling literature as in TDT research: needless complexity, complex inefficacy. In the next section, we show how SEER and our understanding can simplify event modelling.

6.2 | The understanding-driven event modeller

Event modelling never had a choice, whether to understand or not. Research looked for semantic understanding, but when it found none, it twisted the problem. Event modelling stopped being the problem of representing events semantically and became the problem of generating the understanding with which to represent events semantically. Yet we have semantic understanding. What we have described in this dissertation are the individual parts of an event modeller, the ones that understand Who does What, Where and When. In this section, they form the understanding-driven event modeller.

The understanding-driven event modeller combines DEPICT, EVATE and SEER to model events. In the architecture in Figure 6.1, our modeller receives understanding about Who participates in events and Where from DEPICT, and understanding about What happens and When from SEER. Understanding acts as an input, not as a task, and our modeller simply models. In the rest of this section, we describe our understanding-driven event modeller.

The Who and the Where

The understanding-driven modeller receives understanding about the Who and the Where from DEPICT. DEPICT automatically follows a process of disambiguation similar to Nebhi [177] but more deliberate; it does not disambiguate using only popularity or contextual cues but by understanding the participants themselves. The attribute profiles tell us Who or Where a participant is and what they do, and augments the modelling process.

Consider the matter of distinguishing between the Who and the Where. The Who represents a person or an organisation, and the Where a location, but the attribute profiles help us distinguish between the two more reliably than simple linguistics. When present, an attribute like `BORN` qualifies uniquely a person, the Who. Only participants without a `BORN` attribute require the use of linguistics to distinguish between organisations and locations, the Where. In such cases, we identify the named entity type from the first sentence of the participant's Wikipedia article with NLTK's NER model.

Consider also the matter of matching a participant to a topic within an event. Fundamentally, and similarly to Edouard et al. [56], we simply require that a participant appear in at least half of a topic's tweets to ascribe it the role of the Who or the Where, but the substance lies in what counts as an appearance. Evidently, we search for participant names in tweets, and for the tweets' named entities in participant names, such that the surname *Truss* also matches the participant *Liz Truss*. Nevertheless, understanding

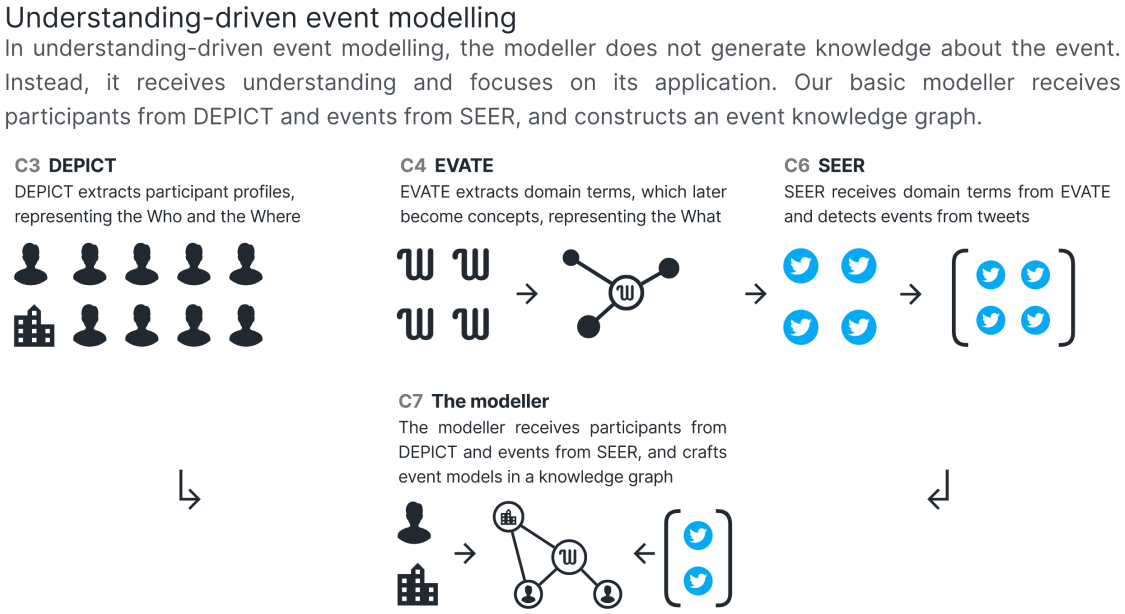

Figure 6.1: DEPICT, EVATE and SEER form a simple event modeller. DEPICT passes the participants, the Who and the Where, directly to the modeller, while EVATE's domain terms, the What, drive SEER and, indirectly, the modeller.

improves our capabilities beyond what essentially amounts to a linguistic process.

We turn, again, to attributes to augment the matching process. Attributes such as `KNOWN-AS` or `REFERRED-TO`, which DEPICT would have extracted in post-processing, list all the nicknames, aliases and lexical variations of participant names. Therefore we also search for participant aliases in tweets, and for the tweets' named entities in participant aliases, such that the colloquialism *Tories* also maps to the United Kingdom's Conservative Party. The event modeller always performs whole-word, case-insensitive searches.

Other heuristics could similarly augment the modelling task further. The attribute `SERVING-AS` could capture indirect references to participants through official titles: *Prime Minister*, *Secretary of State* or *Chancellor of the Exchequer*. Moreover, and although irrelevant to our localised events, attributes could link events with multiple regions, forming a bottom-up taxonomy of the Where: what happens in London also happens in England and in Europe. We leave such heuristics to future work, and as further evidence of the expressive power of semantic understanding.

The What and the When

The understanding-driven modeller receives understanding about the What and the When from SEER. A TDT algorithm that understands can be a TDT algorithm that models, and SEER understands. Like participant timelines express Who participated in an event [101], and like manually-crafted patterns express Who made What happen [136], SEER's streams embody What happens and When. Therefore we model the What and the When directly after SEER's outputs.

In particular, SEER simplifies our modelling of What happens in events. SEER's streams evoke the event types of Edouard et al. [56], Reuters Tracer [129; 130]'s topic models and ASRAEL [222]'s schema labels: coarse subjects. Streams automatically segregate events into concepts—*policies*, *polling* and *protests*, among many others—but without having to infer event types linguistically, train models or cluster schemas. Every event in the stream of *voting* concerns *voting*.

In fact, SEER also augments our modelling of the What, albeit subtly. We could have described What happens with nouns or verbs, or even bursty keywords, but they do not distinguish between the Who and the Where, and the What. Or we could have matched EVATE's terms to SEER's events, like we did with the Who and the Where, but a term could be relevant without being eventful. When a stream flares, however, it reveals something deeper: not merely that a keyword burst or that a term gained relevance, but that something happened within, an event. SEER's streams, EVATE's terms, explicitly capture the actions and changes, the What.

Data alone does not tell a story, and big data obscures it. As Reuters discovered, the events, processed and reduced, remain overwhelming to digest and analyse. To distil information from big data, we need new visualisations, new tools and new forms of storytelling [65; 77; 178; 217]. Fortunately, a definition of knowledge graphs by Ehrlinger and Wöß [57] serves all three:

Definition 8 (**Knowledge graph**). "A knowledge graph acquires and integrates information into an ontology and applies a reasoner to derive new knowledge." — Ehrlinger and Wöß [57]

Ehrlinger and Wöß [57] came up with the definition as they wrestled with literature's scattered interpretations of knowledge graphs. Inadvertently, they also tied the new definition with the problem of EMM. Event modelling's structure of Who does What, Where and When acts as the ontology that gives structure to the knowledge graph, and

event mining's process to deduce the Why and the How acts as the reasoner. Therefore to minimise information overload, we transform our event models into a knowledge graph.

Our knowledge graph represents the different aspects of event models. The nodes represent the Who, the Where and the What, and the relations among them represent interactions. We add frequency-weighted relations among the Who and the Where when two participants co-exist in the same event, and between the What and the Who or the Where when a participant co-occurs with a subject in an event. The knowledge graph thus serves as a visualisation, a tool to discover and tell stories: a summary of happenings and interactions, Who did What and Where across all events.

The knowledge graph concludes the event modelling process as a testament to the benefits of understanding. Like in TDT, understanding brought the benefits of precision, comprehensiveness and simplicity to event modelling. It eliminated the onerous, inexact task of understanding an event from the microcosm of a cluster or a single document and learned, instead, from the entire event and its domain. Like in TDT, understanding only sacrificed portability. We study the sacrifices of understanding next before returning to the knowledge graph in Section 6.4.

6.3 | The sacrifices of understanding

A week is a long time in politics. It is an especially-long time for Liz Truss, once prime minister of the United Kingdom, who could only count on six full weeks during her tenure, the shortest-ever for a UK prime minister to date. Elected on 5 September 2022 and gone less than two months later, Truss' first week in office foreshadowed the turmoil to come. On her first six days in power, we study the sacrifices of understanding to portability.

Our data spans a week, starting one day before Truss won the Conservative Party leadership contest. Over 11 hours on Sunday, 4 September 2022, as Twitter speculated on the next prime minister, we tracked Truss and her adversary, Rishi Sunak; the more than 130,000 tweets constitute our understanding period. Over the next six days, as Twitter dissected and debated the new prime minister's fiery initiation, we continued to track Truss, Sunak and other secondary personalities; the almost three million tweets constitute our event period. We describe our datasets in more detail in Appendix D.4.

Our data alone challenges portability, how well an algorithm adapts to a shifting landscape. The timeline in Figure 6.2 shows the public's engagement changing, but so does the nature of the events. On Monday, 5 September 2022, Truss won the lead-

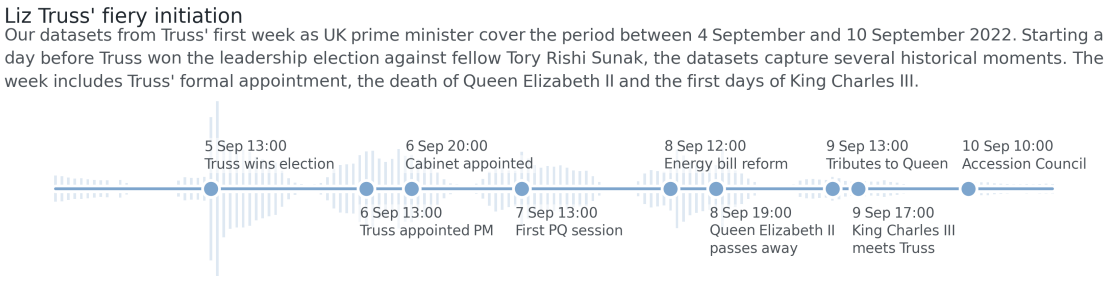

Figure 6.2: A timeline of notable events from Liz Truss' first week as UK prime minister.

ership contest, and on Tuesday, she officially became prime minister. On Wednesday, she fielded parliamentary questions, and on Thursday she unveiled her plan to tackle mounting energy bills. Then, the political agenda transformed. Queen Elizabeth II died, and on Friday and on Saturday, Truss met with King Charles III and attended ceremonial functions. Every day revealed a new side of politics, a new test for our algorithms.

The real test of portability, however, lies in our understanding. We test portability through SEER, driven by EVATE's 250 terms from Section 4.5, but while broadly political, our understanding also has a distinct American flavour. Our understanding of US politics describes a president, our data a monarch and a prime minister; our understanding describes a nationwide election, our data a leadership contest; our understanding describes a congress, our data a parliament. The differences allow us to investigate whether our political understanding causes SEER to miss important topics in another, slightly unfamiliar domain.

We measure the sacrifices of understanding to portability by counting the number of topics that SEER misses from the baseline, ELD. Of course, we know from Section 5.3 that ELD's precision and comprehensiveness suffer in the absence of understanding. In this experiment, however, ELD does not challenge SEER in precision and recall but in portability, and the absence of understanding poses no threats to portability. Later, we also test SEER more thoroughly, against a version of itself without understanding.

The domain requires a few changes to both algorithms. News breaks and spreads differently in politics than in football matches, with more quoted tweets and URLs, both of which we now accept. Discourse changes more slowly too [5], so we combine topics that occur within 15 minutes of each other. Moreover, to reflect the increased stability, we widen SEER's time windows from one minute to fifteen minutes, and ELD's from half a minute to five minutes. ELD's shorter time windows were required both by its resource-intensive tweet buffering, and by the broader subject and increased volatility of its solitary stream.

Finer tweaks to the two algorithms prove more difficult. As we explain later, we could not optimise the F-score. We varied ELD's configurations manually, balancing precision with comprehensiveness, but we could not do the same for SEER. EVATE's 250 terms, grouped into 50 streams, attracted disparate levels of activity, from a mere 383 tweets in the stream of *runoff* and *senate* to 600,225 in the stream of *work*, *want* and other general terms. Every stream would have required its own, tailored configuration. Instead, we let the streams regulate themselves: we lowered the static threshold to a modest 50 tweets per 15 minutes and raised the dynamic threshold to 1 standard deviation above the mean, similarly to Hsieh et al. [96]. We list our configurations in Appendix E.2 and SEER's streams in Table F.14.

The ramifications of a domain as vast as politics forcibly reach our annotation process too. Like in Section 5.3, we annotate summaries manually, in SEER's case separately for each stream. We use The Guardian and other sources as references, but the innumerable topics render a comprehensive ground truth unachievable [62; 252], so the metrics change. We no longer measure recall nor, consequently, the F-score.

The labels change too. We keep the redundant, subjective and noisy labels, but we no longer distinguish between enumerable and non-enumerable topics. Instead, we merge them into a new label, newsworthy topics, which also includes statements by politicians and other authoritative figures. To measure portability, how many topics an algorithm missed, we manually match the newsworthy topics of one algorithm with those of another. We present and discuss our results next.

The sacrifices of understanding to portability

The experiment opens with a familiar scene: ELD's struggles and SEER's triumphs. As Table 6.1a shows, ELD avoided the noise and the subjectivity but not the redundancy, and newsworthiness graced fewer than half of its topics (47.11%). In contrast, SEER thrived. Our novel algorithm captured more noise (6.61% ↑ 7.85%) and more subjectivity (3.31% ↑ 14.39%) than ELD but significantly less redundancy (42.98% ↓ 5.42%). We annotated as newsworthy almost three out of every four topics in SEER (72.34%). We suspected that ELD would struggle, and SEER confirmed our suspicions with aplomb. Yet of sacrifices to portability, we observed few signs.

SEER lost little with understanding. Of ELD's 57 newsworthy topics, just 14 (24.56%) did not appear in SEER, as Table 6.1b shows. More importantly, our algorithm missed details but rarely whole stories. It missed two updates on a post-Brexit trade agreement and Truss' fleeting vow to "ride out the storm", but mostly, it omitted details. In general, SEER missed idle commentary, newspaper headlines and other non-key topics in ELD,

Algorithm	Redundant	Noise	Subjective	Newsworthy	Topics
ELD	▲ 42.98%	6.61%	▼ 3.31%	▽ 47.11%	▽ 121
$SEER_{Default}$	5.49%	▲ 35.16%	△ 25.27%	▼ 34.07%	▽ 91
SEER	5.42%	7.85%	14.39%	72.34%	535

(a) The baselines, in particular ELD, adapted poorly to the new domain, while SEER maintained its performance from Chapter 5. The table reports the micro-average distribution of annotations across all days, and the total number of topics. In our interpretation, precision includes only newsworthy topics. We present a full breakdown of the annotations in Tables F.13 and F.14.

	Baseline's topics		SEER's topics	
Baseline	In SEER	Not in SEER	In baseline	Not in baseline
ELD	43 (75.44%)	14 (24.56%)	120 (31.01%)	267 (68.99%)
$SEER_{Default}$	12 (38.71%)	19 (61.29%)	21 (5.43%)	366 (94.57%)

(b) Despite using understanding, SEER missed few newsworthy topics and detected many more that ELD missed. 75.44% of ELD's newsworthy topics also appeared in SEER, whereas 68.99% of SEER's newsworthy topics did not appear in ELD.

Table 6.1: Far from hindering portability, understanding benefited SEER, both in precision and in comprehensiveness. SEER maintained its gains in precision over ELD and missed few of the baseline's newsworthy topics. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to SEER.

some of which felt more like summarisation anomalies than news stories, such as the nondescript congratulations of a Latvian politician.

What SEER gained with understanding far surpassed what it lost. While 75.44% of ELD's newsworthy topics also appeared in SEER, only 31.01% of SEER's appeared in ELD.[1] The remaining 68.99% of SEER's newsworthy topics captured the congratulations of prominent British politicians, a brewing scandal about Truss' chief-of-staff, and other key and non-key topics from the daily incidents of politics. EVATE's understanding may not have covered the entire domain, but coverage mattered little. Even imperfect understanding compensated for the technical flaws of traditional methods with greater freedom to design a more precise, more comprehensive algorithm.

The results challenge our interpretation of portability. Like Buntain et al. [28], we had assumed that only understanding harms portability, but ELD did not prove portable even without understanding. Consider ELD's uncharacteristically-large share of redundancy (42.98%). ELD's architecture suits fast-moving domains, like football matches,

[1] We annotated SEER's streams separately, but their subjects overlap in closed domains [161]. Whether noisy, subjective or newsworthy, topics frequently appeared in multiple streams at once. Therefore not all of SEER's 387 newsworthy topics are unique.

but in the slow-moving world of politics, the same topics expressed differently appeared as if new. Like so many other TDT algorithms, ELD primarily models user and tweeting behaviours, but behaviours change, and so portability suffers.

In contrast, SEER proved portable despite understanding. Our method primarily models What can be newsworthy, and it made few and reasonable technical assumptions. Like in football matches, for SEER to detect duplicate topics, the conversations about them had to change drastically. Similarly, to detect subjective topics, they had to originate from journalists, politicians or other influential sources capable of shaping Twitter's discourse. Therefore the distribution of annotations barely budged from Chapter 5. We recorded similar figures of redundancy (4.43% ↑ 5.42%), noise (6.90% ↑ 7.85%), subjectivity (17.24% ↓ 14.39%) and newsworthiness (71.43% ↑ 72.34%).

Clearly, portability depends on technique as much as it depends on understanding. In fact, the difference in technical capabilities between ELD and SEER obscures the real influence of understanding on portability. We can deduce that SEER misses a few aspects of key topics but not why, whether due to technique or to understanding. To truly appreciate what our algorithms gains and loses with understanding, we need to compare it with another that boasts similar technical capabilities: SEER itself.

We compare SEER with itself but without understanding. Previously, SEER discarded any tweet that did not mention one of EVATE's 250 political terms. Now, instead of discarding those tweets, it passes them on to a default stream, $\text{SEER}_{\text{Default}}$: a stream like any other except in understanding. The default stream thus reveals which events escaped EVATE's understanding and SEER's scrutiny.

This time, SEER lost more with understanding. The default stream received around a fifth of all tweets (21.19%)—a fifth that SEER never accessed. Logically, the majority of its newsworthy topics, 19 of 31 (61.29%), did not appear in the other streams. The default stream exposed SEER's inability to detect early reports that Thérèse Coffey would become deputy prime minister and that Truss would meet King Charles III. Primarily, however, the default stream exposed its own lack of political understanding. Many of the missed topics lay on the periphery of politics: newspaper round-ups, reactions to a comedian's quips on Truss and whispers about Queen Elizabeth II's health.

The lack of understanding about What can be newsworthy affected performance. The default stream's 31 newsworthy topics constituted a small fraction of its 91 topics—the highest tally of any stream—and precision fell to almost half of SEER's (72.34% ↓ 34.07%). This time, unlike ELD, the default stream did not succumb to redundancy but to inanities: 10.88% more subjectivity than SEER (14.39% ↑ 25.27%) and around five times more noise (7.85% ↑ 35.16%). Including the default stream with all others would have reduced SEER's precision by 5.57%, from 72.34% to 66.77%. The default stream's

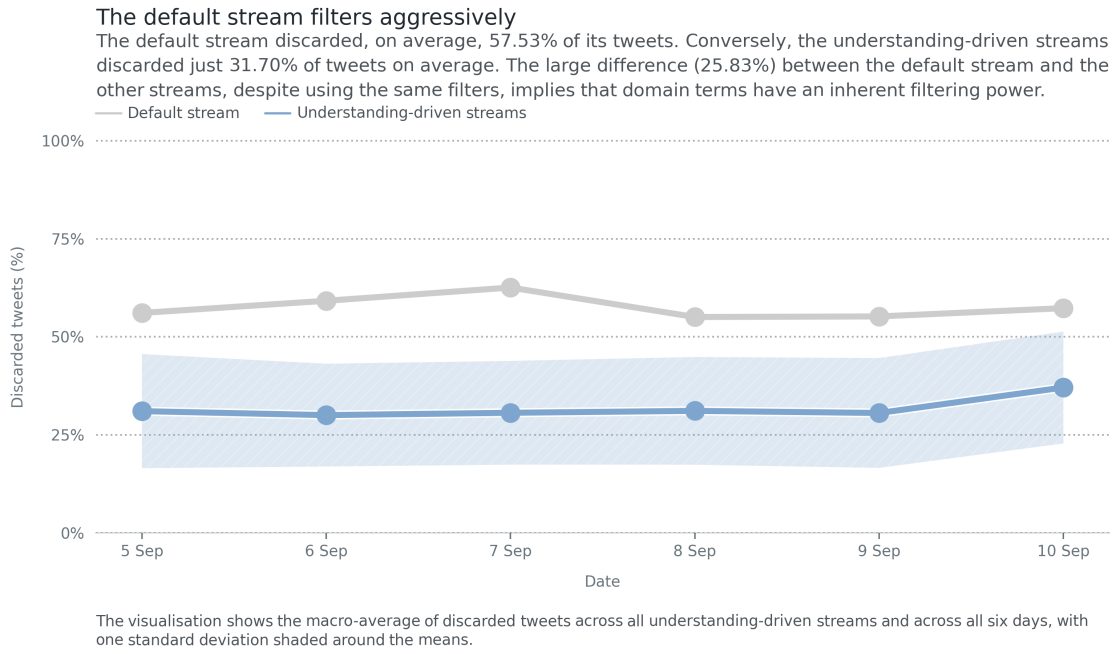

Figure 6.3: The default stream filters more aggressively than the understanding-driven streams. Tweets without domain terms were 25.83% (31.70% ↑ 57.53%) more likely to be filtered than tweets with domain terms.

performance betrayed that it received the dregs that the other streams rejected.

The lack of understanding also appeared forcefully in the filtering behaviours. The default stream detected and tracked like any other stream, and it filtered with the same rules, but it filtered far more aggressively, as Figure 6.3 demonstrates. In fact, we found a significant difference in the way the default stream and all other streams filtered (Mann-Whitney U test: $p = 0.0001$). The default stream filtered 25.83% more tweets than the other streams (31.70% ↑ 57.53%)—an 81.48% relative increase.

The default stream stands as a reflection of the quality of our understanding. If the default stream's technique did not change but performance did, then the tweets themselves must have changed. The tweets changed in one aspect only: they mentioned no domain terms. Those tweets tended to violate our rules about what constitutes noise more often than the rest, and when they did not, they still predominantly contributed noise and subjectivity. In other words, even without consulting user or tweet metadata, understanding acted as a latent filter. TDT research has strived for so long to understand and filter noise when the solution may have lain, all along, in understanding news.

The default stream also reflects on our application of understanding. It vindicates our decision to drive SEER with the What, rather than the Who or the Where; most

tweets and topics in the default stream mentioned Truss, Sunak and the other named entities who we tracked, but without being topical. It also vindicates our decision to design the algorithm around topical streams. The overlapping vocabulary gave SEER the ability to catch the same topics from different aspects, and the comforting security of knowing that even if we miss a topic in one stream, we might detect it in another.

Ultimately, the default stream reflects on portability. Even as we challenged our understanding in a foreign domain, SEER lost little: 19 topics, just 4.55% of the 418 newsworthy topics in SEER's default and understanding-driven streams. Evidently, understanding binds us to one broad domain; we cannot apply our understanding of football matches in politics. Within that domain, however, understanding makes few sacrifices. For portability, TDT literature has sacrificed precision and comprehensiveness, expressiveness and efficiency, sensitivity and scalability. Everything that it sacrificed, it sacrificed in vain.

The benefits of understanding give event tracking another purpose. The improvements in precision, comprehensiveness and sensitivity at a negligible cost to portability lend event tracking and modelling technology a more practical value. At the dawn of a new wave of automation in journalism, we talked with Professor Charlie Beckett, ex-journalist and director of the JournalismAI initiative at the London School of Economics (LSE). We summarise our conversation in the next section.

6.4 | Perspective: Prof. Charlie Beckett

Much of Professor Charlie Beckett's work has focused on the future of news. In 2010, he predicted the rise of networked journalism, the fusion of traditional journalism with modern media and digital tools [19]; today, the internet and social media have become indispensable for journalists [250]. He also predicted that live blogs would become the new front page of newspapers, and today they have become centrepieces on the websites of many of its pioneers, like The Guardian [82]. If his predictions offer any guarantee, then we already know the provenance of journalism's next technological wave: AI. We talked with Professor Beckett about journalism and AI.

Beckett joined the London School of Economics (LSE) in 2016 after seven years working as a programme editor for Channel 4 News. Previously, he had spent a decade as a senior producer and programme editor at the BBC, and worked as a journalist at local news stations. In our conversation, transcribed in full in Appendix C, he told us that the LSE hired him to found and direct Polis, a journalism think-tank, precisely for his practical experience as a journalist. "They deliberately wanted someone from the profession

to come into the university, to bring that kind of perspective."

Since 2019, Beckett has also been directing JournalismAI, a project within Polis to research the intersection where journalism meets AI. "I saw AI as the next technological wave that's happening", he recalled. "[JournalismAI's] main mission is to support good journalism, to be honest, but it also functions as a form of research. By working with these journalists, by teaching them, by doing innovation workshops, we find out both what they think about AI, but also we're actually learning about what you can do with AI: what works, what doesn't work, what impact it has, what consequences it has when you do it."

AI's technological wave has already started. The same week that we met, AI had again made the news, figuratively and literally. In an awkward incident, CNET was forced to defend itself after it was caught furtively using AI to write articles [240]. The episode reflected the wider trend of rapidly-developing generative technology, such as OpenAI's ChatGPT writer and Midjourney's AI-driven image generator, and gave journalism a glimpse of what the future holds. AI is no longer emerging but evolving and developing, and journalism finds itself having to keep pace.

Still, Beckett remains steadfast: AI will not replace the human in journalism. His 2019 survey, where JournalismAI began, opens with a terse declaration: "No, the robots are not going to take over journalism" [18]. He imagines a more complementary role for AI. "AI is generally used to supplement human labour. It very rarely replaces it, and in many cases it actually creates new labour, which can be good, either by creating new formats or by the need to review and edit the actual technology."

Beckett's perspective is shared by his peers [51; 149]. The right question to ask, he told us, is what humans do better than machines. He believes in the human qualities of judgement and morality, efficiency and creativity, emotion and empathy. In fact, the use of technology could make more space for those human qualities. "Perhaps human journalists will get better and do more of the creative, empathy, judgment stuff. Again though, it's a false binary to say that's completely different and can't be supported by technology".

What about understanding? Yes, machines lack understanding too. By that, Beckett did not mean "some deep intellectual genius", but something more basic, more fundamental—more human. He illustrated his point with VAR, the technology on the sidelines of football matches. VAR too burdens humans with the final judgements: what counts as a deliberate handball or as a natural movement of the arm.

Another human quality, social connection, underpins Beckett's earlier prediction, networked journalism. Networked journalism encapsulates the relationship between journalists and the public, a digital connection that extends beyond social media. It

extends especially beyond Twitter, which he called too "small" and "unrepresentative" to capture the social connection alone. "We think of social media as the platform, but it's not. In social media, it's the social bit that's important. ... What I think is partly that journalists are learning about the limits of a platform like Twitter [and] thinking about other forms of data apart from social media discourse." Stricter regulations, financial troubles and erratic management may well make a market reckoning unavoidable.

Event tracking, even on Twitter, still has a role to play in the newsroom. Beckett's recollection of the live-blogging process was of a largely-manual operation: a journalist sitting at their desk, browser open and scouring webpages for news. Perhaps they are supported by technology too, he added. Perhaps they could be supported better by event tracking technology. When we presented Beckett with the interactive demos shown in Figure 6.4, he remarked about the timeline's relevance, SEER's precision during Truss' fateful first week as UK prime minister. "Something like this can be the backbone to a live blog."

Event tracking technology does not exclude the human. In line with Beckett's vision of journalism and AI, the journalist plays a complementary role. "All those things that journalists do in a formulaic way, [SEER] will do well", he told us, but for the rest, it requires the human. We returned, again, to human judgement. "The interesting bit—and this is what journalists talked to me about when they say they're using things like this—they say that they use this, and when they see something that's a bit stand-out, they can follow up on it. That idea of judgment."

Nevertheless, it feels inexact to say that event tracking simply detects the news. Rather, in its contemporary form it detects the news that Twitter's "small" and "unrepresentative" network finds interesting. When we presented this dilemma to Beckett, however, he responded with one of his own: between reporting the news that editors and journalists find interesting, and the news that resonates with audiences. The argument fits within the context of rising news avoidance [181]. "We see [news avoidance] as a problem. We aren't seeing it as quite a normal response to the world, a world of abundant information". Event tracking can thus play a second, less obvious role:

> I think that we can use these technologies firstly to understand better what's happening in the world, but secondly to understand what interests people. How do things connect to people's lives and what does our audience do? That's possibly the biggest revolution in journalism in the last ten years: audience data. We now understand what people do with news. We don't understand why or how they feel about it particularly, but we can at least measure their behaviour. How do we do that? We do it with this software.

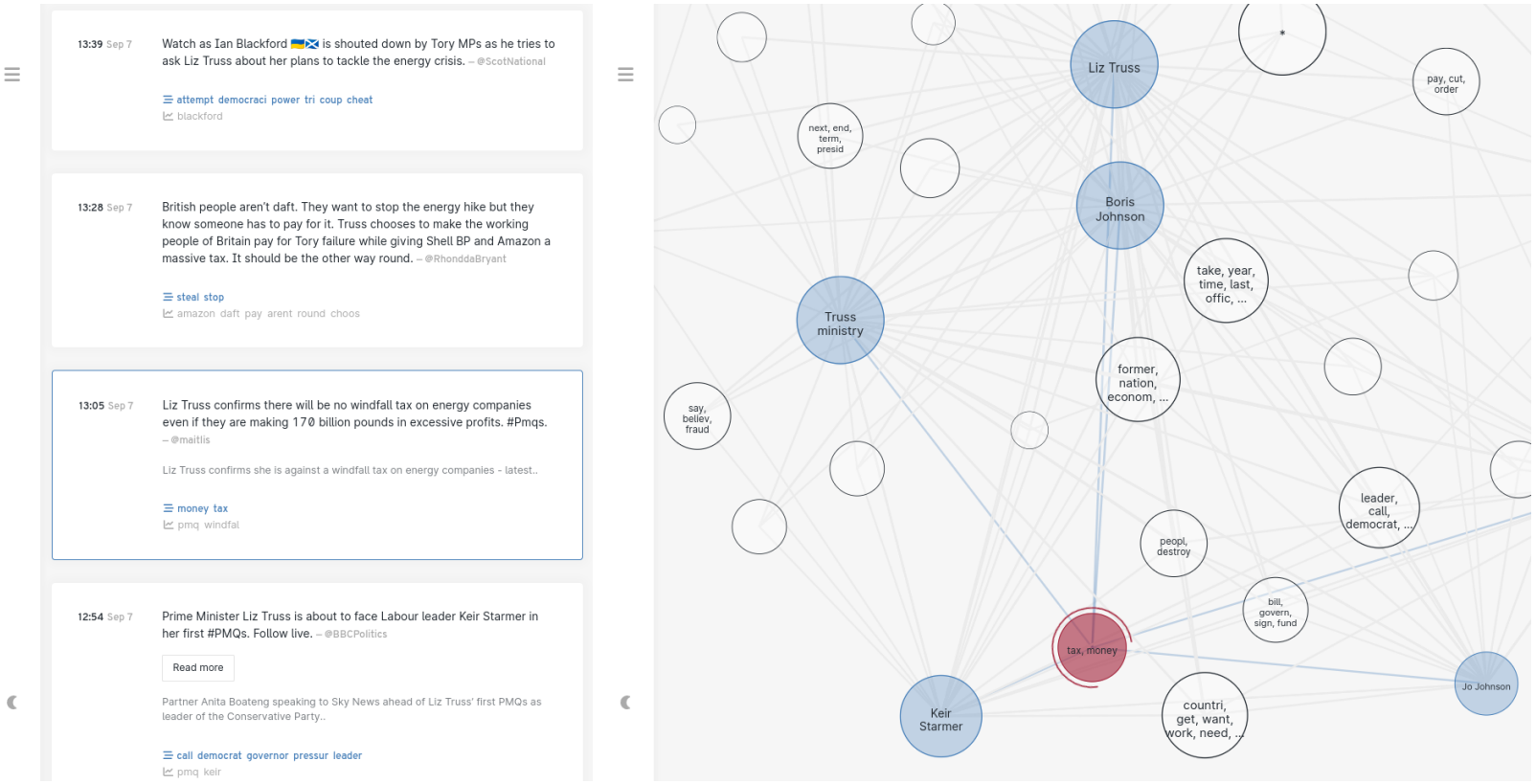

Figure 6.4: Before our conversation, we sent Professor Charlie Beckett two interactive demos. On the left, SEER's timeline supports filtering based on topic popularity, streams and keywords. On the right, the event knowledge graph has adjustable level-of-detail, and clickable nodes and edges to filter the timeline based on Who did What and Where.

— Professor Charlie Beckett

The event knowledge graph can also support efforts to counter news avoidance. Beckett expressed his annoyance at the endless cavilling of journalists that alienates audiences. "They'll talk about the emphasis from the Prime Minister that was subtly different today. ... The general public are thinking 'I don't understand, sounds to me like the same thing, this is so fine-tuning.' They don't understand the context: what did he say before?" The knowledge graph could provide that context and guide explanatory journalism, which now finds near-universal acceptance among newsrooms [181]. "You can match with your knowledge graph: this is what they said about taxation before; this is what they are saying now; this is what it means."

Identifying more specific applications of the knowledge graph proved more difficult. Partly, the difficulty stems from the graph's function: not as a final output but as a resource to contextualise the news or discover news angles [188], or as Beckett called it, a form of data journalism. "I think we are seeing increasingly the kind of journalism—and it's usually data journalism of some sort, which is what this is, in a sense—that makes connections. We saw a lot of it during the [COVID-19] pandemic, of course, where there was that kind of 'how is the pandemic changing society?' I could imagine you doing this kind of thing with an issue like the pandemic."

Our brief conversation served to confirm another of Beckett's predictions. AI continues to deliver to journalism the new powers and new responsibilities that named his report [18]. Yet it also continues to expose the new needs of journalism. JournalismAI has been exploring the opportunities of AI through its fellowship programmes, and almost unfailingly, projects depart from first principles: What is a quote or a claim? What is a politician? A promising sign of academic rigour, perhaps, but also of newsrooms wrestling with novel problems that AI research has yet to address. A disconnect remains between journalists and technologists, both firmly encamped in their fields.

At the end, we did not ask Beckett for a prediction but for an appeal: how AI research can address journalism's needs. "We're not so good at thinking about the consequences of potential application in a more general way: how it might change the journalistic practice. That's the thing I look at. The people who know about the technology can help think through those kind of applications. It's not just how this can help journalism—that's the first one. ... The second one is: how might it change journalism?" Journalists have started to understand the technology. Now, they need technologists to start understanding journalism. We conclude this chapter next.

Principal contributions

- The first event modeller led by understanding and understanding-driven TDT
- Confirmation that the benefits of understanding to precision, recall and sensitivity outweigh any sacrifices to portability
- An interview with Professor Charlie Beckett on the future of event tracking in the newsroom

Recap

Without understanding, Reuters Tracer grew in complexity. Reuters had to build a distributed architecture to process, classify and filter more than 12 million tweets daily. Every day, an incremental clustering algorithm developed, merged and filtered over 16,000 clusters [129], and still, Reuters Tracer had to deploy a set of heuristics and models to help journalists make sense of the thousands of events that remained [130]. Reuters Tracer stands as an exemplar of computational journalism, but also as a lesson and as a warning.

With understanding, Reuters Tracer might have assisted journalists better. It might have detected more news with a higher precision, or even earlier than it did. Perhaps it might have required fewer resources or helped organise events better. Certainly, understanding held greater promise of benefits than of sacrifices for Reuters Tracer, like it held for SEER. In this chapter, we demonstrated that understanding could be TDT research's next paradigm shift by answering the following questions:

- What makes event modelling so complex? Event modelling literature seeks to model TDT research's events, but without proper event understanding, it falls to the same traps. In Section 6.1, we discussed how understanding could simplify event modelling like it simplified event tracking.
- How can understanding make resources of events? Modelling events formally only poses a complex task because algorithms must understand events instead of simply applying understanding. In Section 6.2, we demonstrated how understanding and understanding-driven TDT could simplify and augment the event modelling process.
- What do we sacrifice with understanding? In retrospect, it seems obvious that un-

derstanding would benefit TDT algorithms, but fears of its sacrifices to portability hindered research on understanding [28]. In Section 6.3, we allayed those fears by showing that what SEER gains with understanding thoroughly outweighs what it loses.

- What role will event tracking play in the modern newsroom? In journalism's technological wave of automation, the machine supplements the human. In Section 6.4, our conversation with Professor Charlie Beckett revealed how event tracking technologies could form the backbone of live blogs, help counter news avoidance and power a new form of data journalism.

This chapter concludes our work on understanding. Throughout this report, we studied TDT literature's decades-long difficulties to interpret the meaning of understanding, to understand and to apply that understanding, and to face the emerging challenges of user-generated content and computational journalism. As we studied the difficulties, however, their roots appeared plainly: simplistic theories, improper understanding and shallow application of it. The difficulties emerged not from understanding but from its very absence. We conclude this report with a summary of our main findings in the next chapter.

7

Conclusion

Can understanding improve TDT? Yes, it can, and even in 1998, the TDT community must have sensed that understanding could improve its algorithms. Its mistake was never about suggesting event knowledge as the solution but never having progressed past the initial suggestion, never having asked the questions that followed: what it means to understand events, and how it could develop and best apply understanding.

Without understanding, the research area's early decades declined into a Sisyphean struggle. Again and again, the TDT community looked for a substitute for understanding, first in linguistics, then in alternative methods and finally, when all else failed, in increasingly-complex algorithms. Again and again, the TDT community found no substitute. This dissertation demonstrates why: proper understanding has no substitute.

Still, our Yes feels reductive. Understanding does not merely improve TDT, as Allan et al. [9] envisaged, but gives it new meaning. Understanding represents a bridge between what TDT can do and what we need it to do, and between what TDT is and what it can be. Mitchell [167, 168] argues that machines do not really understand, do not grasp true meaning; they only pretend to. We do not purport to have given machines true understanding, but we hope to have given them the tools with which to pretend better. In this dissertation, we addressed the following aims and objectives:

- What does it mean to understand events? To understand events means, first and foremost, to define events. In Chapter 2, we chose a semantic and structured definition to guide our understanding and connect TDT with other event-related research areas: Who does What, Where and When.

- When does a named entity become a participant? Named entities become participants when Who or Where they are determines What happens in events. In

Principal contributions

- In Chapter 1, the `NicholasMamo/EvenTDT` repository, the largest open-source TDT library
- In Chapter 2, the first literature review on understanding in TDT and other event-related research areas
- In Chapter 3, DEPICT, a novel APD algorithm that understands Who participates in events and Where by understanding the participants themselves
- In Chapter 4, EVATE, the first ATE algorithm that understands What happens in events from Twitter
- In Chapter 5, SEER, the first TDT algorithm driven by foreknowledge of What can happen in an event, and which proves that understanding can improve event tracking
- In Chapter 6, a novel event modeller simplified and augmented by understanding and understanding-driven TDT

Chapter 3, we proposed DEPICT, a novel APD algorithm that understands the Who and the Where by learning what makes participants of named entities.

- When does a word become a domain term? Words become domain terms when they carry a semantic value that describes the actions and changes—What happens in events. In Chapter 4, we proposed EVATE, a novel ATE algorithm that understands the What much like humans, by observing events.

- How can understanding improve TDT algorithms? Allan et al. [9] hoped to make algorithms more accurate, but understanding can elevate event tracking in other ways too. In Chapter 5, we proposed SEER, our answer to the suggestion made by Allan et al. [9] and a monument to the many virtues of understanding.

- Where does TDT's next revolution lie? It no longer suffices for TDT algorithms to detect and track events without describing. In Chapter 6, we proposed the understanding-driven event modeller, a demonstration of how understanding can transform event trackers into event modellers, and their outputs into resources.

7.1 | What comes after understanding

Understanding is a beginning, not the end. As the TDT pilot study reached its completion, Allan et al. [7] concluded the final report on a hopeful note: "the technologies applied solve large portions of the problem, but leave substantial room—and hope—for improvement." We conclude our report similarly. This time, understanding solves large portions of the problem, but leaves substantial room—and hope—for improvement. Developing understanding means developing TDT.

TDT research needs understanding, today perhaps even more than in 1998. Literature has unsolved problems and unmet needs, but in understanding, it may well find the solutions. TDT research needs understanding for its own sake, to fulfil the potential that Allan et al. [7] saw in those early techniques, and for the sake of the research areas that depend on it. Above all, it needs understanding to move past solving data issues and technical problems, and start solving real-world challenges. In our vision, the road to understanding goes on as follows:

- In Chapter 2, the very thing that eluded TDT research for so long—a semantic definition of events—unlocked our thesis, but we have only started filling in the theoretical void. To create a meaningful connection among event-related research areas, we need a common language of events, one idea of what we must understand. Future work should extend the theory of Who does What, Where and When to cover the Why and the How, the relationships that link events together.

- In Chapter 3, DEPICT understood the Who and the Where retrospectively and from the microcosm of one event. If we could understand the prototypical participant of an event domain, not a singular event, then APD could serve in more scenarios, including real-time disambiguation. Future work should generalise DEPICT's understanding, from inferring the common attributes of one event's participants to inferring the common attributes of an event domain's participants.

- In Chapter 4, EVATE understood the What semantically, not unlike a human but far simpler. Human understanding takes a more abstract form: not isolated terms but connected concepts that serve more complex applications, like SEER's topical streams. Future work should develop EVATE's understanding of What happens in event domains into semantically-richer ideas, such as by mapping terms to WordNet senses.

- In Chapter 5, SEER showed one application of understanding, but event knowledge has many uses. TDT research has proposed innumerable algorithms, without

understanding but with limitless ways for it to suffuse them. Future work should explore how understanding could make techniques more precise and comprehensive, expressive and timely, sensitive and scalable, lightweight and efficient, parameter-free and portable, on Twitter and elsewhere.

- In Chapter 6, we showed how understanding can simplify event modelling, but the process to understand remains daunting. We will not harness the ideas in this dissertation until we have understanding readily-available to simplify and augment our algorithms. Future work should build upon our principles and our vision in this chapter to create a connected whole that describes Who usually does What, Where and When, and Why and How: an event domain ontology.

The year is no longer 1996. The question of whether understanding can improve TDT belongs to the past. New questions can take its place. How does a TDT algorithm understand? How does it apply understanding and to what effect?

References

[1] Abhishek, K., Pratihar, V., Shandilya, S. K., Tiwari, S., Ranjan, V. K., and Tripathi, S. An Intelligent Approach for Mining Knowledge Graphs of Online News. *International Journal of Computers and Applications*, page 1–9, August 2021. doi:10.1080/1206212X.2021.1957551.

[2] Adedoyin-Olowe, M., Gaber, M. M., Dancausa, C. M., Stahl, F., and Gomes, J. B. A Rule Dynamics Approach to Event Detection in Twitter with its Application to Sports and Politics. *Expert Systems with Applications*, 55:351–360, Aug 2016. doi:10.1016/j.eswa.2016.02.028.

[3] AFP. Germany 'No Longer Always Win' - Lineker Tweaks Quote, Jun 2018. URL `https://www.france24.com/en/20180627-germany-no-longer-always-win-lineker-tweaks-quote`. Accessed on September 23, 2021.

[4] Ahmad, K., Gillam, L., and Tostevin, L. University of Surrey Participation in TREC8: Weirdness Indexing for Logical Document Extrapolation and Retrieval (WILDER). In *TREC-8: Proceedings of the Eighth Text Retrieval Conference*, pages 717–724, Gaithersburg, Maryland, USA, Nov 1999. URL `https://trec.nist.gov/pubs/trec8/t8_proceedings.html`.

[5] Aiello, L. M., Petkos, G., Martin, C., Corney, D., Papadopoulos, S., Skraba, R., Goker, A., Kompatsiaris, Y., and Jaimes, A. Sensing Trending Topics in Twitter. *IEEE Transactions on Multimedia*, 15(6): 1268–1282, Oct 2013. doi:10.1109/TMM.2013.2265080.

[6] Akhtar, N. and Siddique, B. Hierarchical Visualization of Sport Events Using Twitter. *Journal of Intelligent & Fuzzy Systems*, 32(4):2953–2961, Mar 2017. doi:10.3233/JIFS-169238.

[7] Allan, J., Carbonell, J. G., Doddington, G., Yamron, J., and Yang, Y. Topic Detection and Tracking Pilot Study Final Report. In *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, pages 194–218, Lansdowne, Virginia, USA, Feb 1998. URL `https://kilthub.cmu.edu/articles/journal_contribution/Topic_Detection_and_Tracking_Pilot_Study_Final_Report/6626252`.

[8] Allan, J., Lavrenko, V., and Papka, R. Event Tracking. Technical report, Center for Intelligent Information Retrieval, Computer Science Department, University of Massachusetts, Amherst, MA, USA, Jan 1998. URL `https://www.academia.edu/2788260/Event_tracking`.

[9] Allan, J., Papka, R., and Lavrenko, V. On-Line New Event Detection and Tracking. In *SIGIR '98: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 37–45, Melbourne, Australia, Aug 1998. ACM. doi:10.1145/290941.290954.

[10] Allan, J., Jin, H., Rajman, M., Wayne, C., Gildea, D., Lavrenko, V., Hoberman, R., and Caputo, D. Topic-Based Novelty Detection. In *Topic-Based Novelty Detection: 1999 Summer Workshop at CLSP, Final Report*, page 1–51, Baltimore, MD, USA, Jul 1999. August. URL `https://www.cs.cmu.edu/~roseh/Papers/baltWs99.ps`.

[11] Allan, J., Lavrenko, V., and Swan, R. Explorations within Topic Tracking and Detection. In *Topic Detection and Tracking—Event-based Information Organization*, volume 12, pages 197–224. Springer, Boston, MA, USA, Topic Detection and Tracking edition, 2002. doi:10.1007/978-1-4615-0933-2_10.

[12] Almuhareb, A. and Poesio, M. Attribute-Based and Value-Based Clustering: An Evaluation . In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, page 158–165, Barcelona, Spain, Jul 2004. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/W04-3221/`.

[13] Astrakhantsev, N. A., Fedorenko, D. G., and Turdakov, D. Y. Methods for Automatic Term Recognition in Domain-Specific Text Collections: A Survey. *Programming and Computer Software*, 41(6): 336–349, Nov 2015. doi:10.1134/S036176881506002X.

[14] Balachandran, K. and Ranathunga, S. Domain-Specific Term Extraction for Concept Identification in Ontology Construction. In *Proceedings of the 2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, page 34–41, Omaha, Nebraska, USA, Oct 2016. IEEE. doi:10.1109/WI.2016.0016.

[15] Basili, R., De Rossi, G., and Pazienza, M. T. Inducing Terminology for Lexical Acquisition. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 125–133, Providence, Rhode Island, USA, Aug 1997. URL `https://www.aclweb.org/anthology/W97-0314/`.

[16] Basili, R., Moschitti, A., Pazienza, M. T., and Zanzotto, F. M. A Contrastive Approach to Term Extraction. In *TIA 2001 : Terminologie et Intelligence Artificielle*, page 119–128, Nancy, France, May 2001. URL `https://pascal-francis.inist.fr/vibad/index.php?action=getRecordDetail&idt=1168884`.

[17] Basili, R., Pazienza, M. T., and Zanzotto, F. M. Modelling Syntactic Context in Automatic Term Extraction. In *Proceedings of Recent Advances in Natural Language Processing (RANLP '01)*, page 28–34, Tzigov Chark, Bulgaria, Sep 2001. URL `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.3.4638`.

[18] Beckett, C. New Powers, New Responsibilities: A Global Survey of Journalism and Artificial Intelligence. Technical report, The London School of Economics and Political Science, Nov 2019. URL `https://www.lse.ac.uk/media-and-communications/polis/JournalismAI/The-report`.

[19] Beckett, C., Herve-Azevedo, J., Moughan, K., McDougall, D., Moreira, D., Kjaernested, B., and Doctor, S. The Value of Networked Journalism. Technical report, The London School of Economics and Political Science, Jun 2010. URL `http://eprints.lse.ac.uk/31050/1/Beckett_Value_networked_journalism_2010.pdf`.

[20] Berven, A., Christensen, O. A., Moldeklev, S., Opdahl, A. L., and Villanger, K. J. A Knowledge-Graph Platform for Newsrooms. *Computers in Industry*, 123:1–10, Dec 2020. doi:10.1016/j.compind.2020.103321.

[21] Bing, L., Lam, W., and Wong, T.-L. Wikipedia Entity Expansion and Attribute Extraction from the Web Using Semi-Supervised Learning. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, page 567–576, Rome, Italy, Feb 2013. Association for Computing Machinery. doi:10.1145/2433396.2433468.

[22] Bird, S., Loper, E., and Klein, E. *Natural Language Processing with Python*. O'Reilly Media Inc., 1 edition, Jun 2009. URL `https://www.nltk.org/book/`.

[23] Bolisani, E. and Bratianu, C. The Elusive Definition of Knowledge. In *Emergent Knowledge Strategies*, volume 4, page 1–22. Springer International Publishing AG, Jul 2017. ISBN 978-3-319-60656-9. URL `https://link.springer.com/chapter/10.1007/978-3-319-60657-6_1`.

[24] Bolshakova, E., Loukachevitch, N., and Nokel, M. Topic Models Can Improve Domain Term Extraction. In *ECIR 2013: Advances in Information Retrieval*, pages 684–687, Moscow, Russia, Mar 2013. Springer, Berlin, Heidelberg. doi:10.1007/978-3-642-36973-5_60.

[25] Bonin, F., Dell'Orletta, F., Venturi, G., and Montemagni, S. A Contrastive Approach to Multi-word Extraction from Domain-Specific Corpora. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, page 3222–3229, Valletta, Malta, May 2010. European Language Resources Association (ELRA). URL `https://www.aclweb.org/anthology/L10-1379/`.

[26] Bontcheva, K. and Rout, D. Making Sense of Social Media Streams Through Semantics: A Survey. *Semantic Web*, 5(5):373–403, 2014. URL `https://content.iospress.com/articles/semantic-web/sw110`.

[27] Brown, G. W., McLean, I., and McMillan, A. *The Concise Oxford Dictionary of Politics and International Relations*. Oxford Quick Reference. Oxford University Press, Oxford, 4th edition, 2018. ISBN 9780199670840. URL `https://www.oxfordreference.com/view/10.1093/acref/9780199670840.001.0001/acref-9780199670840`.

[28] Buntain, C., Lin, J., and Golbeck, J. Discovering Key Moments in Social Media Streams. In *2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, page 366–374, Las Vegas, NV, USA, Jan 2016. IEEE. doi:10.1109/CCNC.2016.7444808.

[29] Burnside, G., Milioris, D., and Jacquet, P. One Day in Twitter: Topic Detection Via Joint Complexity. In *Proceedings of the SNOW 2014 Data Challenge*, page 41–48, Seoul, Korea, Apr 2014. CEUR. URL `http://ceur-ws.org/Vol-1150/`.

[30] Campos, R., Mangaravite, V., Pasquali, A., Jorge, A., Nunes, C., and Jatowt, A. YAKE! Keyword Extraction from Single Documents Using Multiple Local Features. *Information sciences*, 509:257–289, Jan 2020. doi:10.1016/j.ins.2019.09.013.

[31] Cao, X., Shi, C., Zheng, Y., Ding, J., Li, X., and Wu, B. A Heterogeneous Information Network Method for Entity Set Expansion in Knowledge Graph. In *Advances in Knowledge Discovery and Data Mining*, page 288–299, Melbourne, VIC, Australia, Jun 2018. Springer International Publishing. ISBN 0302-9743. doi:10.1007/978-3-319-93037-4_23.

[32] Cataldi, M., Caro, L. D., and Schifanella, C. Personalized Emerging Topic Detection Based on a Term Aging Model. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(1):1–27, Dec 2013. doi:10.1145/2542182.2542189.

[33] Chakrabarti, D. and Punera, K. Event Summarization Using Tweets. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, pages 66–73, Barcelona, Spain, Jul 2011. The AAAI Press. doi:10.1609/icwsm.v5i1.14138.

[34] Chen, C. and Terejanu, G. Sub-Event Detection on Twitter Network. In *AIAI 2018: AIAI: 14th IFIP International Conference on Artificial Intelligence Applications and Innovations*, page 50–60, Rhodes, Greece, May 2018. Springer International Publishing. doi:10.1007/978-3-319-92007-8_5.

[35] Chen, H.-H. and Ku, L.-W. An NLP & IR Approach to Topic Detection. In *Topic Detection and Tracking—Event-based Information Organization*, volume 12, pages 243–364. Springer, Boston, MA, USA, Topic Detection and Tracking edition, 2002. doi:10.1007/978-1-4615-0933-2_12.

[36] Chen, J., Chen, Y., Du, X., Zhou, X., and Zhang, X. SEED: A System for Entity Exploration and Debugging in Large-Scale Knowledge Graphs. In *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*, page 1350–1353, Helsinki, Finland, May 2016. IEEE. doi:10.1109/ICDE.2016.7498342.

[37] Chen, J., Chen, Y., Zhang, X., Du, X., Wang, K., and Wen, J.-R. Entity Set Expansion with Semantic Features of Knowledge Graphs. *Journal of Web Semantics*, 52–53:33–44, Oct 2018. doi:10.1016/j.websem.2018.09.001.

[38] Chen, L., Chun, L., Ziyu, L., and Quan, Z. Hybrid Pseudo-Relevance Feedback for Microblog Retrieval. *Journal of Information Science*, 39(6):773–788, May 2013. doi:10.1177/0165551513487846.

[39] Chen, X. and Li, Q. Event Modeling and Mining: A Long Journey Toward Explainable Events. *The VLDB Journal*, 29(1):459–482, Jan 2020. doi:10.1007/s00778-019-00545-0.

[40] Chierichetti, F., Kleinberg, J., Kumar, R., Mahdian, M., and Pandey, S. Event Detection via Communication Pattern Analysis. In *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*, page 51–60, Ann Arbor, MI, USA, Jun 2014. Association for the Advancement of Artificial Intelligence. doi:10.1609/icwsm.v8i1.14536.

[41] Choi, H.-J. and Park, C. H. Emerging Topic Detection in Twitter Stream Based on High Utility Pattern Mining. *Expert Systems with Applications*, 115:27–36, Jan 2019. doi:10.1016/j.eswa.2018.07.051.

[42] Choudhury, S. and Breslin, J. G. Extracting Semantic Entities and Events from Sports Tweets. In *Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts': Big Things Come in Small Packages*, pages 22–32, Heraklion, Crete, May 2011. URL `http://oro.open.ac.uk/32460/`.

[43] Chung, T. M. A Corpus Comparison Approach for Terminology Extraction. *Terminology*, 9(2):221–246, 2003. doi:10.1075/term.9.2.05chu.

[44] Corney, D., Martin, C., and Göker, A. Spot the Ball: Detecting Sports Events on Twitter. In *ECIR 2014: Advances in Information Retrieval*, pages 449–454, Amsterdam, The Netherlands, Apr 2014. Springer. doi:10.1007/978-3-319-06028-6_40.

[45] Crow, J. M. Verifying Baselines for Crisis Event Information Classification on Twitter. In *ISCRAM 2020 Conference Proceedings – 17th International Conference on Information Systems for Crisis Response and Management*, page 670–687, Blacksburg, VA, USA, May 2020. ISCRAM. URL http://idl.iscram.org/files/justinmichaelcrow/2020/2263_JustinMichaelCrow2020.pdf.

[46] Das, P. and Das, A. K. A Word Clustering-Based Crime Report Categorization Technique. In *Proceedings of CIPR 2020: Computational Intelligence in Pattern Recognition*, page 451–463, Kolkata, West Bengal, India, Jan 2020. Springer Singapore. doi:10.1007/978-981-15-2449-3_39.

[47] Dataminr. Deutsche Welle Uses Dataminr to Stay Competitive and Keep Pace With Evolving Media Industry, 2022. URL https://www.dataminr.com/resources/deutsche-welle-uses-dataminr-to-stay-competitive-and-keep-pace-with-evolving-media-industry. Accessed on Jan 5, 2023.

[48] Davidov, D. and Rappoport, A. Efficient Unsupervised Discovery of Word Categories Using Symmetric Patterns and High Frequency Words. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, page 297–304, Sydney, Australia, Jul 2006. Association for Computational Linguistics. doi:10.3115/1220175.1220213.

[49] De Boom, C., Van Canneyt, S., and Dhoedt, B. Semantics-Driven Event Clustering in Twitter feeds. In *Proceedings of the 5th Workshop on Making Sense of Microposts*, page 2–9, Florence, Italy, May 2015. CEUR. ISBN 1613-0073. URL http://ceur-ws.org/Vol-1395/.

[50] Di Corso, E., Proto, S., Vacchetti, B., Bethaz, P., and Cerquitelli, T. Simplifying Text Mining Activities: Scalable and Self-Tuning Methodology for Topic Detection and Characterization. *Applied Sciences*, 12 (10):1–41, May 2022. doi:10.3390/app12105125.

[51] Diakopoulos, N. *Automating the News: How Algorithms are Rewriting the Media*. Harvard University Press, Cambridge, Jun 2019. ISBN 9780674976986. URL https://www.hup.harvard.edu/catalog.php?isbn=9780674976986.

[52] Dole Institute of Politics. Political Glossary, 2021. URL https://doleinstitute.org/get-involved/civic-engagement-tools/political-glossary/. Accessed on Jul 14, 2021.

[53] Dorow, B. and Widdows, D. Discovering Corpus-Specific Word Senses. In *EACL '03: Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics - Volume 2*, page 79–82, Budapest, Hungary, Apr 2003. Association for Computational Linguistics. doi:10.3115/1067737.1067753.

[54] Doyle, P. Southampton 0-2 Arsenal: Premier League – As It Happened, Jun 2020. URL https://www.theguardian.com/football/live/2020/jun/25/southampton-v-arsenal-premier-league-live. Accessed on January 24, 22.

[55] Earle, P. S., Bowden, D., and Guy, M. Twitter Earthquake Detection: Earthquake Monitoring in a Social World. *Annals of Geophysics*, 54(6):708–715, Jun 2011. doi:10.4401/ag-5364.

[56] Edouard, A., Cabrio, E., Tonelli, S., and Le Thanh, N. Graph-Based Event Extraction from Twitter. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, page 222–230, Varna, Bulgaria, Sep 2017. INCOMA Ltd. doi:10.26615/978-954-452-049-6_031.

[57] Ehrlinger, L. and Wöß, W. Towards a Definition of Knowledge Graphs. In *Joint Proceedings of the Posters and Demos Track of the 12th International Conference on Semantic Systems - SEMANTiCS2016 and the 1st International Workshop on Semantic Change & Evolving Semantics (SuCCESS'16)*, page 1–4, Leipzig, Germany, Sep 2016. CEUR-WS. URL `https://ceur-ws.org/Vol-1695/`.

[58] El-Kassas, W. S., Salama, C. R., Rafea, A. A., and Mohamed, H. K. Automatic Text Summarization: A Comprehensive Survey. *Expert Systems with Applications*, 165:1–26, Mar 2021. doi:10.1016/j.eswa.2020.113679.

[59] English First. 1000 Most Common Words in English, 2021. URL `https://www.ef.com/wwen/english-resources/english-vocabulary/top-1000-words/`. Accessed on July 16, 2021.

[60] F1technical.net. Formula One Glossary, 2021. URL `https://www.f1technical.net/glossary/`. Accessed on August 12, 2021.

[61] Farnaghi, M., Ghaemi, Z., and Mansourian, A. Dynamic Spatio-Temporal Tweet Mining for Event Detection: A Case Study of Hurricane Florence. *International Journal of Disaster Risk Science*, 11(3): 378–393, Jun 2020. doi:10.1007/s13753-020-00280-z.

[62] Farzindar, A. and Khreich, W. A Survey of Techniques for Event Detection in Twitter. *Computational Intelligence*, 31(1):132–164, Feb 2015. doi:10.1111/coin.12017.

[63] Federici, T. and Rego, R. Résultat et Résumé Paris-SG - Marseille, Trophée des Champions, Trophée des Champions, Mercredi 13 Janvier 2021, Jan 2021. URL `https://www.lequipe.fr/Football/match-direct/trophee-des-champions/2020/psg-om-live/505182`. Accessed on March 13, 2021.

[64] Feng, W., Zhang, C., Zhang, W., Han, J., Wang, J., Aggarwal, C., and Huang, J. STREAMCUBE: Hierarchical Spatio-Temporal Hashtag Clustering for Event Exploration over the Twitter Stream. In *2015 IEEE 31st International Conference on Data Engineering*, page 1561–1572, Seoul, South Korea, Apr 2015. IEEE. ISBN 1063-6382. doi:10.1109/ICDE.2015.7113425.

[65] Fernando, S., Amador Díaz López, J., Şerban, O., Gómez-Romero, J., Molina-Solana, M., and Guo, Y. Towards a Large-Scale Twitter Observatory for Political Events. *Future generation computer systems*, 110:976–983, Sep 2020. doi:10.1016/j.future.2019.10.013.

[66] Filatova, E., Hatzivassiloglou, V., and McKeown, K. Automatic Creation of Domain Templates. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 207–214, Sydney, Australia, Jul 2006. Association for Computational Linguistics. URL `https://aclanthology.org/P06-2027/`.

[67] Formula 1. F1 Glossary - A–Z List of the Top Formula 1 Terms, 2021. URL `https://www.formula1.com/en/championship/inside-f1/glossary.html`. Accessed on August 12, 2021.

[68] Formula 1 Dictionary. Formula 1 Dictionary, 2021. URL `formula1-dictionary.net`. Accessed on August 12, 2021.

[69] Founta, A.-M., Djouvas, C., Chatzakou, D., Leontiadis, I., Blackburn, J., Stringhini, G., Vakali, A., Sirivianos, M., and Kourtellis, N. Large Scale Crowdsourcing and Characterization of Twitter Abusive Behavior. In *Proceedings of the Twelfth International AAAI Conference on Web and Social Media*, page

491–500, Stanford, CA, USA, Jun 2018. Association for the Advancement of Artificial Intelligence. doi:10.1609/icwsm.v12i1.14991.

[70] Frantzi, K., Ananiadou, S., and Mima, H. Automatic Recognition of Multi-Word Terms: the C-value/NC-value Method. *International Journal on Digital Libraries*, 3(2):115–130, Aug 2000. doi:10.1007/s007999900023.

[71] Freitas, J. and Ji, H. Identifying News from Tweets. In *Proceedings of the First Workshop on NLP and Computational Social Science*, page 11–16, Austin, Texas, USA, Nov 2016. Association for Computational Linguistics. doi:10.18653/v1/W16-5602.

[72] Fung, G. P. C., Yu, J. X., Yu, P. S., and Lu, H. Parameter Free Bursty Events Detection in Text Streams. In *VLDB '05: 31st International Conference on Very Large Data Bases*, page 181–192, Trondheim, Norway, Aug 2005. VLDB Endowment. URL `https://dl.acm.org/doi/10.5555/1083592.1083616`.

[73] GabAllah, N. and Rafea, A. Unsupervised Topic Extraction from Twitter: A Feature-pivot Approach. In *Proceedings of the 15th International Conference on Web Information Systems and Technologies - Volume 1: WEBIST*, page 185–192, Vienna, Austria, Sep 2019. SCITEPRESS – Science and Technology Publications. doi:10.5220/0007959001850192.

[74] Gabrilovich, E. and Markovitch, S. Computing Semantic Relatedness using Wikipedia-Based Explicit Semantic Analysis. In *IJCAI'07: Proceedings of the 20th International Joint Conference on Artificial Intelligence*, page 1606–1611, Hyderabad, India, Jan 2007. Morgan Kaufmann Publishers. URL `https://dl.acm.org/doi/abs/10.5555/1625275.1625535`.

[75] Gallofré Ocaña, M. and Opdahl, A. L. Challenges and Opportunities for Journalistic Knowledge Platforms. In *CIKMW2020: Proceeding of the CIKM 2020 Workshops*, page 1–9, Galway, Ireland, Oct 2020. CEUR-WS. URL `http://ceur-ws.org/Vol-2699/`.

[76] George, Y., Karunasekera, S., Harwood, A., and Lim, K. H. Real-Time Spatio-Temporal Event Detection on Geotagged Social Media. *Journal of Big Data*, 8(1):1–28, Jun 2021. doi:10.1186/s40537-021-00482-2.

[77] Germann, U., Liepins, R., Gosko, D., and Barzdins, G. SUMMA: Integrating Multiple NLP Technologies into an Open-Source Platform for Multilingual Media Monitoring. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, page 47–51, Melbourne, VIC, Australia, Jul 2018. Association for Computational Linguistics. doi:10.18653/v1/W18-2508.

[78] Germann, U., Liepins, R., Barzdins, G., Gosko, D., Miranda, S., and Nogueira, D. The SUMMA Platform: A Scalable Infrastructure for Multi-lingual Multi-Media Monitoring. In *Proceedings of ACL 2018, System Demonstrations*, page 99–104, Melbourne, VIC, Australia, Jul 2018. Association for Computational Linguistics. doi:10.18653/v1/p18-4017.

[79] Gillani, M., Ilyas, M. U., Saleh, S., Alowibdi, J. S., Aljohani, N., and Alotaibi, F. S. Post Summarization of Microblogs of Sporting Events. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 59–68, Perth, Australia, May 2017. International World Wide Web Conferences Steering Committee. doi:10.1145/3041021.3054146.

[80] Girvan, M. and Newman, M. E. J. Community Structure in Social and Biological Networks. *https://www.pnas.org/content/99/12/7821*, 99(12):7821–7826, Jun 2002. doi:10.1073/pnas.122653799.

[81] Glendenning, B. Brazil 1-7 Germany: World Cup 2014 Semi-Final – As It Happened, Jul 2014. URL `https://www.theguardian.com/football/2014/jul/08/brazil-v-germany-world-cup-2014-semi-final-live-report`. Accessed on September 23, 2021.

[82] GNM Press Office. The Guardian's Politics Live Blog with Andrew Sparrow Celebrates its 2,500th Edition, Jan 2022. URL `https://www.theguardian.com/gnm-press-office/2022/jan/27/the-guardians-politics-live-blog-with-andrew-sparrow-celebrates-its-2500th-edition`. Accessed on Jan 28, 2023.

[83] Golding, N. Tweet Annotations Added to the Tweet Object for the Sampled Stream and Filtered Stream Endpoints in Labs, Dec 2019. URL `https://twittercommunity.com/t/tweet-annotations-added-to-the-tweet-object-for-the-sampled-stream-and-filtered-stream-endpoints-in-labs/132407`. Accessed on July 9, 2022.

[84] Goldstein, J. and Carbonell, J. Summarization: Using MMR for Diversity-Based Reranking and Evaluating Summaries. In *Proceedings of a Workshop held at Baltimore, Maryland*, page 181–195, Baltimore, Maryland, USA, Oct 1998. Association for Computational Linguistics. doi:10.3115/1119089.1119120.

[85] Gottschalk, S. and Demidova, E. EventKG: A Multilingual Event-Centric Temporal Knowledge Graph. In *ESWC 2018: The Semantic Web*, page 272–287, Crete, Greece, Jun 2018. Springer International Publishing. doi:10.1007/978-3-319-93417-4_18.

[86] Gu, H., Xie, X., Lv, Q., Ruan, Y., and Shang, L. ETree: Effective and Efficient Event Modeling for Real-Time Online Social Media Networks . In *WI-IAT '11: Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology - Volume 01*, page 300–307, Lyon, France, Aug 2011. IEEE Computer Society. doi:10.1109/WI-IAT.2011.126.

[87] Guille, A. and Favre, C. Mention-Anomaly-Based Event Detection and Tracking in Twitter. In *Proceedings of the 2014 IEEE/ACM International Conference on Advances in Social Network Analysis and Mining*, page 375–382, Beijing, China, Aug 2014. IEEE. doi:10.1109/ASONAM.2014.6921613.

[88] Guille, A., Favre, C., Hacid, H., and Zighed, D. SONDY: An Open Source Platform for Social Dynamics Mining and Analysis. In *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, page 1005–1008, New York, NY, USA, Jun 2013. Association for Computing Machinery. doi:10.1145/2463676.2463694.

[89] Ha, A. Y. H. and Hyland, K. What is Technicality? A Technicality Analysis Model for EAP Vocabulary. *Journal of English for Academic Purposes*, 28:35–49, Jul 2017. doi:10.1016/j.jeap.2017.06.003.

[90] Hamborg, F., Lachnit, S., Schubotz, M., Hepp, T., and Gipp, B. Giveme5W: Main Event Retrieval from News Articles by Extraction of the Five Journalistic W Questions. In *iConference 2018: Transforming Digital Worlds*, page 356–366, Sheffield, United Kingdom, Mar 2018. Springer International Publishing. ISBN 0302-9743. doi:10.1007/978-3-319-78105-1_39.

[91] Hammad, M. and El-Beltagy, S. R. Towards Efficient Online Topic Detection through Automated Bursty Feature Detection from Arabic Twitter Streams. *Procedia Computer Science*, 117:248–255, 2017. doi:10.1016/j.procs.2017.10.116.

[92] Harris, Z. S. Distributional Structure. *WORD*, 10(2-3):146–162, 1954. doi:10.1080/00437956.1954.11659520.

[93] Hasan, M., Orgun, M. A., and Schwitter, R. Real-Time Event Detection from the Twitter Data Stream Using the TwitterNews+ Framework. *Information Processing & Management*, 56(3):1146–1165, May 2019. doi:10.1016/j.ipm.2018.03.001.

[94] Hettiarachchi, H., Adedoyin-Olowe, M., Bhogal, J., and Gaber, M. M. Embed2Detect: Temporally Clustered Embedded Words for Event Detection in Social Media. *Machine Learning*, 111(1):49–87, May 2021. doi:10.1007/s10994-021-05988-7.

[95] Hossny, A. H. and Mitchell, L. Event Detection in Twitter: A Keyword Volume Approach. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, page 1200–1208, Singapore, Nov 2018. IEEE. doi:10.1109/ICDMW.2018.00172.

[96] Hsieh, L.-C., Lee, C.-W., Chiu, T.-H., and Hsu, W. Live Semantic Sport Highlight Detection Based on Analyzing Tweets of Twitter. In *2012 IEEE International Conference on Multimedia and Expo*, page 949–954, Melbourne, VIC, Australia, Jul 2012. IEEE. ISBN 1945-7871. doi:10.1109/ICME.2012.135.

[97] Hsu, P.-F., Fan, Y.-C., and Chen, H. On Semantic Annotation for Sports Video Highlights by Mining User Comments from Live Broadcast Social Network. In *BWCCA 2018: Advances on Broadband and Wireless Computing, Communication and Applications*, page 367–380, Taichung, Taiwan, Oct 2018. Springer International Publishing. ISBN 2367-4512. doi:10.1007/978-3-030-02613-4_33.

[98] Hu, M., Liu, S., Wei, F., Wu, Y., Stasko, J., and Ma, K.-L. Breaking News on Twitter. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, page 2751–2754, Austin, Texas, USA, May 2012. Association for Computing Machinery. doi:10.1145/2207676.2208672.

[99] Hua, T., Chen, F., Zhao, L., Lu, C.-T., and Ramakrishnan, N. Automatic Targeted-Domain Spatiotemporal Event Detection in Twitter. *GeoInformatica*, 20(4):765–795, Oct 2016. doi:10.1007/s10707-016-0263-0.

[100] Huang, Y., Li, A., Zhou, B., Huang, J., Lan, L., Yin, X., and Jia, Y. Person Entity Attribute Extraction Based on Siamese Network. *IEEE Access*, 7:64506–64516, May 2019. doi:10.1109/ACCESS.2019.2917302.

[101] Huang, Y., Shen, C., and Li, T. Event Summarization for Sports Games using Twitter Streams. *World Wide Web*, 21(3):609–627, May 2018. doi:10.1007/s11280-017-0477-6.

[102] Ifrim, G., Shi, B., and Brigadir, I. Event Detection in Twitter using Aggressive Filtering and Hierarchical Tweet Clustering. In *Proceedings of the SNOW 2014 Data Challenge*, page 33–40, Seoul, Korea, Apr 2014. CEUR. URL `http://ceur-ws.org/Vol-1150/`.

[103] Igo, S. P. and Riloff, E. Corpus-Based Semantic Lexicon Induction with Web-Based Corroboration. In *UMSLLS '09: Proceedings of the Workshop on Unsupervised and Minimally Supervised Learning of Lexical Semantics*, page 18–26, Boulder, Colorado, USA, Jun 2009. Association for Computational Linguistics. URL `https://dl.acm.org/doi/10.5555/1641968.1641971`.

[104] Jia, C., Carson, M. B., Wang, X., and Yu, J. Concept Decompositions for Short Text Clustering by Identifying Word Communities. *Pattern Recognition*, 76:691–703, Apr 2018. doi:10.1016/j.patcog.2017.09.045.

[105] Jucker, A. H. 'Audacious, Brilliant!! What a Strike!' Live Text Commentaries on the Internet as Real-Time Narratives. In *Narrative Revisited: Telling a Story in the Age of New Media*, pages 57–78. Benjamins, Nov 2010. URL `https://www.torrossa.com/en/resources/an/5000861`.

[106] Jurgens, D. Word Sense Induction by Community Detection. In *TextGraphs-6: Proceedings of TextGraphs-6: Graph-based Methods for Natural Language*, page 24–28, Portland, Oregon, USA, Jun 2011. Association for Computational Linguistics. URL `https://dl.acm.org/doi/abs/10.5555/2024277.2024282`.

[107] Justia. Criminal Law Glossary, Apr 2018. URL `https://www.justia.com/criminal/glossary/`. Accessed on July 14, 2021.

[108] Kattenberg, M., Beloki, Z., Soroa, A., Artola, X., Fokkens, A. S., Huijgen, P. E. M., and Verstoep, K. Two Architectures for Parallel Processing of Huge Amounts of Text. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, page 4513–4519, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL `https://aclanthology.org/L16-1714/`.

[109] Keogh, E., Lonardi, S., and Ratanamahatana, C. Towards Parameter-Free Data Mining. In *KDD '04: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Dining*, pages 206–215, Seattle, WA, USA, Aug 2004. Association for Computing Machinery. doi:10.1145/1014052.1014077.

[110] Khondker, H. H. Role of the New Media in the Arab Spring. *Globalizations*, 8(5):675–679, Oct 2011. doi:10.1080/14747731.2011.621287.

[111] Kit, C. and Liu, X. Measuring Mono-Word Termhood by Rank Difference via Corpus Comparison. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 14(2): 204–229, 2008. doi:10.1075/term.14.2.05kit.

[112] Klapaftis, J. P. and Manandhar, S. Word Sense Induction Using Graphs of Collocations. In *Proceedings of the 2008 Conference on ECAI 2008: 18th European Conference on Artificial Intelligence*, page 298–302, Patras, Greece, Jul 2008. IOS Press. URL `https://dl.acm.org/doi/abs/10.5555/1567281.1567349`.

[113] Kleinberg, J. Bursty and Hierarchical Structure in Streams. *Data Mining and Knowledge Discovery*, 7: 373–397, Oct 2003. doi:10.1023/A:1024940629314.

[114] Kolajo, T., Daramola, O., and Adebiyi, A. A. Real-Time Event Detection in Social Media Streams Through Semantic Analysis of Noisy Terms. *Journal of Big Data*, 9(1):1–36, Jul 2022. doi:10.1186/s40537-022-00642-y.

[115] Kozakov, L., Park, Y., Fin, T.-H., Drissi, Y., Doganata, Y., and Cofino, T. Glossary Extraction and Utilization in the Information Search and Delivery System for IBM Technical Support. *IBM Systems Journal*, 43(3):546–563, Sep 2004. doi:10.1147/sj.433.0546.

[116] Kozareva, Z., Riloff, E., and Hovy, E. Semantic Class Learning from the Web with Hyponym Pattern Linkage Graphs. In *Proceedings of ACL-08: HLT*, page 1048–1056, Columbus, Ohio, USA, Jun 2008. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/P08-1119/`.

[117] Krauthammer, M. and Nenadic, G. Term Identification in the Biomedical Literature. *Journal of Biomedical Informatics*, 37(6):512–526, Dec 2004. doi:10.1016/j.jbi.2004.08.004.

[118] Kubo, M., Sasano, R., Takamura, H., and Okumura, M. Generating Live Sports Updates from Twitter by Finding Good Reporters. In *Proceedings of the 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, volume 1, pages 527–534, Atlanta, Georgia, USA, Nov 2013. IEEE Computer Society. doi:10.1109/WI-IAT.2013.74.

[119] Kumar, S., Liu, H., Mehta, S., and Subramaniam, L. V. Exploring a Scalable Solution to Identifying Events in Noisy Twitter Streams. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, page 496–499, Paris, France, Aug 2015. IEEE. doi:10.1145/2808797.2809389.

[120] Kumaran, G. and Allan, J. Text Classification and Named Entities for New Event Detection. In *SIGIR '04: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 297–304, Sheffield, South Yorkshire, United Kingdoms, Jul 2004. ACM. doi:10.1145/1008992.1009044.

[121] Lanagan, J. and Smeaton, A. F. Using Twitter to Detect and Tag Important Events in Live Sports. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, page 542–545, Barcelona, Spain, Jul 2011. Association for the Advancement of Artificial Intelligence. doi:10.1609/icwsm.v5i1.14170.

[122] Lappas, T., Arai, B., Platakis, M., Kotsakos, D., and Gunopulos, D. On Burstiness-Aware Search for Document Sequences. In *KDD '09: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 477–486, Paris, France, Jun 2009. Association for Computing Machinery. doi:10.1145/1557019.1557075.

[123] Leban, G., Fortuna, B., Brank, J., and Grobelnik, M. Event Registry: Learning about World Events from News. In *WWW '14 Companion: Proceedings of the 23rd International Conference on World Wide Web*, page 107–110, Seoul, South Korea, Apr 2014. Association for Computing Machinery. doi:10.1145/2567948.2577024.

[124] Lee, S., Lee, S., Kim, K., and Park, J. Bursty Event Detection from Text Streams for Disaster Management. In *WWW '12 Companion: Proceedings of the 21st International Conference on World Wide Web*, page 679–682, Lyon, France, Apr 2012. ACM. doi:10.1145/2187980.2188179.

[125] Li, B., Li, W., Lu, Q., and Wu, M. Profile-Based Event Tracking. In *SIGIR '05: Proceedings of the 28th ACM/SIGIR International Symposium on Information Retrieval 2005*, pages 631–632, Salvador, Brazil, Aug 2005. ACM. doi:10.1145/1076034.1076163.

[126] Li, Q., Nourbakhsh, A., Shah, S., and Liu, X. Real-Time Novel Event Detection from Social Media. In *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*, page 1129–1139, San Diego, CA, USA, Apr 2017. IEEE. doi:10.1109/ICDE.2017.157.

[127] Li, S., Li, J., Song, T., Li, W., and Chang, B. A Novel Topic Model for Automatic Term Extraction. In *SIGIR '13: Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 885–888, Dublin, Ireland, Jul 2013. ACM. doi:10.1145/2484028.2484106.

[128] Liu, C., Xu, R., and Gui, L. Burst Events Detection on Microblogging. In *Proceedings of the 2013 International Conference on Machine Learning and Cybernetics*, page 1921–1924, Tianjin, China, Jul 2013. IEEE. doi:10.1109/ICMLC.2013.6890909.

[129] Liu, X., Li, Q., Nourbakhsh, A., Fang, R., Thomas, M., Anderson, K., Kociuba, R., Vedder, M., Pomerville, S., Wudali, R., Martin, R., Duprey, J., Vachher, A., Keenan, W., and Shah, S. Reuters Tracer: A Large Scale System of Detecting & Verifying Real-Time News Events from Twitter. In *CIKM '16: Proceedings of the 25th ACM International Conference on Information and Knowledge Management*, page 207–216, Indianapolis, IN, USA, Oct 2016. Association for Computing Machinery. doi:10.1145/2983323.2983363.

[130] Liu, X., Nourbakhsh, A., Li, Q., Shah, S., Martin, R., and Duprey, J. Reuters Tracer: Toward Automated News Production Using Large Scale Social Media Data. In *2017 IEEE International Conference on Big Data (Big Data)*, page 1483–1493, Boston, MA, USA, Dec 2017. IEEE. doi:10.1109/BigData.2017.8258082.

[131] Lopes, L., Fernandes, P., and Vieira, R. Estimating Term Domain Relevance Through Term Frequency, Disjoint Corpora Frequency - TF-DCF. *Knowledge-Based Systems*, 97:237–249, Apr 2016. doi:10.1016/j.knosys.2015.12.015.

[132] Lopes, L., Fernandes, P., and Vieira, R. ExATO - High Quality Term Extraction for Portuguese and English. In *Proceedings of the 2016 IEEE/WIC/ACM International Conference on Web Intelligence*, page 540–545, Omaha, Nebraska, USA, Oct 2016. IEEE. doi:10.1109/WI.2016.0092.

[133] Lopreite, M., Panzarasa, P., Puliga, M., and Riccaboni, M. Early Warnings of COVID-19 Outbreaks across Europe from Social Media. *Scientific Reports*, 11:1–7, Jan 2021. doi:10.1038/s41598-021-81333-1.

[134] Lowrance, S. Was the Revolution Tweeted? Social Media and the Jasmine Revolution in Tunisia. *Domes*, 25(1):155–176, Feb 2016. doi:10.1111/dome.12076.

[135] Luo, Z., Wang, H., and Xie, R. Extract Domain Terminologies for Knowledge Graph Construction Using Domain Feature Vectors. In *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, page 53–57, Beijing, China, Mar 2017. IEEE. doi:10.1109/ICBDA.2017.8078715.

[136] Löchtefeld, M., Jäckel, C., and Krüger, A. TwitSoccer: Knowledge-Based Crowd-Sourcing of Live Soccer Events. In *MUM '15: Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia*, pages 148–151, Linz, Austria, Nov 2015. ACM. doi:10.1145/2836041.2836055.

[137] Madani, A., Boussaid, O., and Zegour, D. E. What's Happening: A Survey of Tweets Event Detection . In *INNOV 2014 : Proceedings of the Third International Conference on Communications, Computation, Networks and Technologies*, page 16–22, Nice, France, Oct 2014. IARIA. doi:10.1007/s10723-019-09482-2.

[138] Madani, A., Boussaid, O., and Zegour, D. Real-Time Trending Topics Detection and Description from Twitter Content. *Social Network Analysis and Mining*, 5(1):1–13, Dec 2015. doi:10.1007/s13278-015-0298-5.

[139] Makkonen, J., Ahonen-Myka, H., and Salmenkivi, M. Simple Semantics in Topic Detection and Tracking. *Information Retrieval*, 7(3):347–368, Sep 2004. doi:10.1023/B:INRT.0000011210.12953.86.

[140] Maldonado, A. and Lewis, D. Self-Tuning Ongoing Terminology Extraction Retrained on Terminology Validation Decisions. In *Proceedings of the 12th International Conference on Terminology and Knowledge Engineering*, page 91–100, Copenhagen, Denmark, Jun 2016. Copenhagen Business School. URL `http://hdl.handle.net/2262/82537`.

[141] Mamo, N. *ELD: Event TimeLine Detection — A Participant-Based Approach to Tracking Events*. MSc dissertation, Department of Artificial Intelligence, Faculty of Information & Communication Technology, University of Malta, Oct 2019. URL `https://hydi.um.edu.mt/primo-explore/fulldisplay?docid=OARatUoM123456789%2F52983&vid=356MALT_VU1`.

[142] Mamo, N. and Azzopardi, J. FIRE: Finding Important News REports. In Szymański, J. and Velegrakis, Y., editors, *Semantic Keyword-Based Search on Structured Data Sources*, pages 20–31, Gdansk, Poland, Sep 2017. Springer International Publishing. doi:10.1007/978-3-319-74497-1_3.

[143] Mamo, N., Azzopardi, J., and Layfield, C. ELD: Event TimeLine Detection - A Participant-Based Approach to Tracking Events. In *HT '19: Proceedings of the 30th ACM Conference on Hypertext and Social Media*, pages 267–268, Hof, Germany, Sep 2019. ACM. doi:10.1145/3342220.3344921.

[144] Mamo, N., Azzopardi, J., and Layfield, C. An Automatic Participant Detection Framework for Event Tracking on Twitter. *Algorithms*, 14(3):92, Mar 2021. doi:10.3390/a14030092.

[145] Mamo, N., Azzopardi, J., and Layfield, C. Who? What? Event Tracking Needs Event Understanding. In *Proceedings of the 13th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management - (Volume 1)*, page 139–146, Remote, Oct 2021. SciTePress. doi:10.5220/0010639600003064.

[146] Mamo, N., Azzopardi, J., and Layfield, C. Fine-grained Topic Detection and Tracking on Twitter. In *Proceedings of the 13th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management - (Volume 1)*, page 79–86, Remote, Oct 2021. SciTePress. doi:10.5220/0010639600003064.

[147] Mamo, N., Azzopardi, J., and Layfield, C. The Myth of Reproducibility: A Review of Event Tracking Evaluations on Twitter. *Frontiers in Big Data*, 6:1067335, Apr 5, 2023. URL `https://www.frontiersin.org/articles/10.3389/fdata.2023.1067335/full`.

[148] Mamo, N., Layfield, C., and Azzopardi, J. From Event Tracking to Event Modelling: Understanding as a Paradigm Shift. In Fred, A., Aveiro, D., Dietz, J., Bernardino, J., Masciari, E., and Filipe, J., editors, *Knowledge Discovery, Knowledge Engineering and Knowledge Management*, volume 1718, page 21–36. Springer Cham, Jul 2023. URL `https://link.springer.com/chapter/10.1007/978-3-031-35924-8_2`.

[149] Marconi, F. *Newsmakers: Artificial Intelligence and the Future of Journalism*. Columbia University Press, New York, NY, USA, Apr 2020. ISBN 9780231191364. URL `http://cup.columbia.edu/book/newsmakers/9780231191371`.

[150] Marcus, A., Bernstein, M., Badar, O., Karger, D., Madden, S., and Miller, R. Twitinfo: Aggregating and Visualizing Microblogs for Event Exploration. In *CHI '11: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, page 227–236, Vancouver, BC, Canada., May 2011. Association for Computing Machinery. doi:10.1145/1978942.1978975.

[151] Marino, G. Parma-Milan 1-3: Pioli Passa al Tardini, Espulso Ibrahimovic, Oct 2021. URL `https://www.goal.com/it/partita/parma-vs-milan/c00f9sqimvmrnkmz5n8xknlsk`. Accessed on March 13, 2022.

[152] Martin-Dancausa, C. and Göker, A. Real-time Topic Detection with Bursty N-Grams. In *Proceedings of the SNOW 2014 Data Challenge co-located with 23rd International World Wide Web Conference (WWW 2014)*, page 9–16, Seoul, Korea, Apr 2014. CEUR. URL `https://ceur-ws.org/Vol-1150`.

[153] Marujo, L., Ling, W., Trancoso, I., Dyer, C., Black, A. W., Gershman, A., de Matos, D. M., Neto, J. P., and Carbonell, J. Automatic Keyword Extraction on Twitter. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers)*, pages 637–643, Beijing, China, Jul 2015. Association for Computational Linguistics. doi:10.3115/v1/P15-2105.

[154] Matsuo, Y., Sakaki, T., Uchiyama, K., and Ishizuka, M. Graph-based Word Clustering using a Web Search Engine. In *EMNLP '06: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, page 542–550, Sydney, Australia, Jul 2006. Association for Computational Linguistics. doi:10.3115/1610075.1610150.

[155] Maynard, D. and Ananiadou, S. Identifying Terms by their Family and Friends. In *COLING '00: Proceedings of the 18th Conference on Computational Linguistics - Volume 1*, page 530–536, Saarbrücken, Germany, Jul 2000. Association for Computational Linguistics. doi:10.3115/990820.990897.

[156] McIntosh, T. and Curran, J. R. Weighted Mutual Exclusion Bootstrapping for Domain Independent Lexicon and Template Acquisition. In *Proceedings of the Australasian Language Technology Association Workshop 2008*, pages 97–105, Hobart, Australia, Dec 2008. URL `https://www.aclweb.org/anthology/U08-1013/`.

[157] McMinn, A., Moshfeghi, Y., and Jose, J. Building a Large-Scale Corpus for Evaluating Event Detection on Twitter. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, page 409–418, San Francisco, CA, USA, Oct 2013. Association for Computing Machinery. doi:10.1145/2505515.2505695.

[158] McMinn, A. J. and Jose, J. M. Real-Time Entity-Based Event Detection for Twitter. In Mothe, J., Savoy, J., Kamps, J., Pinel-Sa, Pinel-Sauvagnat, K., Jones, G., San Juan, E., Capellato, L., and Nicola, F., editors, *CLEF 2015: Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 65–77, Toulouse, France, Sep 2015. Springer International Publishing. doi:10.1007/978-3-319-24027-5_6.

[159] Medelyan, O., Witten, I. H., Divoli, A., and Broekstra, J. Automatic Construction of Lexicons, Taxonomies, Ontologies, and Other Knowledge Structures. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 3(4):257–279, Jul 2013. doi:10.1002/widm.1097.

[160] Meijer, K., Frasincar, F., and Hogenboom, F. A Semantic Approach for Extracting Domain Taxonomies from Text. *Decision Support Systems*, 62:78–93, Jun 2014. doi:10.1016/j.dss.2014.03.006.

[161] Meladianos, P., Nikolentzos, G., Rousseau, F., Stavrakas, Y., and Vazirgiannis, M. Degeneracy-Based Real-Time Sub-Event Detection in Twitter Stream. In *Proceedings of the Ninth International AAAI Conference on Web and Social Media*, page 248–257, Oxford, United Kingdom, May 2015. The AAAI Press. doi:10.13140/RG.2.1.1908.8803.

[162] Meladianos, P., Xypolopoulos, C., Nikolentzos, G., and Vazirgiannis, M. An Optimization Approach for Sub-Event Detection and Summarization in Twitter. In *Advances in Information Retrieval*, page 481–493, Grenoble, France, Mar 2018. Springer International Publishing. ISBN 0302-9743. doi:10.1007/978-3-319-76941-7_36.

[163] Mele, I., Bahrainian, S. A., and Crestani, F. Event Mining and Timeliness Analysis from Heterogeneous News Streams. *Information Processing & Management*, 56(3):969–993, May 2019. doi:10.1016/j.ipm.2019.02.003.

[164] Mifsud, M., Layfield, C., Azzopardi, J., and Abela, J. "To Trust a LIAR": Does Machine Learning Really Classify Fine-grained, Fake News Statements? In *Proceedings of the Second Workshop on Online Misinformation- and Harm-Aware Recommender Systems*, page 1–13, Remote, Amsterdam, The Netherlands, Oct 2021. CEUR. URL `http://ceur-ws.org/Vol-3012/`.

[165] Miroshnichenko, A. AI to Bypass Creativity. Will Robots Replace Journalists? (The Answer Is "Yes"). *Information*, 9(7):183, Jul 2018. doi:10.3390/info9070183.

[166] Mishra, S. and Diesner, J. Semi-Supervised Named Entity Recognition in Noisy-Text. In *Proceedings of the 2nd Workshop on Noisy User-generated Text*, page 203–212, Osaka, Japan, Dec 2016. The COLING 2016 Organizing Committee. URL `https://www.aclweb.org/anthology/W16-3927/`.

[167] Mitchell, M. Artificial Intelligence Hits the Barrier of Meaning. *Information*, 10(2):51, Feb 2019. doi:10.3390/info10020051.

[168] Mitchell, M. *Artificial Intelligence: A Guide for Thinking Humans*. Farrar, Straus and Giroux, New York, NY, USA, 1 edition, Oct 2019. URL `https://us.macmillan.com/books/9780374257835`.

[169] Mohd, M. Named Entity Patterns Across News Domains. In *Proceedings of the BCS IRSG Symposium: Future Directions in Information Access 2007*, pages 1–6, Glasgow, Scotland, Aug 2007. BCS, The Chartered Institute for IT. URL `https://dl.acm.org/doi/10.5555/2227895.2227901`.

[170] Murray, S. Scotland 0-2 Czech Republic: Euro 2020 - As It Happened, Jun 2021. URL `https://www.theguardian.com/football/live/2021/jun/14/scotland-v-czech-republic-euro-2020-live`. Accessed on January 24, 2022.

[171] Murray, S. Wales 1-1 Switzerland: Euro 2020 - As It Happened, Jun 2021. URL `https://www.theguardian.com/football/live/2021/jun/12/wales-v-switzerland-euro-2020-live-score-updates`. Accessed on January 24, 2022.

[172] Murray, S. Turkey 0-3 Italy: Euro 2020 Opener -As It Happened, Jun 2021. URL `https://www.theguardian.com/football/live/2021/jun/11/turkey-v-italy-euro-2020-opening-ceremony-first-game-live-scores-updates`. Accessed on January 24, 2022.

[173] Mykowiecka, A., Marciniak, M., and Rychlik, P. Recognition of Non-Domain Phrases in Automatically Extracted Lists of Terms. In *Proceedings of the 5th International Workshop on Computational Terminology (Computerm2016)*, page 12–20, Osaka, Japan, Dec 2016. The COLING 2016 Organizing Committee. URL https://www.aclweb.org/anthology/W16-4703/.

[174] Nakagawa, H. and Mori, T. A Simple but Powerful Automatic Term Extraction Method. In *COMPUTERM '02: COLING-02 on COMPUTERM 2002: Second International Workshop on Computational Terminology - Volume 14*, pages 1–7. Association for Computational Linguistics, Aug 2002. doi:10.3115/1118771.1118778.

[175] Navigli, R. and Velardi, P. Learning Domain Ontologies from Document Warehouses and Dedicated Web Sites. *Computational Linguistics*, 30(2):151–179, Jun 2004. doi:10.1162/089120104323093276.

[176] Nazar, R. A Statistical Approach to Term Extraction. *International Journal of English Studies*, 11(2):159, 2011. doi:10.6018/ijes/2011/2/149691.

[177] Nebhi, K. Ontology-Based Information Extraction from Twitter. In *Proceedings of the Workshop on Information Extraction and Entity Analytics on Social Media Data*, pages 17–22, Mumbai, India, Dec 2012. The COLING 2012 Organizing Committee. URL https://aclanthology.org/W12-5502/.

[178] Newman, M. E. J. and Girvan, M. Finding and Evaluating Community Structure in Networks. *Physical Review E*, 69(2):026113, Feb 2004. URL https://journals.aps.org/pre/abstract/10.1103/PhysRevE.69.026113.

[179] Newman, N. Journalism, Media, and Technology Trends and Predictions 2021. Technical report, Reuters Institute for the Study of Journalism, Jan 2021. URL https://reutersinstitute.politics.ox.ac.uk/journalism-media-and-technology-trends-and-predictions-2021.

[180] Newman, N. Journalism, Media, and Technology Trends and Predictions 2022. Technical report, Reuters Institute for the Study of Journalism, Jan 2022. URL https://reutersinstitute.politics.ox.ac.uk/journalism-media-and-technology-trends-and-predictions-2022.

[181] Newman, N. Journalism, Media, and Technology Trends and Predictions 2023. Technical report, Reuters Institute for the Study of Journalism, Jan 2023. URL https://reutersinstitute.politics.ox.ac.uk/journalism-media-and-technology-trends-and-predictions-2023.

[182] Ni, X., Ni, X., Quan, X., Quan, X., Lu, Z., Lu, Z., Wenyin, L., Wenyin, L., Hua, B., and Hua, B. Short Text Clustering by Finding Core Terms. *Knowledge and Information Systems*, 27(3):345–365, Jun 2011. URL https://link.springer.com/article/10.1007/s10115-010-0299-7.

[183] Nichols, J., Mahmud, J., and Drews, C. Summarizing Sporting Events Using Twitter. In *Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces*, pages 189–198, Lisbon, Portugal, Feb 2012. ACM. URL https://dl.acm.org/doi/10.1145/2166966.2166999.

[184] Nolasco, D. and Oliveira, J. Intelligent Subevent Detection Based on Social Network Data. In *2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech)*, page 820–827, Orlando, FL, USA, Nov 2017. IEEE. URL https://ieeexplore.ieee.org/document/8328483.

[185] Nugumanova, A., Bessmertny, I., Baiburin, Y., and Mansurova, M. A New Operationalization of Contrastive Term Extraction Approach Based on Recognition of Both Representative and Specific Terms. In *Knowledge Engineering and Semantic Web*, page 103–118, Prague, Czech Republic, Sep 2016. Springer, Cham. doi:10.1007/978-3-319-45880-9_9.

[186] Nutakki, G. C., Nasraoui, O., Abdollahi, B., Badami, M., and Sun, W. Distributed LDA Based Topic Modeling and Topic Agglomeration in a Latent Space. In *Proceedings of the SNOW 2014 Data Challenge*, page 17–24, Seoul, Korea, Apr 2014. CEUR. doi:10.13140/2.1.4606.6889.

[187] Olteanu, A., Castillo, C., Diaz, F., and Vieweg, S. CrisisLex: A Lexicon for Collecting and Filtering Microblogged Communications in Crises. In *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*, page 376–385, Ann Arbor, MI, USA, Jun 2014. Association for the Advancement of Artificial Intelligence. doi:10.1609/icwsm.v8i1.14538.

[188] Opdahl, A. L. and Tessem, B. Ontologies for Finding Journalistic Angles. *Software and Systems Modeling*, 20(1):71–87, Jun 2020. doi:10.1007/s10270-020-00801-w.

[189] Ozdikis, O., Senkul, P., and Oguztuzun, H. Semantic Expansion of Tweet Contents for Enhanced Event Detection in Twitter. In *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, page 20–24, Istanbul, Turkey, Aug 2012. IEEE. doi:10.1109/ASONAM.2012.14.

[190] Paikens, P., Barzdins, G., Mendes, A., Ferreira, D., Broscheit, S., Almeida, M. S. C., Miranda, S., Nogueira, D., Balage, P., and Martins, A. F. T. SUMMA at TAC Knowledge Base Population Task 2016. In *Proceedings of the Ninth Text Analysis Conference (TAC 2016)*, page 1–9, Gaithersburg, MD, USA, Nov 2016. Zenodo. doi:10.5281/zenodo.827316.

[191] Panagiotou, N., Katakis, I., and Gunopulos, D. *Detecting Events in Online Social Networks: Definitions, Trends and Challenges*, volume 9580 of *Lecture Notes in Computer Science*. Springer International Publishing, Cham, Jul 2016. ISBN 3319417053. doi:10.1007/978-3-319-41706-6_2.

[192] Pantel, P. and Lin, D. Discovering Word Senses from Text. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 613–619, Edmonton, Alberta, Canada, Jul 2002. Association for Computing Machinery. doi:10.1145/775047.775138.

[193] Papadopoulos, S., Corney, D., and Aiello, L. SNOW 2014 Data Challenge: Assessing the Performance of News Topic Detection Methods in Social Media. In *Proceedings of the SNOW 2014 Data Challenge co-located with 23rd International World Wide Web Conference (WWW 2014)*, page 1–8, Seoul, South Korea, Apr 2014. CEUR. URL `http://ceur-ws.org/Vol-1150/`.

[194] Park, Y., Byrd, R. J., and Boguraev, B. K. Automatic Glossary Extraction: Beyond Terminology Identification. In *COLING '02: Proceedings of the 19th International Conference on Computational linguistics - Volume 1*, page 1–7, Taipei, Taiwan, Aug 2002. Association for Computational Linguistics. doi:10.3115/1072228.1072370.

[195] Park, Y., Patwardhan, S., Visweswariah, K., and Gates, S. C. An Empirical Analysis of Word Error Rate and Keyword Error Rate. In *INTERSPEECH 2008: Proceedings of the 9th Annual Conference of the International Speech Communication Association*, pages 2070–2073, Brisbane, Australia, Sep 2008. URL `https://www.isca-speech.org/archive_v0/interspeech_2008/i08_2070.html`.

[196] Pazienza, M., Pennacchiotti, M., and Zanzotto, F. Terminology Extraction: An Analysis of Linguistic and Statistical Approaches. In *Knowledge Mining*, volume 185, pages 255–279. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005. ISBN 3540250700. doi:10.1007/3-540-32394-5_20.

[197] Petkos, G., Papadopoulos, S., and Kompatsiaris, Y. Two-Level Message Clustering for Topic Detection in Twitter. In *Proceedings of the SNOW 2014 Data Challenge*, page 49–56, Seoul, Korea, Apr 2014. CEUR. URL https://ceur-ws.org/Vol-1150.

[198] Petroni, F., Raman, N., Nugent, T., Nourbakhsh, A., Panić, Z., Shah, S., and Leidner, J. An Extensible Event Extraction System With Cross-Media Event Resolution. In *KDD '18: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, page 626–635, London, United Kingdom, Aug 2018. Association for Computing Machinery. doi:10.1145/3219819.3219827.

[199] Petrović, S., Osborne, M., and Lavrenko, V. Streaming First Story Detection with Application to Twitter. In *HLT '10: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, page 181–189, Los Angeles, CA, United States of America, Jun 2010. Association for Computational Linguistics. URL https://dl.acm.org/doi/abs/10.5555/1857999.1858020.

[200] Petrović, S., Osborne, M., and Lavrenko, V. Using Paraphrases for Improving First Story Detection in News and Twitter. In *NAACL HLT '12: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 338–.346, Montreal, Canada, Jun 2012. Association for Computational Linguistics. URL https://dl.acm.org/doi/abs/10.5555/2382029.2382072.

[201] Petrović, S., Osborne, M., McCreadie, R., Macdonald, C., Ounis, I., and Shrimpton, L. Can Twitter Replace Newswire for Breaking News? In *Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media*, page 713–716, Cambridge, MA, United States, Jul 2013. Association for the Advancement of Artificial Intelligence. doi:10.1609/icwsm.v7i1.14450.

[202] Peñas, A., Verdejo, F., and Gonzalo, J. Corpus-Based Terminology Extraction Applied to Information Access. In *Proceedings of the Corpus Linguistics 2001*, pages 458–465, Lancaster, United Kingdom, Mar 2001. URL http://ucrel.lancs.ac.uk/publications/CL2003/CL2001%20conference/contents.htm.

[203] Phipps, C. and Rawlinson, K. Paris Attacks Kill More Than 120 People – As It Happened, Nov 2015. URL https://www.theguardian.com/world/live/2015/nov/13/shootings-reported-in-eastern-paris-live. Accessed on November 19, 2021.

[204] Phuvipadawat, S. and Murata, T. Breaking News Detection and Tracking in Twitter. In *Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, page 120–123, Toronto, ON, Canada, Aug 2010. IEEE Computer Society. doi:10.1109/WI-IAT.2010.205.

[205] Popescu, A.-M., Pennacchiotti, M., and Paranjpe, D. Extracting Events and Event Descriptions from Twitter. In *Proceedings of the 20th International Conference Companion on World Wide Web*, pages 105–106, Hyderabad, India, Mar 2011. ACM. doi:10.1145/1963192.1963246.

[206] Pradhan, A. K., Mohanty, H., and Lal, R. P. Event Detection and Aspects in Twitter: A BoW Approach. In *ICDCIT 2019: Proceedings of the 15th International Conference on Distributed Computing and Internet Technology*, page 194–211, Bhubaneswar, India, Jan 2019. Springer International Publishing. ISBN 1611-3349. doi:10.1007/978-3-030-05366-6_16.

[207] Preoţiuc-Pietro, D., Srijith, P. K., Hepple, M., and Cohn, T. Studying the Temporal Dynamics of Word Co-Occurrences: An Application to Event Detection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, page 4380–4387, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL `https://aclanthology.org/L16-1694/`.

[208] Qadir, A., Mendes, P. N., Gruhl, D., and Lewis, N. Semantic Lexicon Induction from Twitter with Pattern Relatedness and Flexible Term Length. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence 2432*, page 2432–2439, Austin, Texas, USA, Jan 2015. Association for the Advancement of Artificial Intelligence. doi:10.1609/aaai.v29i1.9519.

[209] Qi, Z., Liu, K., and Zhao, J. Choosing Better Seeds for Entity Set Expansion by Leveraging Wikipedia Semantic Knowledge. In *Pattern Recognition*, page 655–662, Beijing, China, Sep 2012. Springer. ISBN 1865-0929. doi:10.1007/978-3-642-33506-8_80.

[210] Qureshi, M. A., O'Riordan, C., and Pasi, G. Short-Text Domain Specific Key Terms/Phrases Extraction Using an n-gram Model with Wikipedia. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, page 2515–2518, Maui, Hawaii, USA, Oct 2012. Association for Computing Machinery. doi:10.1145/2396761.2398680.

[211] Raff, E. A Step Toward Quantifying Independently Reproducible Machine Learning Research. In *NeurIPS 2019: Advances in Neural Information Processing Systems 32*, page 5462–5472, Vancouver, BC, Canada, Dec 2019. Neural Information Processing Systems Foundation, Inc. (NIPS). URL `https://proceedings.neurips.cc/paper/2019/hash/c429429bf1f2af051f2021dc92a8ebea-Abstract.html`.

[212] Rayson, P. and Garside, R. Comparing Corpora Using Frequency Profiling. In *WCC '00: Proceedings of the Workshop on Comparing Corpora - Volume 9*, page 1–6, Hong Kong, Oct 2000. Association for Computational Linguistics. doi:10.3115/1117729.1117730.

[213] Reed, J. W., Jiao, Y., Potok, T. E., Klump, B. A., Elmore, M. T., and Hurson, A. R. TF-ICF: A New Term Weighting Scheme for Clustering Dynamic Data Streams. In *2006 5th International Conference on Machine Learning and Applications (ICMLA'06)*, pages 258–263, Orlando, Florida, USA, Dec 2006. IEEE. doi:10.1109/ICMLA.2006.50.

[214] Riloff, E. and Jones, R. Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, pages 474–479, Orlando, Florida, USA, Jul 1999. American Association for Artificial Intelligence. URL `https://www.aaai.org/Library/AAAI/1999/aaai99-068.php`.

[215] Riloff, E. and Shepherd, J. A Corpus-Based Approach for Building Semantic Lexicons. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 117–124, Providence, Rhode Island, USA, Aug 1997. URL `https://www.aclweb.org/anthology/W97-0313/`.

[216] Rincón, J. Athletic Club - Real Sociedad: Resumen, Resultado y Goles - Final Copa del Rey 2020, Apr 2021. URL https://www.marca.com/futbol/copa-del-rey/athletic-real-sociedad-directo/2021/04/03/01_0105_20210403_174_188.html. Accessed on March 13, 2022.

[217] Rinehart, A. and Kung, E. Artificial Intelligence in Local News - A Survey of US Newsrooms' AI Readiness. Technical report, The Associated Press, Mar 2022. URL https://www.ap.org/assets/files/ap_local_news_ai_report_march_2022.pdf.

[218] Ritter, A., Mausam, Etzioni, O., and Clark, S. Open Domain Event Extraction from Twitter. In *KDD '12: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 1104–1112, Beijing, China, Aug 2012. Association for Computing Machinery. doi:10.1145/2339530.2339704.

[219] Roark, B. and Charniak, E. Noun-Phrase Co-Occurrence Statistics for Semiautomatic Semantic Lexicon Construction. In *ACL '98/COLING '98: Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 2*, pages 1110–1116, Montreal, Quebec, Canada, Aug 1998. Association for Computational Linguistics. doi:10.3115/980691.980751.

[220] Rogers, S. The Boston Bombing: How Journalists Used Twitter to Tell the Story, Jul 2013. URL https://blog.twitter.com/en_us/a/2013/the-boston-bombing-how-journalists-used-twitter-to-tell-the-story. Accessed on September 22, 2021.

[221] Rospocher, M., van Erp, M., Vossen, P., Fokkens, A., Aldabe, I., Rigau, G., Soroa, A., Ploeger, T., and Bogaard, T. Building Event-Centric Knowledge Graphs from News. *Journal of Web Semantics*, 37–38: 132–151, Mar 2016. doi:10.1016/j.websem.2015.12.004.

[222] Rudnik, C., Ehrhart, T., Ferret, O., Teyssou, D., Troncy, R., and Tannier, X. Searching News Articles Using an Event Knowledge Graph Leveraged by Wikidata. In *WWW '19: Companion Proceedings of The 2019 World Wide Web Conference*, page 1232–1239, San Francisco, CA, USA, May 2019. Association for Computing Machinery. doi:10.1145/3308560.3316761.

[223] Rudra, K., Ghosh, S., Ganguly, N., Goyal, P., and Ghosh, S. Extracting Situational Information from Microblogs during Disaster Events. In *Proceedings of the 24th ACM International on conference on information and knowledge management*, pages 583–592, Melbourne, Australia, Oct 2015. ACM. doi:10.1145/2806416.2806485.

[224] Ruzzo, W. L. and Tompa, M. A Linear Time Algorithm for Finding All Maximal Scoring Subsequences. In *Proceedings of the 7th International Conference on Intelligent Systems for Molecular Biology*, page 234–241, Heidelberg, Germany, Aug 1999. American Association for Artificial Intelligence. ISBN 1553-0833. URL https://www.aaai.org/Library/ISMB/1999/ismb99-027.php.

[225] Saeed, Z., Abbasi, R. A., Maqbool, O., Sadaf, A., Razzak, I., Daud, A., Aljohani, N. R., and Xu, G. What's Happening Around the World? A Survey and Framework on Event Detection Techniques on Twitter. *Journal of Grid Computing*, 17(2):279–312, May 2019. doi:10.1007/s10723-019-09482-2.

[226] Saeed, Z., Abbasi, R. A., Razzak, I., Maqbool, O., Sadaf, A., and Xu, G. Enhanced Heartbeat Graph for Emerging Event Detection on Twitter using Time Series Networks. *Expert Systems with Applications*, 136:115–132, Dec 2019. doi:10.1016/j.eswa.2019.06.005.

[227] Sakaki, T., Okazaki, M., and Matsuo, Y. Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors. In *WWW '10: Proceedings of the 19th International Conference on World Wide Web*, page 851–860, Raleigh, North Carolina, USA, Apr 2010. Association for Computing Machinery. doi:10.1145/1772690.1772777.

[228] Samant, S. S., Bhanu Murthy, N. L., and Malapati, A. Improving Term Weighting Schemes for Short Text Classification in Vector Space Model. *IEEE Access*, 7:166578–166592, 2019. doi:10.1109/ACCESS.2019.2953918.

[229] Sankaranarayanan, J., Samet, H., Teitler, B., Lieberman, M., and Sperling, J. TwitterStand: News in Tweets . In *GIS '09: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, page 42–51, Seattle, WA, USA, Nov 2009. Association for Computing Machinery. doi:10.1145/1653771.1653781.

[230] Saraf, P. and Ramakrishnan, N. EMBERS AutoGSR: Automated Coding of Civil Unrest Events. In *KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 599–608, San Francisco, CA, USA, Aug 2016. Association for Computing Machinery. doi:10.1145/2939672.2939737.

[231] Schmitt, X., Kubler, S., Robert, J., Papadakis, M., and LeTraon, Y. A Replicable Comparison Study of NER Software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate. In *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, page 338–343, Granada, Spain, Oct 2019. IEEE. doi:10.1109/SNAMS.2019.8931850.

[232] Schäfer, J., Rösiger, I., Heid, U., and Dorna, M. Evaluating Noise Reduction Strategies for Terminology Extraction. In *Proceedings of the 11th International Conference on Terminology and Artificial Intelligence*, page 123–131, Granada, Spain, Nov 2015. CEUR. URL `http://ceur-ws.org/Vol-1495/`.

[233] Shamma, D., Kennedy, L., and Churchill, E. Peaks and Persistence: Modeling the Shape of Microblog Conversations. In *CSCW'11: Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work*, page 355–358, Hangzhou, China, Mar 2011. Association for Computing Machinery. doi:10.1145/1958824.1958878.

[234] Shen, C., Liu, F., Weng, F., and Li, T. A Participant-Based Approach for Event Summarization Using Twitter Streams. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1152–1162, Atlanta, Georgia, USA, Jun 2013. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/N13-1135/`.

[235] Shi, C., Ding, J., Cao, X., Hu, L., Wu, B., and Li, X. Entity Set Expansion in Knowledge Graph: a Heterogeneous Information Network Perspective. *Frontiers of Computer Science*, 15(1):1–12, Jan 2021. doi:10.1007/s11704-020-9240-8.

[236] Smyth, R. Tottenham 1-0 Everton: Premier League – As It Happened, Jul 2020. URL `https://www.theguardian.com/football/live/2020/jul/06/tottenham-v-everton-premier-league-live`. Accessed on October 16, 2021.

[237] Smyth, R. Leicester 0-2 Manchester United: Solskjær's Side Finish Third – As It Happened, Aug 2020. URL https://www.theguardian.com/football/live/2020/jul/26/leicester-v-manchester-united-premier-league-live. Accessed on January 24, 2022.

[238] Smyth, R. Women's Champions League final: Wolfsburg 1-3 Lyon – As It Happened, Aug 2020. URL https://www.theguardian.com/football/live/2020/aug/30/womens-champions-league-final-wolfsburg-v-lyon-live. Accessed on February 23, 2021.

[239] Smyth, R. Hungary 1-1 France: Euro 2020 - As It Happened, Jun 2021. URL https://www.theguardian.com/football/live/2021/jun/19/hungary-v-france-euro-2020-live. Accessed on January 24, 2022.

[240] Strachan, M. CNET Defends Use of AI Blogger After Embarrassing 163-Word Correction: 'Humans Make Mistakes, Too', Jan 2023. URL https://www.vice.com/en/article/bvmep3/cnet-defends-use-of-ai-blogger-after-embarrassing-163-word-correction-humans-make-mistakes-too. Accessed on Jan 22, 2023.

[241] Suchanek, F., Kasneci, G., and Weikum, G. Yago: a Core of Semantic Knowledge Unifying WordNet and Wikipedia. In *WWW '07: Proceedings of the 16th International Conference on World Wide Web*, page 697–706, Banff, Alberta, Canada, May 2007. Association for Computing Machinery. doi:10.1145/1242572.1242667.

[242] Suárez, E. Tips from the 'Guardian' on Live-Blogging and Covering Breaking News on Ukraine, Feb 2022. URL https://reutersinstitute.politics.ox.ac.uk/news/tips-guardian-live-blogging-and-covering-breaking-news-ukraine. Accessed on March 1, 2022.

[243] Swan, R. and Jensen, D. TimeMines: Constructing Timelines with Statistical Models of Word Usage. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Boston, MA, USA, Aug 2000. URL https://www.cs.cmu.edu/~dunja/PapersWshKDD2000.html.

[244] Syed, S., Spruit, M., and Borit, M. Bootstrapping a Semantic Lexicon on Verb Similarities. In *Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2016) - Volume 1: KDIR*, volume 1, page 189–196, Porto, Portugal, Nov 2016. SciTePress. doi:10.5220/0006036901890196.

[245] Tandoc Jr., E. C. and Johnson, E. Most Students Get Breaking News First from Twitter. *Newspaper research journal*, 37(2):153–166, May 2016. doi:10.1177/0739532916648961.

[246] Telegraph Sport. Parachutist Landing on Pitch Disrupts Inter's Serie A Win Against Sassuolo, Oct 2019. URL https://www.telegraph.co.uk/sport/2019/10/20/parachutist-landing-pitch-disrupts-inters-serie-win-againstsassuolo/. Accessed on November 19, 2021.

[247] Temnikova, I., Castillo, C., and Vieweg, S. EMTerms 1.0: A Terminological Resource for Crisis Tweets. In *12th Proceedings of the International Conference on Information Systems for Crisis Response and Management*, page 134–146, Krystiansand, Norway, May 2015. University of Agder (UiA). URL http://idl.iscram.org/files/irinatemnikova/2015/1229_IrinaTemnikova_etal2015.pdf.

[248] The Royal Family. The Queen Died Peacefully at Balmoral this Afternoon., Sep 2022. URL https://twitter.com/RoyalFamily/status/1567928275913121792. Accessed on September 9, 2022.

[249] Thelen, M. and Riloff, E. A Bootstrapping Method for Learning Semantic Lexicons using Extraction Pattern Contexts. In *EMNLP '02: Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, pages 214–221, Philadelphia, Pennsylvania, USA, Jul 2002. Association for Computational Linguistics. doi:10.3115/1118693.1118721.

[250] Thurman, N. Social Media, Surveillance, and News Work. *Digital Journalism*, 6(1):76–97, Jan 2018. doi:10.1080/21670811.2017.1345318.

[251] Tokunaga, K., Kazama, J., and Torisawa, K. Automatic Discovery of Attribute Words from Web Documents. In *IJCNLP 2005: Natural Language Processing – IJCNLP 2005*, page 106–118, Jeju Island, Korea, Oct 2005. Springer Berlin Heidelberg. ISBN 0302-9743. doi:10.1007/11562214_10.

[252] Tonon, A., Cudré-Mauroux, P., Blarer, A., Lenders, V., and Motik, B. ArmaTweet: Detecting Events by Semantic Tweet Analysis. In *The Semantic Web: 14th International Conference, ESWC 2017*, page 138–153, Portorož, Slovenia, May 2017. Springer International Publishing. ISBN 0302-9743. doi:10.1007/978-3-319-58451-5_10.

[253] Tweepy. Tweepy: Twitter for Python, Oct 2021. URL https://github.com/tweepy/tweepy. Accessed on October 16, 2021.

[254] Twitter. Developer Agreement and Policy, Mar 2020. URL https://developer.twitter.com/en/developer-terms/agreement-and-policy. Accessed on October 16, 2021.

[255] Twitter. Platform Manipulation and Spam Policy, Sep 2020. URL https://help.twitter.com/en/rules-and-policies/platform-manipulation. Accessed on October 16, 2021.

[256] Twitter. Moments that Defined a Record-Breaking Summer on Twitter, Sep 2022. URL https://blog.twitter.com/en_us/topics/insights/2022/moments-that-defined-record-breaking-summer-twitter. Accessed on September 21, 2022.

[257] Twitter. How Many People Come to Twitter for News? As it Turns out, a LOT, Sep 2022. URL https://blog.twitter.com/en_us/topics/insights/2022/how-many-people-come-twitter-for-news. Accessed on September 21, 2022.

[258] Twitter Support. 500 million Tweets are sent every day!, Jan 2015. URL https://twitter.com/twittersupport/status/555076845293432834. Accessed on September 22, 2021.

[259] UEFA. Dictionary - Inside UEFA, 2021. URL https://www.uefa.com/insideuefa/dictionary/index.html. Accessed on July 7, 2021.

[260] Unankard, S., Li, X., and Sharaf, M. A. Emerging Event Detection in Social Networks with Location Sensitivity. *World Wide Web*, 18:1393–1417, Sep 2015. doi:10.1007/s11280-014-0291-3.

[261] United States Courts. Glossary of Legal Terms, 2021. URL https://www.uscourts.gov/glossary. Accessed on July 14, 2021.

[262] Unwin, W. Brazil v Argentina: World Cup Qualifier Abandoned as Authorities Try to Deport Four Players – As It Happened, Sep 2021. URL https://www.theguardian.com/football/live/2021/sep/05/brazil-v-argentina-world-cup-qualifier-live. Accessed on December 8, 2021.

[263] U.S. Election Assistance Commission. Glossary of Terms Database, 2021. URL https://www.eac.gov/glossary. Accessed on July 14, 2021.

[264] Van Canneyt, S., Feys, M., Schockaert, S., Demeester, T., Develder, C., and Dhoedt, B. Detecting Newsworthy Topics in Twitter. In *Proceedings of the SNOW 2014 Data Challenge*, page 25–32, Seoul, Korea, Apr 2014. CEUR. URL http://ceur-ws.org/Vol-1150/.

[265] van Oorschot, G., van Erp, M., and Dijkshoorn, C. Automatic Extraction of Soccer Game Events from Twitter. In *Proceedings of the Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2012)*, page 21–30, Boston, Massachusetts, USA, Nov 2012. CEUR. URL http://ceur-ws.org/Vol-902/.

[266] Vasudevan, V., Wickramasuriya, J., Zhao, S., and Zhong, L. Is Twitter a Good Enough Social Sensor for Sports TV? . In *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, page 181–186, San Diego, CA, USA, Mar 2013. IEEE. doi:10.1109/PerComW.2013.6529478.

[267] Velardi, P., Missikoff, M., and Basili, R. Identification of Relevant Terms to Support the Construction of Domain Ontologies. In *Proceedings of the ACL 2001 Workshop on Human Language Technology and Knowledge Management*, pages 1–8, Toulouse, France, Jul 2001. Association for Computational Linguistics. doi:10.3115/1118220.1118225.

[268] Vivaldi, J. and Rodríguez, H. Finding Domain Terms Using Wikipedia. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, page 386–393, Valletta, Malta, May 2010. European Language Resources Association (ELRA). URL https://www.aclweb.org/anthology/L10-1518/.

[269] Vossen, P., Agerri, R., Aldabe, I., Cybulska, A., van Erp, M., Fokkens, A., Laparra, E., Minard, A.-L., Palmero Aprosio, A., Rigau, G., Rospocher, M., and Segers, R. NewsReader: Using Knowledge Resources in a Cross-Lingual Reading Machine to Generate more Knowledge from Massive Streams of News. *Knowledge-Based Systems*, 110:60–85, Oct 2016. doi:10.1016/j.knosys.2016.07.013.

[270] Vychegzhanin, S. and Kotelnikov, E. Comparison of Named Entity Recognition Tools Applied to News Articles. In *2019 Ivannikov Ispras Open Conference (ISPRAS)*, page 72–77, Moscow, Russia, Dec 2019. IEEE. doi:10.1109/ISPRAS47671.2019.00017.

[271] Véronis, J. HyperLex: Lexical Cartography for Information Retrieval. *Computer Speech & Language*, 18(3):223–252, 2004. doi:10.1016/j.csl.2004.05.002.

[272] Wang, Y., Kim, K., Lee, B., and Youn, H. Y. Word Clustering Based on POS Feature for Efficient Twitter Sentiment Analysis. *Human-centric Computing and Information Sciences*, 8(1):1–25, 2018. doi:10.1186/s13673-018-0140-y.

[273] Waseem, Z. and Hovy, D. Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter . In *Proceedings of the NAACL Student Research Workshop*, page 88–93, San Diego, CA, USA, Jun 2016. Association for Computational Linguistics. doi:10.18653/v1/N16-2.

[274] Weiler, A., Grossniklaus, M., and Scholl, M. H. Evaluation Measures for Event Detection Techniques on Twitter Data Streams. In *Lecture Notes in Computer Science book series (LNCS, volume 9147)*, page 108–119, Edinburgh, UK, Jul 2015. Springer. ISBN 1611-3349. doi:10.1007/978-3-319-20424-6_11.

[275] Weiler, A., Grossniklaus, M., and Scholl, M. H. Run-Time and Task-Based Performance of Event Detection Techniques for Twitter. In *Lecture Notes in Computer Science book series (LNCS, volume 9097)*, page 35–49, Stockholm, Sweden, Jun 2015. Springer International Publishing. ISBN 0302-9743. doi:10.1007/978-3-319-19069-3_3.

[276] Weiler, A., Beel, J., Gipp, B., and Grossniklaus, M. Stability Evaluation of Event Detection Techniques for Twitter. In *Lecture Notes in Computer Science book series (LNCS, volume 9897)*, page 368–380, Stockholm, Sweden, Oct 2016. Springer. doi:10.1007/978-3-319-46349-0_32.

[277] Weiler, A., Grossniklaus, M., and Scholl, M. H. Survey and Experimental Analysis of Event Detection Techniques for Twitter. *The Computer Journal*, 60(3):329–346, Mar 2017. doi:10.1093/comjnl/bxw056.

[278] Weiler, A., Schilling, H., Kircher, L., and Grossniklaus, M. Towards Reproducible Research of Event Detection Techniques for Twitter. In *2019 6th Swiss Conference on Data Science (SDS)*, pages 69–74, Bern, Switzerland, Jun 2019. IEEE. doi:10.1109/SDS.2019.000-5.

[279] Widdows, D. and Dorow, B. A Graph Model for Unsupervised Lexical Acquisition. In *COLING '02: Proceedings of the 19th international Conference on Computational Linguistics - Volume 1*, page 1–7, Taipei, Taiwan, Aug 2002. Association for Computational Linguistics. doi:10.3115/1072228.1072342.

[280] Wikipedia. Glossary of Motorsport Terms, Jul 2021. URL `https://en.wikipedia.org/wiki/Glossary_of_motorsport_terms`. Accessed on August 12, 201.

[281] Wikipedia. Glossary of Association Football Terms, Jul 2021. URL `https://en.wikipedia.org/wiki/Glossary_of_association_football_terms`. Accessed on August 11, 2021.

[282] Wong, W., Liu, W., and Bennamoun, M. Determining Termhood for Learning Domain Ontologies in a Probabilistic Framework. In *AusDM '07: Proceedings of the Sixth Australasian Conference on Data Mining and Analytics - Volume 70*, page 55–63, Gold Coast, Australia, Dec 2007. Australian Computer Society. URL `https://dl.acm.org/doi/abs/10.5555/1378245.1378254`.

[283] Wong, W., Liu, W., and Bennamoun, M. Determining Termhood for Learning Domain Ontologies using Domain Prevalence and Tendency. In *AusDM '07: Proceedings of the Sixth Australasian Conference on Data Mining and Analytics - Volume 70*, page 47–54, Gold Coast, Australi, Dec 2007. Australian Computer Society. URL `https://dl.acm.org/doi/abs/10.5555/1378245.1378253`.

[284] Wu, F., Hoffmann, R., and Weld, D. Information Extraction from Wikipedia: Moving Down the Long Tail. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 731–739, Las Vegas, NV, USA, Aug 2008. Association for Computing Machinery. doi:10.1145/1401890.1401978.

[285] Xiaowei, W., Longbin, J., Jialin, M., and Jiangyan. Use of NER Information for Improved Topic Tracking. In *ISDA '08: Proceedings of the 2008 Eighth International Conference on Intelligent Systems Design and Applications*, volume 3, pages 165–170, Kaohsiung, Taiwan, Nov 2008. IEEE. ISBN 2164-7143. doi:10.1109/ISDA.2008.136.

[286] Xu, Y., Jones, G., and Wang, B. Query Dependent Pseudo-Relevance Feedback Based on Wikipedia. In *SIGIR '09: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 59–66, Boston, MA, USA, Jul 2009. ACM. doi:10.1145/1571941.1571954.

[287] Yang, S., Huang, G., and Cai, B. Discovering Topic Representative Terms for Short Text Clustering. *IEEE Access*, 7:92037–92047, Jul 2019. doi:10.1109/access.2019.2927345.

[288] Yang, Y., Carbonell, J. G., Brown, R. D., Pierce, T., Archibald, B. T., and Liu, X. Learning Approaches for Detecting and Tracking News Events. *IEEE Intelligent Systems and their Applications*, 14(4):32–43, Jul 1999. doi:10.1109/5254.784083.

[289] Yang, Y., Zhang, J., Carbonell, J., and Jin, C. Topic-Conditioned Novelty Detection. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 688–693, Alberta, Canada, Jul 2002. ACM. doi:10.1145/775047.775150.

[290] Yu, J., Chen, R., Xu, L., and Wang, D. Concept Extraction for Structured Text using Entropy Weight Method. In *2019 IEEE Symposium on Computers and Communications (ISCC)*, page 1–6, Barcelona, Spain, Jun 2019. IEEE. doi:10.1109/ISCC47284.2019.8969759.

[291] Zhang, J., Yao, J.-g., and Wan, X. Towards Constructing Sports News from Live Text Commentary . In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1361–1371, Berlin, Germany, Aug 2016. Association for Computational Linguistics. doi:10.18653/v1/P16-1129.

[292] Zhang, Y., Shirakawa, M., and Hara, T. A General Method for Event Detection on Social Media. In *Advances in Databases and Information Systems: 25th European Conference, ADBIS 2021*, page 43–56, Tartu, Estonia, Aug 2021. Springer Cham. doi:10.1007/978-3-030-82472-3_5.

[293] Zhang, Z., Sun, L., and Han, X. A Joint Model for Entity Set Expansion and Attribute Extraction from Web Search Queries. In *AAAI '16: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 3101–3107, Phoenix, Arizona, USA, Feb 2016. AAAI Press. doi:10.1609/aaai.v30i1.10385.

[294] Zhang, Z., Gao, J., and Ciravegna, F. SemRe-Rank: Improving Automatic Term Extraction by Incorporating Semantic Relatedness with Personalised PageRank. *ACM Transactions on Knowledge Discovery from Data*, 12(5):1–41, Jul 2018. doi:10.1145/3201408.

[295] Zhang, Z., Petrak, J., and Maynard, D. Adapted TextRank for Term Extraction: A Generic Method of Improving Automatic Term Extraction Algorithms. *Procedia Computer Science*, 137:102–108, 2018. doi:10.1016/j.procs.2018.09.010.

[296] Zhao, S., Zhong, L., Wickramasuriya, J., and Vasudevan, V. Human as Real-Time Sensors of Social and Physical Events: A Case Study of Twitter and Sports Games, Jun 2011. URL `https://arxiv.org/abs/1106.4300`.

[297] Zhou, D., Chen, L., and He, Y. An Unsupervised Framework of Exploring Events on Twitter: Filtering, Extraction and Categorization. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, page 2468–2474, Austin, Texas, USA, Jan 2015. The AAAI Press. doi:10.1609/aaai.v29i1.9526.

[298] Zhou, D., Chen, L., Zhang, X., and He, Y. Unsupervised Event Exploration from Social Text Streams. *Intelligent Data Analysis*, 21(4):849–866, Aug 2017. doi:10.3233/IDA-160048.

[299] Zhou, Y., De, S., and Moessner, K. Real World City Event Extraction from Twitter Data Streams. *Procedia Computer Science*, 98:443–448, 2016. doi:10.1016/j.procs.2016.09.069.

[300] Ziering, P., van der Plas, L., and Schütze, H. Bootstrapping Semantic Lexicons for Technical Domains. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, page 1321–1329, Nagoya, Japan, Oct 2013. Asian Federation of Natural Language Processing. URL `https://www.aclweb.org/anthology/I13-1188/`.

[301] Zingla, M. A., Chiraz, L., and Slimani, Y. Short Query Expansion for Microblog Retrieval. *Procedia Computer Science*, 96:225–234, Sep, 2016. doi:10.1016/j.procs.2016.08.135.

A

Review

An Apology for TDT's Manual Evaluations

When Meladianos et al. [161] evaluated their TDT approach, they engaged human evaluators to manually annotate more than 1,600 tweets. They called the ordeal "daunting". But then, three years later, when it was time to evaluate a newly-designed algorithm, Meladianos et al. [162] adopted the same daunting methodology, only this time to annotate 2,600 tweets. TDT research complains about manual evaluations but keeps finding itself inexplicably drawn back to them.

In this appendix, we study the research community's complicated but necessary relationship with manual evaluations. We explore the challenges that TDT research faces when evaluating its approaches and the community's failed attempts at designing an automatic evaluation methodology. In the end, we show how research has little choice but to keep returning to manual evaluations, but without excusing devious practices. In this appendix, we answer the following questions:

- What challenges does the TDT community face with manual evaluations? Researchers fixate on the manual, financial and time-related costs of manual evaluations but ignore other aspects that influence the primary concern: scientific correctness. In Appendix A.1, we study what makes manual evaluations in TDT so problematic.

- How and why did the TDT community's endeavours towards automatic evaluation methodologies fail? When Twitter's data sharing policy made labelled datasets obsolete, researchers tried to innovate, in vain. In Appendix A.2, we ex-

plain why automatic evaluation methodologies did not suffice and why they will probably never suffice.

- How can the TDT community make evaluations more reproducible? Researchers accepted the limits of manual evaluation methodologies but took excessive licence with analyses and abandoned reproducibility. In Appendix A.3, we build on ideas by Weiler et al. [278] to propose more reproducible TDT evaluations, which guide our analyses in Chapter 5.

Material from this appendix is in the last stages of peer review [147].

A.1 | The challenges of manual evaluations

Meladianos et al. [161, 162]'s story is not unique. The research community laments the challenges of manual evaluations [5; 274] but, like a vice, keeps returning to them. Weiler et al. [277] found that at least 18 out of 42 surveyed TDT evaluations included a manual component.[1] Manual evaluations were only outnumbered by studies with no evaluation at all: 19 out of 42.

In this section, we consolidate the findings by Weiler et al. [277]. Our annotations of the 79 studies in Table A.2, which follow the methodology in Table A.1, show how manual evaluations of Twitter-based systems persist in TDT literature. However, we also explain how the manual efforts, which research deplores, represent only one of the drawbacks of manual evaluations. In particular, we identify three flaws in manual evaluations: the manual, financial and time-related costs, the indirect effects of those costs on the analyses, and human error and bias.

The costs of manual evaluations

The efforts entailed by manual evaluations are the natural recipients of the TDT community's complaints. It is difficult to carry out a manual evaluation without remarking about the debilitating manual, financial and time-related costs. Even in general, it is "difficult to obtain human relevance judgement[s]" [38], let alone to label the massive tweet corpora of TDT research [62]. When Aiello et al. [5] considered obtaining

[1] Meladianos et al. [161] recruited human annotators to manually label one-minute time windows, but Weiler et al. [277] marked their approach as using "Match facts", ground truth. Therefore Weiler et al. [277] interpreted manual evaluations differently from us, albeit slightly; we consider such an approach semi-automatic.

None	Publications with no experiments whatsoever
Empirical	Publications with only the authors' qualitative analyses to discuss the outputs
Manual	Publications with quantitative analyses based on human annotations
Semi-automatic	Publications with both manual and automatic analyses, or which engage humans to manually annotate a corpus to later evaluate automatically
Automatic	Publications with quantitative analyses that require no human input, normally using annotated corpora or validity indices

(a) The five broad types of evaluation methodologies that we identified in our review.

Other	Publications that include original evaluations that do not fit in any other category
Validity indices	Publications that estimate the quality of clustering algorithms using automatic measures
Keyword matching	Publications that use datasets split into time windows, each associated with a list of keywords if it covers an event
Window classification	Publications that use datasets split into time windows, each of which may be linked to an event
Document classification	Publications that use datasets whose documents may be linked to an event

(b) The five types of semi-automatic and automatic evaluation methodologies that we identified in our review.

Researchers	Publications in which the authors annotated the algorithms' outputs themselves
External	Publications in which the authors recruited students, Amazon Mechanical Turk workers or anyone else without authorship to annotate the algorithms' outputs

(c) The two types of annotators that we identified in empirical, manual or semi-automatic evaluations in our review; some publications employ both the authors and external actors as annotators.

None	Publications that compare the novel algorithms with no others
Parameter tweaking	Publications that only compare the novel algorithms with different configurations of the same algorithms
Trivial algorithms	Publications that compare the novel algorithms with newly-invented, simple algorithms not published in literature
Published algorithms	Publications that compare the novel algorithms with other peer-reviewed algorithms
Published results	Publications that compare the novel algorithms' results with the results of other peer-reviewed publications on the same datasets

(d) The five types of baselines that we identified in our review; some publications include multiple types of baselines.

Table A.1: Our methodology to review TDT evaluations, including analyses of the TDT components in EMM architectures.

human relevance judgements for a manual evaluation, they shunned the "overwhelming amount of effort". They curtly described manual evaluations as "extremely time-consuming, and infeasible in practice", and moved on to a semi-automatic analysis.

While inconvenient, manual evaluations are not as infeasible as Aiello et al. [5] claimed. Our survey corroborates the findings by Weiler et al. [277]: manual evaluations remain ubiquitous in TDT. 38 out of 79 (48.10%) studies in Table A.2 rely exclusively on

Publication	Evaluation	Annotators	Datasets	Tweets	Baseline
Sankaranarayanan et al. [229]	None				None
Petrović et al. [199]	Manual	External			Published algorithms, parameter tweaking, trivial baselines
Phuvipadawat and Murata [204]	Empirical				Parameter tweaking
Sakaki et al. [227]	Manual		2		None
Chakrabarti and Punera [33]	Manual	External			Parameter tweaking
Choudhury and Breslin [42]	Semi-automatic	External	1	1,500	None
Earle et al. [55]	Manual	Researchers			None
Gu et al. [86]	Manual	Researchers and external			Published algorithms, parameter tweaking
Lanagan and Smeaton [121]	Manual	External	8	101–4,073	Parameter tweaking
Marcus et al. [150]	Manual	Researchers	4		None
Popescu et al. [205]	Semi-automatic				Parameter tweaking
Shamma et al. [233]	Empirical		2		None
Zhao et al. [296]			33		Parameter tweaking
Hsieh et al. [96]	Manual	Researchers	18	81,655–1,455,151	None
Nichols et al. [183]	Manual	Researchers	3	72,335–113,189	None
Ozdikis et al. [189]	Manual	Researchers			Parameter tweaking
Petrović et al. [200]	Semi-automatic	External		50,000,000	Published algorithms, parameter tweaking, published results
van Oorschot et al. [265]	Automatic		63	1,050,343	Parameter tweaking
Aiello et al. [5]	Semi-automatic	Researchers	3		Published algorithms
Cataldi et al. [32]	Manual	External			None
Guille et al. [88]	Empirical	External		7,874,772	None
Shen et al. [234]	Semi-automatic	External	7	163,775–345,335	Published algorithms

Publication	Evaluation	Annotators	Datasets	Tweets	Baseline
Vasudevan et al. [266]			15	3,000,000	None
Burnside et al. [29]	Manual	External	4		Published results
Chierichetti et al. [40]	Automatic		2	1,490,000–342,000,000	Trivial baselines
Corney et al. [44]	Semi-automatic	Researchers	2		None
Guille and Favre [87]	Manual	External	2	1,437,126–2,086,136	Published algorithms, parameter tweaking
Ifrim et al. [102]	Manual	External	4		Published results
Martin-Dancausa and Göker [152]	Manual	External	4		Published results
Nutakki et al. [186]	Manual	External	4		Published results
Petkos et al. [197]	Empirical				None
Van Canneyt et al. [264]	Manual	External	4		Published results
De Boom et al. [49]	Semi-automatic	Researchers			Parameter tweaking
Feng et al. [64]	Manual	External		9,000,000	Published algorithms
Kumar et al. [119]	Manual	Researchers	2		None
Löchtefeld et al. [136]	Manual		42	103–11,894	None
Madani et al. [138]	Manual	Researchers			Published algorithms
McMinn and Jose [158]	Manual	External			Published algorithms
Meladianos et al. [161]	Semi-automatic	External	13	72,335–1,907,999	Trivial baselines
Unankard et al. [260]	Manual	Researchers	1	196,834	Published algorithms
Weiler et al. [274]					Published algorithms, trivial baselines
Weiler et al. [275]					Published algorithms, trivial baselines
Zhou et al. [297]	Manual		2		Published algorithms
Adedoyin-Olowe et al. [2]	Manual	Researchers	3		None
Buntain et al. [28]			3	809,426–1,166,767	Published algorithms
Hua et al. [99]					Published algorithms

Publication	Evaluation	Annotators	Datasets	Tweets	Baseline
Liu et al. [129]	Semi-automatic			357,000,000	Published algorithms, parameter tweaking
Preoţiuc-Pietro et al. [207]	Semi-automatic	External	3	2,500,000–150,000,000	Published algorithms, parameter tweaking
Weiler et al. [276]	Automatic				Published algorithms
Zhou et al. [299]	None				None
Akhtar and Siddique [6]			1	58,000	Parameter tweaking
Edouard et al. [56]	Automatic			2,342–43,000,000	Published algorithms
Hammad and El-Beltagy [91]	Manual	Researchers			None
Li et al. [126]	Semi-automatic	External			Published algorithms
Liu et al. [130]	Manual	External			Published algorithms
Mamo and Azzopardi [142]	Manual	Researchers			Published algorithms
Nolasco and Oliveira [184]	Manual		1	432,975	None
Tonon et al. [252]	Manual	Researchers		195,700,000	None
Weiler et al. [277]	Manual	Researchers			Published algorithms, trivial baselines
Zhou et al. [298]	Manual		2		Published algorithms
Chen and Terejanu [34]			3		Parameter tweaking
Hossny and Mitchell [95]					Published algorithms
Huang et al. [101]	Semi-automatic	External	5	218,313–345,335	Published algorithms
Meladianos et al. [162]	Semi-automatic	External	20	41,539–973,985	Trivial baselines
Petroni et al. [198]	None			25,000	None
Choi and Park [41]	Automatic		3		Published results
GabAllah and Rafea [73]	Manual				Published algorithms
Hasan et al. [93]	Manual	External			Published algorithms
Mele et al. [163]	Manual	External		80,134	Published algorithms
Pradhan et al. [206]	Manual	External	3	1,653–24,667	None

Publication	Evaluation	Annotators	Datasets	Tweets	Baseline
Saeed et al. [226]	Automatic		3	124,524–2,335,105	Published results
Weiler et al. [278]	Automatic				Published algorithms
Farnaghi et al. [61]	Automatic		1		Parameter tweaking
George et al. [76]	Manual	Researchers		203,519	Published algorithms, trivial baselines
Hettiarachchi et al. [94]	Semi-automatic	Researchers		99,995–174,498	Published algorithms, parameter tweaking
Mamo et al. [146]	Manual	Researchers	6	63,891–303,982	Published algorithms
Zhang et al. [292]	Automatic				Published algorithms, trivial baselines
Di Corso et al. [50]	Automatic		6	60,005	Published algorithms, parameter tweaking
Kolajo et al. [114]	Automatic			82,887	Published algorithms

Table A.2: A review of TDT evaluation methodologies on Twitter, including analyses of the TDT components in EMM architectures. We only filled in the data when the authors made their approaches explicit. Moreover, we excluded from any calculations the number of datasets used in unspecified TDT tasks, which detect breaking news from general streams and thus rarely require more than one dataset.

Figure A.1: On Twitter, journalists often mix opinions with factual observations. In the tweet above, French football expert Jonathan Johnson thought PSG could count themselves lucky for Gueye's goal because performance had been poor.

human annotators. 14 (17.72%) other semi-automatic methodologies require, at least, a manual effort to compile a ground truth compatible with an automatic evaluation. Of the remaining publications, only 11 (13.92%) perform a purely-automatic analysis.

Despite the prevalence of manual evaluations, constructing the ground truth or annotating the outputs of algorithms manually still requires considerable efforts [225]. Often, the financial costs of engaging neutral annotators force researchers to annotate the output themselves; in 19 out of 52 manual or semi-automatic evaluations (36.54%) in our survey, the authors labelled the outputs themselves. One (1.92%) other study by Gu et al. [86] engaged both the researchers and two external annotators. The efforts weigh heavy on research, but more importantly they distract from more discreet yet equally-consequential challenges.

The indirect effects of manual evaluations

Manual evaluations on Twitter feel conspicuously unscalable [274], but they cause other, more obscure problems too. For instance, the TDT community rarely evaluates algorithms directly. Feature-pivot methods only produce bursty terms with little context about what happened [5; 96]. Volume-based approaches provide even less context: spikes but not what prompted them [28; 97]. Therefore annotators turn to human-readable summaries generated by usually-trivial summarisation algorithms. As a result, research often only evaluates algorithms indirectly, through summaries.

Manual evaluations also limit the scope of the analyses. Normally, researchers only present results for one empirically-set configuration of the algorithm. The compromise appears understandable; researchers cannot configure an algorithm experimentally because each set-up entails an additional daunting manual evaluation. Only 19 (24.05%) studies compared different configurations of their own algorithms, and just eight (10.13%) of those publications compared their algorithms with other baselines from literature. Empirically-set parameters prevail.

Parameters do not only have to be set for novel algorithms but for baselines too. Away from TDT literature, Keogh et al. [109] illustrated the dangers of empirical settings; a previous study, which experimented with too few configurations of the baseline, wrongfully claimed improvements. In TDT literature, George et al. [76] evaluated the baseline with 21 different configurations to identify the best settings but had to limit the evaluation to only a small sample of topics.

Similarly, non-automatic evaluations normally consider few datasets. While the median number of corpora did not differ greatly, semi-automatic and automatic evaluations generally permitted the use of more datasets. Manual evaluations in our survey used, on average, 5.95 corpora ($n = 20$), whereas semi-automatic ($n = 8$) and automatic ($n = 6$) evaluations averaged 6.75 and 13.00 corpora respectively. The small number of datasets in manual evaluations is not a negligible factor. Fewer datasets implies representing fewer scenarios, which consequently makes it difficult to gauge the reliability and generalisability of results [157].

Human error and bias

Research rarely broaches the subject of human error and human bias. Weiler et al. [274] acknowledged curtly that manual evaluations "might suffer from human error or bias", and Panagiotou et al. [191] attributed subjectivity to the nature of manual evaluations. However, the absence of a frank conversation on human error and bias does not absolve manual evaluations of subjectivity. On the contrary, if research cannot decide on what constitutes a domain term or agree on a definition of events, then human bias and error cannot but exist in TDT evaluations.

The prime example of subjectivity appears in an experiment by Allan et al. [11], and Swan and Jensen [243], an early exercise in formalising TDT's manual annotations. As part of the evaluation, the researchers of the two linked studies recruited four students to judge how many topics an algorithm's features captured. The four students rarely agreed. In fact, the experiment ended with such low inter-annotator agreement (kappa $= 0.233$) that the authors could not draw any reliable conclusions.

The presence of human error and bias does not surprise us. What surprises us is the lack of effort by TDT research to minimise it. In fairness, it is difficult to come up with rules to minimise subjectivity; Sakaki et al. [227] mentioned that annotators sometimes struggled to decide whether a tweet discussed an earthquake. In other cases, a fine line separates an opinion from an observation. The tweet in Figure A.1, for example, describes a topic, makes an observation and expresses an opinion all at once.

Notwithstanding the difficulties, few TDT researchers even attempt to annotate output systematically. Chakrabarti and Punera [33] had human annotators label tweets as describing a topic, an observation or something else, and Zhou et al. [297, 298] sought the 'four Ws' in summaries. Elsewhere, the rules written by Hsieh et al. [96] come off as arbitrary: a topic is either a popular tweet or a group of keywords, but only if the authors could decipher a meaning. Few others offer any system for manual annotation. Research implicitly accepts subjectivity as it leaves labelling to the annotators' discretion or, more frankly, bias.

A.2 | The futility of automatic evaluations

In theory, a TDT evaluation should not be any more complex than the analyses from the research area's pilot study [7]: a document—a news report or a tweet—either belongs to a topic or to an event, or it does not. Today, however, those early evaluations survive only as an ideal case-study, a relic out of reach for modern TDT: a comparison of configurations and algorithms with well-defined metrics and minimal human effort, but permitted, in the first place, by a shared dataset.

Twitter's restrictions on data sharing rendered the TDT pilot study's automatic evaluations obsolete. As we explain in Appendix A.3, Twitter limits data sharing, making it unreasonable, infeasible and unscalable to annotate datasets for one-time use. Instead, the research community improvised and sought a different automatic evaluation methodology tailored to Twitter. The principal motivation was the convenience of minimising the manual efforts, but convenience could also lead to sounder evaluations.

In this section, we identify and explore two broad automatic evaluation methodologies in TDT: classification and keyword analyses. The research community has not widely-adopted either automatic methodology, an indication of the intrinsic difficulty of TDT's evaluations. In the rest of this section, we discuss the virtues of classification and keyword analyses, and explore the reasons behind their low adoption.

Publication	Evaluation	Type	Annotators
Choudhury and Breslin [42]	Semi-automatic	Document classification	External
Popescu et al. [205]	Semi-automatic	Window classification	
Petrović et al. [200]	Semi-automatic	Document classification	External
van Oorschot et al. [265]	Automatic	Window classification	
Aiello et al. [5]	Semi-automatic	Keyword matching	Researchers
Shen et al. [234]	Semi-automatic	Window classification	External
Chierichetti et al. [40]	Automatic	Window classification	
Corney et al. [44]	Semi-automatic	Keyword matching	Researchers
De Boom et al. [49]	Semi-automatic	Document classification	Researchers
Meladianos et al. [161]	Semi-automatic	Window classification	External
Liu et al. [129]	Semi-automatic	Other	
Preoţiuc-Pietro et al. [207]	Semi-automatic	Document classification	External
Weiler et al. [276]	Automatic	Other	
Edouard et al. [56]	Automatic	Document classification	
Li et al. [126]	Semi-automatic	Document classification	External
Huang et al. [101]	Semi-automatic	Window classification	External
Meladianos et al. [162]	Semi-automatic	Window classification	External
Choi and Park [41]	Automatic	Keyword matching	
Saeed et al. [226]	Automatic	Keyword matching	
Weiler et al. [278]	Automatic	Document classification	
Farnaghi et al. [61]	Automatic	Validity indices	
Hettiarachchi et al. [94]	Semi-automatic	Keyword matching	Researchers
Zhang et al. [292]	Automatic	Keyword matching	
Di Corso et al. [50]	Automatic	Validity indices	
Kolajo et al. [114]	Automatic	Document classification	

Table A.3: A review of semi-automatic and automatic TDT evaluation methodologies on Twitter. We only filled in the data when the authors made their approaches explicit.

Classification

The obvious route for an automatic evaluation methodology in TDT pointed to classification. Twitter's large datasets, however, made traditional classification, with labelled training and test sets, prohibitively expensive, more so than a manual evaluation [260; 277]. A few, eight of 25 (32.00%) semi-automatic or automatic methodologies, still obstinately annotated a selection of tweets, if not whole corpora. Many others

sought creative alternatives.

If annotators could not label all tweets, they could, at least, label time windows. Seven of 25 (28.00%) studies in Table A.3, had human judges annotate time windows, or aligned the timelines produced by their TDT algorithms with updates from minute-by-minute reports: a precise topic is one that coincides with a ground truth topic [101; 234; 265]. Literature thus found a way to give traditional classification a modern veneer.

Classification may be elegant, but it has too narrow a scope. Classification lumps noise, spam and opinions with observations, statistics and other newsworthy but non-enumerable topics. Every topic has to fit in a rigid two-by-two confusion matrix, but non-enumerable topics with a subjective importance [161; 162], like injuries and missed chances in football matches, do not fit neatly. Some refused to evaluate such non-enumerable topics [161; 162]. Others filtered them manually, still subjectively [101; 234].

Even ignoring the rigidity of the confusion matrix, questionable assumptions undermine classification. Classification assumes that the ground truth is complete, an unrealistic expectation. In our evaluation in Chapter 5, The Guardian's minute-by-minute report failed to mention two yellow cards for Arsenal's Shkodran Mustafi and Bukayo Saka against Southampton. Reports cannot include every interesting statistic and observation.

Classification also assumes that the ground truth can be aligned with the TDT algorithm's timeline. The ground truth might report a topic late or more than once, or the algorithm could detect the topic a few minutes late. Even the slightest delays proliferate errors [265]. The only alternative is projecting topics onto the ground truth manually [101; 162]. Yet in the process, TDT research submits to the same manual efforts that automatic evaluation methodologies sought to eliminate.

Nonetheless, classification's major flaw lies in how it only evaluates accuracy. One of ELD's clusters in Chapter 5 collected opinions on Manchester United's Harry Maguire. The cluster appeared accurate because it coincided with a ground-truth topic, but it missed altogether what provoked the scorn: a rash yellow card. As the TDT community looked for a suitable automatic evaluation methodology, it disregarded the research area's full scope: to detect and track but also to describe [191]. Keyword evaluations, which we discuss next, partially overcame this hurdle.

Keyword evaluations

Keyword evaluations simultaneously measure a TDT algorithm's ability to detect and describe topics. To the best of our knowledge, Lee et al. [124] were the first to propose keyword evaluations, although it was Aiello et al. [5] who popularised the approach.

Topic	Keywords [5]	YAKE! [30]
Chelsea 1 - 0 Liverpool Ramires scores a goal from inside the box to the bottom left corner of the goal.	Ramires, goal, 1-0, Chelsea, score, yes	Chelsea, Liverpool, Ramires, goal, scores
Newt Gingrich: "Thank you Georgia! It is gratifying to win my home state so decisively to launch our March Momentum"	Newt, Gingrich, thank, Georgia, March, Momentum, gratifying	Gingrich, Georgia, Newt, Momentum, March
Republican Party keeps control of the House of Representatives	GOP, Republican, House, control	Representatives, Party, House, Republican, control

Table A.4: Keyphrase extraction can replace manual annotation in keyword evaluations. Above, we set YAKE! [30] to extract the five highest-scoring unigrams from each topic's description. Its output correlates closely with the annotations by Aiello et al. [5].

Keywords evaluations calculate precision and recall by comparing a feature-pivot approach's bursty keywords with a set of ground truth terms, like the ones shown in Table A.4. Nowadays, several re-use the dataset that Aiello et al. [5] built [41; 226] or propose their own [44].

Keyword evaluations do not eliminate human error and bias entirely, at least in the way researchers conduct them. Aiello et al. [5] constructed the ground truth themselves and employed a journalist as an editor.[2] Hettiarachchi et al. [94] followed with seemingly far less rigour. The occasionally vague ground truth, with keywords such as *wrong*, *stupid* and *awful*, do not assuage our worries. On the contrary, it confirms our concerns that researchers could reverse-engineer the ground truth from the algorithm's output to embellish results. In the end, not only did Aiello et al. [5] and Hettiarachchi et al. [94] repeat manual efforts, but they also introduced subjectivity.

In practice, we might find ways to automate a part of the process. Modern keyphrase extraction algorithms, like YAKE! [30], could substitute for humans, as shown in Table A.4, and with none of the manual efforts or subjectivity. However, not even YAKE! could solve the other, more subtle challenges of keyword-based evaluations.

Keyword evaluations only loosely indicate the quality of a feature-pivot technique. Like in classification, keyword evaluations do not compensate for an incomplete ground truth: missing observations, statistics and other topics. More critically, keyword evaluations do not cater to lexical variety; Aiello et al. [5] chose *goal* and *score* to describe Ramires' goal in Table A.4, but you could also say that Chelsea has taken the *lead* or that Ramires put Chelsea *ahead*. The only reliable manner to adapt to the richness of language is to return, again, to manual evaluations.

[2] Aiello et al. [5] did not describe the dataset annotation in the paper itself, but Professor Aiello explained the annotation process in correspondence with us.

The prevalence of manual evaluations is not the result of some hidden virtue. Rather, the prevalence of manual evaluations points to an absence of adequate alternatives. Nevertheless, the futility of the TDT community's automatic evaluations and the way they always fail to replace manual analyses is reminiscent of another research area: summarisation.

The summarisation community also struggles to evaluate. Summarization literature complements the objective measures of automatic evaluations, most prominently BLEU and ROUGE, with the subjective measures of manual evaluations: clarity, cohesion and coverage [58]. It balances the objectivity of automatic evaluations with the reliability of manual evaluations. Just like the summarisation community, TDT researchers must accept the limits of automatic evaluations and the difficulties of manual evaluations, and strive to make both more reproducible. We address reproducibility next.

A.3 | The quandary of reproducibility

While conducting this survey, we often struggled to understand evaluation procedures. We struggled to understand what data researchers used, when a ground truth topic became "of interest to fans" [44] or what constituted a precise judgement, and how one algorithm measured against another. Our experience confirmed what Weiler et al. [277, 278] observed before us: TDT research is marked by an abundance of *ad hoc* studies [277] and a lack of reproducible research [278].

The flaws of manual and automatic evaluations have become a well-accepted fact, but they cannot be allowed to grow into an excuse for irreproducible evaluations. This section concludes the survey with a study on reproducible research in TDT. Our study develops previous work by Weiler et al. [278], but instead of proposing a new evaluation methodology, we suggest ways how research can strengthen existing ones through four aspects: reproducible data, ground truth, metrics and algorithms.

Reproducible data

At the root of all problems to evaluate TDT algorithms lies Twitter's data sharing policy [254]. If studies on particular tasks, such as earthquake detection [55; 225], could share a dataset, they could annotate and share a common ground truth, like early research did. However, researchers rarely share tweet datasets [45], and when they do,

Twitter's policy only lets them share tweet IDs, not the full datasets. Researchers seeking to re-use datasets must download corpora anew, hindering comparative analyses [191].

Setting aside the changing tweeting habits—more noise [161], new features, and looser restrictions on tweet length—downloading datasets results in smaller corpora, without unavailable tweets. Twitter might temporarily suspend or permanently ban an account, or a user might make their account private or delete individual tweets. Every time, the dataset grows smaller and less comparable. Crow [45] aptly refers to these changes as dataset "rot", and the effects can be tremendous. It took Weiler et al. [277] a week to download just a sample of a dataset published by McMinn et al. [157] four years earlier. By then, only 40% of the sample remained.

Naturally, the scale of missing tweets undermines reproducibility [276]. Perhaps the difference between the original and downloaded datasets could be justifiable, to an extent, if the downloaded data remained representative of the original. To the best of our knowledge, no TDT study has ever challenged this assumption. On the contrary, the community rests blindingly on it, as proven by the large number of studies that re-used the datasets of Aiello et al. [5] and McMinn et al. [157]. The assumption is, unfortunately, deeply flawed.

To test this hypothesis, we re-downloaded four datasets that we had collected earlier, as we describe in Appendix D.5. Between one day and almost three years had passed since we first downloaded the datasets, as shown in Table A.5. When we compared the tweets we could retrieve with those we could not, we found that the available tweets contained around 17.44% fewer mentions and 44.14% fewer URLs than unavailable tweets. Retweets were less likely to be available than the typical tweet, and on average, we could not retrieve 88.47% of tweets mentioning the word *stream*, even after just three months.

The average author changed too. Except for the match between Liverpool and Atlético de Madrid, which we downloaded again after one day, accounts that posted retrievable tweets were between one and two years older than accounts whose tweets had been deleted. They were also far less likely to have empty profile descriptions and far more likely to be popular; in the match between Crystal Palace and Chelsea, authors of retrieved tweets averaged five times as many followers as authors of deleted tweets. Moreover, although we could not retrieve between 12.61% and 35.78% of data, between 91.59% and 95.62% of tweets by verified authors always remained available.

The patterns did not distribute uniformly within events either. Tweets published just before or just after the match started had a higher likelihood of becoming unavailable, as shown in Figure A.2. On average, only 69.45% of tweets published within the first 15 minutes of a match remained available, as opposed to 81.27% of tweets in the last 15

	Download date		Tweets	
	Original	**Downloaded**	**Original**	**Downloaded (% available)**
Crystal Palace - Chelsea	30 Dec 2018	29 Aug 2021	63,891	41,028 (64.22%)
Southampton - Arsenal	25 Jun 2020	29 Aug 2021	97,874	70,656 (72.19%)
Turkey - Italy	11 Jun 2021	30 Aug 2021	109,888	90,543 (82.40%)
Liverpool - Atlético de Madrid	3 Nov 2021	4 Nov 2021	107,607	94,040 (87.39%)

(a) Statistics about the original datasets and the same datasets downloaded anew after a period of time. The re-downloaded datasets shrunk considerably due to tweets becoming irretrievable.

	Change between unavailable and available tweets			
	Crystal Palace Chelsea	**Southampton Arsenal**	**Turkey Italy**	**Liverpool Atlético de Madrid**
Average account age	28.62%	38.48%	46.49%	0.77%
Average number of account followers	420.83%	338.50%	30.97%	391.41%
URLs per tweet	-22.85%	-26.29%	-57.62%	-69.80%
Mentions per tweet	-6.78%	-12.00%	-21.16%	-29.80%

(b) The change in mean values of selected attributes between the sets of unavailable and available tweets. Positive values mean that the value was higher in the available tweets than in the unavailable tweets, and vice-versa. For example, in the match between Turkey and Italy, the average account was 46.49% older for available tweets than for unavailable tweets.

	Percentage of available tweets			
	Crystal Palace Chelsea	**Southampton Arsenal**	**Turkey Italy**	**Liverpool Atlético de Madrid**
All tweets	64.22%	72.19%	82.40%	87.39%
Tweets by verified authors	91.59%	93.28%	91.54%	95.62%
Tweets by new accounts (age < week)	42.55%	48.37%	57.73%	69.49%
Tweets by authors with no description	56.23%	68.42%	73.77%	88.85%
Tweets by authors with no followers	48.95%	36.30%	38.19%	29.03%
Retweets	59.89%	66.71%	78.93%	79.66%
Tweets containing URLs	60.24%	67.87%	71.47%	68.57%
Tweets mentioning *stream*	7.33%	11.71%	9.05%	18.02%

(c) The percentage of available tweets calculated for selected groups with particular attributes. For the downloaded dataset to be representative of the original dataset, the percentage of available tweets in each group should be approximately equal to the percentage of all tweets that were still available. Many meaningful metrics change drastically.

Table A.5: Tweet datasets self-sanitise over time. Tweets containing spam-related features, like URLs, are less likely to remain available, even a month after they were published, while most tweets by verified authors remain retrievable for years.

minutes. Tweet availability fell when the use of the word *stream* rose, and rose when the use of the word *stream* fell (Pearson correlation coefficient: $r = -0.9622$).

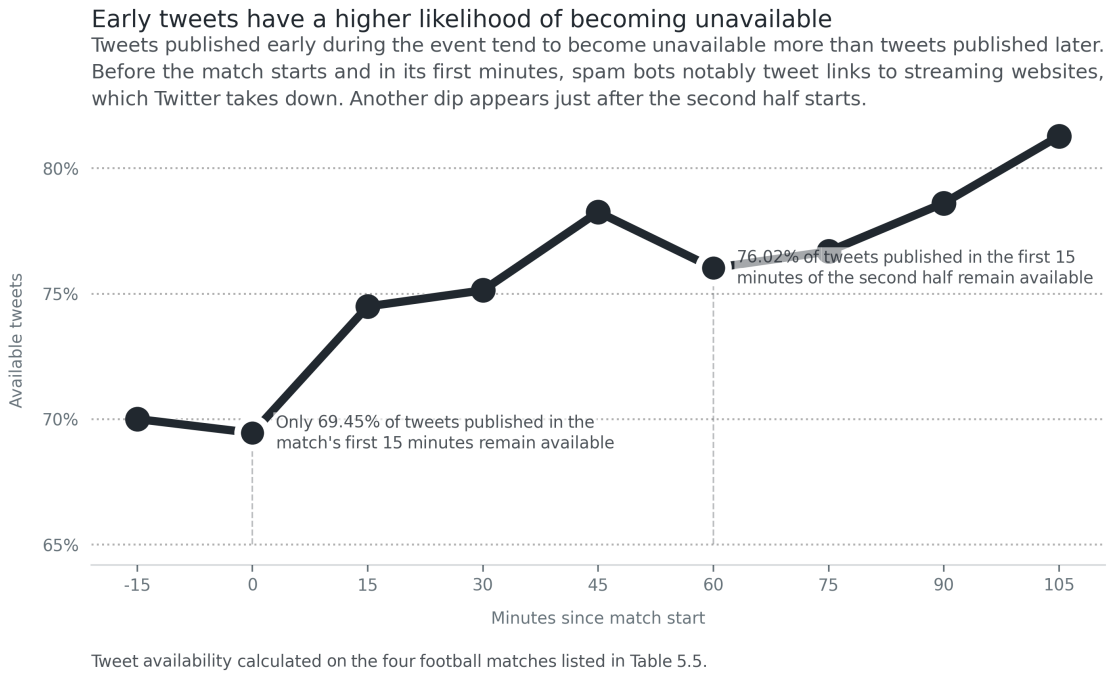

Figure A.2: Unavailable tweets do not distribute uniformly during events. Tweets published fifteen minutes before or after a football match starts tend to become unavailable much more frequently than later tweets. During this period, spam bots tweet links to streams, which Twitter promptly removes.

The changes to the datasets did not occur gradually. 24 hours after the match between Liverpool and Atlético de Madrid, 12.61% of tweets had already become unavailable. By then, the average number of URLs and mentions in tweets had plummeted. Twitter had removed 81.98% of tweets mentioning the word *stream* after one day, and the figure rose to 86.13% after one week. 14.86% of all tweets had become unavailable after just seven days, and yet, in 2019, Choi and Park [41] and Saeed et al. [226] were still using the dataset that Aiello et al. [5] had collected seven years earlier.

These observations are anything but incidental. The average irretrievable tweet represents an archetype of the noisy content for which we configured spam filters in our previous works [142; 146]. The conclusion is unavoidably linked with Twitter's zero-tolerance policy regarding spam [255]. The tweets that cannot be retrieved are more likely to be spam and pollute the data; those that can be retrieved are more likely to be reliable. In short, as time goes by, datasets sanitise themselves.

To confirm our findings, we downloaded labelled datasets that Waseem and Hovy [273], and Founta et al. [69] had made available. The two datasets, both collected to study the characteristics of spam and abusive tweets, were labelled manually to iden-

	Download date		Tweets	
	Original	Downloaded	Original	Downloaded (% available)
Waseem and Hovy [273]	7 Apr 2013–7 Jul 2015	Oct 29, 2021	16,907	10,365 (61.31%)
Founta et al. [69]	30 Mar 2017–9 Apr 2017	Oct 28, 2021	99,799	53,641 (53.75%)

(a) Statistics about the hate speech detection datasets collected by Waseem and Hovy [273] and Founta et al. [69], and the same datasets downloaded anew after a period of time. Their re-downloaded datasets shrunk considerably too due to tweets becoming irretrievable.

Available tweets from Waseem and Hovy [273]	
All tweets	61.31%
Normal tweets	66.11%
Racist tweets	0.61%
Sexist tweets	80.25%

(b) Almost every single racist tweet in the dataset collected by Waseem and Hovy [273] had become unavailable by the time we downloaded their dataset.

Available tweets from Founta et al. [69]	
All tweets	53.75%
Normal tweets	64.10%
Abusive tweets	34.35%
Hateful tweets	41.92%
Spam tweets	55.61%

(c) Abusive, hateful and spam tweets in the dataset collected by Founta et al. [69] were more likely to have been deleted than normal tweets.

Table A.6: Tweet datasets self-sanitise not just in event domains. The labelled datasets collected by Waseem and Hovy [273] and Founta et al. [69] clearly demonstrate how abusive and spam tweets are more likely to become unavailable than normal tweets.

tify different types of noise. Years after the authors first collected the datasets, around two-thirds of normal tweets remained available, as shown in Table A.6. In contrast, Twitter had eliminated almost every single racist tweet that Waseem and Hovy [273] had collected,[3] and the majority of abusive and hateful tweets that Founta et al. [69] had collected. Spam was 8.49% less likely than normal tweets to remain available [69].

Apart from characterising noise, our findings have two implications. First, the self-sanitising behaviour of datasets rejects the assumption that underpins so much TDT research. As Twitter deletes spam, trustworthy tweets by verified or popular authors increasingly occupy a larger share of the dataset, so it would only be natural for precision to increase. In turn, our findings cast doubts on so many studies' outcomes. How much of the improvements over the results that Aiello et al. [5] achieved could Choi and Park [41] attribute to design? How much to time? Re-using datasets only gives a false

[3]The fact that four-fifths of all sexist tweets in the corpus collected by Waseem and Hovy [273] remained available reflects more on the annotation process than on Twitter's moderation. Waseem and Hovy [273] annotated the dataset themselves alongside an external annotator and admitted that they interpreted sexism differently from the annotator; the authors labelled any criticism towards women as sexist, whereas the annotator only flagged flagrant examples of sexism. In fact, 85% of disagreements occurred in the *sexist* label, many over opinions and other genuine, non-sexist criticism of women.

sense of reproducibility: downloaded datasets do not faithfully reproduce the original.

Second, since we know that the missing tweets mainly comprise noise, we can envisage solutions to replace the missing tweets with synthetic noise. It is difficult to replace topical tweets because the algorithm would have to understand a topic's complex structure: Who is involved, What happened, Where and When. Conversely, noise, especially spam and advertisements, does not depend on the event. Noise could be literal gibberish without affecting its purpose, to be noisy, and a TDT algorithm's relationship with it, to avoid it. Deleted tweets indicate where noise might have once existed, and where we might re-introduce it. We leave the task of replacing missing noise for future work.

Synthetic datasets have appeared briefly in TDT research. Although unaware of the cleansing of datasets with time, Weiler et al. [278] ultimately proposed a new type of dataset: the artificial stream. The artificial stream is a statistical distribution modelled over background terms, and into which the researcher injects events by defining topical terms. The solution is a well-intended and reproducible concept, even though it caters only to feature-pivot techniques—it injects terms, not tweets. More glaringly, the artificial stream does not solve other problems.

The artificial stream does not eliminate subjectivity; it merely shifts bias from the annotations to the dataset. Does the researcher choose the quiet setting of an unimportant match, or the erratic, noisy environment of a cup final? A researcher could still be selective with collected datasets [109], but the artificial stream gives the user near-absolute control over the simulated behaviour. Moreover, synthetic data does not capture the nuances of Twitter's behaviour, like opinions or the event shadow [121]. Neither is it unthinkable that the artificial stream could be reverse-engineered to optimise results.

No straightforward solution exists to make datasets fully-reproducible, but TDT literature has ample room to improve how it describes corpora. We suggest that research describes any notable characteristics of the event that might prejudice performance. A quiet goalless draw would logically generate less noisy discussions than a football match in which a fight breaks out among teammates [236]. In this survey, we also observed researchers failing to declare how they collected datasets [5; 161; 162], or how much data they collected [5] or downloaded from an existing dataset [41]. The TDT community should, at least, be more transparent about its data.

Reproducible ground truth

As a research area revolving around news, TDT literature has no difficulty finding ground truth. Whether detecting earthquakes [55], discovering worldwide breaking news [260] or following specific events [161; 162], researchers could always rely on

widely-available and reliable reporting in the news media. Generally, evaluations harness the widespread coverage for robust and reproducible ground truth.

In football, for example, matches enjoy detailed media coverage. In turn, media coverage transforms into high-quality, accessible ground truth [28]. Sports gained popularity in TDT precisely because they allow researchers to share a common domain and measure the recall of objective, easily-enumerable topics. Almost all analyses on football matches evaluate the recall of goals [146; 162], like analyses on American football games evaluate the recall of touchdowns [33; 296].

Of course, not every topic can be enumerated as easily. A goal is unquestionably a goal, but a weak shot might easily be misconstrued as a misplaced pass or cross. Injuries, missed chances and offsides can all be interesting, but their presence in the ground truth depends on the reporter's subjective judgement [161; 162]. Some researchers ignored subjective topics [161; 162], while in our previous work, we only considered them when measuring precision [146].

Still, at times research finds ways to avoid scrutiny. Aiello et al. [5], Corney et al. [44] and others [96; 101; 150] constructed the ground truth of sports events manually from interesting or significant topics. Neither explained what makes a topic interesting or significant. Others sought objective topics but with fickle reasoning. Aiello et al. [5] only retained "some key bookings" from football matches. Meladianos et al. [161, 162], who followed Nichols et al. [183], looked for the start and end of the first half, but only the end of the second half: did the second half kick-off prove more problematic for the TDT algorithm to capture?

Creative interpretation of the ground truth harms reproducibility. It turns what should be an objective reference into subjective annotation thinly-veiled as ground truth. TDT researchers can, however, make the ground truth more reproducible, if only by defining it clearly and objectively, without dubious manipulations. We suggest, in particular, that evaluations follow recall-based analyses on easily-enumerable developments: goals in football matches, touchdowns in American football, electoral victories in politics. At least, such analyses permit comparisons among algorithms, even from afar, across events and papers.

Reproducible metrics

Before Weiler et al. [278] proposed the artificial stream, they proposed standardised TDT metrics [274; 275]. Two metrics automated the evaluation of throughput and redundancy, but three others tied the metrics closely with the ground truth. Weiler et al. [274, 275]'s solution: calculate precision and recall by performing look-ups on Google,

or in the archives of Bloomberg, the New York Times and Reuters, or by mapping events to DBpedia concepts.

Like in automatic evaluations, Weiler et al. [274, 275] made some compromises. The three metrics only addressed a particular niche, popular events in unspecified TDT, and forfeited reliability. More importantly, Weiler et al. [274, 275] reckoned with the futility of adapting IR's simple metrics to a complex domain. In IR and early TDT research, a document was either precise or imprecise, recalled or missed, but on Twitter, the two metrics proved reductive.

On Twitter, few topics are entirely precise or entirely imprecise. Spam and advertisements clearly constitute noise [206], but other topics lie somewhere in-between. In such cases, the annotator still has to decide, subjectively, which label to assign: precise or imprecise. Therefore in the end, the judgement on a topic's validity still depends on the annotator's interpretation of what makes a topic precise [161; 162]. In this section, we focus on two types of topics that we could neither easily call precise nor imprecise: opinions and redundant topics.

TDT literature criticises and implicitly rejects any opinionated topic, but opinions have a place in event timelines. McMinn et al. [157] argue that opinions define Twitter and make the social network a popular choice for event tracking. Elsewhere too, in the news media, subjectivity plays a role in event reporting. The BBC's football match timelines interpose punditry with reporting, and The Guardian's minute-by-minute accounts exude personality. Responding to questions on The Guardian's coverage of the Russian invasion of Ukraine in February 2022, Chris Moran, the newsroom's Head of Editorial Innovation, explicitly recognised the role of opinions in journalism:

> Explainers, visual journalism and opinion pieces are most obviously crucial [to give context for developments and to explain the fundamental concepts]. ... The 'What we know so far' format is of particular use in a story that moves this quickly and which can become overwhelming very fast. Opinion's role here is also critical. It's really telling that opinion in this context shows deep engagement and this speaks to the value it has in explaining and contextualising in interesting ways.
>
> — Chris Moran for the Reuters Institute for the Study of Journalism [242]

In fact, nowadays, most news media publishes opinion pieces or even mixes opinions with facts. 38% of respondents to a survey by Newman [180], holding various roles in the news media, actively encouraged journalists to share their opinions alongside news, a high figure for an industry that traditionally took pride in objectivity. Evidently

opinions reserve some value, unlike spam and advertisements, but the TDT community's narrow definition of precision still bundles opinions with regular noise.

While TDT research criticises opinions, it avoids altogether discussing redundancy, or detecting duplicate topics [277]. Not even Meladianos et al. [161], who designed an algorithm to minimise repeated topics, addressed redundancy in their evaluation. Yet in the context of the event shadow [121], or how discussion about key topics persists for a long time, it is unreasonable to expect an event tracking algorithm to have no redundancy. On the contrary, Weiler et al. [277], who also criticised the lack of discussion on redundancy, demonstrated how the high precision scores conceal high rates of duplicate topics.

TDT literature on Twitter makes a convenient error in ignoring redundancy. By definition, redundant information runs contrary to the research area's first task, first-story detection, and thus brings no value to TDT. Moreover, by ignoring redundancy, the research community disavows the second task: topic tracking. Consequently, researchers should reject redundancy as unequivocally as they reject noise.

Clearly, precision cannot describe the range of topics captured by modern TDT methods. Nevertheless, we do not mean that the research community should abandon precision and recall. We only mean that TDT evaluations should adopt other, more nuanced measures: interpretations that capture the distinct character of social media content.

Weiler et al. [274, 275]'s first two metrics, throughput and redundancy, represent a positive step towards more expressive metrics. With redundancy, Weiler et al. [274, 275] acknowledged the difference between redundant and precise topics. Redundant topics are neither precise nor imprecise and do not depend on the whims of the annotators; redundant topics are simply different from precise and imprecise topics. In the same way, opinions or opinion-based topics differ from traditional metrics.

TDT evaluations would be richer, more objective and, ultimately, more reproducible if the community did not rely on an oversimplified measure of precision. We suggest that the TDT community adopts more expressive metrics than precision by distinguishing between and reporting about the types of captured topics: redundant, noisy, opinionated and true topics. Consequently, like with recall, the community can compare the precision of different algorithms in a limited form.

Reproducible algorithms

Without shareable datasets, common ground truth and a singular interpretation of the metrics, TDT remains without an established state-of-the-art [45]. Meaningful comparisons between systems appear scarcely in literature [274; 277]. Not even Meladi-

Publication	Language	Interface	Domains	GitHub repository
Guille et al. [88]	Java 8	GUI	Unspecified	`AdrienGuille/SONDY`
Ifrim et al. [102]	Python 2	CLI	Finance, politics, war	`heerme/twitter-topics`
Van Canneyt et al. [264]	Java	CLI	Finance, politics, war	`svcanney/twittertopics`
Hettiarachchi et al. [94]	Python 3	CLI	Football, politics	`HHansi/Embed2Detect`
Mamo et al. [146]	Python 3	CLI	Football	`NicholasMamo/eld-data`

Table A.7: TDT literature has very few open-source algorithms. Practical considerations, such as needing an algorithm designed for a particular event domain, whittles down the choice of baselines even further.

anos et al. [162], who designed an improved algorithm as a follow-up to their previous work [161], compared the new technique with its predecessor.

Of course, most TDT publications compare novel techniques with some form of a baseline; only 22 (27.85%) of the 79 studies in our survey do not. However, the comparisons allow us to draw few meaningful conclusions about the quality of novel methods when compared to other techniques. Meladianos et al. [161, 162], like many others, proposed a simple baseline, and 11 other studies (13.92%) presented only results from different configurations of the same algorithms.

In fairness, the scarcity of baselines in TDT research seldom seems a self-serving choice. Like Weiler et al. [277, 278] before us, we struggled to find authors who shared their source code—just five, including ourselves. In reality, practical considerations whittle down further the selection in Table A.7. Some implementations use obsolete programming languages, or impose data or software requirements. Others have poor documentation or no easy-to-use interface with which to configure the algorithms. Others yet may have been designed around the characteristics of particular event domains, and do not transfer to the domain under study.

Without open-source systems, researchers struggle to evaluate the relative quality of novel algorithms or establish a state-of-the-art. Implementing baselines is inconvenient [157], bordering on infeasible without clear implementation details. Even minor, unintended changes could destabilise the algorithm [211; 276]. Instead, researchers often implement trivial, non-peer-reviewed techniques (11.39%) as basics benchmarks.

Others turn to peer-reviewed algorithms, but normally only to the simple ones from early TDT research (45.57%). Shen et al. [234] and Huang et al. [101] used a volume-based method by Marcus et al. [150], simple enough to fit in 33 lines of pseudo-code. Seven used Petrović et al. [199, 200]'s LSH [5; 93; 114; 126; 129; 158; 207], but every one of them used the implementation from 2010 [199], not the more complex, improved version from 2012 [200]. As a result, even the studies that use baselines provide few

Principal contributions

- The most comprehensive review yet of TDT evaluation methodologies
- Proof that tweet corpora self-sanitise over time, discrediting the re-use of datasets in the name of reproducibility
- Proposals to make TDT's datasets, ground truth, metrics and algorithms more reproducible

insights into the quality of novel algorithms.

This time, the solution for more reproducible TDT evaluations is more straightforward: open-source algorithms. In the 2014 SNOW Data Challenge [193], human annotators did not have access to the source code, but they had common datasets and a ground truth, and clear instructions on how to apply metrics [29; 102; 152; 186; 197; 264]. There must have been bias, but it must have been applied uniformly too. Such a pristine environment is impossible to re-create without shareable data, but open-source algorithms can substitute.

Raff [211] argues that open-source code alone does not make a paper reproducible; we see it as a start. If there is going to be human bias in the choice of dataset, ground truth and interpretation of IR's metrics, then let annotators use open-source algorithms to apply bias uniformly to the baselines too. It is our hope that open-sourcing the `NicholasMamo/EvenTDT` repository will facilitate the future development and dissemination of TDT algorithms.

Recap

When Meladianos et al. [162] returned to the "daunting" evaluation from their previous work [161], they might have felt forced to repeat the experiments. Meladianos et al. [162] must considered the alternatives but confronted the reality that no automatic evaluation could replicate the reliability of manual evaluations. In this appendix, we surveyed TDT literature's difficulties with manual evaluations and the reasons why none of the automatic alternatives prevailed by answering the following questions:

- What challenges does the TDT community face with manual evaluations? The manual efforts of manual evaluations conceal other unintended consequences on

analyses. In Appendix A.1, we explained how manual evaluations also force researchers to evaluate algorithms indirectly and subjectively.

- How and why did the TDT community's endeavours towards automatic evaluation methodologies fail? The research area's approaches to automatic evaluations, namely classification and keyword-based analyses, never replaced manual evaluations. In Appendix A.2, we argued that TDT's two automatic alternatives minimised the manual effort and subjectivity but traded away reliability.
- How can the TDT community make evaluations more reproducible? The challenges associated with manual evaluations do not excuse the utter failure to make results comparable or reproducible. In Appendix A.3, we dispelled the notion that Twitter datasets are re-usable and suggested how TDT research could make its datasets, ground truth, metrics and algorithms more reproducible.

B

Encore
The Benefits of Understanding

In Paris, it is time for *Le Classique*, the historic rivalry between Paris Saint-Germain and Olympique de Marseille. Matches between the two French clubs are always a passionate affair, and not in the romantic ways of Paris. Today, however, the French capital hosts no ordinary *Le Classique*: the winner takes the coveted *Trophée des Champions*. Notwithstanding the significance of a cup final, you could not sense the fervour in the English-speaking neighbourhood of Twitter. Ambivalence reigns.

Ambivalence is Twitter's rule, not the exception. For every popular match, countless others draw little attention, at least from the English-speaking world. Out of 20 datasets collected by Meladianos et al. [162] during the 2014 World Cup, the most popular match, Germany's infamous battering of Brazil (973,985 tweets), generated more tweets than the 10 least popular matches combined (967,464 tweets). By all measures, even the least popular match, between Honduras and Switzerland (41,539 tweets), was a popular one. Yet while TDT research has always designed algorithms to scale up to popular events, the community rarely sought to scale down to unpopular events—to be sensitive.

In this appendix, we present what we believe to be event tracking literature's first formal study on sensitivity. In our sensitivity analysis, we study how an algorithm's performance changes as the number of available tweets decreases. When designing the novel experimental methodology, we wanted the results to reflect only the algorithm's abilities. We wanted to avoid uncertainty. We did not want bias, even unintended bias, clouding the results: did performance deteriorate because of our choice of events or because the algorithm really struggles with unpopular events?

To eliminate bias, our experimental methodology fixes the events. We assume that no material difference exists between how Twitter behaves in popular and unpopular

events. The assumption allows us to scale down the same six matches from the previous analyses to simulate real unpopular events without loss of generality. At the end, however, we also present results from whole datasets of genuinely unpopular events.

We scaled down the six datasets using systematic sampling. Systematic sampling minimises the sampling error. It faithfully preserves not only what happened and when but also the event's characteristics: how Twitter reacted to what happened, whether with disinterest, subjectivity or noise. In this way, we gradually reduced the corpora from between 87,717 and 209,132 tweets to 50,000, 25,000 and, eventually, 10,000 tweets. Table B.1 presents a summary of ELD's and SEER's results.

At 50,000 tweets, ELD's precision increased and its recall decreased. The diminished datasets transformed ELD's previously-permissive three-tweet threshold for clusters into a repressive filter. ELD detected topics with caution and so precision increased, but it paid for it in recall. While the baseline comfortably captured the majority of key topics—almost all goals and both red cards—it missed many more non-key topics. At 50,000 tweets, ELD's caution already seemed irrationally excessive, but logically, we could not lower the cluster threshold any further than 3 tweets.

In contrast, SEER's precision and recall dropped together. At 50,000 tweets, the aggregate data about topics became meagre and the shifts in vocabulary unreliable. SEER's precision could only decrease, but the volume still spikes and the vocabulary still shifts with topics in every event, even an unpopular one. Therefore SEER did not have to trade precision for recall and out-performed ELD's F-score, although, for the only time in these analyses, not to a statistically-significant extent (50.69% ↑ 55.54%; one-tailed paired samples t-test: $p = 0.0519$).

At 25,000 tweets, ELD's precision kept increasing and its recall kept decreasing. Throughout our research, two classes of algorithms ventured close to or below 25,000 tweets: trivial methods [121] and extraction or classification models [136; 265]. ELD represented neither and, predictably, struggled. Our baseline still detected most of the key topics comfortably, but overall, it only captured one of every four enumerable topics. Still, not even aggressive filtering, which by now had raised ELD's precision to 66.22%, could reach $\text{ELD}_{\text{Filtered}}$'s (71.53%) and SEER's (71.43%) precision on all data.

Meanwhile, SEER's precision and recall continued to drop, but more gently, this time. Its precision remained above ELD's (66.22% ↑ 68.09%), and so did recall (25.00% ↑ 45.19%), yielding statistically-significant gains in F-score (36.68% ↑ 54.52%; one-tailed paired samples t-test: $p = 0.0519$). Perhaps the most telling sign of sensitivity lies in Figure B.1: with just 25,000 tweets, SEER only obtained a marginally-lower F-score when compared with ELD using all data (55.04% ↓ 54.52%).

At 10,000 tweets, ELD faltered completely. Unable to capture more than four topics

Data	Algorithm	Topics	Precise topics	Precision	Recall	F-score
All tweets	ELD	37.83	20.33	53.74%	56.73%	55.04%
	SEER	33.83	24.17	△ 71.43%	55.77%	△ 62.89%
50,000 tweets	ELD	24.50	14.83	60.54%	43.27%	50.69%
	SEER	△ 27.00	▲ 17.67	65.43%	48.08%	55.54%
25,000 tweets	ELD	12.33	8.17	66.22%	25.00%	36.68%
	SEER	▲ 23.50	▲ 16.00	68.09%	▲ 45.19%	▲ 54.52%
10,000 tweets	ELD	2.67	2.50	93.75%	11.54%	19.67%
	SEER	▲ 27.17	▲ 16.50	▼ 60.74%	▲ 42.31%	▲ 50.34%

(a) With fewer tweets, ELD's precision increased sharply but recall plunged. SEER's precision and recall decreased together but remained at functional levels. The table reports the macro-average number of topics and F-score, and the micro-average precision and recall. We present a full breakdown of the results in Tables F.10 and F.11.

Data	Algorithm	Goals	Cards	Halves	Substitutions
All tweets	ELD	87.50%	52.94%	37.50%	57.45%
	SEER	93.75%	52.94%	50.00%	46.81%
50,000 tweets	ELD	87.50%	23.53%	25.00%	44.68%
	SEER	100.00%	41.18%	41.67%	▽ 36.17%
25,000 tweets	ELD	75.00%	11.76%	12.50%	19.15%
	SEER	93.75%	41.18%	▲ 41.67%	△ 31.91%
10,000 tweets	ELD	62.50%	0.00%	4.17%	2.13%
	SEER	△ 87.50%	△ 47.06%	△ 37.50%	▲ 27.66%

(b) In smaller datasets, ELD still captured key topics, although not as reliably as SEER. Nevertheless, it barely captured any non-key topics. The table reports the micro-average recall for each type of enumerable topic. We present a full breakdown of the results in Table F.12.

Table B.1: SEER degraded more gracefully than ELD in increasingly-small datasets. Our understanding-driven algorithm captured many non-key topics even from datasets with 10,000 tweets. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the baseline, ELD, at each stage.

in any match, ELD's precision increased sharply to 93.75% but recall decreased with equal surety to 11.54%. ELD captured the majority of goals but only one substitution and one start to a half, and not a single card. By the end, precision had increased by 40.01% (53.74% ↑ 93.75%) when compared with using all data, but recall had dropped by 45.19% (56.73% ↓ 11.54%).

SEER degraded more gracefully. Precision and recall still dropped—precision decreased by 10.69% (71.43% ↓ 60.74%) and recall by 13.46% (55.77% ↓ 42.31%) when compared with using all data—but not to ELD's extent. SEER only missed two goals; ELD

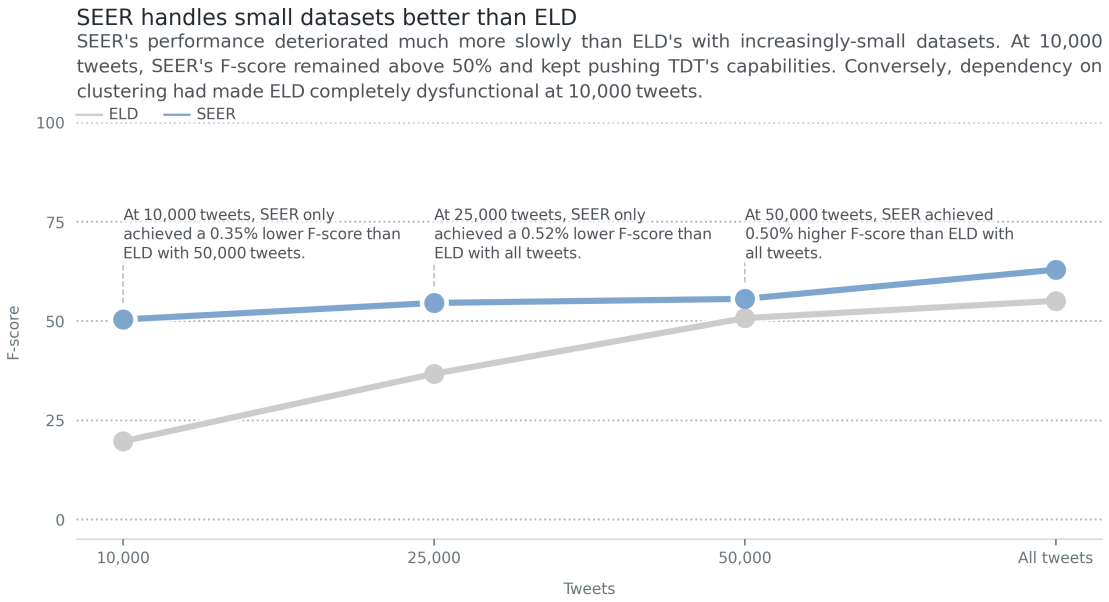

Figure B.1: SEER scales down to the small datasets of unpopular events much better than ELD. At 25,000 tweets, SEER performed almost as well as ELD on all tweets. In fact, it remained functional even when we scaled datasets down to 10,000 tweets.

missed six. SEER captured more than a third of halves and more than a quarter of substitutions; ELD captured one start to a half and one substitution. Finally, SEER captured almost half of all cards; ELD not a single one. Even at 10,000 tweets, our novel algorithm remained functional.

Therefore we pushed SEER further. We wanted to test our assumption about how Twitter behaves in unpopular events and confirm our findings on real events. We wanted to eliminate the mere possibility of sampling error. Thus, we evaluated SEER at levels far out of reach for ELD's document-pivot approach, on datasets that we collected from real and unpopular events, as shown in Table B.2.

At 10,163 tweets, in the Trophée des Champions [63], SEER performed similarly to how it had performed with 10,000 tweets. It missed Paris Saint-Germain's first disallowed goal but captured the next one and the trophy's three deciding goals. SEER also captured the event's progression in the key substitutions, interruptions and missed chances. Its performance bore a clear resemblance to its own results on the simulated datasets with 10,000 tweets, both in precision (60.74% ↑ 61.90%) and in recall (42.31% ↑ 45.83%).

At 7,943 tweets, in the match between Parma and Milan [151], SEER performed with the assuredness of an algorithm processing much larger datasets. It captured all goals,

Event	Dataset size (tweets)	Topics	Precise topics	Precision	Recall	F-score
Trophée des Champions	10,163	21	13	61.90%	45.83%	52.67%
Parma - Milan	7,943	29	21	72.41%	57.69%	64.22%
Copa del Rey	2,774	32	14	43.75%	37.50%	40.38%

(a) SEER maintained a relatively-high recall even in the small datasets of unpopular events.

Event	Redundant	Noise	Subjective	Non-enumerable	Enumerable
Trophée des Champions	4.76%	28.57%	4.76%	19.05%	42.86%
Parma - Milan	6.90%	6.90%	13.79%	24.14%	48.28%
Copa del Rey	28.13%	3.13%	25.00%	25.00%	18.75%

(b) As datasets grew smaller, and as SEER monitored changes in increasingly-sparse data, redundancy and subjective content started to infect the timelines.

Event	Goals	Cards	Halves	Substitutions
Trophée des Champions	80.00%	0.00%	75.00%	40.00%
Parma - Milan	100.00%	44.44%	100.00%	33.33%
Copa del Rey	100.00%	0.00%	100.00%	12.50%

(c) SEER could capture some non-key topics even in the Copa del Rey final, whose dataset had fewer than 3,000 tweets.

Table B.2: SEER's improvements persisted in real datasets from unpopular events. Even on datasets with a few thousand tweets, it captured almost all key topics and the most interesting non-key topics.

every start and end to a half, and other non-key topics. You could barely tell that it had less than a tenth of the data of the match between Southampton and Arsenal.

And at 2,774 tweets, in the Copa del Rey final [216], SEER finally faltered. It captured every half start and end, as well as the winning goal, but it yielded to redundancy and subjectivity—although rarely to noise.

C

Interview
Prof. Charlie Beckett

This appendix includes a transcript of our interview with Professor Charlie Beckett. We interviewed Beckett on 20 January 2023 via a Zoom meeting, which we recorded. The full transcript of our conversation, edited for readability, follows next.

N. MAMO: Let's start with what I hope is an easy question. Who is Charlie Beckett, first of all, and why did he launch JournalismAI?

C. BECKETT: I'm a professor at the Department of Media and Communications and I'm a former journalist, and I teach and research journalism. I'm at the LSE [London School of Economics] because I run a journalism think-tank called Polis. That's why they hired me: they deliberately wanted someone from the profession to come into the university, to bring that kind of perspective. That's very much what I still do.

On the JournalismAI project, I'm doing it because I saw AI as the next wave of technological change in journalism. My work has all been about the future of news, partly about technology but more generally about how journalism is changing and what consequences that has for both the news media and society. I saw AI as the next technological wave that's happening.

[JournalismAI's] main mission is to support good journalism, to be honest, but it also functions as a form of research. It's kind of active-learning research for me and for everyone else that is a part of it. We do a lot of normal research—surveys and so on—but by working with these journalists, by teaching them, by doing innovation workshops, we find out both what they think about AI, but also we're actually learning about what you can do with

AI: what works, what doesn't work, what impact it has, what consequences it has when you do it. That's how I do my research.

N. MAMO: JournalismAI started with your 2019 report [18], at least in the public. How have you seen the applications of AI change? And how have the challenges changed as well?

C. BECKETT: It was four years ago that we did the research, and a lot has changed in that time, although it has mainly been about exponential acceleration. It has been about the increase in adoption. You know how adoption works: you have pioneers, and then you have follower-on, and that creates a kind of momentum where people feel that they have to keep up and so on. We're now at the keeping-up phase, where there is enough use of AI out there for it not to be a discovery. It's not emerging. It's now evolving and developing. Otherwise though, it hasn't changed so much. I think the framework that we established in that report is pretty resilient actually. For example, AI is generally used to supplement human labour. It very rarely replaces it, and in many cases it actually creates new labour, which can be good, either by creating new formats or by the need to review and edit the actual technology. Generally-speaking, that report has been pretty robust in terms of the opportunities and the challenges. The reasons for not adopting it are very similar still: it's a lack of resources, lack of time, lack of skills, lack of knowledge, some cultural obstacles as well, and also the intrinsic challenges around the technologies.

N. MAMO: You've been asked this question plenty of times—whether robots will replace journalists—and you always say that no, AI will supplement journalism and even create new opportunities. Just this week, for example, we saw CNET getting caught, in a way of speaking, using an AI to write articles [240]. In a way, AI is entering newsrooms everywhere. I won't ask you whether robots will replace journalists, but I will ask you why not. As in, what do humans do so much better than robots?

C. BECKETT: That is the right way to ask the question, actually. It's not whether humans are going to disappear; it's asking the questions that we always ask of technology. What can a technology do and what do we need humans for? And also, the secondary question to that one is: what do humans need to do differently? That's interesting: how differently do humans have to behave because of this? As you said, it's not just about "here's a lump of labour and robots are going to take it over", or "here's a human role that the technology is going to do."

Sometimes, that's going to be the case. So for example, one I always give, when I was a journalist and I wanted archives, if I wanted to do research, I went to a room full of newspaper cuttings. There was a lovely lady there who spent her day cutting up newspapers and putting them in bits into envelopes. She doesn't work there any more. She's fired, she's gone, she's doing something else. That job does not exist, so that's an interesting example of a technology replacing it. Research killed her job.

There will be loads of things that are, as you say, easily-identifiable. The ones that aren't fall into two categories. One is around the use-case of journalism, and that depends very much on what your journalism is. If your journalism is publishing share prices, well, that's more vulnerable to technological adoption. If your journalism, though, is rapid response assessment of events, or what somebody has said and how important it is, there's a degree of judgment and selection. That may be what humans are better at.

Then there's the other category, which I call the "fluffy category", the so-called creativity and perhaps the emotional, empathy bit. Algorithms are very bad at sentiment still; they struggle hard at that because sentiment is messy—humans struggle with it, so let alone machines. And there's a positive there, which is that perhaps human journalists will get better and do more of the creative, empathy, judgment stuff. Again though, it's a false binary to say that's completely different and can't be supported by technology, but it kind of falls into this category.

There's a third category, actually, which is efficiency. Sometimes the machine can do it, but why bother? It's just not worth-it. It wasn't very difficult for a human to do it; frankly, it's quicker if a human does it. By the time you program the thing, and the risk of it getting it slightly wrong and you having to check it, the return on investment is not worth-it. So there's that third category, of efficiency.

N. MAMO: You mentioned the human judgment in news production. Would you also say that machines lack a certain human-like understanding of how the world works, how news works, that makes it difficult to reason about and judge events or news?

C. BECKETT: Yeah, and I don't necessarily mean some deep intellectual genius. I mean, think of a football match. You can program the software to say "okay, I'm going to look out for goals. I'm going to tell you how many corners they got, if somebody got sent off." But what if there's a VAR judgment? You know, it's one of the big stories in every football match now. The VAR, the

software itself made an intervention in the game and they have to decide: was that a deliberate handball or was it accidental?

It's really funny, you can hear the commentators saying things like "that was a natural movement." What? What does that mean? Is there an "unnatural movement"? What does that mean? But we all kind of know what it means, and it's a controversy. There isn't an answer; maybe some people will say it's a handball, and some will say no, it was accidental. To capture that in an interesting way, which is very trivial and banal and doesn't really matter—you don't need a university degree to have an opinion—is harder for the machines to do.

One other thing that machines find difficult to do is morality. So much of journalism, quite rightly, should be about "this is unjust." I think it's quite difficult to program for social justice, partly because it's an ideological stance, but you can program ideologically, we know that. With ChatGPT, some people are saying that it has a liberal bias, apparently, which I think is very funny. I saw a conservative commentator say that it's got a liberal bias because the program relies upon facts. He was saying that "the trouble is that the left and the liberals have facts on their side; we conservatives have beliefs." You think "oh dear, that's interesting."

But anyway, you take my point. Karin Wahl-Jorgensen did a very good study of Pulitzer prize winners, and one of the key emotions was anger, in the sense of anger about social injustice and so on. I think that might be hard to program for.

N. MAMO: I want to put AI aside for a bit; we'll come back to it very soon. Before looking at the applications, I just want to take a look at your idea of networked journalism. In 2010, you predicted that the live blog would become the front page of newspapers and, more importantly, that social networks would become much more important in the day-to-day work of newsrooms [19]. Thirteen years later, what role does social media play, in particular in the newsgathering task?

C. BECKETT: Yeah, it's interesting. When I wrote the book in 2008, Facebook had just started, and the whole social media thing hadn't really taken off. So when I was talking about "networked" I was talking about the internet more generally and the audiences of digital tools. It's quite interesting looking back at that. I wasn't really talking about social media as such.

In a way that's quite a narrow definition, where journalists use digital tools to engage the public in the process of production and dissemination of jour-

nalism, and the creation of journalism. It's quite a narrow definition, really. Obviously, you can always think of it more generally with Manuel Castells' network society; because of the internet, digital and social media, everything is interactive, everything is connected, potentially digitised and datafiable and so on. So in that sense, it has become environmental, as Roger Silverstone said: media has become environmental, it's not a separate entity.

That's the key to understanding social media and the stuff that I've written about emotions, if you look at that. The key idea is that the affective nature of media is very much conditioned literally by the mobile phone. In a sense, we're going back to the 2008 thing of the device, like the internet, the hyperlink. The mobile phone, the device where we are all now interlinked and hyperlinked, embodies that because it's so integral to our lives, so embedded in our lives, literally and physically. You know, we go to sleep [...], it's always on, it's always there.

In that sense, we think of social media as the platform, but it's not. In social media, it's the social bit that's important. It's the behavioural bit that I think is really important, and by that I mean the way we relate to it physically and emotionally, and socially and practically, and informationally and intellectually is so important—the way we integrate it into our lives.

That includes things like the performative nature of it. When we use this thing, we're performing in a particular range of ways that we don't generally in society; when we go to work, we behave differently to how we behave down at the pub, or when we're with our family. And when we do social media, again we behave in different ways. It's a no-brainer but I still find it amazing that journalists still have this idea that they're going to just take their journalism and do it on a platform; they will take their journalism and they're going to just transfer it to the internet, or they're going to transfer it to a social media platform.

Obviously, the best ones don't. The most recent example would be TikTok. As usual, I was right, I was ahead of the curve in the same way I said about live blogging and all these other things. I said "look, TikTok, get on it—it's going to be massive, it's got a particular quality to it and a particular sociality to it, and if you're not there, then you're not going to be part of a huge area of people's lives."

What was key about it was that you couldn't just go on TikTok and say "hello, I'm a journalist, here's my one-and-a-half minute report from this morning's TV bulletin." You have to do something ludicrously different:

Sophia Smith Galer did her sea shanty stuff about her new story and things like that. The guy from Washington Post... They do things differently, and some of that will be good, some will be bad.

I think that's the continuing impact of social media, although I think that we're now entering, as you know, a very interesting phase around networks and platforms, which of course is regulation. Well, it's both regulation but it's also a kind of market reckoning as well: what are these things there for? As we've all said from the beginning, these are particular things. Facebook, for example, is a private company owned by basically one guy, and if he changes his mind about it, it's going to change.

N. MAMO: That's what happened with Twitter. That's what I wanted to ask you next, in fact. Twitter is changing; it is in danger, even financially. You said that the social aspect of social media is bigger than social networks and individual platforms. At the same time, we've seen Twitter transform into a public town square, not just in the way that [Elon] Musk refers to it to get advertising money, but also in the way it gave a voice to everyone, for better or for worse. That kind of voice has been exploited or harnessed in scientific research for newsgathering. Do you see that sort of thing also in journalism, where Twitter is used—or perhaps journalists even depend on it—to find leads? And what do you think would happen if Twitter suddenly had to disappear?

C. BECKETT: Well, firstly I think that something would replace it. There would be other versions of it. Obviously there's a huge caveat around Twitter, which is that it's not that big. It's particularly important to journalists, but it's very distorting. In some countries, like the UK, it's quite significant, but in many countries it's irrelevant. Even in the UK, when you actually look at the number of people actually doing stuff on Twitter, it's vanishingly small, really. But it's much bigger than a journalist would normally have by just wandering around the street.

The other caveats around it is it being very unrepresentative. And as I said; there's a way people have discourse on Twitter, which is not generalisable to other platforms or the internet in general. And it's certainly not generalisable to real-life: your Twitter will be so different to my Twitter because of the people you follow. So when people say "Twitter this, Twitter that", they really mean "my Twitter" or "my imaginary Twitter", including when academics do research on it.

What I think is partly that journalists are learning about the limits of a platform like Twitter. They're thinking "hang on, I'm spending my whole day on this thing, and this is ridiculous: I'm doing a story based on three tweets as if they're quotes, as if they're interviews, and they're not." There's a kind of hollowness there if you're not careful.

What I think is perhaps more interesting is thinking about social and data journalism more broadly, where people—especially with the AI technologies but also just generally with data journalism—are thinking about other forms of data apart from social media discourse. Yesterday I read a story about India and decreasing population. Statistics are notoriously bad in India, for anything, so what they've done in this story was that they were quoting things like searches for baby bottles, or baby prams or baby carriers. They were saying that they've gone up 20%, which would suggest that there's some correlation with increasing fertility. I thought that was a really nice way of expressing it, of telling a story, and more interesting than "I've seen a tweet."

Data journalism, we tend to think a bit about pandemics or bank records and so on. But if you think about it more generally, the increasing datafication of our lives is going to be more interesting than those things, and we're going to be interested in other forms of discourse. For example, other social media platforms like Reddit, Tumblr [...] often have much more interesting sources for trends and stories than an obvious one like Twitter.

N. MAMO: You mentioned datafication and it's not just that we have data now, that we quantify various aspects of the world and of the news, but there's also the problem of there being too much of it sometimes. If I had to condense my research, it would be solving that problem, the problem of information overload. I developed two applications and in the last part of this conversation, I would like to focus a little bit more on them.

When I asked you what humans have that machines do not, and when I prodded you on understanding, that's because I approached the problem of information overload from an understanding perspective. The idea is that machines don't really understand who is participating in an event. I gave you the example of Liz Truss' premiership in the UK: machines don't understand who the important figures are in UK politics.

More importantly, they don't understand what matters. Twitter gave a platform to subjectivity, to bias. That is not news; as you said, three tweets don't make a story. I developed algorithms to detect who is participating and what

is important in different domains, mostly focusing on football and politics. The idea is that if we know that speeches or resignations are important in politics, we can drive newsgathering with that information; we can focus just on speeches and we can focus just on resignations.

I developed these two applications, and my first question on this timeline or live blog is about your first impression on quality. What does it do well and what does it not do well? More importantly, do you see any applications for it in the newsroom?

C. BECKETT: I think that the live blogging thing is interesting. As you mentioned earlier, it's one of the few things I predicted accurately, and it's this sort of perverse thing. When The Guardian first started doing it, what was interesting about it is that it broke so many rules. Newspapers are supposed to be a set record of events; they're not supposed to be a commentary upon events—that's kind of broadcasting. They also did things like all the hyperlinking, even to other media; again, you're not supposed to do that—you're supposed to be the fount of knowledge.

I think that in that sense, the two examples—the football live blogs and the politics live blogs—are both interesting ones. I should revisit to see how they do it now, but in a way, the journalist would be sitting at his or her desk and have various [browser] windows open and be combing through stuff. They probably have alerts and so on—perhaps they're helped by the technology as well. So I think that something like this can be the backbone to a live blog. The interesting bit—and this is what journalists talked to me about when they say they're using things like this—they say that they use this, and when they see something that's a bit stand-out, they can follow up on it. That idea of judgment.

Always remember that people want different types of news. That's why people go to The Guardian's website or their app. And sometimes they'll want the podcast, the Today in Focus podcast, which does a deep dive on a story, or they'll go to the long reads section because they want a long, complicated read. Sometimes they just want to scroll rapidly through some headlines just to check-in, the pause thing; people have a pause in their day, and they just want to make sure there's nothing happening: "oh there's a live blog about the queen dying", and they'll flick through that live blog. I doubt that there are many people who will sit there for an hour looking at a live blog as it goes click-click-click. I'm sure that's not the experience, in the same way that people tend not to with 24-hour TV news. They dip in an

hour, or it's there in the background.

The judgment around it can be kind of odd. We can see how bad Twitter trending topics is, for example. It's really gotten especially bad at the moment. We don't know what the algorithm is doing now. The point is that sometimes, something has a resonance. Resonance is really hard to program for. You can have a relatively trivial event, something like an MP has been rude to his secretary, and you think "well okay, that's not a criminal offence, he was not a very important MP", but it may resonate. It may have a turn of phrase that he used, or it may be symbolic of wider fears and concerns around sexual harassment or something—it's a resonance.

There's also the kind of relevance, what you put into the live blog, and this [SEER's timeline] looks like a good one to me. This looks like it's hitting all the right things, as you say, hitting the right names, understanding status. All those things that journalists do in a formulaic way, it will do well.

Will it get it quite [as well] if, for example... It will recognise Jeremy Corbyn, who was the Labour leader, that he's quite important. It may even understand that he has a certain ideology. But what if Corbyn says something a bit unusual? What if he says "I like bankers"? "Capitalism is great"? A human would go "Jeremy Corbyn is saying that capitalism is great, that's interesting!" The machine might not. The machine might say "yeah, capitalism is great, that's not a controversial statement."

N. MAMO: That is a problem of understanding as well.

C. BECKETT: But that's alright because that's what journalism has. Famously, John Birt, who was Director-General of the BBC when I was there, talked about journalism's bias against understanding. It's a strange thing because journalism has to simplify things and turn everything into a formula, and often reduce to something simple. Because journalism is obsessed, for example, with conflict, it's always attracted to that rather than what is necessarily-true or significant, and it doesn't always help you understand. It's more interested in the drama of the story, for example, than helping you to understand.

So I think there's a danger of exaggerating how good journalism is at always giving you the full, balanced, explained narrative about something. Does that make sense? You want the algorithm to be almost better than routine journalism.

N. MAMO: The problem—and this sort of links with what you said earlier about what resonates with people—is that since we're using tweets, we're not just

detecting the news, but rather we are detecting the news that people find interesting. That can be helpful because it tells the editor, for example, what can potentially resonate with the audience, or what deserves its own article, but at the same time, it also conflicts a little with what you said about morality at the beginning of our conversation: just because something is interesting, it doesn't make it newsworthy, or maybe it can amplify certain ideas. How much would you trust an AI, in particular an event detection system like this?

C. BECKETT: I would trust it as much as a human journalist because it is going to be programmed and edited by humans. I think that, in a way, that's a misleading question. You talked earlier about how you can use technology to counter news avoidance. I think the phrase news avoidance is a strange phrase because journalists use it to mean that people are doing something perverse, that there's this wonderful thing called news that journalists create and for some bizarre reason, a lot of people are avoiding it. Either they are totally avoiding it, they say that on any news, or they're saying there's too much news.

We see this as a problem. We aren't seeing it as quite a normal response to the world, a world of abundant information, of constant, persistent overabundant information. I think it's actually the [other] way round: we should assume that it's quite logical to want your news to be rationed, and your news to be relevant, and for a lot of people, they don't care that Liz Truss met the King at Buckingham Palace. What relevance does that have to their life? They're worried about their heating bill, they're worried about their cat who is sick, they're worried about whether their kids are going to get to school on time. They're worried about them being a bit overweight, and they don't know if they're going to be able to get to the gym in time.

There's all these other things and they're interested in other things that journalism isn't interested in. Journalism doesn't care about those things often, and journalism isn't very entertaining. It can be depressing and boring and complicated.

N. MAMO: So you are comfortable with AI prodding the journalist and telling them "listen, this could be more relevant than other news to your audience."

C. BECKETT: I think [that is] one of the most interesting things about using AI, and one of our Collab teams did this about countering human bias. Journalists have this bias towards whatever, a certain type of story; they went to Oxford and they studied PPE [Philosophy, Politics and Economics] so they really

do think that what the Prime Minister does is the most important thing everyday. They didn't do science so they're ignoring completely the fact that there's this scientist who has done this amazing biotechnology experiment that's really going to change our lives. "No, no, no! What the Prime Minister does is really exciting."

I think that we can use these technologies firstly to understand better what's happening in the world, but secondly to understand what interests people. How do things connect to people's lives and what does our audience do? That's possibly the biggest revolution in journalism in the last ten years: audience data. We now understand what people do with news. We don't understand why or how they feel about it particularly, but we can at least measure their behaviour. How do we do that? We do it with this software.

N. MAMO: I want to move to the second application very quickly. Even with live blogs, if you have Liz Truss' premiership—even though it was short, 44 days are still a lot of information to consume—or if you have Russia's invasion of Ukraine, some events tend to change very quickly and stretch for a long time. The idea of this event visualisation is that you could focus, for example, just on tax-related information. You could just filter at a whim. In other words, navigate events like searching or filtering. Same question as before: what was your first impression of this visualisation, and do you see any application for it in the newsroom?

C. BECKETT: I don't know about the actual application here, the way it's structured. I think we are seeing increasingly the kind of journalism—and it's usually data journalism of some sort, which is what this is, in a sense—that makes connections. We saw a lot of it during the pandemic, of course, where there was that kind of "how is the pandemic changing society?" So we were tracking the virus itself and the health effects, but there was also all this interest, for example, in working from home, trying to measure how that has progressed over time, how it has had an impact on education over time. I could imagine you doing this kind of thing with an issue like the pandemic. And also I think journalism and the public are becoming increasingly interested in historical context, and by that, I don't necessarily mean "let's look back nostalgically." I mean what has changed, what the direction of travel is around an issue. I mentioned demography, the population story, because there are so many aspects to that: Why is it changing? What are the other factors in it? When we look at that, what are the other stories that are being told to us? There's a change in women's rights, there's a change in con-

traception, there's a change in the wealth of certain countries—that's why demography is changing. That is increasingly-interesting for journalists and the public as they seek to understand their world. Those connections, I think, are increasingly-interesting to people.

If you think about it, what is the attraction of conspiracy theories? They're completely bogus, but people love to make connections. They love to see causations and relations between things. That's because, as you know as a scholar, correlation and causation are often confused. Just because something is related doesn't make it causation, but it's related and that's interesting. I think this kind of way of thinking about an event or a person is really interesting.

N. MAMO: In fact, these kind of visualisations, in research we call them either the event knowledge graph—a graph that represents knowledge—or event models. They could be used as a visualisation, which is our primary purpose given the constraints that we have, but also one of the ideas that has been proposed is that they could be mined automatically to identify news angles [188]. For example, if someone gave a job to somebody else, and the first person is the father of the second person, then that's nepotism.

C. BECKETT: Yeah, it would be a great nepotism graph!

N. MAMO: Yeah, so you have software, AI, that continuously mines the event knowledge graph to come up with news leads or news angles. That's one of the applications that we came up with.

I just have a couple of questions left, and I want to go back to JournalismAI. I've been following JournalismAI for two-and-a-half years now, I think, and one pattern that emerged from the Collab Challenges of 2021 and the Fellowships of last year is that most teams seem to approach problems from first principles. For example, one thing that really stuck with me from The Guardian's quote detection program is that they even had to define what a quote is. It seems as if the problems that journalism is tackling are completely novel. Do you feel like AI technology and research are not addressing the needs of journalism right now?

C. BECKETT: Yes, basically, in a word: yes. I think you make a really interesting point there, actually. One of the things that pleasantly-surprised me when we did the Collabs and the Fellowships was that people did often go to first principles. They sort of said, as we keep saying, before you start using this thing, think about what problem it is that you are trying to solve. As you know, again as an academic, defining your terms is so important. Defining

your research question is so important before you go do your research. As you said, it was quite interesting how they went back and asked: "what are we trying to do here?"

One of the teams this year looked at how you can fact-check politicians' claims. There was a technical issue there about how on Earth they did that: what is a politician and what is a claim? Is it just anything they say? How do you conceptualize that? It's quite an interesting question.

I used to be a political journalist and one of the things that I find most tedious, certainly about British political journalism, is the endless sort of "oh you said something slightly different to yesterday so therefore that's a U-turn." They'll talk about the emphasis from the Prime Minister that was subtly different today. And we know what they mean, but they're talking in a kind of code and they're making assumptions. The general public are thinking "I don't understand, sounds to me like the same thing, this is so fine-tuning." They don't understand the context: what did he say before? So again you can match with your knowledge graph: this is what they said about taxation before; this is what they are saying now; this is what it means.

I have been having this conversation the past few days, funnily enough, because we were thinking about what we'll do this year by thinking about new trends. One of the things that we're trying to grapple with more this year is trying to get the technologists who don't know anything about journalism to listen to journalists. I don't just mean to provide a specific tool but just generally, to say: here's our first-order problem, is that of interest to you?

N. MAMO: That's the reason why I asked for this interview. Out of hundreds of papers that I've reviewed, I can only name one that actually got feedback from a journalist, and they got it after the solution was done [150], similar to this interview, at the end of the day. That's excluding Reuters Tracer [129; 130], which is a special case because it was developed in-house to address existing needs.

C. BECKETT: I was literally having this conversation yesterday. We were trying to work out how we would construct this. Would it be that we would go and find five Google engineers, sit them down in a room and say "this is journalism"?

I remember doing this in San Francisco. I met someone who is a top expert on blockchain, some five years ago, and I sat down with her and she didn't know anything about journalism apart from being a normal person. So she said "what's your problem in journalism? What's your problem?" And I

said that the advertising's disappearing or we've got this problem with fake news. I said "could the blockchain help with this? Could the blockchain help with that?" And every time, she went "um, nah" or "you could do that, but I don't see why you'd do it; you could solve it much better with this other thing." So we went through this whole thing.

What was interesting about it was that this conversation was with somebody who knew the tech. They didn't know the journalism, which is a version of what you're talking about.

N. MAMO: It is, and the problem goes both ways, by the way. One of the problems that we had was defining what it means to understand the news, and I didn't find the solution in research, primarily. Primarily, it was inspired by the news. In fact, our definition of understanding is Who does What, Where and When, Why and How: tools you must be familiar with. So this challenge goes both ways. One last question: what is your final appeal to AI researchers with this in mind?

C. BECKETT: In the context of journalism, so this is quite different to AI researchers elsewhere, there is an interesting issue about the obsession with AI ethics. I think this applies to digital generally, and technology in general. Everyone says technology is neutral but often times—in the social sciences anyway—we take a kind of political approach to technology. We say—and this is a bit paradoxical—we say that there's a danger with this technology and we're implying that it's therefore dangerous. We're not so good—or certainly in academe—we're not so good at thinking about the consequences of potential application in a more general way: how it might change the journalistic practice. That's the thing I look at.

So my appeal would be: help me, or help is when the people who know about the technology—which I don't properly—can help think through those kind of applications. I think that's the interesting research question. It's not just how this can help journalism—that's the first one: how can this particular technology help journalism? The second one is: how might it change journalism? And don't always frame it as a good and a bad [thing]; don't always frame it in terms of losing jobs or making mistakes.

D

Data

This appendix includes details about the tweet corpora used throughout this work. We collected tweets using the Tweepy [253] library for Python, which we wrapped in a custom collection tool in the `NicholasMamo/EvenTDT` library. We have made the corpora available in the `NicholasMamo/phd-data` repository.

Throughout this dissertation, we often refer to a general corpus. Most notably, we use the general corpus in Chapter 3 to resolve participants and in Chapter 4 to contrast event domains with everything else. The general corpus is a dataset of 457,429 English tweets, which we collected over 12 hours between 11 April 2020 at 21:35 and 12 April 2020 at 9:35 using Twitter's Sample API. For all other datasets, we used Twitter's Filter API to collect tweets mentioning certain keywords. The rest of this appendix describes, in detail, the datasets used in this dissertation: when and how we collected them, and any other notable characteristics.

D.1 | Data used in Chapter 3

In Chapter 3, we extract participants from event domains. Our contributions follow our previous work, the six-step APD framework [144]. In our original implementation, the resolver and extrapolator compared the prospective participants with the domain, which involved a TF-ICF term-weighting scheme. To construct the term-weighting scheme, we use the general corpus, which we describe at the beginning of this appendix.

Differently from other chapters, we start Chapter 3 with a short experiment. In Table 3.1 on page 31, we evaluate two NER models on six football match datasets. Since we re-used the corpora from Chapter 5, we do not describe them here but in Appendix D.3. Instead, we focus on the datasets from Section 3.3.

Data used in Section 3.3

In Section 3.3, we collected datasets slightly differently than in other chapters. Other chapters required us to collect datasets from two periods: the understanding period, before the event starts, and the actual event period. In Chapter 3, however, we collected data only during the understanding period to simulate extracting the Who and the Where with no knowledge of the event itself.

For most of Section 3.3, we focused on the domain of football matches. We collected datasets for an hour, starting 75 minutes before the events started. During this period, teams released line-ups, allowing us to understand a little better Who would be participating and Where. We left 15 minutes between the understanding period and the event period, as if to give the APD process time to conclude before the match started. We collected datasets using the event hashtag, and the names and common references to the two teams. Details about each match follow in the next table.

Event	Date	Time	Tweets	Keywords
Juventus - Inter	3 Apr 2022	19:30–20:30	3,803	#JuveInter, Juventus, Juve, Inter
Crystal Palace - Arsenal	4 Apr 2022	19:45–20:45	20,105	#CRYARS, Crystal Palace, Palace, Arsenal
Manchester City - Atlético	5 Apr 2022	19:45–20:45	8,753	#MCIATM, Manchester City, Atleti, Atletico Madrid
Burnley - Everton	6 Apr 2022	19:15–20:15	6,021	#BUREVE, Burnley, Everton
Watford - Leeds	9 Apr 2022	14:45–15:45	4,550	#WATLEE, Watford, Leeds
Aston Villa - Tottenham	9 Apr 2022	17:15–18:15	14,372	#AVLTOT, Aston Villa, Tottenham, Spurs
Manchester City - Liverpool	10 Apr 2022	16:15–17:15	41,251	#MCILIV, Manchester City, Liverpool
Real Madrid - Chelsea	12 Apr 2022	19:45–20:45	43,158	#RMACHE, Real Madrid, Chelsea
Newcastle - Leicester	17 Apr 2022	14:00–15:00	3,950	#NEWLEI, Newcastle, Leicester
Liverpool - Manchester United	19 Apr 2022	19:45–20:45	63,549	#LIVMUN, Liverpool, Manchester United
			209,512	

Table D.1: The football match datasets used in Section 3.3.

Conversely, Formula 1 Grands Prix have no line-ups—the drivers and constructors only change in-between seasons. The domain thus has no clear window when to collect datasets. In this dissertation, we followed an identical process as in football matches: we collected tweets over an hour, starting 75 minutes before the formation lap. Details about each Grand Prix from the first part of the 2022 season follow in the next table.

Event	Date	Time	Tweets	Keywords
Australian GP	10 Apr 2022	05:45–06:45	3,011	#AustralianGP, Formula 1, Formula One
Imola GP	24 Apr 2022	13:45–14:45	12,375	#ImolaGP, #F1, Formula 1, Formula One
Spanish GP	22 May 2022	13:45–14:45	12,067	#SpanishGP, Formula 1, Formula One
Monaco GP	29 May 2022	13:45–14:45	15,686	#SpanishGP, Formula 1, Formula One
Azerbaijan GP	12 Jun 2022	11:45–12:45	8,091	#AzerbaijanGP, Formula 1, Formula One
Canadian GP	Jun, 19 2022	18:45–19:45	15,065	#CanadianGP, Formula 1, Formula One
British GP	Jul, 3 2022	14:45–15:45	19,159	#BritishGP, Formula 1, Formula One
			85,454	

Table D.2: The Formula 1 datasets used in Section 3.3.

We conclude Section 3.3 with a short analysis on the 2021 Canadian federal election. We collected the dataset over 24 hours, between 13:00 on 20 September 2021 and 13:00 on 21 September 2021, covering the period when Canadians voted and when the election's results became known. In total, we collected 410,749 tweets, but in the analysis we only used a random sample: 82,150 tweets, or 20% of the data. We generated the sample using the `shuf` bash command-line utility.

Date	Time	Tweets	Keywords
Sep 20–21, 2022	13:00–13:00	82,150	#CanadaElection, #CanadaElection2021, #CanadaVotes, #ItsOurVote, #polcan, #CdnPoli, #Elxn44, Trudeau, O'Toole, Blanchet, Jagmeet Singh, Annamie Paul, Maxime Bernier, Canada election, Canadian election, Canada elections

Table D.3: The 2021 Canadian federal election datasets used in Section 3.3.

D.2 | Data used in Chapter 4

Throughout Chapter 4, we often refer to general corpora in the context of ATE techniques. We use a general corpus to calculate EVATE's ICF component, and many of our ATE baselines also use the same dataset. The general corpus is the sample collection of tweets that we describe at the beginning of this appendix.

Data used in Section 4.3

For the ATE evaluation of Section 4.3, we collected tweets from 24 football matches. We refer to each match by its event hashtag, which includes shortened versions of the two

teams. For example, *#BVBS04* refers to two German teams: Borussia Dortmund, *BVB*, and Schalke 04, *S04*.

Each football match dataset includes two parts: an understanding period and an event period. The understanding period, which we consider to last an hour, precedes the football match. The event period, then, starts shortly before the match and ends a little after the match's scheduled end. Only ELD, described in Section 2.1, uses the understanding period to prioritise keywords that appear intensely during the match but not before. In total, we collected more than 4.5 million tweets, averaging 192,426 tweets per match. Details about each dataset follow in the next table.

		Time		Tweets	
Event	**Date**	**Understanding**	**Event**	**Understanding**	**Event**
Dortmund - Schalke 04	16 May 2020	14:15–15:15	15:15–17:30	13,549	169,208
Dortmund - Bayern Munich	26 May 2020	17:15–18:15	18:15–20:30	9,444	107,536
Aston Villa - Sheffield United	17 Jun 2020	17:45–18:45	18:45–21:00	10,185	117,400
Manchester City - Arsenal	17 Jun 2020	20:00–21:00	21:05–23:20	48,617	334,557
Tottenham - Manchester United	19 Jun 2020	20:00–21:00	21:00–23:15	33,785	294,775
Brighton - Arsenal	20 Jun 2020	14:45–15:45	15:45–18:00	18,140	174,742
Aston Villa - Chelsea	21 Jun 2020	16:00–17:00	17:00–19:15	26,755	196,205
Leicester - Chelsea	28 Jun 2020	15:50–16:50	16:45–19:10	16,626	112,463
Barcelona - Atlético	30 Jun 2020	20:45–21:45	21:45–00:10	6,889	157,579
Everton - Leicester City	01 Jul 2020	17:45–18:45	18:45–21:10	10,358	86,060
Wolves - Arsenal	04 Jul 2020	17:15–18:15	18:15–20:40	26,467	152,288
Aston Villa - Manchester United	09 Jul 2020	20:00–21:00	21:00–23:20	16,044	268,149
Liverpool - Burnley	11 Jul 2020	14:45–15:45	15:45–18:05	9,491	75,782
Arsenal - Liverpool	15 Jul 2020	20:00–21:00	21:00–23:20	26,498	241,731
Tottenham - Leicester	19 Jul 2020	15:45–16:45	16:45–19:05	4,006	110,317
Juventus - Lyon	07 Aug 2020	19:45–20:45	20:45–23:05	6,162	140,839
Barcelona - Napoli	08 Aug 2020	19:45–20:45	20:45–23:05	11,223	206,740
Bayern Munich - Chelsea	08 Aug 2020	19:45–20:45	20:45–23:05	34,317	197,189
Atalanta - Paris Saint-Germain	12 Aug 2020	19:45–20:45	20:45–23:05	9,318	226,661
Leipzig - Atlético	13 Aug 2020	19:45–20:45	20:45–23:05	7,266	97,959
Manchester City - Lyon	15 Aug 2020	19:45–20:45	20:45–23:05	12,103	194,865
Leipzig - Paris Saint-Germain	18 Aug 2020	19:45–20:45	20:45–23:05	14,015	163,877
Lyon - Bayern Munich	19 Aug 2020	19:45–20:45	20:45–23:05	14,361	241,349
Sevilla - Inter	21 Aug 2020	19:45–20:45	20:45–23:05	12,391	151,940

Event	Date	Time		Tweets	
		Understanding	Event	Understanding	Event
				398,010	4,220,211

Table D.4: The football match datasets used in Section 4.3.

To collect the datasets, we used Twitter's Filter API, which allowed us to collect tweets mentioning certain keywords. During the understanding period, we collected tweets mentioning the event hashtag, colloquial versions of the team names and any other references to the match; the hashtag *#Revierderby*, for example, refers to the derby between Borussia Dortmund and Schalke 04. During the event period, we also collected tweets mentioning the names of the stadium, coaches and players, including substitutes. The tracking keywords that we used to collect tweets during the understanding and event periods of each match follow in the next table.

Period	Keywords
	Dortmund - Schalke 04
Understanding	#BVBS04, #Revierderby, Borussia, Dortmund, Schalke
Event	#BVBS04, #Revierderby, Borussia, Dortmund, BVB, Schalke, Signal Iduna Park, Favre, Wagner, Burki, Hakimi, Delaney, Dahoud, Guerreiro, Hummels, Akanji, Haaland, Brandt, Piszczek, Reyna, Sancho, Goetze, Balerdi, Morey, Hazard, Schmelzer, Hitz, Raschl, Schubert, McKennie, Nastasic, Serdar, Raman, Caligiuri, Jonjoe Kenny, Todibo, Oczipka, Harit, Salif Sane, Miranda, Gregoritsch, Matondo, Kutucu, Burgstaller, Schoepf, Becker, Nuebel, Mercan
	Dortmund - Bayern Munich
Understanding	#BVBFCB, BVB, Dortmund, Bayern, Munich
Event	#BVBFCB, BVB, Dortmund, Bayern, Munich, Signal Iduna Park, Favre, Hans Flick, Burki, Piszczek, Hummels, Akanji, Hakimi, Delaney, Dahoud, Guerreiro, Thorgan Hazard, Brandt, Haaland, Sancho, Gotze, Balerdi, Morey, Emre Can, Witsel, Schmelzer, Reyna, Hitz, Neuer, Pavard, Boateng, Alaba, Davies, Kimmich, Goretzka, Coman, Muller, Gnabry, Lewandowski, Odriozola, Javi Martinez, Cuisance, Perisic, Lucas Hernandez, Ulreich, Lukas Mai, Meier, Zirkzee
	Aston Villa - Sheffield United
Understanding	#AVLSHU, Villa, Sheffield
Event	#AVLSHU, Villa, Sheffield, Villa Park, Dean Smith, Chris Wilder, Nyland, Konsa, Hause, Mings, Targett, Hourihane, Douglas Luiz, McGinn, Ghazi, Davis, Grealish, Baston, Vassilev, Neil Taylor, Nakamba, Trezeguet, Samatta, Jota, Mohamady, Reina, Dean Henderson, Basham, Egan, Jack Robinson, Baldock, Lundstram, Norwood, Berge, Stevens, McBurnie, Billy Sharp, Luke Freeman, Jagielka, McGoldrick, Kieron Freeman, Mousset, Osborn, Clarke, Moore, Rodwel
	Manchester City - Arsenal
Understanding	#MCIARS, Manchester City, Arsenal

Period	Keywords
Event	#MCIARS, Manchester City, Arsenal, Etihad, Guardiola, Arteta, Ederson, Kyle Walker, Eric Garcia, Laporte, Mendy, David Silva, Gundogan, Bruyne, Mahrez, Gabriel Jesus, Sterling, Aguero, Zinchenko, Rodri, Leroy Sane, Bernardo Silva, Fernandinho, Otamendi, Carson, Foden, Leno, Bellerin, Mari, Mustafi, Tierney, Xhaka, Guendouzi, Nketiah, Willock, Saka, Aubameyang, Ceballos, Lacazette, Maitland-Niles, Nicolas Pepe, David Luiz, Reiss Nelson, Emiliano Martinez, Kolasinac, Martinelli
	Tottenham - Manchester United
Understanding	#TOTMUN, Tottenham, Manchester United
Event	#TOTMUN, Tottenham, Manchester United, Tottenham Hotspur Stadium, Mourinho, Solskjaer, Lloris, Aurier, Davinson Sanchez, Dier, Davies, Sissoko, Harry Winks, Heung-Min Son, Heung Min Son, Lamela, Bergwijn, Kane, Alderweireld, Vertonghen, Celso, Sessegnon, Gazzaniga, Ndombele, Skipp, Gedson Fernandes, Harvey White, De Gea, Bissaka, Wan-Bissaka, Lindelof, Maguire, Shaw, McTominay, Fred, Daniel James, Fernandes, Rashford, Martial, Bailly, Pogba, Mata, Lingard, Romero, Ighalo, Greenwood, Matic, Brandon Williams
	Brighton - Arsenal
Understanding	#BHAARS, Brighton, Arsenal
Event	#BHAARS, Brighton, Arsenal, The American Express Community Stadium, The Amex, Graham Potter, Arteta, Mathew Ryan, Schelotto, Webster, Lewis Dunk, Daniel Burn, Propper, Bissouma, Mooy, Pascal Gross, Trossard, Maupay, Lamptey, Duffy, Stephens, Mac Allister, Murray, Solly March, Montoya, David Button, Connolly, Leno, Bellerin, Mustafi, Rob Holding, Kolasinac, Pepe, Ceballos, Guendouzi, Saka, Lacazette, Aubameyang, Tierney, Ozil, Maitland-Niles, Reiss Nelson, Emiliano Martinez, Willock, Nketiah, Martinelli, Zech Medley
	Aston Villa - Chelsea
Understanding	#AVLCHE, Villa, Chelsea
Event	#AVLCHE, Villa, Chelsea, Villa Park, Dean Smith, Lampard, Nyland, Konsa, Mings, Hause, Targett, Hourihane, Douglas Luiz, McGinn, El-Ghazi, Davis, Grealish, Neil Taylor, Nakamba, Trezeguet, Baston, Samatta, Jota, Mohamady, Reina, Vassilev, Kepa, Azpilicueta, Rudiger, Christensen, Alonso, Loftus-Cheek, Kante, Kovacic, Willian, Giroud, Mason Mount, Barkley, Abraham, Pedro, Caballero, Zouma, Pulisic, Reece James, Emerson, Gilmour
	Leicester - Chelsea
Understanding	#LEICHE, Leicester, Chelsea
Event	#LEICHE, Leicester, Chelsea, King Power Stadium, Rodgers, Lampard, Schmeichel, James Justin, Evans, Soyuncu, Chilwell, Ndidi, Perez, Praet, Tielemans, Barnes, Vardy, Wes Morgan, Demarai Gray, Albrighton, Danny Ward, Iheanacho, Choudhury, Mendy, Fuchs, Bennett, Caballero, Reece James, Rudiger, Zouma, Emerson, Kante, Gilmour, Willian, Mount, Pulisic, Abraham, Kepa, Marcos Alonso, Jorginho, Barkley, Pedro, Loftus-Cheek, Kovacic, Giroud, Azpilicueta
	Barcelona - Atlético
Understanding	#BarcaAtleti, Barca, Barcelona, Atleti, Atletico

Period	Keywords
Event	#BarcaAtleti, Barca, Barcelona, Atleti, Atletico, Camp Nou, Setien, Simeone, ter Stegen, Semedo, Pique, Lenglet, Alba, Rakitic, Busquets, Vidal, Messi, Suarez, Griezmann, Arthur, Neto, Braithwaite, Sergi Roberto, Umtiti, Firpo, Pena, Puig, Collado, Fati, Araujo, Monchu, Oblak, Arias, Felipe, Gimenez, Lodi, Hector Herrera, Saul, Felix, Llorente, Lemar, Diego Costa, Adan, Partey, Morata, Correa, Saponjic, Vitolo, Carrasco, Hermoso, Trippier, Ricard Sanchez, Manuel Sanchez, Alvaro Garcia
	Everton - Leicester City
Understanding	#EVELEI, Everton, Leicester
Event	#EVELEI, Everton, Leicester, Goodison Park, Ancelotti, Rodgers, Pickford, Coleman, Keane, Holgate, Digne, Iwobi, Andre Gomes, Sigurdsson, Anthony Gordon, Calvert-Lewin, Richarlison, Baines, Mina, Bernard, Stekelenburg, Tom Davies, Kean, Virginia, Branthwaite, Baningime, Schmeichel, James Justin, Evans, Soyuncu, Chilwell, Ndidi, Albrighton, Tielemans, Praet, Barnes, Vardy, Wes Morgan, Demarai Gray, Maddison, Danny Ward, Iheanacho, Ayoze Perez, Choudhury, Mendy, Fuchs
	Dortmund - Schalke 04
Understanding	#BVBS04, #Revierderby, Borussia, Dortmund, Schalke
Event	#BVBS04, #Revierderby, Borussia, Dortmund, BVB, Schalke, Signal Iduna Park, Favre, Wagner, Burki, Hakimi, Delaney, Dahoud, Guerreiro, Hummels, Akanji, Haaland, Brandt, Piszczek, Reyna, Sancho, Goetze, Balerdi, Morey, Hazard, Schmelzer, Hitz, Raschl, Schubert, McKennie, Nastasic, Serdar, Raman, Caligiuri, Jonjoe Kenny, Todibo, Oczipka, Harit, Salif Sane, Miranda, Gregoritsch, Matondo, Kutucu, Burgstaller, Schoepf, Becker, Nuebel, Mercan
	Dortmund - Bayern Munich
Understanding	#BVBFCB, BVB, Dortmund, Bayern, Munich
Event	#BVBFCB, BVB, Dortmund, Bayern, Munich, Signal Iduna Park, Favre, Hans Flick, Burki, Piszczek, Hummels, Akanji, Hakimi, Delaney, Dahoud, Guerreiro, Thorgan Hazard, Brandt, Haaland, Sancho, Gotze, Balerdi, Morey, Emre Can, Witsel, Schmelzer, Reyna, Hitz, Neuer, Pavard, Boateng, Alaba, Davies, Kimmich, Goretzka, Coman, Muller, Gnabry, Lewandowski, Odriozola, Javi Martinez, Cuisance, Perisic, Lucas Hernandez, Ulreich, Lukas Mai, Meier, Zirkzee
	Aston Villa - Sheffield United
Understanding	#AVLSHU, Villa, Sheffield
Event	#AVLSHU, Villa, Sheffield, Villa Park, Dean Smith, Chris Wilder, Nyland, Konsa, Hause, Mings, Targett, Hourihane, Douglas Luiz, McGinn, Ghazi, Davis, Grealish, Baston, Vassilev, Neil Taylor, Nakamba, Trezeguet, Samatta, Jota, Mohamady, Reina, Dean Henderson, Basham, Egan, Jack Robinson, Baldock, Lundstram, Norwood, Berge, Stevens, McBurnie, Billy Sharp, Luke Freeman, Jagielka, McGoldrick, Kieron Freeman, Mousset, Osborn, Clarke, Moore, Rodwel
	Manchester City - Arsenal
Understanding	#MCIARS, Manchester City, Arsenal

Period	Keywords
Event	#MCIARS, Manchester City, Arsenal, Etihad, Guardiola, Arteta, Ederson, Kyle Walker, Eric Garcia, Laporte, Mendy, David Silva, Gundogan, Bruyne, Mahrez, Gabriel Jesus, Sterling, Aguero, Zinchenko, Rodri, Leroy Sane, Bernardo Silva, Fernandinho, Otamendi, Carson, Foden, Leno, Bellerin, Mari, Mustafi, Tierney, Xhaka, Guendouzi, Nketiah, Willock, Saka, Aubameyang, Ceballos, Lacazette, Maitland-Niles, Nicolas Pepe, David Luiz, Reiss Nelson, Emiliano Martinez, Kolasinac, Martinelli
	Tottenham - Manchester United
Understanding	#TOTMUN, Tottenham, Manchester United
Event	#TOTMUN, Tottenham, Manchester United, Tottenham Hotspur Stadium, Mourinho, Solskjaer, Lloris, Aurier, Davinson Sanchez, Dier, Davies, Sissoko, Harry Winks, Heung-Min Son, Heung Min Son, Lamela, Bergwijn, Kane, Alderweireld, Vertonghen, Celso, Sessegnon, Gazzaniga, Ndombele, Skipp, Gedson Fernandes, Harvey White, De Gea, Bissaka, Wan-Bissaka, Lindelof, Maguire, Shaw, McTominay, Fred, Daniel James, Fernandes, Rashford, Martial, Bailly, Pogba, Mata, Lingard, Romero, Ighalo, Greenwood, Matic, Brandon Williams
	Brighton - Arsenal
Understanding	#BHAARS, Brighton, Arsenal
Event	#BHAARS, Brighton, Arsenal, The American Express Community Stadium, The Amex, Graham Potter, Arteta, Mathew Ryan, Schelotto, Webster, Lewis Dunk, Daniel Burn, Propper, Bissouma, Mooy, Pascal Gross, Trossard, Maupay, Lamptey, Duffy, Stephens, Mac Allister, Murray, Solly March, Montoya, David Button, Connolly, Leno, Bellerin, Mustafi, Rob Holding, Kolasinac, Pepe, Ceballos, Guendouzi, Saka, Lacazette, Aubameyang, Tierney, Ozil, Maitland-Niles, Reiss Nelson, Emiliano Martinez, Willock, Nketiah, Martinelli, Zech Medley
	Aston Villa - Chelsea
Understanding	#AVLCHE, Villa, Chelsea
Event	#AVLCHE, Villa, Chelsea, Villa Park, Dean Smith, Lampard, Nyland, Konsa, Mings, Hause, Targett, Hourihane, Douglas Luiz, McGinn, El-Ghazi, Davis, Grealish, Neil Taylor, Nakamba, Trezeguet, Baston, Samatta, Jota, Mohamady, Reina, Vassilev, Kepa, Azpilicueta, Rudiger, Christensen, Alonso, Loftus-Cheek, Kante, Kovacic, Willian, Giroud, Mason Mount, Barkley, Abraham, Pedro, Caballero, Zouma, Pulisic, Reece James, Emerson, Gilmour
	Leicester City - Chelsea
Understanding	#LEICHE, Leicester, Chelsea
Event	#LEICHE, Leicester, Chelsea, King Power Stadium, Rodgers, Lampard, Schmeichel, James Justin, Evans, Soyuncu, Chilwell, Ndidi, Perez, Praet, Tielemans, Barnes, Vardy, Wes Morgan, Demarai Gray, Albrighton, Danny Ward, Iheanacho, Choudhury, Mendy, Fuchs, Bennett, Caballero, Reece James, Rudiger, Zouma, Emerson, Kante, Gilmour, Willian, Mount, Pulisic, Abraham, Kepa, Marcos Alonso, Jorginho, Barkley, Pedro, Loftus-Cheek, Kovacic, Giroud, Azpilicueta
	Barcelona - Atlético de Madrid
Understanding	#BarcaAtleti, Barca, Barcelona, Atleti, Atletico

Period	Keywords
Event	#BarcaAtleti, Barca, Barcelona, Atleti, Atletico, Camp Nou, Setien, Simeone, ter Stegen, Semedo, Pique, Lenglet, Alba, Rakitic, Busquets, Vidal, Messi, Suarez, Griezmann, Arthur, Neto, Braithwaite, Sergi Roberto, Umtiti, Firpo, Pena, Puig, Collado, Fati, Araujo, Monchu, Oblak, Arias, Felipe, Gimenez, Lodi, Hector Herrera, Saul, Felix, Llorente, Lemar, Diego Costa, Adan, Partey, Morata, Correa, Saponjic, Vitolo, Carrasco, Hermoso, Trippier, Ricard Sanchez, Manuel Sanchez, Alvaro Garcia
	Everton - Leicester City
Understanding	#EVELEI, Everton, Leicester
Event	#EVELEI, Everton, Leicester, Goodison Park, Ancelotti, Rodgers, Pickford, Coleman, Keane, Holgate, Digne, Iwobi, Andre Gomes, Sigurdsson, Anthony Gordon, Calvert-Lewin, Richarlison, Baines, Mina, Bernard, Stekelenburg, Tom Davies, Kean, Virginia, Branthwaite, Baningime, Schmeichel, James Justin, Evans, Soyuncu, Chilwell, Ndidi, Albrighton, Tielemans, Praet, Barnes, Vardy, Wes Morgan, Demarai Gray, Maddison, Danny Ward, Iheanacho, Ayoze Perez, Choudhury, Mendy, Fuchs
	Wolves - Arsenal
Understanding	#WOLARS, Wolves, Arsenal
Event	#WOLARS, Wolves, Arsenal, Molineux, Nuno Espirito Santo, Arteta, Patricio, Boly, Coady, Saiss, Doherty, Dendoncker, Neves, Moutinho, Jonny, Adama, Jimenez, Jordao, Neto, Gibbs-White, Jota, Ruddy, Vinagre, Campana, Kilman, Buur, Martinez, Mustafi, David Luiz, Kolasinac, Cedric, Ceballos, Xhaka, Tierney, Aubameyang, Nketiah, Saka, Bellerin, Sokratis, Lacazette, Torreira, Maitland-Niles, Rob Holding, Nelson, Willock, Matt Macey
	Aston Villa - Manchester United
Understanding	#AVLMUN, Villa, Manchester United
Event	#AVLMUN, Villa, Manchester United, Villa Park, Dean Smith, Solskjaer, Reina, Konsa, Hause, Mings, Neil Taylor, El-Ghazi, Douglas Luiz, McGinn, Trezeguet, Grealish, Samatta, Lansbury, Nakamba, Hourihane, Jota, Guilbert, Nyland, El Mohamady, Vassilev, Davis, De Gea, Wan-Bissaka, Wan Bissaka, Lindelof, Maguire, Luke Shaw, Pogba, Matic, Greenwood, Bruno, Rashford, Martial, Bailly, Mata, Pereira, Fred, Daniel James, Romero, Ighalo, McTominay, Brandon Williams
	Liverpool - Burnley
Understanding	#LIVBUR, Liverpool, Burnley
Event	#LIVBUR, Liverpool, Burnley, Anfield, Klopp, Dyche, Alisson, Neco Williams, Gomez, van Dijk, Robertson, Wijnaldum, Fabinho, Curtis Jones, Salah, Firmino, Mane, Lovren, Keita, Adrian, Oxlade-Chamberlain, Minamino, Shaqiri, Origi, Alexander-Arnold, Elliott, Nick Pope, Bardsley, Kevin Long, Tarkowski, Charlie Taylor, Pieters, Westwood, Brownhill, McNeil, Chris Wood, Jay Rodriguez, Gudmundsson, Robert Brady, Peacock-Farrell, Vydra, Max Thompson, Dunne, Benson, Goodridge, Driscoll-Glennon
	Arsenal - Liverpool
Understanding	#ARSLIV, Arsenal, Liverpool

Period	Keywords
Event	#ARSLIV, Arsenal, Liverpool, Emirates Stadium, Klopp, Arteta, Martinez, Rob Holding, David Luiz, Tierney, Cedric, Torreira, Xhaka, Reiss Nelson, Pepe, Lacazette, Saka, Bellerin, Sokratis, Ceballos, Aubameyang, Maitland-Niles, Mustafi, Willock, Kolasinac, Macey, Alisson, Alexander-Arnold, Gomez, van Dijk, Robertson, Oxlade-Chamberlain, Fabinho, Wijnaldum, Salah, Firmino, Mane, Lovren, Keita, Adrian, Minamino, Shaqiri, Origi, Curtis Jones, Elliott, Neco Williams
	Tottenham - Leicester
Understanding	#TOTLEI, Tottenham, Leicester
Event	#TOTLEI, Tottenham, Leicester, Tottenham Hotspur Stadium, Mourinho, Rodgers, Lloris, Aurier, Davinson Sanchez, Alderweireld, Ben Davies, Sissoko, Harry Winks, Lo Celso, Moura, Kane, Heung-Min Son, Vertonghen, Lamela, Sessegnon, Foyth, Gazzaniga, Bergwijn, Skipp, Gedson Fernandes, Tanganga, Schmeichel, Bennett, Wes Morgan, Evans, James Justin, Ndidi, Tielemans, Luke Thomas, Ayoze Perez, Vardy, Barnes, Demarai Gray, Danny Ward, Iheanacho, Choudhury, Matthew James, Mendy, Praet, Johnson, Hirst
	Juventus - Lyon
Understanding	#JuveOL, Juventus, Lyon
Event	#JuveOL, Juventus, Lyon, Allianz Stadium, Sarri, Rudi Garcia, Szczesny, Cuadrado, Bonucci, de Ligt, Alex Sandro, Bentancur, Pjanic, Rabiot, Bernardeschi, Higuain, Ronaldo, Chiellini, Ramsey, Dybala, Danilo, Matuidi, Rugani, Demiral, Pinsoglio, Olivieri, Muratore, Buffon, Lopes, Denayer, Marcelo, Marcal, Guimaraes, Dubois, Caqueret, Aouar, Maxwel Cornet, Depay, Toko Ekambi, Diomande, Andersen, Rafael, Dembele, Traore, Mendes, Reine-Adelaide, Jean Lucas, Tete, Tatarusanu, Melvin Bard, Cherki
	Barcelona - Napoli
Understanding	#BarcaNapoli, Barcelona, Barca, Napoli
Event	#BarcaNapoli, Barcelona, Barca, Napoli, Camp Nou, Setien, Gattuso, ter Stegen, Semedo, Pique, Lenglet, Alba, Sergi Roberto, Rakitic, de Jong, Messi, Suarez, Griezmann, Neto, Firpo, Pena, Puig, Fati, Araujo, Monchu, Mingueza, de la Fuente, Reis, Orellana, Ospina, Di Lorenzo, Manolas, Koulibaly, Mario Rui, Fabian, Demme, Zielinski, Callejon, Mertens, Insigne, Meret, Allan, Llorente, Lozano, Elmas, Luperto, Maksimovic, Politano, Hysaj, Karnezis, Lobotka, Milik
	Bayern Munich - Chelsea
Understanding	#BAYCHE, Bayern, Chelsea
Event	#BAYCHE, Bayern, Chelsea, Allianz Arena, Flick, Lampard, Neuer, Kimmich, Boateng, Alaba, Davies, Goretzka, Alcantara, Gnabry, Thomas Muller, Perisic, Lewandowski, Odriozola, Sule, Javier Martinez, Coutinho, Cuisance, Lucas Hernandez, Tolisso, Ulreich, Hoffmann, Tillman, Musiala, Arrey-Mbi, Caballero, Reece James, Christensen, Zouma, Emerson, Barkley, Kante, Kovacic, Hudson-Odoi, Tammy Abraham, Mason Mount, Kepa, Rudiger, Giroud, Batshuayi, Tomori, Cumming, Broja, Henry Lawrence, Maatsen, Bate, Simeu
	Atalanta - Paris Saint-Germain
Understanding	#ATAPSG, Atalanta, PSG

Period	Keywords
Event	#ATAPSG, Atalanta, PSG, Estadio da Luz, Gasperini, Tuchel, Sportiello, Toloi, Caldara, Djimsiti, Hateboer, de Roon, Freuler, Gosens, Pasalic, Gomez, Zapata, Sutalo, Palomino, Czyborra, Muriel, Piccoli, Malinovsky, Da Riva, Castagne, Bellanova, Gelmi, Rossi, Colley, Navas, Kehrer, Thiago Silva, Kimpembe, Bernat, Ander Herrera, Marquinhos, Gueye, Sarabia, Icardi, Neymar, Mbappe, Daniel Paredes, Sergio Rico, Choupo-Moting, Diallo, Draxler, Bakker, Kalimuendo-Muinga, Bulka, Dagba, Mbe Soh, Ruiz-Atil
	Leipzig - Atlético
Understanding	#RBLATL, #RBLAtleti, #UCL, Leipzig, Atleti, Atletico Madrid
Event	#RBLATL, #RBLAtleti, #UCL, Leipzig, Atleti, Atletico Madrid, Estadio Jose Alvalade, Nagelsmann, Simeone, Gulacsi, Halstenberg, Upamecano, Klostermann, Angelino, Sabitzer, Kampl, Laimer, Nkunku, Poulsen, Olmo, Orban, Haidara, Forsberg, Tyler Adams, Lookman, Schick, Mukiele, Mvogo, Tschauner, Novoa, Borkowski, Wosz, Oblak, Trippier, Savic, Gimenez, Lodi, Carrasco, Saul, Hector Herrera, Koke, Llorente, Diego Costa, Adan, Arias, Partey, Joao Felix, Morata, Lemar, Saponjic, Felipe, Vitolo, Mario Hermoso, Manuel Sanchez, Moya
	Manchester City - Lyon
Understanding	#ManCityOL, #UCL, Manchester City, Lyon
Event	#ManCityOL, #UCL, Manchester City, Lyon, Estadio Jose Alvalade, Pep, Guardiola, Rudi Garcia, Ederson, Kyle Walker, Eric Garcia, Laporte, Cancelo, Fernandinho, Rodri, Ilkay Gundogan, De Bruyne, Gabriel Jesus, Raheem Sterling, Claudio Bravo, John Stones, Zinchenko, Bernardo Silva, David Silva, Mendy, Mahrez, Otamendi, Foden, Doyle, Palmer, Bernabe, Lopes, Denayer, Marcelo, Marcal, Dubois, Caqueret, Guimaraes, Aouar, Cornet, Toko Ekambi, Depay, Diomande, Andersen, Rafael, Dembele, Bertrand Traore, Thiago Mendes, Reine-Adelaide, Jean Lucas, Kenny Tete, Tatarusanu, Melvin Bard, Cherki
	Leipzig - Paris Saint-Germain
Understanding	#RBLPSG, #UCL, Leipzig, PSG
Event	#RBLPSG, #UCL, Leipzig, PSG, Estadio da Luz, Nagelsmann, Tuchel, Gulacsi, Klostermann, Upamecano, Mukiele, Laimer, Kampl, Sabitzer, Angelino, Olmo, Nkunku, Poulsen, Orban, Haidara, Forsberg, Tyler Adams, Lookman, Schick, Halstenberg, Mvogo, Tschauner, Novoa, Borkowski, Wosz, Rico, Kehrer, Thiago Silva, Kimpembe, Bernat, Ander Herrera, Marquinhos, Paredes, Di Maria, Mbappe, Neymar, Verratti, Choupo-Moting, Icardi, Sarabia, Kurzawa, Diallo, Draxler, Bakker, Gueye, Bulka, Dagba, Garissone Innocent
	Lyon - Bayern Munich
Understanding	#OLFCB, #UCL, Lyon, Bayern
Event	#OLFCB, #UCL, Lyon, Bayern, Estadio Jose Alvalade, Rudi Garcia, Flick, Anthony Lopes, Denayer, Marcelo, Marcal, Dubois, Caqueret, Guimaraes, Aouar, Cornet, Toko Ekambi, Depay, Diomande, Andersen, Rafael, Dembele, Traore, Mendes, Reine-Adelaide, Jean Lucas, Kenny Tete, Tatarusanu, Bard, Cherki, Neuer, Kimmich, Boateng, Alaba, Davies, Alcantara, Goretzka, Gnabry, Thomas Muller, Perisic, Lewandowski, Odriozola, Niklas Sule, Pavard, Javier Martinez, Coutinho, Cuisance, Lucas Hernandez, Tolisso, Ulreich, Coman, Zirkzee, Hoffmann
	Sevilla - Inter
Understanding	#SevillaInter, #UEL, #UELFinal, Inter, Sevilla

Period	Keywords
Event	#SevillaInter, #UEL, #UELFinal, Inter, Sevilla, RheinEnergieSTADION, Lopetegui, Conte, Bounou, Jesus Navas, Kounde, Diego Carlos, Reguilon, Joan Jordan, Fernando, Banega, Ocampos, de Jong, Suso, Vaclik, Sergi Gomez, El Haddadi, Gudelj, Escudero, Oliver Torres, Vazquez, Jose Alonso Lara, Javier Diaz, Genaro Rodriguez, Pablo Perez, En-Nesyri, Handanovic, Godin, de Vrij, Bastoni, D'Ambrosio, Barella, Brozovic, Gagliardini, Ashley Young, Lautaro, Lukaku, Alexis Sanchez, Victor Moses, Sensi, Ranocchia, Valero, Eriksen, Padelli, Esposito, Pirola, Biraghi, Skriniar, Candreva

Table D.5: The keywords used to collect the football match datasets used in Section 4.3.

Data used in Section 4.4

For the ATE evaluation of Section 4.4, we collected datasets from the 2020 Formula 1 season. Due to the COVID-19 pandemic, the 2020 season was a relatively short one, with just 17 Grands Prix—four fewer than in 2019. Over the course of the season, we collected data from 15 out of the 17 races. We missed the Belgian Grand Prix on 30 August 2020 due to human error in the collection, and the Russian Grand Prix on 27 September 2020 due to API downtime.

In the Formula 1 Grands Prix evaluations, we reverted to the same dataset model as in the football matches analyses, with an understanding period and an event period. The understanding period only covered 30 minutes, while the event period generally lasted between two hours and two hours and a half. We further extended the event period for Grands Prix that were suspended with red flags and resumed later. In total, we collected more than 2.2 million tweets, averaging 147,232 tweets per Grand Prix. Details about each Grand Prix follow in the next table.

		Time		Tweets	
Event	**Date**	**Understanding**	**Event**	**Understanding**	**Event**
Austrian GP	05 Jul 2020	14:25–14:55	14:55–17:25	17,285	197,166
Austrian GP (2)	12 Jul 2020	14:25–14:55	14:55–17:10	5,911	113,467
Hungarian GP	19 Jul 2020	14:25–14:55	14:55–17:10	10,170	98,014
British GP	02 Aug 2020	14:25–14:55	14:55–17:10	7,665	125,416
F1 70th Anniversary	09 Aug 2020	14:25–14:55	15:00–17:10	8,331	120,678
Spanish GP	16 Aug 2020	14:25–14:55	15:00–17:10	5,151	87,006
Belgian GP	30 Aug 2020				
Italian GP	Sep 06, 2020	14:25–14:55	14:55–17:10	10,232	178,050
Tuscan GP	Sep 13, 2020	14:25–14:55	14:55–17:40	8,790	177,615
Russian GP	Sep 27, 2020				

		Time		Tweets	
Event	**Date**	**Understanding**	**Event**	**Understanding**	**Event**
Eifel GP	Oct 11, 2020	13:25–13:55	13:55–16:10	6,677	125,857
Portuguese GP	Oct 25, 2020	13:25–13:55	13:55–16:10	5,563	115,509
Imola GP	01 Nov 2020	12:25–12:55	12:55–15:10	3,614	102,168
Turkish GP	15 Nov 2020	10:25–10:55	10:55–13:10	7,789	119,922
Bahrain GP	29 Nov 2020	14:25–14:55	14:55–18:40	4,196	275,029
Sakhir GP	06 Dec 2020	17:25–17:55	17:55–20:10	5,428	180,268
Abu Dhabi GP	13 Dec 2020	13:25–13:55	13:55–16:10	8,474	77,036
				115,276	2,093,201

Table D.6: The Formula 1 datasets used in Section 4.4. Each denotes a red flag, which led to longer race times.

One major difference between football matches and Formula 1 Grands Prix is that the Formula 1 drivers and constructors are known ahead of each race. Therefore differently from the football match datasets, during the understanding period we tracked not just the event hashtag, such as *#AustrianGP* and *#F1*, but also all drivers and constructors. Consequently, ELD's TF-ICF term-weighting scheme could still gauge the popularity of drivers and constructors from the understanding period dataset.

Because the drivers and constructs are known ahead of the Grands Prix, we normally used the same keywords to collect datasets during the understanding and event periods. We only made a few changes to the tracking keywords between the two periods, always to rectify human errors. The tracking keywords used to collect tweets during the understanding and event periods of each Grand Prix follow in the next table.

Period	**Keywords**
	Austrian GP
Understanding	#AustrianGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
Event	#AustrianGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
	Austrian GP (2)

Period	Keywords
Understanding	#AustrianGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
Event	#AustrianGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
	Hungarian GP
Understanding	#HungarianGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
Event	#HungarianGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
	British GP
Understanding	#BritishGP, Silverstone, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
Event	#BritishGP, Silverstone, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell, Hulkenberg
	F1 70th Anniversary
Understanding	#F170, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell, Hulkenberg
Event	#F170, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell, Hulkenberg
	Spanish GP
Understanding	#SpanishGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell

Period	Keywords
Event	#SpanishGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
	Italian GP
Understanding	#ItalianGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
Event	#ItalianGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
	Tuscan GP
Understanding	#TuscanGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
Event	#TuscanGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
	Eifel GP
Understanding	#EifelGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell, Hulkenberg
Event	#EifelGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell, Hulkenberg
	Portuguese GP
Understanding	#PortugueseGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
Event	#PortugueseGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
	Imola GP

Period	Keywords
Understanding	#ImolaGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
Event	#ImolaGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
	Turkish GP
Understanding	#TurkishGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
Event	#TurkishGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
	Bahrain GP
Understanding	#BahrainGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
Event	#BahrainGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
	Sakhir GP
Understanding	#SakhirGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
Event	#SakhirGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell
	Abu Dhabi GP
Understanding	#AbuDhabiGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell

Period	Keywords
Event	#AbuDhabiGP, #F1, Alfa Romeo, AlphaTauri, Ferrari, Haas, McLaren, Mercedes, Racing Point, Red Bull Racing, Renault, Williams, Spielberg, Raikkonen, Giovinazzi, Gasly, Kvyat, Vettel, Leclerc, Grosjean, Magnussen, Lando, Norris, Carlos Sainz, Lewis Hamilton, Bottas, Sergio Perez, Checo Perez, Stroll, Albon, Verstappen, Ricciardo, Ocon, Latifi, George Russell

Table D.7: The keywords used to collect the Formula 1 datasets used in Section 4.4.

Data used in Section 4.5

The politics analyses too required a tailored data collection process. First, as we explained in Section 4.5, we considered each day to be a separate event. Second, we did not collect an understanding period for ELD. Instead, we used the general corpus described in the beginning of this appendix as ELD's understanding period to contrast what happens in general with what happens in politics.

The datasets span around three months, from 20 October 2020 to 21 January 2021. We only missed three days, between 12 January and 14 January 2021 due to server downtime. To assemble the dataset, we automatically tracked keywords from midnight to the next midnight, Central European time. Over these three months, we collected more than 85 million tweets, averaging 939,763 tweets per day.

The 91 days cover several important events from the US political scene. Early on, the datasets cover the lead-up to the presidential election, ballot casting (3 November 2020) and counting (until after 7 November 2020), and the day when news networks called the election for Joe Biden (7 November 2020). Later, the datasets also cover the electoral college casting its votes (14 December, 2020), the Capitol riot (6 January 2021) and President Joe Biden's Inauguration Day (20 January 2021). Details about how we collected each dataset follow in the next table.

Date	Tweets	Keywords
Oct 20, 2020	800,129	#elections, #Elections2020, #Election2020, Trump, Biden
Oct 21, 2020	816,396	#elections, #Elections2020, #Election2020, Trump, Biden
Oct 22, 2020	915,016	#elections, #Elections2020, #Election2020, Trump, Biden
Oct 23, 2020	1,021,130	#elections, #Elections2020, #Election2020, Trump, Biden
Oct 24, 2020	953,337	#elections, #Elections2020, #Election2020, Trump, Biden
Oct 25, 2020	891,355	#elections, #Elections2020, #Election2020, Trump, Biden
Oct 26, 2020	944,575	#elections, #Elections2020, #Election2020, Trump, Biden
Oct 27, 2020	1,008,524	#elections, #Elections2020, #Election2020, Trump, Biden
Oct 28, 2020	973,792	#elections, #Elections2020, #Election2020, Trump, Biden

Date	Tweets	Keywords
Oct 29, 2020	997,428	#elections, #Elections2020, #Election2020, Trump, Biden
Oct 30, 2020	1,039,370	#elections, #Elections2020, #Election2020, Trump, Biden
Oct 31, 2020	996,352	#elections, #Elections2020, #Election2020, Trump, Biden
01 Nov 2020	1,019,263	#elections, #Elections2020, #Election2020, Trump, Biden
02 Nov 2020	1,078,132	#elections, #Elections2020, #Election2020, Trump, Biden
03 Nov 2020	1,191,254	#elections, #Elections2020, #Election2020, #ElectionDay, Trump, Biden
04 Nov 2020	1,218,193	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
05 Nov 2020	1,138,791	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
06 Nov 2020	1,228,534	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
07 Nov 2020	1,231,376	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
08 Nov 2020	1,245,691	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
09 Nov 2020	1,170,033	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
10 Nov 2020	1,103,084	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
11 Nov 2020	1,116,066	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
12 Nov 2020	1,069,789	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
13 Nov 2020	1,048,658	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
14 Nov 2020	1,014,234	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
15 Nov 2020	956,601	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
16 Nov 2020	1,021,980	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
17 Nov 2020	983,282	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
18 Nov 2020	1,000,717	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
19 Nov 2020	960,814	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
20 Nov 2020	1,064,809	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
21 Nov 2020	993,736	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
22 Nov 2020	1,011,573	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
23 Nov 2020	946,513	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
24 Nov 2020	1,065,894	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
25 Nov 2020	1,027,134	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
26 Nov 2020	879,987	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
27 Nov 2020	891,475	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
28 Nov 2020	956,810	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
29 Nov 2020	866,458	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
30 Nov 2020	954,964	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
01 Dec 2020	983,740	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
02 Dec 2020	982,248	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden

Date	Tweets	Keywords
03 Dec 2020	962,032	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
04 Dec 2020	981,950	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
05 Dec 2020	889,500	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
06 Dec 2020	893,041	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
07 Dec 2020	851,002	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
08 Dec 2020	961,751	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
09 Dec 2020	986,540	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
10 Dec 2020	941,485	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
11 Dec 2020	986,838	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
12 Dec 2020	1,075,637	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
13 Dec 2020	924,343	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
14 Dec 2020	916,663	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
15 Dec 2020	1,054,386	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
16 Dec 2020	1,011,836	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
17 Dec 2020	869,909	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
18 Dec 2020	845,646	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
19 Dec 2020	751,313	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
20 Dec 2020	759,096	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
21 Dec 2020	769,415	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
22 Dec 2020	830,913	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
23 Dec 2020	883,521	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
24 Dec 2020	790,989	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
25 Dec 2020	507,839	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
26 Dec 2020	542,201	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
27 Dec 2020	641,878	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
28 Dec 2020	703,022	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
29 Dec 2020	736,368	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
30 Dec 2020	759,738	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
31 Dec 2020	690,442	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
01 Jan 2021	663,665	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
02 Jan 2021	724,707	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
03 Jan 2021	790,928	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
04 Jan 2021	835,254	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
05 Jan 2021	860,095	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
06 Jan 2021	1,014,673	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden

Date	Tweets	Keywords
07 Jan 2021	974,257	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
08 Jan 2021	1,035,643	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
09 Jan 2021	1,035,824	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
10 Jan 2021	942,608	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
11 Jan 2021	952,909	#elections, #Elections2020, #Election2020, #ElectionDay, #ElectionNight, Trump, Biden
12 Jan 2021		
13 Jan 2021		
14 Jan 2021		
15 Jan 2021	881,553	#Inauguration, #InaugurationDay, #Inauguration2021, Trump, Biden
16 Jan 2021	830,411	#Inauguration, #InaugurationDay, #Inauguration2021, Trump, Biden
17 Jan 2021	825,258	#Inauguration, #InaugurationDay, #Inauguration2021, Trump, Biden
18 Jan 2021	832,624	#Inauguration, #InaugurationDay, #Inauguration2021, Trump, Biden
19 Jan 2021	917,626	#Inauguration, #InaugurationDay, #Inauguration2021, Trump, Biden
20 Jan 2021	1,113,018	#Inauguration, #InaugurationDay, #Inauguration2021, Inauguration, Trump, Biden
21 Jan 2021	988,869	#Inauguration, #InaugurationDay, #Inauguration2021, Inauguration, Trump, Biden
	85,518,453	

Table D.8: The 2020 US presidential election datasets used in Section 4.5.

D.3 | Data used in Chapter 5

For the TDT analyses of Chapter 5, we largely followed the same approach as in our previous work [146]. SEER, like ELD, uses a corpus of tweets from before the event starts to construct the TF-ICF term-weighting scheme. Therefore in all cases, we collected a dataset before the event and another during the event. We collected nine events, of which we use six regularly in Section 5.3 and the other three exclusively in Appendix B. Details about the nine datasets and how we collected them follow in the next table.

		Time		Tweets	
Event	Date	Understanding	Event	Understanding	Event
Southampton - Arsenal	25 Jun 2020	17:45–18:45	18:45–21:00	15,135	97,874
Leicester - Manchester United	26 Jul 2020	15:45–16:45	16:45–19:05	34,253	209,132
Turkey - Italy	11 Jun 2021	19:45–20:45	20:45–23:00	28,022	109,888
Wales - Switzerland	12 Jun 2021	13:45–14:45	14:45–17:00	15,914	87,717
Scotland - Czech Republic	14 Jun 2021	13:45–14:45	14:45–17:00	17,390	120,194

		Time		Tweets	
Event	**Date**	**Understanding**	**Event**	**Understanding**	**Event**
Hungary - France	19 Jun 2021	13:45–14:45	14:45–17:00	9,886	122,069
Trophée des Champions	13 Jan 2021	19:45–20:45	20:45–23:05	886	10,163
Copa del Rey	03 Apr 2021	20:15–21:15	21:17–23:32	656	2,774
Parma - Milan	10 Apr 2021	16:45–17:45	17:45–19:00	1,098	7,943
				123,240	767,754

Table D.9: The football match datasets used in Section 5.3.

Although we used nine datasets in our analyses, we focus extensively on the first six. We only use the 2020 Trophée des Champions final between Paris Saint-Germain and Olympique de Marseille, the 2020 Copa Del Rey final between Athletic Club and Real Sociedad, and the match between Parma and Milan as real-life examples of low-coverage events in Appendix B. Here, too, we focus on the first six datasets.

While arguing about the need for parameter-free algorithms in IR, Keogh et al. [109] briefly referred to the dataset, which they called a "meta parameter" of the evaluation. The simplest manner of eliminating human bias from the meta parameter, and the one Keogh et al. [109] adopted, is to experiment with a variety of datasets and scenarios. Nevertheless, as we explained in Appendix A, TDT's manual evaluations hardly encourage such comprehensive analyses. Instead, in our evaluation we opted to observe our algorithms' behaviours in several scenarios by hand-picking the six events. The six matches had the following characteristics and key topics:

- The match between Southampton and Arsenal was quiet [54]. Early on, Arsenal's Eddie Nketiah had a goal disallowed for an offside in the build-up, before later pouncing on a mistake by Southampton's goalkeeper to open the scoring. In the last few minutes, Southampton's Jack Stephens received a red card and, shortly after, Arsenal's Joe Willock scored the second goal. The uncontroversial nature of the key topics facilitated precise topic detection, and all algorithms performed remarkably well.

- The match between Leicester City and Manchester United was agitated [237]. Manchester United's Bruno Fernandes had a goal disallowed for offside in the first half, but he scored a penalty in the second half. Towards the end, Leicester City's Jonny Evans received a red card for a reckless challenge before substitute Jesse Lingard scored a second, late goal for Manchester United. The match was filled with

opinions and reactions to an eventful match: Manchester United's record-setting 14^{th} penalty of the season, Lingard's first goal of the season in the last minute, and Manchester United's return to Champions League football.

- The match between Turkey and Italy was impassioned [172]. The two national teams met in the EURO 2020 opener, held in Summer of 2021 due to the COVID-19 pandemic. After a quiet first half, Italy scored three goals in the second half to win the match. The special occasion, the opening of an awaited major tournament, rallied Twitter, with the community sharing its unsolicited opinions and predictions. The Italians would go on to win the EURO 2020 tournament.

- The match between Wales and Switzerland was calm [171]. The Swiss scored first through Breel Embolo in the second half, but Wales' Kieffer Moore equalised a quarter of an hour later. Five minutes before the end of the match, Switzerland took the lead, but Mario Gavranović had strayed offside and the referee disallowed the goal. All algorithms captured very few opinions during the event and, consequently, achieved relatively-high precision scores.

- The match between Scotland and the Czech Republic was remarkable [170]. The match was quiet for a long time, even after Patrik Schick opened the scoring for the Czech Republic just before half-time, but Twitter's engagement increased drastically in the second half. Schick picked up a loose ball in midfield, shot from the halfway line and scored his second. The goal, which later won the EURO 2020 goal of the tournament, created a long-tailed event shadow, misleading all algorithms into detecting redundant topics.

- The match between Hungary and France was unexpected [239]. France, then reigning World Cup champions, had just defeated Germany in the first match of the EURO 2020 tournament. Hungary, however, took a surprise lead at the end of the first half through Attila Fiola, whose goal cast an event shadow that persisted throughout half-time. France's Antoine Griezmann equalised in the second half, but their struggles led to observers expressing their opinions on the match and criticising the French coach's decisions. The draw foreshadowed France's difficulties in the tournament and an unceremonious exit against Switzerland.

A detailed breakdown of the number and types of topics in each event follow in the next table.

Event	Goals	Cards	Halves	Substitutions	Total
Southampton - Arsenal	3	3	4	4	14
Leicester - Manchester United	3	2	4	9	18
Turkey - Italy	3	3	4	8	18
Wales - Switzerland	3	7	4	9	23
Scotland - Czech Republic	2	0	4	10	16
Hungary - France	2	2	4	7	15
Trophée des Champions	1	3	4	8	16
Copa del Rey	4	9	4	9	26
Parma - Milan	5	5	4	10	24
	26	34	36	74	170

Table D.10: The ground truth topics in the football match datasets used in Section 5.3.

Similarly to our previous work [146], with the line-ups still unknown, during the understanding period we could only collect tweets that mentioned the event hashtags and the team names. However, during the event, we also tracked the names of the stadium, coaches and players, including substitutions. In all cases, including matches that did not involve English clubs, we only collected tweets in English to facilitate the annotation process.

We sought to maximise the dataset sizes, especially during the event, and so we favoured colloquial names when collecting datasets. Normally, for example, supporters refer to players and coaches by surnames or nicknames, not full names. Therefore unless the surname was common enough to be ambiguous or a common English lexeme, we did not track the full name. The tracking keywords used to collect each dataset follow in the next table.

Period	Keywords
	Southampton - Arsenal
Understanding	#SOUARS, Southampton, Arsenal
Event	#SOUARS, Southampton, Arsenal, St. Mary's Stadium, Hasenhuttl, Arteta, McCarthy, Yan Valery, Stephens, Bednarek, Bertrand, Redmond, Ward-Prowse, Emile Hoejbjerg, Stuart Armstrong, Ings, Obafemi, Vestergaard, Shane Long, Che Adams, Romeu, Walker-Peters, Smallbone, Gunn, Vokins, Tella, Emiliano Martinez, Bellerin, Mustafi, Rob Holding, Tierney, Ceballos, Xhaka, Saka, Pepe, Nketiah, Aubameyang, Sokratis, Lacazette, Ozil, Maitland-Niles, Reiss Nelson, Willock, Kolasinac, Macey, Matthew Smith
	Leicester - Manchester United
Understanding	#LEIMUN, Leicester, Manchester United

Period	Keywords
Event	#LEIMUN, Leicester, Manchester United, King Power Stadium, Rodgers, Solskjaer, Schmeichel, James Justin, Wes Morgan, Jonny Evans, Albrighton, Choudhury, Ndidi, Tielemans, Luke Thomas, Iheanacho, Vardy, Demarai Gray, Danny Ward, Barnes, Perez, Matthew James, Mendy, Praet, Bennett, Hirst, De Gea, Wan-Bissaka, Lindelof, Maguire, Brandon Williams, Pogba, Matic, Greenwood, Fernandes, Rashford, Martial, Bailly, Mata, Lingard, Fred, Daniel James, Romero, Fosu-Mensah, Ighalo, McTominay
	Turkey - Italy
Understanding	#TURITA, #ITATUR, #TUR, #ITA, #EURO2020
Event	#TURITA, #ITATUR, #TUR, #ITA, Stadio Olimpico, Cakir, Celik, Soyuncu, Demiral, Meras, Yokuslu, Karaman, Tufan, Yazici, Calhanoglu, Yilmaz, Insigne, Immobile, Berardi, Locatelli, Jorginho, Barella, Spinazzola, Chiellini, Bonucci, Florenzi, Donnarumma, Gunok, Bayindir, Cengiz Under, Tokoz, Antalyali, Kabak, Unal, Kokcu, Kahveci, Ayhan, Muldur, Dervisoglu, Gunes, Sirigu, Meret, Di Lorenzo, Belotti, Pessina, Emerson, Chiesa, Acerbi, Cristante, Bernardeschi, Raspadori, Bastoni, Mancini
	Wales - Switzerland
Understanding	#WALSUI, #SUIWAL, #WAL, #SUI, #EURO2020
Event	#WALSUI, #SUIWAL, #WAL, #SUI, #EURO2020, Baku Olimpiya, Danny Ward, Connor Roberts, Rodon, Ben Davies, Mepham, Morrell, Joe Allen, Ramsey, Daniel James, Moore, Bale, Seferovic, Embolo, Shaqiri, Rodriguez, Freuler, Xhaka, Mbabu, Akanji, Schaer, Elvedi, Sommer, Hennessey, Adam Davies, Gunter, Neco Williams, Lockyer, Harry Wilson, Tyler Roberts, Ampadu, Norrington-Davies, Jonathan Williams, Brooks, Levitt, Robert Page, Mvogo, Omlin, Widmer, Zakaria, Vargas, Zuber, Sow, Fassnacht, Benito, Mehmedi, Gavranovic, Comert, Petkovic
	Scotland - Czech Republic
Understanding	#SCOCZE, #CZESCO, #SCO, #CZE, #EURO2020
Event	#SCOCZE, #CZESCO, #SCO, #CZE, #EURO2020, Hampden Park, David Marshall, Hanley, Liam Cooper, Hendry, O'Donnell, Stuart Armstrong, McGinn, McTominay, Andrew Robertson, Dykes, Ryan Christie, Schick, Jankto, Darida, Masopust, Kral, Soucek, Boril, Kalas, Celustka, Coufal, Vaclik, Craig Gordon, McLaughlin, Callum McGregor, Che Adams, Greg Taylor, David Turnbull, Nisbet, Ryan Fraser, Patterson, Billy Gilmour, Forrest, McKenna, Steve Clarke, Mandous, Kaderabek, Brabec, Barak, Tomas Holes, Krmencik, Sevcik, Zima, Hlozek, Vydra, Mateju, Pekhart, Silhavy
	Hungary - France
Understanding	#HUNFRA, #FRAHUN, #HUN, #FRA, #EURO2020
Event	#HUNFRA, #FRAHUN, #HUN, #FRA, #EURO2020, Puskas Arena, Gulacsi, Botka, Willi Orban, Szalai, Nego, Kleinheisler, Nagy, Schafer, Fiola, Szalai, Sallai, Mbappe, Benzema, Griezmann, Rabiot, Kante, Pogba, Digne, Kimpembe, Varane, Pavard, Lloris, Dibusz, Bogdan, Lang, Kecskes, Cseri, Holender, Lovrencsics, Varga, Siger, Varga, Nikolic, Schon, Marco Rossi, Mandanda, Maignan, Lenglet, Lemar, Giroud, Ousmane Dembele, Tolisso, Moussa Sissoko, Lucas Hernandez, Dubois, Kounde, Thuram, Deschamps
	Trophée des Champions
Understanding	#TropheeDesChampions, #TDC2020, #PSGOM, #OMPSG, Trophee des Champions, PSG, Paris Saint-Germain, Marseille

Period	Keywords
Event	#TropheeDesChampions, #TDC2020, #PSGOM, #OMPSG, Trophee des Champions, PSG, Paris Saint-Germain, Marseille, Stade Bollaert-Delelis, Pochettino, Villas-Boas, Navas, Florenzi, Marquinhos, Diallo, Kurzawa, Verratti, Ander Herrera, Paredes, Di Maria, Icardi, Mbappe, Radonjic, Payet, Thauvin, Gueye, Kamara, Rongier, Nagatomo, Alvaro Gonzalez, Caleta-Car, Sakai, Mandanda, Sergio Rico, Kimpembe, Bakker, Pembele, Danilo Pereira, Draxler, Neymar, Kean, Sarabia, Yohann Pele, Balerdi, Sanson, Cuisance, Khaoui, Benedetto, Marley Ake, Germain, Lirola
	Copa del Rey
Understanding	#CopaDelRey, Copa del Rey, #AthleticClub, Athletic Club, Bilbao, Sociedad
Event	#CopaDelRey, Copa del Rey, #AthleticClub, Athletic Club, Bilbao, Sociedad, Estadio de La Cartuja, Unai Simon, de Marcos, Yeray Alvarez, Inigo Martinez, Berchiche, Dani Garcia, Vencedor, Berenguer, Raul Garcia, Muniain, Inaki Williams, Oyarzabal, Isak, Portu, David Silva, Zubimendi, Merino, Monreal, Le Normand, Zubeldia, Gorosabel, Remiro, Ezkieta, Unai Nunez, Vesga, Ibai Gomez, Unai Lopez, Lekue, Villalibre, Capa, Balenziaga, Marcelino Garcia, Ayesa, Elustondo, Sagnan, Guevara, Fernandez, Januzaj, Barrenetxea, Bautista, Munoz, Alguacil
	Parma - Milan
Understanding	#ParmaMilan, Parma, Milan
Event	#ParmaMilan, Parma, Milan, Stadio Ennio Tardini, Sepe, Conti, Bani, Gagliolo, Pezzella, Kucka, Hernani, Kurtic, Dennis Man, Pelle, Gervinho, Colombi, Grassi, Cornelius, Brugman, Laurini, Bruno Alves, Osorio, Dierckx, Valenti, Chaka Traore, Camara, Busi, D'Aversa, Donnarumma, Kalulu, Kjaer, Tomori, Theo Hernandez, Bennacer, Kessie, Saelemaekers, Calhanoglu, Rebic, Ibrahimovic, Tatarusanu, Donnarumma, Dalot, Castillejo, Tonali, Mandzukic, Hauge, Leao, Meite, Brahim Diaz, Krunic, Gabbia, Pioli

Table D.11: The keywords used to collect the football match datasets used in Section 5.3.

In reality, the algorithms discarded many tweets from the datasets. In the first analysis of Section 5.3, we experimented with $\text{ELD}_{\text{Filtered}}$, which filtered tweets that did not contain any domain term. In the second analysis, SEER similarly filtered tweets as it deployed them to the relevant streams. Then, both $\text{ELD}_{\text{Filtered}}$ and SEER, like ELD itself, filtered the tweets that they received. Details about how many tweets each algorithm processed at each stage follow in the next table.

Algorithm	Dataset size	Domain filtering (%)	Algorithm filtering (%)
	Summary		
ELD	124,479	124,479.00 (100%)	100,856.33 (80.37%)
$\text{ELD}_{\text{Filtered}}$	124,479	59,131.67 (46.97%)	46,909.50 (36.93%)
SEER	124,479	82,632.67 (65.19%)	35,073.67 (26.84%)
	Southampton - Arsenal		
ELD	97,874	97,874 (100.00%)	84,400 (86.23%)

Algorithm	Dataset size	Domain filtering (%)	Algorithm filtering (%)
$ELD_{Filtered}$	97,874	41,207 (42.10%)	34,869 (35.63%)
SEER	97,874	58,227 (59.49%)	23,897 (24.42%)
Leicester - Manchester United			
ELD	209,132	209,132 (100.00%)	178,743 (85.47%)
$ELD_{Filtered}$	209,132	106,505 (50.93%)	88,906 (42.51%)
SEER	209,132	154,918 (74.08%)	74,048 (35.41%)
Turkey - Italy			
ELD	109,888	109,888 (100.00%)	81,897 (74.53%)
$ELD_{Filtered}$	109,888	50,545 (46.00%)	37,013 (33.68%)
SEER	109,888	70,663 (64.30%)	26,453 (24.07%)
Wales - Switzerland			
ELD	87,717	87,717 (100.00%)	67,641 (77.11%)
$ELD_{Filtered}$	87,717	42,901 (48.91%)	32,107 (36.60%)
SEER	87,717	57,942 (66.06%)	19,420 (22.14%)
Scotland - Czech Republic			
ELD	120,194	120,194 (100.00%)	94,811 (78.88%)
$ELD_{Filtered}$	120,194	61,480 (51.15%)	48,247 (40.14%)
SEER	120,194	81,120 (67.49%)	35,419 (29.47%)
Hungary - France			
ELD	122,069	122,069 (100.00%)	97,646 (79.99%)
$ELD_{Filtered}$	122,069	52,152 (42.72%)	40,315 (33.03%)
SEER	122,069	72,926 (59.74%)	31,205 (25.56%)

Table D.12: The breakdown of the dataset filtering of Table 5.1. The table reports the number of tweets that pass each filter across all datasets. Domain filtering refers to standard tweet filtering for $ELD_{Filtered}$ and streaming for SEER.

Here, we re-iterate the nuances of $ELD_{Filtered}$'s and SEER's figures. $ELD_{Filtered}$ received, on average, 46.97% of any dataset, which means that it processed 46.97% of all unique tweets in the dataset. SEER received the same number of unique tweets, but it processed some of them multiple times across several streams. Therefore the 65.19% of tweets that SEER received actually means that SEER processed the equivalent of 65.19% of all unique tweets of any dataset.

D.4 | Data used in Chapter 6

Throughout Chapter 6, we focused on British politics. In particular, we followed Liz Truss' first days as Prime Minister of the United Kingdom, starting on the eve of her election as prime minister on Sunday, 4 September 2022, and ending six days later at midnight on Sunday, September 11. We tracked Truss for the whole week, but a few other politicians also ebbed in and out of relevance: Rishi Sunak, Boris Johnson and Keir Starmer.

The data evokes that of Section 4.5, but we treated our corpora somewhat differently. In the ATE task of Section 4.5, we wanted to capture what happens in politics but not outside. Conversely, in the TDT task of Chapter 6, we wanted to capture the topics that emerged from the political baseline. Therefore while in Section 4.5 we used the general corpus to construct our TF-ICF scheme, to compare politics with a neutral baseline of discourse, in Chapter 6 we used the data from September 4 as ELD's and SEER's understanding periods.

The event period, between Monday, 5 September 2022 and Saturday, September 10 captured several historic events. On September 5, Truss won a mandate to lead the Conservative Party, defeating fellow Tory Rishi Sunak. On September 6, Truss formally assumed the role of prime minister, and the next day, on September 7, she met her cabinet for the first time. On September 8, Truss announced her much-awaited energy bill, but news of Queen Elizabeth II's death in the afternoon overshadowed the announcement. On September 9 and 10, Truss attended official ceremonies related to the new King. Details about how we collected our datasets follow in the next table.

Date	Time	Tweets	Keywords
Sep 4, 2022	13:00–14:00	16,427	Truss, Sunak
Sep 4, 2022	14:00–15:00	15,824	Truss, Sunak
Sep 4, 2022	15:00–16:00	14,066	Truss, Sunak
Sep 4, 2022	16:00–17:00	12,864	Truss, Sunak
Sep 4, 2022	17:00–18:00	13,018	Truss, Sunak
Sep 4, 2022	18:00–19:00	11,994	Truss, Sunak
Sep 4, 2022	19:00–20:00	11,207	Truss, Sunak
Sep 4, 2022	20:00–21:00	10,496	Truss, Sunak
Sep 4, 2022	21:00–22:00	10,638	Truss, Sunak
Sep 4, 2022	22:00–23:00	4,360	Truss, Sunak
Sep 4, 2022	23:00–0:00	11,234	Truss, Sunak

Date	Time	Tweets	Keywords
Sep 5, 2022	0:00–1:00	7,118	Truss, Sunak
Sep 5, 2022	1:00–2:00	3,864	Truss, Sunak
Sep 5, 2022	2:00–3:00	2,317	Truss, Sunak
Sep 5, 2022	3:00–4:00	1,509	Truss, Sunak
Sep 5, 2022	4:00–5:00	1,265	Truss, Sunak
Sep 5, 2022	5:00–6:00	1,391	Truss, Sunak
Sep 5, 2022	6:00–7:00	1,968	Truss, Sunak
Sep 5, 2022	7:00–8:00	4,992	Truss, Sunak
Sep 5, 2022	8:00–9:00	10,055	Truss, Sunak
Sep 5, 2022	9:00–10:00	14,615	Truss, Sunak
Sep 5, 2022	10:00–11:00	13,790	Truss, Sunak
Sep 5, 2022	11:00–12:00	13,782	Truss, Sunak
Sep 5, 2022	12:00–13:00	14,863	Truss, Sunak
Sep 5, 2022	13:00–14:00	90,170	Truss, Sunak
Sep 5, 2022	14:00–15:00	109,836	Truss, Sunak
Sep 5, 2022	15:00–16:00	61,183	Truss, Sunak
Sep 5, 2022	16:00–17:00	50,367	Truss, Sunak
Sep 5, 2022	17:00–18:00	48,545	Truss, Sunak
Sep 5, 2022	18:00–19:00	48,897	Truss, Sunak
Sep 5, 2022	19:00–20:00	46,584	Truss, Sunak
Sep 5, 2022	20:00–21:00	39,905	Truss, Sunak
Sep 5, 2022	21:00–22:00	36,996	Truss, Sunak
Sep 5, 2022	22:00–23:00	34,591	Truss, Sunak
Sep 5, 2022	23:00–0:00	32,750	Truss, Sunak
Sep 6, 2022	0:00–1:00	22,534	Truss, Sunak
Sep 6, 2022	1:00–2:00	13,102	Truss, Sunak
Sep 6, 2022	2:00–3:00	7,782	Truss, Sunak
Sep 6, 2022	3:00–4:00	5,720	Truss, Sunak
Sep 6, 2022	4:00–5:00	5,120	Truss, Sunak
Sep 6, 2022	5:00–6:00	5,633	Truss, Sunak
Sep 6, 2022	6:00–7:00	7,264	Truss, Sunak
Sep 6, 2022	7:00–8:00	14,454	Truss, Sunak
Sep 6, 2022	8:00–9:00	24,875	Truss, Sunak
Sep 6, 2022	9:00–10:00	29,739	Truss, Sunak

Date	Time	Tweets	Keywords
Sep 6, 2022	10:00–11:00	30,668	Truss, Sunak
Sep 6, 2022	11:00–12:00	31,413	Truss, Sunak
Sep 6, 2022	12:00–13:00	41,991	Truss, Sunak, Boris Johnson
Sep 6, 2022	13:00–14:00	47,029	Truss, Sunak, Boris Johnson
Sep 6, 2022	14:00–15:00	47,805	Truss, Sunak, Boris Johnson
Sep 6, 2022	15:00–16:00	37,985	Truss, Sunak, Boris Johnson
Sep 6, 2022	16:00–17:00	33,642	Truss, Sunak, Boris Johnson
Sep 6, 2022	17:00–18:00	37,592	Truss, Sunak, Boris Johnson
Sep 6, 2022	18:00–19:00	49,912	Truss, Sunak, Boris Johnson
Sep 6, 2022	19:00–20:00	41,399	Truss, Sunak, Boris Johnson
Sep 6, 2022	20:00–21:00	42,502	Truss, Sunak, Boris Johnson
Sep 6, 2022	21:00–22:00	36,577	Truss, Sunak, Boris Johnson
Sep 6, 2022	22:00–23:00	35,835	Truss, Sunak, Boris Johnson
Sep 6, 2022	23:00–0:00	31,896	Truss, Sunak, Boris Johnson
Sep 7, 2022	0:00–1:00	20,672	Truss, Sunak, Boris Johnson
Sep 7, 2022	1:00–2:00	11,977	Truss, Sunak, Boris Johnson
Sep 7, 2022	2:00–3:00	6,802	Truss, Sunak, Boris Johnson
Sep 7, 2022	3:00–4:00	5,329	Truss, Sunak, Boris Johnson
Sep 7, 2022	4:00–5:00	4,824	Truss, Sunak, Boris Johnson
Sep 7, 2022	5:00–6:00	5,039	Truss, Sunak, Boris Johnson
Sep 7, 2022	6:00–7:00	6,343	Truss, Sunak, Boris Johnson
Sep 7, 2022	7:00–8:00	12,302	Truss, Sunak, Boris Johnson
Sep 7, 2022	8:00–9:00	21,437	Truss, Sunak, Boris Johnson
Sep 7, 2022	9:00–10:00	28,464	Truss, Sunak, Boris Johnson
Sep 7, 2022	10:00–11:00	26,334	Truss, Sunak, Boris Johnson
Sep 7, 2022	11:00–12:00	23,148	Truss, Sunak, Boris Johnson
Sep 7, 2022	12:00–13:00	22,434	Truss, Sunak, Boris Johnson
Sep 7, 2022	13:00–14:00	46,485	Truss, Sunak, Boris Johnson, Starmer
Sep 7, 2022	14:00–15:00	37,735	Truss, Sunak, Boris Johnson, Starmer
Sep 7, 2022	15:00–16:00	29,700	Truss, Sunak, Boris Johnson, Starmer
Sep 7, 2022	16:00–17:00	26,611	Truss, Sunak, Boris Johnson, Starmer
Sep 7, 2022	17:00–18:00	28,637	Truss, Sunak, Boris Johnson, Starmer
Sep 7, 2022	18:00–19:00	27,762	Truss, Sunak, Boris Johnson, Starmer
Sep 7, 2022	19:00–20:00	29,280	Truss, Sunak, Boris Johnson, Starmer
Sep 7, 2022	20:00–21:00	26,434	Truss, Sunak, Boris Johnson, Starmer

Date	Time	Tweets	Keywords
Sep 7, 2022	21:00–22:00	23,124	Truss, Sunak, Boris Johnson, Starmer
Sep 7, 2022	22:00–23:00	22,035	Truss, Sunak, Boris Johnson, Starmer
Sep 7, 2022	23:00–0:00	22,451	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	0:00–1:00	15,602	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	1:00–2:00	8,427	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	2:00–3:00	4,834	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	3:00–4:00	3,388	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	4:00–5:00	2,783	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	5:00–6:00	2,836	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	6:00–7:00	4,226	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	7:00–8:00	9,425	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	8:00–9:00	19,131	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	9:00–10:00	23,514	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	10:00–11:00	22,288	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	11:00–12:00	21,108	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	12:00–13:00	27,392	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	13:00–14:00	49,649	Truss, Sunak, Boris Johnson, Starmer
Sep 8, 2022	14:00–15:00	41,574	Truss, Starmer
Sep 8, 2022	15:00–16:00	33,229	Truss, Starmer
Sep 8, 2022	16:00–17:00	30,281	Truss, Starmer
Sep 8, 2022	17:00–18:00	28,345	Truss, Starmer
Sep 8, 2022	18:00–19:00	26,717	Truss, Starmer
Sep 8, 2022	19:00–20:00	29,849	Truss, Starmer
Sep 8, 2022	20:00–21:00	51,604	Truss, Starmer
Sep 8, 2022	21:00–22:00	26,561	Truss, Starmer
Sep 8, 2022	22:00–23:00	22,975	Truss, Starmer
Sep 8, 2022	23:00–0:00	19,220	Truss, Starmer
Sep 9, 2022	0:00–1:00	12,913	Truss, Starmer
Sep 9, 2022	1:00–2:00	8,565	Truss, Starmer
Sep 9, 2022	2:00–3:00	5,274	Truss, Starmer
Sep 9, 2022	3:00–4:00	3,818	Truss, Starmer
Sep 9, 2022	4:00–5:00	3,264	Truss, Starmer
Sep 9, 2022	5:00–6:00	2,867	Truss, Starmer
Sep 9, 2022	6:00–7:00	3,236	Truss, Starmer

Date	Time	Tweets	Keywords
Sep 9, 2022	7:00–8:00	5,603	Truss, Starmer
Sep 9, 2022	8:00–9:00	10,154	Truss, Starmer
Sep 9, 2022	9:00–10:00	11,959	Truss, Starmer
Sep 9, 2022	10:00–11:00	12,256	Truss, Starmer
Sep 9, 2022	11:00–12:00	11,610	Truss, Starmer
Sep 9, 2022	12:00–13:00	10,985	Truss, Starmer
Sep 9, 2022	13:00–14:00	14,787	Truss, Starmer
Sep 9, 2022	14:00–15:00	13,267	Truss, Starmer
Sep 9, 2022	15:00–16:00	9,918	Truss, Starmer
Sep 9, 2022	16:00–17:00	9,017	Truss, Starmer
Sep 9, 2022	17:00–18:00	10,171	Truss, Starmer
Sep 9, 2022	18:00–19:00	9,255	Truss, Starmer
Sep 9, 2022	19:00–20:00	8,768	Truss, Starmer
Sep 9, 2022	20:00–21:00	8,959	Truss, Starmer
Sep 9, 2022	21:00–22:00	10,483	Truss, Starmer
Sep 9, 2022	22:00–23:00	8,431	Truss, Starmer
Sep 9, 2022	23:00–0:00	7,802	Truss, Starmer
Sep 10, 2022	0:00–1:00	5,885	Truss, Starmer
Sep 10, 2022	1:00–2:00	3,778	Truss, Starmer
Sep 10, 2022	2:00–3:00	2,638	Truss, Starmer
Sep 10, 2022	3:00–4:00	1,725	Truss, Starmer
Sep 10, 2022	4:00–5:00	1,478	Truss, Starmer
Sep 10, 2022	5:00–6:00	1,381	Truss, Starmer
Sep 10, 2022	6:00–7:00	1,487	Truss, Starmer
Sep 10, 2022	7:00–8:00	2,355	Truss, Starmer
Sep 10, 2022	8:00–9:00	4,521	Truss, Starmer
Sep 10, 2022	9:00–10:00	7,020	Truss, Starmer
Sep 10, 2022	10:00–11:00	7,106	Truss, Starmer
Sep 10, 2022	11:00–12:00	8,677	Truss, Starmer
Sep 10, 2022	12:00–13:00	8,084	Truss, Starmer
Sep 10, 2022	13:00–14:00	7,980	Truss, Starmer
Sep 10, 2022	14:00–15:00	6,825	Truss, Starmer
Sep 10, 2022	15:00–16:00	6,286	Truss, Starmer
Sep 10, 2022	16:00–17:00	5,820	Truss, Starmer
Sep 10, 2022	17:00–18:00	5,619	Truss, Starmer

Date	Time	Tweets	Keywords
Sep 10, 2022	18:00–19:00	5,642	Truss, Starmer
Sep 10, 2022	19:00–20:00	6,077	Truss, Starmer
Sep 10, 2022	20:00–21:00	5,606	Truss, Starmer
Sep 10, 2022	21:00–22:00	6,110	Truss, Starmer
Sep 10, 2022	22:00–23:00	6,217	Truss, Starmer
Sep 10, 2022	23:00–00:00	5,882	Truss, Starmer
		2,883,828	

Table D.13: The UK politics datasets used in Chapter 6.

D.5 | Data used in Appendix A

Although we present Appendix A as a review of TDT research's evaluation methodologies, we collected various datasets to verify some of literature's assumptions about Twitter data. In this section, we present both the repurposed datasets from other chapters and the data collected specifically for Appendix A.3.

Differently from other chapters, in Appendix A we only used the data from the event period. Moreover, we downloaded the data twice. The first time, we downloaded the datasets while the football match was ongoing; we originally collected the first dataset, from the match between Crystal Palace and Chelsea, for our previous work [146]. The second time, we exported the tweet IDs and used them to re-download the datasets anew after a few days, weeks, months or years. The differences between the two versions allowed us to analyse what kind of tweets we lost with Twitter's data-sharing policy [254]. Details about each match follow in the next table.

	Download date		Tweets	
Event	Original	Downloaded	Original	Downloaded (% available)
Crystal Palace - Chelsea	30 Dec 2018	29 Aug 2021	63,891	41,028 (64.22%)
Southampton - Arsenal	25 Jun 2020	29 Aug 2021	97,874	70,656 (72.19%)
Turkey - Italy	11 Jun 2021	30 Aug 2021	109,888	90,543 (82.40%)
Liverpool - Atlético de Madrid	3 Nov 2021	4 Nov 2021	107,607	94,040 (87.39%)
			379,260	296,267 (78.12%)

Table D.14: The football match datasets used in Appendix A.3.

Like in other chapters, we generally downloaded the original datasets by tracking the event hashtag, and the names of the teams, players, coaches and the stadium. We prioritised colloquial references to names, thus maximising the amount of data we collected. The tracking keywords used to collect each dataset follow in the next table.

Event	Keywords
Crystal Palace - Chelsea	#CRYCHE, Crystal Palace, Chelsea, Selhurst Park, Sarri, Hodgson, Guaita, Wan-Bissaka, Tomkins, Sakho, van Aanholt, McArthur, Kouyate, Milivojević, Meyer, Zaha, Townsend, Hennessey, Joel Ward, Dann, Puncheon, Schlupp, Wickham, Ayew, Arrizabalaga, Azpilicueta, Rüdiger, Luiz, Alonso Mendoza, Kante, Jorginho, Kovacic, Willian, Giroud, Eden Hazard, Caballero, Palmieri, Christensen, Ampadu, Zappacosta, Barkley, Morata
Southampton - Arsenal	#SOUARS, Southampton, Arsenal, St. Mary's Stadium, Hasenhuttl, Arteta, McCarthy, Yan Valery, Stephens, Bednarek, Bertrand, Redmond, Ward-Prowse, Emile Hoejbjerg, Stuart Armstrong, Ings, Obafemi, Vestergaard, Shane Long, Che Adams, Romeu, Walker-Peters, Smallbone, Gunn, Vokins, Tella, Emiliano Martinez, Bellerin, Mustafi, Rob Holding, Tierney, Ceballos, Xhaka, Saka, Pepe, Nketiah, Aubameyang, Sokratis, Lacazette, Ozil, Maitland-Niles, Reiss Nelson, Willock, Kolasinac, Macey, Matthew Smith
Turkey - Italy	#TURITA, #ITATUR, #TUR, #ITA, Stadio Olimpico, Cakir, Celik, Soyuncu, Demiral, Meras, Yokuslu, Karaman, Tufan, Yazici, Calhanoglu, Yilmaz, Insigne, Immobile, Berardi, Locatelli, Jorginho, Barella, Spinazzola, Chiellini, Bonucci, Florenzi, Donnarumma, Gunok, Bayindir, Cengiz Under, Tokoz, Antalyali, Kabak, Unal, Kokcu, Kahveci, Ayhan, Muldur, Dervisoglu, Gunes, Sirigu, Meret, Di Lorenzo, Belotti, Pessina, Emerson, Chiesa, Acerbi, Cristante, Bernardeschi, Raspadori, Bastoni, Mancini
Liverpool - Atlético de Madrid	#LIVATM, Liverpool, Atleti, Atletico, Anfield, Alisson, Alexander-Arnold, Matip, van Dijk, Tsimikas, Oxlade-Chamberlain, Fabinho, Henderson, Salah, Jota, Mane, Suarez, Felix, Correa, Carrasco, Koke, De Paul, Trippier, Hermoso, Gimenez, Felipe, Oblak, Adrian, Kelleher, Konate, Alcantara, Firmino, Minamino, Robertson, Origi, Phillips, Neco Williams, Tyler Morton, Klopp, Lecomte, Lodi, Hector Herrera, Cunha, Vrsaljko, Serrano, Iturbe, Carlos Martin, Fran Gonzalez, Simeone

Table D.15: The keywords used to collect the football match datasets used in Appendix A.3.

E

Configurations

This appendix includes details about the algorithms' configurations as used in this work. In the absence of an automatic evaluation methodology, almost all TDT algorithms must be tweaked manually to optimise performance. Therefore this appendix describes which parameters we tweaked and why, and lists the final configurations for each experiment.

E.1 | Configurations for the analyses in Chapter 5

In Chapter 5, we improved ELD [146] with the application of understanding. ELD's novelty did not lie in its understanding—it had none—but in the way it harnessed the combination of document-pivot and feature-pivot techniques, which improved the granularity of our algorithm at a small cost to precision. Therefore we used ELD itself as a baseline without understanding.

Nevertheless, ELD's highly-parametric structure posed challenges. The combined model inherited the parameters of both TDT families: the document-pivot algorithm's minimum cluster size and freeze period, the similarity measure and threshold, and the feature-pivot algorithm's window length and minimum burst. Naturally, the manual evaluations of TDT literature, whose challenges we discussed at length in Appendix A, force us to set some of the parameters empirically [146].

In our analyses in Chapter 5, we re-used some of the parameters from our previous work [146]. We had developed ELD within the context of football matches, and its parameters then applied again in our case study on football matches. In particular, we used cosine similarity to compare incoming tweets with existing clusters, using a similarity threshold of 0.5, and we set the time window length to half a minute. We

considered a cluster to be topical if it had one word with a burst of 0.8 or higher, or two or more words with a burst of 0.5 or higher.

We could not fix two other parameters, however. First, the minimum cluster size acts as a threshold, as in other popular TDT literature: the more popular an event, the higher the threshold to accept a cluster as topical. As a rule of thumb, a newsworthy cluster must have, at least, three tweets. Second, the freeze period acts as a setting to tweak performance, allowing the resource-heavy clustering algorithm to match the throughput of popular events. Furthermore, since we do not focus on timeliness, we slowed down ELD's input to allow the clustering algorithm to keep up with the stream.

Our first experiment with ELD is the trivial application of understanding, $\text{ELD}_{\text{Filtered}}$. With our trivial application, we sought to identify the limits of understanding, and therefore we re-used ELD's configurations in $\text{ELD}_{\text{Filtered}}$. Larger datasets might have permitted us to relax $\text{ELD}_{\text{Filtered}}$'s parameters by reducing the minimum cluster size, for example, without degrading recall, but we leave such experiments for future work.

Later, in the sensitivity experiments of Appendix B, as we evaluated ELD's performance on smaller datasets, we did relax the parameters. In reality, however, we could only tweak, slightly, the cluster size and freeze period, bringing the two settings to their bare minimum. All of ELD's and $\text{ELD}_{\text{Filtered}}$'s configurations follow in the next table.

Event	Cluster size (tweets)	Freeze period (seconds)	Throttle (multiplier)
	All tweets		
Southampton - Arsenal	3	20	0.5
Leicester - Manchester United	10	5	0.25
Turkey - Italy	3	20	0.5
Wales - Switzerland	3	20	0.5
Scotland - Czech Republic	3	20	0.5
Hungary - France	5	20	0.5
	50,000 tweets		
Southampton - Arsenal	3	20	0.5
Leicester - Manchester United	3	20	0.5
Turkey - Italy	3	20	0.5
Wales - Switzerland	3	20	0.5
Scotland - Czech Republic	3	20	0.5
Hungary - France	3	20	0.5
	25,000 tweets		
Southampton - Arsenal	3	20	0.5

Event	Cluster size (tweets)	Freeze period (seconds)	Throttle (multiplier)
Leicester - Manchester United	3	20	0.5
Turkey - Italy	3	20	0.5
Wales - Switzerland	3	20	0.5
Scotland - Czech Republic	3	20	0.5
Hungary - France	3	20	0.5
	10,000 tweets		
Southampton - Arsenal	3	20	0.5
Leicester - Manchester United	3	20	0.5
Turkey - Italy	3	20	0.5
Wales - Switzerland	3	20	0.5
Scotland - Czech Republic	3	20	0.5
Hungary - France	3	20	0.5

Table E.1: ELD's configurations in the evaluations of Section 5.3.

SEER's structure simultaneously simplified the algorithm but complicated parameter tweaking. Our novel algorithm only inherited the parameters from the feature-pivot technique: the length of the sliding time windows, the static threshold and the minimum burst. Nevertheless, the algorithm's sensitivity made optimising the algorithm a more trying task: the three parameters can take a much broader range of values than ELD's. We could only fix one of the three parameters, the time window length, which depends on the domain; we fixed it to one minute in football matches.

Therefore we only tweaked two parameters: the static threshold and the minimum burst. The static threshold replaces the role of ELD's minimum cluster size and changes proportionally with the event's popularity. Conversely, the minimum burst changed little, ranging from 0.5 to 0.8. SEER's configurations, in each event and at different dataset sizes, follow in the next table.

Event	Static threshold (activity)	Minimum burst
	All tweets	
Southampton - Arsenal	15	0.7
Leicester - Manchester United	30	0.6
Turkey - Italy	15	0.6
Wales - Switzerland	10	0.7
Scotland - Czech Republic	15	0.7

Event	Static threshold (activity)	Minimum burst
Hungary - France	15	0.7
Trophée des Champions	3	0.8
Copa del Rey	2	0.8
Parma - Milan	3	0.8
50,000 tweets		
Southampton - Arsenal	12	0.6
Leicester - Manchester United	14	0.6
Turkey - Italy	14	0.5
Wales - Switzerland	9	0.8
Scotland - Czech Republic	9	0.6
Hungary - France	14	0.5
25,000 tweets		
Southampton - Arsenal	8	0.6
Leicester - Manchester United	10	0.6
Turkey - Italy	10	0.6
Wales - Switzerland	8	0.8
Scotland - Czech Republic	5	0.8
Hungary - France	7	0.8
10,000 tweets		
Southampton - Arsenal	4	0.8
Leicester - Manchester United	4	0.8
Turkey - Italy	4	0.8
Wales - Switzerland	4	0.8
Scotland - Czech Republic	3	0.8
Hungary - France	3	0.8

Table E.2: SEER's configurations in the evaluations of Section 5.3.

E.2 | Configurations for the analyses in Chapter 6

In Chapter 6, we compared ELD with SEER to study the sacrifices of understanding to portability. We did not expect to contrast ELD's performance in precision and recall with SEER's, nor did we want to. The thorough analyses of Section 4.5 revealed how SEER greatly out-performs ELD when we configured both optimally. We simply expected the

comparison in Chapter 6 to expose what SEER misses with understanding.

With portability our priority, we loosened the configurations of both algorithms. The nature of the election itself weighed heavy; the impulses, variety and duration of the event period precluded any lengthy experiments. In both ELD and SEER, we set the burst threshold to 0.7 and merged any topics occurring within 15 minutes of each other. In SEER, we merged nodes for each stream separately.

One change deserves explanation: ELD has a shorter time window than SEER. While SEER uses 15-minute time windows, ELD creates checkpoints every five minutes. ELD owes the difference to two reasons. First, ELD buffers tweets before creating checkpoints, but the data in 15-minute blocks would have exhausted our limited memory, overflown into the swap space and rendered the algorithm unusable. Second, ELD covers the entire event in one monolithic timeline whose discourse changes more quickly than SEER's topical streams. Nevertheless, ELD's five-minute time windows still capture a general idea of the event's changing vocabulary.

We barely needed to configure SEER further. We found one configuration to suffice as long as we gave the algorithm the liberty to oversee itself. We set the static threshold to a humble 50 tweets per 15 minute-time window and the dynamic threshold to one standard deviation above the mean tweeting activity.

Conversely, we could not but vary ELD's configuration. The volume varied from barely a thousand tweets to a hundred-fold more around the time when Truss won the leadership election. Therefore in the two hours surrounding Truss' election, we tweaked the minimum cluster size, the freeze period and throttled the stream to reflect the discourse. ELD's configurations follow in the next table.

Date	Time	Cluster size (tweets)	Freeze period (seconds)	Throttle (multiplier)
Sep 5, 2022	00:00–13:00	5	30	0.5
Sep 5, 2022	13:00–15:00	10	5	0.1
Sep 5, 2022	15:00–00:00	5	30	0.5
Sep 6, 2022	00:00–00:00	5	30	0.5
Sep 7, 2022	00:00–00:00	5	30	0.5
Sep 8, 2022	00:00–00:00	5	30	0.5
Sep 9, 2022	00:00–00:00	5	30	0.5
Sep 10, 2022	00:00–00:00	5	30	0.5

Table E.3: ELD's configurations in the evaluation of Section 6.3.

F

Results

This appendix includes comprehensive detail about the results presented in this work. For the sake of clarity, the tables in the main text often include aggregate results as a summary, especially when the full tables would occupy too much space. The rest of this appendix presents a full breakdown of the results, including how they should be interpreted in the main text. We have made all outputs, results and annotations available in the `NicholasMamo/phd-data` repository.

F.1 | Results from the analyses of Chapter 3

In Chapter 3, we annotated rankings of named entities and participants using standard IR metrics, namely precision, recall and AP. To those, we added a balance metric, which measures the bias in a ranking; the higher the balance, the less the bias in a ranking. The summary tables listed the macro-average performance, thus valuing each event equally. In reality, however, the difference between the micro-average and the macro-average matters little in our experiments: normally every ranking has the same number of elements and ground truth items.

Results from Section 3.1

Unlike the other chapters, Chapter 3 included a short analysis in the first section, Section 3.1. The first analysis helped us understand better the suitability of named entities as participants, a common assumption in TDT circles. Therefore we did not extract named entities before the event started but as it happened. Given that we collected the six datasets before Twitter rolled out the second version of its API, we could not eval-

uate the performance of Twitter's own annotations. The breakdown of results from the NER experiment follows in the next table.

Model	Precision	Recall	AP	Balance
Summary				
NLTK	48.33%	31.46%	30.42%	0.3533
TwitterNER	48.00%	28.85%	34.76%	0.2829
Southampton - Arsenal				
NLTK	50.00%	35.56%	42.33%	0.1429
TwitterNER	54.00%	35.56%	37.66%	0.4545
Leicester - Manchester United				
NLTK	52.00%	35.56%	41.39%	0.4545
TwitterNER	52.00%	35.56%	42.04%	0.6000
Turkey - Italy				
NLTK	46.00%	31.37%	22.27%	0.2308
TwitterNER	40.00%	21.57%	24.71%	0.1000
Wales - Switzerland				
NLTK	42.00%	29.41%	21.26%	0.8750
TwitterNER	44.00%	25.49%	33.26%	0.3000
Scotland - Czech Republic				
NLTK	54.00%	29.41%	29.58%	0.2500
TwitterNER	68.00%	37.25%	56.44%	0.1176
Hungary - France				
NLTK	46.00%	27.45%	25.68%	0.1667
TwitterNER	30.00%	17.65%	14.41%	0.1250

Table F.1: NLTK's and TwitterNER's participant detection results in football matches, summarised in Table 3.1. Neither NER model registered statistically-significant gains over the other.

Results from Section 3.3

In the first analysis of Section 3.3, we compared NLTK [22], TwitterNER [166] and Twitter's own annotations. The next table displays the differences between the three models. The summary, like in all other tables in this section, lists the MAP, or the macro-average

AP over all events. For each event, then, the table lists the individual AP values. The full breakdown of the comparison between NER models follows in the next table.

Model	Precision	Recall	Precision (lenient)	MAP (AP)	Balance
Summary					
NLTK	29.80%	21.55%	31.20%	17.36%	0.4885
TwitterNER	△ 32.80%	▲ 26.67%	△ 34.40%	20.09%	△ 0.7273
Twitter	▲ 44.40%	▲ 33.98%	▲ 46.40%	▲ 30.60%	0.5786
Juventus - Inter					
NLTK	16.00%	10.42%	16.00%	9.36%	0.6667
TwitterNER	18.00%	12.50%	18.00%	10.12%	1.0000
Twitter	30.00%	25.00%	32.00%	14.64%	0.7143
Crystal Palace - Arsenal					
NLTK	30.00%	26.67%	36.00%	18.30%	0.3333
TwitterNER	30.00%	31.11%	34.00%	19.14%	0.5556
Twitter	42.00%	33.33%	50.00%	32.12%	0.5000
Manchester City - Atlético					
NLTK	26.00%	10.64%	26.00%	12.02%	0.2500
TwitterNER	26.00%	21.28%	26.00%	10.56%	1.0000
Twitter	54.00%	31.91%	54.00%	36.64%	0.5000
Burnley - Everton					
NLTK	40.00%	40.00%	40.00%	20.78%	1.0000
TwitterNER	52.00%	53.33%	52.00%	35.44%	0.8462
Twitter	70.00%	64.44%	70.00%	52.26%	0.9333
Watford - Leeds					
NLTK	12.00%	8.89%	14.00%	6.69%	0.3333
TwitterNER	20.00%	15.56%	22.00%	9.05%	0.4000
Twitter	16.00%	13.33%	20.00%	7.07%	0.5000
Aston Villa - Tottenham					
NLTK	20.00%	13.33%	20.00%	11.26%	0.5000
TwitterNER	28.00%	20.00%	28.00%	13.62%	0.8000
Twitter	30.00%	24.44%	30.00%	14.01%	0.8333
Manchester City - Liverpool					
NLTK	38.00%	22.22%	38.00%	23.78%	0.6667

Model	Precision	Recall	Precision (lenient)	MAP (AP)	Balance
TwitterNER	36.00%	24.44%	36.00%	25.65%	0.5714
Twitter	58.00%	35.56%	58.00%	49.85%	0.7778
Real Madrid - Chelsea					
NLTK	42.00%	30.00%	42.00%	21.77%	0.2500
TwitterNER	40.00%	24.00%	40.00%	25.44%	0.2000
Twitter	50.00%	34.00%	50.00%	29.05%	0.2143
Newcastle - Leicester					
NLTK	24.00%	13.33%	24.00%	13.05%	0.5000
TwitterNER	30.00%	22.22%	32.00%	21.11%	1.0000
Twitter	30.00%	22.22%	30.00%	17.86%	0.2500
Liverpool - Manchester United					
NLTK	50.00%	40.00%	56.00%	36.61%	0.3846
TwitterNER	48.00%	42.22%	56.00%	30.72%	0.9000
Twitter	64.00%	55.56%	70.00%	52.53%	0.5625

Table F.2: The NER tools' participant detection results in football matches, summarised in Table 3.3. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the model in the row above.

Following the conclusion that NER, linguistic understanding, cannot substitute for semantic understanding, we applied the APD process. In the next experiments, we used each NER model to extract named entities, serving as the APD framework's first step. In the next table, the subscript refers to the NER model that each APD algorithm uses.

The table also includes indications of statistical significance. The significance should be interpreted as one model's improvement over the one immediately above it: DEPICT over ELD, and ELD over the NER model. We present statistical significance in this way because in almost every case, when ELD out-performs the NER model, DEPICT does too. The full breakdown of results follows in the next table.

Model	Precision	Recall	Precision (lenient)	MAP (AP)	Balance
Summary					
NLTK	29.80%	21.55%	31.20%	17.36%	0.4885
ELD_{NLTK}	▲ 48.68%	▲ 50.55%	▲ 66.52%	▲ 38.56%	▽ 0.2856

Model	Precision	Recall	Precision (lenient)	MAP (AP)	Balance
$DEPICT_{NLTK}$	△ 54.00%	53.77%	△ 76.04%	40.28%	0.2581
TwitterNER	32.80%	26.67%	34.40%	20.09%	0.7273
$ELD_{TwitterNER}$	▲ 57.40%	▲ 60.93%	▲ 78.00%	▲ 47.66%	▽ 0.5337
$DEPICT_{TwitterNER}$	▲ 66.08%	▲ 70.57%	△ 87.67%	△ 56.05%	0.6161
Twitter	44.40%	33.98%	46.40%	30.60%	0.5786
$ELD_{Twitter}$	▲ 60.40%	▲ 64.20%	▲ 80.80%	▲ 51.73%	0.5542
$DEPICT_{Twitter}$	62.13%	66.05%	84.84%	54.80%	0.5431
Juventus - Inter					
NLTK	16.00%	10.42%	16.00%	9.36%	0.6667
ELD_{NLTK}	34.00%	29.17%	36.00%	29.75%	0.0769
$DEPICT_{NLTK}$	42.00%	43.75%	58.00%	28.72%	0.0500
TwitterNER	18.00%	12.50%	18.00%	10.12%	1.0000
$ELD_{TwitterNER}$	34.00%	33.33%	38.00%	25.77%	0.3333
$DEPICT_{TwitterNER}$	84.00%	85.42%	94.00%	72.03%	0.8636
Twitter	30.00%	25.00%	32.00%	14.64%	0.7143
$ELD_{Twitter}$	70.00%	68.75%	90.00%	60.11%	0.9412
$DEPICT_{Twitter}$	60.00%	60.42%	78.00%	55.47%	0.3810
Crystal Palace - Arsenal					
NLTK	30.00%	26.67%	36.00%	18.30%	0.3333
ELD_{NLTK}	46.00%	51.11%	54.00%	39.08%	0.2105
$DEPICT_{NLTK}$	46.00%	51.11%	54.00%	43.00%	0.2778
TwitterNER	30.00%	31.11%	34.00%	19.14%	0.5556
$ELD_{TwitterNER}$	70.00%	77.78%	86.00%	58.00%	0.8421
$DEPICT_{TwitterNER}$	74.00%	82.22%	92.00%	66.77%	0.9474
Twitter	42.00%	33.33%	50.00%	32.12%	0.5000
$ELD_{Twitter}$	74.00%	82.22%	90.00%	61.04%	0.9474
$DEPICT_{Twitter}$	72.00%	80.00%	92.00%	63.94%	1.0000
Manchester City - Atlético					
NLTK	26.00%	10.64%	26.00%	12.02%	0.2500
ELD_{NLTK}	42.00%	42.55%	52.00%	33.77%	0.1765
$DEPICT_{NLTK}$	44.00%	46.81%	54.00%	39.72%	0.2222
TwitterNER	26.00%	21.28%	26.00%	10.56%	1.0000

Model	Precision	Recall	Precision (lenient)	MAP (AP)	Balance
$ELD_{TwitterNER}$	58.00%	61.70%	70.00%	36.69%	0.5263
$DEPICT_{TwitterNER}$	64.00%	68.09%	74.00%	46.24%	0.7778
Twitter	54.00%	31.91%	54.00%	36.64%	0.5000
$ELD_{Twitter}$	62.00%	65.96%	76.00%	52.51%	0.7222
$DEPICT_{Twitter}$	68.00%	72.34%	82.00%	59.77%	1.0000
Burnley - Everton					
NLTK	40.00%	40.00%	40.00%	20.78%	1.0000
ELD_{NLTK}	70.00%	75.56%	94.00%	66.30%	0.5455
$DEPICT_{NLTK}$	70.00%	75.56%	98.00%	61.70%	0.7895
TwitterNER	52.00%	53.33%	52.00%	35.44%	0.8462
$ELD_{TwitterNER}$	70.00%	75.56%	96.00%	68.89%	0.5455
$DEPICT_{TwitterNER}$	72.00%	77.78%	100.00%	67.81%	0.7500
Twitter	70.00%	64.44%	70.00%	52.26%	0.9333
$ELD_{Twitter}$	72.00%	77.78%	96.00%	73.03%	0.5909
$DEPICT_{Twitter}$	72.00%	77.78%	100.00%	70.08%	0.8421
Watford - Leeds					
NLTK	12.00%	8.89%	14.00%	6.69%	0.3333
ELD_{NLTK}	40.82%	40.00%	59.18%	25.58%	0.0000
$DEPICT_{NLTK}$	61.29%	42.22%	90.32%	26.52%	0.0000
TwitterNER	20.00%	15.56%	22.00%	9.05%	0.4000
$ELD_{TwitterNER}$	42.00%	42.22%	60.00%	31.51%	0.0556
$DEPICT_{TwitterNER}$	48.78%	44.44%	70.73%	29.62%	0.0000
Twitter	16.00%	13.33%	20.00%	7.07%	0.5000
$ELD_{Twitter}$	46.00%	44.44%	64.00%	29.78%	0.1111
$DEPICT_{Twitter}$	51.28%	44.44%	74.36%	28.67%	0.0526
Aston Villa - Tottenham					
NLTK	20.00%	13.33%	20.00%	11.26%	0.5000
ELD_{NLTK}	46.00%	51.11%	62.00%	28.61%	0.2105
$DEPICT_{NLTK}$	46.00%	51.11%	62.00%	31.25%	0.0952
TwitterNER	28.00%	20.00%	28.00%	13.62%	0.8000
$ELD_{TwitterNER}$	72.00%	77.78%	94.00%	65.10%	0.6667
$DEPICT_{TwitterNER}$	74.00%	82.22%	98.00%	67.40%	0.8500

Model	Precision	Recall	Precision (lenient)	MAP (AP)	Balance
Twitter	30.00%	24.44%	30.00%	14.01%	0.8333
$ELD_{Twitter}$	70.00%	77.78%	92.00%	62.72%	0.6667
$DEPICT_{Twitter}$	78.00%	86.67%	100.00%	78.73%	0.6957
Manchester City - Liverpool					
NLTK	38.00%	22.22%	38.00%	23.78%	0.6667
ELD_{NLTK}	46.00%	46.67%	84.00%	30.73%	0.6154
$DEPICT_{NLTK}$	58.00%	64.44%	92.00%	41.91%	0.5263
TwitterNER	36.00%	24.44%	36.00%	25.65%	0.5714
$ELD_{TwitterNER}$	50.00%	53.33%	88.00%	40.15%	0.4118
$DEPICT_{TwitterNER}$	56.00%	62.22%	90.00%	47.40%	0.4737
Twitter	58.00%	35.56%	58.00%	49.85%	0.7778
$ELD_{Twitter}$	52.00%	55.56%	88.00%	47.07%	0.4706
$DEPICT_{Twitter}$	64.00%	71.11%	96.00%	58.84%	0.7778
Real Madrid - Chelsea					
NLTK	42.00%	30.00%	42.00%	21.77%	0.2500
ELD_{NLTK}	58.00%	56.00%	70.00%	50.90%	0.4000
$DEPICT_{NLTK}$	56.00%	56.00%	68.00%	48.73%	0.2727
TwitterNER	40.00%	24.00%	40.00%	25.44%	0.2000
$ELD_{TwitterNER}$	54.00%	52.00%	64.00%	49.38%	0.2381
$DEPICT_{TwitterNER}$	50.00%	50.00%	60.00%	43.91%	0.1364
Twitter	50.00%	34.00%	50.00%	29.05%	0.2143
$ELD_{Twitter}$	52.00%	54.00%	62.00%	45.41%	0.2273
$DEPICT_{Twitter}$	50.00%	50.00%	60.00%	42.77%	0.1364
Newcastle - Leicester					
NLTK	24.00%	13.33%	24.00%	13.05%	0.5000
ELD_{NLTK}	48.00%	53.33%	72.00%	38.35%	0.2000
$DEPICT_{NLTK}$	64.71%	48.89%	94.12%	40.36%	0.0476
TwitterNER	30.00%	22.22%	32.00%	21.11%	1.0000
$ELD_{TwitterNER}$	68.00%	75.56%	96.00%	56.42%	0.7895
$DEPICT_{TwitterNER}$	74.00%	82.22%	98.00%	64.01%	0.7619
Twitter	30.00%	22.22%	30.00%	17.86%	0.2500
$ELD_{Twitter}$	44.00%	48.89%	64.00%	36.01%	0.1000

Model	Precision	Recall	Precision (lenient)	MAP (AP)	Balance
$\text{DEPICT}_{\text{Twitter}}$	46.00%	51.11%	68.00%	40.36%	0.0455
Liverpool - Manchester United					
NLTK	50.00%	40.00%	56.00%	36.61%	0.3846
ELD_{NLTK}	56.00%	60.00%	82.00%	42.48%	0.4211
$\text{DEPICT}_{\text{NLTK}}$	52.00%	57.78%	90.00%	40.94%	0.3000
TwitterNER	48.00%	42.22%	56.00%	30.72%	0.9000
$\text{ELD}_{\text{TwitterNER}}$	56.00%	60.00%	88.00%	44.74%	0.9286
$\text{DEPICT}_{\text{TwitterNER}}$	64.00%	71.11%	100.00%	55.31%	0.6000
Twitter	64.00%	55.56%	70.00%	52.53%	0.5625
$\text{ELD}_{\text{Twitter}}$	62.00%	66.67%	86.00%	49.60%	0.7647
$\text{DEPICT}_{\text{Twitter}}$	60.00%	66.67%	98.00%	49.39%	0.5000

Table F.3: ELD's and DEPICT's participant detection results in football matches, summarised in Table 3.4. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the model in the row above.

We concluded the section with a briefer analysis on Formula 1 Grands Prix and another on the 2021 Canadian federal election. Differently from before, we could not evaluate precision leniently in Formula 1 since the same drivers always participated. Furthermore, we could not calculate balance because participants separate into more than two teams or constructors. The full breakdown of the precision, recall and AP results follows in the next table.

Model	Precision	Recall	MAP (AP)
Summary			
NLTK	25.43%	26.73%	11.28%
ELD_{NLTK}	▼ 18.29%	26.73%	△ 15.88%
$\text{DEPICT}_{\text{NLTK}}$	▲ 36.86%	▲ 58.06%	▲ 35.45%
TwitterNER	36.57%	37.33%	33.50%
$\text{ELD}_{\text{TwitterNER}}$	▼ 20.86%	▽ 30.88%	▽ 21.38%
$\text{DEPICT}_{\text{TwitterNER}}$	▲ 34.00%	▲ 53.92%	▲ 35.13%
Twitter	40.86%	38.71%	33.16%

Model	Precision	Recall	MAP (AP)
$ELD_{Twitter}$	▼ 21.14%	30.88%	▽ 21.29%
$DEPICT_{Twitter}$	△ 35.43%	▲ 56.68%	▲ 33.75%
Australian GP			
NLTK	22.00%	19.35%	5.58%
ELD_{NLTK}	14.00%	19.35%	11.49%
$DEPICT_{NLTK}$	42.00%	67.74%	40.54%
TwitterNER	32.00%	29.03%	20.52%
$ELD_{TwitterNER}$	16.00%	22.58%	17.44%
$DEPICT_{TwitterNER}$	30.00%	48.39%	33.21%
Twitter	38.00%	38.71%	22.32%
$ELD_{Twitter}$	22.00%	35.48%	26.60%
$DEPICT_{Twitter}$	40.00%	64.52%	37.76%
Imola GP			
NLTK	34.00%	32.26%	21.49%
ELD_{NLTK}	20.00%	29.03%	21.60%
$DEPICT_{NLTK}$	44.00%	64.52%	58.34%
TwitterNER	38.00%	38.71%	43.57%
$ELD_{TwitterNER}$	20.00%	29.03%	18.30%
$DEPICT_{TwitterNER}$	32.00%	48.39%	34.54%
Twitter	38.00%	35.48%	29.91%
$ELD_{Twitter}$	22.00%	25.81%	18.86%
$DEPICT_{Twitter}$	40.00%	61.29%	37.49%
Spanish GP			
NLTK	20.00%	19.35%	11.06%
ELD_{NLTK}	20.00%	25.81%	18.37%
$DEPICT_{NLTK}$	44.00%	67.74%	49.89%
TwitterNER	30.00%	29.03%	32.56%
$ELD_{TwitterNER}$	20.00%	29.03%	26.46%
$DEPICT_{TwitterNER}$	38.00%	61.29%	47.95%
Twitter	38.00%	35.48%	31.08%
$ELD_{Twitter}$	26.00%	35.48%	37.43%
$DEPICT_{Twitter}$	44.00%	70.97%	58.02%

Model	Precision	Recall	MAP (AP)
Monaco GP			
NLTK	22.00%	25.81%	6.58%
ELD_{NLTK}	12.00%	16.13%	3.47%
$DEPICT_{NLTK}$	18.00%	29.03%	5.72%
TwitterNER	26.00%	32.26%	19.00%
$ELD_{TwitterNER}$	16.00%	19.35%	11.84%
$DEPICT_{TwitterNER}$	32.00%	48.39%	20.89%
Twitter	36.00%	38.71%	22.87%
$ELD_{Twitter}$	8.00%	9.68%	2.73%
$DEPICT_{Twitter}$	10.00%	16.13%	2.70%
Azerbaijan GP			
NLTK	34.00%	38.71%	16.87%
ELD_{NLTK}	24.00%	38.71%	25.20%
$DEPICT_{NLTK}$	42.00%	67.74%	42.82%
TwitterNER	38.00%	45.16%	38.76%
$ELD_{TwitterNER}$	26.00%	41.94%	28.92%
$DEPICT_{TwitterNER}$	38.00%	61.29%	39.21%
Twitter	52.00%	48.39%	50.06%
$ELD_{Twitter}$	26.00%	41.94%	25.83%
$DEPICT_{Twitter}$	42.00%	67.74%	39.86%
Canadian GP			
NLTK	22.00%	19.35%	6.54%
ELD_{NLTK}	16.00%	22.58%	13.41%
$DEPICT_{NLTK}$	34.00%	54.84%	24.47%
TwitterNER	50.00%	45.16%	52.95%
$ELD_{TwitterNER}$	22.00%	32.26%	21.61%
$DEPICT_{TwitterNER}$	32.00%	51.61%	36.50%
Twitter	46.00%	38.71%	50.59%
$ELD_{Twitter}$	22.00%	32.26%	17.05%
$DEPICT_{Twitter}$	34.00%	54.84%	28.92%
British GP			
NLTK	24.00%	32.26%	10.85%

Model	Precision	Recall	MAP (AP)
ELD_{NLTK}	22.00%	35.48%	17.59%
$DEPICT_{NLTK}$	34.00%	54.84%	26.38%
TwitterNER	42.00%	41.94%	27.15%
$ELD_{TwitterNER}$	26.00%	41.94%	25.11%
$DEPICT_{TwitterNER}$	36.00%	58.06%	33.58%
Twitter	38.00%	35.48%	25.30%
$ELD_{Twitter}$	22.00%	35.48%	20.50%
$DEPICT_{Twitter}$	38.00%	61.29%	31.47%

Table F.4: The NER tools' and APD models' participant detection results in Formula 1, summarised in Table 3.5. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the model in the row above.

F.2 | Results from the analyses of Chapter 5

In Chapter 5, we based our evaluation on standard IR metrics: precision, recall and the F-score. We only calculated the micro-average for the first two metrics, precision and recall, and we presented the macro-average F-score and the average number of topics per match. The following table presents a summary of results and the six matches on which we calculated the summary. We only use the last three matches in the last analysis of Section 5.3, and therefore the summary does not include them.

Algorithm	Topics	Precise topics	Precision	Recall	F-score
		Summary			
ELD	37.83	20.33	53.74%	56.73%	55.04%
$ELD_{Filtered}$	▼ 22.83	▼ 16.33	▲ 71.53%	▽ 42.31%	52.09%
SEER	33.83	24.17	△ 71.43%	55.77%	△ 62.89%
		Southampton - Arsenal			
ELD	35	22	62.86%	72.22%	67.22%
$ELD_{Filtered}$	20	17	85.00%	55.56%	67.19%
SEER	32	24	75.00%	66.67%	70.59%
		Leicester - Manchester United			

Algorithm	Topics	Precise topics	Precision	Recall	F-score
ELD	49	29	59.18%	43.48%	50.13%
$ELD_{Filtered}$	30	23	76.67%	43.48%	55.49%
SEER	35	25	71.43%	43.48%	54.05%
Turkey - Italy					
ELD	39	20	51.28%	55.56%	53.33%
$ELD_{Filtered}$	24	18	75.00%	44.44%	55.81%
SEER	36	24	66.67%	50.00%	57.14%
Wales - Switzerland					
ELD	29	19	65.52%	57.14%	61.04%
$ELD_{Filtered}$	19	13	68.42%	50.00%	57.78%
SEER	39	30	76.92%	57.14%	65.57%
Scotland - Czech Republic					
ELD	32	14	43.75%	56.25%	49.22%
$ELD_{Filtered}$	20	10	50.00%	12.50%	20.00%
SEER	24	17	70.83%	62.50%	66.41%
Hungary - France					
ELD	43	18	41.86%	60.00%	49.32%
$ELD_{Filtered}$	24	17	70.83%	46.67%	56.26%
SEER	37	25	67.57%	60.00%	63.56%
Trophée des Champions					
SEER	21	13	61.90%	45.83%	52.67%
Copa Del Rey					
SEER	32	14	43.75%	37.50%	40.38%
Parma - Milan					
SEER	29	21	72.41%	57.69%	64.22%

Table F.5: The TDT algorithms' results in football matches, summarised in Table 5.2a. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the baseline, ELD.

Throughout the chapter, we annotated the algorithmic output manually, assigning one of five labels to each topic's summary, as we explain at the beginning of Section 5.3.

We adopted a strict interpretation of precision, which considers only non-enumerable and enumerable labels as precise. The next table shows a full breakdown of the distribution of labels by all algorithms and in all matches. Similarly to before, the summary only includes the first six matches.

Algorithm	Redundant	Noise	Subjective	Non-enumerable	Enumerable
Summary					
ELD	5.73%	19.82%	20.70%	30.84%	22.91%
$ELD_{Filtered}$	9.49%	6.57%	▽ 12.41%	△ 43.07%	28.47%
SEER	4.43%	▼ 6.90%	17.24%	▲ 45.81%	25.62%
Southampton - Arsenal					
ELD	5.71%	14.29%	17.14%	28.57%	34.29%
$ELD_{Filtered}$	0.00%	15.00%	0.00%	45.00%	40.00%
SEER	0.00%	6.25%	18.75%	40.63%	34.38%
Leicester - Manchester United					
ELD	4.08%	24.49%	12.24%	38.78%	20.41%
$ELD_{Filtered}$	6.67%	6.67%	10.00%	43.33%	33.33%
SEER	11.43%	2.86%	14.29%	42.86%	28.57%
Turkey - Italy					
ELD	5.13%	12.82%	30.77%	35.90%	15.38%
$ELD_{Filtered}$	8.33%	0.00%	16.67%	50.00%	25.00%
SEER	0.00%	8.33%	25.00%	47.22%	19.44%
Wales - Switzerland					
ELD	3.45%	20.69%	10.34%	37.93%	27.59%
$ELD_{Filtered}$	15.79%	5.26%	10.53%	31.58%	36.84%
SEER	2.56%	7.69%	12.82%	56.41%	20.51%
Scotland - Czech Republic					
ELD	12.50%	15.63%	28.13%	18.75%	25.00%
$ELD_{Filtered}$	25.00%	0.00%	25.00%	40.00%	10.00%
SEER	8.33%	4.17%	16.67%	41.67%	29.17%
Hungary - France					
ELD	4.65%	27.91%	25.58%	23.26%	18.60%
$ELD_{Filtered}$	4.17%	12.50%	12.50%	45.83%	25.00%
SEER	5.41%	10.81%	16.22%	43.24%	24.32%

Algorithm	Redundant	Noise	Subjective	Non-enumerable	Enumerable
Trophée des Champions					
SEER	4.76%	28.57%	4.76%	19.05%	42.86%
Copa del Rey					
SEER	28.13%	3.13%	25.00%	25.00%	18.75%
Parma - Milan					
SEER	6.90%	6.90%	13.79%	24.14%	48.28%

Table F.6: The TDT algorithms' annotations in football matches, summarised in Table 5.2b. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the baseline, ELD.

Out of the five labels, we only calculated recall over enumerable topics. We consider redundant, noisy and subjective topics as undesirable, and as the name implies, non-enumerable topics cannot be enumerated. The following table presents a breakdown of recall values of the four types of enumerable topics that we considered: goals, cards, halves and substitutions. Once more, the summary only includes the first six matches.

Algorithm	Goals	Cards	Halves	Substitutions
Summary				
ELD	87.50%	52.94%	37.50%	57.45%
$ELD_{Filtered}$	100.00%	52.94%	29.17%	▽ 25.53%
SEER	93.75%	52.94%	50.00%	46.81%
Southampton - Arsenal				
ELD	100.00%	66.67%	50.00%	75.00%
$ELD_{Filtered}$	100.00%	66.67%	75.00%	25.00%
SEER	100.00%	100.00%	75.00%	37.50%
Leicester - Manchester United				
ELD	33.33%	42.86%	50.00%	44.44%
$ELD_{Filtered}$	100.00%	42.86%	50.00%	22.22%
SEER	100.00%	42.86%	50.00%	22.22%
Turkey - Italy				
ELD	100.00%	50.00%	50.00%	44.44%

Algorithm	Goals	Cards	Halves	Substitutions
$ELD_{Filtered}$	100.00%	50.00%	50.00%	22.22%
SEER	100.00%	0.00%	25.00%	55.56%
Wales - Switzerland				
ELD	100.00%	66.67%	25.00%	50.00%
$ELD_{Filtered}$	100.00%	66.67%	0.00%	50.00%
SEER	66.67%	66.67%	50.00%	50.00%
Scotland - Czech Republic				
ELD	100.00%	0.00%	25.00%	60.00%
$ELD_{Filtered}$	100.00%	0.00%	0.00%	0.00%
SEER	100.00%	0.00%	50.00%	60.00%
Hungary - France				
ELD	100.00%	50.00%	25.00%	71.43%
$ELD_{Filtered}$	100.00%	50.00%	0.00%	57.14%
SEER	100.00%	50.00%	50.00%	57.14%
Trophée des Champions				
SEER	80.00%	0.00%	75.00%	40.00%
Copa del Rey				
SEER	100.00%	0.00%	100.00%	12.50%
Parma - Milan				
SEER	100.00%	44.44%	100.00%	33.33%

Table F.7: The TDT algorithms' recall of enumerable topics in football matches, summarised in Table 5.2c. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the baseline, ELD.

In the next experiments, when we evaluated SEER's streams in Section 5.3, we simplified our evaluation. Recall did not concern us; the behaviour of the streams did, and so we focused on the annotations. We note that two out of 15 streams, with concepts *baller* and ~~*Arsenal*~~, and *clear* and *handball*, generated no topics, and thus we exclude them from the analysis. The breakdown of results for each stream, separated per match, follows in the next table.

Stream	Topics	Precise topics	Precision
Summary			
champion, final, league, football, win	15.33	7.67	50.00%
take, knee, player	11.83	6.17	52.11%
touch, cross, ball, pass	11.33	10.33	91.18%
goal, score, concede, equalise, offside, assist	10.00	7.50	75.00%
need, half, sub, second, lead, 2nd	10.00	7.00	70.00%
keeper, best, goalkeeper, defend, ~~Kepa~~, save	6.83	4.50	65.85%
foul, referee, book, decision, VAR, given, pen, dive, ref, penalty	5.00	3.50	70.00%
gol, stream, online, free, Reddit, link, ~~Manchester~~, FFS, live	4.67	4.00	85.71%
deflect, kick, corner, shot, net	4.50	4.00	88.89%
world, class, striker	3.00	1.00	33.33%
tackle, dribble, yellow, red, card	2.83	2.50	88.24%
~~man~~, utd	1.17	1.00	85.71%
hit, post	0.67	0.67	100.00%
Southampton - Arsenal			
champion, final, league, football, win	11	4	36.36%
take, knee, player	7	4	57.14%
touch, cross, ball, pass	5	4	80.00%
goal, score, concede, equalise, offside, assist	13	10	76.92%
need, half, sub, second, lead, 2nd	11	11	100.00%
keeper, best, goalkeeper, defend, ~~Kepa~~, save	2	2	100.00%
foul, referee, book, decision, VAR, given, pen, dive, ref, penalty	3	3	100.00%
gol, stream, online, free, Reddit, link, ~~Manchester~~, FFS, live	5	5	100.00%
deflect, kick, corner, shot, net	3	3	100.00%
world, class, striker	1	0	0.00%
tackle, dribble, yellow, red, card	3	2	66.67%
~~man~~, utd	0	0	
hit, post	0	0	
Leicester - Manchester United			
champion, final, league, football, win	14	6	42.86%
take, knee, player	7	2	28.57%
touch, cross, ball, pass	10	7	70.00%
goal, score, concede, equalise, offside, assist	8	6	75.00%
need, half, sub, second, lead, 2nd	10	10	100.00%

Stream	Topics	Precise topics	Precision
keeper, best, goalkeeper, defend, ~~Kepa~~, save	10	8	80.00%
foul, referee, book, decision, VAR, given, pen, dive, ref, penalty	7	6	85.71%
gol, stream, online, free, Reddit, link, ~~Manchester~~, FFS, live	5	4	80.00%
deflect, kick, corner, shot, net	2	2	100.00%
world, class, striker	0	0	
tackle, dribble, yellow, red, card	4	3	75.00%
~~man~~, utd	1	1	100.00%
hit, post	7	6	85.71%
Turkey - Italy			
champion, final, league, football, win	16	10	62.50%
take, knee, player	17	7	41.18%
touch, cross, ball, pass	13	11	84.62%
goal, score, concede, equalise, offside, assist	14	7	50.00%
need, half, sub, second, lead, 2nd	11	10	90.91%
keeper, best, goalkeeper, defend, ~~Kepa~~, save	3	2	66.67%
foul, referee, book, decision, VAR, given, pen, dive, ref, penalty	1	1	100.00%
gol, stream, online, free, Reddit, link, ~~Manchester~~, FFS, live	4	4	100.00%
deflect, kick, corner, shot, net	7	6	85.71%
world, class, striker	4	0	0.00%
tackle, dribble, yellow, red, card	11	7	63.64%
~~man~~, utd	1	1	100.00%
hit, post	0	0	
Wales - Switzerland			
champion, final, league, football, win	23	12	52.17%
take, knee, player	18	14	77.78%
touch, cross, ball, pass	15	10	66.67%
goal, score, concede, equalise, offside, assist	12	9	75.00%
need, half, sub, second, lead, 2nd	20	17	85.00%
keeper, best, goalkeeper, defend, ~~Kepa~~, save	9	7	77.78%
foul, referee, book, decision, VAR, given, pen, dive, ref, penalty	3	3	100.00%
gol, stream, online, free, Reddit, link, ~~Manchester~~, FFS, live	7	5	71.43%
deflect, kick, corner, shot, net	7	5	71.43%
world, class, striker	4	2	50.00%
tackle, dribble, yellow, red, card	5	5	100.00%

Stream	Topics	Precise topics	Precision
~~man~~, utd	0	0	
hit, post	0	0	
Scotland - Czech Republic			
champion, final, league, football, win	11	7	63.64%
take, knee, player	6	4	66.67%
touch, cross, ball, pass	7	6	85.71%
goal, score, concede, equalise, offside, assist	7	5	71.43%
need, half, sub, second, lead, 2nd	2	2	100.00%
keeper, best, goalkeeper, defend, ~~Kepa~~, save	5	1	20.00%
foul, referee, book, decision, VAR, given, pen, dive, ref, penalty	0	0	
gol, stream, online, free, Reddit, link, ~~Manchester~~, FFS, live	2	2	100.00%
deflect, kick, corner, shot, net	2	2	100.00%
world, class, striker	1	0	0.00%
tackle, dribble, yellow, red, card	9	7	77.78%
~~man~~, utd	1	1	100.00%
hit, post	0	0	
Hungary - France			
champion, final, league, football, win	17	7	41.18%
take, knee, player	16	6	37.50%
touch, cross, ball, pass	10	7	70.00%
goal, score, concede, equalise, offside, assist	6	5	83.33%
need, half, sub, second, lead, 2nd	14	12	85.71%
keeper, best, goalkeeper, defend, ~~Kepa~~, save	1	1	100.00%
foul, referee, book, decision, VAR, given, pen, dive, ref, penalty	3	2	66.67%
gol, stream, online, free, Reddit, link, ~~Manchester~~, FFS, live	5	4	80.00%
deflect, kick, corner, shot, net	6	6	100.00%
world, class, striker	8	4	50.00%
tackle, dribble, yellow, red, card	9	3	33.33%
~~man~~, utd	1	1	100.00%
hit, post	0	0	

Table F.8: SEER's precision results across all streams in football matches, summarised in Table 5.3a. For clarity, we lemmatised the terms and struck out terms that we had used as tracking keywords, which SEER ignores. Empty cells indicate that the stream did not generate any topics.

To annotate the streams, we followed the exact same process as before. Once again, we annotated each stream's timeline with one of five labels: redundant, noisy, subjective, non-enumerable or enumerable. This time, however, we annotated each of SEER's timelines separately and independently. The breakdown of annotations for each stream, separated per match, follows in the next table.

Stream	Redundant	Noise	Subjective	Non-enumerable	Enumerable
			Summary		
champion, final, league, football, win	6.52%	14.13%	29.35%	23.91%	26.09%
take, knee, player	4.23%	11.27%	32.39%	22.54%	29.58%
touch, cross, ball, pass	1.47%	7.35%	0.00%	70.59%	20.59%
goal, score, concede, equalise $_{+2 \text{ terms}}$	11.67%	0.00%	13.33%	43.33%	31.67%
need, half, sub, second, lead, 2nd	11.67%	1.67%	16.67%	20.00%	50.00%
keeper, best, goalkeeper, defend $_{+2 \text{ terms}}$	4.88%	9.76%	19.51%	43.90%	21.95%
foul, referee, book, decision $_{+6 \text{ terms}}$	6.67%	10.00%	13.33%	46.67%	23.33%
gol, stream, online, free, Reddit $_{+4 \text{ terms}}$	0.00%	7.14%	7.14%	25.00%	60.71%
deflect, kick, corner, shot, net	7.41%	0.00%	3.70%	51.85%	37.04%
world, class, striker	11.11%	11.11%	44.44%	16.67%	16.67%
tackle, dribble, yellow, red, card	0.00%	5.88%	5.88%	29.41%	58.82%
~~man~~, utd	0.00%	0.00%	14.29%	28.57%	57.14%
hit, post	0.00%	0.00%	0.00%	75.00%	25.00%
			Southampton - Arsenal		
champion, final, league, football, win	0.00%	36.36%	27.27%	27.27%	9.09%
take, knee, player	0.00%	14.29%	28.57%	28.57%	28.57%
touch, cross, ball, pass	0.00%	0.00%	0.00%	81.82%	18.18%
goal, score, concede, equalise $_{+2 \text{ terms}}$	0.00%	0.00%	20.00%	20.00%	60.00%
need, half, sub, second, lead, 2nd	7.69%	0.00%	15.38%	23.08%	53.85%
keeper, best, goalkeeper, defend $_{+2 \text{ terms}}$	0.00%	33.33%	0.00%	33.33%	33.33%
foul, referee, book, decision $_{+6 \text{ terms}}$	0.00%	0.00%	0.00%	0.00%	100.00%
gol, stream, online, free, Reddit $_{+4 \text{ terms}}$	0.00%	0.00%	0.00%	20.00%	80.00%
deflect, kick, corner, shot, net	0.00%	0.00%	0.00%	0.00%	100.00%
world, class, striker	0.00%	0.00%	100.00%	0.00%	0.00%
tackle, dribble, yellow, red, card	0.00%	0.00%	0.00%	0.00%	100.00%
~~man~~, utd					
hit, post					
			Leicester - Manchester United		

Stream	Redundant	Noise	Subjective	Non-enumerable	Enumerable
champion, final, league, football, win	0.00%	7.14%	50.00%	28.57%	14.29%
take, knee, player	0.00%	28.57%	42.86%	0.00%	28.57%
touch, cross, ball, pass	0.00%	0.00%	0.00%	90.00%	10.00%
goal, score, concede, equalise +2 terms	20.00%	0.00%	10.00%	40.00%	30.00%
need, half, sub, second, lead, 2nd	0.00%	0.00%	25.00%	25.00%	50.00%
keeper, best, goalkeeper, defend +2 terms	0.00%	25.00%	0.00%	50.00%	25.00%
foul, referee, book, decision +6 terms	10.00%	10.00%	0.00%	60.00%	20.00%
gol, stream, online, free, Reddit +4 terms	0.00%	20.00%	0.00%	20.00%	60.00%
deflect, kick, corner, shot, net	0.00%	0.00%	0.00%	100.00%	0.00%
world, class, striker					
tackle, dribble, yellow, red, card	0.00%	14.29%	0.00%	28.57%	57.14%
~~man~~, utd	0.00%	0.00%	14.29%	28.57%	57.14%
hit, post	0.00%	0.00%	0.00%	100.00%	0.00%
Turkey - Italy					
champion, final, league, football, win	0.00%	12.50%	25.00%	31.25%	31.25%
take, knee, player	11.76%	11.76%	35.29%	17.65%	23.53%
touch, cross, ball, pass	9.09%	0.00%	0.00%	72.73%	18.18%
goal, score, concede, equalise +2 terms	7.69%	0.00%	7.69%	53.85%	30.77%
need, half, sub, second, lead, 2nd	21.43%	0.00%	28.57%	14.29%	35.71%
keeper, best, goalkeeper, defend +2 terms	0.00%	0.00%	36.36%	36.36%	27.27%
foul, referee, book, decision +6 terms	0.00%	0.00%	33.33%	66.67%	0.00%
gol, stream, online, free, Reddit +4 terms	0.00%	0.00%	0.00%	25.00%	75.00%
deflect, kick, corner, shot, net	14.29%	0.00%	0.00%	57.14%	28.57%
world, class, striker	0.00%	0.00%	100.00%	0.00%	0.00%
tackle, dribble, yellow, red, card	0.00%	0.00%	0.00%	100.00%	0.00%
~~man~~, utd					
hit, post	0.00%	0.00%	0.00%	100.00%	0.00%
Wales - Switzerland					
champion, final, league, football, win	13.04%	13.04%	21.74%	30.43%	21.74%
take, knee, player	5.56%	0.00%	16.67%	44.44%	33.33%
touch, cross, ball, pass	0.00%	15.00%	0.00%	60.00%	25.00%
goal, score, concede, equalise +2 terms	13.33%	0.00%	20.00%	40.00%	26.67%
need, half, sub, second, lead, 2nd	0.00%	8.33%	16.67%	33.33%	41.67%
keeper, best, goalkeeper, defend +2 terms	0.00%	0.00%	0.00%	40.00%	60.00%

Stream	Redundant	Noise	Subjective	Non-enumerable	Enumerable
foul, referee, book, decision +6 terms	11.11%	11.11%	0.00%	44.44%	33.33%
gol, stream, online, free, Reddit +4 terms	0.00%	14.29%	14.29%	28.57%	42.86%
deflect, kick, corner, shot, net	14.29%	0.00%	14.29%	42.86%	28.57%
world, class, striker	0.00%	0.00%	50.00%	25.00%	25.00%
tackle, dribble, yellow, red, card	0.00%	0.00%	0.00%	33.33%	66.67%
hit, post					
~~man~~, utd					
Scotland - Czech Republic					
champion, final, league, football, win	9.09%	0.00%	27.27%	9.09%	54.55%
take, knee, player	0.00%	0.00%	33.33%	16.67%	50.00%
touch, cross, ball, pass	0.00%	0.00%	0.00%	50.00%	50.00%
goal, score, concede, equalise +2 terms	0.00%	0.00%	14.29%	71.43%	14.29%
need, half, sub, second, lead, 2nd	28.57%	0.00%	0.00%	0.00%	71.43%
keeper, best, goalkeeper, defend +2 terms	11.11%	11.11%	0.00%	66.67%	11.11%
foul, referee, book, decision +6 terms	0.00%	20.00%	60.00%	20.00%	0.00%
gol, stream, online, free, Reddit +4 terms	0.00%	0.00%	0.00%	0.00%	100.00%
deflect, kick, corner, shot, net	0.00%	0.00%	0.00%	100.00%	0.00%
world, class, striker	0.00%	0.00%	100.00%	0.00%	0.00%
tackle, dribble, yellow, red, card					
~~man~~, utd					
hit, post	0.00%	0.00%	0.00%	100.00%	0.00%
Hungary - France					
champion, final, league, football, win	11.76%	17.65%	29.41%	11.76%	29.41%
take, knee, player	0.00%	18.75%	43.75%	12.50%	25.00%
touch, cross, ball, pass	0.00%	14.29%	0.00%	64.29%	21.43%
goal, score, concede, equalise +2 terms	20.00%	0.00%	10.00%	30.00%	40.00%
need, half, sub, second, lead, 2nd	16.67%	0.00%	0.00%	16.67%	66.67%
keeper, best, goalkeeper, defend +2 terms	11.11%	11.11%	44.44%	33.33%	0.00%
foul, referee, book, decision +6 terms	0.00%	0.00%	0.00%	100.00%	0.00%
gol, stream, online, free, Reddit +4 terms	0.00%	0.00%	20.00%	40.00%	40.00%
deflect, kick, corner, shot, net	0.00%	0.00%	0.00%	50.00%	50.00%
world, class, striker	25.00%	25.00%	0.00%	25.00%	25.00%
tackle, dribble, yellow, red, card	0.00%	0.00%	33.33%	33.33%	33.33%
~~man~~, utd					

Stream	Redundant	Noise	Subjective	Non-enumerable	Enumerable
hit, post	0.00%	0.00%	0.00%	0.00%	100.00%

Table F.9: SEER's annotations across all streams in football matches, summarised in Table 5.3b. For clarity, we lemmatised the terms and struck out terms that we had used as tracking keywords, which SEER ignores. Empty cells indicate that the stream did not generate any topics.

Finally, in the sensitivity evaluation of Appendix B, we reduced dataset sizes to emulate the lower activity of unpopular events. A full breakdown of the number of topics, precision, recall, and F-score across all datasets and at varying amounts of data follows in the next table.

Data	Algorithm	Topics	Precise topics	Precision	Recall	F-score
		Summary				
All tweets	ELD	37.83	20.33	53.74%	56.73%	55.04%
	SEER	33.83	24.17	△ 71.43%	55.77%	△ 62.89%
50,000 tweets	ELD	24.50	14.83	60.54%	43.27%	50.69%
	SEER	△ 27.00	▲ 17.67	65.43%	48.08%	55.54%
25,000 tweets	ELD	12.33	8.17	66.22%	25.00%	36.68%
	SEER	▲ 23.50	▲ 16.00	68.09%	▲ 45.19%	▲ 54.52%
10,000 tweets	ELD	2.67	2.50	93.75%	11.54%	19.67%
	SEER	▲ 27.17	▲ 16.50	▼ 60.74%	▲ 42.31%	▲ 50.34%
		Southampton - Arsenal				
All tweets	ELD	35	22	62.86%	72.22%	67.22%
	SEER	32	24	75.00%	66.67%	70.59%
50,000 tweets	ELD	26	18	69.23%	61.11%	64.92%
	SEER	30	21	70.00%	66.67%	68.29%
25,000 tweets	ELD	12	10	83.33%	38.89%	53.03%
	SEER	29	21	72.41%	66.67%	69.42%
10,000 tweets	ELD	2	2	100.00%	5.56%	10.53%
	SEER	25	18	72.00%	50.00%	59.02%
		Leicester - Manchester United				
All tweets	ELD	49	29	59.18%	43.48%	50.13%
	SEER	35	25	71.43%	43.48%	54.05%
50,000 tweets	ELD	30	16	53.33%	30.43%	38.75%

Data	Algorithm	Topics	Precise topics	Precision	Recall	F-score
	SEER	30	20	66.67%	39.13%	49.32%
25,000 tweets	ELD	13	8	61.54%	13.04%	21.52%
	SEER	24	19	79.17%	34.78%	48.33%
10,000 tweets	ELD	3	2	66.67%	8.70%	15.38%
	SEER	38	23	60.53%	34.78%	44.18%
		Turkey - Italy				
All tweets	ELD	39	20	51.28%	55.56%	53.33%
	SEER	36	24	66.67%	50.00%	57.14%
50,000 tweets	ELD	23	13	56.52%	38.89%	46.08%
	SEER	28	15	53.57%	38.89%	45.06%
25,000 tweets	ELD	15	8	53.33%	16.67%	25.40%
	SEER	21	14	66.67%	33.33%	44.44%
10,000 tweets	ELD	4	4	100.00%	27.78%	43.48%
	SEER	24	16	66.67%	50.00%	57.14%
		Wales - Switzerland				
All tweets	ELD	29	19	65.52%	57.14%	61.04%
	SEER	39	30	76.92%	57.14%	65.57%
50,000 tweets	ELD	24	16	66.67%	50.00%	57.14%
	SEER	24	19	79.17%	64.29%	70.95%
25,000 tweets	ELD	14	10	71.43%	28.57%	40.82%
	SEER	15	13	86.67%	57.14%	68.87%
10,000 tweets	ELD	3	3	100.00%	7.14%	13.33%
	SEER	18	12	66.67%	42.86%	52.17%
		Scotland - Czech Republic				
All tweets	ELD	32	14	43.75%	56.25%	49.22%
	SEER	24	17	70.83%	62.50%	66.41%
50,000 tweets	ELD	21	11	52.38%	43.75%	47.68%
	SEER	24	14	58.33%	43.75%	50.00%
25,000 tweets	ELD	12	6	50.00%	31.25%	38.46%
	SEER	27	14	51.85%	37.50%	43.52%
10,000 tweets	ELD	2	2	100.00%	6.25%	11.76%
	SEER	20	12	60.00%	37.50%	46.15%
		Hungary - France				
All tweets	ELD	43	18	41.86%	60.00%	49.32%

Data	Algorithm	Topics	Precise topics	Precision	Recall	F-score
	SEER	37	25	67.57%	60.00%	63.56%
50,000 tweets	ELD	23	15	65.22%	40.00%	49.59%
	SEER	26	17	65.38%	40.00%	49.64%
25,000 tweets	ELD	8	7	87.50%	26.67%	40.88%
	SEER	25	15	60.00%	46.67%	52.50%
10,000 tweets	ELD	2	2	100.00%	13.33%	23.53%
	SEER	38	18	47.37%	40.00%	43.37%

Table F.10: The TDT algorithms' results in small datasets, summarised in Table B.1a. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the baseline, ELD.

We note that performance does not necessarily decrease with less data. In the match between Turkey and Italy, for example, SEER achieved a very high F-score (57.14%) with just 10,000 tweets. Such anomalies can be explained by the sampling error. Sampling may influence performance directly by over-representing certain non-key topics, but it may also affect performance indirectly: it may choose descriptive tweets that allow the summarisation algorithm to describe several non-key topics at once.

For brevity, we did not discuss our annotation process in Appendix B. However, we followed the same process as before, annotating topics as either redundant, noisy, subjective, non-enumerable or enumerable. The full breakdown across all datasets and at varying amounts of data follows in the next table.

Data	Algorithm	Redundant	Noise	Subjective	Non-enumerable	Enumerable
			Summary			
All tweets	ELD	5.73%	20.26%	20.26%	30.84%	22.91%
	SEER	4.43%	▼ 6.90%	17.24%	▲ 45.81%	25.62%
50,000 tweets	ELD	7.48%	13.61%	18.37%	32.65%	27.89%
	SEER	7.41%	▽ 6.79%	20.37%	▲ 37.65%	27.78%
25,000 tweets	ELD	10.81%	10.81%	12.16%	33.78%	32.43%
	SEER	7.80%	9.93%	14.18%	38.30%	29.79%
10,000 tweets	ELD	0.00%	6.25%	0.00%	25.00%	68.75%
	SEER	▲ 7.98%	12.27%	▲ 19.02%	34.97%	▽ 25.77%

Data	Algorithm	Redundant	Noise	Subjective	Non-enumerable	Enumerable
			Southampton - Arsenal			
All tweets	ELD	5.71%	14.29%	17.14%	28.57%	34.29%
	SEER	0.00%	6.25%	18.75%	40.63%	34.38%
50,000 tweets	ELD	3.85%	7.69%	19.23%	30.77%	38.46%
	SEER	3.33%	6.67%	20.00%	33.33%	36.67%
25,000 tweets	ELD	8.33%	0.00%	8.33%	33.33%	50.00%
	SEER	0.00%	10.34%	17.24%	34.48%	37.93%
10,000 tweets	ELD	0.00%	0.00%	0.00%	50.00%	50.00%
	SEER	4.00%	4.00%	20.00%	36.00%	36.00%
			Leicester - Manchester United			
All tweets	ELD	4.08%	24.49%	12.24%	38.78%	20.41%
	SEER	11.43%	2.86%	14.29%	42.86%	28.57%
50,000 tweets	ELD	6.67%	20.00%	20.00%	30.00%	23.33%
	SEER	10.00%	10.00%	13.33%	36.67%	30.00%
25,000 tweets	ELD	0.00%	23.08%	15.38%	38.46%	23.08%
	SEER	12.50%	0.00%	8.33%	45.83%	33.33%
10,000 tweets	ELD	0.00%	33.33%	0.00%	0.00%	66.67%
	SEER	2.63%	15.79%	21.05%	39.47%	21.05%
			Turkey - Italy			
All tweets	ELD	5.13%	12.82%	30.77%	35.90%	15.38%
	SEER	0.00%	8.33%	25.00%	47.22%	19.44%
50,000 tweets	ELD	4.35%	8.70%	30.43%	30.43%	26.09%
	SEER	10.71%	7.14%	28.57%	35.71%	17.86%
25,000 tweets	ELD	6.67%	20.00%	20.00%	33.33%	20.00%
	SEER	14.29%	9.52%	9.52%	42.86%	23.81%
10,000 tweets	ELD	0.00%	0.00%	0.00%	0.00%	100.00%
	SEER	12.50%	4.17%	16.67%	37.50%	29.17%
			Wales - Switzerland			
All tweets	ELD	3.45%	20.69%	10.34%	37.93%	27.59%
	SEER	2.56%	7.69%	12.82%	56.41%	20.51%
50,000 tweets	ELD	12.50%	8.33%	12.50%	37.50%	29.17%
	SEER	4.17%	8.33%	8.33%	41.67%	37.50%
25,000 tweets	ELD	7.14%	7.14%	14.29%	42.86%	28.57%
	SEER	0.00%	6.67%	6.67%	40.00%	46.67%

Data	Algorithm	Redundant	Noise	Subjective	Non-enumerable	Enumerable
10,000 tweets	ELD	0.00%	0.00%	0.00%	66.67%	33.33%
	SEER	16.67%	5.56%	11.11%	33.33%	33.33%
			Scotland - Czech Republic			
All tweets	ELD	12.50%	15.63%	28.13%	18.75%	25.00%
	SEER	8.33%	4.17%	16.67%	41.67%	29.17%
50,000 tweets	ELD	14.29%	19.05%	14.29%	23.81%	28.57%
	SEER	8.33%	8.33%	25.00%	33.33%	25.00%
25,000 tweets	ELD	41.67%	0.00%	8.33%	16.67%	33.33%
	SEER	7.41%	18.52%	22.22%	33.33%	18.52%
10,000 tweets	ELD	0.00%	0.00%	0.00%	50.00%	50.00%
	SEER	15.00%	10.00%	15.00%	30.00%	30.00%
			Hungary - France			
All tweets	ELD	4.65%	30.23%	23.26%	23.26%	18.60%
	SEER	5.41%	10.81%	16.22%	43.24%	24.32%
50,000 tweets	ELD	4.35%	17.39%	13.04%	43.48%	21.74%
	SEER	7.69%	0.00%	26.92%	46.15%	19.23%
25,000 tweets	ELD	0.00%	12.50%	0.00%	37.50%	50.00%
	SEER	12.00%	12.00%	16.00%	36.00%	24.00%
10,000 tweets	ELD	0.00%	0.00%	0.00%	0.00%	100.00%
	SEER	5.26%	23.68%	23.68%	31.58%	15.79%

Table F.11: The TDT algorithms' annotations in small datasets, summarised in Table B.1a. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the baseline, ELD.

The sensitivity evaluation also returned to a more extensive analysis that considered, once more, recall. The breakdown of the recall of goals, cards, halves and substitutions across all datasets and at varying amounts of data follows in the next table.

Data	Algorithm	Goals	Cards	Halves	Substitutions
		Summary			
All tweets	ELD	87.50%	52.94%	37.50%	57.45%
	SEER	93.75%	52.94%	50.00%	46.81%

Data	Algorithm	Goals	Cards	Halves	Substitutions
50,000 tweets	ELD	87.50%	23.53%	25.00%	44.68%
	SEER	100.00%	41.18%	41.67%	▽ 36.17%
25,000 tweets	ELD	75.00%	11.76%	12.50%	19.15%
	SEER	93.75%	41.18%	▲ 41.67%	△ 31.91%
10,000 tweets	ELD	62.50%	0.00%	4.17%	2.13%
	SEER	△ 87.50%	△ 47.06%	△ 37.50%	▲ 27.66%
		Southampton - Arsenal			
All tweets	ELD	100.00%	66.67%	50.00%	75.00%
	SEER	100.00%	100.00%	75.00%	37.50%
50,000 tweets	ELD	100.00%	66.67%	50.00%	50.00%
	SEER	100.00%	100.00%	75.00%	37.50%
25,000 tweets	ELD	100.00%	0.00%	25.00%	37.50%
	SEER	66.67%	100.00%	75.00%	50.00%
10,000 tweets	ELD	33.33%	0.00%	0.00%	0.00%
	SEER	66.67%	100.00%	50.00%	25.00%
		Leicester - Manchester United			
All tweets	ELD	33.33%	42.86%	50.00%	44.44%
	SEER	100.00%	42.86%	50.00%	22.22%
50,000 tweets	ELD	66.67%	14.29%	25.00%	33.33%
	SEER	100.00%	28.57%	25.00%	33.33%
25,000 tweets	ELD	33.33%	14.29%	0.00%	11.11%
	SEER	100.00%	42.86%	25.00%	11.11%
10,000 tweets	ELD	66.67%	0.00%	0.00%	0.00%
	SEER	100.00%	42.86%	25.00%	11.11%
		Turkey - Italy			
All tweets	ELD	100.00%	50.00%	50.00%	44.44%
	SEER	100.00%	0.00%	25.00%	55.56%
50,000 tweets	ELD	100.00%	0.00%	0.00%	44.44%
	SEER	100.00%	0.00%	0.00%	44.44%
25,000 tweets	ELD	33.33%	0.00%	0.00%	22.22%
	SEER	100.00%	0.00%	25.00%	22.22%
10,000 tweets	ELD	100.00%	0.00%	25.00%	11.11%
	SEER	100.00%	50.00%	25.00%	44.44%
		Wales - Switzerland			

Data	Algorithm	Goals	Cards	Halves	Substitutions
All tweets	ELD	100.00%	66.67%	25.00%	50.00%
	SEER	66.67%	66.67%	75.00%	50.00%
50,000 tweets	ELD	100.00%	33.33%	25.00%	50.00%
	SEER	100.00%	66.67%	75.00%	25.00%
25,000 tweets	ELD	100.00%	33.33%	0.00%	0.00%
	SEER	100.00%	33.33%	50.00%	50.00%
10,000 tweets	ELD	33.33%	0.00%	0.00%	0.00%
	SEER	66.67%	33.33%	50.00%	25.00%
		Scotland - Czech Republic			
All tweets	ELD	100.00%	0.00%	25.00%	60.00%
	SEER	100.00%	0.00%	50.00%	60.00%
50,000 tweets	ELD	50.00%	0.00%	25.00%	50.00%
	SEER	100.00%	0.00%	50.00%	30.00%
25,000 tweets	ELD	100.00%	0.00%	25.00%	20.00%
	SEER	100.00%	0.00%	25.00%	30.00%
10,000 tweets	ELD	50.00%	0.00%	0.00%	0.00%
	SEER	100.00%	0.00%	25.00%	30.00%
		Hungary - France			
All tweets	ELD	100.00%	50.00%	25.00%	71.43%
	SEER	100.00%	50.00%	50.00%	57.14%
50,000 tweets	ELD	100.00%	0.00%	25.00%	42.86%
	SEER	100.00%	0.00%	25.00%	42.86%
25,000 tweets	ELD	100.00%	0.00%	25.00%	14.29%
	SEER	100.00%	0.00%	50.00%	42.86%
10,000 tweets	ELD	100.00%	0.00%	0.00%	0.00%
	SEER	100.00%	0.00%	50.00%	28.57%

Table F.12: The TDT algorithms' recall of enumerable topics in small datasets, summarised in Table B.1b. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to the baseline, ELD.

F.3 | Results from the analyses of Chapter 6

In Section 6.3, we performed a quantitative analysis on SEER as a TDT algorithm. The domain changed, and it changed the way we applied the metrics from Chapter 5. In politics, we could no longer enumerate the innumerable topics, and neither could we calculate recall. Therefore we simplified the evaluation: we annotated topics as redundant, noisy, subjective or newsworthy—the latter an alias for non-enumerable topics but also for precision. The breakdown of annotations follows in the next table.

Algorithm	Redundant	Noise	Subjective	Newsworthy	Topics
		Summary			
ELD	▲ 42.98%	6.61%	▼ 3.31%	▽ 47.11%	▽ 121
$SEER_{Default}$	5.49%	▲ 35.16%	△ 25.27%	▼ 34.07%	▽ 91
SEER	5.42%	7.85%	14.39%	72.34%	535
		September 5			
ELD	55.88%	2.94%	8.82%	32.35%	34
$SEER_{Default}$	0.00%	35.29%	41.18%	23.53%	17
SEER	0.00%	2.44%	17.07%	80.49%	82
		September 6			
ELD	29.41%	11.76%	0.00%	58.82%	34
$SEER_{Default}$	0.00%	30.77%	15.38%	53.85%	13
SEER	6.29%	10.86%	13.14%	69.71%	175
		September 7			
ELD	50.00%	10.00%	0.00%	40.00%	10
$SEER_{Default}$	0.00%	50.00%	10.00%	40.00%	10
SEER	3.26%	9.78%	14.13%	72.83%	92
		September 8			
ELD	36.67%	3.33%	3.33%	56.67%	30
$SEER_{Default}$	13.64%	18.18%	22.73%	45.45%	22
SEER	7.64%	7.64%	14.01%	70.70%	157
		September 9			
ELD	58.33%	8.33%	0.00%	33.33%	12
$SEER_{Default}$	8.33%	25.00%	33.33%	33.33%	12
SEER	10.71%	0.00%	17.86%	71.43%	28
		September 10			

Algorithm	Redundant	Noise	Subjective	Newsworthy	Topics
ELD	0.00%	0.00%	0.00%	100.00%	1
$SEER_{Default}$	5.88%	58.82%	23.53%	11.76%	17
SEER	0.00%	0.00%	0.00%	100.00%	1

Table F.13: ELD's and SEER's results in UK politics, summarised in Table 6.1a. △ and ▲ indicate statistically-significant increases at the 95% and 99% confidence levels, and ▽ and ▼ statistically-significant drops at the 95% and 99% confidence levels (one-tailed paired samples t-test or Wilcoxon Signed-Rank test) compared to SEER.

While ELD and SEER's default stream produced monolithic timelines, SEER itself produced several. As we explain in Section 6.3, we clustered EVATE's top 250 terms from Section 4.5 into 50 concepts: every concept a stream, every stream detecting topics individually. For brevity, we provide the full breakdown in the `NicholasMamo/phd-data` repository, alongside the full list of domain terms. A summary follows in the next table.

Stream	Redundant	Noise	Subjective	Newsworthy	Topics
Summary					
charge, pro-trump, assault, arrest, riot +6 terms	0.00%	16.67%	8.33%	75.00%	12
popular, project, official, affirm, win +4 terms	0.00%	0.00%	0.00%	100.00%	2
foreign, economy, brief, advise, nation +4 terms	15.00%	7.50%	12.50%	65.00%	40
policy, cooperation, administration +6 terms	7.14%	7.14%	7.14%	78.57%	28
work, want, country, congratulate, peace +3 terms	2.17%	8.70%	28.26%	60.87%	46
appeal, lawyer, judge, dismiss, campaign +3 terms	42.86%	0.00%	0.00%	57.14%	7
Chinese, military, raise, defense +4 terms	0.00%	11.11%	0.00%	88.89%	9
lead, state, process, ballot, recount +3 terms	0.00%	22.22%	0.00%	77.78%	9
evidence, Republican, outcome, claim +4 terms	0.00%	20.00%	33.33%	46.67%	15
ally, legal, court, overturn, challenge +3 terms	0.00%	7.69%	15.38%	76.92%	13
turnout, county, find, poll, major, vote, override	7.14%	7.14%	28.57%	57.14%	14
hour, take, spent, time, year, last, office	2.50%	5.00%	27.50%	65.00%	40
senior, agency, director, community +2 terms	0.00%	0.00%	0.00%	100.00%	1
cabinet, nominee, invoke, announce, week, new	0.00%	14.29%	35.71%	50.00%	14
destroy, clemency, trump, people, enemy +1 term	0.00%	15.79%	26.32%	57.89%	19
attempt, democracy, power, try, coup, cheat	0.00%	23.08%	15.38%	61.54%	13
retire, receive, confirm, nominate, secretary, first	13.64%	4.55%	4.55%	77.27%	22
demand, package, stimulus, check +2 terms	0.00%	0.00%	9.09%	90.91%	11
children, justice, family, separate, border +1 term	0.00%	0.00%	0.00%	100.00%	6

Stream	Redundant	Noise	Subjective	Newsworthy	Topics
sign, fund, govern, send, bill, pass	11.11%	0.00%	5.56%	83.33%	18
health, team, education, pick $_{+2 \text{ terms}}$	0.00%	0.00%	11.11%	88.89%	9
transition, lose, lost, concede, acknowledge	0.00%	50.00%	0.00%	50.00%	2
troop, pay, cut, service, order	5.26%	5.26%	5.26%	84.21%	19
effort, reject, victory, reverse, presidential	0.00%	0.00%	0.00%	100.00%	2
counsel, attorney, said, investigate, general	0.00%	11.11%	27.78%	61.11%	18
impeach, article, incite, suspend, twitter	0.00%	100.00%	0.00%	0.00%	1
law, crime, federal, account, criminal	0.00%	0.00%	0.00%	100.00%	2
call, Democrat, governor, pressure, leader	0.00%	9.38%	3.13%	87.50%	32
Russian, media, disinformation, lie	0.00%	0.00%	25.00%	75.00%	4
death, virus, travel, vaccine	0.00%	33.33%	0.00%	66.67%	3
term, next, president, end	33.33%	0.00%	0.00%	66.67%	18
rig, fraud, say, believe	10.53%	0.00%	21.05%	68.42%	19
son, staff, position, appoint	0.00%	0.00%	14.29%	85.71%	7
member, remove, leave, resign	0.00%	0.00%	9.09%	90.91%	11
bid, block, certification					0
wear, rally, mask					0
deliver, message, remark	0.00%	20.00%	0.00%	80.00%	5
dose, refuse, million	0.00%	0.00%	0.00%	100.00%	7
final, second, debate	0.00%	0.00%	20.00%	80.00%	5
press, conference, hold	0.00%	0.00%	33.33%	66.67%	3
early, county, mail	0.00%	0.00%	0.00%	100.00%	1
steal, stop	0.00%	16.67%	0.00%	83.33%	6
money, tax	5.00%	5.00%	10.00%	80.00%	20
terrorist, group					0
woman, vice					0
inauguration, oath					0
attack, die	0.00%	0.00%	0.00%	100.00%	2
runoff, senate					0
violence					0
cast					0

Table F.14: SEER's annotations across all streams in UK politics, summarised in Table 6.1a. For clarity, we lemmatised the terms. Empty cells indicate that the stream did not generate any topics.