Extended No-K-Means for Search Results Clustering

Conference Paper · June 2016

CITATION
1

READS
387

3 authors:

Joel Azzopardi
University of Malta
51 PUBLICATIONS **206** CITATIONS
SEE PROFILE

Chris Staff
University of Malta
20 PUBLICATIONS **111** CITATIONS
SEE PROFILE

Colin Layfield
University of Malta
29 PUBLICATIONS **49** CITATIONS
SEE PROFILE

Extended No-K-Means for Search Results Clustering

Joel Azzopardi, Chris Staff, and Colin Layfield
University of Malta, Msida MSD 2080, Malta
Email: {joel.azzopardi,chris.staff,colin.layfield}@um.edu.mt

Abstract

The No-K-Means clustering algorithm is used for Search Results Clustering. It clusters using a similarity threshold and a Cluster Validity Index to determine cluster membership rather than using prior knowledge of the target number of clusters to create. In this paper, we present an improvement to the algorithm and several new results. We justify the selection of Generalized Dunn's Index as the Cluster Validity Index. We compare results obtained by No-K-Means, Bisecting K-Means, Suffix Tree Clustering, and Lingo on the same Gold Standard collection. No-K-Means achieves even higher accuracy than previously reported when any Wikipedia snippets appearing in the list of results are used to 'seed' clusters. To show that No-K-Means is not dependent Wikipedia results snippets, we remove them from the test and Gold Standard collection and compare No-K-Means and the other clustering algorithms' accuracy. No-K-Means consistently produces better clusters. Finally, we show that No-K-Means's time complexity is favourable compared to other clustering algorithms.

I. Introduction

No-K-Means was developed as an agglomerative partitional on-line clustering algorithm that clusters a stream of news reports by news event, creating a new news event cluster whenever an incoming report does not 'fit' into any existing cluster [2]. The algorithm was refined to apply it to the problem of Search Results Clustering using only the search result snippets themselves during the entire process [21]. Although the number of snippets to cluster is small (a few hundred snippets at a time), and No-K-Means is applied after a list of search results for a given query are obtained, it still used an agglomerative approach, deciding, for each snippet, whether to add it into an existing cluster, if any, and updating the cluster centroid (a simple averaging of term weights), or creating a new cluster with a representation of the result snippet as the initial centroid. The resulting clusters represent the different *senses* of the query in the results list. The algorithm out-performs the state-of-the-art, compared to others who evaluated their approach on the same dataset [8, 10, 15]. In this paper, we present a modified approach that uses the Wikipedia snippets, if any, in the results list as the set of initial cluster centroids (in so doing, modifying it to be a hybrid divisive-agglomerative approach), and to perform Outlier Migration, after the snippets in the results list have been clustered, to identify and move outliers in clusters that are closer to other cluster centroids than to their own. Unlike K-Means (e.g., [13]), which also migrates outliers, No-K-Means updates a cluster's centroid each time a document is classified into it (whereas K-Means updates centroids once per iteration), and No-K-Means migrates outliers in one pass (K-Means migrates outliers until no outliers migrate or a maximum iteration depth is reached).

Rather than intuiting or parameterising the number of clusters, or query senses, to create, No-K-Means generates several cluster configurations at different similarity thresholds to determine cluster membership, using a Cluster Validity Index to identify which configuration is best. In Section II we discuss current approaches to clustering and cluster validity. In Section III we describe Extended No-K-Means and compare its time complexity to that of other clustering algorithms. In Section IV we report the results of several new experiments: to compare several Cluster Validity Indices; to compare our new results to the top performers based on the dataset and Gold Standard created for SemEval 2013 Task 11 (Word Sense Induction and Disambiguation within an End-User Application) [15]; and to compare our results to Lingo, Suffix Tree Clustering, and Bisecting K-Means applied to the same collection.

II. Clustering and Validity

In this section, we discuss document clustering algorithms, focusing on those that estimate or do not need to know in advance the target number of clusters present in a search results list. As a mechanism is needed to recognise the best cluster configuration achievable, we discuss some Cluster Validity Indices.

Jain has a comprehensive survey on data clustering techniques, including those suitable for document clustering [9]. Carpineto *et al.* describe the state-of-the-art in Web clustering [4]. Navigli *et al.* describe the methods used to construct a test dataset and Gold Standard collection for Search Results Clustering (SRC) for the evaluation of approaches to Word Sense Induction and Disambiguation and an automatic evaluator [15]. The documents to cluster are the top-n ranked documents in a results list returned by a search engine following a query.

Clustering algorithms often need to be guided to determine when the best achievable clustering has been reached. This can take the form of the target number of clusters (K) to create, or an internal or external measure of cluster 'goodness'. Usually, even if K is given, as there are potentially large numbers of cluster configurations that would satisfy K, a mechanism is needed to determine when the best clustering has been achieved. One of the most popular cluster-

ing algorithms due to its efficiency, simplicity, and accuracy, is K-Means (e.g., [13]). If the number of clusters, K, is known in advance, K documents from the collection are randomly selected to be the initial cluster centroids and the remaining documents are classified into those clusters. Subsequently, over a number of iterations, the centroids are recalculated and outliers in the clusters are tested to see if there are other clusters to which they are more similar. If so, they are moved. The iteration continues until no outliers are migrated or the iteration maximum is reached. The limitation of the algorithm is that K must be known. X-Means [17] and Y-Means [7] both estimate K by optimising Bayesian Information Criterion or Akaike Information Criterion following a single pass through the collection, or leveraging on statistical properties of the collection to estimate a 'semi-optimal number of clusters' respectively. Alternatively, the collection can be processed to identify candidate cluster 'labels', creating a cluster for each label, classifying documents into the clusters and then performing cluster merging or splitting (e.g., Suffix Tree Clustering [23] and Lingo [16]). SnS/SenseSearcher performs Part-of-Speech tagging on snippets and uses Wikipedia to identify proper nouns [10]. Often, documents that do not properly 'fit' into any of the clusters are merged into an 'Others' cluster.

A common problem with clustering, and, for that matter, information retrieval in general, is identifying the degree to which different fragments of text are semantically similar. If terms are treated at only the syntactic level, then semantically related but different terms could be selected as different cluster labels, and documents containing those terms will be classified into different clusters rather than a single cluster containing semantically related documents. To overcome this, Latent Semantic Analysis (LSA) (e.g., [5, 14]), Latent Dirichlet Allocation (LDA) (e.g., [18]), and Hierarchical Dirichlet Processing (HDP) (e.g., unimelb's HDP-Clusters[12] and Huang *et al.*'s approach [8]) can be applied to 'background' corpora to automatically learn which terms are semantically related in an unsupervised way by analysing how terms co-occur in documents. The background corpora are usually massive text repositories such as Wikipedia, Stanford Core, or the Open Directory Project. Using LSA, clusters will contain documents that are similar, even though they may not share terms, following dimensional reduction to create a semantic space whereby 'latent semantics' is exposed. LDA and HDP are used to discover hidden topics in a massive collection. LDA requires the number of hidden topics as a parameter, but HDP is capable of discovering the hidden topics without supervision. Once the topic hierarchies are captured, the snippets in the search results list are mapped onto the topic hierarchy to yield clusters.

Rather than using a predetermined number of clusters to stop the clustering process, a Cluster Validity Index (CVI) can be used to determine when a good clustering configuration has been achieved [6]. No-K-Means utilises Bezdek and Pal's Generalized Dunn's Index (GDI) [3] to determine the similarity threshold that yields the 'best' clusters. The similarity threshold identified by GDI gives state-of-the-art results for search results clustering [21]. In this paper, we explore other CVIs to determine if Staff *et al.*'s choice of CVI is justified. We selected Silhouette [20], H3 [6], and Score Function [20] based upon prior work [1, 20].

A CVI generates a score for a given cluster configuration, where a cluster configuration is a number of clusters each with one or more members. Typically, cluster configurations containing well spaced out clusters (high average inter-cluster distance) where each cluster contains documents 'bunched' around the centroid (low average intra-cluster distance) receive better scores. The CVIs differ on factors like whether the collection's centroid is taken into account; and whether intra-cluster distance is calculated as an average distance or by taking into account the number of outliers.

• **Silhouette** [20] calculates an average score for the collection based upon, for each document in the collection, the average distance between itself and all other documents in the same cluster, and the average distance between the document and the most similar documents (its nearest neighbours) in other clusters. This indicates how good a fit each document is in its cluster.

• In **H3** [6] compactness is the average distance of cluster members to their centroid, and intra-cluster similarity is the distance between the centroids.

• Saitta *et al.* point out weaknesses with other CVIs and propose **Score Function** (SF) [20]. SF prefers configurations where the inter-cluster distance is maximised and the intra-cluster similarity is minimised. Inter-cluster distance is calculated in relation to the distance of each cluster centroid from the document space centroid, taking into account the total number of clusters and the cluster size.

• **Generalized Dunn's Index** (GDI) has three methods for calculating the inter-cluster distance Δ and six for calculating the intra-cluster distance δ (described in detail in [3]). Staff *et al.* use Δ_3 and δ_4 [21]. GDI was devised to better deal with configurations that include a 'badly behaved' cluster.

III. Extended No-K-Means

No-K-Means is an unsupervised, on-line clustering algorithm that does not require advance knowledge of the number of clusters to create; works on result snippets; does not use external resources, and creates non-overlapping clusters [21].

No-K-Means has been extended to take five parameters: the query; a possibly empty list of results snippets from Wikipedia, if any, appearing in

the list of results snippets; the remaining snippets in the results list; the CVI to use; and whether to apply Latent Semantic Analysis (withLSA) or not (noLSA) to the Term-by-Document matrix created during pre-processing. The snippets are processed to remove stopwords[1] and to stem the remaining terms using the Porter Stemmer [19]. The query terms occurring in the snippets are also removed. A Term-by-Document (snippet) matrix is created using Term Frequency (TF) only as the term weights. Staff *et al.* show that better results are obtained using TF only, that removing query terms from the snippet representations and clustering process significantly improves results, and that noLSA outperforms withLSA [21]. In this paper we focus on the noLSA approach.

In Extended No-K-Means, the motivation for leveraging the Wikipedia snippets is that they often appear in results lists and previous work has used Wikipedia to evaluate search engine results diversification algorithms (e.g., [11]). This suggests that when they are present, they may act as good discriminators between the different senses of the query terms. All topics (queries) in the SemEval test collection have at least one Wikipedia document in their results list (obtained from the Google Search Engine), with a maximum of 25, a median of 4, and an average of 5.73 Wikipedia result snippets per query. In contrast, the Gold Standard for the test collection has an average of 7.69 clusters per query (see Table 1). A cluster is created for each snippet from Wikipedia in the results list, with a representation of the snippet as the centroid. Each remaining snippet is compared to the centroids of all existing clusters using the standard Cosine Similarity Measure to determine the similarity score, *maxSim*, of the cluster to which it is most similar. If *maxSim* is greater than the similarity threshold *simThres* then the snippet is allocated to that cluster and the cluster's centroid is recalculated (by averaging the term weights), otherwise a new cluster is created with the snippet's representation as the first member and initial centroid. This process continues until all snippets are classified. Next, Outlier Migration is performed in a single pass, to move outliers in each cluster to other clusters if they are closer to the latter clusters' centroids. Finally, the singleton clusters (clusters with just one member) are merged into an 'Others' cluster.

When No-K-Means is called with the CVI parameter 'GDI_Varied', the algorithm is run eighteen times for the {query, results list} pair, varying *simThres* in steps of 0.01 between 0.01 and 0.09, and in steps of 0.1 between 0.1 and 0.9, generating a cluster configuration at each *simThres*. Once all cluster configurations have been generated, GDI is used to select the configuration with the highest validity index, which identifies the *simThres* that yielded that configuration. When the parameter value is 'GDI_Fixed', then No-K-Means runs once with the *simThres* provided (Staff *et al.* report 0.04 gives best results [21]).

[1]Using Onix Text Retrieval Toolkit's Stop Word List 1 available from `http://www.lextek.com/manuals/onix/stopwords1.html`.

Table 1: *Previous best performers on the SemEval-2013 Task 11 dataset.*

	F1	**RI**	**ARI**	**JI**	**Ave. # of clusters**	**Ave. cluster size**
HDP-clusters-lemma	68.30	65.22	21.31	33.02	6.63	11.07
HDP-clusters-nolemma	68.03	64.86	21.49	33.75	6.54	11.68
SnS	70.16	65.84	22.19	34.26	8.82	8.46
Huang *et al.*	70.73	66.37	23.34	33.57	na	na
No-K-Means	71.78	68.30	26.19	35.13	8.00	9.49
Gold Standard	100.00	100.00	99.90	100.00	7.69	11.56
Baseline 1: all-in-one	54.42	39.90	0.00	39.00	1.00	64.00
Baseline 2: singletons	100.00	60.09	0.00	0.00	64.00	1.00

Table 2: *Time complexity to cluster* $\boldsymbol{n}$ *documents into* $\boldsymbol{k}$ *clusters using* $\boldsymbol{m}$ *document features.*

Algorithm	**Complexity**
Most hierarchical	$O(n^2)$
Lingo	$O(m^2n + n^3)$
Bisecting K-Means	$O(n)$
STC	$O(n)$
X-Means	$O(kn)$
Y-Means	$O(mkn)$
Extended No-K-Means	$O(mn^2)$

Table 2 shows the time complexity for Extended No-K-Means (noLSA and GDI_Fixed) compared to the time complexity of the clustering algorithms discussed in Section II: Most hierarchical, Lingo, and Suffix Tree Clustering (STC) (all from [4]); Bisecting K-Means [22]; X-Means [17]; and Y-Means [7].

IV. Experiments

We report on the results of several new experiments. First, we experiment with different Cluster Validity Indices. In Subsection IV.II, we compare results obtained by the Lingo, Suffix Tree Clustering, and Bisecting K-Means to those obtained by Extended No-K-Means. Extended No-K-Means obtains best results when Wikipedia snippets occurring in a list of search results are used to seed an initial cluster configuration. In Subsection IV.II.1, we *remove* the Wikipedia snippets from the the test collection and Gold Standard, and compare the accuracy of No-K-Means, K-Means, STC, and Lingo. We show that No-K-Means is more resilient than the other clustering algorithms with and without the presence of Wikipedia snippets in results lists.

I. Comparing CVI performance

Each of the four CVIs that we selected for comparison were given the same cluster configurations generated by Extended No-K-Means (eighteen per query, for all 100 queries in the SemEval test dataset). Table 3 gives the accuracy, using Navigli *et al.*'s automatic WSI-Evaluator [15], of the best cluster config-

uration identified by each CVI; the similarity threshold *simThres* that yields it; and the average number of clusters created (including the 'Others' cluster). As *simThres* increases so does the number of singleton clusters created. In the Fixed setting, the CVI was used to find the best cluster configuration when the similarity threshold was fixed for all queries. For Varied, the similarity threshold was varied for each query (see Section III), and the CVI was used to identify the similarity threshold with the highest validity index for that query, so *simThres* cannot be reported. Despite prior work indicating Generalized Dunn's Index is outperformed by Score Function, H3, and Silhouette (e.g., [1, 20]), we find that GDI identifies a better cluster configuration than the other CVIs. Indeed, none of the other CVIs select configurations that beat the previous best performers (Table 1) on all of the metrics (F1, Rand Index, Adjusted Rand Index, and Jaccard Index). Score Function and H3 prefer configurations with a high *simThres*, resulting in a large number of singleton clusters that are then merged into an 'Others' cluster. They obtain lower WSI-Evaluator accuracy than the configurations identified using GDI. This behaviour can be attributed to the characteristics of the 'Others' cluster No-K-Means creates. Although these CVIs can handle outliers, these are considered to be clusters (in their own right) whose members are 'far', in vector space, from any other cluster, but which exhibit behaviour that is outside of the range of the other clusters (see e.g., [20]). This implies that the members of the outlying cluster do share some common characteristics, but they are generally less similar to each other than the members of other clusters are, on average. However, in the No-K-Means approach to clustering, some initial clusters will be created if there are Wikipedia snippets in the results list, and the remaining results are either classified into one of these clusters if the snippets' similarity to the centroid exceeds some threshold or into other clusters that may be created on-the-fly. Outliers in each cluster are then migrated to some another clusters, if they are more similar to those clusters' centroids than to the one in which they are currently. The 'Others' cluster is created by merging any singleton clusters that remain following the outlier migration step. The 'Others' cluster is therefore likely to contain several snippets that definitely have a pairwise similarity of less than *simThres* (otherwise they would have been classified into the same cluster during the clustering process) but there is no guarantee that their similarity > 0. In vector space, although some of the documents in the Others cluster could be close to each other, but further apart than *simThres*, the cluster could also contain snippets that are much closer to some other cluster, but the similarity to those clusters' centroids falls just below *simThres*. Consequently, because the 'Others' cluster possibly contains snippets that are closer to other clusters than to each other, but it still has a centroid, the CVIs other than GDI generally prefer configurations created with higher values of *simThres*, which result in larger but better behaved 'Others' clusters, but which which obtain low WSI-Evaluator scores. In Extended No-K-Means, GDI is highest when $simThres = 0.03$.

Table 3: *Comparison of different CVIs*

	F1	RI	ARI	JI	Ave. # of clusters	Fixed simThres
Score Function Fixed	55.78	40.37	00.37	37.93	2.18	0.70
Score Function Varied	57.67	43.21	02.58	37.23	3.04	-
H3 Fixed	57.50	41.66	-00.38	35.30	3.49	0.70
H3 Varied	62.02	50.63	11.87	38.21	5.02	-
Silhouette Fixed	74.84	62.37	16.87	25.50	9.94	0.20
Silhouette Varied	75.80	64.58	18.89	26.69	10.72	-
GDI_Fixed	73.91	71.43	29.34	35.87	8.45	0.03
GDI_Varied	74.52	69.26	27.01	34.09	8.94	-

Table 4: *Comparing Extended No-K-Means, Lingo, Bisecting K-Means, and STC*

	F1	RI	ARI	JI	Ave. # of clusters	Ave. cluster size
Lingo	**78.03**	61.46	09.06	11.76	20.08	3.22
Lingo NoQT	76.63	60.58	06.67	10.27	20.44	3.16
STC	71.77	62.96	19.04	27.18	10.11	8.39
STC NoQT	73.02	64.31	18.91	25.08	11.04	6.09
Bisecting	68.80	59.79	01.64	04.30	20.29	3.16
Bisecting NoQT	68.31	59.79	01.62	04.28	20.25	3.17
Extended No-K-Means	73.91	**71.43**	**29.34**	**35.87**	8.45	8.48

Table 5: *Forcing Extended No-K-Means to use Wikipedia clusters only*

	F1	RI	ARI	JI	Ave. # of clusters	Ave. cluster size
Extended No-K-Means	69.19	70.38	27.16	38.93	5.73	14.21

Table 6: *Removing Wikipedia snippets.*

	F1	RI	ARI	JI	Ave. # of clusters	Ave. cluster size
Bisecting K-Means	68.31	64.00	07.82	08.41	18.29	3.19
STC	73.91	68.10	26.52	30.22	9.83	7.82
Lingo	79.45	65.99	15.52	15.58	18.73	3.14
Extended No-K-Means	73.01	73.76	35.75	39.71	7.28	8.86
Gold Standard*	100.00	100.00	100.00	100.00	6.71	12.05

II. Lingo, K-Means, and STC results

We used the same SemEval 2013 Task 11 test dataset[2] as Staff *et al.* [21]. It contains 100 queries and lists of 64 result snippets per query [15]. We ran them through Carrot[2]'s Lingo, Bisecting (K-Means), and STC clustering algorithms[3], each using their default parameters, and evaluated the resulting clusters using the automatic WSI-Evaluator (Table 4). Lingo and STC produce overlapping clusters (a document can be classified into more than one cluster), but Bisecting K-Means produces non-overlapping clusters. The Gold Standard contains

[2] `http://www.cs.york.ac.uk/semeval-2013/task11/`.
[3] Carrot[2] is available from `http://www.carrot2.org`.

non-overlapping clusters. The WSI-Evaluator ignores all but the first mention of a document in a set of clusters. No-K-Means performs clustering after removing the query terms [21], so we also removed the query terms from the snippets before feeding them to the Carrot^2 algorithms (Lingo NoQT, Bisecting (K-Means) NoQT, and STC NoQT in Table 4). Lingo's and K-Means's accuracy decrease when the query terms are removed. STC improves on F1 and RI, but decreases on ARI and JI. Extended No-K-Means outperforms them, as well as the previous best performers on the same dataset reported in Table 1 (which includes the version of No-K-Means reported in [21]). Extended No-K-Means's F1 ($p < 0.05$) and RI ($p < 0.01$) results are statistically significant compared to Huang's using the One Sample T-Test, but the ARI and JI score improvements are not. However, maximising both JI and F1, and both RI and ARI is difficult to achieve [15]. For instance, Lingo has a higher F1 score than No-K-Means, but Baseline 2 (Table 1) shows that a perfect F1 score can be achieved by creating a cluster for each snippet. Lingo has an average of 20.08 clusters; Extended No-K-Means has 8.45; and the Gold Standard has 7.69.

II.1 The role of Wikipedia snippets

X-Means and Y-Means rely on different features of the document collection to choose which documents will be the initial cluster centroids (see Section II). Bisecting K-Means selects two random documents to start the clustering process but then needs to decide how to bisect the subsequent clusters. Extended No-K-Means uses snippets from Wikipedia, if any, in the results list to select the initial cluster centroids, classifying each remaining snippet into those clusters, if the snippet-centroid similarity is high enough, otherwise creating a new cluster for the snippet. If there are no Wikipedia snippets then the snippets are processed in the order received, without pre-creating clusters. In the next experiment, we evaluate the algorithm when non-Wikipedia result snippets are forced into the Wikipedia snippet cluster they are most similar to (Table 5). Snippets that fall below *simThres* are placed into the 'Others' cluster. GDI_Fixed was used to identify which of the eighteen generated cluster configurations had the highest validity index, and this configuration was then run through the WSI-Evaluator. Results are worse than when No-K-Means is unrestricted (Table 4). Finally, in Table 6, we compare the results of the three Carrot^2 algorithms and No-K-Means when all results snippets from Wikipedia are *removed* from the test collection and Gold Standard (giving Gold Standard*). Interestingly, removing the Wikipedia results leads to greater accuracy in all systems, which suggests that the Wikipedia snippets contain elements that cause greater apparent similarity between snippets, but resulting in a lower accuracy measured by the WSI-Evaluator. No-K-Means is more resilient to the presence of Wikipedia snippets, even when they are not selected as the initial cluster centroids (No-K-Means in Table 1 from [21]), but selecting the Wikipedia snippets as the initial cluster centroids improves results (Extended No-K-Means in Table 4).

V. Conclusion

No-K-Means has been used to create sense clusters from lists of query result snippets [21]. It generates cluster configurations for the list of result snippets at different similarity thresholds, using Generalized Dunn's Index (GDI) to identify the similarity threshold that yields the 'best' configuration. In this paper, we discuss an improved algorithm and provide the results of several new experiments. We compared the accuracy of different validity indices to justify the use of GDI, which is more resilient to the presence of a badly behaved 'Others' cluster. We have also compared Bisecting K-Means, Suffix Tree Clustering, and Lingo to No-K-Means on a Search Results Clustering test dataset and Gold Standard and have shown that Extended No-K-Means achieves higher accuracy. It takes advantage of snippets from Wikipedia, if any, that appear in a results list, but still outperforms these other clustering algorithms when the Wikipedia snippets are removed from the test collection and Gold Standard. Additionally, it uses the result snippets only, whereas other approaches using the same test dataset first perform cluster label selection, or apply Latent Semantic Analysis, Latent Dirichlet Allocation, or Hierarchical Dirichlet Processing, usually on massive external collections of text. We also give the time complexity of No-K-Means and compare it to other clustering algorithms. Next, we intend to apply No-K-Means to the challenge of Semantic Textual Similarity, and use it to support interactive cluster space-based navigation through a search results space.

References

[1] Olatz Arbelaitz, Ibai Gurrutxaga, Javier Muguerza, Jesús M. Pérez, and Iñigo Perona. An extensive comparative study of cluster validity indices. *Pattern Recogn.*, 46(1):243–256, January 2013.

[2] Joel Azzopardi and Christopher Staff. Incremental clustering of news reports. *Algorithms*, 5(3):364–378, 2012.

[3] J. C. Bezdek and N. R. Pal. Some new indexes of cluster validity. *Trans. Sys. Man Cyber. Part B*, 28(3):301–315, June 1998.

[4] Claudio Carpineto, Stanislaw Osiński, Giovanni Romano, and Dawid Weiss. A survey of

web clustering engines. *ACM Comput. Surv.*, 41(3):17:1–17:38, July 2009.

[5] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[6] Ahmad El Sayed, Hakim Hacid, and Djamel Zighed. Exploring validity indices for clustering textual data. Zighed, Djamel A. (ed.) et al., Mining complex data. Berlin: Springer. Studies in Computational Intelligence 165, 281-300 (2009)., 2009.

[7] Ali A. Ghorbani and Iosif-Viorel Onut. Y-means: An autonomous clustering algorithm. In *Hybrid Artificial Intelligence Systems, 5th International Conference, HAIS 2010, San Sebastián, Spain, June 23-25, 2010. Proceedings, Part I*, pages 1–13, 2010.

[8] Yanzhou Huang, Xiaodong Shi, Jinsong Su, Yidong Chen, and Guimin Huang. Unsupervised word sense induction using rival penalized competitive learning. *Engineering Applications of Artificial Intelligence*, 41(0):166 – 174, 2015.

[9] Anil K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recogn. Lett.*, 31(8):651–666, June 2010.

[10] Marek Kozłowski and Henryk Rybiński. SnS: A novel word sense induction method. In Marzena Kryszkiewicz, Chris Cornelis, Davide Ciucci, Jesús Medina-Moreno, Hiroshi Motoda, and ZbigniewW. Raś, editors, *Rough Sets and Intelligent Systems Paradigms*, volume 8537 of *Lecture Notes in Computer Science*, pages 258–268. Springer International Publishing, 2014.

[11] Ralf Krestel and Peter Fankhauser. Reranking web search results for diversity. *Inf. Retr.*, 15(5):458–477, October 2012.

[12] Jey Han Lau, Paul Cook, and Timothy Baldwin. Topic modelling-based word sense induction for web snippet clustering. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 217–221, Atlanta, Georgia, USA, June 2013. Association for Computational Linguistics.

[13] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *In 5-th Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, 1967.

[14] Giansalvatore Mecca, Salvatore Raunich, and Alessandro Pappalardo. A new algorithm for clustering search results. *Data Knowl. Eng.*, 62(3):504–522, 2007.

[15] Roberto Navigli and Daniele Vannella. SemEval-2013 task 11: Word sense induction and disambiguation within an end-user application. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 193–201, Atlanta, Georgia, USA, June 2013. Association for Computational Linguistics.

[16] Stanislaw Osinski and Dawid Weiss. A concept-driven algorithm for clustering search results. *IEEE Intelligent Systems*, 20(3):48–54, May 2005.

[17] Dan Pelleg and Andrew W. Moore. X-means: Extending k-means with efficient estimation of the number of clusters. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000), Stanford University, Stanford, CA, USA, June 29 - July 2, 2000*, pages 727–734, 2000.

[18] Xuan Hieu Phan, Minh Le Nguyen, and Susumu Horiguchi. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th International Conference on World Wide Web, WWW 2008, Beijing, China, April 21-25, 2008*, pages 91–100, 2008.

[19] Martin F Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[20] Sandro Saitta, Benny Raphael, and Ian F. Smith. A bounded index for cluster validity. In *Proceedings of the 5th International Conference on Machine Learning and Data Mining in Pattern Recognition*, MLDM '07, pages 174–187, Berlin, Heidelberg, 2007. Springer-Verlag.

[21] Chris Staff, Joel Azzopardi, Colin Layfield, and Daniel Mercieca. Search results clustering without external resources. In Marcus Spies, Roland R. Wagner, and A Min Tjoa, editors, *Proceedings of the 26th International Workshop on Database and Expert Systems Applications DEXA 2015, Valencia, Spain, September 1-4, 2015*, pages 276–280, 2015.

[22] M. Steinbach, G. Karypis, and V. Kumar. A comparison of document clustering techniques. In *6th ACM SIGKDD, World Text Mining Conference*, 2000.

[23] Oren Zamir and Oren Etzioni. Grouper: A dynamic clustering interface to web search results. *Computer Networks*, 31(11-16):1361–1374, 1999.