Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Technology 00 (2012) 000-000

Procedia Technology

www.elsevier.com/locate/procedia

2nd International Conference on Communication, Computing & Security [ICCCS-2012]

Heuristic Frequent Term-Based Clustering of News Headlines

Nibir Nayan Bora[a]*, Bhabani Shankar Prasad Mishra[a]*, Satchidananda Dehuri[b]

[a]*School of Computer Engineering, KIIT University, Bhubaneswar, India 751024*
[b]*Department of Information and Communication Technology, Fakir Mohan University, Balsore, India 756019*

Abstract

Document clustering deals with assigning documents to groups (called clusters) in accordance with the general clustering rule, '*high intra-cluster document similarity and low inter-cluster document similarity*'. In this study, we propose a novel heuristics for clustering news headlines. News headlines are grammatically and semantically different from larger bodies of text, like blog posts and reviews. Based on the heuristics, we implemented versions of the *frequent term-based* and *frequent noun-based* clustering algorithms. Both these algorithms, along with k-means, regular frequent term and frequent noun clustering were evaluated using five datasets - Reuters343, Reuters2388 (news headlines), CICLing-2002, Hep-ex and KnCr (scientific abstracts). On interpreting the results based on common external cluster quality evaluation measures (*purity*, *entropy* and *F measure*), it was found that the heuristics performed at par with, or even better than, traditional clustering algorithms and few other intuitive algorithms, when tested using the datasets comprising of news headlines. However, on using the datasets comprising of scientific abstracts, the results were not favorable.

Selection and/or peer-review under responsibility of the Department of Computer Science & Engineering, National Institute of Technology Rourkela.

Keywords: Document clustering; cluster evaluation; frequent term clustering; k-means

1. Introduction

Cluster analysis or *clustering* is the task of assigning a set of objects into groups (called *clusters*) so that the objects in the same cluster are more similar (in some sense or another) to each other than to those in other clusters. Cluster analysis finds applications in numerous fields. *Document clustering* (also referred to as *text categorization*) is closely related to the concept of data clustering. Document clustering is a more specific technique for unsupervised document organization, where generally high dimensional documents are the objects in consideration. A large number of studies has investigated document clustering as a methodology for improving search and retrieval, automatic topic identification, document browsing, as well as the primitive task of classification.

* Corresponding author.
E-mail address: nibirbora@gmail.com, mishra.bsp@gmail.com

News headlines are different from previously studied larger bodies of text, such as search results, blog posts and reviews. They are constricted, succinct (often limited to a single sentence) and do not guarantee correctness of grammar. This results in smaller dimensionality of a corpus, allowing us to try modified versions of the regular clustering methods as well as various heuristics. Clustering headlines instead of the news posts themselves could provide a significant gain in execution time. Further, an efficient algorithm for clustering of news headlines may find many applications, like creating groups of similar news from different sources[1].

In this paper, we propose a novel heuristics which could be incorporated into frequent term-based clustering to yield better results. We implemented heuristic versions of both frequent term and frequent noun clustering algorithm. These algorithms were then evaluated using two datasets (*Reuters343* and *Reuters2388*) of 343 and 2388 news headlines pertaining to 26 topics (extracted from the Reuters corpus). We also tested our heuristics on three other short text corpuses namely CICLing-2002, hep-ex, and KnCr (comprising of scientific abstracts). The results were then interpreted using commonly used cluster quality measures (*purity*, *entropy*, *F measure*). Since we are concerned with the clustering quality of the algorithms and do not investigate clustering with an application in mind, the evaluation parameters are external measures which assess the results of the clustering algorithms against a pre-defined set of classes (or categories).

The next section is a summary of different studies in document clustering, classification of clustering algorithms, and various innovative algorithms. Our heuristics is introduced in section 3 and the clustering algorithms, defined. Section 4 accounts for the datasets used to evaluate the results and the evaluation parameters. Results of the studies are discussed in section 5 and a final conclusion is drawn in section 6.

2. Background

Clustering algorithms can broadly be classified as partitional or hierarchical (Han & Kamber 2006). Partitional algorithms partition the document space into a specific number of groups, using the iterative relocation principle. *K-means*, *k-medioids*, *bisecting k-means* (Steinbach, et al. 2000) are a few examples of partitional algorithms. Hierarchical clustering algorithms, on the other hand, begin with all the documents in the document space as individual clusters and iteratively merge the most similar clusters. In contrast to this bottom-up approach (also called the *agglomerative* approach), hierarchical clustering can be top down (or the *divisive* approach). While it has often been argued that hierarchical clustering yields better quality clusters, partitional methods are preferred often because of their linear time complexity. Apart from this, Carpineto et al. (Carpineto, et al. 2009) proposed an alternate classification of clustering algorithms based on how well they are prepared to produce sensible, comprehensive, and compact cluster labels. However, this classification type is based on clustering as a technique to improve browsing of search results.

Over the years, a number of intuitive clustering algorithms have been developed. Cutting et al. (Cutting, et al. 1992) introduced two linear time partitional clustering algorithms - Buckshot and Fractionation, both being used as methods to choose initial centers for their cluster subroutine. Cutting et al. also pointed out that Buckshot is a faster clustering algorithm whereas Fractionation produced a better clustering. Beil et al. (Beil, et al. 2002) discussed two variants of frequent term-based clustering, one partitional and the other hierarchical. Both the algorithms are based on a greedy technique to identify frequent term sets. Zamir and Etzioni (Zamir & Etzioni 1998) introduced another interesting linear time clustering algorithm, Suffix Tree Clustering (STC). One notable difference between STC and previously discussed algorithms is that STC does not treat a document as a set of words, but as an ordered sequence of words. This algorithm proceeds by constructing a suffix tree of all the sentences of all the documents in the corpus. Apart from these, there are a number of proposed clustering algorithms (Guha, et al. 2000) (Pantel & Lin 2002) (Steinbach et al. 2000) each with their benefits.

A popular aspect of investigating document clustering as an application is in improving web search results, towards which many studies have been directed ((Maarek, et al. 2000) (Zamir & Etzioni 1998) (Zamir & Etzioni 1999) (Cutting et al. 1992)). However, our heuristic is motivated by the recent trend in clustering short-text. Shrestha et al. (Shrestha, et al. 2012), in their study of clustering short text evaluated variants of hierarchical agglomerative clustering and spectral clustering (Luxburg 2007) on four different short text corpora - CICLing-2002, hep-ex, and KnCr (Pinto,

[1] Such an approach is used by Google News.

et al. 2007) and LDC (paragraphs of news posts concerning the *Death of Diana*). Few other studies have dealt with clustering abstracts instead of full text in papers (Makagonov, et al. 2004) (Pinto, et al. 2006) and other short text domains (Pinto & Rosso 2006) (Koller & Sahami 1997). Zamir and Etzioni (Zamir & Etzioni 1998) also found that clusters based on snippets were almost as good as clusters created using full text of web documents.

3. Clustering Algorithm

In this section we introduce the novel heuristics that we propose, and discuss in details the clustering algorithms being used. This has been followed up with an account of the refinement process.

Before feeding our corpus of headlines (and abstracts) to the clustering algorithms, common preprocessing steps are applied and each feature is separated. We consider only *unigram* features, and use *feature presence* measure to represent in the *vector space model*. In preliminary experiments, we found that considering *term frequency * inverse document frequency* had an adverse effect as the total dimensionality of the corpus was substantially low.

3.1. The Hypothesis

The heuristics is intended to be used for modifying the general frequent term-based clustering algorithm, which forms clusters by first finding the most frequent terms in the corpus and them bundling the documents containing each of these terms into a cluster. This is based on the premise that each document is related to a topic which is manifested by the presence of a related term (*key term*). We aim at making the process of finding these key terms easy.

Our hypothesis states, "*the lesser the number of terms in a document, the easier it is to identify the key term.*" Evidently, this obligates the assumption that each document contains only one such key term. To understand this hypothesis, consider two news headlines:

H1: *BALDRIGE PREDICTS SOLID U.S. HOUSING GROWTH*

H2: *JANUARY HOUSING SALES DROP, REALTY GROUP SAYS*

Both these headlines belong to the class "HOUSING". After preprocessing, H1 is left with 4 terms, while H2 has 6 terms. Table 1 shows the terms (and corresponding frequencies in the entire corpus) of both H1 and H2 as well as the list of terms in H1 and H2 combined, in order of their occurrence in the corpus. On proceeding with the general frequent term-based clustering algorithm, it is noticed that `januari` will be chosen as a key term, followed by `sale` and so on. However, there are no classes `januari` or `sale` in the corpus. While applying our hypothesis, we apply the frequent term-based clustering algorithm, to each document separately and try to find a key term in each (although term frequency in the entire corpus is considered). Now, on processing H2 first, `januari` will be chosen as a key term, which again is incorrect. The correct key term, `hous` comes third in H2. On the other hand, processing H1 (which has 4 terms) directly identifies `hous` as the key term, which is correct. Thus, it is easier to identify the key term from a document with less number of terms.

Table 1: Illustration of the hypothesis. (Terms are stemmed to their roots)

Entire corpus		H1 *(4 terms)*		H2 *(6 terms)*	
Term	Freq.	Term	Freq.	Term	Freq.
januari	12			januari	**12**
sale	10			sale	10
hous	9	*hous*	**9**	*hous*	9
group	6			group	6
growth	5	growth	5		
baldrig	2	baldrig	2		
drop	2			drop	2
solid	2	solid	2		
realti	1			realti	1

3.2. The Heuristics Algorithms

We built two heuristic algorithms - Frequent Term *plus*, which is the heuristic version of the frequent term-based clustering algorithm and Frequent Noun *plus*, which has the modification that all non-nouns are removed in preprocessing. Algorithm 1 is a representation of the Frequent Term *plus* algorithm.

While implementing the heuristics in clustering algorithms, the number of terms (after preprocessing) in each document is counted. The documents are then arranged in reverse order of number of terms in each. For each document, an attempt is made to identify the key term. Whenever a key term is found, all unclustered documents containing the key term are gathered to form a cluster. While finding a key term, each term is checked if it occurs more in unclustered documents. This is followed till the third most occurring term in a document.

Algorithm 1 Frequent Term *plus* algorithm

Require: *featureList*; {List of features and their frequencies in the corpus}
Require: *documentList*; {List of documents and the number of features contained in each}
1: reverseSort(*documentList*);
2: **for all** *document* in *documentList* **do**
3: sort(*document*) {Sort terms in *document* in order of frequency in corpus}
4: **repeat**
5: Select a *term* from the *document*
6: **if** countTermIn(*term*, *unclusteredDocs*) > countTermIn(*term*, *clusteredDocs*) **then**
7: createCluster(*term*)
8: update(*unclusteredDocs*)
9: **else**
10: Skip *term*
11: **end if**
12: **until** *Key term* found **or** 3 iterations over
13: **end for**
14: applyRefinement()

3.3. Refinement

Refinement is applied to the clustering, produced by an algorithm, to possibly increase the quality of the clusters. In out experiments, all clusters with three or less number of documents are marked as unclustered, and these documents are assigned to the most similar cluster. Further, k-means algorithm is applied to the remaining clusters, taking them as the initial seeds.

4. Evaluation

The evaluation technique used in this study focuses on the overall quality of the clusters produced by the algorithm. This is essentially an external measure of cluster quality, as indicated by Steinbach et al. (Steinbach et al. 2000) and in (Manning, et al. 2008), which requires the corpus to be pre-categorized into classes by a human. We evaluate our heuristic algorithms using five different datasets, and compare the results with general frequent term-based clustering (as well as the frequent noun version) and traditional k-means algorithm. Comparisons are also made with few well known related studies.

4.1. The Datasets

Five different datasets were used to evaluate the results of our clustering algorithms. The datasets along with the number of documents in each and their class distribution is described in Table 2. Since our study was motivated by the unique nature of news headlines, the first two corpuses (Reuters343 and Reuters2388) consist purely of news

Table 2: Datasets used for evaluation of results.

Name of Dataset	Document type	No. of docs.	No. of classes	Class distribution
Reuters343	headlines	343	26	$\{24,23,22,21,21,20,19,19,19,15,14,13,12,12,11,10,10,8,8,8,8,7,5,5,5,4\}$
Reuters2388	headlines	2388	26	$\{4,6,13,16,18,18,21,23,30,35,37,37,37,39,46,47,55,78,94,111,126,181,197,212,369,538\}$
CICLing-2002	abstracts	48	4	$\{11,15,11,11\}$
Hep-ex	abstracts	2922	9	$\{2623,271,18,3,1,1,1,1,1\}$
KnCr	abstracts	900	16	$\{169,160,119,99,66,64,51,31,30,29,22,20,14,12,8,6\}$

headlines. Both these datasets were created by extracting headlines from the 90 category split of Reuters-21578 corpus used by Joachims (Joachims 1998). The other three corpora namely CICLing-2002, Hep-ex, and KnCr, were created from scientific abstracts, and have been used previously for short text clustering ((Pinto et al. 2007), (Shrestha et al. 2012)). The abstracts in the CICLing-2002 dataset belong to the field of computational linguistics collected from the CICLing 2002 conference. Those in the Hep-ex datasets are from the domain of Physics, while the KnCr corpus is accumulated from the MEDLINE documents (Pinto & Rosso 2006).

4.2. Cluster Quality Measures

The three cluster quality measures used to interpret the output of the algorithms are Purity, Entropy and F measure. Their mathematical expressions, as described below, are defined for the set of classes $C = c_1, c_2, ..., c_i$ and set of clusters $K = k_1, k_2, ..., k_j$. n is the total number of documents in the corpus, n_i the size of class i, n_j the size of cluster j and n_{ij} is the number of documents of class i in cluster j. p_{ij} is the probability that a member of cluster j belongs to class i.

- *Purity* $(K,C) = \frac{1}{n}\sum_j \max_i |k_j \cup c_i|$
- *Entropy* $E_K = \frac{1}{n}\sum_j n_j \sum_i -p_{ij} log(p_{ij})$
- *F measure* $F_K = \frac{1}{n}\sum_i n_i \max_j F(i,j)$
  The calculation of overall F measure requires the calculation of F measure of cluster j and class i using the formula, $F(i,j) = \frac{2*Recall(i,j)*Precision(i,j)}{Recall(i,j)+Precision(i,j)}$, where, $Precision(i,j) = \frac{n_{ij}}{n_j}$ and $Recall(i,j) = \frac{n_{ij}}{n_i}$

Purity values range between 0 and 1, with the purity measure of a good clustering approaching 1 and vice versa. It has also been pointed out in (Manning et al. 2008) that high purity can be easily achieved if the number of clusters is large, or purity is 1 if each document gets its own cluster. Thus, purity measure cannot alone be used to trade off the quality of the clustering against the number of clusters. Entropy, a measure introduced by Shannon (Shannon 2001), is an external evaluation method which accounts for the 'goodness' of a flat clustering, as indicated by Steinbach et al. (Steinbach et al. 2000). However, just like purity measure, maximum entropy is achieved when each cluster contains only one document. F measure is another external quality measure which combines the precision and recall ideas from information retrieval.

5. Results

For evaluation of our proposed heuristics, we implement four algorithms, in addition to k-means clustering algorithm. The five algorithms in comparison are:

K-means (*regular*)
Frequent-term (*regular*) [FT]
Frequent-nouns (*regular*) [FN]
Frequent-term *plus* (*heuristic*) [FT*plus*]
Frequent-nouns *plus* (*heuristic*) [FN*plus*]

The results discussed in this section were obtained on performing the experiments on an Intel Core 2 Duo machine with clock frequency 2.4GHz and 1GB of primary memory.

Table 3(a) and Figure 1(a) summarizes the cluster quality evaluation measures obtained for each of the five algorithms using the Reuters343 dataset. To determine the quality of the clusters produced in terms of these values, note that for purity measure, the closer it is to 1, the better is the clustering. Similarly, for F measure, the larger the F measure, the better is the cluster quality. However, in case of entropy, it is just the opposite. The lesser the entropy measure, the better is the clustering produced. In short, the cluster quality is directly proportional to the purity and F measure values, whereas it is inversely proportional to the entropy measure.

Similarly, Tables 3(b) to 3(e) and Figures 1(b) to 1(e) summarizes the cluster quality evaluation measures obtained for each of the algorithms using the Reuters2388, CICLing-2002, Hep-ex and KnCr datasets respectively. For the later three datasets it was not possible to evaluate the FN and FN*plus* algorithms, as these datasets were available in preprocessed (and stemmed) form.

As evident in Figure 1(a) (using Reuters343 dataset), frequent noun *plus* produced the best cluster quality. Both the heuristic algorithms, frequent term *plus* and frequent noun *plus* showed improved performance compared to the regular frequent term and frequent noun clustering algorithms. Moreover, k-means clustering produced the poorest cluster quality. On observing the evaluation measures, it was noticed that the purity and F measure values increase in the order k-means, FT, FT*plus*, FN, FN*plus*, and entropy measure decreases in the same order. The quality of the clusters produced by each of these algorithms can hence be arranged in this order.

In Figure 1(b) (using Reuters2388 dataset), although the entropy measure decreases in the order k-means, FT, FT*plus*, FN, FN*plus*, the purity and F measure values remain almost same in each case. Judging only in terms of entopy value, FN*plus* produces the best clustering. A thing to be noted in this regard is that, the lesser is the entropy value, the less out-of-place documents are in the produced clusters (relate to entropy in chemical atoms).

Using the other three datasets (CICLing-2002, Hep-ex, and KnCr), the results were a little absurd. With the CICLing-2002 dataset (Figure 1(c)) and KnCr dataset (Figure 1(e)) each of the purity, entropy and F measure values remained almost constant (with a little drop or rise in FT) for all algorithms. In case of the Hep-ex dataset (Figure 1(d)) k-means was the best performer in terms of all evaluation measure. In fact, FT and FT*plus* performed worse.

Although the average F measure values in most studies (Larsen & Aone 1999) (Steinbach et al. 2000) lies around 0.5 and 0.6, our study projects a much higher score (using Reuters343 dataset). This may be attributed to two significant dimensions of our study, which makes it different from many related work - *1*. News headlines are essentially small text, compared to larger bodies of text like news posts and reviews. *2*. The corpus used for evaluation is small and hand manipulated. However, Beil et al. (Beil et al. 2002) reported a F measure similar to that of FT*plus* with their implementation of bisecting k-means on a corpus of papers related to aeronautical system, medicine and information retrieval. They also reported entropy values similar to the ones achieved by us, although on evaluating using the Reuters corpus, the entropy was much higher. Steinbach et al. (Steinbach et al. 2000) also reported

Table 3: Cluster quality measures for each algorithm.

(a) Reuters343

	k-means	FT	FT*plus*	FN	FN*plus*
Purity	0.57	0.63	0.64	0.72	0.83
Entropy	1.09	0.83	0.59	0.55	0.27
F measure	0.50	0.62	0.68	0.76	0.86

(b) Reuters2388

	k-means	FT	FT*plus*	FN	FN*plus*
Purity	0.4	0.34	0.3	0.34	0.28
Entropy	1.51	1.12	1.01	0.93	0.91
F measure	0.44	0.4	0.38	0.42	0.38

(c) CICLing

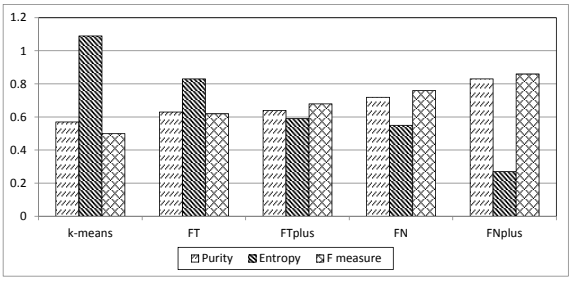
	k-means	FT	FT*plus*
Purity	0.58	0.52	0.56
Entropy	1.18	1.02	1.17
F measure	0.49	0.42	0.37

(d) Hep-ex

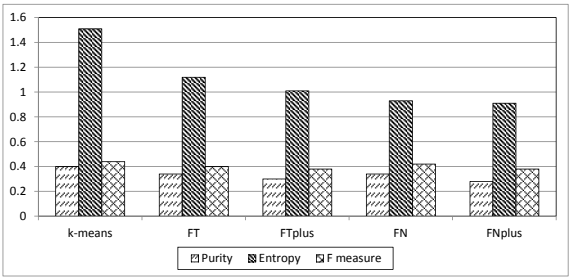
	k-means	FT	FT*plus*
Purity	0.22	0.07	0.07
Entropy	0.22	0.13	0.1
F measure	0.33	0.13	0.12

(e) KnCr

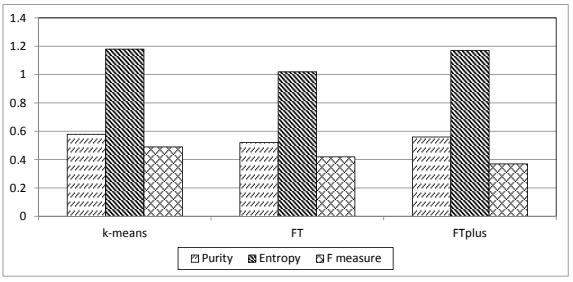
	k-means	FT	FT*plus*
Purity	0.18	0.31	0.12
Entropy	0.82	1.03	0.83
F measure	0.18	0.18	0.14



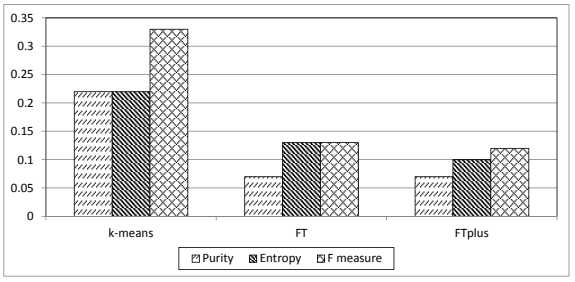
(a) Reuters343



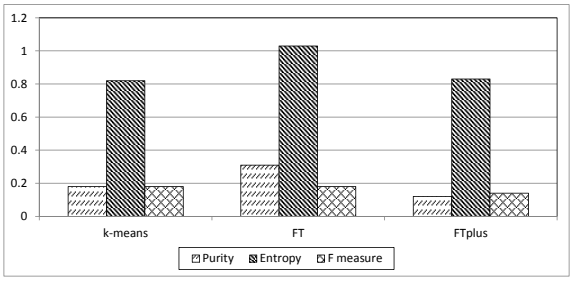
(b) Reuters2388



(c) CICLing



(d) Hep-ex



(e) KnCr

Fig. 1: Cluster quality measures for each algorithm.

much lower entropy values in their original implementation of bisecting k-means algorithm. Thus, it can be inferred that our heuristics perform comparably (or even better) with traditional clustering methods as well as other intuitive algorithms.

While the results obtained using the Reuters343 dataset indicated enhanced performance with the heuristic algorithms, the results were not consistent with the Reuters2388 dataset. A possible explanation of this outcome is that, while our heuristics was aimed at identifying the key term corresponding to a class and subsequently building a cluster using it, the Reuters2388 corpus comprises of numerous headlines which do not contain the same key term as other documents in the class (and sometimes do not contain a key term at all). As each document contains a limited number of features, finding a secondary key term is difficult and thus leads to a certain degree of noise.

The absurd cluster quality measures obtained while using the CICLing-2002, Hep-ex and KnCr datasets can be explained as a result of the different nature of these datasets. Unlike news headlines, these datasets comprises of scientific abstracts, which are full-fledged paragraphs with multiple complete sentences. This leads to a higher dimensionality of the corpus. Further, this can also be attributed to the uneven class distribution of documents in these datasets.

6. Conclusion

We proposed a novel heuristics which could be incorporated into the regular frequent term-based clustering and frequent noun-based clustering algorithms to produce better results. The heuristics was implemented along with traditional k-means, frequent term and frequent noun clustering, evaluated using five datasets - Reuters343, Reuters2388 (news headlines), CICLing-2002, Hep-ex and KnCr (scientific abstracts), and the results interpreted based on purity, entropy and F measure.

On investigating the results, it was concluded that, using a small noiseless dataset of news headlines (Reuters343) our heuristic clustering algorithms performed at par with, or even better than, traditional clustering algorithms and few intuitive algorithms like bisecting k-means, buckshot and fractionation. The general order of increasing performance was k-means, FT, FT*plus*, FN, FN*plus*, based on all three evaluation measures. However, with a larger and noisier dataset of news headlines (Reuters2388) the performance variations were mild; Although, based on a single evaluation measure (entropy), the general order of increasing performance remained the same. While using datasets comprising of scientific abstracts (CICLing-2002, Hep-ex and KnCr) there was no apparent performance increase. In fact, in one case (using Hep-ex dataset), the cluster quality of frequent term and frequent term *plus* algorithms degraded over k-means.

Future work may include finding a method to reduce the dimensionality of short text corpuses. The heuristics may also be extended to news headlines and description (instead of headlines only) to address the problem of noisy headlines. Further, *word sense disambiguation* can be done as news headlines are indited by experts and they usually follow a variety of linguistic styles.

References

F. Beil, et al. (2002). 'Frequent term-based text clustering'. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pp. 436–442, New York, NY, USA. ACM.

C. Carpineto, et al. (2009). 'A survey of Web clustering engines'. *ACM Comput. Surv.* **41**(3):17:1–17:38.

D. R. Cutting, et al. (1992). 'Scatter/Gather: a cluster-based approach to browsing large document collections'. In *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '92, pp. 318–329, New York, NY, USA. ACM.

S. Guha, et al. (2000). 'ROCK: A Robust Clustering Algorithm for Categorical Attributes'. In *In Proc.ofthe15thInt.Conf.onDataEngineering*.

J. Han & M. Kamber (2006). *Data Mining: Concepts and Techniques*. The Morgan Kaufmann Series in Data Management Systems. Elsevier.

T. Joachims (1998). 'Text Categorization with Support Vector Machines: Learning with Many Relevant Features'.

D. Koller & M. Sahami (1997). 'Hierarchically Classifying Documents Using Very Few Words'. In *Proceedings of the Fourteenth International Conference on Machine Learning*, ICML '97, pp. 170–178, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

B. Larsen & C. Aone (1999). 'Fast and effective text mining using linear-time document clustering'. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '99, pp. 16–22, New York, NY, USA. ACM.

U. Luxburg (2007). 'A tutorial on spectral clustering'. *Statistics and Computing* **17**(4):395–416.

Y. S. Maarek, et al. (2000). 'Ephemeral Document Clustering for Web Applications'. Tech. rep., IBM RESEARCH REPORT RJ 10186.

P. Makagonov, et al. (2004). 'Clustering Abstracts instead of Full Texts'. *In: Text, Speech, Dialog, LNAI N 3206, Springer* **2004**:129–135.

C. Manning, et al. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

P. Pantel & D. Lin (2002). 'Document clustering with committees'. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '02, pp. 199–206, New York, NY, USA. ACM.

D. Pinto, et al. (2007). 'Clustering Narrow-Domain Short Texts by Using the Kullback-Leibler Distance'. In A. F. Gelbukh (ed.), *Proceedings of the 8th International Conference on Computational Linguistics and Intelligent Text Processing*, vol. 4394 of *CICLing'07*, pp. 611–622, Berlin, Heidelberg. Springer.

D. Pinto, et al. (2006). 'Clustering abstracts of scientific texts using the transition point technique'. In *Proceedings of the 7th international conference on Computational Linguistics and Intelligent Text Processing*, CICLing'06, pp. 536–546, Berlin, Heidelberg. Springer-Verlag.

D. Pinto & P. Rosso (2006). 'KnCr: A short-text narrow-domain sub-corpus of medline'. In *Proc. of the TLH 2006 Conference*, Advances in Computer Science, pp. 266–269.

C. E. Shannon (2001). 'A mathematical theory of communication'. *SIGMOBILE Mob. Comput. Commun. Rev.* **5**(1):3–55.

P. Shrestha, et al. (2012). 'Clustering Short Text and its Evaluation'. In *Proceedings of the 13th International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing'12, pp. 169–180, Berlin, Heidelberg. Springer.

M. Steinbach, et al. (2000). 'A comparison of document clustering techniques'. In *In KDD Workshop on Text Mining*.

O. Zamir & O. Etzioni (1998). 'Web document clustering: a feasibility demonstration'. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pp. 46–54, New York, NY, USA. ACM.

O. Zamir & O. Etzioni (1999). 'Grouper: a dynamic clustering interface to Web search results'. In *Proceedings of the eighth international conference on World Wide Web*, WWW '99, pp. 1361–1374, New York, NY, USA. Elsevier North-Holland, Inc.