CREST: Cluster-based Representation Enrichment for Short Text Classification*

Zichao Dai[1], Aixin Sun[2], and Xu-Ying Liu[1]

[1] MOE Key Laboratory of Computer Network and Information Integration, School of Computer Science and Engineering, Southeast University, Nanjing, China,
daixiaodai.geek@gmail.com, liuxy@seu.edu.cn
[2] School of Computer Engineering, Nanyang Technological University, Singapore
axsun@ntu.edu.sg

Abstract. Text classification has gained research interests for decades. Many techniques have been developed and have demonstrated very good classification accuracies in various applications. Recently, the popularity of social platforms has changed the way we access (and contribute) information. Particularly, short messages, comments, and status updates, are now becoming a large portion of the online text data. The shortness, and more importantly, the sparsity, of the short text data call for a revisit of text classification techniques developed for well-written documents such as news articles. In this paper, we propose a cluster-based representation enrichment method, namely CREST, to deal with the shortness and sparsity of short text. More specifically, we propose to enrich a short text representation by incorporating a vector of topical relevances in addition to the commonly adopted *tf-idf* representation. The topics are derived from the knowledge embedded in the short text collection of interest by using hierarchical clustering algorithm with purity control. Our experiments show that the enriched representation significantly improves the accuracy of short text classification. The experiments were conducted on a benchmark dataset consisting of Web snippets using Support Vector Machines (SVM) as the classifier.

Keywords: Short text classification, Representation enrichment, Clustering

1 Introduction

The prevalence of Internet-enabled devices (e.g., laptops, tablets, and mobile phones) and the increasing popularity of social platforms are changing the way we consume and produce information online. A large portion of the data accessible online is user-generated content in various forms, such as status updates, micro-blog posts, comments, and short product reviews. In other words, much

* This work was partially done while the first author was visiting School of Computer Engineering, Nanyang Technological University, Singapore, supported by MINDEF-NTUDIRP/2010/03, Singapore

user-generated textual content is in the form of *short text*. The unique characteristics (e.g., shortness, noisiness, and sparsity) distinguish short text from the well written documents such as news articles and most Web pages. These unique characteristics call for a revisit of the techniques developed for text analysis and understanding, including text classification.

Text classification refers to the task of automatically assigning a textual document one or more predefined categories. It has been heavily studied for decades and many techniques have been proposed and have demonstrated good classification accuracies in various application domains [13, 16]. Nevertheless, most text classification techniques take advantage of the information redundancy naturally contained in the well-written documents (or long documents in contrast to short text). When facing with short text, the shortness, noisiness, and sparsity, adversely affect the classifiers from achieving good classification accuracies. To improve short text classification accuracy has since attracted significant attention from both the industries and academia.

To deal with the shortness and sparsity, most solutions proposed for short text classification aim to enrich short text representation by bringing in additional semantics. The additional semantics could be from the short text data collection itself (e.g., named entities, phrases) [7] or be derived from a much larger external knowledge base like Wikipedia and WordNet [4, 7, 10]. The former requires shallow Natural Language Processing (NLP) techniques while the later requires a much larger and "appropriate" dataset. Very recently, instead of enriching short text representation, another approach known as search-and-vote is proposed to improve short text classification [15]. The main idea is to mimic human judging processing by identifying a few topical representative keywords from each short text and use the identified topical keywords as queries to search for similar short texts from the labeled collection. Very much similar to k-nearest-neighbor classifier, the category label of the short text for classification is voted by using the search results. Note that, the aforementioned different approaches deal with the shortness and sparsity of short text from very different perspectives and are mostly orthogonal to each other. In other words, on the one hand, these different approaches could be combined to potentially achieve much better classification accuracies than any of the approaches alone; on the other hand, this calls for further research to improve each individual researches.

In this paper, we focus on improving short text classification accuracy by enriching the text representation, by not only using its raw words (e.g., bag-of-words) but also topical representations. Our approach naturally falls under the *representation enrichment* approach. However, our approach is different from the earlier works in representation enrichment because of two reasons. First, we do not use shallow NLP techniques to extract phrases or any specific patterns because most short texts are noisy preventing many existing NLP toolkits from achieving good accuracy. Second, we do not use external knowledge base like Wikipedia because some of the short text data collection might be from very specific or niche areas where it is hard to find an "appropriate" and large dataset. In other words, we consider that if we can discover internally useful knowledge

solely from the training dataset when an "appropriate" large external dataset is not available. More specifically, we propose a generic method named CREST to first discover "high-quality" topic clusters from the training data by grouping similar (but not necessary from the same category) training examples together to form clusters. Each short text instance is then represented using the topical similarities between the short text and the topic clusters in addition to its words feature vector. The main advantages of CREST include the following:

- *Low-cost in knowledge acquisition.* As we mentioned above, CREST does not rely on any external knowledge source. It mines topic clusters solely from the training examples.
- *Reduction in data sparsity.* The topic clusters discovered from the training data define a new feature space that each short text instance can be mapped to. In this new space, the dimensionality is the number of "high-quality" clusters discovered from the training data, which is much smaller than the number of words in the bag-of-words representation.
- *Easy in implementation and combination.* The CREST framework is easy to implement and can be easily combined with other approaches dealing with short text classification.

The rest of the paper is organized as follows. Section 2 surveys the related work in short text classification. Section 3 describes the CREST method. Section 4 reports the experimental results and Section 5 concludes this paper.

2 Related Work

Short text processing has attracted research interests for a long time, particularly in the meta-search applications to group similar search results into meaningful topic clusters. Nevertheless, the key research problem in search snippet clustering is to automatically generate meaningful cluster labels [3]. Another direction of research in short text processing is to evaluate the similarity of a pair of short texts using external knowledge obtained from search engines [11, 17]. In [1], semantic similarity between words is obtained by leveraging page counts and text snippets returned by search engine.

For short text classification, the work on query classification is more related as each query can be treated as a piece of short text. In [14], the authors use titles and snippets to expand the Web queries and achieve better classification accuracy on query classification task compared to using the queries alone. However, the efficiency and the reliability issues of using search engine limit the employment of search-based method, especially when the set of short text under consideration is large. To address these issues, researchers turn to utilize explicit taxonomy/concepts or implicit topics from external knowledge source. These corpora (e.g., Wikipedia, Open Directory) have rich predefined taxonomy and human labelers assign thousands of Web pages to each node in the taxonomy. Such information can greatly enrich the short text. These research

has shown positive improvement though they only used the man-made categories and concepts in those repositories. Wikipedia is used in [6] to build a concept thesaurus to enhance traditional content similarity measurement. Similarly, in [8], the authors use Wikipedia concept and category information to enrich document representation to address semantic information loss caused by bag-of-words representation. A weighted vector of Wikipedia-based concepts is also used for relatedness estimation of short text in [5]. However, lack of adaptability is one possible shortcoming of using predefined taxonomy in the above ways because the taxonomy may not be proper for certain classification tasks. To overcome this shortcoming, the authors in [10] derived latent topics from a set of documents from Wikipedia and then used the topics as additional features to expand the short text. The idea is further extended in [4], to explore the possibility of building classifier by learning topics at multi-granularity levels. Experiments show that the methods above using the discovered latent topics achieve the state-of-the-art performance. In summary, these methods try to *enrich* the representation of a short text using additional semantics from an external collection of documents. However, in some specific domain (e.g., military or healthcare) it might be difficult to get such high quality external corpora due to privacy or confidentiality reasons.

Most germane to this work is the approach proposed in [2] which applies probabilistic latent semantic analysis (pLSA) on text collection and enriches document representation using the latent factors identified. However, pLSA becomes less reliable in identifying latent topics when applying to very short texts, due to the difficulties of sparsity and shortness. In this paper, we use a different approach to find the topics embedded in the short text collection by clustering the documents in the collection.

3 The CREST Method

Most existing topic-based methods rely on large external sources (such as Wikipedia or search engines). However, there exist tough situations in some specific domains (e.g., military or healthcare) where lack of reliable high quality external knowledge repositories. This limits the employment of these methods. In this scenario, the only available resource is the collection of labeled short texts. How to exploit the limited collection at utmost becomes crucial in short text classification.

The good performance of topic-based methods shows latent topics can be very useful to short text classification. Since the document collection is the only available resource in our scenario, we derive latent topics from the document collection itself by exploiting clustering. Then, we use the topic clusters to enrich the representation for short texts. The general process of CREST (***C**luster-based **R**epresentation **E**nrichment for **S**hort **T**ext Classification*) method is illustrated in Figure 1.

Suppose a document collection $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$ has n short text documents, where $\mathbf{x}$ is pre-processed short text document and $\mathbf{x} \in X = R^d$. In this paper, we

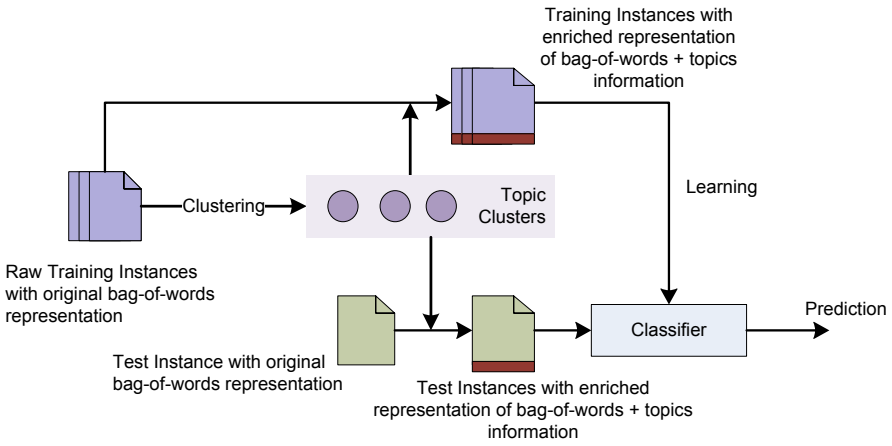

Fig. 1. Procedure of CREST

adopt tf-idf [12] representation. And y is category label, $y \in Y = \{1, 2, \ldots, k\}$. L is a learning algorithm, training a classifier $h : X \to Y$.

3.1 Topic Clusters Generation

Clustering is good at finding knowledge structure inside data. CREST exploits clustering to find topics. Intuitively, for each high-level category, for example "Business", it has its a few sub-topics, such as "Accounting","Finance". The sub-topics could have different topical words, especially when the text is very short. In other word, each cluster contains terms and concepts mainly in one sub-topic which we could take advantage of to enrich short texts and reduce their sparsity.

However, due to the sparsity of short text, the similarity of a pair of short text instances may not be reliable enough when it is reflected by distance in a clustering method. Thus, the resulting clusters may not be qualified as topics. The challenge here is to select "high-quality" clusters as *topic clusters*. Note that, even though there exist many clustering methods, not all clusters generated by a clustering method is useful. For instance, a cluster containing very few documents (say, only one) or a large number of documents from many different categories are not useful clusters. The clusters with very few documents fail to cover enough concepts in a sub-topic while the clusters containing too many documents are not topically specific.

In summary, CREST selects "high-quality" clusters as topic clusters with two criteria: (i) *high support*, i.e., the number of documents in a cluster is large; and (ii) *high purity*, i.e., the percentage of dominant category of the short texts in a cluster is high.

Suppose a cluster Q contains a set of short text instances, $Q = \{(\mathbf{x}_i, y_i)\}_{i=1}^{q}$, then the *support* of Q is the number of instances in it, i.e.,

$$support(Q) = |Q|. \tag{1}$$

And the *purity* of Q is the percentage of dominant category of the short texts in it, which is defined as:

$$purity(Q) = \frac{\max_y \sum_{\mathbf{x}_i \in Q} I(y_i = y)}{|Q|}, \tag{2}$$

where, $I(x)$ is indicator function, $I(x) = 1$ if $x = 1$ and 0 otherwise.

More specifically, CREST uses a clustering method, such as $EfficientHAC$ [9], to group short texts into clusters. When a cluster's purity is low, it does not represent a sub-topic even if its support is high. Therefore, we select the clusters whose purity values are larger than a pre-defined threshold. We then get a set of candidate-clusters $\mathcal{C} = \{C_1, C_2, \ldots, C_{|C|}\}$. To select clusters with high "support" and "purity", we assign a weight to each cluster in $\mathcal{C}$ indicating the quality to be a topic cluster of each cluster. Let w_i be C_i's weight:

$$w_i = support(C_i) \times purity(C_i), \tag{3}$$

Then the top N clusters with the highest weights are selected as topic clusters $\mathcal{T}$, which are rich of representative terms or concepts in particular sub-topics, and are later used to enrich short text's representation.

In most cases, the weights of candidate-clusters in $\mathcal{C}$ are influenced more by their support values. It is reasonable, since the purity values of candidate-clusters in $\mathcal{C}$ are all larger than a purity threshold, which is often a relatively high value to assure all clusters in $\mathcal{C}$ be of high purity.

3.2 Representation Enrichment Using Topic Clusters

CREST enriches representation of short text by combining a short text instance's original feature vector, i.e., *tf-idf* vector, and the additional information from the topic clusters. To extract knowledge from topic clusters, a good choice is to use the similarity between a short text instance $\mathbf{x}$ and each of the topic cluster T_i in $\mathcal{T}$, which contains the common terms or concepts of a sub-topic. So the similarity between a short text instance $\mathbf{x}$ and a topic cluster T_i reflects how likely the common terms or concepts of the sub-topic represented by T_i would appear in the text if the "short" text were longer.

For example, a short text (taken from the benchmark dataset used in our experiments) is "manufacture manufacturer directory directory china taiwan products manufacturers directory- taiwan china products manufacturer directory exporter directory supplier directory suppliers business". And there are two topic clusters: cluster 1 represents a sub-topic of "business" category, and cluster 2 represents a sub-topic of "health" category. Cluster 1 contains concepts like "relation", "produce", "machine", and so on. Cluster 2 contains concepts

Algorithm 1: The CREST Algorithm

Input : Training set $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, Learning algorithm L to train a classifier $h : X \to Y$, Purity threshold $p \in [0, 1]$, Hierarchical clustering algorithm $EfficientHAC$, The number of topic clusters N

Training phrase:
%Generate topic clusters
Use $EfficientHAC$ algorithm to generate raw cluster set R
Candidate-cluster set $C = \{r | r \in R \wedge purity(r) \geq p\}$
for $i = 1$ *to* $|C|$ **do**
    $w_i = support(C_i) \times purity(C_i)$ %cluster weight
Select top N clusters from C with highest weights into topic cluster set T
%Enrich representation
for $i = 1$ *to* n **do**
    for $j = 1$ *to* N **do**
        Calculate similarity $sim(\mathbf{x}_i, T_j)$ according to Eq.4
    $\mathbf{x}'_i = (\mathbf{x}_i, sim_1(\mathbf{x}), \ldots, sim_N(\mathbf{x}))$
New data set $D' = \{(\mathbf{x}'_i, y_i)\}_{i=1}^{n}$
Output: A classifier $h = L(D')$

Test phrase: for a test instance $\mathbf{x}$
for $j = 1$ *to* N **do**
    Calculate the similarity $sim(\mathbf{x}, T_j)$ according to Eq.4
$\mathbf{x}' = (\mathbf{x}, sim_1(\mathbf{x}), \ldots, sim_N(\mathbf{x}))$
Prediction $\hat{y} = h(\mathbf{x}')$

like "symptoms", "treatment", "virus", "diet". Obviously, the short text is more similar to cluster 1. And if it were longer, the word "produce", "machine" have a larger chance to appear in the text.

Define the similarity between a short text $\mathbf{x}$ and a topic cluster T as:

$$sim(\mathbf{x}, T) = \frac{\mathbf{x} \cdot T}{\|\mathbf{x}\| \, \|T\|} \tag{4}$$

In $sim(\mathbf{x}, T)$, the dot product is used to compute the initial similarity value between short text and topic cluster. Since the lengths of topic clusters are varying, to reduce their influence, we normalize the lengths of both short text and topic cluster to get final similarity, i.e., cosine similarity.

Let $\mathbf{s} = (sim(\mathbf{x}, T_1), \ldots, sim(\mathbf{x}, T_N))$ be the similarity vector, then the enriched representation of $\mathbf{x}$ is:

$$\mathbf{x}' = (\mathbf{x}, \mathbf{s}) \tag{5}$$

The pseudo code of CREST is shown in Algorithm 1, in which the clustering algorithm $EfficientHAC$ can be replaced by another hierarchical clustering algorithm.

Table 1. Basic Statistics of Experiment Dataset

Category	# training instances	# test instances
Business	1200	300
Computer	1200	300
Culture	1880	330
Education	2360	300
Engineering	220	150
Health	880	300
Politics	1200	300
Sports	1120	300
Total	10060	2280

4 Experiments

Since the problem setting of this paper is that there is no external knowledge sources, it is inappropriate to compare CREST with methods relying on some external knowledge source. We compare CREST with original representation of short text (i.e., *tf-idf* vectors, denoted by "Raw"). In CREST, the clustering strategies EfficientHAC [9] is single-link, and the purity threshold is set to be 0.9. We test different values $10, 30, 50, 70, 100, 120$ for the number of topic clusters N. We use SVM as learning algorithm for both CREST and Raw representations using SVM^{light} with default parameter settings[3]. We run experiments on the benchmark dataset of search snippets collected by [10] and the statistics of the dataset is shown in Table 1.

For each parameter settings, we run the experiment for 20 times, then compute the average value. We record the F_1 measurement. Table 2 shows the F_1 results, where the tabular in boldface means that CREST's result is significantly better than Raw by pairwise t-test with significance level at 0.95, "*best*" is the best F_1 value among CREST with different N's, "*avg.*" is the average F_1 value over all categories. The results are plotted in Fig. 2.

These results show that CREST improves the classification performance considerably compared to Raw in every category with almost all parameter settings. Especially, in some specific categories such as "business" and "politics", the improvement is as large as 17.13% and 19.51%, respectively. The results show that CREST method utilizing topic clusters extracted from limited training examples to enrich short texts is a useful way to overcome the shortness and sparsity of short texts. From Fig. 2 we can see that CREST is very robust to the change of N, the number of topic clusters. Even when N is very small, CREST improves the performance largely in almost all categories. This shows the power of the enriched representation by exploring topic clusters. The only exception is that in category "engineering", only when the number of topic clusters N is greater

[3] `http://svmlight.joachims.org/`

Table 2. F_1 Results (%)

Method	busin.	compu.	cultu.	educa.	engin.	healt.	polit.	sport.	*avg.*
Raw	50.23	67.64	66.41	67.49	29.37	59.69	33.32	78.24	56.55
CREST $N = 10$	**58.79**	**68.92**	**68.01**	**69.57**	25.58	**63.78**	**37.09**	**80.23**	**59.00**
CREST $N = 30$	**55.87**	**69.60**	**68.38**	**68.78**	15.95	**62.49**	**38.51**	**80.23**	**57.48**
CREST $N = 50$	**53.97**	**68.63**	**66.91**	**69.48**	25.58	**61.20**	**39.68**	**80.15**	**58.20**
CREST $N = 70$	**56.37**	**70.54**	**66.91**	**69.48**	**31.11**	**60.16**	**39.78**	**80.31**	**59.33**
CREST $N = 100$	**55.65**	**69.72**	**67.15**	**70.48**	**33.14**	**60.47**	**39.36**	**78.89**	**59.36**
CREST $N = 120$	**54.91**	**68.90**	**68.36**	**69.62**	**33.14**	**60.9**	**38.95**	**78.65**	**59.18**
best	**58.79**	**70.54**	**68.38**	**70.48**	**33.14**	**63.78**	**39.78**	**80.31**	

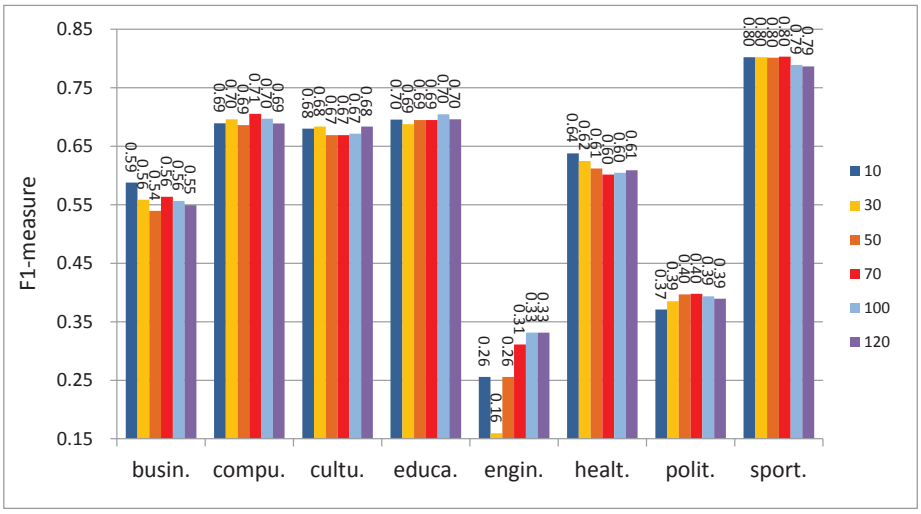

Fig. 2. Comparison among Different Embedded Number of Topic Clusters

than 70 can CREST improves the performance. One possible reason is that "engineering" category has fewer instances than other categories but covers relatively a large topic. The instances in this category are harder to be gathered together by a clustering method. CREST manages to improve the performance of this category by increasing the number of topic clusters in N.

To further study how parameters will affect CREST, we record the F_1 results of CREST with different clustering strategies (single-link or complete-link) and different purity thresholds (0.85, 0.90, 0.95) while fixing $N = 70$. The results are shown in Fig. 3. Generally speaking, CREST is very robust to the change of these parameters when purity threshold is above 0.90. Since the topic clusters with higher purity would be more topic-specific, higher purity threshold leads to more helpful critical terms or concepts. On the other hand, clustering strategy doesn't affect the performance significantly. CREST is slightly more sensitive to

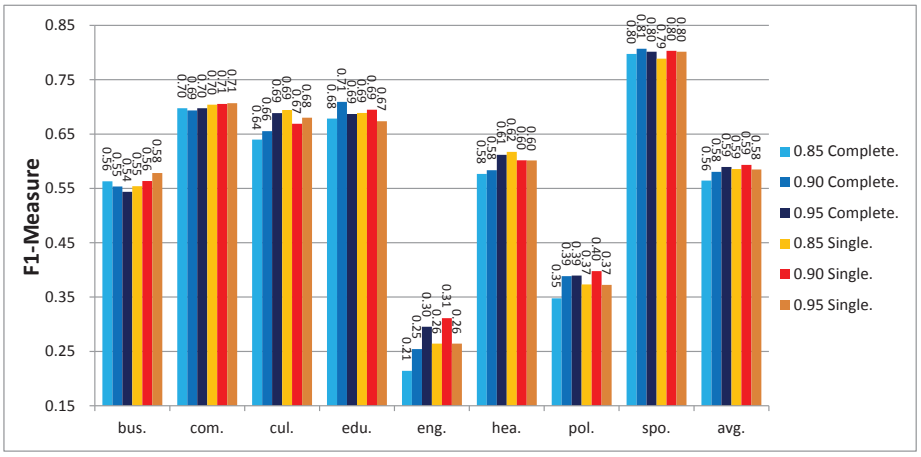

Fig. 3. Comparison among Different Clustering Strategies and Purity Thresholds

purity threshold when using the single-link strategy than using the complete-link strategy.

The above experimental results lead to the following conclusions: (1) CREST can greatly improve the short text classification performance in term of F_1 measure by enriching the representation with topic information; and (2) CREST is robust to parameter settings.

5 Conclusion

Short text classification problem attracts much attention from information retrieval field recently. In order to handle its shortness and sparsity, various approaches have been proposed to enrich short text to get more features like latent topics or other information. However, most of them rely on large external knowledge sources more or less. These methods solve the problem to some extent, but still leave large space for improvement, especially under the hard condition that no external knowledge source can be acquired. We proposed CREST method to handle the short text classification in such tough situation. CREST generates "high-quality" clusters as topic clusters from training data by exploiting clustering method, and then uses the topic information to extend representation for short text. The experimental results showed that compared to the original representation, CREST can significantly improves the classification performance.

Though we see positive improvement brought by CREST, there are still room for further consideration to boost the performance. For example, we can try to combine CREST with other methods for short text classification, such as methods relying on external knowledge sources. And organizing "high-quality" clusters in

multi-granularity way to investigate whether it can further improve CREST is another interesting problem worth exploring.

6 Acknowledgement

This work was supported by NSFC (No. 61105046), SRFDP (Specialized Research Fund for the Doctoral Program of Higher Education, by Ministry of Education, No. 20110092120029), and Open Foundation of National Key Laboratory for Novel Software Technology of China (KFKT2011B01). The work of the second author was supported by MINDEF-NTU-DIRP/2010/03, Singapore.

References

1. D. Bollegala, Y. Matsuo, and M. Ishizuka. Measuring semantic similarity between words using web search engines. In *Proceedings of the 16th International Conference on World Wide Web*, pages 757–766, New York, 2007.
2. L. Cai and T. Hofmann. Text categorization by boosting automatically extracted concepts. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, pages 182–189, New York, 2003.
3. C. Carpineto, S. Osiński, G. Romano, and D. Weiss. A survey of web clustering engines. *ACM Computing Surveys (CSUR)*, 41(3):17:1–17:38, 2009.
4. M. Chen, X. Jin, and D. Shen. Short text classification improved by learning multi-granularity topics. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 1776–1781, 2011.
5. E. Gabrilovich and S. Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, pages 1606–1611, San Francisco, CA, 2007.
6. J. Hu, L. Fang, Y. Cao, H.-J. Zeng, H. Li, Q. Yang, and Z. Chen. Enhancing text clustering by leveraging wikipedia semantics. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 179–186, New York, NY, 2008.
7. X. Hu, N. Sun, C. Zhang, and T.-S. Chua. Exploiting internal and external semantics for the clustering of short texts using world knowledge. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pages 919–928, 2009.
8. X. Hu, X. Zhang, C. Lu, E. K. Park, and X. Zhou. Exploiting wikipedia as external knowledge for document clustering. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 389–396, New York, NY, 2009.
9. C. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press Cambridge, 2008.
10. X.-H. Phan, L.-M. Nguyen, and S. 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*, pages 91–100, New York, NY, 2008.

11. M. Sahami and T. D. Heilman. A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of the 15th International Conference on World Wide Web*, pages 377–386, New York, NY, 2006.
12. G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.
13. D. Shen, Z. Chen, Q. Yang, H. Zeng, B. Zhang, Y. Lu, and W. Ma. Web-page classification through summarization. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 242–249, 2004.
14. D. Shen, R. Pan, J.-T. Sun, J. J. Pan, K. Wu, J. Yin, and Q. Yang. Query enrichment for web-query classification. *ACM Transactions on Information Systems*, 24(3):320–352, 2006.
15. A. Sun. Short text classification using very few words. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 1145–1146, New York, NY, 2012.
16. J. Tang, X. Wang, H. Gao, X. Hu, and H. Liu. Enriching short text representation in microblog for clustering. *Frontiers of Computer Science in China*, 6(1):88–101, 2012.
17. W.-T. Yih and C. Meek. Improving similarity measures for short segments of text. In *Proceedings of the 22nd National Conference on Artificial Intelligence*, pages 1489–1494, 2007.