Topic Detection from Short Text: A Term-based Consensus Clustering Method

Hao Lin[1], Bo Sun[2], Junjie Wu[1], Haitao Xiong[1*]
[1]School of Economics and Management, Beihang University, Beijing, China
[2]National Computer Network Emergency Response Technical Team/Coordination Center of China, Beijing, China
[*]Corresponding Author: xionghaitao@btbu.edu.cn

***Abstract*—The process of Topic Detection from Short Text Systems (SMS) is to extract distinct topics hidden inside short text collections, such as Twitter, Weibo, and instant messages. With the recent emergence of large volume user generated content collections enabled by online social media, topic detection from SMS becomes a challenging yet promising means for online public opinion analysis. In available literature, many forms and methods of topic detection have been proposed, but obtaining meaningful and coherent data is still difficult to reliably obtain for the extreme sparsity brought by SMS. To this end, we developed a Term-based Consensus Clustering topic detection (TCC) framework to provide an unsupervised methodology for finding distinct topics from within SMS collections. Specifically, we adopt a consensus clustering technique called K-means-based Consensus Clustering to handle SMS clustering, due to its low computational complexity and robust clustering performance. To further enrich the features of the information of the sparse SMS data, we conduct term clustering in the highly dense term space instead of the conventionally targeted sparse document space. To be more specific, we first use a feature space transfer technique to represent short text collections as a pseudo-document matrix, where rows, namely instances, correspond to terms and columns, namely features, correspond to adjacent terms. Basic partitions are generated from the pseudo-document matrix for term clustering and consensus clustering is followed to obtain the final term clustering result. Finally, a document classification process is adopted and a document is assigned to a cluster, where most terms occurred. Extensive experiments on real-world data sets demonstrate that TCC is comparable to several widely used methods in terms of topic detection quality. Particularly, we demonstrate that TCC obtains best clustering performance when observing a large number of the predefined topics across short text collections.**

Keywords*—*Topic Detection; SMS; Consensus Clustering; Feature Space Transfer

I. Introduction

In online social media, users usually post short text. For example, twitter limits the length of each tweet to 140 characters. Therefore, to develop a robust technique to handle the large volume of short texts has become an important goal. In this paper, we consider the problem of topic detection from SMS, which is to extract distinct topics hidden inside short text collections. Since we are not given a list of topics, we treat the problem of document clustering as the core of the traditional topic detection task. Most of the early works on document clustering are traditional clustering based approaches [1], which can be further divided into two categories, namely hierarchical clustering and iterative clustering methods, respectively. When using document clustering methods, a text document is usually represented as a data point in a high dimensional document space, each dimension corresponding to a single term. Moreover, the extreme sparsity native to the SMS format makes for a challenging yet promising task for document clustering. Recently, although a variety of efforts have been made on document clustering for SMS [2], the problem of high dimensionality and sparsity still exists.

In light of this, we propose an unsupervised method of incorporating consensus clustering and feature space transfer techniques to facilitate basic clustering results and conduct dimensionality reduction, simultaneously. Consensus clustering, also known as cluster ensembles or clustering aggregation, aims to find single partitions of data from multiple existing basic partitions [3]. It is widely recognized that consensus clustering can help to generate robust clustering results and handle noise. For further enhancing clustering performance on short text, we apply a feature space transfer (FST) technique to reduce the high dimensionality of the text corpus. We can revisit the most common document representation model vector space model (VSM). The idea of the VSM is to represent each document in a collection as a single point in a space (a vector in a vector space). The main idea of our proposed FST is that we suppose that terms can be partitioned into term clusters according to their semantic similarity in a term space, which is much denser than in a document space. Using these term clusters, the original document is assign with a cluster where most of terms in term space appear.

Our method is described as follows, we first apply a feature space transfer technique to convert documents to a much lower dimensional term space. Then consensus clustering technique is introduced. We generate basic partitions using K-means algorithm. During the consensus part, we apply a consensus clustering method called K-means-based Consensus Clustering to equivalently transfer the consensus clustering problem into a K-means like optimization problem for high efficiency. Extensive experiments on a real-world data set demonstrate that TCC is comparable to several widely used methods in terms of topic detection quality. Particularly, we demonstrate that TCC obtains best clustering performance when observing a large number of the predefined topics across short text collections.

Overview. The remainder of this paper is organized as follows. Section II gives the details of our proposed method TCC. In Section III, we provide a brief review of related works and our innovations. In Section IV, we report the evaluation results based on a real-world data set of Weibo. Finally, we conclude the paper in Section V.

978-1-5090-2842-9/16/$31.00 ©2016 IE

II. Term-based Consensus Clustering Topic Detection

In this section, we give a comprehensive view of the proposed Term-based Consensus Clustering method (TCC) for topic detection from short text collections. First, we will give the motivations of our proposed method. Then, we will briefly introduce basic concepts of consensus clustering, especially the K-means-based Consensus Clustering. The feature space transfer process is further introduced to enrich the feature space of consensus clustering. Finally, we present the framework for Term-based Consensus Clustering Topic Detection.

A. Motivations

Our work in this paper focuses on topic detection from short text collections. Specifically, we aim to answer these questions: 1) Can trending topics be detected across collections in spite of the limited length of SMS? 2) Which of these documents belongs to specific topics?

To answer these questions, we first need an unsupervised method for clustering documents with semantically similar terms and finding distinct topics from these documents, with the assumption that each document contains a single topic. Thus, we transfer the topic detection problem to a document clustering problem. In addition, we need to build a feature-enhancing model suitable for short text document clustering due to the high dimensionality and sparsity of the SMS collections.

To this end, our main task is refined as designing a feature-enhancing document clustering method for topic detection from SMS entries. Along this line, two key problems are addressed in the model. The first one is how to achieve robust clustering performance across large volumes of SMS collections, which will be discussed in detail in Sect. II-B. The second problem is how to solve the high dimensionality and sparsity issues inherent to SMS entries in document clustering tasks, which will be discussed in detail in Sect. II-C.

Here in this subsection, we give motivations of our proposed model. To address the first problem, ensemble clustering [3] (also known as consensus clustering or clustering aggregation) is applied to utilize multiple existing basic partitions to find a single partition of data, which have been recognized as state-of-the-art techniques for generating robust clustering results. Among multiple consensus clustering approaches, the K-means-based Consensus Clustering (KCC) is undoubtedly a competitive candidate in terms of high efficiency and robustness.

To handle the second problem, let us first consider the data representation used by clustering algorithms, which are based on the conventional (real-value) vector space model [4]. In this model, each document is represented in a high-dimensional space, in which each dimension of the space corresponds to a term in the vocabulary set. To help alleviate the high dimensionality and sparsity, it is intuitive to develop a term-based clustering method in order to directly transfer the high-dimensional sparse document space to low-dimensional dense term space. Thus, a word co-occurrence network proposed by Zuo [5] is applied to generate pseudo-documents, which models the rich word co-occurrence patterns in documents.

TABLE I: Contingency Matrix

		π'				
		C'_1	C'_2	$\cdots$	$C'_{K'}$	$\sum$
π	C_1	n_{11}	n_{12}	$\cdots$	$n_{1K'}$	n_{1+}
	C_2	n_{21}	n_{22}	$\cdots$	$n_{2K'}$	n_{2+}
	$\cdot$	$\cdot$	$\cdot$	$\cdots$	$\cdot$	$\cdot$
	C_K	n_{K1}	n_{K2}	$\cdots$	$n_{KK'}$	n_{K+}
	$\sum$	n_{+1}	n_{+2}	$\cdots$	$n_{+K'}$	n

And then, KCC is carried out on pseudo-documents to partition terms into term clusters, which helps convert the original document space into the transferred term space. After the space transferred, each term is assigned with a specific cluster label. Finally, the document is classified with the cluster in which most of terms appear.

B. Consensus Clustering & KCC

In this subsection, we introduce basic concepts of consensus clustering and give details about KCC techniques.

Let $\mathcal{X} = \{x_1, x_2, \cdots, x_n\}$ be a set of n data points belonging to K crisp clusters, denoted as $\mathcal{C} = \{C_1, \cdots, C_k\}$, where $C_k \bigcap C_{k'} = \emptyset$, $\forall k \neq k'$, and $\bigcup_{k=1}^{K} C_k = \mathcal{X}$. Given r basic partitions represented as $\Pi = \{\pi_1, \pi_2, \cdots, \pi_r\}$, each of which partitions $\mathcal{X}$ into K_i clusters, and maps each data point to a cluster label ranged from 1 to K_i. The goal of ensemble cluster is to find an optimal consensus partition π based on the input basic partitions Π. It is, in essence, a combinatorial optimization problem. The typical consensus clustering problem has the following formulation:

$$\max_{\pi} \sum_{i=1}^{r} w_i U(\pi, \pi_i),$$

where U is a utility function that measures the similarity between two partitions, and $w_i \in [0, 1]$ is the weight for each partition, with $\sum_{i=1}^{r} w_i = 1$.

The computation issue is a critical concern for consensus clustering and in this paper, a unified framework for solving KCC proposed by Wu [6] is applied to take a KCC utility function to define the consensus function and rely on the K-means heuristic to find the consensus partition. We will revisit KCC with the use of the contigency matrix.

A contigency matrix in Table I is employed to compare two partitions π and π', which are the optimal consensus partition namely the ground truth and the clustering result returned by some algorithm, respectively. Let n_{ij} denote the number of data objects belonging both cluster C'_j in π' and cluster C_i in π, $n_{i+} = \sum_{j=1}^{K'} n_{ij}$, and $n_{+j} = \sum_{i=1}^{K} n_{ij}$, $1 \leq i \leq K, 1 \leq j \leq K'$. Based on the contingency table, the well-known Category Utility Function can be computed as follows.

$$U_c(\pi, \pi') = \sum_{i=1}^{K} p_{i+} \sum_{j=1}^{K'} (\frac{p_{ij}}{p_{i+}})^2 - \sum_{j=1}^{K'} (p_{+j})^2,$$

where $p_{ij} = n_{ij}/n$ is the joint probability of one instance belonging to both C_i in π and C_j in π', $p_{i+} = n_{i+}/n$ is the portion of C_i in π and $p_{+j} = n_{+j}/n$ is the portion of C_j in π'. Since almost every clustering algorithm needs parameters (such as the predefined cluster number for K-means), Random Parameter Selection (RPS) strategy is employed to generate basic partitions. The choice of the KCC utility function is

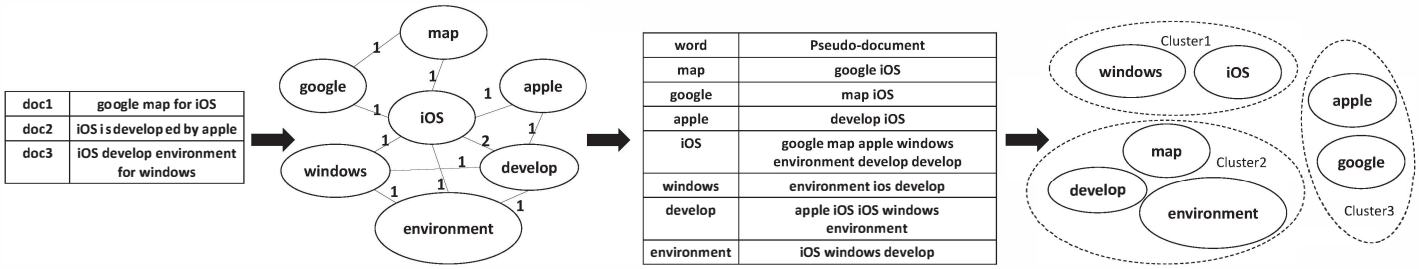

Fig. 1: An Example of Feature Space Transfer.

critical for the success of a consensus clustering, of which we will not go into the details for the sake of brevity.

C. Feature Space Transfer

In this subsection, we introduce the Feature Space Transfer (FST) technique employed to convert the high-dimensional sparse document space to the much lower-dimensional term space.

As illustrated in Sect. II-A, short text clustering incur the high dimensionality and sparsity, which motivates us to adopt a feature space transfer technique for enhanced consensus clustering. The main idea of FST is that we first generate pseudo-documents in a term space, then conduct pseudo-document clustering with KCC and finally obtain a cluster label for each term, which is used for the final document clustering. We give details of FST as follows.

In order to incorporate the rich word co-occurrence patterns in documents, we first employ a word co-occurrence network (also known as word network) generation strategy proposed by Yuan in [5]. This is motivated by the fact that the original word-by-document space is extremely sparse while the transferred term-to-term space is still rather dense. The following gives the detail of word network generation heuristic:

1) We begin by filtering out single words and stopping words, and then a sliding window is moved to scan each document. As the window scans word by word through the document, any two distinct words that appear in the same window would be regarded as co-occurred with each other.

2) We then accumulate times that two words co-occurred and defined as the weight of the corresponding edge between these two words.

3) We further count words that co-occur near each other more times than those that co-occur far from each other. In this manner, we can encourage the model to put words that co-occur near each other into the same cluster.

With the word network obtained from the above heuristic, we represent word network back to a pseudo-document set. As illustrated in Fig. 1, each word in the network can be treated as a pseudo-document with content constituded by the list of its adjacent words.

After that, we apply KCC to partition terms in the word network into term clusters using pseudo-document representation. Assume that K_p is the predefined cluster number of pseudo-document clustering and we will discuss K_p in Sect. IV. Finally, we obtain a cluster label for each term in the text corpus, ranges from 1 to K_p.

Consequently, after this FST procedure, the feature space of the original documents are reduced to the term space, while the rich word co-occurrence patterns are also incorporated in the word network generation process.

D. Framework for TCC

Here in this subsection, we revisit the framework for Term-based Consensus Clustering Topic Detection from SMS. The framework of our approach is demonstrated in Fig. 2, which consists of four main phases:

Phase i aims to transfer the original document space to the much lower-dimensional yet dense term space. Here, word network generation stategy is applied in the feature space transfer procedure.

Phase ii generates basic partitions (BPs) on the transferred term space using Random Parameter Selection (RPS). Spherical K-means implemented in Cluto with cosine similarity as distance measure is employed for this purpose.

Phase iii incorporate cluster knowledge from BPs. KCC with the basic partition matrix as input is conducted to obtain the final consensus partition. Each term is thus assgined with a specific cluster label.

Phase iv classify the original document using the term cluster information. Here, we classify the document into a cluster to which most of terms belong.

It is worth noting that the framework is suitable for short text document clustering, where high dimensionality and sparsity are two key challenges. By feature space transferring, term space is generated which helps to alleviate the curse of dimensionality and preserve the rich word co-occurrence patterns, simultaneously. Furthermore, ensemble techniques are adopted to handle noise text data and achieve robust clustering performance with high efficiency.

III. Related works

A. Consensus Clustering

Tremendous research efforts have been devoted to consensus clustering. These studies can be roughly divided into two categories: CC with implicit objectives (CCIO) and CC with explicit objectives (CCEO). The methods in CCIO do not set any global objective functions for CC. Rather, they employ

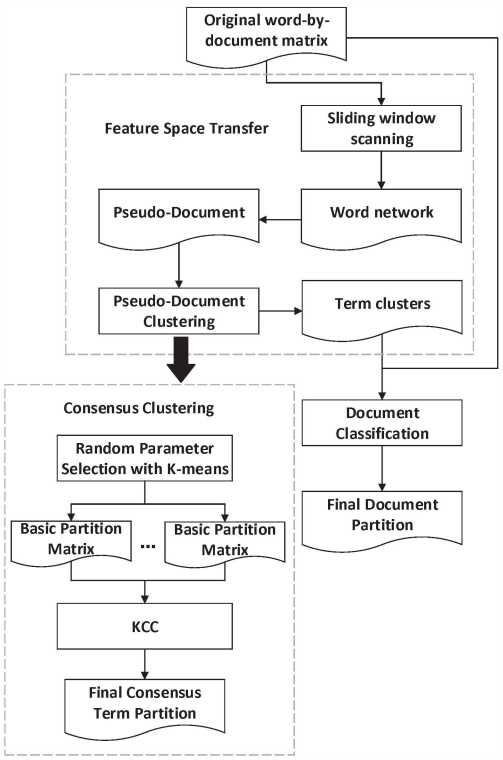

Fig. 2: The Framework of Our Approach.

heuristics to find approximate solutions. In the pioneer work, [3] developed three graph-based algorithms for CC. Although an objective function was defined on the normalized mutual information measure, the proposed algorithms actually do not address this optimization problem directly [7]. Following this idea, [8] built different types of graphs to improve the clustering quality. The methods in CCEO have explicit global objective functions for consensus clustering. For instance, to find the Median Partition based on Mirkin distance [9] proposed three simple heuristics. Along this line, [10] further proposed some new heuristics for enhancement.

B. Topic Modeling

Topic models are tools for clustering co-occurrent words into various topics. Many topic models have been proposed, such as probabilistic latent semantic indexing (pLSI) [11], Latent Dirichlet Allocation (LDA) [12], author-topic model [13], correlated topic model (CTM) [14], and dynamic topic model (DTM) [15], etc. Most of them are designed to handle normal texts with special additional properties, such as time, social relationship and authorship. The sparse short texts have also attracted much research interest in the previous literature, and most early studies mainly focus on increasing data density through utilizing auxiliary information. For example, Hong and Davison [16] train topic models on aggregated tweets that share the same word, and find those models work better than those being directly trained on original tweets. Sahami and Heilman [17] propose a search-snippet-based similarity measure for short texts. Jin et al. Regarding the topic modeling on imbalanced texts, the prior knowledge has been widely used to alleviate skewed distributions over documents of different topics. Andrzejewski et al. [18] propose Dirichlet forest priors to incorporate must-links and cannot-links constraints into topic models.

C. Topic Detection

The main task involved in topic detection problem is document clustering from SMS. Most of the early works on document clustering are traditional clustering based approaches [1], which can be further divided into two categories, namely hierarchical clustering and iterative clustering methods, respectively. As a iterative partitioning based approach, K-means has undoubtedly become a competitive candidate for document clustering due to its linear time complexity in the number of documents [19]. In this paper, we facilitate K-means-based Consensus Clustering [20] to further boost the document clustering performance.

Recently, with the popularity of probabilistic topic modeling, more unsupervised methods are proposed for document clustering. For instance, Xie proposes a method integrating document clustering and topic modeling [21]. Others use matrix factorization based techniques [22] [23]. However, none of these works are designed for short text document clustering. It is worth noting that, with emergence of SMS corpus, some works have been focus on the term-term correlation in the SMS document [2] [24]. But, little has dove deep in the enhancing clustering of the term space, which is the main contribution of our job.

IV. Experimental Results

In this section, we present the experimental results of TCC on real-world short text collections. We mainly compare TCC with several partitioning based clustering methods and probabilistic topic modeling in terms of document clustering performance.

A. Experimental Setup

Data. In the paper, we only study the topic detection problem on short text collections. To investigate TCC's ability of learning coherent topics on real-world SMS collections, we carry out experiments on one day's microblogs sampled from Weibo. As a Twitter-like service in China, Weibo has recently just canceled the length limitation for each tweet, specifically, a 140 Chinese characters limit for each document. Thus, we collect micro-blogs from the Sep. 1st of 2013 to guarantee the SMS characteristics. In order to demonstrate the trending topic detection quality of our method, we extract micro-blogs containing one of the top 10 hashtags in terms of hashtag frequency, and each tweet just only contains one hashtag, indicating its true topic label.

Preprocessing. Since the textual content of micro-blogs is not formal, careful preprocessing is quite necessary. In the preprocessing, we take the following steps to wash the collected corpus:(a) using NLPIR[1] to do tokenization; (b) removing single words and stopping words; (c) filtering out //@, URLs, emoticons, Hashtags, in which //@ indicates the retweet users' nicknames; (e) removing micro-blogs with length < 10. To further reduce the noise information in tweets, we extract top 2,000 tweets with the largest document length in

[1] http://ictclas.nlpir.org/downloads.

each topic category. Finally, 20,000 micro-blogs retained with 1,969 distinct words in total. The average number of tokens in documents is 15.1.

Validity. We use the normalized Mutual Information (NMI) [25] as a external clustering validity. In order to better using the validity indice, we will give details of it. The Mutual Information (MI) measure is developed in the field of information theory. The MI measures how much information one random variable can tell about another one. We use the mutual information between π and $\pi^{'}$ to measure the performance of clustering algorithms. Moreover, normalizing techniques are introduced to normalize the measure into [0, 1] range. A contingency matrix as shown in Table I is used to calculate this measure. Thus, the normalized Mutual Information is defined as follows,

$$NMI = 2\sum_{i}\sum_{j} p_{ij}\log\frac{p_{ij}}{p_i p_j}/(-\sum_{i} p_i \log p_i - \sum_{j} p_j \log p_j).$$

B. Baseline Methods

For partitioning based clustering method, CLUTO is adopted. Note that CLUTO is conducted on the original document space and the transferred term space, respectively. Moreover, we employ LDA on both the original document space and the transferred term space for the integration of document clustering and topic modeling.

CLUTO-D: We use the well-known publicly available tool CLUTO with the implemented spherical K-means algorithm. The CLUTO[2] tool is a state-of-the-art software package for clustering high-dimensional datasets. Since we conduct clustering on the original document space, we denote this method as CLUTO-D.

CLUTO-T: We first conduct feature space transfer on the original document space and then use CLUTO for term clustering on the transferred term space. Finally, we conduct document classification using term clusters similar to TCC. We use CLUTO-T to denote this approach.

LDA-D: For integrating document clustering and topic modeling, we first use Gibbs sampling [26] to infer document-topic distribution θ from the text corpus. θ can be deemed as a mixture proportion vector over clusters and can be utilized for clustering. Then a document is assigned to cluster x if $x = argmax_j \theta_j$. Note that this approach is a naive solution for integrating document clustering and topic modeling together. Since we conduct LDA on the original document space, we use LDA-D to denote this approach.

LDA-T: This is a simple but general solution for SMS topic modeling described in [5]. We summarize the main steps as follows. First, pseudo-documents are generated from the original text corpus. Then Gibbs sampling on the pseudo-documents is applied to discover latent word groups, which are taken as topic components of a corpus. Given topic proportions for all words, document topic proportions can be obtained accordingly. After that, document clustering can be achieved in a similar way to the LDA-D approach.

[2]http://glaros.dtc.umn.edu/gkhome/cluto/cluto/download.

TABLE II: Document Clustering Performance Comparison

	Original Document Space		Transferred Term Space		
#Topics	CLUTO-D	LDA-D	CLUTO-T	LDA-T	TCC
10	**0.3185**	0.2681	0.2682	0.2523	*0.2760*
30	**0.3195**	0.2544	0.3069	0.3149	*0.3179*
50	0.3249	0.2440	0.3253	*0.3270*	**0.3346**
70	0.3298	0.2388	*0.3330*	0.3328	**0.3421**
90	0.3352	0.2359	*0.3421*	0.3366	**0.3448**
110	0.3371	0.2348	*0.3419*	0.3356	**0.3461**

C. Comparisons

In this subsection, we will report the comparison results of TCC versus other baseline methods described in Sect. IV-B. Note that all the experiments in this section are carried out on a Linux Server with two Intel Xeon E5-2609 v2 2.50GHz CPUs and 64G memory.

As shown in Table II, the clustering results of thses methods are illustrated, where the best results are highlighted in bold and the second best are denoted in italic. Note that all the experiment results shown in Table II is the average NMI value in 10 times run. CLUTO tool using in CLUTO-D and CLUTO-T methods implements the spherical K-means with default settings. In CLUTO default settings, cosine similarity is chosen as distance function and each optimization is run for 20 iterations. Moreover, each run is carried out with 10 trials selecting the solution that has the best value the criterion function and each trial is generated with different seeds, enabled for more random partition results. For both LDA-D and LDA-T, we use a Java open-source implementation JGibbLDA[3]. For JGibbLDA, we set $\alpha = 0.1$, $\beta = 0.01$ and each Gibbs sampling is run for 2000 iterations. In term-based methods including CLUTO-T, LDA-T and TCC, we set the length of the sliding window to 10 for generating word network. Then, for clustering terms in pseudo-document space, we set the number of term clusters the same as the topic number of document collections, of which we will give details in the following discussion.

Note that, for TCC, we use CLUTO tool and Random Parameter Selection (RPS) strategy for generating BPs and thus we sample uniformly at random with replacement from the values range from 10 to 100 as the predefined cluster number in BPs. We set the number of BPs to 100 and KCC is run for 10 times to obtain the best result.

In our experiments, it is intuitive to set the final topic number of document clustering to the ground truth number of topics in the weibo text corpus, namely 10. However, we find that clustering performance might be significantly improved by increasing the predefined number of topics in most methods. Therefore, we vary the number of topics from 10 to 110 in Table II to discuss the effect of parameters.

As can be seen from the table, three observations are very clear. First, as for the model parameter, almost all the methods is greatly improved with the increasing number of topics for both original document space and the transferred term space, except for the LDA-D. For instance, TCC achieves a roughly 7% higher performance with #Topics = 110 than with #Topics = 10. This is in great contrast to LDA-D, which shows a slight decline in terms of clustering performance. Second, as for spaces, when #Topics $>=$ 50, methods in

[3]http://jgibblda.sourceforge.net/.

the transferred term space consistently achieve much better clustering performance than methods in the original document space. For instance, LDA based method in the transferred term space obtains a 10% performance improvement with #Topics ranges from 70 to 110. When #Topics < 50, CLUTO-D achives the best performance and TCC obtain the second best. Third, as for models, partitioning based clustering methods are generally better than topic modeling methods, in terms of document clustering performance. For instance, CLUTO-D is 10% higher than LDA-D with #Topics ranges from 90 to 110. More importantly, we discover that in the transferred term space, consensus clustering based method is better than single clustering method. For instance, our consensus clustering based method TCC consistently outperforms the single clustering method CLUTO-T in all topic parameter settings.

In summary, the optimal document clustering performance from short text collections is achieved by incorporating censensus clustering in term space while choosing a large topic number, which is our main effort in this paper. The reasons behind this optimal performance are given as follows. First, term space is much lower dimensional and more dense and thus enrich the feature information of the original SMS corpus. Second, larger number of topics enhance the generation of more cohesive clusters. Moreover, ensembling of basic partitions help to produce more robust result in the term space.

V. Conclusion

In this paper, we propose the TCC, namely Term-based Consensus Clustering, a simple but effective approach for topic detection from short text. In particular, we adopt a consensus clustering technique called K-means-based Consensus Clustering to facilitate basic partitions for generating robust clustering result. To further alleviate the extreme sparsity of SMS corpus, we employ a feature space transfer techique to transfer the original text corpus to the dense term space. Our method is evaluated on a real-world SMS dataset and achieve comparable performance to several widely used methods in terms of topic detection quality. Experimental results show that our approach utilizes the ensemble effect of consensus clustering and improves the document clustering performance in term space, simultaneously. Particularly, we demonstate that TCC obtains the best clustering performance when observing a large number of the predefined topics across short text collections.

Acknowledgment

This work was partially supported by the National Natural Science Foundation of China (71322104,71471009,71531001), the National High Technology Research and Development Program of China (863 Program SS2014AA012303), and the Fundamental Research Funds for the Central Universities (Junjie Wu).

References

[1] Y.-W. Seo and K. Sycara, "Text clustering for topic detection," 2004.

[2] S. Seifzadeh, A. K. Farahat, M. S. Kamel, and F. Karray, "Short-text clustering using statistical semantics," in *Proceedings of the 24th International Conference on World Wide Web Companion*. International World Wide Web Conferences Steering Committee, 2015, pp. 805–810.

[3] A. Strehl and J. Ghosh, "Cluster ensembles—a knowledge reuse framework for combining multiple partitions," *The Journal of Machine Learning Research*, vol. 3, pp. 583–617, 2003.

[4] G. Salton, "Automatic text processing: The transformation, analysis, and retrieval of," *Reading: Addison-Wesley*, 1989.

[5] Y. Zuo, J. Zhao, and K. Xu, "Word network topic model: a simple but general solution for short and imbalanced texts," *Knowledge and Information Systems*, pp. 1–20.

[6] J. Wu, H. Liu, H. Xiong, J. Cao, and J. Chen, "K-means-based consensus clustering: A unified view," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 27, no. 1, pp. 155–169, 2015.

[7] S. Vega-Pons and J. Ruiz-Shulcloper, "A survey of clustering ensemble algorithms," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 25, no. 03, pp. 337–372, 2011.

[8] X. Z. Fern and C. E. Brodley, "Solving cluster ensemble problems by bipartite graph partitioning," in *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, p. 36.

[9] B. Mirkin, "The problems of approximation in spaces of relations and qualitative data analysis," *Information & Remote Control*, 1974.

[10] P. Tsaparas, H. Mannila, and A. Gionis, "Clustering aggregation." *Acm Transactions on Knowledge Discovery from Data*, vol. 1, no. 1, pp. 341–352, 2007.

[11] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1999, pp. 50–57.

[12] D. M. Blei and A. Y. Ng, "Latent dirchlet allocation," 2003, pp. 362 – 365.

[13] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth, "The author-topic model for authors and documents," in *Proceedings of the 20th conference on Uncertainty in artificial intelligence*. AUAI Press, 2004, pp. 487–494.

[14] J. D. Lafferty and D. M. Blei, "Correlated topic models," in *Advances in neural information processing systems*, 2005, pp. 147–154.

[15] D. M. Blei and J. D. Lafferty, "Dynamic topic models," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 113–120.

[16] L. Hong and B. D. Davison, "Empirical study of topic modeling in twitter," in *Proceedings of the first workshop on social media analytics*. ACM, 2010, pp. 80–88.

[17] 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*. AcM, 2006, pp. 377–386.

[18] D. Andrzejewski, X. Zhu, and M. Craven, "Incorporating domain knowledge into topic modeling via dirichlet forest priors," in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 25–32.

[19] M. Steinbach, G. Karypis, V. Kumar *et al.*, "A comparison of document clustering techniques," in *KDD workshop on text mining*, vol. 400, no. 1. Boston, 2000, pp. 525–526.

[20] B. Mirkin, "Reinterpreting the category utility function," *Machine Learning*, vol. 45, no. 2, pp. 219–228, 2001.

[21] P. Xie and E. P. Xing, "Integrating document clustering and topic modeling," *arXiv preprint arXiv:1309.6874*, 2013.

[22] T. Li, C. Ding, Y. Zhang, and B. Shao, "Knowledge transformation from word space to document space," in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2008, pp. 187–194.

[23] D. Cai, X. He, and J. Han, "Locally consistent concept factorization for document clustering," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 23, no. 6, pp. 902–913, 2011.

[24] N. Slonim and N. Tishby, "Document clustering using word clusters via the information bottleneck method," in *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2000, pp. 208–215.

[25] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012.

[26] G. Heinrich, "Parameter estimation for text analysis," *University of Leipzig, Tech. Rep*, 2008.