ISBN 978-981-11-7861-0
Proceedings of 2018 the 8th International Workshop on Computer Science and Engineering (WCSE 2018)
Bangkok, 28-30 June, 2018, pp. 245-250
doi: 10.18178/wcse.2018.06.044

Microblog Text Clustering Based on BK-Means Algorithm

Qianru Li [1], Xiuliang Mo [1+] and Chundong Wang [1]

[1] Tianjin Intelligent Computing and Software New Technology Key Laboratory School of Computer Science and Engineering, Tianjin university of technology, Tianjin 300384, China

Abstract. In recent years, the increasing popularity of social media such as WeChat and Weibo has facilitated the communication among people. However, due to the characteristics like large scale, fast propagation, low quality and diverse modalities of social short texts, the short text clustering faces the challenge of sparse features, high dimension and noise interference. The traditional clustering method based on vector space model is not good for short text data processing. With the improvement of K-means algorithm, this paper proposes a short-text clustering algorithm named BK-means which alleviates the effect of data sparseness. Firstly, we preprocess the wordset by means of word segmentation, stop-of-word and other operations, then extract the biterm using the BTM to model the document, and get the document-topic, the topic-word distribution matrix. Finally, we use the proposed BK-means algorithm to cluster short texts of documents represented by vectors. Experiments on the short text data of Sina Weibo have proved that the short text clustering algorithm based on BK-means is superior to the traditional one, and both the F-measure and the purity are improved.

Keywords: microblog text, BTM, BK-means algorithm, F-measure, purity.

1. Introduction

Into the Web 2.0 era, Microblog and other social networking media which are rising provide people with the convenience of communication. Weibo is a social platform that can share various kinds of information and get popular topics. Its notable feature is the word limit. When users publish a microblog, they can only publish and share their views of 140 characters. With the short forging of microblogs, a large number of short texts are flooding the internet. Compared to the traditional text, Weibo short text lacks of context information. These short texts are short for content, large in data size and contain a large amount of hidden information. Clustering microblog short texts are of great research value of mining user interests [1], hot topic discovery[2] and personalized recommendation system[3].Aiming at the problems such as high complexity, sparsity of feature words and much noise data, the paper analyzes the clustering effect of the existing short text clustering algorithms. Based on this, a new method based on BK-means Short Text Clustering Algorithm.

This paper is organized as follows. Section 2 shows related work. Section 3 introduces the design and implementation of short text clustering. Section 4 contains the experiments finally Section 5 summarizes the full text and looks forward to the future work.

2. Related Work

At present, the research on short text clustering is mostly based on the traditional Vector Space Model (VSM) [4]. The vector space model is the most commonly used model of document representation, which has achieved good results when dealing with the traditional long text problem [5]. Using VSM to vectorize Weibo text effectively improves the ability of processing and analyzing Weibo text, but because VSM is

+ Corresponding author. Tel.: +18602246736.
E-mail address: moxiuliang@163.com.

based on the independent premise of the words with the text, it ignores the semantics of the words in the Weibo text information. Because of the special stylistic features such as short, uniqueness and arbitrariness of the Weibo texts, the constructed vector space of the document has high dimension and sparse data, which can't achieve the desired result [6-7]. Peng Min et al.[8] proposed frequent itemisers spectral clustering algorithm with adaptive clustering number adaptively, realized dimension reduction in frequent itemised filtering based on similarity, and realized mass short text clustering quickly and effectively. For feature extraction and representation of short texts, traditional vector space models often lose semantic information and may result in sparse feature dimensions. Latent Semantic Analysis (LSA), Probablistic Latent Semantic Analysis (PLSA) and Latent Dirichlet Allocation (LDA) are all common topic models [9-10]. These models are often applied to traditional texts, did not take into account the special nature of the short text microblog can't be well applied to short text, thus affecting the quality of microblog short text clustering. For the topic model, the BTM (biterm topic model) proposed by Yan Xiaohui [11] can find more prominent and semantic topics from short texts.

Based on the above analysis, this paper proposes a microblog short text clustering method named BK-means algorithm, to a large extent alleviate the impact on data sparse.

3. Design and Implementation

Weibo short text data set is crawled through the open source web crawler, the original data set is very important. As the traditional theme model learns thematic topics through document-word co-occurrence, they will face the problem of feature sparseness when dealing with short texts, making the clustering effect greatly reduced. Based on the BK-means algorithm, this paper establishes the overall framework of microblog short text data clustering. This process is mainly divided into four stages, including data preprocessing, topic modeling, document representation, algorithm clustering, etc. In order to overcome short text data Set features sparse effect, to achieve the microblog short text clustering. It is illustrated in Figure 1.

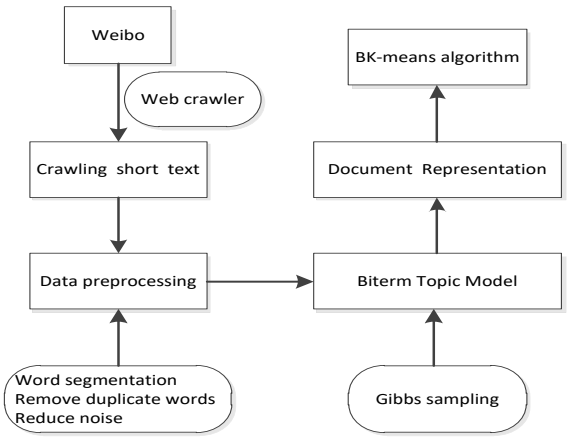

Fig. 1: Microblog short text cluster based on BK-means algorithm.

3.1. Biterm topic model

BTM is a production model whose key idea is to use the biterm generated throughout the corpus to learn the topic of short texts. Any two disparate words that co-appear in the same text segment after preprocessing are called a biterm [11]. The biterm are extracted from all the text in the corpus, and the extracted biterm are modeled.

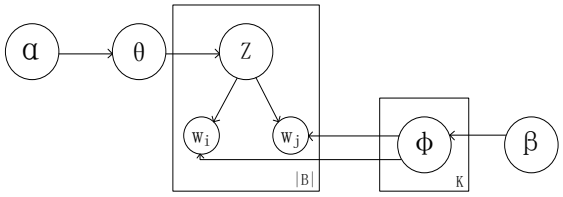

Fig. 2: Biterm topic model.

In Figure 2, θ is the corpus-level topic distribution, φ is the distribution of thematic keywords, | B | is the number of biterm in the entire corpus, and w_i is a word-to-biterm, and w_j is two different words. The Biterm topic model uses the information on the entire microblog short text set to form the biterm of the microblog short texts so as to describe the microblog topic Z at the whole corpus level ,thus not only maintaining the relativity between words and obtain different words but also expressing the independence of different topics.

The joint probability of a biterm is expressed as follows:

$$P(b) = \sum_z P(z)P(w_i|z)P(w_j|z) = \sum_z \theta_z \varphi_{i|z} \varphi_{j|z} \tag{1}$$

Then the probability of the entire BTM corpus is expressed as:

$$P(B) = \prod_{(i,j)} \sum_z \theta_z \varphi_{i|z} \varphi_{j|z} \tag{2}$$

The parameters φ and θ of the BTM are extrapolated using the Gibbs sampling method. The Gibbs Sampling Method is an efficient Markov Chain-Monte Carlo MCMC sampling method that utilizes the conditional distribution of each variable to achieve sampling in the joint distribution [12]. The initial state of the Markov chain should be chosen randomly before Gibbs sampling, then the conditional probability $P(z|z_{-b}, B, \alpha, \beta)$ of each biterm b = (w_i, w_j) is calculated by applying the rules of Markov chain to the whole The joint probability of data up to obtain the conditional probability. The formula is as follows:

$$P(z|z_{-b}, B, \alpha, \beta) \propto (n_z + \alpha) \frac{(n_{w_i|z} + \beta)(n_{w_j|z} + \beta)}{(\sum_w n_{w|z} + M\beta)^2} \tag{3}$$

where z_{-b} denotes the topic assignment to all biterm except b; B denotes all biterm in the corpus; n_z denotes the number of times the biterm are assigned the topic z; $n_{w|z}$ denotes the number of times the word w is assigned the topic z; M the number of different words in the corpus.

The subject distribution θ and the subject-word distribution φ in the corpus are obtained by combining Gibbs sampling results from the following formula:

$$\theta_z = \frac{n_z + \alpha}{|B| + K\alpha} \tag{4}$$

$$\varphi_{w|z} = \frac{n_{w|z} + \beta}{\sum_w n_{w|z} + M\beta} \tag{5}$$

$\Phi_{w|z}$ is the probability of word w in topic z, θ_z is the probability of topic z, and | B | is the total number of biterm.

3.2. Document Representation

Biterm topic model can be more ideal document theme, keyword probability distribution. Taking advantage of the probability distribution of the topic of BTM training results, the feature words of Top N are extracted from each topic, and the short texts are transformed into feature words vectors combined with TF-IDF (word frequency-inverse text frequency) calculation strategy, denoted as d_{VSM}; meanwhile, after the training of the Biterm topic model, the subject-matter distribution matrix is used to represent the document, which is denoted as d_{BTM}. Finally, the similarity values calculated by the two methods are fused proportionally in the document clustering process. The most classical method for weighting feature words in vector space models is TF-IDF.

The weight of the characteristic words in the article is calculated as:

$$W = TF_{(i,k)} \times IDF_{(i,k)} \tag{6}$$

The similarity in text clustering is measured by JS distance. JS (Jenson-Shannon) distance can measure the distance from the probability distribution, so it is used as a measure of BK-means similarity. Define as follows:

$$d_{js} = \frac{1}{2}\left(\sum_{j=1}^{k} p_j \ln \frac{p_j}{q_j} + \sum_{j=1}^{k} q_j \ln \frac{q_j}{p_j}\right) \tag{7}$$

Here, $p = (p_1, p_2, ..., p_k)$, $q = (q_1, q_2, ..., q_k)$ are the theme probability vector.

3.3. BK-means Algorithm

K-means is a cluster-based clustering algorithm, but the initial clustering centers and K values need to be manually intervened, which is easily affected by the discrete values to make the clustering effect fall into the local optimal solution [13]. Qiu Rongtai proposed using the Canopy algorithm to optimize the K-means algorithm to further optimize the initial selection of the center, but the determination of the initial threshold size of the Canopy algorithm is generally based on manual selection, so the effect is not stable [14].In the BK-means algorithm proposed in this paper, the central idea is to optimize the first stage Canopy algorithm preprocessing, the overlapping subset formed by each Canopy algorithm is called the cover set. In the process of K-means clustering, it will no longer consider the distance between each point and all centers as the traditional K-means algorithm, and only needs to calculate the distance between the points and the center of the cover to which it belongs. With the K-means algorithm the iteration, each cover center will also continue to change until convergence. The specific implementation process was shown in Figure 3.

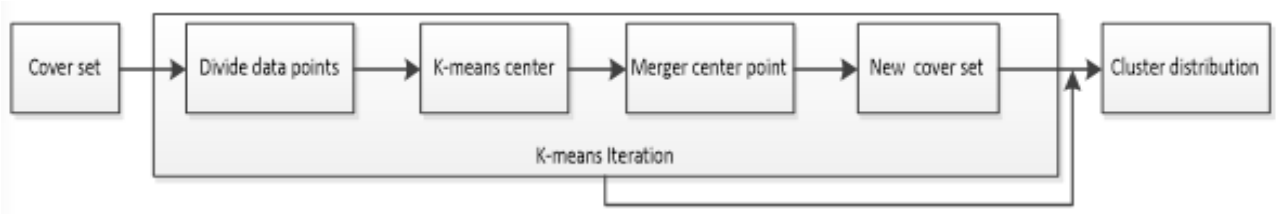

Fig. 3: BK-means algorithm.

As can be seen from Figure 3, after introducing Canopy, BK-means algorithm only compares the distance between the object and its own cover center in the same region at each time. By reducing the number of times of comparison, it greatly reduces the running time of the whole cluster and improves the algorithm's calculation effectiveness. Specific implementation steps are as follows:

1) Generate a cover set. That is, for the dataset $Z = (z_1, z_2, \ldots, z_n)$, the initial set of center points $\{p_1, p_2, \ldots, p_n\}$ are iteratively identified by the thresholds T1, T2 specified by the Canopy algorithm to form a cover set.

2) Divide data points. After generating the cover set, that is, for any point z_i, suppose that $p_i = \{p_1, p_2, \ldots, p_n\}$ is the set of cover centers to which it belongs and if z_i is the smallest distance from the center points p_j of one cover set to which it belongs, To p_j belongs cover set, and remove it from the other cover set.

3) K-means center. Using the cluster generated in the previous step, a new center point of each cluster is calculated, the center points closer to each other are merged, the clusters corresponding to the cluster are merged accordingly, and the merged new center is calculated to generate the final of one iteration K center to avoid the problem of instability of the clustering effect caused by the artificial radius.

4) Form new clusters, create new ones, and iterate. Calculate which K-centers of the merged K-centers fall on the previous one, and replace the centers of these sets with the new K-centers to form a new one. This step and step 3 is the process of combining to create a new Canopy. From step 2, repeat the above steps until the algorithm converges.

4. Experiments

In this paper, we compared the clustering method based on BK-means algorithm and the traditional K-means algorithm on the effect of short text clustering to compare the clustering results and deficiencies in clustering algorithm.

4.1. Dataset

In this article, we use the open source web crawler to crawl as a short text dataset of social hot events in Sina Weibo. There are about 2 million short text data used to verify the performance of the clustering algorithm. Using the word segmentation tool ICTCLAS of Chinese Academy of Sciences, word segmentation, stop-of-word, de-emphasis, noise reduction and other data preprocessing operations are used to obtain a more accurate short text data set.

4.2. Quantitative Evaluation

For clustering algorithm, F-measure and Purity are used to measure the advantages and disadvantages of clustering algorithm.

4.3. Experimental Results Analysis

We test the validity of BK-means algorithm by using 2 million short text data sets from Sina Weibo. Repeated extraction of data to experiment, to a certain extent, eased the problem of high feature sparseness in short texts dimension. In order to reduce the experimental error, the data were repeated several experiments, Table 1 is the number of iterations when the test results of 200 times. Because each time the data selected are random, excluding a large difference in the average of several data points.

Table 1: Experimental results of two algorithms

Number	K-means algorithm		BK-means algorithm	
	F1	P1	F2	P2
1	0.3512	0.3430	0.5334	0.5021
2	0.3677	0.3521	0.5544	0.5245
3	0.3892	0.3734	0.5789	0.5434
4	0.3965	0.3777	0.5609	0.5545
5	0.3884	0.3756	0.5678	0.5578
6	0.4001	0.3829	0.5698	0.5590
7	0.3922	0.3873	0.5601	0.5500
8	0.3978	0.3856	0.5790	0.5691
9	0.4024	0.3901	0.5701	0.5655
10	0.4019	0.3889	0.5726	0.5678
11	0.3997	0.3898	0.5744	0.5669
Average	0.3897	0.3769	0.5656	0.5510

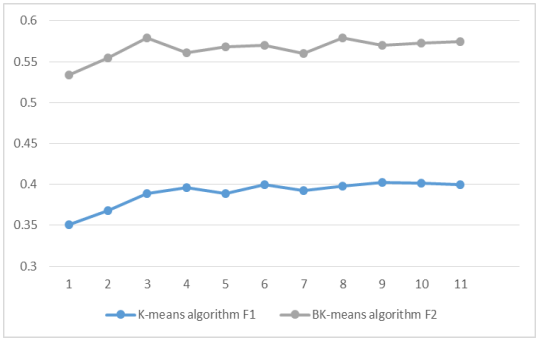

Fig. 4: F-measure comparison chart.

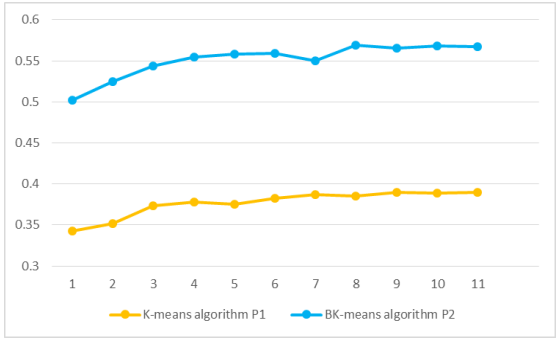

Fig. 5: Purity comparison chart.

As can be seen from Figure 4 and Figure 5, the F-measure and Purity of the BK-means algorithm are higher than the traditional K-means algorithm. It shows that the algorithm in this paper optimizes the choice of the center, and the clustering results have higher interclass similarity and faster convergence rate. Experiments show that the short text clustering algorithm based on BK-means is superior to the traditional short text clustering algorithm, and the F-measure and the value of the purity are obviously improved.

5. Conclusion

The research of microblog short text clustering has important practical significance and application requirements. In this paper, we propose a microblog short text clustering method based on BK-means

algorithm, which largely alleviates the effect of data sparseness. By using the BTM to model the lexical learning subject in the corpus, the problem of the sparseness of the short text in the microblog is overcome. And by improving the TF-IDF algorithm to adapt to the requirements of microblog feature extraction. Experiments conducted on the Sina Weibo short text data set to demonstrate that the short text clustering algorithm based on BK-means is superior to the traditional K-means clustering algorithm. The research in this paper mainly focuses on improving the effect of microblog short text clustering. In the experiment, there is a problem that the BTM modeling speed is slow. Therefore, in the follow-up study, we will consider how to improve the efficiency of modeling to adapt to the application of massive Weibo data. The next step is to explore ways to improve algorithmic efficiency and algorithm parallelism.

6. Acknowledgements

Our work was supported by the Foundation of the Educational Commission of Tianjin, China (Grant No.20130801), the General Project of Tianjin Municipal Science and Technology Commission under Grant(No.15JCYBJC15600), the Major Project of Tianjin Municipal Science and Technology Commission under Grant(No.15ZXDSGX00030), and NSFC: The United Foundation of General Technology and Fundamental Research (No.U1536122). and the Major Project of Tianjin Smart Manufacturing (No.15ZXZNCX00050).We would like to give thanks to all the pioneers in this field, and also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the quality of this paper.

7. References

[1] J. Weng, E. Lim, J. Jiang, and Q. He *Twitterrank: finding topic-sensitive influential twitterers.* Proceedings of the third ACM conference on Web search and data mining. 2010, pp, 261-270.

[2] X. Yan, J. GUO, S. Liu, X. Cheng, and Y. Wang. L*earning topics in short texts by non-negative martrix factorization on term correlation matrix.* Proceedings of the 2013 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics. 2013, pp. 749-757.

[3] Phelan, K. McCarthy, and B. Smyth. *Using twitter to recommend real-time topical news.* Proceedings of the third ACM conference on Recommender systems. 2009, pp. 385-388.

[4] G. Salton, A. Wong, CS. Yang. *A vector space model for automatic indexing[J].* Communications of the ACM. 1975, **18** (11): 613-620.

[5] MW. Berry, M. Castellanos. *Survey of text mining II[M]*. New York: Springer, 2008.

[6] Z. Wu. *The Study of Genre Characteristics of Chinese Weibo [D].* Wuhan: Central China Normal University, 2012.

[7] S. Ding, M. Ren, X. Li. *Research on Viewpoint Sentences for Chinese Weibo [J]*. Acta Intelligence. 2014, 33 (2): 175-182.

[8] M. Peng, J. Huang, J. Zhu, et al. *Massive short texts and topic extraction based on frequent itemsets [J].* Computer Research and Development. 2015, 52 (9): 1941-1953.

[9] Y. Zhang, J. Zhu, Z. Xiong. *Text clustering algorithm based on improved probabilistic latent semantic analysis [J].* Computer Applications. 2011, 31 (3): 674-676.

[10] R. Lu, L. Xiang, M. Liu, et al. *Discovery of news topics in micro-blog based on implicit topic analysis and text clustering [J].* Pattern Recognition and Artificial Intelligence. 2015, 25 (3): 382-387.

[11] X. Yan, J. Guo, Y. Lan, et al. *A Biterm Topic Model for Short Texts [C]*. International Conference on World Wide Web. 2013, pp. 1445-1456.

[12] G. Stuart, G. Donald. *Stochastic relaxation, Gibbs distributions, and the bayesian reStoratlon of images [J]*. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1984, 6 (6): 721-741.

[13] N. Fan. *Simulation study on commercial bank custermer segmentation on K-means clustering algorithm [J]*. Computer Simulation. 2011, 28 (3): 369-372.

[14] R. Qin. *Canopy for K-means on multi-core [J].* Microcomputer Information. 2012, 9: 486-487.