DVM-based Topic Detection for Microblog

LV Jia-Guo[1], JIANG Xiu-Ying[1*], CHI Qing-Yun[1], Zhang Wei[1], JOCSHI Allen[2]
[1] School of Information Science and Engineering, Zaozhuang University, Zaozhuang 277100, China
[2] Network Information Center for Design and Analysis, MCCN Ltd, Gdansk ,11952, Poland
*Corresponding Author, email: 805700065@qq.com

Journal of Digital Information Management

ABSTRACT: *With the rise of microblog, topic detection in microblog posts has been a hotspot in natural language processing and text mining. Different from regular text, microblog post is a kind of short and idiomatic text. Microblog post contains little information, which brings great challenge for its topic detection. To address the issue of topic detection in microblog, a new single pass algorithm based on a double-vector model (DVM; Single Pass_DM) is proposed. First, a support vector machine (SVM) based algorithm is employed to filter irrelevant posts, thereby improving the accuracy of the algorithm. As for the representation model, on the basis of the traditional vector space model, a DVM that includes event and keyword vector is put forward. Subsequently,, a combination of Jacoby ,cosine and semantic similarity is used for similarity computation. Finally, some structural characteristics of microblog posts are used to support the topic detection problem. To validate the performance of the proposed algorithm, experiments are conducted on a real-world dataset. Experimental results show that, comparing with three benchmark algorithms SinglePass, Agglomerative Hierarchical Clustering (AHC) and Density-based Spatial Clustering (DBSCAN), the performance of SinglePass_DM has been improved greatly.*

Subject Categories and Descriptors
J.1 [Computer Applications]: Data mining;

General Terms
Microblog, Text mining, Topic detection

Keywords: Topic Detection, Microblog, DVM, SinglePass_DM

Received: 1 March 2016, Revised: 27 May 2016, Accepted: 3 June 2016.

1. Introduction

In recent years, along with the convenience of following, forwarding and commenting, microblog has become an important platform for information exchanging and sharing. The last few years have witnessed the rapid development of microblog platforms, such as Twitter and Sina Weibo. Topic detection on microblog contributes in filtering noisy information and improving users' experience. Moreover, topic detection plays an important role in sentiment analysis. Microblog posts are limited to 140 characters, and 70% of them are within 10 words. Because of the limited information that microblog post contains,several traditional topic detection algorithms are inapplicable. In content, microblog post is from daily life. In language style, micorblog post is usually personalized, symbolic, colloquial and noisy, which reduces the accuracy of similarity computing among posts. Moreover, the emergency of events requires the topic detection algorithm to provide users with topics quickly.

Based on the status discussed above, most state-of-the-art algorithms cannot meet the demand of topic detection in microblog posts. To cope with this problem, the similarity accuracy of posts and the running efficiency of

the current topic detection algorithm should be improved. So, in this work, a new single-pass algorithm based on a double-vector model (DVM; SinglePass_DM) is put forward. To improve the running efficiency of SinglePass_DM, the algorithm is based on the highly efficient SinglePass algorithm. To improve the accuracy of the topic detection, a few measures, such as irrelevant posts filtering, introduction of DVM, and adoption of combined similarity, are performed in the proposed algorithm.

2. State of The Art

Topic detection starts from the topic detection and tracking program [1], which automatically detects different topics from the stream of texts. In this field, topics are usually represented by the vector space model (VSM) or the probabilistic model. In the probabilistic models, Latent Dirichlet allocation (LDA)[2] is the most well-known one. LDA assumes that text can be the mixtures of topics, and each of which is a distribution over words. However, the topic obtained from LDA usually has a large semantic granularity, and it is difficult to describe the theme of the text accurately. When a text is expressed by VSM, the vector is composed of keywords extracted from the text. Due to the lack of information carried by VSM, the accuracy of the method cannot meet the demand of topic description of texts. To this end, DVM, in which a topic is expressed by event and keywords vectors, is proposed. In DVM, if the event elements and the keywords extracted from two posts are similar, they are likely to discuss the same topic.

As previously discussed, microblog post is short and contains little information, which makes those traditional topic detection algorithms unsuitable. There are extensive studies for topic detection on microblog, and most of them are based on Twitter [3,4]. As for the topic detection on microblog, the most important technologies are the representation model and the topic detection method. There are two widely used representation models, one is VSM based on TF-IDF[5], and the other is LDA. As for the topic detection algorithms, most of them are based on the clustering method, such as Single-Pass[6, 7], K-Means, and HMM [8]. To improve the performance of topic detection, a few researchers combined the detection algorithm with microblog structures. With the dialog property of microblog, a two-stage clustering algorithm was proposed in [9]. In microblog, the rapid growth of emotional words usually indicates a hot topic. Based on this observation, in [10], an emotional language model was proposed, and then, a new hot event detection method based on the difference of two emotional language models in two neighboring periods was put forward.

In this study, to address the inaccuracy of similarity between two microblog posts and improve the performance of topic detection, a new topic detection algorithm SinglePass_DM is proposed. The main contributions of this study are as follows.

(1) A new representation model of microblog post and topic, which contains the event and keyword vectors, is proposed;

(2) An irrelevant posts filtering algorithm, which is based on support vector machine (SVM), is put forward;

(3) A new method integrated with TF-IDF and momentum model is proposed to calculate the weight of a term in extracting feature words;

(4) In similarity calculation, a combined similarity integrated with cosine similarity, Jacoby similarity and semantic similarity is put forward;

(5) With the structural characteristics of microblog, based on the SinglePass algorithm, SinglePass_DM is proposed.

The remainder of this paper is organized as follows. The framework of the topic detection is presented in section 3. The experimental results are reported in section 4. Finally, the conclusion of the study is presented in section 5.

3. Methodology

As discussed above, microblog is a kind of short text with structure information. To solve the challenges of runtime efficiency and accuracy for topic detection in microblog, a topic detection framework is proposed.

3.1 The representation model

The topic detection method in this study is based on time window. Suppose the size of time window is w, the i-th time window $w_i = [t_i, t_i + w]$, where t_i is the beginning of the i-th time window. For the DVM model, for a given time window w_i, the microblog post $d_{ij} = (de_{ij}, dk_{ij})$, where de_{ij} and dk_{ij} denote the event and keyword vector of post respectively. Suppose, L_e and L_k are the upper dimension limitations of the event and keyword vectors, then, de_{ij} is represented as follows.

$$de_{ij} = (t_{ij1}, w_{ij1},....,t_{ijk}, w_{ijk},...,t_{ijdim(deif)}, w_{ijdim(deif)} \quad (1)$$

where $dim\ (de_{ij})$ is the dimension of event vector de_{ij}; $dim\ (de_{ij}) < L_e$; t_{ijk} is the k-th term of de_{ij}, w_{ijk} and is the weight of t_{ijk}.The three element factors of the event are location, time and character. When the event vector is established, the three factors are extracted from the results of word segmentation and acted as the terms. For a term j in post i, the weight is evaluated as follows.

$$w_{ij} = TFIDF_{ij} \quad \frac{tf_{ij} * log\ (N/n_j + 0.01)}{\sqrt{\sum_{p=1}^{K} [tf_{ij} * log\ (N/n_j + 0.01)]_2}} \quad (2)$$

$$tf_{ij} = m_{ij} / M_i \quad (3)$$

In Equation (3), m_{ij} is the frequency of term j in post i, and M_i is the number of terms in post i. In Equation (2), K is the number of terms in post i, N is the number of posts in

microblog corpus, and n_j is the number of posts that contain term j.

For the keyword vector, dk_{ij} is represented as follows:

$$dk_{ij} = (t_{ij1}, w_{ij1},....,t_{ijk}, w_{ijk},...,t_{ijdim(dkif)}, w_{ijdim(dkif)} \quad (4)$$

In Equation (4), $dim\ (dk_{ij})$ is the dimension of event vector dk_{ij}; $< L_k$; t_{ijk} is the k-th term of dk_{ij}, w_{ijk} and is the weight of t_{ijk}. After removing the event-related terms and stopping words, the nouns of the microblog post are selected as the candidate term of the keywords vector. The top L_k candidate terms with maximum weight are selected as the terms of the keyword vector.

3.2 Prepocess

Microblog posts heavily relate to ordinary people and contain numerous noise data. Therefore, these posts must be preprocessed before the topic detection. Preprocessing is important for the accuracy of topic detection.

3.2.1 Filtering irrelevant posts

There are a lot of microblog posts, which are irrelevant to the topic detection. In topic detection, the two kinds of irrelevant posts are the noise posts and weak-influence posts. Noise post includes the garbage post that is full of advertising information, and the log post that is filled with contents related to the daily life and emotion of the author. Due to the weak influence, the weak-influence post is disregarded in topic detection. The characteristics of noise post are as follows:

(1)Most noise posts have more than two URLs, whereas regular post usually has no more than one;

(2)The expression symbols in noise posts are usually more than 30% of the text, whereas that in normal posts are less than 7%;

(3)The proportion of continuous special symbols in noise posts is approximately 15%, whereas regular posts do not contain these symbols;

(4)The number of sentences in noise posts is usually less than four, whereas that in regular posts is usually more than five.

Based on these characteristics, a new SVM-based algorithm is proposed. For SVM, the classification function f(x) is obtained when the appropriate kernel function K(x, y) and the corresponding parameters are determined, which is shown as follows:

$$f(x) = \mathrm{sgn}\ (\sum_{n-1}^{n} a_i * yK(x_i, x) + b) \quad (5)$$

where x is an n-dimension training sample, x_i is the i-th support vector, and b is the threshold determined by the training samples. The SVM classifier is formed as follows:

(1) Based on the microblog posts with their class labels, the training sample set $T = \{(x_1, y_1), (x_2, y_2),..., (x_m, y_m)\}$ is obtained, where x_i is the training sample, and y_i is the class where x_i belongs.

(2)In this study, RBF is selected as the kernel function. $K(x_i, x_j) = exp\ (- \gamma\ ||x_i - x_j\ ||^2)$ In the experiments, $\gamma = 1.0$, penalty coefficient $C = 2.0$, and the number of training samples $m = 400$.

(3)Based on the kernel function and corresponding parameters, the optimal solution of a and b will be obtained.

(4) With K, a, b and Equation (5), the classification function $f(x)$ will be got.

With the SVM classifier, a noise post filtering algorithm alg_filter1 is obtained. The main steps are as follows.

Step1.For a post p, some corresponding features are extracted, such as the number of URL, expression symbols, continuous special symbols and the number of sentences.

Step2.Post p is classified with the SVM classifier, and the noise posts are removed.

With the noise post filtering algorithm alg_filter1, the irrelevant post filtering algorithm alg_filter2 is obtained. The main steps are as follows:

Step1. The number of microblog fans threshold FT that of micorblog comments threshold RT are determined; then, the microblog post stream is entered;

Step2. The algorithm alg_filter1 filters the noise posts from the post stream of the microblog;

Step3. For a post in post stream, if its fan number < FT or comment number < RT, it will be removed from the post stream; otherwise, the post is saved to the list of output posts.

3.2.2 Preprocessing microblog text

After filtering the irrelevant posts, the remaining ones are preprocessed to improve the accuracy of topic detection. In this study, the rules for preprocessing of these posts are as follows:

(1)The non-text messages are removed from the post.

(2)The messages unrelated to topic detection, such as "@someone," "#topic#," "forward," etc., are removed.

(3)The cyber-popular words widely used in microblog posts are replaced with normal words with the same meaning.

3.2.3 Word segmentation and stopwords processing

Word segmentation is the key in topic detection for microblog posts. In this work, ICTCLAS is employed for word segmentation. When ICTCLAS works, it can obtain an accuracy of 98.45% and a speed of 500KB/S.

After segmenting the words, stopwords should be removed. In removing stopwords, the stopword list and POS filtering are used. First, the stopwords are organized

into a list. All words from the microblog post that belong in the stopword list are removed. All punctuations, prepositions, conjunctions, interjections, and pronouns are deleted from the microblog posts using the ICTCLAS POS tagging.

3.3 Feature words extraction

There are two kinds of feature words in DVM. The feature words for the event vector are extracted by Equation (2).

Thanks to the services of microblog platform, such as short messages publishing, comments, forwarding, etc, some topics may spread suddenly and quickly. Most existing algorithms are unsuitable for the topic detection of short text. It is observed that the stages of microblog topic spreading is similar to the phases of a body in dynamics from the state of rest to the state of motion, and finally to the state of rest. In microblog platform, the frequency of the feature words related to the topic will change synchronously with the hot degree of the topic. Therefore, for the feature words extracting in keyword vector, a method integrating the TF-IDF method and the momentum model is employed. In this work, the bursty of a word is evaluated by its momentum,and the weight of a candidate word is adjusted by its momentum.

Definition 1. Quality. The quality of a candidate word refers to its importance, which is stable in a long period of time. Suppose D_{sj} denotes the post set that contains all the posts with term j in the time window w_s, then the quality of term is evaluated as follows:

$$m(j) = (1/|D_{sj}|)\sum_{i\in Dsj}(m_{ij}/M_i) \quad (6)$$

Definition 2. Speed. The speed of a candidate word refers to the attention degree at a certain time, which is related to its frequency in time window w_s and the influence degree of the corresponding post. The speed is calculated as follows:

$$v(j,s) = |tf(j,s) - tf(j,s\text{-}1)| + \sum_{i\in Dsj} p_i(s) * tf(i,j) \quad (7)$$

where $tf(j,s)$ refers to the frequency of term j in w_s, $p_i(s)$ is the interest degree of post i in w_s, and $tf(i,j)$ denotes the frequency of term j in post i. $p_i(s)$ is related to the fan number of the author of post i, the number of forwarding and commenting of post i. The frequency is calculated as follows:

$$p_i(s) = \log(f_i(s) + 1) + \sqrt{d_i(s)} + m_i(s) \quad (8)$$

where $f_i(s)$ denotes the fan number of the author of post i in w_s, is the number of times that post i is forwarded in w_s, and $m_i(s)$ is the number of comments for post i in w_s. The weight of term j of the keyword vector in post i in w_s is calculated as follows:

$$w_{ij}(s) = TFIDF_{ij} + a * m(j) * v(j,s)^2 \quad (9)$$

where ($a \in [0, 1]$) is a parameter that denotes the importance of the momentum of the term.

In feature word selection, the term weight of a candidate word for the event and keyword vectors is calculated by Equations (2) and (9), respectively. All candidate words are sorted by their weight in descending order. The top L_e (L_k) words are selected as the terms of the event (keyword) vector.

3.4 Similarity computation

As previously discussed, an improved SinglePass algorithm is used in topic detection. An integration strategy that combines cosine, Jacoby, and semantic similarity is used in this algorithm to improve the accuracy of similarity computation. Suppose post $p_i = (t_{i1}, w_{i1}; t_{i2}, w_{i2};...t_{im}, w_{im})$ and topic $p_i = (t_{j1}, w_{j1}; t_{j2}, w_{j2};...t_{jm}, w_{jm})$ then the cosine similarity between the post and the topic is calculated as follows:

$$sim_{\cos}(p_i, T_j) = \frac{|p_i \cdot T_j|}{|p_i| * |T_j|} \quad (10)$$

The Jacoby similarity between the post and the topic is calculated as follows:

$$sim_{jac}(p_i, T_j) = \frac{|p_i \cap T_j|}{|p_i \cup T_j|} \quad (11)$$

where $|p_i \cap T_j|$ is the number of common features of p_i and T_i, and $|p_i \cup T_j|$ is the number of union features of and T_j.

The semantic similarity between the post and the topic is calculated as follows:

$$sim_{yuyi}(p_i, T_j) = \frac{\sum_{r=1}^{m} w_{ir} * sim_{yuyi1}(t_{ir}, T_j)}{\sum_{r=1}^{m} w_{ir}} \quad (12)$$

where w_{ir} is the weight of the r-th term of post p_i, and $sim_{yuyi}(t_{ir}, T_j)$ is the semantic similarity between term t_{ir} and T_j. $sim_{yuyi}(t_{ir}, T_j)$ is defined as follows:

$$sim_{yuyi1}(t_{ir}, T_j) = max\{sim_{yuyi2}(t_{ir}, t_{j1}),...,sim_{yuyi2}(t_{ir}, t_n)\} \quad (13)$$

where $sim_{yuyi2}(t_{ir}, t_{js})$ is the semantic similarity between t_{ir} terms and t_{js}.

In this study, the semantic similarity between the two terms is calculated based on How Net. Suppose p_1 and p_2 are the two terms, then the semantic similarity between p_1 and p_2 is defined as follows:

$$sim_{yuyi2}(p_1, p_2) = \frac{\alpha}{\alpha + d} \quad (14)$$

where α is the adjustment parameter, and d is the path length of p_1 and p_2 in How Net.

In summary, the similarity of post p_i and topic T_j is defined as follows:

$$sim(p_i, T_j) = a_1 * sim_{cos}(p_i, T_j) + a_2 * sim_{joc}(p_i, T_j) + a3 * sim_{yuyi}(p_i, T_j) \quad (15)$$

where $\alpha_1(\alpha_1 \geq 0)$, α_2 $(\alpha_2 \geq 0)$ and α_3 $(\alpha_3 \geq 0)$ are weights for cosine, Jacoby, and semantic similarities, respectively, and $\alpha_1 + \alpha_2 + \alpha_3 = 0$

3.5 Topic updating strategy

In topic detection, with the continuing addition of microblog posts, the topic vector should be updated constantly. Because of the accuracy of topic detection algorithm, some unrelated posts may be incorrectly added into the topic. An error may occur when the topic vector is updated with only the posts in the current time window. Therefore, the history topic vector is integrated in the topic updating strategy. For simplicity, only the previous topic vector is considered. Thus, the updating strategy of topic *j* in time window w_s is defined as follows:

$$T_j(s) = 0.5 * T_j(s-1) + T_{temp} \quad (16)$$

where T_{temp} is the new topic vector obtained from all posts in w_s.

3.6 DVM-based SinglePass algorithm

SinglePass is a widely used algorithm in topic detection. For a new post, clustering threshold and innovation threshold are used in SinglePass to decide whether to create a new topic or join an existing one. With DVM, an improved algorithm SinglePass_DM is proposed. The major improvements of the algorithm are as follows:

(1)In this algorithm, event vector has priority over keyword vector. If a post is similar to a topic in the event vector, then the post is included in the topic. Otherwise, the keyword vector decision will be carried out.

(2)Some structural characteristics, such as the relationships of forwarding and commenting among posts, and of attention and concern between the authors of microblog posts, are used to support the deciding of relationship between post and topic.

The SinglelPass_DM is outlined as follows.

Step 1: Post preprocessing. First, alg_filter2 filters the irrelevant posts according to the properties of irrelevant topic posts. Subsequently, the remaining posts are preprocessed with the rules described in Section 3.2. Finally, word segmentation and stopword removal are conducted using the stopword list and ICTCLAS system.

Step 2: Selecting feature words and building feature vectors. Each post is divided into the corresponding post set by its posting time according to the time window size. For each post, the entity words related to the event are selected as the candidate terms. After weight computation, the top L_e terms are selected to form the event vector of the post. Then, the weights of the candidate keywords calculated, and the top L_k terms are selected as the keyword vector terms. In this way, the vector of each post can be obtained.

Step 3: The algorithm reads the posts according to their posting time. If the post is the first one of the first time window, the algorithm creates a new topic for the post. Otherwise, the algorithm turns to step4.

Step 4: If some relationship (such as forwarding and reposting) occurs between the current post p and some processed post q, p is added into the topic where post q belongs. The algorithm then updates the topic with post p and turns to Step9. Otherwise, the algorithm turns to Step5.

Step 5: After computing the event vector similarity between post p and other existing topics, the maximum event similarity sim_e is obtained. If sim_e is higher than the event vector clustering threshold v_{ec}, p is added into the topic, and then the topic is updated with post p. The algorithm then turns to Step9. Otherwise, it turns to step6.

Step 6: After computing the keyword vector similarity between post p and other existing topics, the maximum keyword similarity sim_k is obtained. If sim_k is higher than the keyword vector clustering threshold v_{kc}, p is added into the topic, and then the topic is updated with p. The algorithm then turns to step9. Otherwise, the algorithm turns to step7.

Step 7: If sim_k is lower than the keyword vector innovation threshold v_{kn}, the algorithm creates a new topic for post p. The algorithm then turns to Step9. Otherwise, the algorithm turns to Step8.

Step 8: The algorithm adds post p to the topic t according to the keyword similarity sim_k. Suppose r is the post in topic r that corresponds to the maximum keyword similarity. If some relationship of attention or concern exists between the authors of p and r, topic t is updated with post p. Otherwise, topic t is not updated. Finally, the algorithm turns to step 9.

Step 9: If p is the last post, the algorithm ends. Otherwise, the algorithm returns to Step3.

4. Resurt Analysis And Discuttion

In this section, the SinglePass_DM algorithm is evaluated empirically. To evaluate the algorithm, the evaluation metrics used in TDD are adopted. The metrics include precision, recall rate, and F-measure. Suppose a is the number of posts in a topic correctly detected by the algorithm, b is the number of posts irrelevant to a topic incorrectly detected by the algorithm, and c is the number of posts that belong to the topic but added into other topics. Precision is computed as follows:

$$P = \frac{a}{a + b} \quad (17)$$

Recall rate is obtained as:

$$R = \frac{a}{a + c} \quad (18)$$

The F-measure is computed as:

$$F = \frac{(\alpha^2 + 1)\, PR}{\alpha^2 P + R} \quad (19)$$

In Equation (19), α is the relative weight of p and R in the F-measure. In experiments, α a = 1.

In these evaluation metrics, precision, recall rate and F-measure denote the accuracy of the topic detection algorithm from different perspectives. A higher value for these three metrics indicates higher accuracy of the algorithm.

4.1 Dataset

To validate the performance of SinglePass_DM algorithm, with Sina microblog API, a total of 178,973 posts have been collected. These posts cover 10 hot topics from May 1 to 7, 2016. In addition to the microblog posts, the dataset also includes the posting time, author, and the relationship among these posts. After filtering those irrelevant posts, 121,378 posts remained. In the experiments, the dataset is divided into D1 and D2. D1 contains posts from May 1 to 4, 2016, and D2 includes those from May 5 to 7,2016. In the experiments, the size of time window is set to one hour.

4.2 DVM evaluation

To evaluate the performance DVM in topic detection, the SinglePass and SinglePass_DM are combined with VSM and DVM model. In the experiments, the thresholds are L_e= 10 and L_k= 20. In SinglePass algorithm, clustering threshold and innovation threshold are also adopted. Experimental results in datasets D1 and D2 using these topic detection methods are shown in Figures 1 and 2.

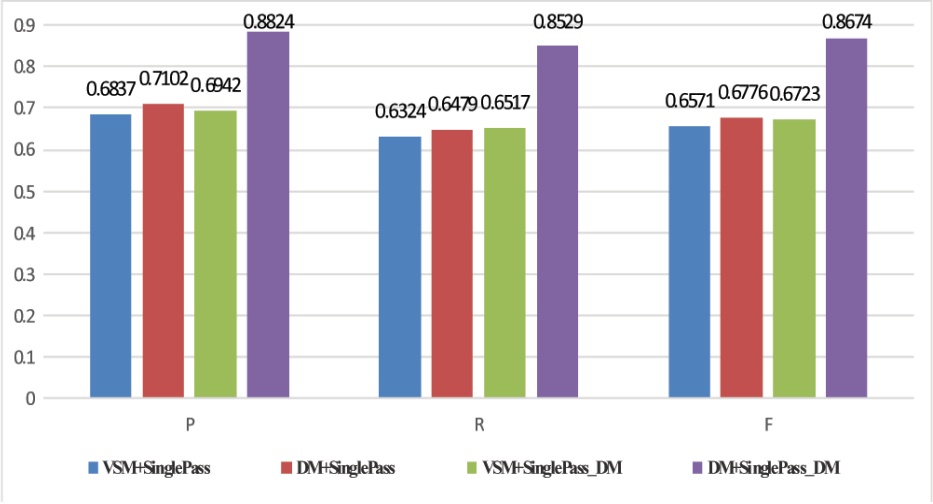

Figure 1. Performance of the two algorithms based on VSM and DVM in D1

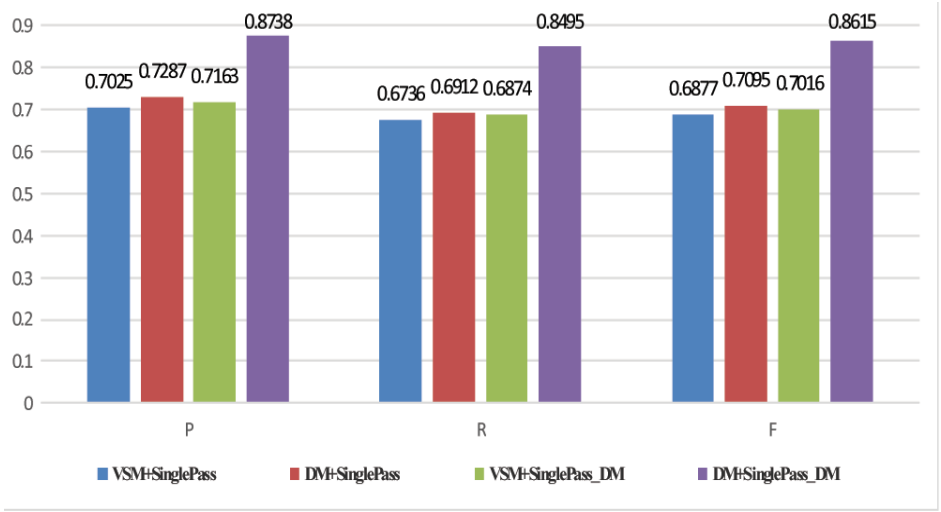

Figure 2. Performance of the two algorithms based on VSM and DVM in D2

As shown in Figures 1 and 2, the proposed method DM+ SinglePass_DM increases recall rate to 22.05% and precision to 19.87% in D1 compared with the traditional method VSM+ SinglePass; whereas the proposed method improved recall rate to 17.13% and precision to 17.59% in D2 compared with the traditional method. Thus, the introduction of DVM greatly improves the performance of SinglePass algorithm and SinglePass_DM algorithm.

4.3 Evaluation of combined similarity

To evaluate the performance of combined similarity, based on datasets D1 and D2, the performances of the cosine similarity, Jacoby similarity, semantic similarity and combined similarity are compared in SinglePass_DM algorithm. In the experiments, every situation of the parameters α_1, α_2 and α_3 in combined similarity with accuracy = 0.1, which satisfies $\alpha_1 + \alpha_2 + \alpha_3 = 1$ (α_1, α_2,α_3> 0), has been tested. The experimental results show that when (α_1, α_2,α_3) = (0.2, 0.3, 0.5) in dataset D1, SinglePass_DM gains the maximal F-measure of 0.8674; when (α_1, α_2,α_3) = (0.3, 0.3, 0.4) in dataset D2, SinglePass_DM gains the highest F-measure of 0.8615. The experimental results are shown in Figures 3 and 4.

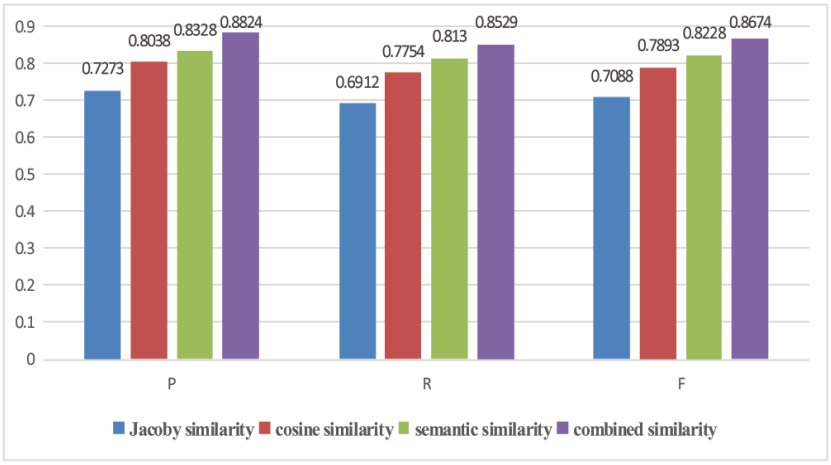

Figure 3. Performances of SinglePass_DM with different similarity in D1

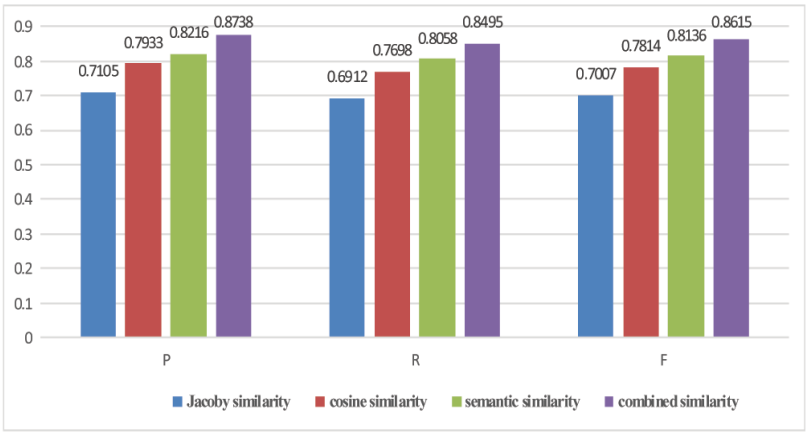

Figure 4. Performances of SinglePass_DM with different similarity in D2

Figures 3 and 4 show that the performance of the combined similarity is better than any single similarity. Among the three single similarities, semantic gained the best performance.

4.4 Evaluation of SinglePass_DM

To evaluate the performance of SinglePass_DM, three benchmark algorithms, namely, agglomerative hierarchical clustering (AHC)[27],density-based spatial clustering (DBSCAN) and SinglePass[6,7] are used. In the experiments, the average metrics of 10 topics are used as the metrics of the algorithm. The experimental results are shown in Figures 5 and 6.

Figures 5 and 6 show that, comparing with the three benchmark algorithms, SinglePass_DM increases the metrics P, R, and F in datasets D1 and D2. This phenomenon is a result of the filtering of irrelevant posts, DVM introduction, the adoption of combined similarity, and the support of the structural characteristics of the microblog. These measures increase the complexity and greatly improve the performance of the algorithm.

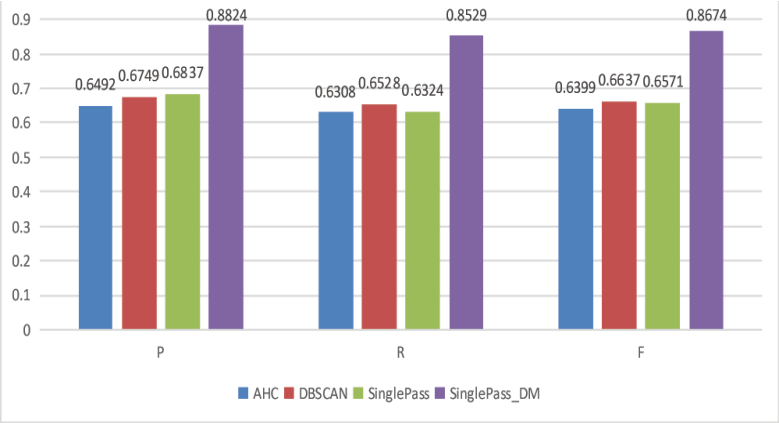

Figure 5. Performances of SinglePass_DM and other algorithms in D1

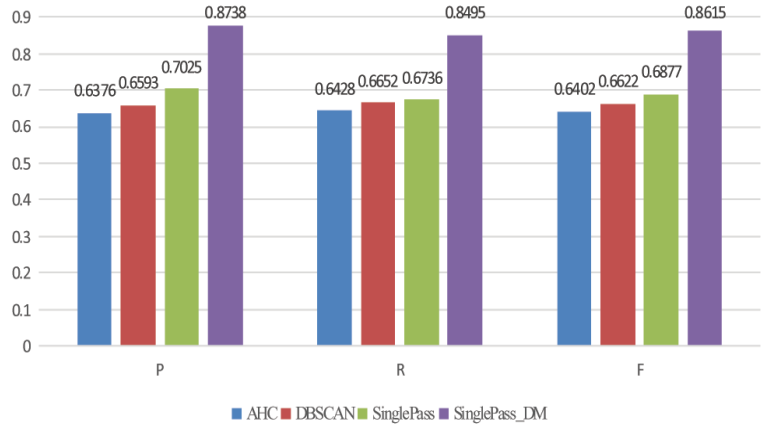

Figure 6. Performances of SinglePass_DM and other algorithms in D2

5. Conclusion

Topic detection in microblog has gained attention from researchers. However, traditional topic detection methods cannot be directly applied to microblog posts. Based on the traditional single-pass algorithm, a new algorithm called SinglePass_DM is proposed to address topic detection microblog posts. The following conclusions are drawn from the experiments.

First, DVM introduction is important to the performance of SinglePass_DM. Relevant experiments show that algorithm SinglePass_DM under DVM has higher performance than with VSM.

Second, the structural characteristics of microblog post is of major practical importance to the accuracy of SinglePass_DM. Experiments show that the accuracy of SinglePass_DM with the support of the structural characteristics of microblog is higher than other benchmark algorithms without the structural information.

To address the topic detection in microblog posts, a threshold-based SinglePass_DM algorithm is proposed. However, the performance of the algorithm is closely related to the choice of the clustering and innovation thresholds. In the experiments, the selection of the two thresholds is subjective. Further work will be performed in the future to solve the automatic selection of the two thresholds.

Acknowledgements

This work was supported in part by the opening Foundation of Shandong province key laboratory of software new technique (No. 2014HX01), Shandong province colleges and universities science and technology research Project (No. J15LN81), and the doctor Foundation of Zaozhuang university(No.1020703)

References

[1]Yu H, Yu Z, Liu T, et al. (2007). Topic detection and tracking review. *Journal of Chinese Information Processing,* 21(6) 71-87.

[2] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation.*Journal of machine Learning research,* 3(1) 993-1022.

[3] Sakaki, T., Okazaki, M., & Matsuo, Y. (2010).

Earthquake shakes Twitter users: real-time event detection by social sensors. *In*: Proceedings of the 19th international conference on World wide web (WWW 2010), p. 851-860.North Carolina, USA:ACM, April.2010.

[4] Phuvipadawat, S., Murata, T. (2010, August). Breaking news detection and tracking in Twitter. In: *Proceedings of the 2010 International Conference on Web Intelligence and ntelligent Agent Technology (WI-IAT'10)*, p. 120-123. Toronto, Canada, August 2010.

[5] Tong, W., Chen, W., & Meng, X. (2012). EDM: an efficient algorithm for event detection in microblogs. *Jisuanji Kexue yu Tansuo*, 6(12) 1076-1086.

[6] Huang, B., Yang, Y., Mahmood, A., Wang, H. (2012). Microblog topic detection based on LDA model and singlepass clustering. *In:* Proceedings of the International Conference on Rough Sets and Current Trends in Computing, p. 166-171. Berlin Heidelberg,German: Springer, August.2012.

[7] Du, Y., He, Y., Tian, Y., Chen, Q., Lin, L. (2011). Microblog bursty topic detection based on user relationship. *In:* Proceedings of the 6th IEEE Joint International Conference on Information Technolog and Artificial Intelligence (ITAIC2011),p. 260-263. Chongqing,China: IEEE Computer Society, August.2011.

[8] Jiang, H., Wang, X., Wu, Z., Zhou, M., Wang, X., Wang, J. (2013). Topic information collection based on the Hidden Markov Model. *In*: Proceedings of the 2nd International Conference on Green Communications andNetworks 2012 (GCN 2012)p. 127-136. Chongqing,China:Springer , December. 2012.

[9] Ma, B., Hong, Y., Lu, J., Yao, J., Zhu, Q. (2012). A Thread-based Two-stage Clustering Method of Micro-blog Topic Detection. *Journal of Chinese Information Processing*, 26 (6) 121-128.

[10] Yang, L., Lin, Y., Lin, H. (2012). Micro-Blog Hot Events Detection Based on Emotion Distribution. *Journal of Chinese Information Processing,* 26 (1) 84-91.

[11] Dai, X. Y., Chen, Q. C., Wang, X. L., Xu, J. (2010). Online topic detection and tracking of financial news based on hierarchical clustering. *In:* Proceedings of the 2010 International Conference on Machine Learning and Cybernetics , p. 3341-3346 .Qingdao,China: IEEE Computer Society, July.2010.

[12] Xue-Yong, L., Guo-hong, G., Jia-xia, S. (2010). A new intrusion detection method based on improved DBSCAN. *In:* Proceedings of the 2010 WASE International Conference on Information Engineering (ICIE 2010), pages 117-120. Beidaihe China: IEEE Computer Society, August 2010.