A Microblog Hot Topic Detection Algorithm Based on Discrete Particle Swarm Optimization

Huifang Ma(✉), Yugang Ji, Xiaohong Li, and Runan Zhou

College of Computer Science and Engineering, Northwest Normal University, Lanzhou, China
mahuifang@yeah.net

Abstract. Traditional hot topic detection algorithms cannot show its optimal performance on microblogs for their inherent flaws in constructing short-text representation model, implementing the core algorithm in large corpus with short time and evaluating the algorithms' qualities during the process of detecting hot topics. In this paper, a novel method for detecting hot topics in microblogs is presented. This approach takes advantage of a probabilistic correlation-based representation measure in order to ensure a dense and low-dimension microblog representation matrix. Besides, we take the clustering as an optimization problem and introduce a discrete particle swarm optimization (DPSO) to simplify the clustering process to detect topics. Furthermore, the clustering quality evaluation criteria is adopted as the optimization objective function for topic detection which can evaluate the algorithms' qualities after each iteration. Experimental results with corpora containing more than 148,000 twitters show that our algorithm is an effective hot topic detection method for microblog.

Keywords: Microblog · Hot topic detection · Probabilistic Correlation-based representation · Short-Text representation model · Discrete particle swarm optimization

1 Introduction

Over the past several years, real-time social networks, such as microblogs, have become a powerful platform where people can vent mood, share opinions, and even disseminate emerging news. Many scholars and researchers are attracted to detect underlying information as hot topics upon these online communication platforms for their perfect interactivity in users' daily life.

As for microblogs, a topic is usually defined as an event on which people focus publicly, and the hot topics are the highly condensed summary of enormous microblogs [1]. It is a challenging problem to detect microblog hot topics because microblog posts are generally much shorter. Thus, traditional topical clustering techniques based on lexical overlap are undoubtedly weak and it is too difficult to construct an effective microblog representation model by using traditional methods like Vector Space Model (VSM) [2], Latent Semantic Analysis (LSA) [3] or Latent Dirichlet Allocation (LDA) [4]. There are

R. Booth and M.-L. Zhang (Eds.): PRICAI 2016, LNAI 9810, pp. 271–282, 2016.
DOI: 10.1007/978-3-319-42911-3_23

many studies attempting to overcome the deficiencies of VSM [5, 6]. Tang et al. [7] devise an approach that enriches microblog representation by employing machine translation to increase the number of features from different languages. Cheng et al. [8] propose a coupled term-term relation model for text representation, which considers both the intra-relation and inter-relation between a pair of words, yet it is obvious that the process of calculating the relationships between words is too complex. A probabilistic correlation-based similarity measure [9] can be introduced to avoid the complex calculation.

Most of researchers prefer to use clustering method when detecting hot topics in microblogs. Since proposed by Kennedy et al. [10], the Particle Swarm Optimization (PSO) has been used in solving many clustering problems. Zhao et al. [11] discover that the PSO has the incomparable superiority in both operation time and complexity. Omran et al. [12] use PSO algorithm in image clustering. There are two types of clustering evaluation validation, one is external validation and the other is internal validation. Traditional external validations [13], such as F-measure and information entropy, are often adopted to evaluate the quality of final cluster while internal validations, the Global Silhouette (GS) coefficient and the Expected Density Measure (EDM), not only can evaluate a cluster's quality effectively, but also have the possibility to optimize the clustering result in process for their feedback characteristic. On the basis of the previous research, Leticia et al. [14, 15] presents an efficient discrete PSO approach to cluster short texts and find that GS is more suitable to be the fitness function than EDM in that algorithm.

Inspired by the observation mentioned above, we present a hot topic detection approach for microblogs based on discrete PSO. First of all, we construct a new representation model for microblogs. And then, to reduce time cost of discrete PSO, a useful method is proposed to check whether particles of different forms are the same clustering results. GS, which is set as the fitness function to optimize the clustering for a better result, can be improved by the probabilistic correlation-based similarity measure mentioned above. Finally, we compare our algorithm with other classical methods and prove its superiority.

The remainder of the paper is organized as follows. In Sect. 2, we describe our representation model for microblogs. Section 3 introduces the discrete PSO in this work. In Sect. 4, experiments are conducted to show the effectiveness of this work in different angles. Section 5 concludes and discusses future work.

2 Representation Model for Microblogs

Though detecting microblog hot topics is a new domain of computer science research, it can be viewed as an instance of mining information from numerous short texts. How to construct an effective microblog representation model is one of the most significant steps for clustering because the characteristics of microblogs may hinder the application of conventional text mining algorithms. In this section, a probabilistic correlation-based similarity measure is adopted to calculate the similarity between words. Furthermore, the intra-correlation and inter-correlation are utilized to construct the representation model and calculate the similarity value between microblogs.

2.1 Probabilistic Correlation Definition

Bag of words (BOW) is a classical model to map documents to a matrix which makes an assumption that words in texts are independent of each other, and the correlations between words are ignored. In practice, word correlations do exit and shouldn't be ignored while detecting hot topics. Considering the conditional probability of word co-occurrence, a probabilistic term correlation model is then developed.

At first, we deem that the words occurring in a microblog have latent relationships and the conditional probability adopted to model the probability of these latent relationships is defined as follows

$$Pr(w_{\mathrm{i}}|w_{\mathrm{j}}) = \frac{Pr(w_{\mathrm{i}}w_{\mathrm{j}})}{Pr(w_{\mathrm{j}})} = \frac{df(w_{\mathrm{i}}w_{\mathrm{j}})/\mathrm{N}}{df(w_{\mathrm{j}})/\mathrm{N}}, \tag{1}$$

where $Pr(w_{\mathrm{i}}w_{\mathrm{j}})$ and $df(w_{\mathrm{i}}w_{\mathrm{j}})$ respectively denote the probability and the number that words w_{i} and w_{j} occur in the same microblog while $Pr(w_{\mathrm{i}})$ and $df(w_i)$ respectively mean the probability and the number that words w_{i} appears in the a microblog, and N is the total number of microblogs in the corpora.

To ensure the similarity between microblogs symmetric, the probabilistic correlation of words is described as

$$cor(w_{\mathrm{i}}, w_{\mathrm{j}}) = Pr(w_{\mathrm{i}}|w_{\mathrm{j}}) \cdot Pr(w_{\mathrm{j}}|w_{\mathrm{i}}). \tag{2}$$

The value of $cor(w_{\mathrm{i}}, w_{\mathrm{j}})$ in range [0, 1] is proportional to the co-occurrence frequency of words w_{i} and w_{j}. When w_{i} and w_{j} appear in all microblogs that contain one of them, we have $cor(w_{\mathrm{i}}, w_{\mathrm{j}}) = 1$.

2.2 The Representation Model for Microblogs

To construct the representation model for microblogs, a microblog-word matrix $S_{N\times M}$ must be initialized at first where M is the number of different words that occur in the corpora and N denotes the total number of these microblogs. Different from traditional initialization method such as *tf-idf*, we present a novel approach by capturing both the intra-relation (explicit) and inter-relation (implicit).

Definition 1 (intra-relation). Given a microblogs mb_{i} with words $\{w_{\mathrm{i}1}, w_{\mathrm{i}2}, \cdots, w_{\mathrm{i}M}\}$, each word in mb_{i} have the intra-relation with other words in the same microblog, i.e., as is shown in Fig. 1, *love* and *flowers* are of intra-relation in mb_1.

Definition 2 (inter-relation). Given two microblogs mb_{i} with words $\{w_{\mathrm{i}1}, w_{\mathrm{i}2}, \cdots, w_{\mathrm{i}\mathrm{M}}\}$ and mb_{n} with words $\{w_{\mathrm{n}1}, w_{\mathrm{n}2}, \cdots, w_{\mathrm{n}\mathrm{M}}\}$, if there are one or more words appear both mb_{i} and mb_{n}, these words are called linking words and each word in mb_{i} have inter-relation with the words in mb_{n}. I.e., *flowers* and *chocolates* have the inter-relation and *love* is the link word in Fig. 1.

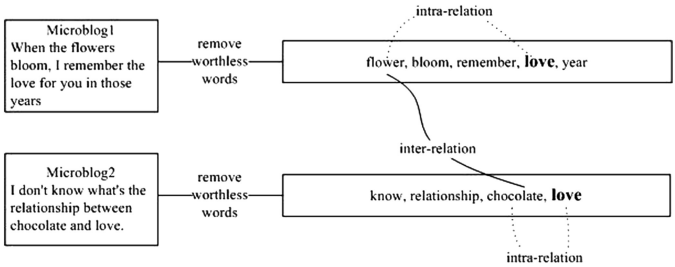

Fig. 1. One specific example of intra-relation and inter-relation, *love* is the link word.

The intra-relation between two words in a microblog is given by:

$$IaR(w_{\mathrm{i}}, w_{\mathrm{k}}) = \begin{cases} 1, & i = k \\ \frac{cor(w_{\mathrm{i}}, w_{\mathrm{k}})}{\sum_{j=1, j\neq k}^{m} cor(w_{\mathrm{j}}, w_{\mathrm{k}})}, & i \neq k \end{cases}, \tag{3}$$

where m is the number of different words in the corpora, and it is easy to derive that $\sum_{i=1, i\neq k}^{m} IaR(w_{\mathrm{i}}, w_{\mathrm{k}}) = 1$ when $i \neq k$. Note that all words mentioned in the context belong to the dictionary based on the corpora, in other words, these words are different from each other which ensures that each word can be identified by its subscript.

However, Eq. 3 can only capture the co-occurrence frequency between w_{i} and w_{k}, while many words are also closely related though they don't co-occur in the same microblog. Therefore, we define inter-relation as

$$IeR(w_{\mathrm{i}}, w_{\mathrm{j}}) = \begin{cases} 0, & i = j \\ \frac{\sum_{\forall w_k \in L} \min\{IaR(w_{\mathrm{i}}, w_{\mathrm{k}}), IaR(w_{\mathrm{j}}, w_{\mathrm{k}})\}}{|\mathrm{L}|}, & i \neq j \end{cases}, \tag{4}$$

where $|\mathbf{L}|$ is the amount of linked words set $\mathbf{L}$, i.e., $|\mathbf{L}| = 1$ and $\mathbf{L} = \{love\}$ in Fig. 1. We let $IeR(w_{\mathrm{i}}, w_{\mathrm{j}}) = 0$ when $i = j$ since it is worth nothing to calculate the inter-relation between a word itself.

Taking both the intra-relation and inter-relation into consideration, the correlation between w_{i} and w_{j} can be defined as

$$WR(w_{\mathrm{i}}, w_{\mathrm{j}}) = \begin{cases} 1, & i = j \\ \alpha \times IeR(w_{\mathrm{i}}, w_{\mathrm{j}}) + (1 - \alpha) \times IaR(w_{\mathrm{i}}, w_{\mathrm{j}}), & i \neq j \end{cases}, \tag{5}$$

where WR is the abbreviation for *word relation*, and $\alpha \in [0, 1]$ determines the importance of intra-relation and inter-relation between w_{i} and w_{j}. It is easy to prove that $WR(w_{\mathrm{i}}, w_{\mathrm{j}}) = 1$ if $i = j$.

Instead of term frequency (*tf*) with $weight_{\mathrm{i}} = 1$ in most cases, a new local weighting scheme of words in a microblog is proposed in this approach, namely correlation weight.

Definition 3 (correlation weight). Given a microblog mb_{n} with an initial $weight_{\mathrm{ni}}$ of each word w_{ni}, the correlation weight of w_{ni} in mb_{n} is defined as

$$cow(w_{\mathrm{ni}}) = weight_{\mathrm{ni}} + \frac{\sum_{w_{\mathrm{nj}} \in mb_{\mathrm{n}}} weight_{\mathrm{nj}} \cdot WR\left(w_{\mathrm{ni}}, w_{\mathrm{nj}}\right)}{|mb_{\mathrm{n}}|}, \tag{6}$$

where $WR\left(w_{\mathrm{ni}}, w_{\mathrm{nj}}\right)$ is the word relation between w_{ni} and w_{nj} in this microblog, $|mb_{\mathrm{n}}|$ is the total number of words in mb_{n}.

Finally, a new representation model for microblog is proposed and each element rm_{ij} in the matrix $\mathbf{RM}_{\mathrm{N}\times\mathrm{M}}$ is defined as

$$rm_{\mathrm{ij}} = cow(w_{\mathrm{ij}}) \cdot idf(w_{\mathrm{ij}}), \tag{7}$$

where $i \in \{1, 2, \cdots, \mathrm{N}\}$ is the subscript of microblogs in the corpora, $j \in \{1, 2, \cdots, \mathrm{M}\}$ is the subscript of words in the dictionary, $idf\left(w_{\mathrm{ij}}\right) = \log\left(\frac{\mathrm{N}}{df(w_{\mathrm{ij}})}\right)$, N is the amount of microblogs and M is the length of words in this dictionary.

With the advantages of the conditional probability between words and inner/inter relation, the new matrix $\mathbf{RM}_{\mathrm{N}\times\mathrm{M}}$ can reflect the relationship between any two words, and is proved denser, lower-dimensional than tradition models in the experimental section. Taking advantages of the inter-relation and inner-relation between words, the similarity function of microblogs is defined as follows [10]:

$$sim\left(mb_{\mathrm{x}}, mb_{\mathrm{y}}\right) = \frac{\sum_{(w_{\mathrm{i}}, w_{\mathrm{j}}) \in \mathbf{D}} weight_{\mathrm{i}} weight_{\mathrm{j}} cow(w_{\mathrm{i}}, w_{\mathrm{j}})}{\left\|mb_{\mathrm{x}} \oplus \mathbf{D}\right\| \cdot \left\|mb_{\mathrm{y}} \oplus \mathbf{D}\right\|}, \tag{8}$$

where $\mathbf{D}$ denotes the words correlation of mb_{x} and mb_{y}, and $\left\|mb_{\mathrm{x}} \oplus \mathbf{D}\right\|$, $\left\|mb_{\mathrm{y}} \oplus \mathbf{D}\right\|$ denote the sizes of mb_{x} and mb_{y} so as to normalize the similarity value.

$$\left\|mb_{\mathrm{x}} \oplus \mathbf{D}\right\| = \sqrt{\sum_{(w_{\mathrm{i}}, w_{\mathrm{j}}) \in \mathbf{D}} \left(weight_{\mathrm{i}}^2 cow(w_{\mathrm{i}}, w_{\mathrm{j}})\right) + \sum_{w_{\mathrm{i}} \in mb_{\mathrm{x}} \setminus mb_{\mathrm{y}}} weight_{\mathrm{i}}^2}, \tag{9}$$

and $\left\|mb_{\mathrm{y}} \oplus \mathbf{D}\right\|$ can be calculate in a similar way.

Thus, $sim\left(mb_{\mathrm{x}}, mb_{\mathrm{y}}\right) \in [0, 1]$.

3 The DPSO Algorithm for Microblog Hot Topic Detection

Particle Swarm Optimization (PSO) is a population-based search algorithm inspired by the behavior of biological communities that exhibit both individual and social behavior; examples of these communities are flocks of birds, swarms of bees. In PSO, each solution to the problem at hand is called a particle and per particle represents a real vector within the search space, corresponding to a solution of the mazy problem.

However, traditional PSO is originally developed for continuous space but many problems are defined for discrete valued spaces where the domain of the variables is

finite. We propose a discrete PSO approach and fit it to clustering microblogs for detecting hot topics in this section. The fitness function of PSO is redefined by GS and the correlation similarity of microblogs. Finally, a valid method to reduce time cost of the algorithm by dealing with particles is also shown then.

3.1 DPSO Algorithm

The basic PSO algorithm contains a swarm of particles in which each particle includes a potential solution. The particles fly through a multi-dimensional search space where the position of each particle is adjusted according to its own experience and the experience of its neighbors during per iteration. In each iteration, the velocity and the position of every particle are calculated by Eqs. 10 and 11. Besides, the current global best position (*gbest*) and individual history best position (*pbest*) are all recorded in the corresponding matrices.

$$v_{id} \leftarrow \omega(v_{id} + \gamma_1(pbest_{\mathrm{id}} - par_{\mathrm{id}}) + \gamma_2(gbest_{\mathrm{d}} - par_{\mathrm{id}})), \tag{10}$$

$$par_{\mathrm{id}} \leftarrow (par_{\mathrm{id}} + v_{id}), \tag{11}$$

where ω is the inertia factor whose goal is to balance global exploration and local exploitation, γ_1 is the personal learning factor, γ_2 is the social learning factor, par_{id}, $pbest_{\mathrm{id}}$, and v_{id} are the position, history best position, and velocity of i^{th} particle in d^{th} dimension respectively, and $gbest_{\mathrm{d}}$ is the best position of the whole swarm in d^{th} dimension. In our context, these concepts are redefined in Definitions 4, 5 and 6 to make it easier to understand and calculate.

DPSO (Discrete Particle Swarm Optimization) is a discrete version of the basic PSO algorithm. In this method, each particle represents a clustering result of the corpora during the process of hot topics detection as the final aim of our algorithm is to get an excellent clustering result.

To detail the process of DPSO, we define four matrices as follows

Definition 4 (particles cluster result matrix). The particles cluster result matrix, $\mathbf{P}_{\mathrm{K}\times\mathrm{N}}$, is defined to store the current position of each particle and each element, p_{id}, is the cluster of i^{th} particle in d^{th} dimension. i.e., $p_{12} = \mathrm{cluster1}$ means the first particle of the swarm in second dimension is clustered to cluster1. In addition, K is the number of particles in this paper and is set manually.

Definition 5 (particles cluster quality matrix). The particles cluster quality matrix, $\mathbf{PAR}_{\mathrm{K}\times\mathrm{N}}$, is proposed to record the quality of **P**, and each element, par_{id} is calculated by the fitness function mentioned below. By comparing the value of par_{id} and $pbest_{\mathrm{id}}$ and $gbest_{\mathrm{d}}$, we can judge whether it is worth putting the corresponding element in **P** to a new cluster.

Definition 6 (particles cluster velocity matrix). The particles cluster velocity matrix, $\mathbf{V}_{\mathrm{K}\times\mathrm{N}}$, is provided to represent the probability of choosing the corresponding particle to optimize. The value of per element, v_{id}, can be changed during each iteration of optimization by Eq. 8.

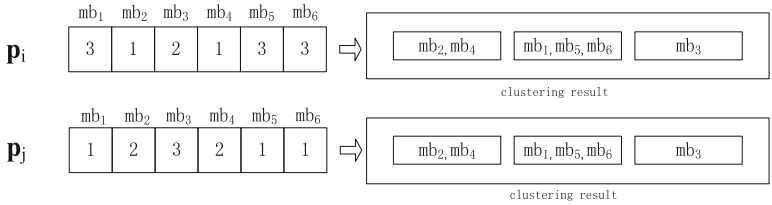

Fig. 2. The situation that different particles represent same clustering result.

In DPSO algorithm, Eq. 11 is modified as

$$par_{\mathrm{id}} \leftarrow pbest_{\mathrm{id}}, \tag{12}$$

During the initialization of **P**, there is a high probability that different particles may refer to the same clustering result as shown in Fig. 2.

In order to avoid unnecessary time cost of the iterative process, we propose an effective method as follows.

Program.1. The algorithm for checking repeated clustering results.

Input: $\mathbf{P} = \bigcup_{i=1}^{k}(p_{i1}, p_{i2}, \ldots, p_{iN})$.
Output: the updated $\mathbf{P} = \bigcup_{i=1}^{k}(p'_{i1}, p'_{i2}, \ldots, p'_{iN})$.
1. An integer set clusters $= \bigcup_{i=1}^{l} i$, the preset clusters's amount l,
   $\mathbf{Q}_{k\times m} = \bigcup_{i=1}^{k}(q_{i1}, q_{i2}, \ldots, q_{iN}) = \mathrm{null}$, count = 0.
2. for every p_{id} in **P**
3. if $p_{\mathrm{id}} \notin \bigcup_{r=1}^{d-1} p_{ir}$
4. q_{id} = clusters[count];
5. count++;
6. else
7. catch t $\in$[1,d-1] to ensure $p_{\mathrm{id}} = p_{\mathrm{it}}$;
8. $q_{\mathrm{id}} = q_{\mathrm{it}}$;
9. end if
10. end for
11. for every $\mathbf{Q}_i$ in **Q**
12. if $\mathbf{Q}_i = \mathbf{Q}_j$ and $i < j$
13. $\mathbf{P}_j = \mathbf{P}_i$;
14. end if
15. end for
16. Output the updated **P**.

By using Program 1, $\mathbf{P}_i$ and $\mathbf{P}_j$ in Fig. 2 are in the same form of {1,2,3,2,1,1} or {a, b,c,b,a,a} etc.

3.2 The Improved Fitness Function

The fitness function in PSO is usually used to measure the quality of particles. In other words, it is a method to judge whether the corresponding cluster result is a better one during clustering.

As an external validation used to deal with the corpora of unknown structure and evaluate the clustering quality based on the corpora only, the Global Silhouette coefficient (GS) is a good choice to be the fitness function since it provides a succinct graphical representation of how well each object lies within its cluster. Assuming the corpora have been clustered into l clusters. For each microblog mb_{i}, let $a(mb_{\mathrm{i}})$ be the average dissimilarity between mb_{i} and all other microblogs within the same cluster, which can be interpreted as how well mb_{i} is assigned to its cluster. Let $b(mb_{\mathrm{i}})$ be the average dissimilarity between mb_{i} and the microblogs in the neighboring cluster of mb_{i}. The formula to calculate the GS value is given by

$$GS(mb_{\mathrm{i}}) = \frac{b(mb_{\mathrm{i}}) - a(mb_{\mathrm{i}})}{\max\{a(mb_{\mathrm{i}}), b(mb_{\mathrm{i}})\}}, \tag{13}$$

with $-1 \leq GS(mb_i) \leq 1$. From this formula it can be observed that negative values for this measure are undesirable and that for this coefficient values as close to 1 as possible are desirable.

Given a mb_{i} belonging to cluster C_{a}, and the neighboring cluster C_{b}, the function to compute $a(mb_{\mathrm{i}})$ is defined as follows

$$a(mb_{\mathrm{i}}) = \frac{\sum_{mb_{\mathrm{j}} \in C_{\mathrm{a}}} (1 - sim(mb_{\mathrm{i}}, mb_{\mathrm{j}}))}{|C_{\mathrm{a}}|}, \tag{14}$$

and $b(mb_{\mathrm{i}})$ is defined as

$$b(mb_{\mathrm{i}}) = \frac{\sum_{mb_{\mathrm{j}} \in C_{\mathrm{b}}} (1 - sim(mb_{\mathrm{i}}, mb_{\mathrm{j}}))}{|C_{\mathrm{b}}|}, \tag{15}$$

where $|C_{\mathrm{a}}|$ and $|C_{\mathrm{b}}|$ are the amount of microblogs in cluster C_{a} and C_{b} respectively.

3.3 The Algorithm's Framework and Details

Combined with correlations (intra/inter relation) of words, a clustering algorithm based on DPSO is proposed to detect microblog hot topics. The algorithm is mainly divided into three steps: constructing a microblog representation model by the conditional probability of word co-occurrence and correlations of words; using DPSO algorithm to initialize and optimize the cluster result of microblogs; judging whether a cluster result need to be optimized and whether the algorithm should end by calculating the corresponding GS (Fig. 3).

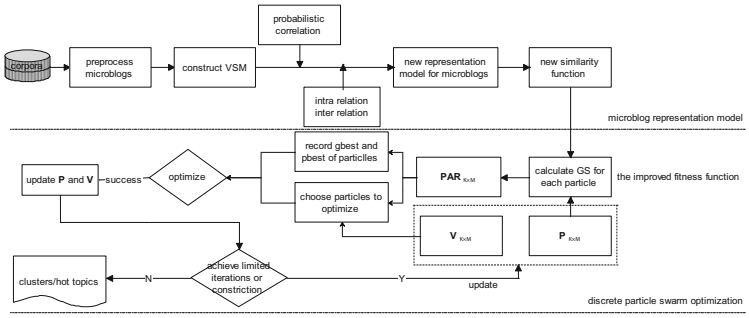

Fig. 3. The algorithm's framework.

Note that $\boldsymbol{V}$ is initialized by a random value in range [0,1] where each element indicates the probability of choosing the corresponding particles to optimize, and **P** is initialized by the random cluster subscripts at the beginning of DPSO.

4 Experiments

In this section, we report our experimental results. Section 4.1 introduces the datasets, parameter settings and effectiveness criteria in the approach. Section 4.2 evaluates the performance of our approach with various parameters and compare our method with existing approaches such as traditional k-means, and MicroBlog Hierarchical Dirichlet Process (MB-HDP) [16] and a topic detection method based on microblog weight named Weighted LDA (W-LDA) [17].

4.1 Data Sets, Parameter Settings and Effectiveness Criteria

Datasets. We grab the first one hundred search results by Twitter API from March 1st 2015 to Oct. 31st 2015. After removing those invalid twitters and those too short twitters, meanwhile merging the same content, the remaining data set is 148090 microblogs in total.

Parameter Settings. In our experiments, 50 independent runs are performed, with 10,000 iterations per run, the swarm size K = 100 particles, dimensions of each particle N = number of twitters, inertia factor ω = 0.9, personal and social learning factors $\gamma 1$, $\gamma 2$ are set to 1.0, and the clusters number l = 10.

Effectiveness Criteria. Two evaluative metrics, the Purity and F1-measure are used to evaluate experiments performance. Purity is a simple and transparent evaluation measure for cluster quality while F1-measure, is a weighted harmonic mean of Recall and Precision (R, P).

4.2 Experimental Results and Analysis

In this section, experimental results concerning our algorithm are discussed, such as the influence of regulatory factor α to determine the importance of intra-relation and inter-relation between words, the size of microblogs to deal with, the number of iterations of the algorithm and the effectiveness of the new similarity function. Since our algorithm has some parameters to be tuned, all the involved parameters are carefully tuned and the parameters with best performance are used to report the final results. And then, we make the comparison among our algorithm and other hot topic detection methods such as W-LDA, traditional DPSO and MB-HD.

Figure 4(1) shows *F1-measure* values, *purity* values and GS values with different regulatory factor α. Obviously, our algorithm shows its best performance while α is in the vicinity of 0.5 which indicates the need for a balance between the intra-relation and inter-relation.

Figure 4(2) shows *F1-measure* values, *purity* values and GS values with the increasing size of iterations for optimizing particles. when the number of iteration is more than 10000, the performance of our algorithm becomes stable.

Figure 4(3) shows *F1-measure* values, *purity* values and GS values with different particles' number. Each horizontal coordinate denotes the corresponding times the number of microblogs in the corpora.

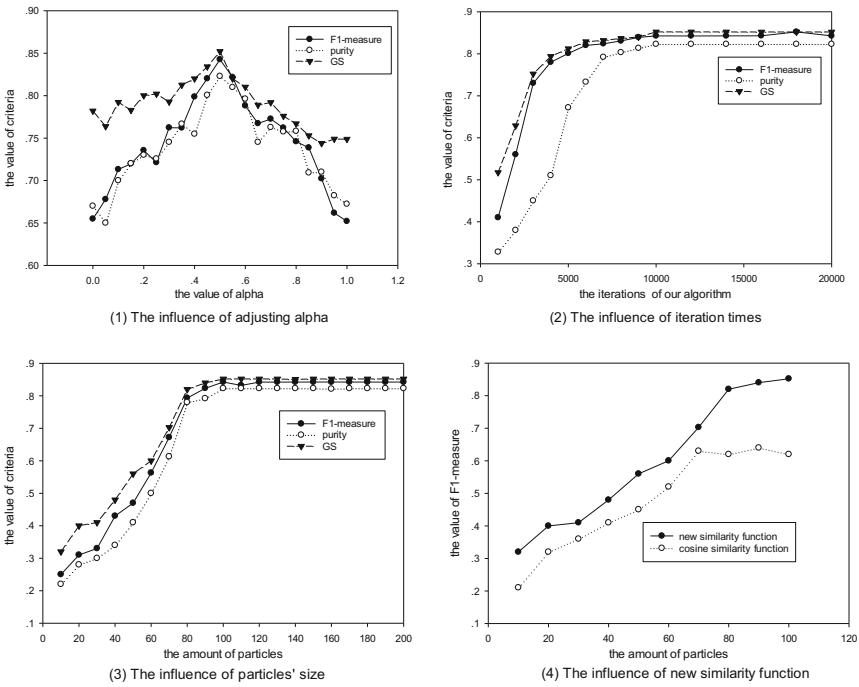

Fig. 4. The influence of adjusting key parameters and the advantage of using the new similarity function based on intra/inter relation.

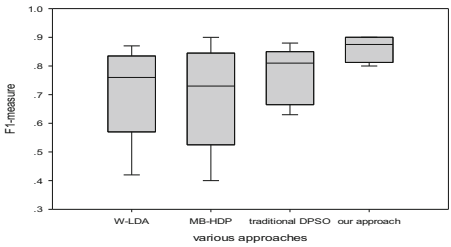

Fig. 5. The comparison of various topic detection approaches.

Figure 4(4) shows *F1-measure* values with the new similarity function and cosine similarity function respectively. The new similarity function shows a better performance for considering the correlation and simplifying the steps of calculating GS which can reduce the algorithm's time cost.

In summary, the above experimental results demonstrate that intra/inter relation, the similarity function play significant roles in this algorithm, which indicates that these aspects should not be ignored.

For effectiveness, we then compare our methods with several existing methods. As we can see, overall, our method outperform all the compared methods. The box plots in Fig. 5 represent that our approach reach the best performance with a median value close to 0.86 and without dispersion except for some outliers. This aspect is important because it shows that our approach is able to find very similar F-measure values in all the runs. The worst one is MB-HDP because its median value is the lowest and its box plot shows the highest dispersion. Both of MB-HDP and W-LDA are based on LDA, which cannot be well generalized. Compared with traditional DPSO, our algorithm present a representational model for microblogs and improve the fitness function which ensure a novel optimization.

5 Conclusion

Hot topic detection for microblogs is a challenging research problem in data mining domain. Different from traditional documents or texts, microblogs are often very short, sparse and spreading rapidly online. In this work, we propose an effective algorithm based on discrete particle swarm optimization. A representation model for microblogs considering correlations and intra/inter relations between words by calculating conditional probability of word-occurrences is constructed to capture the semantic associations between words. In addition, a new method to compute the similarity between microblogs is proposed so as to improve the fitness function, GS. Furthermore, DPSO algorithm is developed to obtain the hot topics in the corpora. Besides, an effective program to overcome the problem that various particles denotes the same clustering result. Finally, experiments have demonstrated its effectiveness in mining microblogs. The future research can be targeted at particle dimension reduction, a more suitable fitness function selection.

Acknowledgement. This work is supported by the National Natural Science Foundation of China (No.61363058), Youth Science and technology support program of Gansu Province (145RJZA232, 145RJYA259), 2016 undergraduate innovation capacity enhancement program and 2016 annual public record open space Fund Project 1505JTCA007.

References

1. Ding, Z.Y., Jia, Y., Zhou, B.: Survey of data mining for micro-blogs. J. Comput. Res. Dev. **04**, 691–706 (2014)
2. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975)
3. Deerwester, S., Dumais, S.T., Furnas, G.W., et al.: Indexing by latent semantic analysis. J. Am. Soc. Inf. Sci. **41**(6), 391 (1990)
4. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. the. J. Mach. Learn. Res. **3**, 993–1022 (2003)
5. Ma, H.F., Zhao, W.Z., Shi, Z.Z.: A nonnegative matrix factorization framework for semi-supervised document clustering with dual constraints. Knowl. Inf. Syst. **36**(3), 629–651 (2013)
6. Ma, H., Jia, M., Xie, M., Lin, X.: A microblog recommendation algorithm based on multi-tag correlation. In: Zhang, S., et al. (eds.) KSEM 2015. LNCS, vol. 9403, pp. 483–488. Springer, Heidelberg (2015). doi:10.1007/978-3-319-25159-2_43
7. Tang, J., Wang, X., Gao, H., et al.: Enriching short text representation in microblog for clustering. Front. Comput. Sci. **6**(1), 88–101 (2012)
8. Cheng, X., Miao, D., Wang, C., et al.: Coupled term-term relation analysis for document clustering. In: The 2013 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2013)
9. Song, S., Zhu, H., Chen, L.: Probabilistic correlation-based similarity measure on text records. Inf. Sci. **289**, 8–24 (2014)
10. Kenndy, J., Eberhart, R.C.: Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948 (1995)
11. Zhao, X.C., Liu, G.L., Liu, H.Q., et al.: Particle swarm optimization algorithm based on non-uniform mutation and multiple stages perturbation. Chin. J. Comput. **9**, 2058–2070 (2014)
12. Omran, M., Engelbrecht, A.P., Salman, A.: Particle swarm optimization method for image clustering. Int. J. Pattern Recogn. Artif. Intell. **19**(03), 297–321 (2005)
13. Zhang, W.J., Liu, C.H., Li, F.Y.: Method of quality evaluation for clustering. J. Comput. Eng. **31**(20), 10–12 (2005)
14. Cagnina, L.C., Errecalde, M.L., Ingaramo, D.A., et al.: An efficient particle swarm optimization approach to cluster short texts. Inf. Sci. **265**, 36–49 (2014)
15. Cagnina, L.C., Errecalde, M.L., Ingaramo, D.A., et al.: A discrete particle swarm optimizer for clustering short-text corpora. In: Proceedings of the Bioinspired Optimization Methods and their Applications, BIOMA-2008, Ljubljana, Slovenia (2008)
16. Liu, S.P., Yin, J., Ouyang, J., et al.: Topic mining from microblogs based on MB-HDP model. Chin. J. Comput. **7**(008), 1408–1419 (2015)
17. Guo, K., Shi, L.: A topic detection method based on microblog weight. In: 2015 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), pp. 209–212. IEEE (2015)