ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)
https://doi.org/10.17559/TV-20210610120531
Preliminary communication

A Hotspot Discovery Method Based on Improved FIHC Clustering Algorithm

Lina LIN, Dezhi WEI*

Abstract: It was difficult to find the microblog hotspot because the characteristics of microblog were short, rapid, change and so on. A microblog hotspot detection method based on MFIHC and TOPSIS was proposed in order to solve the problem. Firstly, the calculation of HowNet similarity was used in the score function of FIHC, the semantic links between frequent words were considered, and the initial clusters based on frequent words were produced more accurately. Then the initial cluster of the text repletion of mircoblog was reduced, and the idea of Single-Pass clustering was used to the reduced topic cluster in order to get the Hotspot. At last, an improved TOPSIS model was used to sort the hot topics in order to get the rank of the hot topics. Compared with the other text clustering algorithms and hotspot detection methods, the method has good effect, and can be a more comprehensive response to the current hot topics.

Keywords: clustering; hotspot detection; internet public opinion; microblog; TOPSIS

1 INTRODUCTION

As a new, open and fast network media, Microblog has been widely used by the netizens. The spread of micro-blog text is controlled at about 140 words, so there is a big difference between the texts of microblog and the texts of traditional news. Anybody can spread original information through "mobile phone + microblog", which is fast and interactive. Scholars have carried out a series of studies on the spread of microblog information and the discovery of hot spots, and have got some research results.

Du et al. [1] proposed a discrete particle swarm algorithm by combining the term association and particle swarm. This algorithm took advantage of the random search ability of intelligent algorithm, simplified the traditional clustering algorithm, and could discover the hot spot information of microblog quickly and effectively. Lu and Zheng [2] designed a clustering algorithm of microblog hotspots based on time series and semantics according to the idea of FIHC clustering. Frequent terms were used to cluster twice, and the tracking and detection of microblog topics was effectively realized in this method. Zhang et al. [3] proposed TCMLPA clustering algorithm to cluster the hot words of microblog and get hot topics. Extraction of hotspot words and the efficiency of clustering time were solved, and the accuracy of hotspot discovery was improved in this method. Fan et al., and Li et al., respectively, improved the single-Pass clustering algorithm, solved the problems of the original algorithm, and effectively improved the discovery efficiency of microblog hotspot [4, 5]. Yi et al., and some other scholars used different clustering algorithms and simulation mathematical models, such as firefly clustering algorithm, OLDA model, SEPPM model, impulse time series behavior dynamic model, to discover and simulate the network hot events, and got some results [6-18].

However, the recent studies show that the main way to discover hot topics on micro-blog is to use clustering algorithm to cluster text content [19, 20]. The traditional clustering algorithms mainly focus on large and traditional text, while the texts of microblog are very short, trendy, and network language. At the same time, microblog has a large number of information, so there are some problems in directly applying traditional clustering to short text clustering of microblog. At present, there are two main research solutions, one is to extend the semantics of short text, to mine similar semantic texts, and to improve the information content of text semantics, and the other is to mine the word order of short text, which mainly combines with frequent terms.

Therefore, this paper proposes an improved FIHC clustering algorithm based on HowNet semantic similarity and topic cluster similarity calculation method in order to solve the problems of short text clustering in microblog, which can effectively find hot topics. At the same time, most clustering algorithms failed to make further analysis of hot topics after finding hot topics. Therefore, this paper extracts the relevant factors affecting the hot topics in the data pre-processing process, and uses the improved TOPSIS model to sort the hot topics, so as to better analyze the relevant information and ranking.

This paper is organized as following. We depict related work to our researcher in Section 1, and discuss the adaptive model system for intrusion detection in detail in Section 3. Besides, we also report our experiment in section 4 and conclude our work in section 5.

2 MODEL STRUCTURE

Microblog texts are short and sparse, so an improved MFIHC clustering method is proposed in order to solve the problems of text clustering in microblog. This method uses frequent terms and the semantic calculation method of HowNet to establish the initial cluster and further eliminate the overlapping cluster text, then cluster to get hot topics. Finally, the TOPSIS model is used to get the ranking of hot topics of clustering. The whole process of the method is shown in Fig. 1.

In Fig. 1, data cleaning is carried out after the collection of microblog text data. In addition to acquiring the relevant content of microblog texts, it is also necessary to collect and store the relevant factors such as "click-to-read amount", "comment number", "forwarding amount" and "collection amount" which can express the heat of microblog texts. Data cleaning mainly adopts the method of manual identification, and the specific principles follow the accuracy, integrity, consistency, uniqueness, timeliness and effectiveness of data to deal with the problems of missing value, cross-border value, inconsistent code, duplicate data and so on. The data format after cleaning is: body content, click reading, comments, forwarding and collection. The semantic database of HowNet is introduced to allocate microblog texts in the process of constructing the initial topic cluster, and the semantic similarity of

words with frequent items is calculated to improve the accuracy of the initial topic cluster. Then, the Single-Pass clustering algorithm is used to calculate the similarity of topic clusters after eliminating duplication, the Semantic Library of HowNet is also introduced to calculate the similarity of the whole cluster, and finally the relevant hot topic clusters are got. Finally, the TOPSIS model is used to analyze the topic clusters of microblog and the other hot data of related microblog texts.

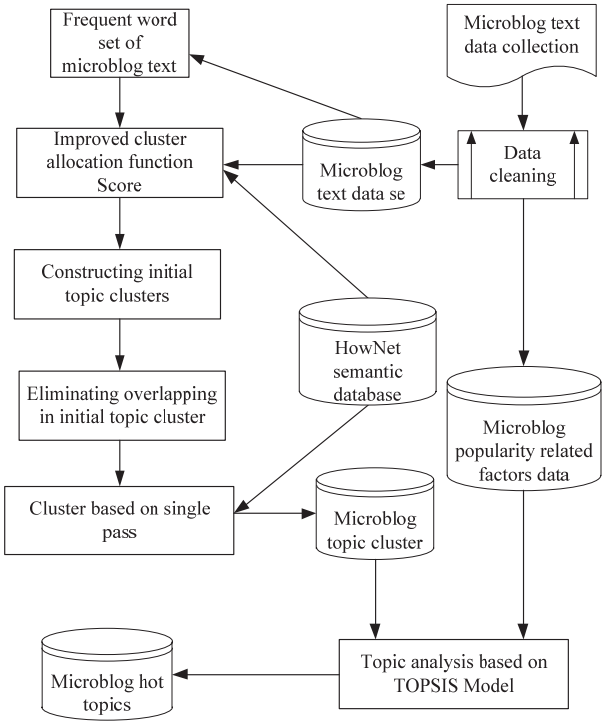

Figure 1 Model structure of the discovery method of hot topic in microblog

3 MFIHC CLUSTERING ALGORITHM FOR THE CLUSTERING OF MICROBLOG TEXTS

FIHC clustering algorithm is a method based on mining frequent words from texts, building clusters according to frequent words, eliminating duplicate clusters, and finally clustering. Information of microblog texts is short and sparse. If FIHC clustering algorithm is directly applied to the clustering of microblog texts, there will be some problems. The main problem is that frequent items do not contain the implicit semantic relationship between words, but only the number of terms, which affects the division of initial clusters and the efficiency of subsequent hierarchical clustering. Therefore, in this paper the FIHC clustering algorithm is improved and a MFIHC clustering method is proposed to better adapt to the clustering microblog texts.

The general steps of MFICH clustering algorithm are as follows:

(1) The initial cluster of topics is determined. Frequent items of microblog text are mined by FP growth algorithm. Frequent items represent the core features of the initial cluster. Frequent items are used as semantic tags of the initial cluster.

(2) The number of repeated topic clusters is reduced. The specific process is as follows: firstly, the initial topic cluster is scanned to obtain the repeated initial cluster and related repeated microblog text, then only the repeated microblog text is calculated, the text is classified into the cluster with the largest value, and the text of other clusters is deleted at the same time.

(3) Topic clusters are clustered. Based on the idea of single pass, the related topic clusters are analyzed, and then the related topic clusters are merged to get more accurate hot topics.

3.1 Mining Frequent Terms in Microblog Texts

Each microblog text contained in the dataset of microblog texts is recorded as D_1, D_2, …, D_m, each microblog text consisting of a collection of terms related to the text is recorded as T_i, $T = \{T_1, T_2, \ldots, T_m\}$.

Definition 1: The dataset of microblog texts recorded as D contains a set of terms recorded as X. If the number of occurrences of X in D reaches a proportion, that is $\sup(x) \geq$ minsup, the global frequent terms defined X to be part of D, and this ratio is defined as the global minimum support recorded as minsup.

Definition 2: Text Data D is divided into several clusters, the term Y is a word in a cluster of clusters C. If the number of times Yappears Cin reaches a ratio, that is $\sup(y) \geq$ cminsup, then Y is defined as the frequent item of text cluster C, cminsup is the minimum cluster support.

According to the above definitions, frequent terms of microblog text were mined. At present, related algorithms of association rule can be used to mine frequent terms. The Apriori algorithm and FP-Growth algorithm are classic methods. FP-Growth algorithm only needs to scan the database twice, which effectively improves the efficiency of the algorithm. Therefore, FP growth algorithm is adopted in this paper to my frequent terms of microblog text. The specific implementation process is as follows: (1) According to the minimum global support, the text data set of microblog was scanned to obtain the list of frequent terms; (2) All the frequent patterns were found by the FP-Tree according to the list.

Frequent words of microblog text were mined by FP-growth algorithm, which could basically represent the core features of the initial cluster. Therefore, frequent words could be used as the semantic label of the initial topic cluster.

3.2 A Method of Reducing the Repetition between Clusters Based on Semantic Similarity of Howne

The final purpose of the clustering of microblog texts is to cluster each micro blog text into a topic cluster. There will be a phenomenon that a micro blog text appears repeatedly in different topic clusters in the initial topic clusters. Therefore, we should reduce overlapped parts and reasonably divide the initial topic clusters according to the characteristics of the short micro blog texts.

Definition 3: If the microblog text doc_j belongs to a cluster C_i, it is recorded as $C_i \leftarrow \text{doc}_j$.

Definition 4: The dataset of microblog texts D_i belongs to a cluster C_i, D_j belongs to a cluster C_j, D_i and D_j share the same microblog text, if $D_i \cap D_j \neq \varphi$, There is overlap between C_i and D_j.

Definition 5: The partition function of the microblog texts Score which is used to determine whether the

microblog texts are assigned to a cluster. A new calculation method based on HowNet's semantic awareness calculation was proposed to better solve the semantic and sparse features of the short microblog texts. The specific calculation formula is defined as Eq. (1).

$$\begin{aligned}&\mathrm{Score}(C_i \leftarrow \mathrm{doc}_j) = (1-\lambda)\\&*(\sum_x n(x) * \mathrm{c\_sup}(x) - \sum_{x'} n(x') * \mathrm{g\_sup}(x'))\\&+\lambda * \mathrm{Sim}(X,Y)\end{aligned} \tag{1}$$

$\mathrm{Score}(C_i \leftarrow \mathrm{doc}_j)$ represents the adaptability of the microblog text doc_j assigned to cluster C_i; x refers to both the global frequent terms of the microblog text doc_j and the cluster frequent terms of the cluster C_i; x' represents only the global frequent terms but not the cluster frequent terms; The frequent word sets of X represent C_i in Sim(X, Y), Y is the characteristic word of the text doc_j, Sim(X, Y) is the similarity value calculated by the HowNet semantic formula [6].

The reduction algorithm needed to calculate the fitness value of each microblog text belongs to each cluster, and then the microblog text was assigned to the cluster with the largest value. In the process of reduction, the value of each microblog text must be calculated whether it was divided into the repeated initial clusters or not. If the original microblog text was not divided into the repeated initial clusters, it did not need to be subtracted or calculated. The calculation of the value of this part of the text would appropriately affect the efficiency of the algorithm. The original subtraction algorithm was modified appropriately. First, the initial topic cluster was scanned to get the repeated initial clusters and the related repeated microblog texts. Then, values of these repeated micro-blog texts were calculated, and the texts were classified into the clusters of the largest value, while the texts of the other clusters were deleted.

3.3 Topic Clustering Algorithm is based on the Single Pass

The topic clusters related to the hot topics could be obtained after eliminating the initial cluster of microblog texts by the subtraction algorithm. These topical clusters may have semantic similarity and belong to the same big topic. Therefore, more accurate hot topics could be got by further cluster analysis of these related topic clusters.

Definition 6: Topic cluster similarity function

If there were two topic clusters C_i and C_j, and all frequent term sets of the topic cluster were defined as the feature vector of the topic, that is $\overrightarrow{C_i} = (t_{i1}, t_{i2}, \cdots, t_{in})$, $\overrightarrow{C_j} = (t_{j1}, t_{j2}, \cdots, t_{jm})$, then the semantic similarity matrix of frequent term C_i and C_j could be obtained, as shown in Tab. 1.

Table 1 Similarity matrix of frequent terms in topic clusters C_i and C_j

	t_{i1}	t_{i2}	…	t_{in}
t_{j1}	sim(t_{j1}, t_{i1})	sim(t_{j1}, t_{i2})	…	sim(t_{j1}, t_{in})
t_{j2}	sim(t_{j2}, t_{i1})	sim(t_{j2}, t_{i2})	…	sim(t_{j2}, t_{in})
…	…	…	…	…
t_{jm}	sim(t_{jm}, t_{i1})	sim(t_{jm}, t_{i2})	…	sim(t_{jm}, t_{in})

In order to simplify the calculation of the similarity of frequent terms and avoid too many words with small similarity affecting the calculation of the similarity of the whole topic cluster, the topic cluster similarity function only calculated the frequent terms with the largest similarity in the similarity matrix, and the specific formula is shown in Eq. (2).

$$C\mathrm{score} = \mathrm{sim}(C_i, C_j) = \frac{\sum_{n=1}^{k} \mathrm{sim}(t_i, t_j)_n}{k} \tag{2}$$

Single pass clustering algorithm is simple and fast, and it has achieved good results in traditional style clustering. The idea of this algorithm was applied to the topic clustering, the specific steps were as follows.

(1) The first topic cluster C_i was input, and the related feature frequent items of the topic cluster were extracted to obtain the topic feature vector.

(2) The next topic cluster was input continuously, and the meaning similarity value of the topic cluster was calculated by the Eq. (2);

(3) If $\mathrm{sim}(C_i, C_j) \geq \lambda$ (λ is the minimum threshold of cluster similarity for two topic clusters merging), C_i and C_j were merged, and the topic cluster eigenvector and similarity matrix were generated again. If $\mathrm{sim}(C_i, C_j) \leq \lambda$, the original topic cluster remained unchanged, the step 2 was continued to be executed;

(4) If the number of rows or columns in the cluster similarity matrix was less than the minimum number of clusters in the budget, the step 5 was executed (5); otherwise, the step 2 was continued to be executed;

(5) At the end of topic cluster clustering, the related clusters and the related hot topic sets were obtained.

4 DISCOVERY OF HOT TOPICS BY TOPSIS MODEL

Hot topic cluster sets were obtained by the clustering algorithm MFIHC. The ranking matrix of related popularity was calculated according to the number of microblog texts related to each hot topic and the related factors of the popularity of the blog, such as "click reading", "comments", "forwarding", "collection", as shown in Eq. (3).

$$\boldsymbol{A} = \begin{bmatrix} L_1 & R_1 & S_1 & M_1 & P_1 \\ L_2 & R_2 & S_2 & M_2 & P_2 \\ \cdots & \cdots & \cdots & \cdots & \vdots \\ L_n & R_n & S_n & M_n & P_n \end{bmatrix} \tag{3}$$

Among them, n represents the number of hot topic clusters, $L_i = \sum_{k=1}^{m} l_k$ represents the sum of microblog texts in each topic cluster, $R_i = \sum_{k=1}^{m} r_k$ represents the sum of "click to read" of each microblog in each topic cluster,

$S_i = \sum_{k=1}^{m} s_k$ represents the sum of "comments" of each microblog in each topic cluster, $M_i = \sum_{k=1}^{m} m_k$ represents the sum of "forwarding volume" of each microblog in each topic cluster, $P_i = \sum_{k=1}^{m} p_k$ represents the sum of "collection volume" of each microblog in each topic cluster.

There are five indexes in the mathematical model of hot topic ranking in Eq. (3). The determination of attribute weight of each index has a great influence on the final ranking result of the whole mathematical model. Entropy weight method is used to determine the weight of the index in order to better determine the attribute weight. Entropy weight method is an objective method to determine the weight according to the sorting data, which are not affected by human factors. It is more suitable for the computer automatic solution of this model.

The specific steps to solve the objective weight of index attribute of sorting matrix by entropy weight method are as follows.

(1) Normalize the index attribute of matrix $\boldsymbol{A}$

$$\dot{H} = \left(\dot{h}_{ij}\right)_{m\times n} = \left(h_{ij} \Big/ \sum_{i=1}^{m} h_{ij} \right)_{m\times n};$$

(2) Calculate index attribute entropy

$$E_j = -\frac{1}{\ln n} \sum_{i=1}^{m} \dot{h}_{ij} \ln \dot{h}_{ij}, \left(j = 1, 2, \cdots, n\right), \ \left(\dot{h}_{ij} = 0, \ln \dot{h}_{ij} = 0\right);$$

(3) Calculate objective weight of target attribute $\omega = \left(\omega_1, \ \omega_2, \ \cdots, \ \omega_n\right)$.

$$\omega_j = \left(1 - E_j\right) \Big/ \sum_{k=1}^{n} (1 - E_k).$$

TOPSIS model is a kind of multi-attribute decision-making method to evaluate various options from the perspective of geometry and attribute, similar to multi-point analysis of dimensional space, and to determine the closeness of the scheme according to the position of points and ideal points. The traditional TOPSIS model gives the index weight of the ranking matrix artificially, which has great subjectivity. In this paper, entropy weight method was used to solve the index weight, and the specific calculation steps for the improvement of TOPSIS model are as follows.

(1) Normalizing the original decision matrix $\boldsymbol{X} = \left(x_{ij}\right)_{m\times n}$

Benefit type attribute $y_{ij} = \dfrac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}}$, cost type attribute: $y_{ij} = \dfrac{1/x_{ij}}{\sqrt{\sum_{i=1}^{m} \left(1/x_{ij}\right)^2}}$

Normalized matrix: $\boldsymbol{Y} = \left(y_{ij}\right)_{m\times n}$

(2) Calculation of weighted normalized decision matrix $\boldsymbol{Z} = \left(z_{ij}\right)_{m\times n}$, $z_{ij} = \omega_j y_{ij}$.

(3) Determining the positive ideal solution $\boldsymbol{Z}^+$ and negative ideal solution $\boldsymbol{Z}^-$ of weighted normalized matrix

$$\boldsymbol{Z}^+ = \left(z_1^+, z_2^+, \cdots, z_n^+\right) = \omega, \ \boldsymbol{Z}^- = \left(z_1^-, z_2^-, \cdots, z_n^-\right) = 0$$

(4) Calculating the distance from each scheme to positive and negative ideal solutions

$$d_i^+ = \sqrt{\sum_{j=1}^{n} \left(z_{ij} - z_j^+\right)^2}, \ d_i^- = \sqrt{\sum_{j=1}^{n} \left(z_{ij} - z_j^-\right)^2}$$

(5) The distance from the solutions to the positive and negative ideal solutions is normalized

$$D_i^+ = \frac{d_i^+}{\max\limits_i d_i^+}, \ \mathrm{D}_i^- = \frac{d_i^-}{\max\limits_i d_i^-}$$

(6) Calculating the relative closeness of each scheme

$$T_i^+ = \frac{D_i^-}{D_i^+ + D_i^-}$$

The larger the value of T_i^+ is, the closer the scheme is to the positive ideal solution. The scheme is sorted according to the size of the combined value T_i^+, and finally the heat ranking of each topic cluster is obtained.

5 EXPERIMENTAL ANALYSES

The experimental data are the data of Sina blog used in literature [7], which contain 4236 relevant pages published by Sina blog, the specific time is from April 1 to April 30, 2014. The microblog texts were selected and the topics were manually annotated, and 10 topic categories were obtained which were manually counted, as shown in Tab. 2. The specific experimental environment: the hardware machine is Dell, CPU: i5 3.2G, RAM: 8G; the operating system is win7; the development language is C#, and the word segmentation tool is ICTCLAS, which is researched and developed by the Chinese Academy of Sciences.

Table 2 Specific situation of manually marking topic cluster

No	Tagging topic clusters	Cluster size
1	{South Korea, shipwreck}	582
2	{Lanzhou, pollution}	498
3	{China, Ma}	383
4	{Fight against corruption, promote integrity}	388
5	{housing price}	487
6	{Urban management}	336
7	{Article,Derailment}	388
8	{urine}	391
9	{Plane,Crash}	349
10	{Train, Cerailment}	434

The purity index and *F*-measure value index of the traditional testing clustering algorithm were used to compare in order to test the effectiveness of the MFIHC clustering algorithm [literature 2]. The calculation formulas of the index are as Eqs. (4) and (5). Under normal circumstances, the larger the sum of indicators, the better the clustering effect.

$$P = \frac{\sum_{p=1}^{k} a_p}{n} \tag{4}$$

C_p and C'_p represent the data set of manually labeled cluster and the data set of non manually (clustering algorithm), a_p represents the specific number of C_p and C'_p at the same time, n is the specific number of clusters.

$$F = \frac{2 * P(C_p^{'}, C_q) * R(C_p^{'}, C_q)}{P(C_p^{'}, C_q) + R(C_p^{'}, C_q)} \tag{5}$$

$P(C_p^{'}, C_q)$ represents the information precision value of clustering algorithm, and $R(C_p^{'}, C_q)$ represents the information recall rate of clustering algorithm.

5.1 Determination of Minimum Support (min sup) and Minimum Cluster Similarity (*Λ*) for the MFIHC Clustering Algorithm

The size of the minimum support called min sup has a great influence on the acquisition of the initial cluster group of the MFIHC clustering algorithm. Too large or too small will have different effects on the results of the MFIHC clustering algorithm.

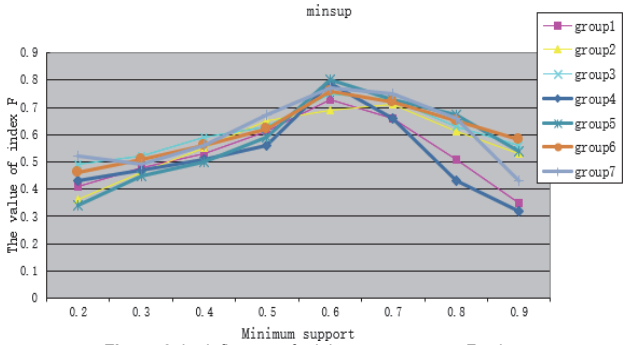

Figure 2 the influence of minimum support on *F*-value

Therefore, in order to determine the size of the min sup better, this paper randomly divides the manually labeled 10 cluster groups into seven groups, namely group1, group2, and group7, each group contains five cluster groups, and then carries out experiments on the seven groups with values from 0.2 to 0.9 to analyze the change process of the indicators, as shown in Fig. 2.

The size of the minimum cluster similarity called λ has a great impact on the grouping of topic clusters in the process of topic cluster clustering. Different similarity values may have different classification for topic clusters, because the minimum cluster similarity is the threshold value used to judge whether two topic clusters are similar. In this paper, 10 manually labeled cluster groups are randomly divided into seven groups, and then the values of the seven groups are tested from 0.2 to 0.8 to analyze the change process of the indicators, as shown in Fig. 3.

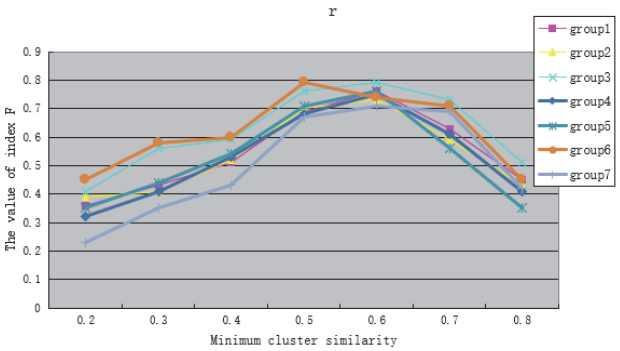

Figure 3 Influence of minimum cluster similarity on *F*-value

The final experimental results show that the MIFHC clustering algorithm has the best clustering effect when the value is in the range of 0.5-0.6.

5.2 Comparison of the MFIHC Clustering Algorithm in Different Topics

In this paper, 10 clusters are randomly divided into 5 groups. Each group contains 2, 4, 6, 8, 10 topics, which are the basic data of the experiment.

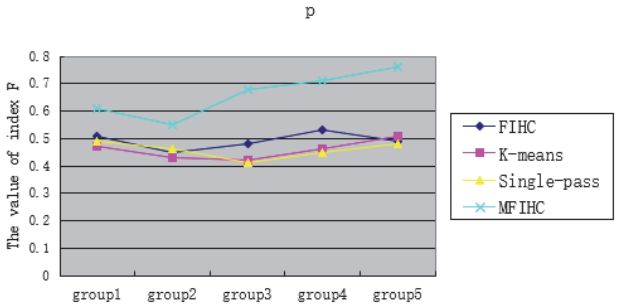

Figure 4 Performance of *p*-value for five groups of different topics

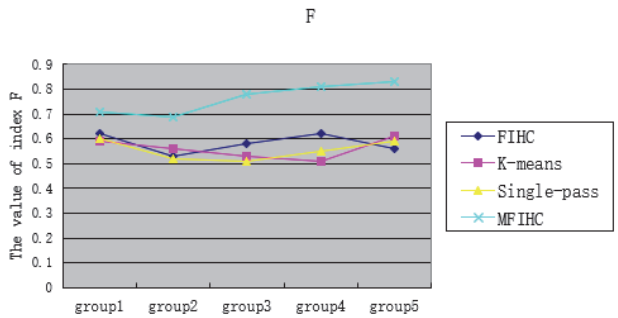

Figure 5 Performance of *F*-value for five groups of different topics

Figs. 4 to 5 shows the different performance of different clustering algorithms on the purity index *p*-value and index *F*-value under different topic numbers. From the graph, we can see that MFIHC clustering algorithm is the best performance on both indexes, which are about 15% higher than the other three clustering algorithms.

To further verify the stability of the algorithm, Fig. 6 shows the performance of four different clustering algorithms under the condition that the number of microblog texts increases from 5000 to 85000. From the

graph, it can be seen that MFIHC clustering algorithm performs better and better with the increase in the number of texts. The results show that the accuracy of the clustering algorithm is better with the increase in the number of samples, which are consistent with the characteristics of the clustering algorithm.

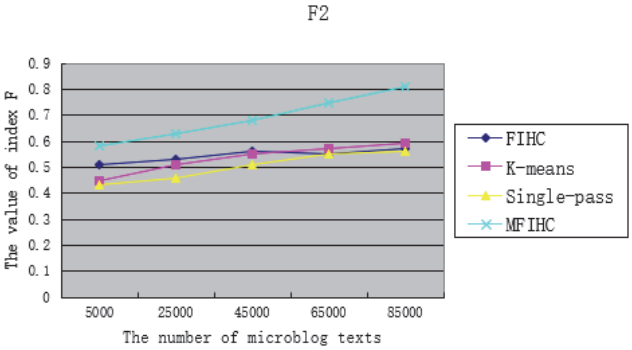

Figure 6 Effect of the number of microblog texts on indicator *F*-value

5.3 Performance Comparisons of MFIHC Clustering Algorithms

In addition to comparing the clustering effects of different clustering algorithms, the performance of clustering algorithms is further analyzed. Figs. 7 to 8 shows the performance of four clustering algorithms in different topic numbers and different microblog text numbers. From Fig. 7, it can be seen that the MFIHC and FIHC clustering algorithms consume less time than the other two clustering algorithms, mainly because these two clustering algorithms based on frequent terms are better suited for short text clustering on microblogs and can get results faster.

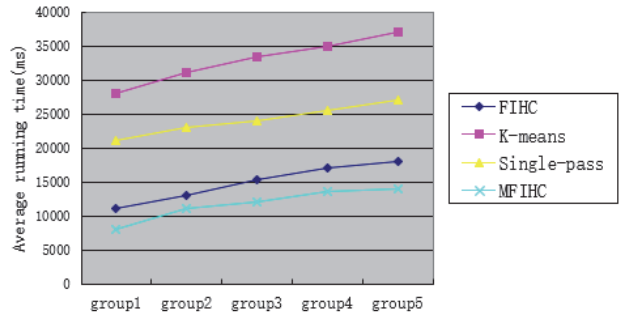

Figure 7 Efficiency of five different topics

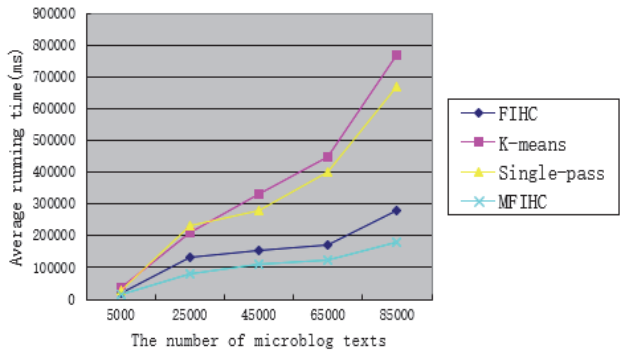

Figure 8 Effect of the number of microblog texts on operating efficiency

From Fig. 8, we can see that MFIHC and FIHC clustering algorithms do not increase significantly with the increasing number of microblog texts, but the other two clustering algorithms increase rapidly with the increasing number of texts. The experimental results show that MFIHC and FIHC clustering algorithms are better suited for clustering analysis of large amount of microblog texts. MFIHC used less time than FIHC in the process of increasing the amount of microblog text data, and its comprehensive performance is better.

5.4 Analysis of Hot Topics for Improving TOPSIS Model

The ranking of hot topics found by the improved TOPSIS model is different from that found by the traditional ranking method based on the number of clusters, which is shown in Tab. 3.

Table 3 Hot topics discovered by clustering and TOPSIS model

Label topic clusters	Number of FIHC clusters	Cluster size sort	TOPSIS-CI	TOPSIS
{South Korea, shipwreck}	582	1	0.840	1
{Lanzhou, pollution}	498	2	0.699	2
{housing price}	487	3	0.489	3
{Train, derailment}	434	4	0.408	6
{urine}	391	5	0.005	10
{Fight against, corruption}	388	6	0.468	4
{Article, derailment}	388	7	0.433	5
{China, Ma}	383	8	0.176	9
{Plane, crash}	349	9	0.361	7
{Urban management}	336	10	0.217	8

The clustering number of microblog texts was mainly considered in the traditional method. The related factors that affect the popularity of microblog, such as "click reading", "comments", "forwarding", "collection" were not considered. The hot topic cluster set was obtained by the MFIHC clustering algorithm. It was very necessary to use the improved TOPSIS model for further analysis. The improved TOPSIS model comprehensively considers the number of microblog texts related to each hot topic and the related factors of the popularity of the microblog, such as "click reading", "comments", "forwarding" and "collection". Finally，it could better analyze other factors that affect the popularity of microblog posts, and the final results could better mine the relevant hot topics.

6 CONCLUSION

Every day, a large number of blog articles are generated in the process of microblog's wide use through continuous dissemination. Microblog hot spots are discovered after some research and exploration based on MFIHC clustering and improved TOPSIS discovery method. This method is verified by experiments and compared with other algorithms. It can effectively cluster the constantly generated microblog articles, and analyze the clustering results to obtain the ranking of hot topics, and effectively find the hot topics in the current massive microblog articles. In short, this method can effectively and quickly analyze and evaluate microblog hot spots, which is helpful for big data collection and case analysis.

Acknowledgements

Funded projects: The program of cultivating outstanding young scientific research talents in Universities of Fujian Province (ZX17033), the doctoral research initiation Fund Program (CK18013), and the program of Fujian Provincial Department of Education (JAT201035).

7 REFERENCES

[1] Du, Y., Yi, Y., Li, X., Chen, X., Fan, Y., & Su, F. (2020). Extracting and tracking hot topics of micro-blogs based on improved latent Dirichlet allocation. *Engineering Applications of Artificial Intelligence, 87*, 103279. https://doi.org/10.1016/j.engappai.2019.103279

[2] Lu, Y. & Zheng, Y. (2018). Subject Analysis of the Microblog about US Presidential Election Based on LDA. *Proceedings of SAI Intelligent Systems Conference*, 998-1008. https://doi.org/10.1007/978-3-030-01054-6_69

[3] Zhang, T., Li, H., Zhang, X., Yu, F., & Zhao, K. (2019). The Evolution of Chinese New Media Research in 1998-2017 - An Analysis Based on Science Mapping by CiteSpace. *Journal of China Studies, 22*(2), 135-153.

[4] Fan, C., Wu, Y., Zhang, J., & Zhao, T. (2016). Research of public opinion hot spot detection model based on Web big data. *2016 IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC)*, 72-77. https://doi.org/10.1109/ICNIDC.2016.7974538

[5] Li, H., Xiao, H., Qiu, T., & Zhou, P. (2013). Food safety warning research based on internet public opinion monitoring and tracing. *2013 Second International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*, 481-484. https://doi.org/10.1109/ArgoGeoinformatics.2013.6621967

[6] Yi, B., Wang, Y., Chen, X., & Wang, Y. (2013). Extracting hot topics from micro blogging based on key words detection and text clustering. *Applied Mechanics and Materials, 303*, 2289-2293. https://doi.org/10.4028/www.scientific.net/AMM.303306.2289

[7] Kishore, D. & Rao, C. S. (2020). A multi-class SVM based content based image retrieval system using hybrid optimization techniques. *Traitement du Signal, 37*(2), 217-226. https://doi.org/10.18280/ts.370207

[8] Battula, B. P. & Balaganesh, D. (2020). Prediction of hospital re-admission using firefly based multi-layer perceptron. *Ingénierie des Systèmes d'Information, 25*(4), 527-533. https://doi.org/10.18280/isi.250417

[9] Das, M., Kumar, R., & Sahana, B. C. (2020). Implementation of effective hybrid window function for E.C.G signal denoising. *Traitement du Signal*, 37(1), 119-128. https://doi.org/10.18280/ts.370116

[10] Liu, D., Wang, W., & Li, H. (2013). Evolutionary mechanism and information supervision of public opinions in internet emergency. *Procedia Computer Science, 17*, 973-980. https://doi.org/10.1016/j.procs.2013.05.124

[11] Fu, X., Li, J., Yang, K., Cui, L., & Yang, L. (2016). Dynamic online HDP model for discovering evolutionary topics from Chinese social texts. *Neurocomputing, 171*, 412-424. https://doi.org/10.1016/j.neucom.2015.06.047

[12] Khan, K., Baharudin, B., Khan, A., & Ullah, A. (2014). Mining opinion components from unstructured reviews: A review. *Journal of King Saud University-Computer and Information Sciences, 26*(3), 258-275. https://doi.org/10.1016/j.jksuci.2014.03.009

[13] Huang, S., Liu, Y., & Dang, D. (2014).Burst topic discovery and trend tracing based on Storm. *Physica A: Statistical Mechanics and its Applications, 416*, 331-339. https://doi.org/10.1016/j.physa.2014.08.059

[14] Ma, B., Zhang, N., Liu, G., Li, L., & Yuan, H. (2016).Semantic search for public opinions on urban affairs: A probabilistic topic modeling-based approach. *Information Processing & Management, 52*(3), 430-445. https://doi.org/10.1016/j.ipm.2015.10.004

[15] Krishna, K. V. S. S. R. & Prakash, B. B. (2019). Intrusion detection system employing multi-level feed forward neural network along with firefly optimization (FMLF2N2). *Ingénierie des Systèmes d'Information, 24*(2), 139-145. https://doi.org/10.18280/isi.240202

[16] Prashanth, J. S. & Nandury, S. V. (2019). A cluster-based approach for minimizing energy consumption by reducing travel time of mobile element in WSN. *International Journal of Computers Communications & Control, 14*(6), 691-709.

[17] Jiang, H. (2020). Solving Multi-Robot Picking Problem in Warehouses: a Simulation Approach. *International Journal of Simulation Modelling, 19*(4), 701-712. https://doi.org/10.2507/IJSIMM19-4-CO19

[18] Zhong, M. J., Tan, L., & Qu, X. L. (2019). Identification of opinion spammers using reviewer reputation and clustering analysis. *International Journal of Computers Communications & Control, 14*(6), 759-772. https://doi.org/10.15837/ijccc.2019.6.3704

[19] Singh, R. K., Singh, P., & Bathla, G. (2020). User-review oriented social recommender system for event planning. *Ingénierie des Systèmes d'Information, 25*(5), 669-675. https://doi.org/10.18280/isi.250514

[20] Pei, J. Y. & Shan, P. (2019). Prediction of the dissemination of health news on microblogging sites based on ample feature selection and support vector machine. *Revue d'Intelligence Artificielle, 33*(5), 359-365. https://doi.org/10.18280/ria.330505

Contact information:

Lina LIN
Information Engineering School,
Jimei University Chengyi College,
Xiamen 361021, China
E-mail: linlina2008@jmu.edu.cn

Dezhi WEI
(Corresponding author)
Information Engineering School,
Jimei University Chengyi College,
Xiamen 361021, China
E-mail: aide@jmu.edu.cn