Semantic Similarity Calculation of Short Texts Based on Language Network and Word Semantic Information

Zhijian Zhan[1], Feng Lin[2], and Xiaoping Yang[1]

[1] School of Information, Renmin University of China, Beijing 100872
[2] School of Opto-electronic and communication Engineering, Xiamen
zhanzj@ruc.edu.cn, linfeng@xmut.edu.cn

Abstract. We first analyzes the deviation when current similarity calculation methods for texts are applied to short texts, and proposes a similarity calculation method for short texts based on language network and word semantic information. Firstly, models the short texts as language network according to the complex-network characteristic of human being's language. Then analyzes the comprehensive eigenvalue of the words in the language network and the word similarity between different texts to obtain the word semantic. Calculate the similarity between short texts combining language network and word semantic. Finally the effectiveness of proposed algorithm is verified through clustering algorithm experiments.

Keywords: language network, text clustering, short texts similarity, word similarity.

1 Introduction

Text Clustering refers to divide text collection into different clusters automatically. Texts in the same cluster are very similar and differentiate in different clusters [1]. Text clustering is the fundamental research for text excavation. Researchers at home and abroad have got an earlier research and development for the algorithm on text clustering and obtain good results. However, there are several principal problems existing in the process of text clustering, include how to define the number of clusters, how to calculate the similarity between texts and how to assess text clustering.

With the rapid development of WEB, short texts such as micro blog, SMS and IM, etc., take more and more importance in people's life. Unlike long texts with rich information, short texts contain poor information. Usually, their lengths don't exceed to 200 words. Short texts generally have explicit themes to transfer the author's intention. Traditional texts similarity calculation methods are to obtain the statistic of word similarity between texts. For long texts, the word number is larger and the method can work effective. But short texts may only contain a few number of words, there may be no common words between them. If the calculation methods of similarity between long texts are applied, we may achieve false results between short texts. Short texts like language that people use in daily life are originated with

J. Wu et al. (Eds.): ACA 2014, CCIS 451, pp. 215–228, 2014.

people's feeling, are uncertain in line with the normal rules of grammar, only if they can express the speaker's meaning. For such short texts with unclear grammar, short length and irregular word order, we can't utilize conventional calculation methods of similarity for long texts. For instance, there're two short texts: "how to download music from internet" and "can I transfer mp3 to my laptop". If only the common words are in statistics, there are few identical words. But the two sentences have a high degree of similarity in fact.

Similarity calculations for short texts have been widely used in many fields. In information retrieval, it's considered as one of the best method to improve the retrieval results [2]. In mail message processing, it can implement mail classification faster [3]. In the interface development of nature language database, it can extend the inquiry interface [4]. Moreover, it also has important applications in health advisory dialogue system [5], property sales [6], telephone sales [7] and smart tour guide [8].

Traditional methods of similarity measure like Vector Space Similarity Measure will cause erroneous results when applied in short texts, because most of them treat texts as a set of words. They calculate the word's number appearing in the text, establish characteristic vector and compute the text similarity using the cosine similarity or Jaccard similarity [9]. Due to fewness of words and brief content of short texts, the method not only ignores the semantics information of the words but also the order information and grammar information. It creates a vector space with very high dimension and necessarily causes a problem of data sparse, finally leads to low computational efficiency.

The innovation of this paper lies in: the first is in accordance with the special characteristics of short text, we introduce the language network model to represents the semantic information of short text; the second is with combine the important features of language network and semantic information of words, we propose a new short text similarity calculation method. Provided the short texts, our method can efficiently and quickly calculate the similarity on the semantic level between them. It can be applied in a wide range.

2 Related Works

With the rapid development of Internet, text resources increase sharply. In fact 80% of Internet resources are texts. In the past few decades, automatically processing of electronic text resources have become the key research of researchers. There's a large quantity of the Internet text resources including Webpage text, email, news messages. With the large number of network texts, researcher's principal interests on text processing are how to mine the needed information [10]. In the early 80s, the major application of text processing is text categorization in knowledge engineering. Experts artificially defined regular knowledge base in first, and then determined the texts to relevant category [11]. In order to avoid the low efficiency caused by excessive artificial involvement in the writing of regular base, in the 90s, researchers proposed many improved method including regular base construction method based on machine learning. The method can get a better result than that based on artificial writing, largely save human resource and improve efficiency [12].

Besides text mining, many other Natural Language Processing application, including data mining, machine learning, pattern recognition, artificial intelligence, statistics, computational linguistics, compute network technology and informatics, also set appropriate requirement for text processing. The text resource on the Internet is massive, heterogeneous and widely distributed. The contents of texts are natural language of human beings and can't be understood by computer directly. The data processed by traditional computer text processing are structured. However, texts are semi-structured or non-structured. In particular, short texts have less content, maybe several sentences, one sentence or several words, even only one word. Consequently, the primary problem is to represent the short text effective in computers to reflect the text characteristic with sufficient information and avoid low computational effectively.

In recent years, the attentions of researchers have been greatly attracted on complex network. Complex network is almost everywhere in our life and their model are widely used in life sciences[13], stress media[14], neural networks[15], space-time game[16], gene controlling network[17] and other self-organized systems. Complex network is composed of nodes and edges, whether it's visual system or not. For example, telephone networks and oil-gas transmission systems are visual and have material nodes and edges. While interpersonal relationship network and social work relationship are invisible. The topology graph of network usually is fully regular or fully random. But many biological networks, technology networks and social networks is between the two [18].

Researchers have demonstrated that human languages also have characteristic of small-world complex network. Common used words severed as nodes and semantic relationship between words as edges, the complex network of human language can be established. Taking it as thinking, we can establish complex network for texts, and obtain the weight and semantic information of characteristic words by computing the comprehensive edge value of each nodes in the language network. As a characteristic word to represent the meaning of the text, it must meet the following four requirements:

1) Distinctly represent the text content ;
2) Clearly distinguish the text meaning from other meanings ;
3) The quantity is small ;
4) The algorithm is not complicated.

Harris believes that the ability to calculate the similarity of text is due to that those element words which represent similar meaning in similar short texts [19]. The thought is confirmed by Firth. Firth supposes that in any language, words with the same meaning appear in different style [20]. Miller further verifies that the words in text are similar to some extend as long as the texts are similar [21]. Thus, a conclusion can be draw that words in similar texts are always similar, since similar texts express similar subject. Instead, if words in texts are similar, the texts are similar. We can firstly compute the similarity of words, and then comprehensively weight them to achieve the similarity of texts. Based on such conclusion, the similarity of short texts can be increased through the improvement of similarity between words and the weighted algorithm.

To address the above problem and after comparing and analysis of other methods for characteristic representation and similarity calculation, the paper proposes an calculation method for the semantic similarity of short texts based on language network and word semantic. The main contributions are as follows:

1) Modeling short texts with language network to provide a proper characterization model for the calculation of semantic similarity.

2) Combining the important features and word semantic information of language network, and presenting the calculation method for the similarity of short texts.

3) Verifying the effectiveness of the proposed method based on classification experiments on several mainstream texts. The experiments have demonstrated that our method is super to the traditional TF-IDF method and the method proposed in [22].

3 Short Texts Similarity Based on Word Semantics

3.1 Important Characteristics of Complex Network

To establish model of complex network with mathematical linguistics and according to the important characteristics of complex networks which are generally accepted in the industry, the graphic definition for complex network is given as below:

Definition1(complex network): Suppose complex network G=(V,E,W) is a graph where V={$v_1,v_i,\ldots\ldots v_n$}, is the nodes collection, E={(v_i, v_j), $v_i \in V$, $v_j \in V$}, is the edges collection and W={w_{ij}|(v_i, v_j)∈E}, is the weight collection. The characteristic equation is listed respectively in the following:

1) Di is the degree of node v_i, defined as:

$$D_i = |\{(v_i, v_j) : (v_i, v_j) \in E, v_i \in V, v_j \in V\}| \tag{3-1}$$

In complex network, D_i represents the number of nodes which have edge with node v_i. D_i indicates the connectivity of one node with others.

2) Ki is the aggregation degree of node v_i, defined as:

$$K_i = |\{(v_j, v_k) : (v_i, v_j) \in E, (v_j, v_k) \in E, v_i \in V, v_j \in V, v_k \in V\}| \tag{3-2}$$

In complex network, K_i represents the connectivity between nodes which are v_i-centered. K_i indicates the nodes' aggregation within a local range.

3) C_i is the clustering degree of v_i,, defined as:

$$C_i = \frac{K_i}{\left(\frac{D_i}{2}\right)} = \frac{2K_i}{D_i(D_i - 1)} \tag{3-3}$$

The numerator in the formula is the aggregation degree of v_i, while the denominator is the degree distribution statistics when the graph is complete connected.

4) WD_i is the weigh degree of node v_i, defined as:

$$WD_i = \sum_{(v_i, v_j) \in E} W_{ij} \tag{3-4}$$

WD_i is the sum of the weight of all edges which are connected with v_i.

5) WK_i is the weighted aggregation degree of v_i, defined as:

$$WK_i = \sum_{(v_j, v_k) \in E} W_{jk} \tag{3-5}$$

WK_i is the sum of the weight of all edges which are *vi*-centered.

6) WC_i is the comprehensive aggregation degree, defined as:

$$WC_i = \frac{WK_i}{WD_i} \times C_i = \frac{WK_i}{WD_i} \times \frac{2K_i}{D_i(D_i - 1)} \tag{3-6}$$

WC_i is proportional to WK_i and K_i, while inversely proportional to WD_i.

7) The aggregation factor of complex network G is defined as:

$$C = \frac{1}{n} \sum_{i=1}^{n} C_i \tag{3-7}$$

The aggregation factor is the average of all nodes' clustering degree.

8) The average shortest path of G is defined as:

$$L = \sum_{v_i, v_j \in V} l(v_i, v_j) \tag{3-8}$$

$l(v_i, v_j)$ represent the shortest path between any two nodes v_i and v_j. In complex network's graph, there may be more than one path between any two nodes. Given that $l(v_i, v_j)$ is the shortest path, then the average shortest path of the complex network can be defined as the sum of all the shortest paths.

9) BC_i is the clustering factor of node v_i, defined as :

$$BC_i = \sum_{i \neq j \neq k} \frac{l_{jk}(i)}{l_{jk}} \tag{3-9}$$

$l_{jk}(i)$ represents the length of the path which is among all the shortest path between v_j and v_k and through v_i. l_{jk} represents all the shortest path between v_j and v_k.

BC_i has strong practical significance and reflects the place flow of v_i toward the complex network. The research has demonstrated that complex network can be regarded as a set of connected sub network. The sub network's connection nodes play a critical role. Consequently, the shortest path between two nodes which belong to two different sub networks is via node v_i.

10) BP_i is the path factor of node v_i, defined as:

$$BP_i = \frac{1}{\sum_{i \neq j \in V} d_{ij}} \tag{3-10}$$

BP_i is defined to address the situation that the clustering factor may be 0. Since when some key nodes are not in the shortest path, BC_i =0. Nevertheless, those nodes are the key nodes of the complex network. And the clustering factor emphasizes the local connectivity, thus the introduction of BP_i is to enhance the global connectivity.

11) Z_i is the comprehensive eigenvalue of node v_i in complex network, defined as:

$$Z_i = \frac{\alpha \times WC_i + \beta \times BC_i + \eta \times BP_i}{(N-1) \times (N-2)} \tag{3-11}$$

α、βandηcan be adjusted according to different applications.

3.2 Text Pre-processing

Although the word number is small and the content is brief, current natural language processing technologies can't fully process a short text message. Before building text characteristic model, pre-processing is necessary for short texts, including word separation, removal of stop-word, stemming, etc. For English texts, words are divided by blank space or obvious punctuations, therefore word division can be quickly realized according to such symbols. But there're no clear boundaries between Chinese texts, hence word separation for Chinese texts through algorithm is needed. At present, the algorithms for word separation are mainly distributed into three classes: matching method based on forward, separation methods based on maximum probability and shortest path. The main idea of matching method is to obtain candidate sub-string from text string by lookup in the dictionary. And the separation methods based on maximum probability is to calculate the probability of separation results of Chinese sentence strings and select a separation result with maximum probability. Method based on shortest path constructs graph for text string, calculate the number of the word with shortest path and conduct it as separation result. Chinese word separation is the important basis of Chinese information processing and its result has great impact on the effect of application. The main reason for separation ambiguity in Chinese is polysemy and synonymy in sentences, so the expression can be various.

After word separation for short texts, removal of stop words should be implemented. Stop words refer to the words whose impact on text expression is negligible and valueless for text processing, like "the、a、of、for、in" in English. The most common method for removal of stop words is the maintenance a stop word list. When a word appears in the list after text separation, it should be removed. Stop words are always related to application field. Since the proposed method need semantic analysis for words, after word separation and removal of stop words, there're two steps should be executed as following:

1) Replace people names, address and organization names in short texts with particular strings. Among them, names are substituted with PEO, address is substituted with ADD, and organization names with COM.
2) Mark the property of words in short texts. Words can be divided into different types according to their characteristic and application. Among them, notional word can best represent the meaning of short texts. Thus, it's necessary to distinguish the words that who are nouns, verbs, adjectives or adverbs.

3.3 Language Network Construction

One of important features that distinguish human language from other biological language is that human language has a large number of words. Statistics show that an ordinary foreign high school student's English vocabulary is more than 100 thousand. People can make decision within 100ms that the combination of a word or term is right or wrong. Research has indicated that the reason for human being's literacy skill is the great deal of connection between human languages. Human language has network characteristics as small-world. The words in human language texts are not random and out-of-order, but express a particular subject according to the relationship between words. The number of word is limited, while different word order can produce tens of thousands of texts that hold different meanings. Texts are mostly composited with paragraphs and sentences. The basic component element of a sentence is word. Taking words as nodes, the relationship between words as edges, and the language network can be constructed for the text. When two different words appear in the same sentence, they have grammar relationship, and the edge is generated. Therefore, edge inevitably exists between adjacent words. However, dose grammar relationship exists between non-adjacent words? How to define a specific distance within which two words have edge relation? If only the relation between adjacent words is collected, the relation between long-distant words may be lost and the significance of some useless words in the network maybe rose. So the correlation between words in the sentence should be determined. If the span is too short, much important correlation can't be recorded, whereas much redundancy information will be generated if the span is too long. The paper explores the regulation in [23] that the maximum correlation span is 2, because it's most common and important in language network. For instance, for the sentence "texts similarity calculation process", "texts", "similarity", "calculation" and "process" will be generated through word separation, thus the language network can be built as the Fig.1 shows. The construction of language network for the whole texts can be generated by combination of the same nodes and edges in each sentence.

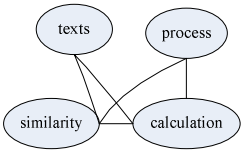


Fig. 1. An example of language network

3.4 Similarity Calculation of Short Texts

After construction of language network for short texts, we can compute the comprehensive eigenvalues of each word node by using formula (3-11) and consider them as an eigenvector to calculate the similarity between short texts. Due to small number of words and brief content of short texts, there are not too many words after preprocessing. Hence, the dimension of the eigenvector can't be high. The next is to consider how to calculate the similarity of short texts. Because those words deliver the most information of short texts, the similarity of short texts can be converted into that of eigenvectors. Moreover, thanks to the variable length of each short text, the dimension of eigenvectors which characterize the short text is also different. Such impact should be eliminated to make the similarity of eigenvectors satisfy the basic measurement standard of similarity.

Suppose v_i and v_j are eigenvectors of two different short text X and Y and $v_i=(w_{i1},w_{i2},\ldots\ldots w_{im})$, $v_j=(w_{j1},w_{j2},\ldots\ldots w_{jn})$. Define the similarity between two vectors as follows:

$$STSim(v_i, v_j) = cf \times VectSim(v_i, v_j) \tag{3-12}$$

$VectSim(v_i,v_j)$ denotes the similarity between v_i and v_j and cf denotes the weight factor. If there're many words whose similarity is high in the two short texts and their comprehensive eigenvalues take a large proportion, they will play important roles in each short texts. Therefore, we can firstly find out the feature words which meet the similarity threshold criteria, and then compute the sum of the comprehensive eigenvalues of the feature words, and finally ratio it with the total comprehensive eigenvalues of the whole text and weight it. The detail calculation formula of the weight factor is defined as following:

$$cf = \frac{1}{2} \times \left\{ \frac{\sum_{k \in \Lambda i} Z_{ik}}{\sum_{k=1}^{m} Z_{ik}} + \frac{\sum_{l \in \Lambda j} Z_{jl}}{\sum_{j=1}^{n} Z_{jl}} \right\} \times \frac{(|\Lambda_i| + |\Lambda_j|)/2}{\max(m,n)} \tag{3-13}$$

Where, Z_{ik} is the comprehensive eigenvalue of language network of feature word w_{ik}. In the right term, the numerator denotes the sum of comprehensive eigenvalue of feature words which meet the similarity threshold criteria, and the denominator signifies that of all feature words. The definitions of the collection Λ_i and Λ_j in (3-13) are:

$$\Lambda_i = \{k : 1 \le k \le m, \max\{sim(w_{ik}, w_{jl})\} \ge \mu\} \tag{3-14}$$

$$\Lambda_j = \{l : 1 \le l \le n, \ \max\{sim(w_{jl}, w_{ik})\} \ge \mu\} \tag{3-15}$$

If the similarity between the word w_{ik} in the eigenvector v_i and another word w_{jl}(l=1,2……n) in the eigenvector v_j exceeds the specified similarity threshold, the feature word w_{ik} will be subsumed to collection Λ_i. Select the feature words from eigenvector v_j in collection Λ_j according to the construction process of Λ_i. $|\Lambda_i|$ and

$|\Lambda_j|$ denote the element number of Λ_i and Λ_j respectively. The more the element of the collection is, the more the number of words who meet the similarity threshold criteria is and the greater significance on similarity they will place. $Sim(w_{jl},w_{ik})$ signifies the semantic similarity between w_{jl} and w_{ik}.

$$\mathrm{VectSim}(\mathrm{v_i},\mathrm{v_j}) = \frac{1}{2}\left(\frac{1}{m}\sum_{k=1}^{m}\max\{Sim(w_{ik},w_{jl})\} + \frac{1}{n}\sum_{l=1}^{n}\max\{Sim(w_{ik},w_{jl})\}\right) + \frac{\sum_{k=1}^{\max(m,n)} Z_{ik}\times Z_{jl}}{\sqrt{\sum_{k=1}^{m} Z_{ik}^2 \times \sum_{l=1}^{n} Z_{jl}^2}} \tag{3-16}$$

$VectSim(v_i,v_j)$ is determined by the word similarity of vector v_i and v_j and cosine similarity between vectors.

3.5 Basic Flow

Input：two short texts X and Y, similarity threshold μ

Output：similarity value between X and Y

Step1: preprocess X and Y, establish corresponding language network and calculate the comprehensive eigenvalue Z for each node in the network by formula (3-11);

Step2: generate the feature word vectors for X and Y, $v_i = (w_{i1},w_{i2},\ldots\ldots w_{im})$ and $v_j = (w_{j1},w_{j2},\ldots\ldots w_{jn})$;

Step3: from the word w_{i1} in vector v_i on, seek the word w_{jk} in v_j which has a highest similarity with w_{i1} and record the similarity value θ between w_{i1} and w_{jk}. Compare θ with μ. Place w_{i1} into the collection Λ_i if θ is larger than μ.

Step4: repeat Step3 until all the words in vector v_i has their corresponding largest-similarity word in vector v_j. Record the similarity value and adjust the collection Λ_i.

Step5: calculate the sum achieved in Step3 and Step4, and divide it by the number of words in vector v_i. Take the result as the similarity $Sim(v_i, v_j)$ between v_i and v_j;

Step6: acquire Λ_j and $Sim(v_j, v_i)$ in the same way;

Step7: get $VectSim(v_i, v_j)$ by using the result of Step5 and Step6 and formula 3-16.

Step8: calculate the total of comprehensive eigenvalue of all the words in collection Λ_i and Λ_j, and gain the weight factor cf by formula 3-13;

Step9: compute the similarity value between X and Y by formula 3-12.

4 Experiments and Analysis

We choose experiment data from the partial text classification corpus library gathered and organized by the natural language processing group of Fudan University. The partial corpus library is divided into 10 categories and contains 2706 articles. Each category is subdivided into different categories based on text content as table 1 shows:

Table 1. Abstract of experimental data

category	Number of text	Number of subcategory	the smallest number of text in subcategory	the biggest number of text in subcategory	The average text number in subcategory
Environment	200	6	8	25	33
Computer	200	5	10	22	40
Transportation	214	8	7	20	26
Education	220	6	6	16	37
Economy	325	5	11	14	65
Military	240	8	12	20	30
Sports	350	9	9	22	39
Medicine	204	6	8	20	34
Arts	248	5	7	23	50
Politics	505	10	7	18	50

The experiment firstly implement the processing on text collection using division software ICTCLAS developed by Chinese Academy of Science and then establish language network, calculate the comprehensive eigenvalue Z for each word. Take feature word as feature vector of the text and comprehensive eigenvalue as weight of vector. The similarity between feature words can be obtained suing the method proposed in [21]. Afterwards, combining with the method for text similarity calculation proposed in the paper, compute the similarity of text data collection to get the similarity matrix.

The experiment is carried out in the Windows 7 operating system, hardware configuration of CPU dual core 3.3G, 4G ram, 1T hard disk space. Using Java language, development tools is Eclipse 3.2.

The experiment verifies the effectiveness of the proposed algorithm, and compare the clustering result gained by text similarity matrix based on TF-IDF[10] and that of TSemSim combining with word semantic information proposed in [17]. Clustering experiments are done with CLUTO toolkit① and algorithms like K-Mean(DKM), bipartite K-mean(BKM) and aggregation K-mean(AKM) are achieved.

The experiment adopts F-metric value to measure the computation of text similarity. F-metric value is a comprehensive evaluation index given by precision P and recall R, defined as follows:

$$F = \frac{2RP}{R+P}$$

$$\mathrm{P} = \frac{\text{num of correctly returned text}}{\text{num of total calculated text}}$$

$$\mathrm{R} = \frac{\text{num of correctly returned text}}{\text{num of text in subcategory}}$$

F-metric value of global clustering is defined as:

$$F = \sum_i \frac{n_i}{n} \max_j(F)$$

In above formula, n_i denotes the text number in each subcategory, n denotes the number of all texts and j is the clustering result after computation. The larger F is, the better the clustering result is.

The first step is to determine the impact that similarity threshold μ places on clustering result. Fig1 shows the impact of μ in the case of DKM clustering algorithm. Seen from the diagram, the clustering effect changes in a parabola trend when μ varies. When μ is in the interval [0.65, 0.7], the clustering effect is best. After analysis, when μ is too small or too large, the number of elements in the selected feature word collectionΛ_i andΛ_j will varies and affect the clustering effect.

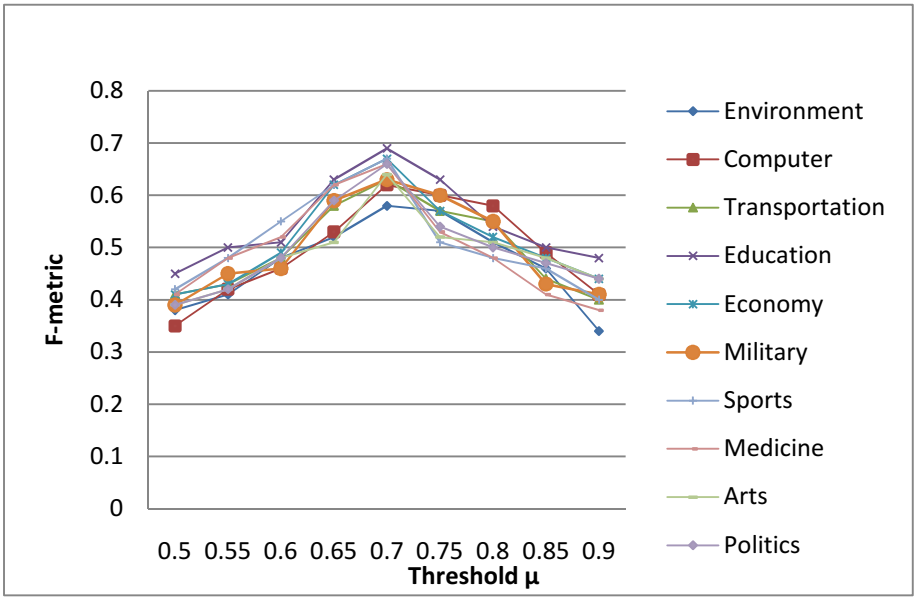
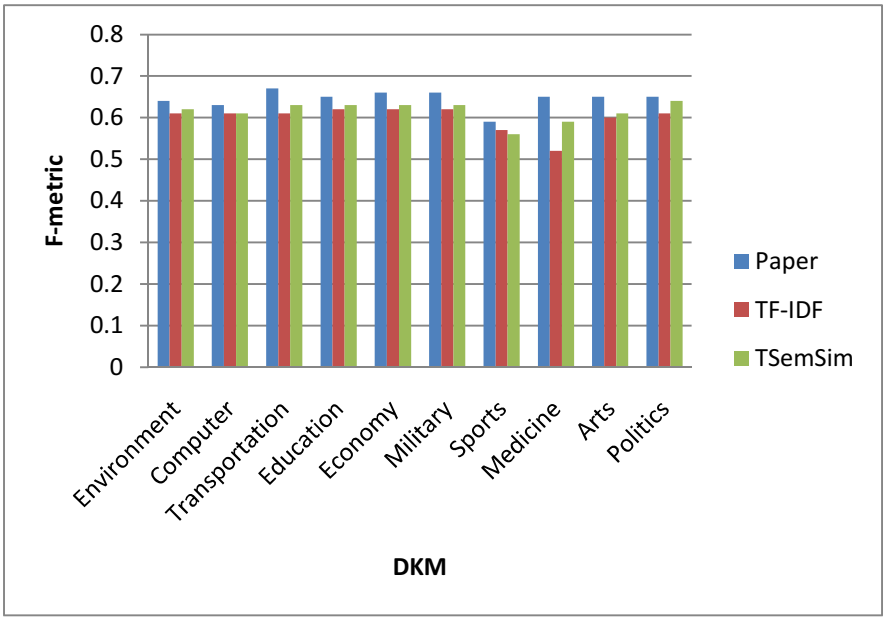


Fig. 2. The impact of similarity threshold μ on clustering result

According to above experimental result, the paper chooses 0.7 as similarity threshold μ. Fig.2 presents the comparison result gained by proposed algorithm, TF-IDF and TSemSim algorithm. We can see from Fig.3 that no matter in the situation of DKM、BKM or AKM clustering algorithm, the proposed algorithm can achieve better *F*-metric value than the other two. Thus the proposed algorithm can effectively enhance clustering result.



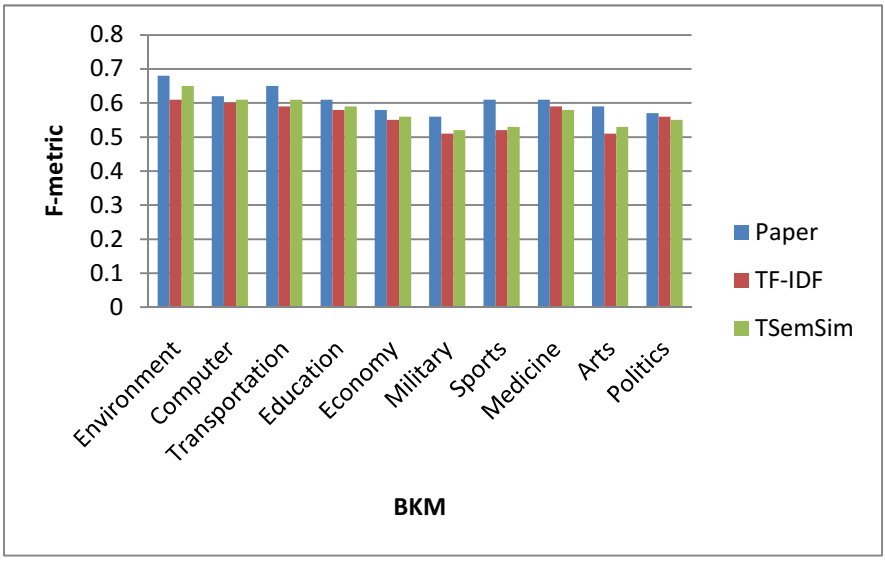


Fig. 3. Comparison result of F-metric gained by proposed algorithm, TF-IDF and TSemSim algorithm in DKM, BKM, AKM clustering algorithm

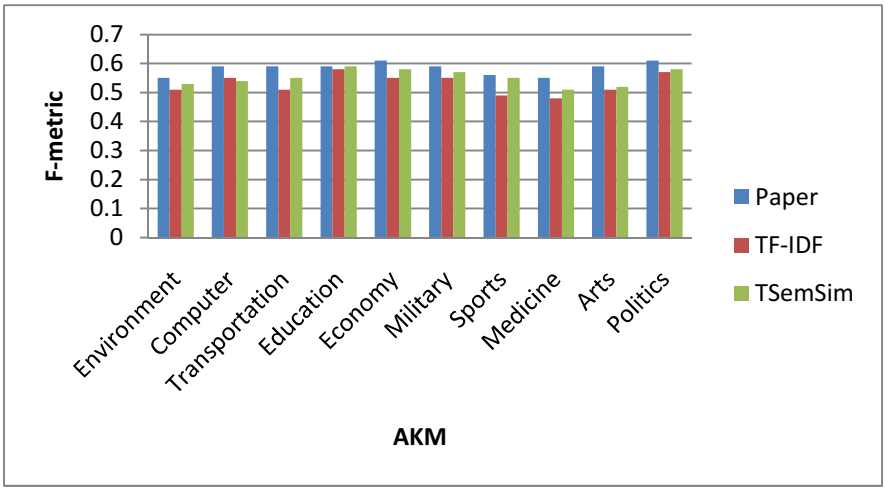


Fig. 3. (*continued*)

5 Conclusion

The paper firstly analyzes the disadvantage of existing measurement method for text similarity based on statistic and semantic analysis and then proposes a new calculation method for text similarity based on language network and word semantic information. Compared with traditional method, the proposed algorithm can decrease the dimension of text representation model and combine the semantic similarity between words to calculate the similarity between texts. Experiments based on classical clustering algorithm are implemented to verify the effectiveness of proposed algorithm.

The further work is to in-depth analyze the influence exerted by the words in different location or with different weight on the similarity calculation result, basing on existing basis of language network and word semantic information analysis. Comprehensively consider other information like the location weight of word and paragraph structure and improve the calculation precision of similarity calculation for texts.

References

1. Fung, B.C.M., Wang, K., Ester, M.: Hierarchical document clustering. In: John, W. (ed.) The Encyclopedia of Data Warehousing and Mining, pp. 970–975. Idea Group (2005)
2. Hall, P., Dowling, G.: Approximate string matching. Computing Survey 12(4), 381–402 (1980)
3. Lamontagne, L., Lee, H.-H.: Textual reuse for email response. In: Funk, P., González Calero, P.A. (eds.) ECCBR 2004. LNCS (LNAI), vol. 3155, pp. 242–256. Springer, Heidelberg (2004)

4. Glass, J., et al.: A Framework for Developing Conversational User Interfaces. In: Fourth International Conference on Computer-Aided Design of User Interfaces, Funchal, Isle of Madeira, Portugal (2004)
5. Bickmore, T., Giorgino, T.: Health dialog systems for patients and consumers. J. Biomed. Inform. 39(5), 556–571 (2006)
6. Cassell, J., et al.: Embodied Conversational Agents (2000)
7. Gorin, A.L., Riccardi, G., Wright, J.H.: How I help you? Speech Communication 23, 113–127 (1997)
8. Graesser, A.C., et al.: AutoTutor: An Intelligent Tutoring System With Mixed Initiative Dialogue. IEEE Transactions on Education 48(4), 612–618 (2005)
9. Salton, G.: The SMART Retrieval System-Experiments in Automatic Document Processing. Prentice Hall Inc., Englewood Cliffs (1971)
10. Dinesh, R., Harish, B.S., Guru, D.S., Manjunath, S.: Concept of Status Matrix in Text Classification. In: The Proceedings of Indian International Conference on Artificial Intelligence, Tumkur, India, pp. 2071–2079 (2009)
11. Mitra, V., Wang, C.J., Banerjee, S.: Text Classification: A least square support vector machine approach. Journal of Applied Soft Computing 2007(7), 908–914 (2007)
12. Fung, G.P.C., Yu, J.X., Lu, H., Yu, P.S.: Text classification without negative example revisit. IEEE Transactions on Knowledge and Data Engineering 2006(18), 23–47 (2006)
13. Strogatz, S.H., Stewart, I.: Coupled oscillators and biological synchronization. Sci. Am. 269(6), 102–109 (1993)
14. Gerhardt, M., Schuster, H., Tyson, J.J.: A cellular automaton model of excitable media including curvature and dispersion. Science 247, 1563–1566 (1990)
15. Hopfield, J.J., Herz, A.V.M.: Rapid local synchronization of action potentials: Toward computation with coupled integrate-and-fire neurons. Proc. Natl Acad. Sci. USA 92, 6655–6662 (1995)
16. Nowak, M.A., May, R.M.: Evolutionary games and spatial chaos. Nature 359, 826–829 (1992)
17. Kauffman, S.A.: Metabolic stability and epigenesis in randomly constructed genetic nets. J. Theor. Biol. (22), 437–467 (1969)
18. Watts, D., Strogatz, S.: Collective dynamics of 'small-world' networks. Nature 393, 440–442 (1998)
19. Harris, Z.: Distributional Structure. Word (10), 146–162 (1954)
20. Firth, J.R.: A Synopsis of Linguistic Theory, 1930–1957. In: Special Volume of the Philological Society. Blackwell, Oxford (1957)
21. Miller, G., Charles, W.: Contextual Correlates of Semantic Similarity. Language and Cognitive Processes 6, 1–28 (1991)
22. Li, Y., McLean, D., Bandar, Z.A., James, D.: Sentence Similarity Based on Semantic Nets and Corpus Statistics. IEEE Transactions on Knowledge and Data Engineering 18(8), 1138–1150 (2006)
23. Ferrer i Cancho, R., Sole, R.V.: The small world of human language. Biological Sciences 268(1482), 2261–2265 (2011)