Research on Semantic Text Similarity Based on Lexical Semantic Web and Deep Learning Semantic Model

Fanzeng Xia [a, *]

Department of Computer Engineering, Queen's University, Kingston K7L 3N6, Canada

Department of Software Engineering, Jilin University, Changchun 130012, China

[*, a] Corresponding email: 17fx@queensu.ca

Abstract

The primary problem of text clustering is the mathematical representation of text data. So far, most text clustering algorithms are currently based on the Vector Space Model (VSM). This paper analyzes the characteristics of short text and proposes a semantic vectorization model of short text based on deep learning and dependent syntactic features. On the basis of this, this paper combines the multiple feature representations of short texts and proposes a short text semantic similarity calculation model for multi-feature fusion. Traditional text processing methods will lead to sparse representation of text features and loss of semantic information due to the diversification of short text expressions and irregular grammatical structure, resulting in the failure of dictionary matching and the appearance of unknown words in Chinese word segmentation, and the lack of semantic representation of Chinese words, which makes the traditional methods not completely applicable to the analysis and calculation of short text. None of the clustering algorithms based on vector space model can solve the two natural language problems that are unique to text data: synonyms and polysemous words. All these problems greatly interfere with the efficiency and accuracy of text clustering algorithm and degrade the performance of text clustering. In view of text semantic similarity computation for sentence level, this paper proposes a method of applying structural features and neural networks, and applies the method to an actual question and answer system, which has achieved good results.

Keywords

Semantic Similarity; Deep Learning; Text Clustering.

1. Introduction

Text is the most important information carrier in the present world. In fact, research shows that 80% of the information is contained in text documents. Therefore, the processing and analysis of text documents has become one of the hotspots in data mining and information retrieval technology[1]. There are many techniques to process and study text documents, among which the most important one is text clustering. Text semantic similarity calculation, as a basic topic in the field of natural language processing, has important application value in the research direction of text classification, text clustering, information retrieval and automatic question answering system[2]. With the popularity and development of social networks and the widespread use of smart interactive applications, short texts are very common in everyday life. Weibo, commodity reviews, intelligent interactive applications and WeChat all have a very close relationship with short texts[3]. Therefore, as the basic technology of short text information processing, short text semantic similarity computing has a very broad prospect and research value. Especially with the introduction of deep learning and its wide application, the process of language information processing can be changed from the vector space of traditional words

to the word Embedding layer vector space, or more complex neural network hidden layer space. Therefore, how to effectively understand and identify the rich meanings of these short texts through machines has become a difficult and hot topic in the field of natural language processing and machine learning[4].

2. Analysis of Transfer Dependency Syntax Based on Global Structure Prediction Model

Syntactic tree is a very important feature in the process of calculating semantic similarity of short text. It reflects the structure of sentences and the relationship between the components. Transitive dependency syntax has a linear time complexity and a simpler analysis process than the Greedy algorithm, which coexists with graph-based dependency analysis and is widely used. So although this method has an advantage in speed, its accuracy has not been modeled. On the basis of the transfer model, if the search space can be expanded so that every step of the prediction takes into account the global characteristics or the global training is carried out directly, the accuracy of the model will be improved compared with the simple transitive dependency syntax analysis. There are two main types of transitive syntax analysis algorithms. The first one is the Yamada algorithm, which consists of three shift actions: shift, left-arc, and right-arc. It is similar to the Input-Statute analyzer, which builds a dependency syntax tree from the bottom up. The other is the arc-eager algorithm, which consists of four transition actions, shift, left-arc, right-arc, and reduce. These two methods have their own strengths and weaknesses and are inconsistent in different languages.

Using the structured transitive syntax analysis model of the Yamada algorithm, the balance sensor and the ordinary perceptron are trained using beam size=8, and the results are shown in Figure 1.

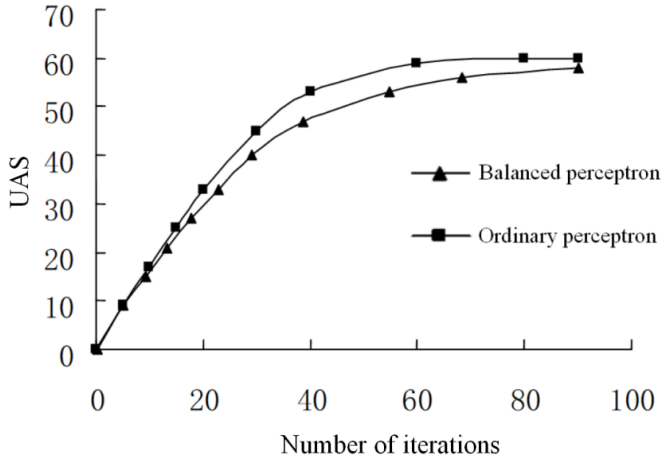

Figure 1 The difference between using balanced perceptron and ordinary perceptron

This paper compares the improved mixing method with three other commonly used methods. Table 1 lists the semantic similarity of words estimated manually and the semantic similarity calculated by edge-based node-based hybrid and improved hybrid methods.

Table 1 Word-to-semantic similarity calculated by M&C, edge-based, node-based, mixing, and extended mixing methods

M&C	Edge-based	Node-based	Mixing method	Extended mixing method
3.26	31	10.57	25	1
3.68	33	9.44	28	0.97
3.24	30	6.55	33	0.89
3.11	29	8.59	32	0.59
2.98	33	5.95	29	1

The greater the correlation coefficient, the greater the correlation between the two vectors. As shown in Table 2.

Table 2 Comparison of node-based, edge-based, mixing and extended mixing methods with M&C

Method of calculating semantic similarity	Relativity
Nodal-based calculation method	0.654
Edge-based calculation method	0.812
Mixing method	0.891
Improved mixing method	0.986

The error rate of the two optimization algorithms decreases in the same number of iterations in the sample, as shown in Figure 2.

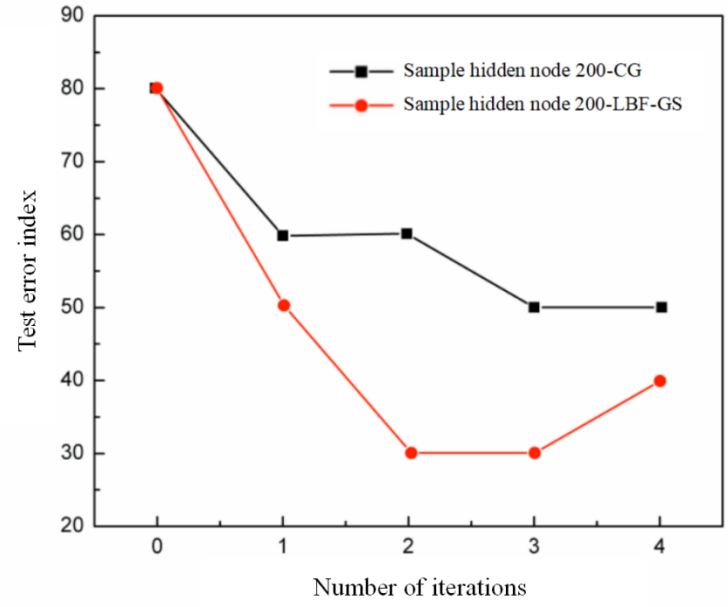

Figure 2 Comparison of Chinese word segmentation models affected by optimization algorithms

The error rate of different context-range models decreases in the same number of iterations in the sample, as shown in Figure 3.

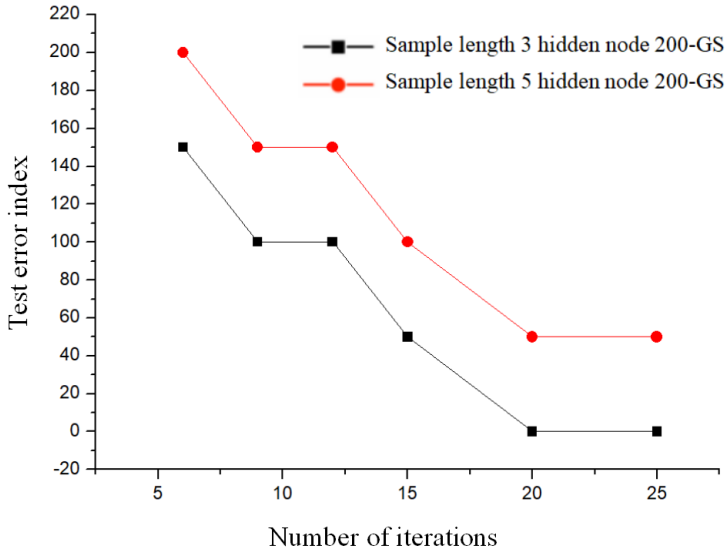

Figure 3 Comparison of Chinese word segmentation models influenced by context

Feature selection is a very important step in transitive dependency syntax analysis. Usually features can be extracted from the stack as well as from the input queue. These features are mainly derived from words, parts of speech and dependencies. In the structured transitive dependency syntax analysis model, the price of words, the length of arcs and other non-local features can be combined. Generally speaking, part of speech is the most important feature in transitive dependency syntax analysis. Part of speech is not only a generalization of words, but also can provide a very important morphological features of dependent syntax analysis. The most obvious is the noun form of the verb. Part of speech tagging uses window features of word sequences, while transitive dependency syntax analysis uses the structural features of trees, which are quite different from each other. Some contextual features that can be used in part of speech tagging may not be used in dependent parsing. While implementing dependency parsing, this paper attempts to remove all features that contain part of speech, leaving only the features that contain the words themselves, resulting in a UAS falling from 85 to around 65. It can be seen that the effect of part of speech on the syntactic effect of dependency. The word itself is also an important feature in the syntactic analysis of dependency. The usage of different words is quite different, and it is difficult to refine these differences when the set of parts of speech is small. For example, whether it is the transitive verb "eat", "play" or intransitive verb "singing", "dancing", etc., their part of speech is the verb "VV" in the labeling system of the Pennsylvania tree library. Therefore, the word itself is a very important feature that must be used. However, the influence of lexical features on dependent syntactic analysis is lower than that on part of speech. After removing all the features that contain the words themselves, UAS can reach about 70.

3. Text Clustering Algorithm Based on Semantic Similarity

The primary problem of text clustering is the mathematical representation of text data. At present, most text clustering algorithms are based on Vector Space Model (VSM). This method of text representation is very simple, but it leads to the problem of high dimensional sparsity. Moreover, the clustering algorithms based on vector space model do not solve the two natural language problems: synonyms and polysemous words which are unique to text data. All these problems greatly interfere with the efficiency and accuracy of text clustering algorithm and make the performance of text clustering worse. Although vector space weight adjustment and dimensionality reduction are proposed to solve these problems, these methods have their own shortcomings. Although vector space weight adjustment and dimensionality reduction are proposed to solve these problems, these methods have their own shortcomings. Although dimensionality reduction method solves the problem of high dimensional sparsity, the cost of dimensionality reduction is usually very high. In addition, most of the existing text clustering algorithms do not give the method of clustering description. A text clustering algorithm based on ext Clustering Using Asymmetric Proximity (TCUAP) is a hierarchical clustering algorithm. TCUAP is based on graph analysis and uses asymmetric similarity between documents in the process of merging hierarchical clusters. Finally, it is found that the cluster is the strong connected component in the similarity matrix. The disadvantages of this algorithm are: firstly, in a cluster, the two nodes directly adjacent to each other are similar, but the non-adjacent nodes are not necessarily similar; secondly, the algorithm is still based on vector space model and does not solve the problems of high dimensional sparse, polysemous and synonyms. Drawing on the idea that TCUAP is based on analysis, and trying to overcome the problem that non-similar texts are classified into one kind due to transitivity, this paper proposes the TCUSS (Text clustering using semantic similarity) algorithm by using text semantic similarity as a metric.

The effects of the two algorithms are compared in different language tree libraries. The effects of the dependent tree tree transformed into the Treetree of Academia Sinica are shown in Table 3 and 4.

Table 3 Arc-eager and arc-standard algorithm comparison

	UAS	LAS
arc-eager	90.56	90.64
arc-standard	92.56	88.59

Table 4 Arc-eager and arc-standard algorithm comparison

	UAS	LAS
arc-eager	80.61	79.51
arc-standard	82.97	85.61

The only difference between the tree library and the segmentation method is that the linear kernel support vector machine and its own feature set are used. In this paper, the characteristics of the balance sensor and the transition part are used for experiments. The experimental results are shown in Table 5.

Table 5 Comparison of arc-eager and arc-standard algorithms in the analysis model implemented in this paper

	UAS	LAS
O-eager-b1	80.26	80.55
O-eager-tc-b1	82.56	78.15
O-yamada-b1	79.69	80.51

Figure 4 and Figure 5 show the comparison of the classification accuracy of TCUSS with K-Means and Bisecting K-Means algorithms on RCV1 and 20Ng, respectively.

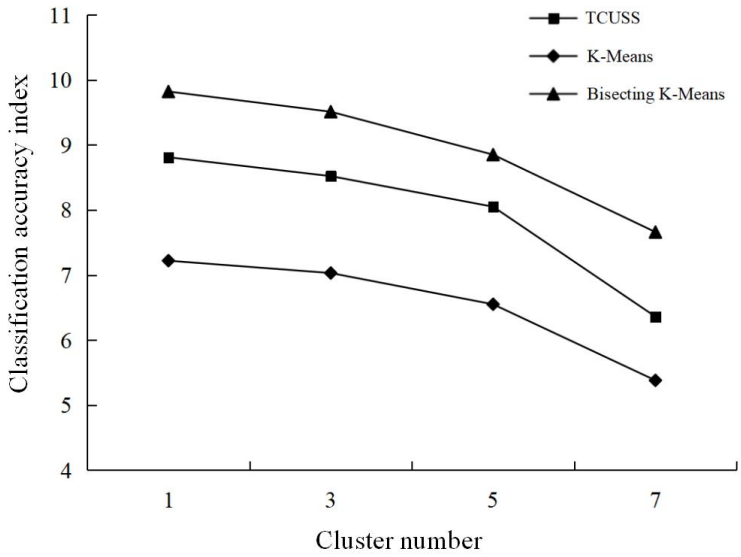

Figure 4 Comparison of TCUSS and K-Means and Bisecting K-Means algorithms on RCV1

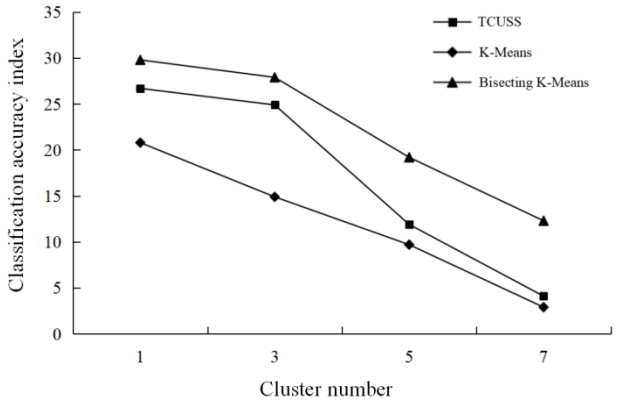

Figure 5 Comparison of TCUSS with K-Means and Bisecting K-Means algorithms at 20Ng

The TCUSS algorithm uses conceptual lists to represent documents and semantic similarity between documents as a measure of the correlation between documents. The similarity matrix is a symmetric matrix. If the semantic similarity of two words in the text is not zero, the semantic similarity of the two words is not zero, so the initial similarity matrix may be a connected graph. Therefore we use a split (top-down) hierarchical clustering idea to reduce the time complexity of the algorithm. In each split, a matrix element that does not satisfy the threshold is set to a flag indicating that the two nodes are no longer connected to each other. If the number of connected components of the reconstruction matrix is greater than the number of input clusters K, then the cycle is stopped, otherwise the splitting continues. When the splitting stops, in order to solve the problem that the non-adjacent nodes in a cluster are not necessarily similar, we obtain a complete graph containing the largest number of nodes in the connected graph corresponding to each cluster. This ensures that each node in the cluster must be similar. Finally, the similarity between each non-cluster node and each cluster is calculated and classified into the closest cluster.

Figure 6 shows the comparison of TCUSS and UPGMA, TCUAP algorithms on RCV1.

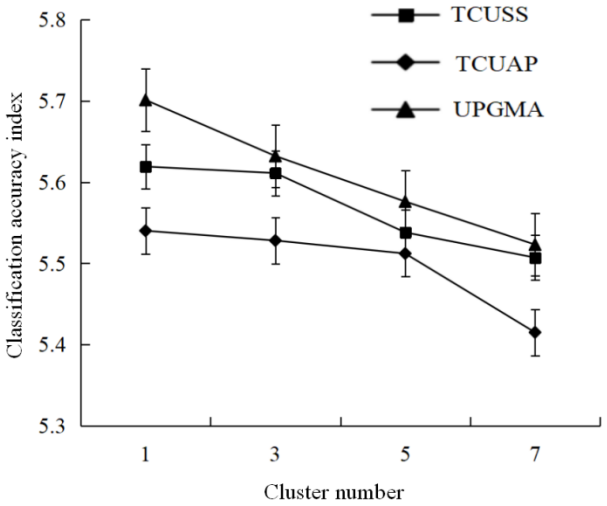

Figure 6 Comparison of TCUSS with UPGMA and TCUAP algorithms on RCV1

4. Conclusion

In this paper, a conceptual list representation is proposed, since the vector space model leads to the problem of high dimensional sparsity, and it is not conducive to semantic computing in the semantic network. The concept list is a triad list that contains the words, word frequencies, and the meaning of the words that have appeared in the text. Since the calculation of semantic similarity is aimed at nouns, the list of concepts contains only noun words that appear in the text. This paper studies the calculation of semantic similarity of short text based on the characteristics of short text. First of all, the text

analyzes the characteristics of short text sparseness and noise, and considers that the semantic similarity calculation of short text requires the use of syntactic features and the Embedding representation of words. Then, based on the global structure prediction model and Yamada dependent syntax analysis algorithm, this paper proposes a structured transitive syntax analysis based on Yamada algorithm, and optimizes the feature selection in this algorithm. Most current text clustering algorithms are based on the Vector Space Model (VSM). This text representation is very simple, but it raises the problem of high dimensional sparsity. Moreover, it can not solve the two natural language problems: synonyms and polysemous words. The TCUSS algorithm classifies a piece of text into only one class, but some texts may belong to more than one class. For example, this paper and the semantic similarity can be related to the content of clustering algorithm, so it can belong to multiple classes.

References

[1] Huang C H, Yin J, Hou F. A Text Similarity Measurement Combining Word Semantic Information with TF-IDF Method[J]. Chinese Journal of Computers, 2011, 34(5):856-864.

[2] Mcinnes B T, Pedersen T. Evaluating measures of semantic similarity and relatedness to disambiguate terms in biomedical text[J]. Journal of Biomedical Informatics, 2013, 46(6):1116-1124.

[3] Kashyap A, Han L, Yus R, et al. Robust semantic text similarity using LSA, machine learning, and linguistic resources[J]. Language Resources & Evaluation, 2016, 50(1):125-161.

[4] Hua X L, Zhu Q M, Pei-Feng L I. Chinese text similarity method research by combining semantic analysis with statistics[J]. Application Research of Computers, 2012, 29(3):833-836.