Chinese Sentence Similarity Calculation Based on Multi-Features Fusion

Hongbin Wang*, Yonghua Wen†, Yantuan Xian†, Runxin Li†

1. Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming, China

Abstract: Sentence similarity computation plays an increasingly important role in text mining, Web page retrieval, Machine Translation, Speech Recognition and Question-answering systems. Existing methods for computing sentence similarity just unilaterally consider word, semantic, sentence structure, syntactic structure or word order, did not consider the feature information of the sentence as a whole. In order to improve the accuracy in sentence similarity calculation, we proposed a new approach to compute the sentence similarity based on more features fusion. Which is divided into three steps: firstly, obtain semantic vector similarity between sentences based on word2vector; secondly, obtain syntactic structure similarity between sentences based on syntactic structure; thirdly, calculate word order similarity between sentences; finally, combine semantic vector similarity, word order similarity and syntactic structure similarity as the final similarity between sentences. Experiments show that our method has relatively higher accuracy in Chinese sentence similarity computation and is feasible.

Keywords: sentence similarity; Words2Vector; semantic vector; word order similarity; syntactic Structure

1. Introduction

In the field of natural language processing, calculation of sentence similarity is a basis and a key research topic. Sentence similarity computation has a wide range of applications in the real world, its research directly determine a number of other research in related fields [1], such as in the document summarization system [2], in the machine translation system [3], in document digest system [2], document classification system [4] and information retrieval system.

Traditionally, techniques for detecting similarity between documents have centered on analyzing shared words. Such methods are usually effective when dealing with long texts because similar long texts will usually contain a degree of same words. However, in sentences, word co-occurrence may be rare. The traditional text similarity calculation method is difficult to application to sentence similarity calculation, this problem poses a difficult computational challenge [5].The focus of this paper is on computing the similarity between Chinese language sentences.

Some methods have already been used to compute the sentence similarity, especially in the Chinese sentence similarity calculation from domestic and foreign scholars are as follows.

Liu Yi et al. [6] depict the sentence information from the surface structure, the structure features and semantic features three aspects. R. B. Wang et al. [7] proposed combined word string granularity and structure to calculate sentence similarity, considering the number of the same word string and length and the corresponding weights information. Y. H. Li et al. [8] calculate the similarity between sentences or passages based on the semantic and word sequence information, this method considers the word order. B. Li et al. [9] took a Chinese sentence similarity computing method based on semantic dependency, the method combine analysis of semantic and dependency grammar, effectively depict the expression meaning of the sentence, but the dependent analysis accuracy rate will affect the accuracy of sentence similarity calculation. F. G. Zhou et al. [10] through keyword extraction, and expand the synonyms dictionary and increase the importance of nouns and verbs in sentences to improve the accuracy of calculation, the quality of automatic word segmentation and part-of-speech tagging also directly affect the accuracy. In ZHOU’s paper, the method improved the accuracy of sentence similarity calculation, but didn’t to detail analysis the grammar, syntax, semantics of the sentence. Y. L. Lan et al. [11] proposed a similarity measure method to compute sentence similarity based on POS and POS dependency, it improved the accuracy of the sentence structure similarity, but didn’t to analysis the semantic. Y. Yu et al. [12] proposed sentence similarity method of key words weight can significantly improve the accuracy of the calculation by keyword extraction and POS assignment, increasing the importance of nouns and verbs in the sentence and taking into account two aspects of the syntax and semantics in sentence similarity. J. L. Zhao et al. [5] proposed a method for measuring the similarity between sentences, based on semantic and word order information, but didn’t to analysis the syntactic structure. L. Wang et al.[1] proposed an improved method of Chinese sentence similarity computation, uses tree kernel algorithm to calculate the sentence structure similarity. H. Ren et al. [13] consider the impacts of membership degree, nonmember ship degree and median point of interval-valued intuitionist fuzzy (IVIF) sets for similarity measure. Although there are many methods of computing sentence similarity, but most the methods only take word information into account and have a high complexity along with a low accuracy rate. Those methods have been used to compute the Chinese sentence similarity did not consider the sentence feature information of semantics, sentence structure and word order as a whole. M. J. Kushner et al. [14] uses the WMD distance to measure the dissimilarity between two text documents as the minimum amount of distance that the embedded words of one document need to “travel” to reach the embedded words of another document. However, a complete sentence consists of main component and modifier. Main component is usually core verb in the sentence, is the mercy of the sentence, modifier used to describe the context, it belong to domin. The same principal component can be modified by different modifier, in order to achieve different rendering effects [9]. In addition, a sentence of word order between word and word will influence the similarity between the two sentences. For instance [5]:

T1: There is a table on the right side of a chair;
T2: There is a chair on the right side of a table;

If we obtain similarity between T1 and T2 based on morphology similarity is 0.9940909090909091, based on word order similarity is 0.6, based on morphology and word order similarity is 0.9152727272727273, based on semantic vector similarity is 1 and based on syntactic structure similarity is 1. So we draw lessons from the existing Chinese sentence similarity computing method and propose a new approach to compute Chinese sentences similarity, which based on Semantic vector, word order and syntactic structure. Finally, we verify our proposed method’s feasibility by experiments. Experiment results proved that our approach is feasible.

The organization of the article is as follows: In Section 2, we describe the Chinese Similarity Computation Based on Semantic Vector. In Section 3, we describe the Chinese Similarity Computation Based on word order. In Section 4, we describe the Chinese Sentence Similarity Computation based on syntactic structure. In Section 5, the proposed Chinese sentence similarity measure based on Semantic Vector, word order, syntactic structure is described. In Section 6, we discuss the experiment and analysis. Section 7, contains the concluding remarks.

2. Chinese language similarity computation based on semantic vector
Language model building and training is an important part of natural language processing field, common language model are classic N-gram model [15] and the deep Learning model (Deep Learning) [16] [17], etc. Words vector as a kind of deep learning model of distributed representation, it can well solve the data sparse effect on statistical modeling, to overcome the influence of the dimension disaster, and achieved good application effect [18], get the attention of the researchers widely. Hinton proposed adopting word distributed representation to denote words vector, also known as Word Embedding [19]. This method use a set of low dimension real numbers vector to describe the word characteristics, its advantage is mainly manifested in two aspects: one is by calculating the distance between words to measure the words relation or similarity. Another through using low dimension characteristics to depict a word, it can greatly reduce the calculation complexity, thus improve the method practical application value. Word vector numerical general comes from a large number of without annotation text data, by non-supervision language model training.Word2vec is Google in 2013 released word vector training and generation tool, it can effectively generate word vector from a large number of without annotation corpus, and provide the realization of Continuous Bag-of-Words Model(CBOW) model and Skip-gram model [20]. This paper adopts any realization Word2vec in Java version as word vector training tool [21]. The Semantic Vector similarity based on this tool to improve, and realize the word semantic similarity calculation, the word is no overlapping. The semantic vector similarity calculation steps are as follows:

Step1: Calculate the public words between sentences $S_1$ and $S_2$. We do word segmentation and part-of-speech tagging respectively for the input sentences $S_1$ and $S_2$, extract overlapping word list $W$ from sentence $S_1$ and $S_2$. $S_1 = \{w_{i1}, w_{i2}, \ldots, w_{i,j}, \ldots, w_{in}\}$; $S_2 = \{w_{j1}, w_{j2}, \ldots, w_{j,j}, \ldots, w_{jn}\}$; $W = \{w_{ij}, w_{ij}, \ldots, w_{ij}\}$.

Step2: To filter overlapping word in sentence $S_1$ and $S_2$, gain the non-overlapping word list $A$ and $B$ for sentence $S_1$ and $S_2$, if the number of word list $A \geq$ the number of word list $B$, the number of word list $A$ is m, the number of word list $B$ is n. Otherwise, the number of word list $A$ is n, the number of word list $B$ is m, the distance between words matrix as shown in formula (1).

$$ P_{d_{\text{dis}}} = \begin{bmatrix} S_{i1} & S_{i2} & \cdots & S_{in} \\ S_{i2} & S_{i3} & \cdots & S_{in} \\ \vdots & \vdots & \ddots & \vdots \\ S_{in} & S_{i2} & \cdots & S_{in} \end{bmatrix} $$

(1)

Step4: Calculate sum of words list $A$ and list $B$ the each corresponding word minimum distance, the value is $\text{Dis}_{\text{min}} = \sum_{i=1}^{n} \min(d_{ij})$, $j$ is from $1 \ldots n$, $d_{ij}$ is corresponding word distance of $i$ word in $A$ with all the words in $B$.

Step5: Calculation sentence similarity of sentence $S_1$ and $S_2$, two sentences corresponding word vector, the smaller the distance is the greater the similarity, the greater the distance is the smaller the similarity. Word vector distance is in $[0, 1]$, if sentence $S_1$ and $S_2$, each word in the corresponding distance is 1, the similarity of two sentences is 0, and if the sentence $A$ and $B$ each word in the corresponding distance is 0, the similarity of two sentences is 1. The sentence $S_1$ and $S_2$ similarity calculation formula as shown in formula (2):

$$ \text{SimVec}_{\text{sim}(A,B)} = \begin{cases} 1, \text{ if} \frac{\text{Dis}_{\text{min}}}{m} = 0; \\
0, \text{ else if} \frac{\text{Dis}_{\text{min}}}{m} = 1; \\
\frac{\text{Dis}_{\text{min}}}{m}, \text{ else.} \end{cases} $$

(2)

3. The Chinese sentence similarity computation based on word order

The meaning of sentence not only determines by implication of words contained in sentence but also relates to the order of words.

All of the words in $S_1$ and $S_2$ are identical, $S_1 = \{w_{11}, w_{12}, \ldots, w_{1,j}, \ldots, w_{1n}\}$ and $S_2 = \{w_{h1}, w_{h2}, \ldots, w_{h,j}, \ldots, w_{hn}\}$, while implications of $S_1$ and $S_2$ are totally different. Therefore, a computational method for sentence similarity should take impact of words order into account.

Given two sentences, the intersection word sets of them contains all the common words from both sentences. For the example pair of sentences $S_1$ and $S_2$, the intersection word set is:

$$ S = \{w_{11}, w_{12}, \ldots, w_{1,j}, \ldots, w_{1n}\}.$$
Every word of intersection set $S$ presents in $S_1$ and $S_2$, therefore there is a mapping relationship between the position of words in $S_1$ and $S_2$. The word $w_1$ in example pair of sentences is the first element in $S_1$ and the sixth element in $S_2$, so we can easily get words order vectors by location mapping relationship:

$$r_1 = [1, 2, 3, 4, 5, 6]$$
$$r_2 = [6, 2, 3, 4, 5, 1]$$

Thus, a word order vector is the basic structural information carried by a sentence. The task of dealing with word order is then to measure how similar the word order in two sentences is. We propose a measure for measuring the word order similarity of two sentences as formula (3):

$$Sim_{max} = 1 - \frac{l - r}{l + r}$$

(3)

4. Chinese Sentence Similarity Computation based on Syntactic Structure

A complete sentence consists of main component and modifier component. Main component is usually core verb in the sentence, is the mercy of the sentence, modifier component used to describe the context, belong to dominate. The same principal component can be modified by different modifier, in order to achieve different rendering effects. Therefore, to overall grasp the sentence meaning, need to know the relationship between the main ingredients and modifier component that is the dependent relationships between continuous word lists. The existing information research interdependency five’s axioms [9] are as follows:

1. A sentence is only one component is independent;
2. The other ingredients directly depend on one particular component;
3. Any one ingredients cannot depend on two or more ingredients;
4. If $A$ component directly dependent on $B$, and $C$ components in the sentence is located between $A$ and $B$, then $C$ directly dependent on $B$, or directly dependent one component between $A$ and $B$ components;
5. The other ingredients in the two sides of center composition don’t relationship to each other.

Sentence component information can be reflected by the POS, the POS of relationship between each component in the interdependence between embodies the integrity of the sentence, and the distance between the words reflects the continuity of a sentence. By analyzing the POS and the POS dependency to compute sentence structure similarity, the similarity of surface structure and sentence structure can be obtained. We reference LAN Y.L. et al. [11] proposed sentence structure similarity calculation method based on the POS and POS dependency, the method from the forward and reverse comparative sentence POS sequence, obtained the optimal matching of two sentences POS and the POS dependency, thus the sentence structure similarity is calculated. The method that the sentence is mainly composed of principal component and modifier; principal components shall be a core verb in the sentence as the dominator of the sentence, modifier component as a dominator. The same principal components can be different modifier modification, achieve different results. Therefore, we use the POS and POS dependency information to grasp the sentence similarity.

In this paper, the sentence structure similarity is defined as the optimum matching degree of POS and POS dependency between two sentences, the optimum matching degree value in $[0, 1]$. When the value is 0, it indicates that two sentences in the POS and POS dependency are completely different. When value is 1, it shows that two sentences on the POS and POS dependency are exactly same.

After data preprocessing, we set long sentence is $(l_1, l_2, \cdots, l_m)$ and the short sentence is $(s_1, s_2, \cdots, s_n)$, among them, $m$ is the total words number of long sentence; $n$ is the total words number of short sentence, and $m > n$. The two sentences POS similarity matrix as shown in formula (4):

$$P_{sim\{any\}} = \begin{bmatrix}
S_{11} & S_{12} & S_{13} & \cdots & S_{1n} \\
S_{21} & S_{22} & S_{23} & \cdots & S_{2n} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
S_{m1} & S_{m2} & S_{m3} & \cdots & S_{mn}
\end{bmatrix}$$

(4)

Where, the $S_{ij}$ denotes the similarity of the $i$ word in long sentences and the $j$ word in short sentence, if two words have the same POS, the $S_{ij}$ is 1; Otherwise the $S_{ij}$ is 0. The definition and sentence structure similarity calculation method is given below. The word in short sentence first found in long sentences in the same POS, the word POS is $sF$, its position in the short sentence is $sT$, its position in the long sentence is $lT$. $sCurPOS$ expresses the current POS in short sentence, initial value is $sF$, $lCurPOS$ expresses the current POS in long sentences, initial value is $sF$.

Definition 1: the corresponding word, if the POS in long $lT$ position same as the POS in short sentence $sT$ position, we called the word is corresponding word, otherwise the word is partial corresponding words.

Definition 2: adjacent corresponding word spacing is $d$, initial $d$ is 0. If the short sentence POS in $sT + 1$ location same as the long sentence POS in $l$ location, and $lT + 1 \leq i < m$, the POS in the long sentence $l$ position is the corresponding word in the short sentence $sL + 1$ position, then $d = i - lT - 1$. Otherwise, the POS in the long sentence $lT + 1$ position is the partial corresponding word in the short sentence $sL + 1$ position, $d + 1$.  

Definition 3: surplus words at the beginning of short sentence, \( s^{Prec} \) is the number of surplus words at the beginning of short sentence, if \( s^T > IT \), \( s^{Prec} = s^T – IT \), the words from the short sentence zero position to \( s^{Prec} - 1 \) position are the surplus words at the beginning of short sentence.

Definition 4: surplus words at the end of short sentence, \( s^{Suf} \) is the number of surplus words at the end of short sentence, if the short sentence the last POS without a corresponding word in the long sentences, and the last word in short sentence with corresponding word POS in the long sentence of the \( i \) position, then \( s^{Suf} = n - i - 1 \), the words from the short sentence \( i + 1 \) position to \( n - 1 \) position are the surplus words at the end of short sentence.

Definition 5: matching path, each POS in the short sentence, it is the location and dependency relationship of the corresponding word in a long sentence is called matching path.

Two sentences structural similarity is \( stu_{wm}(S_1, S_2) \), the sentences structure similarity calculation formula (5):

\[
stu_{wm}(S_1, S_2) = \frac{\sum W_i}{\sum \frac{W_i}{2} + \sum W_j + \sum W_k}
\]

Which, the \( C \) is the number of same POS in \( S_1 \) and \( S_2 \), \( W_i \) is the same word for the POS weight, \( R \) is the total number of corresponding words and partial corresponding words; \( \sum W_j \) is the sum of corresponding words POS weight; \( \sum W_k \) is the sum of surplus words at the beginning of short sentence and surplus words at the end of short sentence POS weight; \( n \) is the sum no corresponding word in long sentence with the weight of POS.

The syntactic structure similarity calculation algorithm is described as follows:

Input: calculate sentence structure similarity of the two sentences \( S_1 \) and \( S_2 \).

Output: syntactic structure similarity between sentence \( S_1 \) and \( S_2 \).

Step1: data preprocessing.

After Chinese sentence \( S_1 \) and \( S_2 \) for word segmentation and POS tagging, then extraction the POS sequence of two sentences.

Step2: calculation POS similarity matrix of sentence \( S_1 \) and \( S_2 \).

We get each POS from short sentence and long sentence, then computing the two words POS similarity, the two words which one is from short sentence and anther is from long sentence, \( P_{sim(men)} \) is the two sentence POS similarity matrix. We traversal the POS similarity matrix by row priority traversal, and save the column \( j \), which is the first 1 in the POS similarity matrix by row priority traversal, then save all the 1 position in the column \( j \), the number of 1 in column \( j \) is the number of matching path.

Step3: calculate the structural similarity of \( S_1 \) and \( S_2 \).

If: all the values is 1 in the POS similarity matrix \( P_{sim(men)} \) for sentence \( S_1 \) and \( S_2 \), the sentence structure of \( S_1 \) and \( S_2 \) are exactly same, the sentence structure similarity is 1, finish.

Else if: all the values is 0 in the POS similarity matrix \( P_{sim(men)} \) for sentence \( S_1 \) and \( S_2 \), the sentence structure of \( S_1 \) and \( S_2 \) are exactly different, the sentence structure similarity is 0, finish.

Else: Search matching path, respectively from the forward and reverse search, calculation each matching path structure similarity value, finally choose the maximum similarity value from the forward maximum structural similarity value and reverse maximum structure similarity value as the structure similarity value of \( S_1 \) and \( S_2 \).

The forward search:

(1) If there is only one that the column \( j \) value of 1, it shows only one matching path, according to the formula (5) to compute this matching path structure similarity, matching path structure similarity value as the structure similarity value of sentence \( S_1 \) and \( S_2 \).

(2) If there is more than one that the column \( j \) value of 1, according to the formula (5) to calculate each matching path structure similarity, the maximum similarity value is the best matching path, the maximum similarity value as forward search structural similarity value.

The reverse search:

We make the sentence \( S_1 \) and \( S_2 \) POS sequence reverse, repeat steps (2) and (3) to calculate the reverse maximum structural similarity value.

5. Chinese language sentence similarity computation based on semantic vector, word order and syntactic structure

As we know, the meaning of sentence can be expressed from many levels. The sentence similarity can be calculated by using semantic vector, word order and syntactic structure information of sentence. Therefore, in order to acquire the most similarity of sentences, this paper proposes an improved method that is combining the word order and syntactic structure with the Semantic Vector. In this way, not only...
consider word order and syntactic structure, but also take the semantic into account. Our method for measuring the similarity of two sentences as formula (6):

\[
Sim(S_1, S_2) = 0.4 \times SimVec_{an}(S_1, S_2) + 0.3 \times Sim_{ord}(S_1, S_2) + 0.3 \times Sim_{synt}(S_1, S_2)
\]  
(6)

The process of overall algorithm is as following figure 1:

![Figure 1: Similarity calculation process of proposed method.](image)

6. Experiment and analysis

In order to verify the accuracy and feasibility, the test set is used to evaluate the performance of the proposed method. We select one group test data, which has five sentences. We compare the performance of our method with other five methods. The first method is the syntactic structure method. The second method is semantic vector method, which is based on Words2Vector. The third method is word method. The fourth method is word order method. The fifth method is word and word order method. We use the ansj to word segmentation and part-of-speech tagging [22]. The test data for the experimental comparison results as shown table 1.

<table>
<thead>
<tr>
<th>The source and the similar sentences</th>
<th>The part of speech sequence</th>
<th>Sentence similar</th>
<th>our method</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Source sentence:</strong> 指控被告人崔世亮犯故意伤害罪 (Charged the defendant Cui Shiliang intentional injury)</td>
<td>指控/v, 被告人/n, 崔世亮/nr, 犯/v, 故意/ad, 伤害罪/n</td>
<td>——</td>
<td>——</td>
</tr>
<tr>
<td><strong>Sentence A1:</strong> 指控被告人崔世亮犯故意伤害罪 (Charged the defendant Cui Shiliang intentional injury)</td>
<td>指控/v, 被告人/n, 崔世亮/nr, 犯/v, 故意/ad, 伤害罪/n</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td><strong>Sentence A2:</strong> 判处被告人林勇犯故意伤害罪 (committed the defendant Lin yong intentional injury)</td>
<td>判处/v, 被告人/n, 林勇/nr, 犯/v, 故意/ad, 伤害罪/n</td>
<td>1.0</td>
<td>0.92</td>
</tr>
<tr>
<td><strong>Sentence A3:</strong> 指控被告人崔世亮犯故意伤害罪罪名成立 (The defendant Cui Shiliang guilty of intentional injury was established)</td>
<td>指控/v, 崔世亮/nr, 犯/v, 故意/ad, 伤害罪/n, 罪名/n, 成立/n</td>
<td>0.14</td>
<td>0.86</td>
</tr>
<tr>
<td><strong>Sentence A4:</strong> 代理人赵鸿江 (Agent Zhao Hongjiang)</td>
<td>代理人/n, 赵鸿江/nr</td>
<td>0.5</td>
<td>0.33</td>
</tr>
<tr>
<td><strong>Sentence A5:</strong> 延庆县人民检察院指派检察官聂智慧出庭支持公诉 (Yangling county people's procuratorate appointed protonercs NieZhiHui support of public prosecution in the court)</td>
<td>延庆县/n, 人民检察院/n, 指派/v, 检察官/n, 聂智慧/nr, 出庭/v, 支持/v, 公诉/n</td>
<td>0.2</td>
<td>0.75</td>
</tr>
</tbody>
</table>

We select 200 sentences of news corpus as test data, sentence from the sentences in the news corpus for 100 different structures of sentences as the source sentence, the remaining 100 sentences as a test set. Convenience to calculate the sentence similarity the person’s name distribution value 9 to nr (name), distribution value 7 to n (noun), distribution value 4 to v, vi, vn (verb), r, rr(pronoun), distribution of weight 2 to p (preposition), distribution value 1 to other parts of speech.

In order to compute the accuracy rate by the number of correct results and the total number of sentences related to the experiment. The Accuracy rate calculation formula as formula (7).

\[
\text{accuracy rate} = \frac{\text{correct results number}}{\text{test sentences number}} \times 100\%
\]  
(7)

In order to verify our method’s accuracy and feasibility, We compare the performance of our method with the test results are shown in table 2.
From the results above, it can be said the method proposed in this paper is better than the other methods; the reason is that it not only considers the semantic meaning, but also take into account syntactic structure and word order information. In a word, the experimental result shows that our method is feasible, but some further work will need to further improve the accuracy.

7. Conclusions

This paper proposes a novel method for Chinese language sentence similarity computation. The method is divided into three steps: first, calculates semantic similarity based on words2vector, second, uses syntactic structure to calculate the sentence structure similarity, third, calculates order similarity based on word. At last, we combine the semantic vector, word order and syntactic structure to calculate the sentence similarity. Experiment shows that the method in Chinese language sentence similarity computation is feasible.

Much further work will be needed. We should improve the accuracy of similarity computing by improving the accuracy of word segment, POS tag, semantic similarity and increase the data size of training word vector.

Acknowledgements

This work is supported by the National Nature Science Foundation of China (61462054, 61363044); the Science and Technology Plan Projects of Yunnan province (2015FB135); the Key Project of Yunnan province Education Department (2014Z021); the Nature Science Research Foundation of Kunming University of Science and Technology (KKSY201403028).

References

[20] https://code.google.com/p/word2vec
[22] https://github.com/NLPchina/ansj_seg

Table 2 The sentence similarity experiment results

<table>
<thead>
<tr>
<th>method</th>
<th>accuracy rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>syntactic structure</td>
<td>65.6</td>
</tr>
<tr>
<td>semantic vector</td>
<td>82.3</td>
</tr>
<tr>
<td>word</td>
<td>73.8</td>
</tr>
<tr>
<td>word order</td>
<td>62.4</td>
</tr>
<tr>
<td>word and word order</td>
<td>75.6</td>
</tr>
<tr>
<td>our method</td>
<td>85.2</td>
</tr>
</tbody>
</table>