Semantic Matching Similarity Algorithm Based on Dependency Trees

Kun HE¹,*, Wei Li² and Bo HUANG³

¹College of Computer Science and Technology, Changchun Normal University, Changchun, China
²College of Computer Science and Technology, Jilin University, Changchun, China
³College of Computer Science and Technology, Jilin University, Changchun, China

*Corresponding author

Keywords: Semantic matching, Similarity measure, Dependency tree.

Abstract. The measure of semantic matching similarity is widely used in question answering system, knowledge discovery, search engine and other fields. This paper presents a semantic matching similarity algorithm based on dependency trees, through the analysis of statements to generate a new dependent tree model. Through experimental analysis, our algorithm has a better correct rates as compared to the traditional algorithm at a reasonable weight.

Introduction

Similarity is a very abstract mathematical definition. It must have complex semantic analysis of sentences, and then it can be calculated through a certain mathematical formula, which is generally applied in natural language processing, such as question answering system [1,2].

Sentence similarity computation is a core technology of natural language processing [3]. It is especially important in the question answering system studied in this paper. The calculation method adopted directly relates to the accuracy of answer extraction. Here is a detailed description of all the current computing methods for similarity.

1. Similarity calculation based on key words.

Keyword based similarity is calculated by vector space model to calculate, it only depends on the surface information of the sentences, just to participle the sentence, the sentence as a linear sequence of words, not grammatical analysis deep for a statement, but according to the sentence similarity of word frequency and other related information to calculate sentence [4].

2. Similarity calculation based on semantic.

Semantic based similarity computation is mainly based on semantic dictionary. Because of the existence of synonyms, such semantic similarity computation can further improve the similarity between sentences. The algorithm analyzes and processes the words in sentences deeper, and identifies the words which are irrelevant on the surface, but actually the words with the same meanings are used to calculate the similarity between sentences by semantics [5].

3. Similarity calculation of collocation pairs based on syntax

The collocation of the syntactic similarity calculation is proposed based on syntactic parsing based on the algorithm between words in a sentence are applied to the dependency relation similarity calculation, the algorithm is relatively simple and effective according to the collocation of the words in the sentence to calculate. Effective collocation is the key word that is obtained after parsing. The whole sentence is expressed as a dependency tree. The core word of the sentence is the root node of the tree [6].

In this paper, in order to improve the accuracy of answer extraction, similarity algorithm is applied in the algorithm, which reflects the degree of semantic matching between users' questions and candidate answers, and returns the correct answer to the user. Similarity is a floating-point number between 0 to 1, two sentences through the similarity calculation results obtained greater mean higher matching degree between the two, such as “I love to eat potatoes” and “I love potato”, after the
semantic analysis, the similarity calculation between 1 was obtained, so semantic conclusion the two are the same.

The paper is outlined as follows: Section 2 presents the algorithm. Section 3 reports detailed comparative experiments. In Section 4, we conclude and give some directions for future work.

The Algorithm

Several algorithms proposed above, some only consider the sentence words, no semantic processing, and some did not use the syntactic structure of a sentence, the advantages and disadvantages of the above methods, this paper proposes a matching similarity algorithm based on semantic dependency tree, the calculation method of similarity algorithm based on semantic analysis and syntactic combination. Used to calculate the question and answer sentence similarity, greatly improves the accuracy of the answer.

Similar to syntactic similarity computation, this method first calculates the dependency tree of sentences, and gets the effective collocation of sentences by dependency tree. The effective words and collocation pairs basically represent the core meaning of sentences. In this algorithm, to generate a dependency tree, the root node of the dependency tree as the core, the child node for those dependent on the core words, such a dependency tree height is s, and then to calculate the semantic similarity hierarchical dependency tree, the first layer is the core of the sentence. The second layer is attached to the core word, so the dependency tree before two, can represent the core content of the sentence, so we only consider the first two layers to dependency tree.

When calculating word similarity, we use synonym dictionaries as semantic knowledge resources to calculate word similarity by computing the semantic distance between two words. After segmenting the sentence and removing redundant information, we can find the semantic encoding of these keywords in the synonym dictionary, and calculate the semantic distance between words and words by analyzing the encoding.

In the synonym dictionary, the semantics of words are divided into 5 layers, each is given one or more semantic codes according to their own semantics. The corresponding semantic encoding rules of words in 5 layers, the semantic code format: <1 layer: capital letters > < 2 Layer: lowercase letters > < 3 layer: numbers > < 4 layer: capital letters > < 5 layer: numbers >.

The First Layer Similarity Calculation Method of Dependency Tree

Because there is only one word on the first level, the first level words of two trees are encoded by A and B respectively. Their semantic distance and similarity can be calculated according to the following formula:

\[
Dist(A, B) = \min\{(dist(m, n))\} \tag{1}
\]

Among them, m and N belong to the semantic set of A and B two words. Because a word can have multiple semantic codes, it is necessary to choose the least distance one as the semantic distance of two words, while the distance formula between semantic m and N is shown as follows:

\[
dist(m, n) = 2 \times (7 - s) \tag{2}
\]

S is the first letter or digit position in the semantic encoding of two words. If the semantic encoding is exactly the same, the semantic distance of S=7 is 0. If the encoding is different, it is calculated according to the above formula. For example, the semantic code of “biology” is Ba02A01, and the semantic code of “animal” is Ba02A02, which is calculated by the above two formulas. For the semantic distance between words, if the distance is 0, the similarity of the two words is 1. If the distance is not 0, the similarity between words can be obtained through the following formula:

\[
Similarity(A, B) = \frac{a}{a + Dist(A, B)} \tag{3}
\]
The value of a is set according to the calculation process, and the similarity between two sentences can be calculated by similarity between words. This value is the similarity of the first level words of dependency tree.

**The Second Layer Similarity Calculation Method of Dependency Tree**

The dependency tree layer second words more, so to calculate the semantic similarity and semantic distance between words, and then gives a comprehensive similarity dependency tree second, assuming that the first second layers of the dependency tree m children nodes, second dependency tree layer second n children nodes, two sentences for S1 and S2, S1 as a sequence of words (M1, M2, M3,..., Mm), the words sequence of S2 (N1, N2, N3,..., Nn), the word Mi (1<= i <=m) and Nj expression (1<= i <=n) of the similarity between the S (Mi Nj), S (Mi, Nj) the calculation method and the calculation of the first layer of single word dependency tree method are the same:

\[ \text{Similarity}(A, B) = \frac{a}{a + \text{Dist}(A, B)} \]  

(4)

Then a m * n order matrix is then recorded to calculate the similarity between the sentence S1 and the S2, as shown below:

\[
M(S_1, S_2) = \begin{bmatrix}
S(M_1, N_1) & \ldots & S(M_1, N_n) \\
\vdots & \ddots & \vdots \\
S(M_m, N_1) & \ldots & S(M_m, N_n)
\end{bmatrix}
\]  

(5)

Based on \(M(S_1, S_2)\), the similarity calculation formula between sentence S1 and sentence S2 is as follows:

\[
\text{Similarity}(S_1, S_2) = \frac{\sum_{i=1}^{m} a_i}{m}
\]  

(6)

\(a_i = \max(S(M_i, N_1), S(M_i, N_2), \ldots, S(M_i, N_n))\). Similarly, we can calculate the similarity between the sentence S2 and the sentence S1. Finally, the average value between the two sentences can get the average similarity between the two sentences, that is, the similarity between the two sentences.

**A Method for Calculating the Comprehensive Similarity of Dependency Tree**

Different application domains have different concerns about the different levels of dependency trees. In this paper, we set weights for each layer of the dependency tree. We get the similarity between two sentences based on the similarity of each layer.

\[
\text{Similarity}(S_1, S_2) = a \times \text{Similarity}_1 + b \times \text{Similarity}_2
\]  

(7)

\(\text{Similarity}_1\) is the first level semantic similarity. \(\text{Similarity}_2\) is the semantic similarity of the second level. A and B are the weights of each layer respectively, and the values of a and B are set up by experimental statistics and analysis.

In the proposed algorithm, based on semantic and syntactic parsing of the sentence between the words of the effective word is not simply consider the interdependence of collocation, nor on the effective collocation of all computing, into the semantic information of those seemingly unrelated words, also want to calculate the similarity between them, which makes the calculation more accurate similarity, of course it is not the perfect algorithm, also need to rely on the abundant degree of accuracy and “Thesaurus” syntactic parsing, but compared with some previous algorithms, its accuracy has been greatly enhanced.
Comparative Experiments

In view of the above four algorithms, this paper implements the C++ development language in the Microsoft visual studio 2015 development environment. Using the lexical and syntactic parser provided by Harbin Institute of Technology, we select 437 questions that Harbin Institute of Technology focuses on, and get the similarity between question sentence and answer sentence by running program. The first three similarity methods are used to calculate the similarity between the question sentence and the candidate answer sentence. Then the average value of the highest similarity returned by the 437 problems is calculated. The result is shown in Table 1.

Table 1. T Analysis result of similarity algorithm.

<table>
<thead>
<tr>
<th>Similarity Algorithm</th>
<th>Average Similarity Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Based on Key Words</td>
<td>0.594</td>
</tr>
<tr>
<td>Based on Semantic</td>
<td>0.641</td>
</tr>
<tr>
<td>Based on Syntax</td>
<td>0.667</td>
</tr>
</tbody>
</table>

In this paper, semantic matching similarity algorithm based on dependency trees were analyzed, for each level are given different weights, is determined in the process of the experiment the weight, setting several groups of weights a, b value in the experiment, then don't calculate the statement similarity. Then calculate the average similarity, as shown in table 2:

Table 2. Similarity of different weight.

<table>
<thead>
<tr>
<th>The value of the weight a</th>
<th>The value of the weight b</th>
<th>Average Similarity Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9</td>
<td>0.1</td>
<td>0.853</td>
</tr>
<tr>
<td>0.8</td>
<td>0.2</td>
<td>0.832</td>
</tr>
<tr>
<td>0.7</td>
<td>0.3</td>
<td>0.817</td>
</tr>
<tr>
<td>0.6</td>
<td>0.4</td>
<td>0.795</td>
</tr>
<tr>
<td>0.5</td>
<td>0.5</td>
<td>0.763</td>
</tr>
</tbody>
</table>

From the above results returned can be seen, the proposed matching similarity algorithm based on semantic dependency tree, if each layer of dependency tree to determine the appropriate weights, the results obtained by the algorithm has improved greatly than the original similarity, proved that this algorithm has a certain practicality.

It analyzes the similarity algorithm from the results, we also analyzed from the answers to the questions the accuracy of the algorithm is set up, all of the questions on the test number is Total_Que_Number, and get the correct answer to the question is Correct_Ans_Number. The calculation method of the accuracy of the answer is Correct_Rate:

\[ Correct \_ Rate = \frac{Correct \_ Ans \_ Number}{Total \_ Que \_ Number} \]  

Through the analysis of 437 problems on the test set, using the semantic dependency tree matching algorithm based on similarity of answer extraction, return the highest similarity candidate answer sentence as the answer to the question, which returns the accurate answer 403 questions, calculation accuracy can reach above 92%.

Summary

In this paper, the answer extraction algorithm in Chinese question answering system is studied, and the general accuracy rate is not very high through the research and analysis of the current answer extraction algorithm. Therefore, this paper proposes a semantic dependency tree matching based on similarity algorithm to calculate the similarity, the syntactic parsing of sentences, sentence
dependency tree, respectively, the similarity calculation of different levels of dependency tree, according to the different levels of dependency set different weights to compute the sentence similarity, the similarity between words by “Thesaurus” calculation. The similarity calculation method proposed in this paper is proved by experiments. Compared with the traditional keyword based similarity computing method, the similarity computation method based on semantic pair is improved, and the similarity degree is improved to some extent based on the similarity computation method of collocation pairs.

This paper studies the answer extraction algorithm, mainly in the method of similarity calculation, this method requires the sentence syntactic parsing accuracy is relatively high, the present research on Chinese syntactic parsing is still in the initial stage, so the research of this aspect is still need deeply.

Acknowledgement

This research was financially supported by Jilin Industrial Technology Research and Development Special Project (2017C31-2).

References


