Semantic Based Text Similarity Computation

Yaqi Liu and Zhijiang Li

Abstract Text similarity algorithm is widely used in plurality fields, such as copy detection, text classification, machine translation, intelligent question answering system and natural language processing. At present, vector space model algorithm, which is more commonly used, does not consider the information of semantic features adequately, and the accuracy of the semantic similarity computation results can be further improved. This paper proposes a text similarity computation method which combines the HowNet with vector space model. Similarity computation is divided into two levels. In the level of words, words-similarity calculation based on HowNet prevents the loss of semantic information. In the level of texts, text-similarity calculation by vector space model ensures the integrity of the information expressed in the texts. This paper designs an experiment of news text classification based on KNN algorithm, in which data obtained from a part of the Chinese news in Sogou data corpora. Experimental results show that the method proposed in this paper is more accurate than the traditional vector space model algorithm.

Keywords Semantic · Text similarity · HowNet · Vector space model

1 Introduction

In daily life, people use text to express and transmit the information, so the information retrieval technology and data mining depends on the processing and operation of text information [1]. Text similarity computation is an effective and direct method to solve the problem of resource acquisition and analysis. Generally, we use text similarity to measure the degree of association between the two different texts. When the similarity value is larger, the correlation degree of the two texts is

Y. Liu · Z. Li (✉)
School of Printing and Packaging, Wuhan University, Wuhan, China
e-mail: lizhijiang@whu.edu.cn

P. Zhao et al. (eds.), *Advanced Graphic Communications and Media Technologies*, Lecture Notes in Electrical Engineering 417,
DOI 10.1007/978-981-10-3530-2_43

higher, and vice versa. Salton et al. proposed vector space model (VSM) which maps the text into a high dimension space vector [2]. In this way, the mathematical computation can be easily carried out, and the computation process is simplified. However, semantic information is often lost in the process of mapping, due to the linear independent hypothesis of the space vector model, which affects the accuracy of the final results.

This paper will introduce the text preprocessing and vector space model, and proposes a computation method of text similarity, which is based on HowNet and vector space model, aiming at the problem of absence of semantic information.

2 Vector Space Model

2.1 Chinese Text Preprocessing

Compared with English text, Chinese text has no space to separate the words. Chinese word segmentation is a technology to segment Chinese text into a series of individual words. There are some words which have a very small contribution to the meaning of the text and appear many times, known as the stop word [3]. These words do not provide any value to the meaning of the text, but also increase the dimension of the text representation. It would increase the computational complexity and time overhead, so it needs to be removed.

2.2 Vector Space Model

Vector space model is simple. It will map a text into a space vector, and use a vector to represent the text information. By using the knowledge of space mathematics, we can directly compute the similarity between two text vectors, which is measured by the cosine value of the angle of the vectors.

The vector space model has a basic assumption, which feature item that can represent text content is only related to the number of times it appears, but has nothing to do with its position and sequence [4]. Vector space model is composed of features and their weights. Features usually refer to non-redundant words after text preprocessing, and each feature corresponds to a weight, which describe the important degree of the feature in the text. When computing the similarity of text, the cosine theorem is generally used, the formula is as follows:

$$\mathrm{Sim}(\mathrm{D}_1, \mathrm{D}_2) = \frac{\sum_{k=1}^{n} \mathrm{W}_{1k} * \mathrm{W}_{2k}}{\sqrt{\sum_{k=1}^{n} \mathrm{W}_{1k}^2} * \sqrt{\sum_{k=1}^{n} \mathrm{W}_{2k}^2}} \tag{1}$$

D_1 and D_2 represent two texts, W_{ik} represents the weight of the k-th feature in the i-th text.

We use the TF-IDF method to calculate the weight. The formula is as follows:

$$W_k = \mathrm{TF}_{(i.k)} * \mathrm{IDF}_{(\mathrm{i.k})} = q * \log\left(\frac{N}{n} + \alpha\right) \tag{2}$$

In Eq. (2), q stands for the number of times that feature T_k appears in the text D_i. N is the total amount of texts, n is the number of texts containing the feature T_k, and α is called empirical constant whose general value is 0.01.

3 Text Similarity Computation

3.1 HowNet

Concepts and sememes are the foundation of HowNet [5]. Concept is the description of the words, and each word can have a number of concepts which are generally described by knowledge description language that is made up of sememes. HowNet uses one or more of the sememes to describe the concept, and a forest model is used to describe the relationship between sememes [6].

There are ten trees in sememe forest. The relationship between sememes is complicated, which is usually gotten from upper and lower relations. This relationship is called sememe hierarchy tree, showed in Fig. 1 [7]. The description of the concept, in addition to the use of sememes, is also applied to a variety of semantic symbols for assist.

3.2 The Similarity of Words

According to the distance between the two sememes on sememe hierarchy tree, the sememe similarity is calculated by Eq. (3).

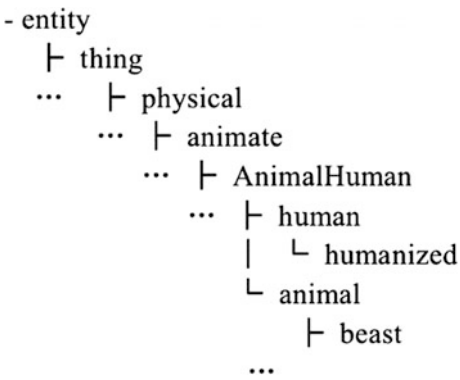

Fig. 1 Sememe hierarchy tree

$$\mathrm{Sim}(p_1, p_2) = \frac{\alpha}{d + \alpha} \tag{3}$$

Two sememes is p_1 and p_2. d is the distance between the two sememes on referenceeme hierarchy tree. If they are on different tree, the value of d towards infinity. α is adjustable parameter which value is1.6.

Because Eq. (3) only considers the upper and lower relations, but not the depth of the sememe, the public father node and the antisense of sememes. It needs to be improved.

$$\mathrm{Sim}(p_1, p_2) = [e^{dis(p_1,p_2)^{-1}}]^{a(p_1,p_2)} \frac{\alpha \cdot c(p_1, p_2)}{\alpha \cdot c(p_1, p_2) + (1 - \frac{dep(p_1,p_2)}{dep(t)} + 0.01) \cdot dis(p_1, p_2)} \tag{4}$$

$$dep(p_1, p_2) = dep(p_1) + dep(p_2) \tag{5}$$

$dep(p_1)$ means the depth of sememe p_1 on the tree. $dep(t)$ is amount of the depth of tree. $dis(p_1, p_2)$ is the distance between the two sememes. $c(p_1, p_2)$ is the amount of the public father nodes of p_1 and p_2. $a(p_1, p_2)$ means whether there is antisense on the path of p_1 and p_2. If there is, the value is 1, otherwise, the value is 0. α is adjustable parameter which value is 1.6.

Knowledge description language is made up of independent sememe description, relation sememe description and symbol sememe description (Table 1).

In general, if two specific words are same, the similarity is 1, otherwise, the similarity is 0.

For independent sememe description, make pairs of sememes or specific words alternately, select the maximum similarity, and record the value. Then select the most similar pairs in the remainder of the match repeatedly until one of them is complete, and discard the remaining sememes and specific words. Finally, carry out an arithmetic average value of maximum similarity values, and record the value as sim_1.

For relation sememe description, if relation is same, make pairs of sememes or specific words alternately, select the maximum similarity, and record the value repeatedly. Get the arithmetic average eventually, and record the value as sim_2.

Table 1 Sememe descriptions

Descriptions	Description method
Independent sememe description	"Sememe", or "(specific word)"
Relation sememe description	"Relation = sememe", or "relation = (specific word)", or "(relation = specific word)"
Symbol sememe description	"Symbol sememe", or "symbol (specific word)"

For symbol sememe description, the processing likes relation sememe description. Here to judge symbol rather than relation, record the arithmetic average value as sim_3.

Formula for calculating the similarity between concept C_1 and C_2 is as follows:

$$\mathrm{Sim}(C_1, C_2) = \sum_{i=1}^{3} \beta_i \prod_{j=1}^{i} sim_j \tag{6}$$

β_i $(1 \le i \le 3)$ is adjustable parameter, and meets the conditions: $\beta_1 + \beta_2 + \beta_3 = 1$, $\beta_1 \ge \beta_2 \ge \beta_3$ General experience is: $\beta_1 = 0.7, \beta_2 = 0.17, \beta_3 = 0.13$.

Assume that the word W_1 has m concepts: $C_{11}, C_{12}, \ldots, C_{1m}$. And the word W_2 has n concepts: $C_{21}, C_{22}, \ldots, C_{2n}$. The similarity between the words W_1 and W_2 is expressed as $\mathrm{Sim}(W_1, W_2)$.

$$\mathrm{Sim}(W_1, W_2) = \max_{i=1\ldots m, j=1\ldots n} \mathrm{SIM}(C_{1i}, C_{2j}) \tag{7}$$

3.3 The Similarity of Texts

Chinese text preprocessing of semantic based text similarity computation is same as text similarity computation based on VSM. Assume two Chinese texts after preprocess and expressed as: $D_1 = (T_{11}, W_{11}, T_{12}, W_{12}, \ldots, T_{1m}, W_{1m})$, $D_2 = (T_{21}, W_{21}, T_{22}, W_{22}, \ldots, T_{2n}, W_{2n})$. $T_{11}, T_{12}, \ldots, T_{1m}$ are feature terms of text D_1, $W_{11}, W_{12}, \ldots, W_{1m}$ are weights of the feature terms.

Use Eq. (7) to calculate the similarity of each feature item between two texts. Then construct feature items similarity matrix A $\left(a_{ij} = \mathrm{Sim}\left(T_{1i}, T_{2j}\right)\right)$. In the matrix A, find the largest element, and compare it with adjustable threshold, the value of the threshold in this paper is 0.7. If the element value is larger than the threshold value, delete the row and column of the element from the matrix A while keeping elements index in matrix A unchanged, and record the row and column of the element. Keep this step repeatedly until all the elements in matrix A are not larger than the threshold value or there is no element in matrix A.

Treat the feature items which the similarity is larger than threshold as the same dimension, and use Eq. (1) to calculate the text similarity.

4 Experimental Results and Analysis

According to the part of the data of Chinese news corpus of Sogou, design an experiment of news text classification based on KNN algorithm which K value is set to 7. The data contains five fields of electronics, sports, tourism, education and

Table 2 The classification accuracy of the experiment

Field	VSM	Equation (3)	Equation (4)
Electronics	0.42	0.52	0.60
Sports	0.72	0.68	0.80
Tourism	0.46	0.54	0.62
Education	0.44	0.60	0.72
Military	0.42	0.62	0.68

military. The total number of text data is 1000 and 200 text data in each field. The training set for each field is 150 text data, and the test set is 50 text data (Table 2).

The classification experimental results show that the two semantic algorithms are much better than the traditional algorithm based on VSM. Besides, the algorithm proposed in this paper is more efficient than the other sematic based algorithm.

5 Conclusions

This paper mainly discusses the text similarity computation based on VSM and the semantic based text similarity computation, and improves the method of computing the similarity between words. Finally, a text classification experiment based on KNN is designed to analyze the advantages and disadvantages of the two different methods of text similarity computation. In the course of the study, we find a problem which is worth studying and solving: considering the order of sequence of words to prevent losing a part of useful text information.

References

1. Jin Xiqian. (2009). Research on Semantic Based Chinese Text Similarity Algorithm. (Doctoral dissertation, Zhejiang University of Technology).
2. G. Salton, A. Wong, ang C.S. Yang, A Vector Space Model for Information Retrieval, Journal of the ASIS, 18:11, 613–620, November 1975.
3. Liu Xiaojun, Zhao Dong, & Yao Weidong. (2007). A Two Factor Similarity Algorithm for Chinese Text Search. Computer Simulation, 24(12), 312–314.
4. Chen Feihong. (2011). Research on Chinese Text Similarity Algorithm Based on Vector Space Model. (Doctoral dissertation, University of Electronic Science and technology).
5. Kuai Yuanyuan. (2014). Research on Semantic Based Text Similarity Algorithm. Computer CD software and Applications (9), 302–303.
6. Liu Qun & Li Sujian. (2002). Based on the HowNet Lexical Semantic Similarity Computation. Chinese of computational linguistics.
7. Fan Hongyi, & Zhang Yangsen (2014). A method for semantic similarity of words based on HowNet. Journal of Beijing Information Science and Technology University: Natural Science Edition (4), 42–45.