Acta Technica *62*, No. 3B/2017, 227–236

A fast measurement of term relatedness based on full Wikipedia link vector space model[1]

Wenhao Zhu[2,5], Jie Chen[2], Guannan Hu[2], Zhiguo Lu[3], Baogang Wei[4]

Abstract. Words semantic relatedness computation is to calculate the correlation degree of words. It is an important research direction in the implementations of natural language processing applications such as information retrieval, text categorization and so on. Usually, semantic relatedness computation needs a lot of training samples. Wikipedia, the largest online encyclopedia, is becoming a promising training and calculation resource. However, due to the large number of documents and the complex text processing algorithms, current Wikipedia-based approaches are usually slow. This paper proposes a fast method to calculate the relevance of terms based on a Full Wikipedia Link Vector Space Model. In this model, term vectors are constructed with full references of Wikipedia links. To improve the processing efficiency, Jaccard similarity is used to perform the evaluation and a variation function is also introduced to avoid unbalanced results. The experimental results show that the Spearman correlation coefficient can reach 0.65 on the test set WordSimilarity-353, as the same time the computational efficiency is greatly improved.

Key words. Semantic relatedness, Wikipedia, vector space model, Jaccard similarity coefficient.

1. Introduction

Semantic relatedness refers to the degree of correlation between two concepts, which means that two concepts may not be similar, but they can be related to each other through some other relationship. For example, words such as "car" and "gas" are not similar but related. Semantic relatedness of words is a numerical value to

[1] The work of this paper is partially supported by National Natural Science Foundation of China (No.61303097 and No. 61572434).

[2] School of Computer Engineering and Science, Shanghai University, Shanghai, 200444, China

[3] Shanghai University Libraries, Shanghai University, Shanghai, 200444, China

[4] College of Computer Science and Technology, Zhejiang University, Zhejiang, 310027, China

[5] Corresponding author; E-mail: `whzhu@shu.edu.cn`

evaluate the degree of correlation between two words, which is generally between 0 and 1. Semantic relatedness calculation plays an important role in Natural Language Processing (NLP) such as word sense disambiguation [1], information retrieval [2], and text categorization [3], etc. However, semantic relatedness measurement based on large-scale corpus is complicated and often involves massive computation.

Usually, semantic computing requires a lot of background knowledge, thus, statistical analysis of large-scale corpus, such as Latent Semantic Analysis algorithm (LSA) [4], got good results. However, the issued approaches rely heavily on the corpus used in the training, which is computationally intensive, complex and susceptible to sparse and noise data.

Wikipedia contains more than 5 million English terms and millions of terms in other languages. These terms are linked to each other so that they form a huge semantic network. Zesch and Gurevyc analyzed the document maps and classification maps of Wikipedia and compared them with the famous semantic relation network [5]. It was found that Wikipedia could be used as an improvement for existing semantic dictionaries or knowledge bases.

At present, there are many methods for semantic relatedness computing based on Wikipedia such as WikiRelate [6], Explicit Semantic Analysis (ESA) [7] and the Wikipedia Link-based Measure (WLM) [8], etc. David Milne were the first to propose a vector space model based on Wikipedia link structure called the Wikipedia Link Vector Model (WLVM) [9], which calculated semantic relatedness between terms using the links found within their corresponding Wikipedia articles. The main idea was the more of common linked terms, the higher relatedness of these terms would be. Subsequently, David Milne et al. improved WLVM, and proposed WLM. For incoming links, WLM used the modified Google Distance method to calculate the correlation between terms. For links extending out of each article, the above-mentioned WLVM was used to calculate the semantic relatedness.

However, these methods need complicated text processing for all the documents in Wikipedia. In order to improve the efficiency, a fast method is proposed in this paper to calculate the relevance of the terms based on a Full Wikipedia Link Vector Space Model (FWLVSM). In this model, all the entries in Wikipedia that have corresponding articles are used as feature items. Each feature item corresponds to a dimension in the vector space. Jaccard similarity [10] is adopted to perform the comparing calculation. To avoid unbalanced results, a variation function is also introduced to smooth the result values.

2. Term relatedness measurement FWLVSM

2.1. Vector Space Model

Vector Space Model (VSM) [11], an algebraic model proposed by Salton and McGill, was the most commonly used representation model in text mining. The main idea is that each text is represented as a vector in the vector space. Each feature item corresponds to a dimension in the vector space, and the value of each dimension is the weight of the corresponding feature in the text, and this weight can be obtained by algorithm such as TF-IDF. The similarity of two texts can be measured according to the cosine of two vectors.

2.2. Full Wikipedia Link Vector Space Model

Every term in Wikipedia has a corresponding article describing it. If the article contains the other Wikipedia terms, there are hyperlinks chained to the terms, so that a huge link network between the entries is constructed. Having links directing to each other means two terms are relevant and that make it possible to calculate relatedness of the terms. Taking full advantage of this feature, this paper presents a Full Wikipedia Link Vector Space Model (FWLVSM). The Model is based on both the links contained in each article and the links made to them. The main idea of the model is that each Wikipedia term is expressed as a vector of links. Each feature item of the vector corresponds to a term located in the Wikipedia, and if the training term is linked to or out of a feature item, the value of the dimension is marked as 1, otherwise is 0.

The latest enwiki-latest-pages-articles.xml.bz2 file from Wikipedia is used as the training sample set. The xml file contains all the pages in Wikipedia, such as classification pages, explain pages, disambiguation pages and so on. Because some of the pages in Wikipedia are irrelevant, the disambiguate pages are removed(e.g. the Wikipedia pages with keywords such as "list", "wiki", "template", "category"). The corresponding articles for the retained terms are treated as follows. The Wikipedia data is stored in the Neo4j [12] graphics database, with the terms as nodes, hyperlinks between terms as relationships. From this database, we can find the incoming links and out links easily.

All terms are treated as feature items, and are numbered from 1 to n. So that each training term can be presented as vectors $(w_1, w_2, \cdots, w_n)$ with dimension $F_1, F_2, \cdots, F_n$. For example, given a term “Water”, there will be multiple terms linked to “Water” and “Water” will also link to other terms. If these terms appear in the corresponding dimension of $F_1, F_2, \cdots, F_n$, the value such as w_i is marked as 1, otherwise 0. According to formula (1), the term "Water" is expressed as an n-dimensional 0,1 binary vector by using the incoming and outgoing links:

$$\nu_{\text{Water}} = \left(w\left((\text{Water} \to F_1)|(F_1 \to \text{Water})\right),\right.$$
$$\left. w\left((\text{Water} \to F_2)|(F_2 \to \text{Water})\right), \cdots, w\left((\text{Water} \to F_n)|(F_n \to \text{Water})\right)\right). \qquad (1)$$

2.3. Algorithm

Adopt Jaccard similarity coefficient to calculate the semantic relatedness of two terms. The algorithm is as follows:

Step 1: Construct a full vector space model based on the Wikipedia link structure. Select all the terms in Wikipedia and number them, denoted as $F_1, F_2, \cdots, F_n$.

Step 2: Map the word ‘a’ and ‘b’ to term A and term B in Wikipedia.

Step 3: Find out the incoming links and outgoing links of term A and term B;

Step 4: Each term in Wikipedia is represented as an n-dimensional vector where n is the number of terms in Wikipedia. Given in the form of vectors $(w_1, w_2, \cdots, w_n)$, where $w_i = 1$ if the term is link to or link out of F_i, otherwise $w_i = 0$.

Step 5: Calculate the Jaccard similarity coefficients of the corresponding vectors of A and B, and the result is the semantic relatedness of word 'a' and 'b'.

2.4. Jaccard similarity coefficient

In this paper, the vector obtained by FWLVSM is a sparse n-dimensional 0,1 vector. According to these features, the Jaccard similarity coefficient is used to calculate the relevance of terms as follows.

Given two n-dimensional binary vectors A, B, each dimensions of A and B can only be 0 or 1:

Symbol $M_{0,1}$ refers to the sum of dimensions in the same dimension, where value of vector A is 0 and vector B is 1, $M_{1,0}$ refers to the sum of dimensions where value of vector A is 1 and vector B is 0 and $M_{1,1}$ refers to the sum of dimensions where values of vector A and vector B are 1.

The Jaccard similarity coefficient is calculated by using formula

$$J(\mathrm{A},\mathrm{B}) = \frac{\mathrm{M}_{1,1}}{\mathrm{M}_{0,1} + \mathrm{M}_{1,0} + \mathrm{M}_{1,1}}\,. \tag{2}$$

The Jaccard similarity coefficient algorithm does not take the potential value in the vector into account, and simply treats as 0 and 1. After processing, the computational efficiency of Jaccard method is certainly high, because it is only need to do set operations. In this experiment, Jaccard similarity coefficient is usually a small value. In order to describe the relevance of words better, a variation function is used to do the corresponding mapping

$$f(x) = x^{1/3}\,. \tag{3}$$

The value obtained by the formula (3) is the final semantic relatedness of terms.

3. Experiment and analysis

3.1. Evaluation standard

3.1.1. Comparison with human judgments. In this paper, WordSimilarity-353 [13] and Mtruk [6] are used as test sets for evaluation. Spearman correlation coefficient [14] is used to measure the correlation between the algorithm and the test set. The higher the Spearman correlation coefficient is, the closer the calculated results get to the result of the artificial marker, that is, the higher the accuracy of the algorithm. The Spearman correlation coefficient is calculated using formula

$$r = 1 - \frac{6 * \sum d^2}{n * (n^2 - 1)}\,, \tag{4}$$

where d is the difference of two ranks, and n is the sample size.

3.1.2. Computational efficiency: In addition to calculate the accuracy of relatedness measures, the calculation efficiency of the algorithm is performed. When the test parameters are same, the time taken to complete the calculation is considered as the standard of evaluation. The smaller is the time cost, the higher is the efficiency.

3.2. Results and analysis

This paper compares the performance of WLVM [8], WLM [9] and FWLVSM. The link-out vector is created by FWLVSM only by using the link out terms of the pre-trained term, the link-in vector is created only by using the incoming links of terms and the link-mixed vector is created by using both link in and link out terms. The accuracy of the three link vectors is calculated separately.

3.2.1. Accuracy: As shown in Table 1, the Spearman correlation coefficient calculated using link-in vector is higher than link-out on test sets. It indicates that link-in vector contains richer semantic information than link-out. The Spearman correlation coefficient calculated by using link-mixed vector is higher than link-out and link-in on test sets. It is quite clear that FWLVSM using link-mixed vector as input outperforms link-in and link-out, so it is reasonable to treat 0.65 as the accuracy of FWLVSM.

Table 1. Performance of three link vectors under FWLVSM on WordSimilarity-353 and Mtruk

Link vector	WordSimilarity-353	Mtruk
Link-out	0.49	0.46
Link-in	0.58	0.53
Link-mixed	0.65	0.58

Table 2 shows that the Spearman correlation coefficient of FWLVSM is higher than WLVM on WordSimilarity-353 and Mtruk, but slightly lower than WLM. The main reason could be that WLVM calculates semantic relatedness between terms only using the links found within their corresponding Wikipedia articles which contain less semantic information. However, WLM has experimented with two measures which are more detailed.

Table 2. Performance of three algorithms on WordSimilarity-353 and Mtruk

Algorithm	WordSimilarity-353	Mtruk
WLVM	0.53	0.49
WLM	0.69	0.61
FWLVSM	0.65	0.58

3.2.2. Efficiency: WikiRelate and ESA both need to analyze all texts in Wikipedia, such as calculating the TF-IDF value. The computation complexity in text preprocessing has been much larger than methods which relying on the link structure to calculate semantic relatedness of terms. Time expenditure of three algorithms on WordSimilarity-353 is depicted in Fig. 1.

As shown in Fig. 2, based on Wikipedia link structure, FWLVSM algorithm in the test set WordSimilarity-353 takes less time than WLVM and WLM. FWLVSM treats Wikipedia terms into a n-dimensional 0,1 vector. According to this feature, Jaccard similarity coefficient that only need to do "AND "and" OR "operations is an appropriate measurement to calculate the relatedness of terms, which is faster than the cosine value calculation method.

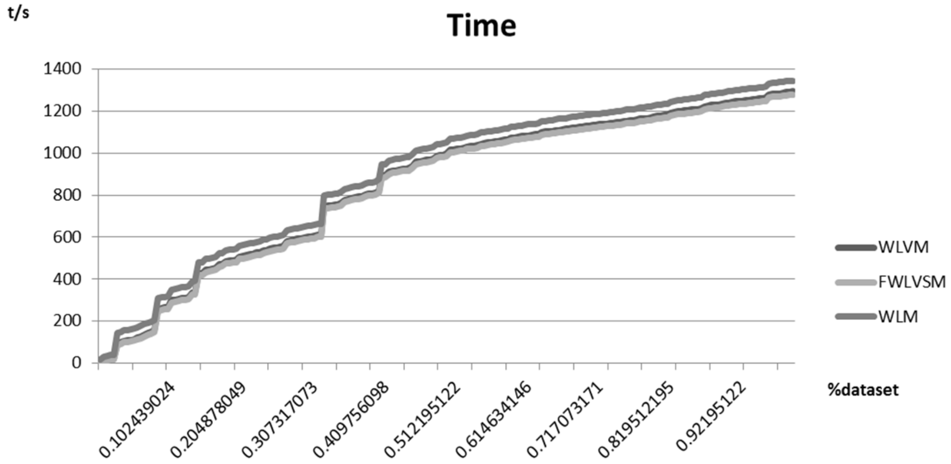

Fig. 1. Time expenditure of three algorithms on WordSimilarity-353

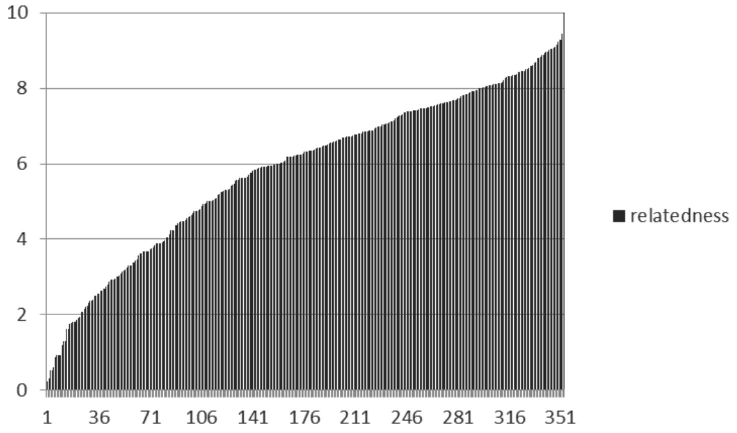

Fig. 2. Distribution of artificial correlation values on WordSimilarity-353

3.2.3. Relatedness quantification: In WordSimilarity-353, the experiment reorders the relevance of the human evaluation in ascending order, resulting in the correlation shown in Fig. 2. Figure 3 shows the distribution of relevance for each pair of words on WordSimilarity-353 using Jaccard similarity coefficient method. And Fig. 4 shows the distribution computed by formula (3). The correlation distribution in Fig. 3 is closer to Fig. 2 than that of Fig. 4.

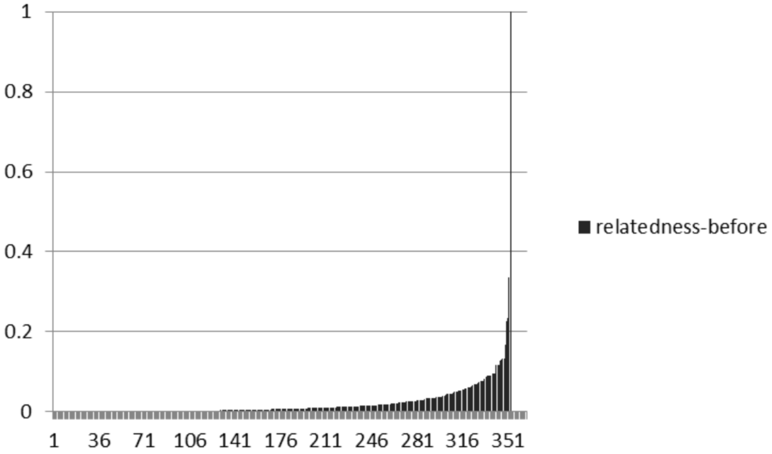

Fig. 3. Distribution of Jaccard similarity coefficient on WordSimilarity-353

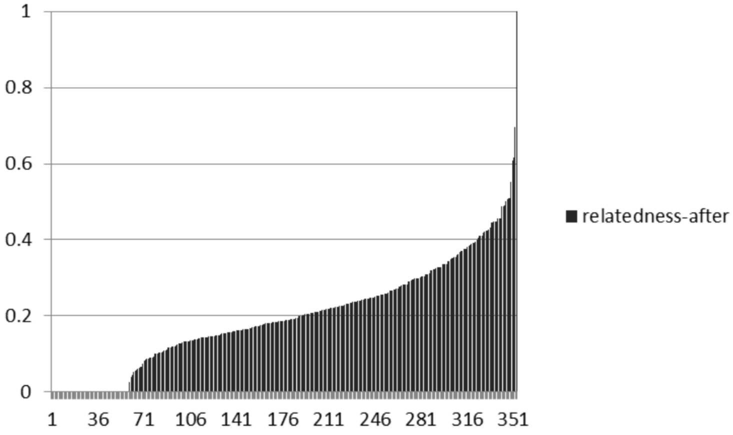

Fig. 4. Distribution computed by formula (3)

4. Conclusion

In this paper, a fast method is proposed to calculate the relatedness of the terms based on FWLVSM. The vectors of FWLVSM are generated with a full correspondence of hyperlinks of Wikipedia pages. That is, a vector includes dimensions describing hyperlinks in the content page of a certain Wikipedia item as well as hyperlinks of other content pages targeting to that item. On that basis, terms in Wikipedia can be represented as a full link vectors and the value of vector dimension is either 0 or 1. The comparing calculation is performed by using Jaccard similarity coefficient and a variation function is introduced to avoid unbalanced results. The experimental results on WordSimilarity-353 and Mtruk show that FWLVSM outperformances WLVM in Spearman correlation coefficient, but slightly lower than WLM. As the needed information of FWLVSM is less than WLM and ESA, the accuracy of FWLVSM is lower than the accuracy of those methods. However, Jaccard similarity coefficient measurement only needs to do the "AND "and" OR "operations, which improves the computational efficiency strongly.

References

[1] J. Sarmah, S. K. Sarma: *Word sense disambiguation for assamese.* Proc. IEEE IC on Advanced Computing (IACC), 27–28 February 2016, Bhimavaram, India, 146–151.

[2] L. Goeuriot, G. J. Jones, L. Kelly, H. Müller, J. Zobel: *Medical information retrieval: Introduction to the special issue.* Information Retrieval Journal *19* (2016), No. 1, 1–5.

[3] B. Tang, H. He, P. M. Baggenstoss, S. Kay: *A Bayesian classification approach using class-specific features for text categorization.* IEEE Transactions on Knowledge and Data Engineering *28* (2016), No. 6, 1602–1606.

[4] M. Boulares, M. Jemni: *Learning sign language machine translation based on elastic net regularization and latent semantic analysis.* Artificial Intelligence Review *46* (2016), No. 2, 145–166.

[5] T. Zesch, I. Gurevych: *Analysis of the Wikipedia category graph for NLP applications.* Proc. TextGraphs-2 Workshop (NAACL-HLT), Rochester, Association for Computational Linguistics (2007), 1–8.

[6] K. Radinsky, E. Agichtein, E. Gabrilovich, S. Markovitch: *A word at a time: Computing word relatedness using temporal semantic analysis.* Proc. International World Wide Web Conference, 28 March–1 April 2011, Hyderabad, India, ACM 337–346.

[7] M. Rybiński, J. F. A. Montes: *DomESA: A novel approach for extending domain-oriented lexical relatedness calculations with domain-specific semantics.* Journal of Intelligent Information Systems (2017), 1–17.

[8] D. Milne: *Computing semantic relatedness using Wikipedia link structure.* Proc. New Zealand Computer Science Research Student Conference, NZ CSRSC'07, 10–13 April 2007, Waikato in Hamilton.

[9] I. H. Witten, D. Milne: *An effective, low-cost measure of semantic relatedness obtained from Wikipedia links.* AAAI Press, Chicago, USA (2008), 25–30.

[10] S. Jimenez, F. A. Gonzalez, A. Gelbukh: *Mathematical properties of soft cardinality: Enhancing Jaccard, dice and cosine similarity measures with element-wise distance.* Information Sciences *367–368*, (2016), 373–389.

[11] B. Hssina, B. Bouikhalene, A. Merbouha: *Evaluation of semantic similarity using vector space model based on textual corpus.* Proc. IC Computer Graphics, Imaging and Visualization (CGiV), 29 March–1 April 2016, Beni Mellal, Morocco, 295–300.

[12] G. Drakopoulos, A. Kanavos: *Tensor-based document retrieval over Neo4j with an application to PubMed mining.* Proc. IC Information, Intelligence, Systems & Applications (IISA), 13–15 July 2016, Chalkidiki, Greece, 1–6.

[13] L. Stanchev: *Creating a phrase similarity graph from Wikipedia.* Proc. IC Semantic Computing, 16–18 June 2014, Newport Beach, CA, USA, 68–75.

[14] M. A. H. Taieb, M. B. Aouicha, A. B. Hamadou: *A new semantic relatedness measurement using WordNet features.* Knowledge and Information Systems *41* (2014), No. 2, 467–497.

Received April 30, 2017