Postprint: Text Similarity Calculation Based on WMF_LDA Topic Model

Authors: Zhang Lu, Lu Tianliang, Du Yanhui

Date: 2018-06-19T00:00:00+00:00

Abstract

The determination and calculation of text similarity represents a significant and valuable research area in the field of natural language processing. Although utilizing LDA models for text similarity calculation takes semantic features into account, it suffers from drawbacks including excessive vocabulary size, failure to incorporate word semantics, and inability to mine and leverage the inherent inter-domain differences among different categories of texts at the textual level. To address these issues, this paper proposes the WMF_LDA (Word Merging and Filtering Latent Dirichlet Allocation) topic model. This model performs unified mapping of domain-specific words and synonyms, filters texts based on part-of-speech, and finally conducts topic modeling. Experimental results demonstrate that this approach significantly reduces vocabulary size during modeling, decreases time consumption in the modeling process, and improves the speed of final text clustering. Furthermore, compared with other text similarity methods, the proposed method achieves a certain degree of improvement in accuracy.

Full Text

Text Similarity Calculation Based on WMF_LDA Topic Model

Zhang Lu[1], Lu Tianliang[1],[2], Du Yanhui[1],[2] [1]Information Technology & Network Security Institute, [2]CIC of Security & Law for Cyberspace People' s Public Security University of China, Beijing 100038, China

Abstract: Text similarity calculation is a significant and valuable research area in natural language processing. While the LDA model considers semantic features for text similarity computation, it suffers from several drawbacks: large vocabulary size, lack of semantic integration, and failure to exploit inherent inter-domain differences among different text categories. To address these issues, this paper proposes the WMF_LDA (Word Merging and Filtering Latent Dirichlet

Allocation) topic model, which maps domain words and synonyms to unified representations and filters text based on part-of-speech before topic modeling. Experiments demonstrate that this approach substantially reduces vocabulary size, decreases modeling time consumption, and improves final text clustering speed. Compared with other text similarity methods, the proposed method also achieves a certain degree of accuracy improvement.

Keywords: word semantics; word merging; POS filtering; text similarity

0 Introduction

Text similarity is an important topic widely studied in linguistics, psychology, and information theory. In natural language processing, text similarity calculation constitutes a crucial research component and direction. In information retrieval and comparison, text similarity algorithms provide essential methods and means, where effective similarity computation can significantly improve, and even largely determine, the accuracy of retrieval and comparison results. Text similarity has extensive applications: in image retrieval, leveraging the similarity of surrounding text can further determine image similarity and achieve better retrieval precision; in text clustering, similarity algorithms provide fundamental criteria that determine clustering outcomes and accuracy. Additionally, text similarity calculation is applied in text summarization generation and document duplication detection.

1 Related Work

Text similarity research has long been a vital topic in natural language processing. The traditional VSM method uses TF-IDF to construct feature vectors and calculates document similarity via cosine distance, but this approach relies solely on term frequency without considering semantic features. Su et al. improved the traditional VSM by incorporating domain weights for feature terms. Huang et al. proposed a term similarity weighted tree to map word similarity to text similarity, though this method suffers from high computational complexity. Gu et al. improved the similarity calculation formula by using TF-IDF values as weights in a modified cosine distance formula. Blanco et al. introduced a novel syntactic and grammatical analysis method that extracts semantic relationships from sentences for similarity computation. Atoum et al. calculated word similarity using distance and content, then extended it to text similarity through weighting. For short text similarity, Huang et al. proposed classifying all terms by part-of-speech and assigning different weights based on importance.

In neural networks and deep learning, Huang et al. proposed a CNN-based text similarity detection model. Kenter et al. integrated word vectors from different dimensions under various conditions and mapped word similarity to text similarity. Kusner et al. employed word mover' s distance (WMD) using word

vectors for text similarity calculation. Neculoiu et al. utilized an LSTM framework to capture semantic similarity between variable-length strings. Kashyap et al. combined latent text semantics with machine learning, integrating data from multiple linguistic resources.

Approaching text similarity from a thematic perspective is another viable method. Sun et al. used LDA for text modeling and represented similarity through topic differences, but this approach suffers from large vocabulary size and slow modeling speed. Zhang et al. improved LDA by incorporating part-of-speech, which reduced vocabulary size and improved modeling speed to some extent, but failed to further leverage semantic relationships between words or mine inherent domain differences among texts. This paper addresses these limitations of LDA-based text similarity calculation by proposing the WMF_LDA topic model that integrates word semantics and part-of-speech information to exploit domain differences among text collections.

2 Methodology

2.1 Model Structure

Different text types possess inherent differences from other categories, primarily manifested in word usage. Each text type has a list of commonly used terms within its domain, which we call a domain vocabulary. Terms in this vocabulary are referred to as domain words for that text type. The proposed WMF_LDA model maximally leverages these domain word differences among different text types based on the original LDA framework. The workflow of the WMF_LDA topic model is shown in [Figure 1: see original paper].

In the WMF_LDA topic model, the original text collection undergoes standard preprocessing (tokenization) before LDA modeling. Then, based on a pre-trained word2vec model, domain words and synonyms are mapped to unified representations at the semantic level. Next, considering that nouns and verbs significantly impact article semantic structure, the mapped word set is filtered by part-of-speech, retaining only nouns and verbs while discarding other parts-of-speech. Finally, LDA topic modeling is performed on the processed result.

In [Figure 1: see original paper], K represents the preset number of document topics, M is the total number of documents in the corpus, N denotes all words in the corpus, W represents observable terms, Z indicates the selected topic for each word, is the document-topic probability distribution, is the topic-word probability distribution, is the hyperparameter for distribution, and is the hyperparameter for .

2.2 Word Similarity Calculation

This paper employs the word2vec model for word vector representation. Its fundamental idea is to compute word vectors based on word positions in text,

incorporating contextual information, thus endowing the resulting vectors with semantic information. It includes two training models: CBOW (continuous bag-of-words) and skip-gram (continuous skip-gram model).

[Figure 2: see original paper] illustrates the word2vec training models. Figure 2(a) shows the CBOW model, which obtains a word' s vector representation through its surrounding context words. Figure 2(b) shows the Skip-gram model, which maps a word to its neighboring context words to obtain the word' s vector representation.

Using the word2vec model, each word is represented as an N-dimensional vector. Word similarity is calculated via cosine similarity:

Sim(,) = ( ) / (√(²)(²))

2.3 Semantic-Based Word Merging

Different news text categories have their own commonly used term sets or professional domain vocabularies. This paper selects 200 random articles from five categories (Space, Art, Agriculture, Economy, Politics) from the Fudan corpus and calculates term frequencies across categories. The results are shown in . Terms like "irrigation"and "rural"appear frequently in the "Agriculture"category but rarely in others; similarly, "piano" and "aerospace" appear predominantly in "Art" and "Space" categories respectively.

Thus, different news text categories each have distinct domain vocabularies where terms appear frequently within their category but infrequently in others. Based on this analysis, we propose:

Assumption 1: Different text categories have relatively fixed domain vocabularies that distinguish them from other categories.

Assumption 2: If two words belong to the same domain vocabulary, their similarity is greater than if they belong to different domain vocabularies.

Furthermore, since LDA modeling first converts text to a term frequency matrix, mapping synonyms and domain words to unified representations can enhance domain distinctiveness and increase domain word frequency. Using Gibbs sampling, we obtain topic distributions for all words. By counting topic occurrences for words in a document, we derive the document' s topic distribution; similarly, counting topic occurrences across the entire corpus yields each topic' s word distribution. This leads to:

Inference 1: The more terms from a particular topic a document contains, the higher the probability that the document covers that topic.

Based on these assumptions and inferences, we propose: calculate similarity to map synonyms and words within the same professional domain to a single representation (e.g., mapping "aviation," "aerospace," and "spaceflight" to "aviation"). This maximizes the representational power of domain vocabulary terms

for their category while increasing synonym frequency. By setting a similarity threshold t to determine word similarity for unified mapping, vocabulary size can be significantly reduced, improving LDA modeling efficiency.

2.4 Part-of-Speech-Based Word Filtering

According to Chinese text characteristics, nouns and verbs play crucial roles in content comprehension and semantic structure. Removing non-noun and non-verb terms does not affect overall semantic understanding. Moreover, nouns and verbs constitute a large proportion of total word count and are essential for text structure.

As core elements of text in both semantic and structural aspects, nouns and verbs are retained after word merging and mapping, while other less impactful words are filtered out. This excludes interference from auxiliary and modal particles in subsequent modeling, further reducing vocabulary size.

2.5 WMF_LDA Topic Modeling and Sampling

The WMF_LDA model uses the original LDA framework, which assumes each document contains multiple topics with different probabilities, and each topic contains multiple words with varying probabilities. In other words, a document is composed of multiple topics following a probability distribution, while each topic consists of terms following a probability distribution, ignoring grammatical structure and word order. For LDA, documents comprise topics, topics comprise words, and both document-topic distribution and topic-word distribution follow multinomial distributions.

The joint probability for generating document m can be expressed as:

P(, , , | ,) = P(, | _{ , }) P(, |) P(|) P(|)

Document generation involves iteratively generating each word. For word n in document m:

a) Sample document-topic distribution from Dirichlet distribution with hyperparameter

b) Sample topic , from multinomial distribution

c) Sample topic-word distribution _{ , } from Dirichlet distribution with hyperparameter

d) Sample word , from multinomial distribution _{ , }

e) Repeat steps a-d times to generate document m

f) Repeat steps a-e M times to generate M documents

The primary parameters to estimate are (document-topic distribution) and _{ , } (topic-word distribution), both multinomial distributions. Hyperparameters and are set to empirical values: = 50/K, = 0.01.

Since , is unknown, we use Gibbs sampling to infer the required parameter distributions from the observed word distributions. The sampling process for WMF_LDA is:

a) Initialize a random topic for each word in the processed word set

b) For each word, update its topic probability using the Gibbs sampling formula, excluding its current topic assignment and re-estimating its probability across topics based on other words' assignments:

P(= k| , w) (, +) (, +)

where counts occurrences of term t in topic k, and counts occurrences of topic k in document m. ¬i denotes exclusion of the word at index i.

c) Repeat until convergence

d) Calculate document-topic distribution:

, = (+) / ((+))

2.6 Text Similarity Calculation

Through WMF_LDA, each text in the corpus obtains a probability distribution over topics. We use topic distribution differences to represent text similarity, selecting relative entropy (KL divergence) as the similarity measure. Since KL divergence is asymmetric, we use its symmetric variant, JS distance:

Sim(,) = ½ [(, (+)/2) + (, (+)/2)]

where and represent the topic probability distributions obtained through WMF_LDA modeling.

(,) = (log(/))

3 Experiments

3.1 Experimental Data

Experimental data comprises two parts: word2vec training data and WMF_LDA modeling with similarity calculation data.

For word2vec training, we combined multiple Chinese text corpora: Fudan Corpus, Tencent News, Sogou Lab News, Phoenix News, NetEase News, People' s Daily, and Wikipedia, totaling 2,813,611 news texts with 830,000 vocabulary entries.

For LDA modeling and similarity calculation, we used selected texts from the Fudan Corpus, specifically 200 random texts from each of five categories (Art, Space, Agriculture, Economy, Politics), totaling 1,000 texts.

3.2 Text Clustering and Similarity Measurement

Using our proposed similarity calculation method, we compute pairwise similarity between texts based on topic distribution and cluster the test set using these distances. Clustering accuracy is evaluated by checking whether each document is assigned to its correct category and whether each cluster contains the appropriate documents. The F1 score measures clustering quality.

Precision and recall for cluster j belonging to category i are:

P(i, j) = /
R(i, j) = /

where is the number of category i texts in cluster j, is the total number of category i texts, and is the total number of texts in cluster j.

The F-score is:

$$F(i, j) = 2 \times P(i, j) \times R(i, j) / (P(i, j) + R(i, j))$$

Global clustering F1 is:

F1 = (/) × max (F(i, j))

where N is the number of categories and n is the total number of test texts. Higher global F1 indicates better clustering and similarity algorithm performance.

3.3 Semantic-Based Word Merging

The 1,000 experimental texts originally contained over 60,000 distinct words. Using word2vec for semantic merging with threshold $t = 0.5$ reduced the vocabulary to 40,000 words—only two-thirds of the original size—effectively improving subsequent LDA modeling speed. shows sample word mappings.

As shown in , terms like "international aviation," "aviation," and "airline" clearly belong to the "Space" category and rarely appear in others, thus they are unified as "international aviation." Similarly, "Central Organization Department," "Central Committee of the Communist Youth League," and "Central Propaganda Department" frequently appear in "Politics" texts and are mapped to "Central Organization Department." This domain-specific merging increases term frequency and enhances domain representational capacity.

3.4 LDA Topic Number Selection

Before LDA modeling, the topic number K must be determined. Small K values prevent topic differentiation, causing multiple topics to collapse into one

and reducing similarity calculation accuracy. Large K values map each text to excessive topic dimensions, ignoring intra-category differences and increasing computation time. Therefore, K selection directly impacts LDA accuracy.

We determine K using the following algorithm: a) Process the 1,000 test texts using the word merging and filtering described in Sections 2.3-2.4
b) Set modeling parameters with empirical and values, and test a range of K values
c) For each K, perform topic modeling to obtain K-dimensional topic distributions for all 1,000 texts
d) Calculate pairwise similarity using Section 2.6
e) Cluster using K-means
f) Compute global F1 for each K

[Figure 3: see original paper] shows that with K = 400, F1 reaches its maximum value of 0.70 (averaged over five runs). Therefore, subsequent experiments use K = 400.

3.5 Word Scale and Runtime Comparison

The proposed WMF_LDA model first maps domain words and synonyms to unified representations, then filters nouns and verbs based on Chinese text characteristics. This compresses corpus size from both semantic and organizational perspectives, reducing vocabulary and improving modeling time.

[Figure 4: see original paper] compares WMF_LDA with traditional LDA on the same dataset. Traditional LDA models over 60,000 words, taking more than 7,000 seconds, while WMF_LDA models 40,000 words in about 4,000 seconds —reducing both vocabulary size and runtime to two-thirds of the original.

3.6 Clustering Accuracy Comparison

Using K-means clustering with F1 as the evaluation metric, we compare our method against traditional TF-IDF, classic LDA, and TF-IDF combined with our word merging and filtering (WMF). Results in show:

Method	Accuracy F1
TF-IDF	60.1%
TF-IDF+WMF	61.8%
LDA	68.1%
WMF_LDA (Our method)	72.5%

Our WMF_LDA significantly outperforms traditional LDA and TF-IDF. Applying WMF to TF-IDF also yields improvement, as filtering less impactful words while unifying domain-specific terms enhances domain differences in both semantic and structural aspects.

4 Conclusion

This paper proposes the WMF_LDA topic model based on analysis of traditional TF-IDF and LDA methods for text similarity. Recognizing that different text types have distinct domain vocabularies, the model maps semantically similar or domain-related terms to unified representations, increasing domain word frequency and enhancing domain representational capacity. During topic modeling, this increased frequency improves document-topic probability distribution. Experiments demonstrate that WMF_LDA reduces vocabulary size, decreases modeling time, and improves clustering accuracy.

Future work will consider pronouns, adjectives, and adverbs to further explore structural and semantic relationships among text components, potentially extending similarity calculation from sentence-level to document-level.

References

[1] Huang Chenghui, Yin Jian, Hou Fang. A text similarity measurement combining word semantic information with TF-IDF method [J]. Chinese Journal of Computers, 2011, 34(5): 856-864.

[2] Xu Guanghao, Wang Ning, Liu Jiaming, et al. Comprehensive computation algorithm of similarity for natural language retrieval [J]. Computer Systems & Applications, 2017, 26(6): 170-175.

[3] Erkan G, Radev D R. LexRank: graph-based lexical centrality as salience in text summarization [J]. Journal of Artificial Intelligence Research, 2004, 22(1): 457-479.

[4] Theobald M, Siddharth J, Paepcke A. SpotSigs: robust and efficient near duplicate detection in large web collections [C]// Proc of International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM Press, 2008: 563-570.

[5] Guo Qinglin, Li Yanmei, Tang Qi. The similarity computing of documents based on VSM [J]. Application Research of Computers, 2008, 25(11): 3256-3258.

[6] Su Xiaohu. Research of sentence similarity based on improved VSM [J]. Computer Technology & Development, 2009, 19(8): 113-116.

[7] Gu Chongyang, Xu Haoyu, Zhou Han, et al. Text similarity computing based on lexical semantic information [J]. Application Research of Computers, 2018(2): 391-395.

[8] Blanco E, Dan M. A Semantic logic-based approach to determine textual similarity [J]. IEEE/ACM Trans on Audio Speech & Language Processing, 2015, 23(4): 683-693.

[9] Atoum I, Otoom A. Efficient hybrid semantic text similarity using wordnet and a corpus [J]. International Journal of Advanced Computer Science & Applications, 2016, 7(9).

[10] Huang Xianying, Li Qindong, Liu Yingtao. A grammatical category-combined short-text similarity algorithm and its application in text categorization [J]. Telecommunication Engineering, 2017, 57(1): 78-82.

[11] Huang Xianying, Zhang Jinpeng, Liu Yingtao, et al. Short text similarity algorithm based on term mapping with semantic [J]. Computer Engineering & Design, 2015(6): 1514-1518.

[12] Huang Jiangping, Ji Donghong. Sentence semantic similarity model based on convolutional networks [J]. Journal of South China University of Technology: Natural Science Edition, 2017, 45(3): 68-75.

[13] Kenter T, Rijke M D. Short text similarity with word embeddings [C]// Proc of ACM International Conference on Information and Knowledge Management. New York: ACM Press, 2015: 1411-1420.

[14] Kusner M J, Sun Y, Kolkin N I, et al. From word embeddings to document distances [C]// Proc of International Conference on Machine Learning. 2015: 957-966.

[15] Neculoiu P, Versteegh M, Rotaru M. Learning text similarity with siamese recurrent networks [C]// Proc of Repl4nlp Workshop at ACL. 2016.

[16] 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.

[17] Sun Changnian, Zheng Cheng, Xia Qingsong. Chinese text similarity computing based on LDA [J]. Computer Technology & Development, 2013(1): 217-220.

[18] Zhang Chao, Chen Li, Li Qiong, et al. Chinese text similarity algorithm based on PST_LDA [J]. Application Research of Computers, 2016, 33(2): 375-377.

Note: Figure translations are in progress. See original paper for figures.