TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 06/15 pp619-632
Volume 22, Number 6, December 2017

Measuring Similarity of Academic Articles with Semantic Profile and Joint Word Embedding

Ming Liu*, Bo Lang, Zepeng Gu, and Ahmed Zeeshan

Abstract: Long-document semantic measurement has great significance in many applications such as semantic searchs, plagiarism detection, and automatic technical surveys. However, research efforts have mainly focused on the semantic similarity of short texts. Document-level semantic measurement remains an open issue due to problems such as the omission of background knowledge and topic transition. In this paper, we propose a novel semantic matching method for long documents in the academic domain. To accurately represent the general meaning of an academic article, we construct a semantic profile in which key semantic elements such as the research purpose, methodology, and domain are included and enriched. As such, we can obtain the overall semantic similarity of two papers by computing the distance between their profiles. The distances between the concepts of two different semantic profiles are measured by word vectors. To improve the semantic representation quality of word vectors, we propose a joint word-embedding model for incorporating a domain-specific semantic relation constraint into the traditional context constraint. Our experimental results demonstrate that, in the measurement of document semantic similarity, our approach achieves substantial improvement over state-of-the-art methods, and our joint word-embedding model produces significantly better word representations than traditional word-embedding models.

Key words: document semantic similarity; text understanding; semantic enrichment; word embedding; scientific literature analysis

1 Introduction

Text semantics is drawing increasing attention from different disciplinary fields. The semantic measurement of long documents has the potential for use in many applications such as semantic searches, plagiarism detection, and citation recommendations in the academic domain. However, document-level semantic measurement is far from being thoroughly studied and, to our knowledge, there are as yet no publically available datasets for evaluation purposes. Human beings can understand documents naturally and grasp the document meanings far beyond the literal information provided. However, it's difficult for computers to capture the meanings of long documents. The main reasons for this are as follows: (1) Unlike a sentence or concept with one pure focus, a long document typically has abundant content and its focus often moves from one topic to another. Therefore, it is difficult to derive and represent the general meaning of the whole document. (2) The structure of texts is also an important factor in text semantic matching; the syntactic analysis of sentences is straightforward whereas the discourse analysis of long documents is difficult. (3) Background knowledges such as basic lexical knowledge and common sense knowledge are often omitted during the process of creating documents.

• Ming Liu, Bo Lang, Zepeng Gu, and Ahmed Zeeshan are with State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China. E-mail: liuming@nlsde.buaa.edu.cn; langbo@nlsde.buaa.edu.cn; guzepeng@nlsde.buaa.edu.cn; zeeshan mscs@yahoo.com.
∗ To whom correspondence should be addressed.
Manuscript received: 2016-12-31; revised: 2017-04-22; accepted: 2017-06-14

As such, it is impractical to attempt to fully comprehend the meanings of documents based only on the literal information given. In Ref. [1], the authors stated that "If the mind goes beyond the data given, another source of information must make up the difference." Therefore, another important issue in document-level semantic measurement is how to incorporate external knowledge resources into this measurement.

The semantic measurement of small text units, e.g., words and sentences, has been fruitful[2–13, 15]. One thought for determining the semantic similarity of short texts is to go from word-level to sentence-level semantics by using the linear combination of word-to-word semantic similarity[5, 6] or incorporating both lexical-level and sentence-level features[11–13]. Another way to achieve text semantic matching is to construct graph representations of texts[14, 16–18], in which the entities occurring in each text are represented as nodes in a graph, with edges being the relationships between the entities derived an external knowledge base. Then, similarity can be calculated by the distance between graphs. However, the above sentence-oriented methods are not suitable for long documents.

Traditional document-level similarity methods are based on the statistical analysis of word morphology for text indexing, such as term frequency–inverse document frequency (tf–idf)[19]. The two phrases, "preparing a manuscript" and "writing an article", are semantically similar, but by the tf-idf metric, they are quite unrelated. In fact, an important factor in understanding documents is to understand the meaning of words such as "manuscript" and "article". To fill in this "semantic gap", many universal and domain knowledge resources have been constructed, such as WordNet[20] and Freebase[21]. Corpora are another kind of external knowledge resource and many statistical methods learn the semantic relatedness of words from an external corpus[22, 23]. With respect to long documents, Latent Dirichlet Allocation (LDA)[24] assumes that each document comprises a distribution of different topics and that the semantic similarity of documents can be measured by the similarity of their topic distributions. The limitation of LDA is that its bag-of-words assumption ignores the word order and discourse structure, which are crucial for understanding meaning in long documents.

Neural networks are another kind of method for measuring text semantics[25, 26], with the key idea being to learn each word representation by its context neighborhoods. Nevertheless, most long documents are self-contained and independent, so the relatedness of documents is weak even in a domain corpus. In addition, documents are often unordered and cannot serve as appropriate document contexts for each other. Hence, it's difficult to directly learn the overall semantic representation for a document using a neural-network-based method.

Humans comprehend the core semantics of a document by connecting multiple information clues specific to the background, document structure, and main points, rather than to single sentences. The core semantics of an academic article present the research work of authors, and the semantic similarity between academic papers is the similarity between different research works. Therefore, to obtain an overall representation of each academic document, we propose the use of a semantic profile to cover several key topics in academic articles. The semantic profile consists of the research target, methodology, research style, publication date, and keywords, which can be regarded as a multi-faceted summary of each document. The quality of external knowledge is also crucial to the semantic measurement of texts. As sources of external knowledge, we include the structured knowledge bases in ontology and external corpora, which are complementary. We take advantage of the accurate semantic relatedness of structured knowledge bases to enrich the items included in the semantic profiles, and also propose a joint word-embedding model in which we incorporate the domain-specific semantic relations into the traditional word-embedding process. Figure 1 shows the framework of our methods.

To summarize, the main contributions of our work are as follows:

(1) We introduce the semantic profile as an overall semantic representation of academic articles and develop a semantic enrichment method, which is based on external knowledge resources.

(2) We propose a framework for calculating document semantic similarity based on the semantic profile, which utilizes the combined semantic relatedness of items in the profile. This framework uses different kinds of knowledge resources to convey the intrinsic semantic relatedness of concepts.

(3) To yield enhanced word representations, we provide a joint word-embedding model that incorporates the traditional context constraint along with a domain-specific semantic relation constraint. Our

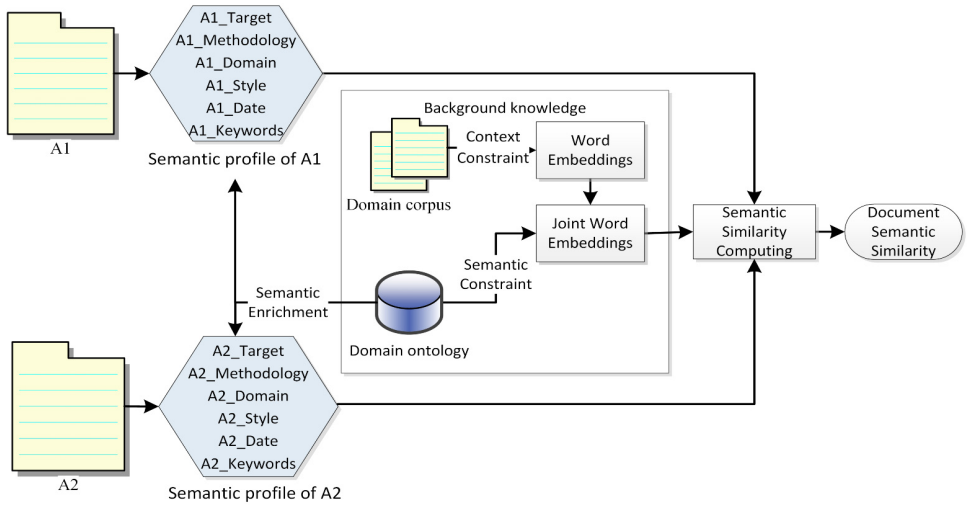


Fig. 1 Document similarity measure using semantic profile and joint word embedding.

experimental results demonstrate the efficacy of different kinds of semantic relations.

The remainder of this paper is organized as follows. In Section 2, we describe work related to our research area. In Section 3, we describe the semantic profile for academic articles and the general framework for calculating document semantic similarity based on this semantic profile. In Section 4, we propose our joint word-embedding model that combines the domain-specific semantic relation constraint and the traditional context constraint. In Section 5, we evaluate our experimental results. The results are discussed in Section 6. Lastly, in Section 7, we draw our conclusions.

2 Related Work

Document-level semantic similarity research is still an emerging research and few studies to date have focused on long documents. Here, we divide the most closely related research methods into three categories—statistical, assembly, and the recent neural network based methods.

Assembly methods. Many knowledge resources[20, 21, 26–28] have been constructed to convey the common sense knowledge of words. The intuitive approach is to utilize the linear combination of word-to-word semantic similarity to determine the overall semantic similarity between two text snippets[5, 6]. The Explicit Semantic Analysis (ESA) method regards each Wikipedia page as a concept, and the meaning of words or text is represented as a weighted combination of those Wikipedia concepts[29]. A more sophisticated approach to gauging the semantic similarity of texts is to combine lexical features and sentence structure features. In Ref. [12], the authors used a support vector regression model to predict the level of semantic similarity between sentences, based on five grades[11], and also used features such as n-gram overlap, weighted words overlap, and syntactic features. This method is also supervised and thus, to work well, requires more labeled data.

Many researchers have represented each piece of text as a graph and measure document similarity by the graphic similarity. A graph-based semantic model for representing document content was proposed in Ref. [16], and this model acquires fine-grained information about entities and their semantic relations from the DBpedia. Then, it calculates semantic similarity by the optimal matching of entities in two documents. In Ref. [17], the authors regarded words in text as graph nodes via WordNet and generated a stability distribution after a random walk. Then, the semantic similarity of texts is represented by the distance between the two distributions. In Refs. [14, 18], the authors proposed to understand the meaning of unstructured text by linking their entities to external knowledge resources and representing documents as triple graphs after information extraction. However, above methods mainly focus on the semantics of sentences or short text and cannot scale up for long documents due to their inherent complexity and wide interpretability. Also, there are rare entities such as persons, organizations, and place names in the content of academic articles that render these methods unsuitable.

Statistical methods. Traditional similarity research between documents has focused on the purpose of

information retrieval, such as the tf-idf method[19]. This method performs document indexing at the surface level and introduces no semantic relatedness. LDA is a generative probabilistic model[24] that, given a collection of documents and a fixed number of topics, will derive the semantic relatedness of words via topics. Each document is regarded as a distribution over a set of topics. As such, LDA can be used in the semantic analysis of long documents based on their topic distributions. In Ref. [30], the authors defined a similarity measure based on topic maps in a document clustering task, in which documents are transformed into topic maps, and the similarity between a pair of documents is represented as a correlation between their common patterns. Along the same lines as the topic model, the semantic similarity between documents is measured based on the divergence of topic distributions in Refs. [31, 32]. The main limitations of the LDA method are its computational complexity and representational opaqueness. When the corpus is large, the consumption of time and memory is huge.

Neural network based methods. Since the introduction of the neural network language model[25], neural networks have greatly advanced the measurement of text semantics. In Ref. [4], the authors proposed the word embedding approach, word2vec, which is based on word co-occurrence in corpora and wherein words are represented as real-valued vectors in the semantic space. The GloVe[28] neural network method avoids large computational costs by not building a full co-occurrence matrix, but training directly on the non-zero elements in a matrix. Besides word embedding, phrase representations can also be directly learned by treating phrases as single tokens. The paragraph vector[26], which was also proposed to measure the semantic similarity of short texts, learns a fixed-length feature representation from variable-length pieces of text by considering the ordering of words and their semantics. One limitation of the neural-network-based method is the context requirement, i.e., semantically related neighbors, due to the difficulty of deriving related documents from document collection.

3 Semantic Profile-Based Academic Article Similarity

3.1 Semantic profile for academic articles

Most academic articles have normative formats and regular structures and their research work comprises similar profiles that can be described in uniform semantic annotations. We define the semantic profile to cover key topics in the research work, as shown in Table 1. The most notable points of an academic paper are its purpose, method, and results, which can be regarded as the semantics in which readers are interested. Keywords can fuzzily represent the core semantics and provide readers with a perception of the general research. The domain of the research issue indicates the research branch and the style of the research work reflects the research approach and difficulty, all of which can be used to filter out research that is irrelevant to readers. The publication dates indicate the different stages of research issues. If two academic articles were published on close dates, they are more likely to focus on related topics.

The style of research work implies certain semantics. Different article styles can reflect differences in the difficulties, approaches, and types of research. There is a distinct difference between a *survey* paper and a *theoretical origination* paper. Generally, *theoretical origination* articles are more innovate and complex than *survey* papers, and they have different application scenarios. *Survey* papers are suitable for beginners who are gathering basic knowledge. *Theoretical origination* articles are more suitable for aspiring experienced professionals. Hence, research style is a distinguishing factor of academic documents and semantic profiles. The practical challenge is how to measure the relatedness of different research styles. To express the knowledge implied by styles of research work, we first propose a style ontology for the academic articles (as shown in Fig. 2), which we demonstrate by analyzing the types of papers in the computer science domain. Specifically, *theoretical origination* means research that is proposing original approaches. *Methodology improvement* represents research articles that are improving upon some

Table 1 Semantic profile of academic articles.

Semantic item	Remark
Target	Target of the research
Methodology	Technologies or algorithms used in the research
Domain	Domain of the research
Style	Essential type of research manner
Date	Data of publication
Keywords	Keywords describing the article

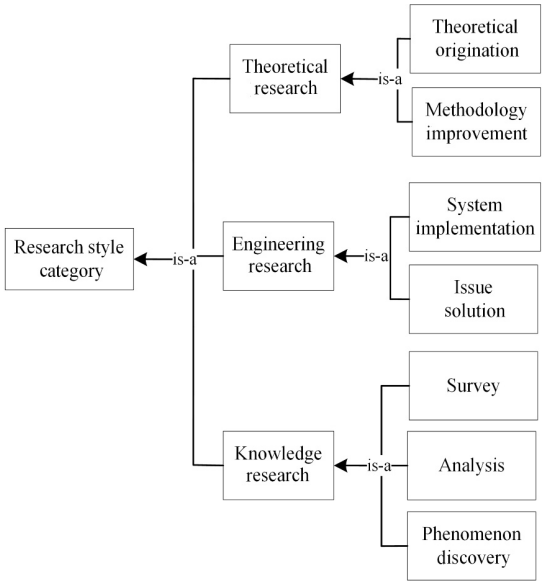


Fig. 2 Hierarchy of the research style ontology.

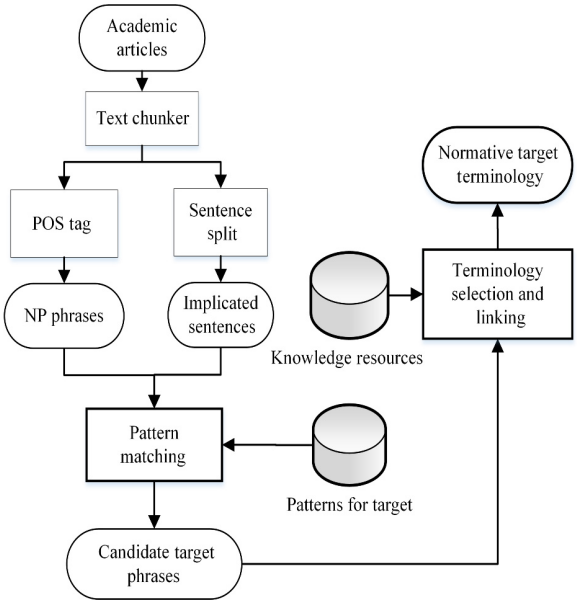


Fig. 3 Semantic enrichment framework for *target* in the semantic profile.

methodology or theory. *System implementation* is a category for research articles that provide some system or tool. *Issue solution* refers to research that solves some problems in existing methods. *Survey* refers to research surveying certain research issues. *Analysis* indicates articles that analyze certain research topics, and the *phenomenon discovery* refers to research uncovering new informations. Research style ontology is a knowledge carrier that distinguishes the semantic differences between different styles of research work.

3.2 Semantic profile enrichment

To a great extent, external knowledge resources can convey the internal semantic relatedness of concepts. Enriching the semantic profile from external knowledge resources can substantially enhance our understanding of documents. However, the real problem is how to find and link normative terminologies from external knowledge resources to the corresponding semantic items shown in Table 1. There are few labeled datasets for training a recognition model to recognize which phrases are candidate semantic items for classifications such as target and methodology in a specific domain. We use pattern-based methods to discover candidate semantic items and then select their normative expressions from external knowledge resources.

General architecture. As shown in Fig. 3, we divide academic articles into different sections, with the *Title*, *Abstract*, *Introduction*, and *Conclusion* sections providing an overview of the research work. Then, we perform sentence splitting and Part-Of-Speech (POS) tagging on each implicated sentence in the selected sections. Implicated sentences are those that may contain semantic items, and we defined 95 trigger words to identify these target-containing sentences. Thirdly, we choose all the Noun Phrases (NP) in each implicated sentence as phrase candidates. Then, we develop many patterns to identify which phrase candidate is the most semantically appropriate for the specific items in our semantic profile. Lastly, using external knowledge resources, we transform the semantic phrases into normative terminologies.

Semantic phrase selection. In this step, we identify the patterns for the extraction of the *target* and *methodology* items, for which Table 2 shows several that are typical. For example, we identify the implicated sentence "*In this paper, we propose a supervised machine learning approach for relation extraction*" by the trigger word "*propose*" in the introduction section of the academic article. This matches a preceding target pattern, i.e., "*approach for*". Thus, we choose the phrase "*relation extraction*" as the target candidate.

Normative terminologies selection. In the semantic phrase selection step above, we recognized the candidate semantic phrases such as the candidate target phrases in Fig. 3. The important issue in this step is to select their formal terminologies as found in external knowledge resources. Many candidate phrases for certain semantic items are inconsistent due to different

Table 2 Typical extraction patterns for identifying *target* and *methodology* candidates.

Semantic item	Pattern preceding semantic items	Pattern following semantic items
Target	Problem of	Overview
	Task of	Evaluation
	System to	Track
	Survey on	System
	Approach for	Called
	Framework for	Process
Methodology	By use of	Based framework
	That using	Method to
	Which employ	Algorithm for
	Takes advantage of	Techniques to
	Methods of	Is applied
	Through an	Performs much better

academic preferences and habits, such as "*semantic relation extraction*", "*semantic relation detection*", and "*relation classification*". To enrich the semantic profile and make it comparable, it is necessary to use normative terminologies for "*relation extraction*". To select the normative terminology for these candidate phrases, in our research task we take advantage of a domain synonym list, which annotates synonyms for the formal terminologies in the knowledge resource, i.e., the domain ontology. We expand these terminologies in the domain ontology via synonyms to cover the range of possible candidate semantic phrases. Then, to identify the proper normative terminology, we calculate the edit distances between each candidate target phrase and all the expanded terminologies of the domain ontology, and choose as the normative terminology having the minimum edit distance from the candidate target phrase. Thus, via the synonym list, we map different candidate phrases with same meaning to the identified normative terminology in the domain ontology. Lastly, we use the normative terminology from the domain ontology to enrich the target item of the semantic profile.

Semantic expansion. Many semantic items are hard to extract but are closely related with other items in the external knowledge resources, such as the research purpose and domain, the adopted methodology and toolkit, and the research objective and dataset. If we have a known semantic item, we can easily deduce a related item from the external knowledge resources. Generally, the research purpose is a core issue in academic articles, and the domain to which an academic article belongs can be deduced from its research purpose. To enrich the target of an academic article, we use the domain ontology to deduce the domain to which it belongs. We have predefined the research domain in the domain ontology that include "*information extraction*", "*information retrieval*", "*machine translation*", "*question answering*", "*text summarization*", "*handwriting recognition*", "*text classification*", "*grammar checker*", "*speech recognition*", and "*ontology learning*" in the domain ontology. After obtaining the research target, we calculate the semantic similarities between the target and each predefined domain concept based on the domain ontology. We then choose the concept having the maximal semantic similarity to the target concept in the domain ontology as the domain of the corresponding research article. This procedure is illustrated in Fig. 4.

3.3 Semantic profile-based similarity calculation

We can measure the semantic similarity between documents based on the semantic items in the semantic profile, and we can obtain the semantic similarity of these items using external knowledge resources. In our method, we use a linear combination of these semantic similarities to establish the global similarity between documents. These enriched semantic items include *target*, *domain*, *style*, *methodology*, *keywords*, and *date*. We define the similarity between two long documents D_1 and D_2 as shown in Eq. (1):

$$\mathrm{sim}(D_1, D_2) = \sum_{i=1}^{6} w_i S_i(L_{1i}, L_{2i}) \tag{1}$$

where w_i is the weight of the i-th element in the semantic profile, S_i is the similarity between i-th elements L_{1i} and L_{2i}, and L represents the items in the semantic profile, i.e., L=*Target*, *Domain*, *Style*, *Methodology*, *Keywords*, *Date*. Since there are three kinds of semantic items in each semantic profile, their semantic similarity must be measured in different

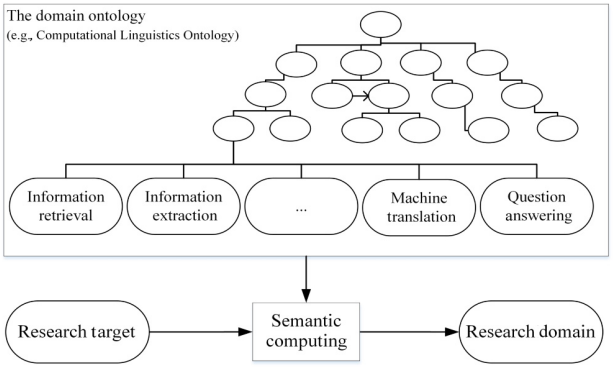


Fig. 4 Semantic expansion procedure.

ways. Semantic similarity with respect to *style* can be measured via our research style ontology, and for *Date* similarity can be measured by the interval. Terminologies, such as *target*, *methodology*, and *keywords*, can be measured by external knowledge such as distributed word representations

Style similarity between different types of research style is measured by the Wu and Palmer's method[33], which is based on the research style ontology shown in Fig. 2 using the formula shown in Eq. (2):

$$\text{sim}_{\text{style}} = \frac{2\text{depth}_{\text{LCS}}}{\text{depth}_{\text{style}_1} + \text{depth}_{\text{style}_2}} \tag{2}$$

where style_1 and style_2 indicate two research style types. LCS is the least common subsumer of the two research style nodes in Fig. 2.

Date similarity is defined by the time interval. We use *years* and *months* to compute the similarity of two dates. We define the *date* similarity formula as shown in Eq. (3).

$$\text{sim}_{\text{date}} = \frac{1}{1 + |(\text{year}_1 + \frac{\text{month}_1}{12}) - (\text{year}_2 + \frac{\text{month}_2}{12})|} \tag{3}$$

Concept semantic similarity can be measured using external knowledge resources. To determine their internal semantic relatedness, we consider the contents of *target*, *domain*, *methodology*, and *keywords* as collections of terminologies, and measure their similarities using word embedding. We can measure the terminology similarity by the cosine distance, which is determined using Eq. (4):

$$\text{sim}_{\text{concept}} = \frac{\text{termvec}_1 \cdot \text{termvec}_2}{|\text{termvec}_1||\text{termvec}_2|} \tag{4}$$

where termvec_1 and termvec_2 are vector representations of the terminologies by word embedding. The qualities of word representation vary widely when using different kinds of corpora. As such, we use several categories of corpora, such as a domain corpus and universal corpus, to learn word vectors and also to evaluate their performances. In addition, to refine the quality of the word representations, in the next section, we propose a joint word-embedding model.

4 Joint Word Embedding

Word embedding is a procedure by which distributed word representations are learned from document corpora. Although word relatedness in a corpora is noisy and implicit, which may lead to biased word representations, it has a wide word coverage. Structured knowledge bases such as domain ontologies represent lexical knowledge via explicit, accurate, and compact semantic relationships. As such, will it produce better word representations by incorporating superior semantic knowledge such as domain ontology into the word embedding procedure? In fact, there is already evidence of the helpfulness of semantic relations in the word vector training process[34, 35]. In contrast to the above methods, which use the overall semantic relations of concepts as constraint with which to revise the distributed representations of words, we individually incorporate domain-specific semantic relations into the traditional context constraint. Additionally, we argue that the knowledge in structured knowledge bases is better quality than the implicated semantics in text, so we prioritize the semantic knowledge restriction in our joint word embedding model.

4.1 Word embedding

In the word2vec approach[4], words are represented by a set of latent variables in the semantic space. Words that are semantically similar should be close to each other in the semantic space. In our experiments, we employ the Continuous Bag-Of-Words (CBOW) model, in which each word is represented by a vector that is averaged with other word vectors in the context, and the resulting vector is used to predict other words in the context. Given a sequence of training words w_1, w_2, ..., w_T, the objective of the CBOW model is to maximize the average log probability in Eq. (5) to learn the representations for each word w_t. T is the token number of the corpus.

$$\max \frac{1}{T} \sum_{t=1}^{T} \log p(w_t | w_{t-c}^{t+c}) \tag{5}$$

where c is the window size and w_{t-c}^{t+c} is the collection of words covered by the window of size c centered at w_t. CBOW defines $p(w_t | w_{t-c}^{t+c})$ with the softmax function as follows:

$$p(w_t | w_{t-c}^{t+c}) = \frac{\exp({e'_{w_t}}^{\top} \sum_{-c \leqslant j \leqslant c, j \neq 0} e_{w_{t+j}})}{\sum_{\theta=1}^{W} \exp({e'_{w_\theta}}^{\top} \sum_{-c \leqslant j \leqslant c, j \neq 0} e_{w_{t+j}})} \tag{6}$$

Specifically, e_w is the "input" vector representation of w, and e'_w is the "output" vector representation of w, where W is the number of words in the vocabulary. After the training converges, words with similar meaning are mapped to a similar position in the vector space. Each word vector is initialized randomly and word vectors are averaged to predict the next word in the context. Neural network based word vectors are generally trained using stochastic gradient descent in which the gradient is obtained via back propagation.

Domain word embedding. Since the focus of our research is on the academic articles, our word representations must be derived from the academic domain corpus. We trained word representations in the CBOW model using the word2vec word-embedding tool and used the ACL Anthology Network (AAN) corpus[36] for the domain word embedding. Specifically, we used the AAN 2013 release of the AAN corpus, which has 20 416 academic articles and summaries in the field of computational linguistics. We set the training context window size to 8 with trained word vectors having 200 dimensions. Finally, we obtained about 230 000 word vectors from the whole AAN corpus in the computational linguistic domain.

Universal word embedding. A full-scale universal corpus also contains words in specific domains. Compared with domain word embedding, which is more suitable for tasks in a specific domain? To answer this question, for comparison, we also utilize the Wikipedia corpora in the universal word-embedding model. We selected the full Wikipedia dump as our corpus for training universal word vectors. There are several trained word-embedding models based on Wikipedia dumps, and we selected that from the "https://github.com/idio/wiki2vec". This word-embedding model is also trained in the CBOW model and has a sliding window of 10 and word vector dimension of 1000.

4.2 Domain ontology and semantic relation constraint

Since common knowledge resources such as WordNet cannot cover domain terminologies, we manually constructed a domain ontology to convey the semantics of different terminologies. We used concepts extracted from the AAN corpus to manually construct a Computational Linguistics (CL) ontology. Currently, our ontology includes 1216 concept nodes with a hierarchy of nine depths, which can be continuously expanded in future work. Figure 5 shows the architecture of our CL ontology. According to the characteristic of the computational linguistic domain, we designed the ontology to have three parts—*Research Topic*, *Infrastructure*, and *General Approaches*—with each part being enriched by more detailed descendant nodes. The main relationships between concepts in the ontology are hyponymic. During construction, we considered and annotated synonyms in the ontology concept nodes.

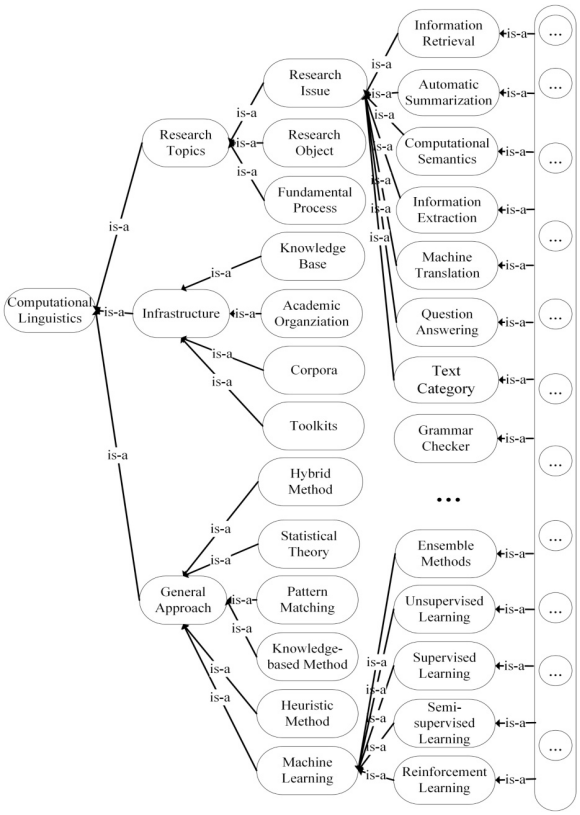


Fig. 5 Main architecture of computational linguistics ontology.

We assume that concepts in the ontology are semantically related if there is a direct relationship between two concepts, so the task is to predict a concept given other concepts in the knowledge bases that have a specific semantic relation with it. In our relation constraint model, we define R as a specific kind of relationship such as *hyponymy* or *synonymy*. Given a word w_t and the words $w_1, w_2, \ldots, w_n$ which have a specific relationship R with w_i in the knowledge base, our objective is to maximize the average log probability of all words by summing over all words.

$$\max \frac{1}{N}\sum_{t=1}^{N} \log p(w_t|w_{t+R}) \tag{7}$$

where N is the vocabulary size of the structured knowledge base.

To obtain a uniform of representation using word2vec, we define $p(w_t|w_{(t+R)})$ by the same softmax function, as shown below:

$$p(w_t|w_{t+c}) = \frac{\exp({e'_{w_t}}^{\top} \sum_{N_R} e_{w_{t+R}})}{\sum_{\theta=1}^{N} \exp({e'_{w_\theta}}^{\top} \sum_{N_R} e_{w_{t+R}})} \tag{8}$$

where e_w and e'_w are the "input" and "output" vector representations of w. $w_{(t+R)}$ is the word collection that has an R relationship with w, and N is the total number

of words in the knowledge resources. R indicates one specific kind of semantic relationship, specifically, hyponymy or synonymy in our domain ontology.

4.3 Joint word embedding model

To build our joint word embedding model, we incorporate domain-specific semantic relations into the word-embedding training procedure. The objective of the model is to maximize the probability of both the context constraint and the specific semantic relation constraint. Many words occur in both the corpus context and semantic relations in the knowledge bases. As demonstrated above, the semantic constraint is more accurate than the context constraint so the weight of the semantic relation constraint has priority. For example, if the incorporated semantic relationship constraint is hyponymy, then the task is to predict a word from both the words that have hyponymic relation with it in the knowledge bases and those surrounding it in the context, although the words from the knowledge bases are preferable to those in the context. Then, we define the objective function of the joint model based on Eqs. (5)–(8):

$$\frac{1}{M}\left(\frac{c-N_R}{c}\sum_{t=1}^{M}\log p(w_t|w_{t-c}^{t+c}) + \frac{N_R}{c}\sum_{t=1}^{M}\log p(w_t|w_{t+R})\right) \tag{9}$$

where M is the total number of words in the corpus and knowledge base, i.e., the size of union set of the two individual vocabularies. N_R is the number of concepts that have an R relationship with w_t in the knowledge base and is forced to be less than the context window size c. Usually, w_{t+R} is also less than c. If N_R is bigger than c, we select the top c semantically related words in our joint model.

When N_R equals c, we can guarantee that the constraint comes only from the semantic relationship constraint, so the joint model reverts to the traditional semantic relation embedding model described in Eq. (7). When N_R is 0, this means that no semantically related words were found in the knowledge bases, so the joint model reverts to the original word-embedding model solo from the context in Eq. (5). In this paper, the semantic relationship constraints are the *hyponymy* or *synonymy* of the CL ontology, and the context is the raw text from the AAN corpus. Our joint model uses the same training scheme as that in word2vec.

5 Experimental Evaluation

Dataset. There are several public datasets that can be used to evaluate the semantic similarity of short texts and sentences[11, 37, 38], but which cannot validate document-level semantic measurement. Since, as far as we know, there is no available document-level evaluation corpus, we constructed a new dataset for the task, in which we manually distinguished document pairs according to their degree of semantic similarity. We constructed this dataset using the academic papers in the CL domain. We generated 1021 paper pairs from the AAN corpus[36], which contains papers published at the past ACL conferences. We labeled each paper pair having both 2- and 5-level annotations as ground truth, and with 1 if they were semantically similar or 0 if they are dissimilar in the 2-level annotation. In the 5-level annotation, we marked a paper pair with integers ranging from 1 to 5 according to their degree of semantic similarity. The more semantically similar were two articles, the higher is their annotation. Annotation 5 indicates that they are totally semantically equal, and 1 indicates that they have no similarity whatsoever. Two experts in our lab independently annotated and cross verified each paper pair. The resulting corpus can be downloaded from the following url: https://github.com/buaaliuming/DSAP-document-semantics-for-academic-papers/tree/buaaliuming-annotation.

Baseline. We chose four representative models as benchmarks: the LDA-based, tf-idf, vectors averaging, and paragraph vector methods.

The LDA-based method described in Ref. [31], which is the most closely related to our research method, measures the semantic similarity between documents with divergent topic distributions. We used the SEMILAR toolkit[32] to measure the document-level semantic similarity. With respect to the LDA-based method, we found several LDA models with different parameters to be trained based on the AAN corpus. In the following presented results, for contrast, we chose the LDA model with 200 topics, which achieves the best performance among these different models under the same conditions.

The tf-idf method is the classical approach to measuring the similarity of long documents. We leveraged the method proposed in Ref. [18]. Averaging word vectors is the most common approach for sentence and document representation, so we adopted the word

averaging method to represent each long document and measure the semantic similarity.

The paragraph vector method[26] learns fixed-length feature representations from variable-length pieces of text. We utilized this state-of-the-art method to measure the similarity of long documents. Since individual long documents have no context, we adopted the Distributed Bag-Of-Words version of Paragraph Vector (PV-DBOW) model in Ref. [31] to train the document embedding model.

Semantic profile. We adopted the pattern-based method described in Section 3.2 for the enrichment of the semantic profile. In this method, the sentences are parsed by the Stanford Parser. We used the annotation tool GATE for the identified patterns of the semantic items in Table 2. In addition to the automatic enrichment of the semantic profile, which may accumulate errors during the following procedure, we also manually annotated the semantic profile, which can be regarded as a golden semantic profile.

As we all know, the research target is crucial in each academic article. The research domains, types of research work, and methods adopted in academic articles are important aspects, that are the discriminative characteristics of each study, whereas the keywords are a set of fuzzy descriptors that lacking sufficient semantic content. The dates of publication indicate the diverse ages of various technologies, which are less discriminative and not directly related to various research work. We set the weights of the aforementioned items according to their relative importance. In our experiments, we set the weights of the items target, domain, style, methodology, keywords, and date to 0.3, 0.25, 0.25, 0.1, 0.05, and 0.05, respectively.

Methods used. We conducted the experiments using several methods that use semantic profiles of different quality and different kinds of external knowledge resources. These semantic profiles are Automatically enriched Semantic Profiles (*AutoSP*) and manually annotated golden Semantic Profiles (*SP*). The external knowledge resources we used in our task were word vectors derived from the domain AAN corpus (*DomVec*) and from the universal Wikipedia dump (*UniVec*). Additionally, we used the CL Ontology (*Onto*), in which the terminology semantic similarities are measured using the Wu and Palmer's method[25].

In our experiment, we tested our semantic profile-based method using knowledge resources of different quality, i.e., the golden SP-based method with DomVec word-embedding model (*SP_DomVec*), the AutoSP-based method with DomVec word-embedding (*AutoSP_DomVec*), the golden SP-based method with Onto (*SP_Onto*), and the AutoSP-based method with Onto (*AutoSP_Onto*). To compare the quality of the different word vectors, we also compared the golden SP-based method with UniVec (*SP_UniVec*).

Evaluation metric. We chose the Spearman's rank correlation to measure the quality of the semantic similarity scores. The larger is the Pearson's correlation, the more correlated are the predicted scores and the ground truth. The Spearman's rank correlation is defined as shown in Eq. (10):

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{\mathbb{E}((X-\mu_X)(Y-\mu_Y))}{\sigma_X \sigma_Y} \quad (10)$$

where X and Y indicate the predicted variables and the real values, respectively. $\operatorname{cov}(X,Y)$ represents the covariance of X and Y, μ_X and μ_Y represent the mean values of X and Y, and σ_X and σ_Y are the standard deviations of X and Y. In this work, X is the predicted semantic similarity score and Y is the annotated semantic similarity value. The larger is the Pearson's correlation, the more they are correlated.

6 Results and Discussion

6.1 Impact of knowledge resources

To verify the quality of different methods, we calculated the Pearson's correlations of their similarity scores with the manually annotated ground truth. The results in Table 3 show that our methods have a significant overall advantage. The best correlation of the *SP* method with 5- and 2-level annotations is 0.710 and 0.579, respectively. We obtained significantly better correlation performance than several of the baseline methods.

The experimental results show that the background knowledge resources influence the final results of document-level semantic measurement. The word-embedding-based SP method performed much better than the ontology-based SP method. The distributed representations of the word-embedding method learned from the domain corpora are objective and more accurate, whereas the manually constructed domain ontology is inevitably subjective.

As demonstrated by the author of worde2Vec[4], a larger sliding window provides more training examples and thus can lead to higher accuracy, and more

Table 3 Pearson's correlations of different knowledge resources with ground truth.

Correlation	5-level annotation correlation	2-level annotation correlation
SP_UniVec	0.695	0.559
SP_DomVec	**0.710**	**0.575**
SP_Onto	0.559	0.461
LDA Method	0.537	0.250
Vector_Averaging	0.517	0.404
TF_IDF	0.519	0.245
Paragraph_Vector	0.541	0.327

dimensions also lead to word vectors of higher quality. In our experiments, the sliding window size of the domain-corpus-based word embedding was 8 and the dimension was 200, whereas the window size of the universal Wikipedia based word embedding was 10 and dimension was 1000. However, the domain word vectors still outperformed the universal vectors, as we can see in the results shown in Table 3.

6.2 Impact of automatic semantic enrichment

To evaluate the impact of automatic semantic enrichment, we also compared the annotations of our methods with the golden *SP* versus those generated by the *AutoSP*. In Table 4, we can see that the golden annotated *SP*s perform much better than the *AutoSP*s because the error associated with automatic semantic enrichment lowers the final performance of the SP-based method. Nevertheless, all of the SP methods still performed better than the the best baseline method. The *SP_DomVec* method showed a minor advantage over the *AutoSP_DomVec* method in both 5- and 2-level correlations, and the *AutoSP_Onto* method performed a little better than the *SP_Onto* method in the 5-level annotation correlation. This shows that the pattern-based extraction method can extract necessary information with the proper precision in a specific domain, and that the process of automatic extraction yields comparable performance to that of manual annotation.

Table 4 Pearson's correlations of different quality semantic profile with ground truth.

Correlations	5-level annotation correlation	2-level annotation correlation
SP_DomVec	**0.710**	**0.575**
AutoSP_DomVec	0.675	0.546
SP_Onto	0.559	0.461
AutoSP_Onto	0.569	0.456
Paragraph_Vector	0.541	0.327

6.3 Performance with different thresholds

In practical applications, a threshold must be set to predict whether two documents semantically match. We increased the threshold from 0.05 to 0.95. Paper pairs are thought to be semantically similar if their similar scores are greater than the threshold, otherwise they are considered to be semantically dissimilar. Our corpus has both 2- and 5-level annotations, and accuracy and the F-score can be used as the general evaluation metrics in addition to correlation. We measured accuracy as the percentage of document pairs predicted correctly. The F-score is a balance between precision and recall that indicates the comprehensive capacity of different methods.

As shown in Figs. 6 and 7, our methods obtained remarkably high performances in most cases. The best accuracy of our *SP* methods was 0.776, whereas the best performance of the other baseline methods, i.e., Vector_Averaging, was 0.758.

The F-scores of our methods are superior to those of the baseline methods at most thresholds. Our best

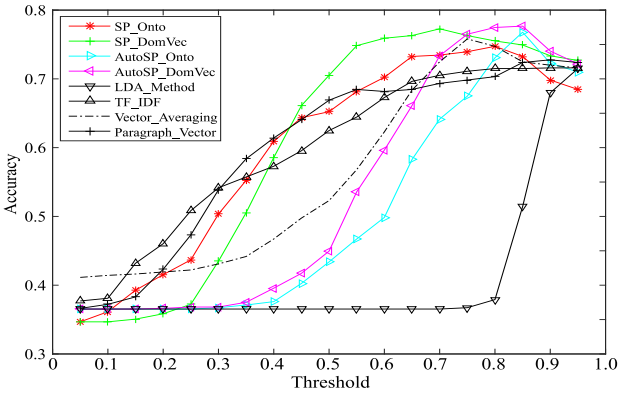


Fig. 6 Comparison of accuracy.

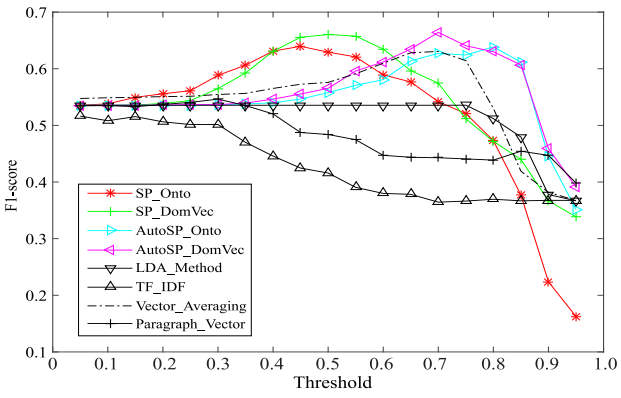


Fig. 7 Comparison of F1.

F-score was 0.664, whereas the best F-score of the baseline methods, i.e., Vector_Averaging, was 0.630. These experimental results demonstrate that the *SP* methods can yield much better overall performance.

Academic papers usually have lengthy content containing lots of duplicate terminologies, and frequent terminologies in a specific domain tend to occur in many domain papers. In addition, academic papers tend to review related work. The LDA-based method determines topic similarity by word overlap. Hence, the similarity scores of the LDA-based method are relatively high, ranging from 0.7 to 1.0, so its performance is good with respect to recall but poor in precision, which lowers its overall performance. The tf-idf method focuses on the similarity of word morphology but cannot capture intrinsic word meanings. The vector averaging and paragraph vector methods capture the semantics of each word but their BOW assumption loses the information about word order and document structure. The up-to-date method cannot capture well the semantic differences between academic papers. Our methods hold the core semantics of a document directly via multi-faceted semantic profiles, the similarity scores of our SP-based methods fluctuate from 0.0 to 1.0, and they are more discriminative and have better overall performance.

6.4 Performance of joint word embedding

As described in Section 5, we train word vectors by incorporating the domain-specific semantic relationship from the CL ontology and context from the domain corpus. To verify the impact of fine-grained relationships in training word vectors, we trained different kinds of word vectors and used them in the final document matching task. We used the task-oriented evaluation of the word vectors in the document semantic similarity task and conducted our experiments with several joint word-embedding models using different kinds of semantic relation constraints from our domain ontology. These include joint word embedding with both synonymy and hyponymy (*SP_Context_AllRelations*), joint word embedding with synonymy (*SP_Context_Synonym*), joint word embedding with hypernymy (*SP_Context_Hypernym*), and the joint word embedding with hyponymy (*SP_Context_Hyponym*), and the original word embedding-based method without any semantic constraint served as our baseline method (*SP_Context)*.

As we can see from the results in Table 5, generally, the sum of the relation constraints from the structured knowledge bases enhanced the word-embedding performance, which accords with our expectation. Specifically, our new discovery is that the synonyms constraint is primarily devoted to the quality of the word representations, whereas the joint word embeddings with the hyponymy constraint produce minor wastage.

Table 5 Correlations of different joint word embeddings with ground truth.

Relation constrain	5-level annotation correlation	2-level annotation correlation
SP_Onto	0.559	0.460
SP_Context_AllRelations	0.697	0.480
SP_Context	0.695	0.495
SP_Context_Synonym	**0.714**	**0.497**
SP_Context_Hypernym	0.693	0.480
SP_Context_Hyponym	0.683	0.460

7 Conclusion

In this paper, we proposed the semantic profile and a semantic enrichment based method for calculating the semantic similarity of long documents and we presented the first semantic evaluation corpus of long documents. First, we proposed the semantic profile to represent the global meaning of academic articles and we enriched this semantic profile with external knowledge resources. Then, we determined the concept semantic similarities via word embedding to evaluate the similarity of the semantic profiles. To obtain superior word representations, we further proposed a joint word-embedding mode, in which we incorporate domain-specific semantic relations with the traditional context constraint to train the word representations. Our experimental results demonstrate that our method achieves significantly better performance than baseline methods. External knowledge is essential for understanding long document, and we found that domain synonym relations can be used to significantly improve word representations in our joint word-embedding model.

Our method requires external knowledge resources of high quality. In practical applications, semantic profiles can be constructed automatically via the utilization of external knowledge resources, and the performance of our method can be promoted along with the improvement of external knowledge resources.

Acknowledgment

This research was supported by the Foundation of the State Key Laboratory of Software Development Environment (No. SKLSDE-2015ZX-04).

References

[1] J. B. Tenenbaum, C. Kemp, T. L. Griffiths, and N. D. Goodman, How to grow a mind: Statistics, structure, and abstraction, *Science*, vol. 331, no. 6022, pp. 1279–1285, 2011.

[2] J. Y. Pan, C. P. J. Cheng, G. T. Lau, and K. H. Law, Utilizing statistical semantic similarity techniques for ontology mapping—with applications to AEC standard models, *Tsinghua Sci. Technol.*, vol. 13, no. S1, pp. 217–222, 2008.

[3] C. Leacock and M. Chodorow, *Combining Local Context and WordNet Similarity for Word Sense Identification*. The MIT Press, 1998.

[4] T. Mikolov, K. Chen, G. Corrado, and J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv: 1301.3781, 2013.

[5] P. Resnik, Using information content to evaluate semantic similarity in a taxonomy, in *Proc*. 14th *Int. Joint Conf. Artificial Intelligence*, Montreal, Canada, 1995.

[6] V. Rus, M. C. Lintean, A. Graesser, and D. McNamara, Assessing student paraphrases using lexical semantics and word weighting, in *Proc*. 14th *Int. Conf. Artificial Intelligence in Education*, Brighton, UK, 2009.

[7] C. Corley and R. Mihalcea, Measuring the semantic similarity of texts, in *Proc. ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, Ann Arbor, MI, USA, 2005, pp. 13–18.

[8] Z. Xu, X. F. Luo, S. X. Zhang, X. Wei, L. Mei, and C. P. Hu, Mining temporal explicit and implicit semantic relations between entities using web search engines, *Future Generat. Comput. Syst.*, vol. 37, pp. 468–477, 2014.

[9] Z. Xu, X. F. Luo, J. Yu, and W. M. Xu, Measuring semantic similarity between words by removing noise and redundancy in web snippets, *Concurr. Comput. Pract. Exp.*, vol. 23, no. 18, pp. 2496–2510, 2011.

[10] Z. Xu, X. F. Luo, L. Mei, and C. P. Hu, Measuring the semantic discrimination capability of association relations, *Concurr. Comput. Pract. Exp.*, vol. 26, no. 2, pp. 380–395, 2014.

[11] E. Agirre, C. Banea, C. Cardie, D. Cer, M. Diab, A. Gonzalez-Agirre, W. W. Guo, I. Lopez-Gazpio, M. Maritxalar, R. Mihalcea, et al., SemEval-2015 task 2: Semantic textual similarity, English, Spanish and pilot on interpretability, in *Proc*. 9th *Int. Workshop on Semantic Evaluation* (*SemEval 2015*), Denver, CO, USA, 2015.

[12] F. Šaric, G. Glavaš, M. Karan, J. Šnajder, and B. D. Bašic, Takelab: Systems for measuring semantic text similarity, in *Proc*. 6th *Int. Workshop on Semantic Evaluation*, Montréal, Canada, 2012, pp. 441–448.

[13] D. Bär, C. Biemann, I. Gurevych, and T. Zesch, UKP: Computing semantic textual similarity by combining multiple content similarity measures, in *Proc*. 1st *Joint Conf. Lexical and Computational Semantics*, Montréal, Canada, 2012.

[14] W. Han, X. Zhu, Z. Zhu, W. Chen, W. Zheng, and J. Lu, A comparative analysis on weibo and twitter, *Tsinghua Sci. Technol.*, vol. 21, no. 1, pp. 1–16, 2016.

[15] M. Y. Zhang, B. Qin, T. Liu, and M. Zheng, Triple based background knowledge ranking for document enrichment, in *Proc. COLING 2014, the 25th Int. Conf. Computational Linguistics: Technical Papers*, Dublin, Ireland, 2014.

[16] M. Schuhmacher and S. P. Ponzetto, Knowledge-based graph document modeling, in *Proc*.7th *ACM Int. Conf. Web Search and Data Mining*, New York, NY, USA, 2014, pp. 543–552.

[17] D. Ramage, A. N. Rafferty, and C. D. Manning, Random walks for text semantic similarity, in *Proc. 2009 Workshop on Graph-Based Methods for Natural Language Processing*, Suntec, Singapore, 2009, pp. 23–31.

[18] M. Y. Zhang, B. Qin, M. Zheng, G. Hirst, and T. Liu, Encoding distributional semantics into triple-based knowledge ranking for document enrichment, in *Proc*. 53rd *Annual Meeting of the Association for Computational Linguistics and the 7th Int. Joint Conf. Natural Language Processing*, Beijing, China, 2015.

[19] G. Salton, A. Wong, and C. S. Yang, A vector space model for automatic indexing, *Commun. ACM*, vol. 11, no. 11, pp. 613–620, 1975.

[20] G. A. Miller, WordNet: A lexical database for English, *Commun. ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[21] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, Freebase: A collaboratively created graph database for structuring human knowledge, in *Proc. 2008 ACM SIGMOD Int. Conf. Management of Data*, Vancouver, Canada, 2008, pp. 1247–1250.

[22] T. K. Landauer, P. W. Foltz, and D. Laham, An introduction to latent semantic analysis, *Dis. Process.*, vol. 25, nos. 2&3, pp. 259–284, 1998.

[23] D. Q. Wang, H. Zhang, R. Liu, X. L. Liu, and J. Wang, Unsupervised feature selection through Gram-Schmidt orthogonalization—A word co-occurrence perspective, *Neurocomputing*, vol. 173, pp. 845–854, 2016.

[24] D. M. Blei, A. Y. Ng, and M. I. Jordan, Latent dirichlet allocation, *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.

[25] Y. Bengio, H. Schwenk, J. S. Senécal, F. Morin, and J. L. Gauvain, Neural probabilistic language models, in *Innovations in Machine Learning*, D. E. Holmes and L. C. Jain, eds. Springer, 2006, pp. 137–186.

[26] Q. V. Le and T. Mikolov, Distributed representations of sentences and documents, arXiv preprint arXiv: 1405.4053, 2014.

[27] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, DBpedia: A nucleus for a web of open data, in *The Semantic Web*, K. Aberer, K. S. Choi, N. Noy, D. Allemang, K. I. Lee, L. Nixon, J. Golbeck, P. Mika, D. Maynard, R. Mizoguchi, et al., eds. Springer, 2007.

[28] J. Pennington, R. Socher, and C. D. Manning, GloVe: Global vectors for *word representation, in Proc.*

2014 Conf. Empirical Methods in Natural Language Processing (*EMNLP*), Doha, Qatar, 2014.

[29] E. Gabrilovich and S. Markovitch, Computing semantic relatedness using Wikipedia-based explicit semantic analysis, in *Proc. 20th Int.Joint Conf. Artifical Intelligence*, Hyderabad, India, 2007, pp. 1606–1611.

[30] M. Rafi and M. S. Shaikh, An improved semantic similarity measure for document clustering based on topic maps, arXiv preprint arXiv: 1303.4087, 2013.

[31] V. Rus, N. Niraula, and R. Banjade, Similarity measures based on latent Dirichlet allocation, in *Computational Linguistics and Intelligent Text Processing*, A. Gelbukh, ed. Springer, 2013, pp. 459–470.

[32] V. Rus, M. Lintean, R. Banjade, N. Niraula, and D. Stefanescu, SEMILAR: The semantic similarity toolkit, in *Proc. 51st Annual Meeting of the Association for Computational Linguistics*, Sofia, Bulgaria, 2013, pp. 163–168.

[33] Z. B. Wu and M. Palmer, Verb semantics and lexical selection, in *Proc. 32nd Annual Meeting on Association for Computational Linguistics*, Las Cruces, NM, USA, 1994, pp. 133–138.

[34] D. Fried and K. Duh, Incorporating both distributional and relational semantics in word representations, arXiv preprint arXiv: 1412.4369, 2014.

[35] M. Yu and M. Dredze, Improving lexical embeddings with semantic knowledge, in *Proc. 52nd Annual Meeting of the Association for Computational Linguistics* (*Short Papers*), Baltimore, MD, USA, 2014, pp. 545–550.

[36] D. R. Radev, P. Muthukrishnan, and V. Qazvinian, The ACL anthology network corpus, in *Proc. 2009 Workshop on Text and Citation Analysis for Scholarly Digital Libraries*, Suntec, Singapore, 2009, pp. 54–61.

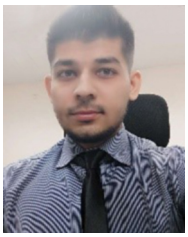
[37] B. Dolan, C. Quirk, and C. Brockett, Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources, in *Proc. 20th Int. Conf. Computational Linguistics*, Geneva, Switzerland, 2004, p. 350.

[38] V. Rus, M. Lintean, C. Moldovan, W. Baggett, N. Niraula, and B. Morgan, The SIMILAR corpus: A resource to foster the qualitative understanding of semantic similarity of texts, in *Proc. 8th Language Resources and Evaluation Conf.*, Instanbul, Turkey, 2012, pp. 23–25.

Ming Liu is a PhD student at the State Key Laboratory of Software Development Environment, Beihang University. He received the BS degree from the China University of Petroleum, China, in 2010, and the master degree in computer networking from Ningxia University, China, in 2013. His current research interests include text mining, scientific literature analysis, and knowledge graph representation learning.

Zepeng Gu is a graduate student in computer science and technology at the State Key Laboratory of Software Development Environment, Beihang University, China. He received the BS degree from Beihang University, China, in 2015. His research interests include text clustering, deep learning, and multimodal retrieval.

Bo Lang received the PhD degree in computer science from Beihang University, Beijing, China in 2003. She is a professor at the School of Computer Science and Engineering, Beihang University. Her current research interests include information security, data management, and information retrieval.



Ahmed Zeeshan is a graduate student in computer science and technology at the State Key Laboratory of Software Development Environment, Beihang University, China. He received the BS degree from University of Education Pakistan, Pakistan, in 2014. His research interests include semi-automatic ontology construction, scientific literature analysis, and key phrase extraction.