Knowledge-Based Systems 190 (2020) 105030

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Word Sense Disambiguation: A comprehensive knowledge exploitation framework☆

Yinglin Wang [a,*], Ming Wang [a], Hamido Fujita [b]

[a] School of Information Management and Engineering, Shanghai University of Finance and Economics, Shanghai, China
[b] Faculty of Information Technology, Ho Chi Minh City University of Technology (HUTECH), Ho Chi Minh City, Vietnam

ARTICLE INFO

Article history:
Received 13 February 2019
Received in revised form 6 September 2019
Accepted 7 September 2019
Available online 24 September 2019

Keywords:
Word sense disambiguation
Background knowledge
Information retrieval
Relation exploitation
Semantic path

ABSTRACT

Word Sense Disambiguation (WSD) has been a basic and on-going issue since its introduction in natural language processing (NLP) community. Its application lies in many different areas including sentiment analysis, Information Retrieval (IR), machine translation and knowledge graph construction. Solutions to WSD are mostly categorized into supervised and knowledge-based approaches. In this paper, a knowledge-based method is proposed, modeling the problem with semantic space and semantic path hidden behind a given sentence. The approach relies on the well-known Knowledge Base (KB) named WordNet and models the semantic space and semantic path by Latent Semantic Analysis (LSA) and PageRank respectively. Experiments has proven the method's effectiveness, achieving state-of-the-art performance in several WSD datasets.

1. Introduction

In NLP area, ambiguity is recognized as a barrier to human language understanding. Word Sense Disambiguation (WSD), an AI-complete problem [1], is shown to be able to solve the essential problems of artificial intelligence, and has received increasing attention due to its promising applications in the fields of Sentiment Analysis [2], Information Retrieval [3], Information Extraction [4], Machine Translation [5], Knowledge Graph Construction [6], etc. WSD tasks are often classified into two types: lexical sample WSD and all-word WSD. The former focuses on disambiguating only some particular target words, while the latter conducts WSD on every word in a document.

Through many years of research, different solutions of WSD has been proposed, which can generally be divided into supervised approaches and knowledge-based ones [1]. For supervised approaches, WSD is modeled as a classification problem, with each classifier dealing with one target word. Each classifier is trained separately with all annotated samples concerning a particular target word. Although with great demand of a large labeled corpus, most approaches in this category can outperform those in the knowledge-based category [7]. However, there is no significant performance improvement of the approaches in this category recently. In comparison, the knowledge-based approaches have gained a rapid development in recent years due to their independence of an expensive sense-annotated corpus. Therefore, the performance gap between the two kinds of approaches has been narrowed. Even, some knowledge-based approaches can achieve better performance than supervised ones in the latest WSD dataset.

The central idea of knowledge-based approaches is to fully make use of the knowledge in KBs such as WordNet [8] and BabelNet [9]. There are mainly two streams of researches in this category. One is to consider overlap, or similarity, between the context of a word under disambiguation and the relevant information such as the definition of a potential sense and its related sense obtained from a KB. Then, the most similar sense is regarded as the predicted sense. The other is to build a graph based on provided context and all connections retrieved from some KBs. Then different graph-based algorithms, e.g. PageRank [10] and Latent Dirichlet Allocation [11] (LDA), are employed to predict the sense of a given word using the built graph. The major challenge in this category is how to take full advantage of background information or knowledge from KBs to mimic the way human disambiguates a word. For example, relations within KBs might be treated differently based on their features. Further, it might contribute to combining the massive background knowledge from different KBs. However, there is currently no comprehensive framework regarding background knowledge utilization with which researchers can follow to model this problem.

In this paper, we focus on the task of all-word WSD and propose a WSD framework that exploits background knowledge

☆ No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.knosys.2019.105030.

* Corresponding author.
E-mail addresses: wang.yinglin@shufe.edu.cn (Y. Wang), lwmlyy@163.com, wangming@163.sufe.edu.cn (M. Wang), h.fujita@hutech.edu.vn (H. Fujita).

https://doi.org/10.1016/j.knosys.2019.105030

from three perspectives. First, a unique distributional vector is used to represent each word in the disambiguation process. Word representation vectors learned from a large corpus has been proven beneficial to disambiguating words [12,13]. We use word representations from LSA [14] to capture the semantic space in documents. Furthermore, most WSD documents are domain-specific, thereby triggering the need for domain knowledge from some KBs. A naïve IR is used to retrieve domain documents from Wikipedia, to further combine different background information together for LSA to learn the semantics in documents. Finally, the core competitor of a knowledge-based WSD approach is to utilize knowledge in lexical knowledge bases such as WordNet. A novel way has been developed in this research to exploit sense relations in WordNet, treating them differently according to their relativity and contribution.

Although inspired by previous researches, our work has its own contributions as follows. Different from the previous researches, we employ a naïve IR to retrieve documents from Wikipedia so as to incorporate domain-specific background information. The contribution of this document retrieval process has been validated by experiments using the latest standard WSD dataset. More importantly, we put forward a new exploitation method of WordNet relations based on relativity and contribution from semantic and statistical perspective. To our best knowledge, this has not been investigated by previous researches. In addition, we propose an approach to model the sense path within a particular sentence, making it possible to reveal the semantic relationship and connections between senses. It is different from other researches by introducing similarity-based connections between senses in the disambiguation network and initializing sense importance with similarity calculated by an overlap-based approach.

The remainder of this paper is organized as follows. In Section 2, a comprehensive review of related works is demonstrated. Section 3 introduces the main proposals regarding how to obtain domain-specific information and its employment, sense relations exploitation and semantic path exploration within sentences. Section 4 illustrates the experiment setup. The experiment results are thoroughly analyzed and discussed in Section 5 from several perspectives. Finally, Section 6 concludes the paper and presents some possible future directions.

2. Related work

Many efforts have been made to tackle the WSD problem since its emergence. The research on WSD has been accelerating due to its extensive applications [2–6]. As a result, a number of new systems [15–21] were developed and validated on standard datasets [22–26] that are specialized in WSD evaluation. In addition, a comprehensive framework[1] [7] has also been put forward, with the framework integrating the major WSD datasets for the fair comparison of different systems' performance and reproducing the state-of-the-art results.

In terms of performance, supervised WSD approaches are superior in most cases since they can learn the mapping between some features and the target sense from a relatively large annotated corpus. This has also been verified by the results of several WSD competitions. Early attempts in this category include Decision Trees [27], Naive Bayes [28], Neural Networks [29], Support Vector Machine (SVM) [30] and even some ensemble methods [31,32].

In recent years, more supervised systems are proposed to cope with WSD tasks. "It makes sense" (IMS) [33] is a system that employs SVM to disambiguate words in context. It is the first comprehensive system publicly available for WSD. Owing to the great progress of deep learning techniques, an extended version [17] of IMS was developed, which incorporates word embeddings as a feature to train the classifiers (word experts) [21] and further improves the performance. SUPWSD [34] is another newly available WSD supervised framework. Its WSD algorithm is the same as that of IMS but it employs a much larger sense-annotated training corpus and provides more flexibility for customization. More importantly, the SUPWSD system performs much better and faster than IMS. In addition, some deep neural network models such as bidirectional long short-term memory (Bi-LSTM) network employed to tackle WSD, including Context2Vec [35], have shown better results compared to other supervised methods. Besides, IMBHN [36] is a network-based approach that models the relationship between context and ambiguous words with bipartite networks and shows its robustness in using even a small training dataset. SDK [37] applies semantic diffusion kernel in SVM to conduct WSD while capturing the semantic similarity between terms. This is further improved by considering WSD labels in the kernel method [38].

Up until then, WSD was dealt with a bunch of word experts to disambiguate target words separately in a low-efficiency manner. Recently, a new and effective approach [21] is proposed for modeling supervised WSD. Instead of a great many classifiers, this approach utilizes a single all word sequence learning model for disambiguation and achieves satisfying results on standard WSD datasets. Other developments in this category includes investigating the performance boost of employing a large semi-constructed sense-annotated corpus [7], automatically constructing a large sense-annotated corpus for supervised WSD training and employing multilingual resources for sense-annotated corpus construction [39].

By taking advantage of abundant information from different knowledge sources, the performance gap between knowledge-based approaches and the supervised ones is narrowing. The earliest and most intuitive approach in this category is called Lesk [40], which calculates the word overlap between potential senses' definition and the target word's context. In order to overcome its disadvantage of high dependence on definition's length or vocabulary, adapted Lesk [41] was developed by considering the sense relations in the sense inventory, which raised the probability of overlapping. Further, enhanced Lesk [12] exploits more information by learning distributional word representation from a large corpus and thus replaces the bag of words overlap with a cosine similarity of vectors, and state-of-the-art performance was achieved on a contemporary WSD dataset. Unlike Lesk approaches, graph-based approaches mainly rely on various graph-based algorithms to operate on the graph constructed based on context and abundant connections within some knowledge bases. UKB [15,16] performs random walks on a large semantic graph (WordNet) to determine the sense of a given word by means of PageRank and personalized PageRank (PPR). Further, adapted PPR [18] integrates a pool of other knowledge bases with WordNet to perform PageRank and takes advantage of sense frequency over competing senses. Similarly, Babelfy [42] conducts random walks on a more comprehensive semantic network (BabelNet, a vast multilingual knowledge graph) to perform disambiguation, regardless of the language. Recently, WSD-TM [19] performs a variant of LDA on the whole ambiguous document with senses from WordNet, in order to fully exploit the context information. Unlike traditional LDA, WSD-TM assumes that a document is formulated with concept of synsets and synset words rather than topics and topic words. Similar to the above approaches, two other systems [43,44] were developed based on game theory and multi-objective optimization, respectively.

Due to adoption of different knowledge exploitation strategies, the performance of various knowledge-based methods can

[1] http://lcl.uniroma1.it/wsdeval

vary greatly. Basically, graph-based approaches achieve better results owing to the utilization of massive sense relations within a semantic network. In other words, it is beneficial to combine all the related senses in context for disambiguation. This is different from the Lesk approaches, which disambiguate a single word each time with all the senses related to that particular word. However, these graph-based approaches are also limited by the strategy they perform WSD with, and they neglect the background knowledge from other sources.

In order to achieve better knowledge exploitation, different techniques can be employed. One effective technique is to represent a word with a dense vector learned from an enormous corpus consisting of both general and domain information. In the case of WSD, while the general information can be obtained from a relatively large corpus, the domain information should be obtained according to the relevance between potential domain documents and WSD texts by means of information retrieval techniques or others [45,46]. In regard to the sense relation exploitation in the latest Lesk approach, although senses are treated differently according to their occurrence and the distance with the potential target sense, different types of sense relations (hypernym, hyponym, etc.) are utilized in the same manner. In addition, the relation of senses from different context words to be disambiguated is not properly considered. Therefore, except for adding domain knowledge, we have also combined the similarity-based and graph-based method together to perform WSD, attempting to make better use of available information.

3. Proposed approach

Fig. 1 is an overall workflow diagram of the proposed WSD framework. There are mainly four components of the framework in order: semantic space exploration, relation extraction, similarity calculation and semantic path exploration. An abstract for each part will be given next and the detailed discussion is illustrated in the following sections.

Semantic space exploration: Starting from WSD documents, a naïve IR is preformed to retrieve domain knowledge documents. Combined with the British National Corpus, these documents are used to learn word representation vectors by means of LSA. The obtained vectors are for further use.

Relation extraction: After WSD documents are preprocessed, lemma and part-of-speech (POS) of all the words in a document are attained, grouped by sentences. For each ambiguous word, all its potential senses are retrieved from WordNet according to its lemma and POS. For each sense (original/potential sense), an extended gloss can be obtained by synset retrieval algorithm (see Algorithm 1 for details). The extended gloss is a bag of words that appear in the gloss and examples (example of usage) of either the original sense or its connected senses. The connection rules of a target sense to its extended senses will be discussed in Section 3.2.

Similarity calculation: With the word collections from the context and the extended gloss of each ambiguous word, their corresponding unified vectors can be obtained from the above learned vectors by certain weighting strategies of the word vectors. The dot product of these two vectors represents the similarity measurement of a potential sense to the correct sense, representing a confidence vote of being the correct sense. Potential senses of each word under disambiguation can be ranked according to the calculated similarity.

Semantic path exploration: In order to capture the semantic path in a particular sentence, top 3 (ranked by the above similarity) extended gloss are drawn out and incorporated into a sentence level network for further disambiguation, with those senses of low confidence discarded. Within a sentence, all top 3 senses and their related synsets can be attained to construct a relatively small network. The words in the gloss and examples of those related synsets are transformed into disambiguated synsets using an external WordNet related resource. The connection of these synsets are provided in WordNet. After that, PageRank is employed to decide the significance of those potential senses in the network. Finally, the node (sense) significance can serve as a weight to adjust the former similarity. The detailed procedure of semantic path exploration will be introduced in Section 3.3.

Table 1
Domain knowledge document name and score.

Wikipedia document name	Score
Vitamin B12	910.2108
Non-small-cell lung carcinoma	909.3884
Treatment of lung cancer	756.6191
Chemical biology	598.07
Squamous cell carcinoma	582.4913
Head and neck cancer	573.975
DNA	482.6179
Renal cell carcinoma	462.0006
Gemcitabine	426.1299
Virus	366.6102
European Medicines Agency	366.3686
Lung cancer	363.0137

3.1. Domain knowledge retrieval and semantic space exploration

In many cases, word ambiguity originates from domain discrepancy. For instance, the word ‘chips’ in the sentence ‘The box contains real chocolate chips’ has several senses which are from highly independent domains such as food, science and technology, finance, etc. This type of ambiguity can be tackled by simple domain knowledge since all the senses of its context word ‘chocolate’ are in food domain. Furthermore, the fact that most knowledge-based systems cannot outperform a simple strategy of picking the most frequent sense indicates that domain knowledge is necessary to bias the word representation vectors learned from a large general corpus to cohere with the documents being disambiguated.

WSD is a complex issue and has many downstream applications. Knowledge-based approaches employ knowledge graphs as a source for disambiguation while the results of WSD are vital for knowledge graph construction. This indicates a mutually reinforcing relationship between WSD and its applications. In order to attain domain knowledge documents, we adopt a simple strategy of information retrieval (a WSD application) based on TF–IDF dot product. The detailed document retrieval mechanism is similar to the ‘Document Retriever’ in [45]. The main purpose of this retrieval process is to search for documents whose major topic is relatively close to that of the ambiguous documents.

The queries come from two sources. One is a moving window of the ambiguous documents. The other is an employment of the sense inventory, WordNet. Specifically, for each ambiguous word, we concatenate the definition and examples of every potential sense, with each sense forming a query. Table 1 is a demonstration of top retrieved documents from Wikipedia, with document scores (TF–IDF dot product of certain query and the document) acting as the ranking values for document selection. The query source is a biomedicine document from Semeval-15 [26], which integrates three domain specific documents. It shows a promising result since the documents are highly related to biomedicine. The same results can be obtained for computer and math documents.

In NLP area, words are often represented as one-hot vectors for practical computation in early years. Then, these vectors can be applied to various machine learning models for particular prediction tasks. Their sparsity and high dimension can lead to

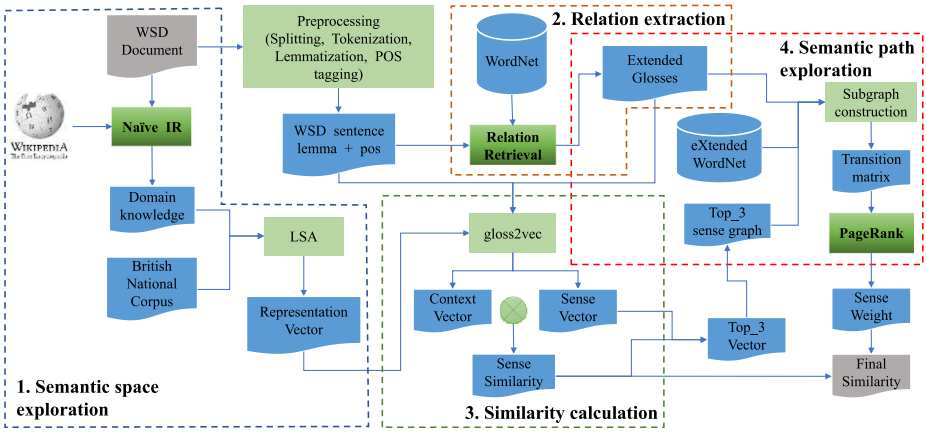

Fig. 1. Workflow of the proposed WSD framework.

low efficiency and effectiveness. Currently, word embeddings learned from a massive corpus are becoming increasingly prevalent due to its high usability and shallow semantic features. There are ongoing researches about learning embeddings for particular tasks including sentiment analysis [47] in fine-tuned approaches. However, these embeddings learned by language models cannot capture document level or topical features, which indicate their disadvantage in solving some deep semantic tasks, especially in the WSD task. Besides, embeddings learned from an enormous corpus are exceedingly general for domain specific tasks.

For semantically clustering words, LDA has attracted many researchers' attention because of its influential effect on various NLP tasks. For a large collection of documents, LDA is capable of clustering words into a given number of topics. It can also return a probability distribution of all the words over topics and all the documents over topics, allowing it to be a distributional semantic model. This type of model can perform dimensionality reduction on a collection of documents and capture the semantic structure and information in the documents simultaneously. However, LDA is regarded unstable [48] and its output is different given the same input (documents in the same order). This will have a severe impact on the WSD systems based on similarity approach. In addition, its time efficiency is relatively low compared to LSA or probabilistic LSA, a simpler way of representing the semantics of words by vectors. Hence, LSA is employed for further research.

3.2. WordNet relation exploitation

WordNet is a vast lexical database for English. The basic concept in WordNet is called synset, a collection of words sharing the same meaning. For each synset, a definition (gloss) and some examples of usage are provided. Synsets of the same POS (noun, verb, adjective and adverb) are connected under some relations separately, although relations might exist when the central idea of two words are the same but in different POS (e.g. adopt and adoption, defined as 'derivationally related synsets' in WordNet). These relations include hypernym, hyponym, meronym, holonym, attribute, entailment, etc. Among these relations, the first two are the most fundamental and frequent ones, with others only existed in particular POS. For example, meronym and holonym are for nouns while entailment is for verb and antonym is for adjectives and adverbs. Therefore, in the interest of a better exploitation of the relations, the focus should be on more common relations, i.e. hypernym and hyponym. Although these two relations only exist for nouns and verbs, a better exploitation of them can benefit the whole disambiguation process to a greater extent since nouns and verbs are the major components in most documents. In the scenario of WSD, hypernyms should be given more attention than hyponyms, which can be interpreted from two perspectives, semantics and statistics.

From a semantic perspective, a particular synset is more related to its hypernyms. In detail, the most important information carriers such as nouns and verbs in WordNet are arranged into hierarchies. Starting from a particular synset within the hierarchy, the deeper the related synset is, the more the information content [49] it carries. This indicates the related synset carries more information that is different from the starting synset. In other words, a synset has all the features that its hypernyms have but its hyponyms always have some different features. For example, in Fig. 2, a 'bank' inherits all the features from a 'financial institution', a 'institution', etc. It only becomes a 'thrift institution' when it encourages personal savings and home buying.

From a statistical point of view, a synset's hypernym words appear more frequently in the mention or context of the synset than hyponym words. Specifically, this can be reflected by the contribution and noise of the word collection retrieved from either hypernym or hyponym relation, which is demonstrated in Fig. 2. The calculation of the contribution and noise for each word is conducted in five standard WSD datasets [22–26]; we average the results over all the words in the datasets for a better comparison. In detail, contribution can be represented as the ratio of overlap between the context and the word collection retrieved by either relation, to the context of the word under disambiguation. For each word in all the WSD datasets, we calculate the above ratio based on the word's correct sense (Note that here we use the correct sense from annotation for statistical analysis) and the whole document that the word is in as context. The average ratio of contribution from hypernyms and hyponyms are 0.48 and 0.41 respectively. This shows a higher contribution of hypernyms than hyponyms from an overall view.

Besides, noise degree is depicted by irrelevant words that do not appear in the context (the whole WSD document) but in the extended hypernym or hyponym gloss. It is averaged over all the ambiguous words in five WSD datasets for both relations separately. The depth of relation extraction varies from 1 to 15 so as to show the ascending trend of noise degree and the gap between two relations. For example, when the extraction depth is 1, the extended gloss only contains words in the hypernyms/hyponyms that are directly connected to the correct sense. As is illustrated in Fig. 3, noise level from hyponym gloss is significantly higher than that from hypernym gloss and the gap expands as the extraction depth increases. It is noteworthy that both curves tend to be flat when the depth exceeds 9 because most synsets do not have hypernyms or hyponyms that deep in WordNet. To sum up, hypernyms are better contributors in the task of WSD from both semantic and statistical viewpoints.

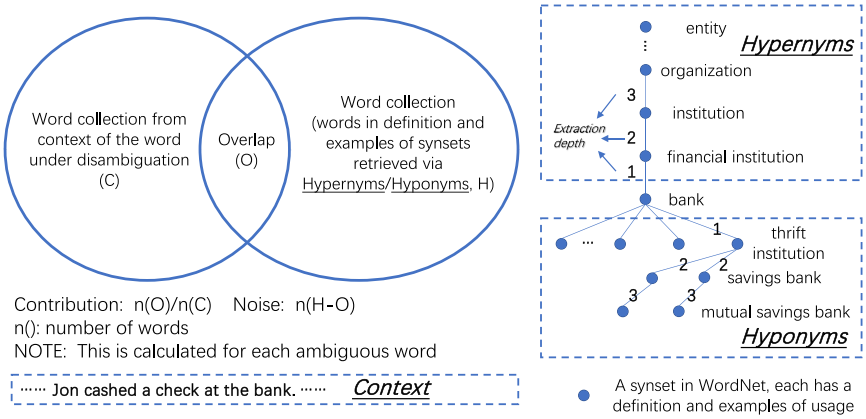

Fig. 2. Calculation of contribution and noise of Hypernyms and Hyponyms.

With the above conclusions in mind, we propose an asymmetric strategy of relation extraction for the gloss extension of a potential sense (synset), paying more attention to hypernyms. The extraction process involves three steps. First, given a potential synset, all its relations mentioned above but hypernyms will be extracted, returning synsets directly connected to the potential synset. All the other relations (hyponym, holonym, meronym, entailment, attribute) are considered because researches have validated their value for disambiguation [12]. Then, based on the bag of synsets retrieved at the first step, all their hypernyms will be extracted regardless of depth. It turns out to be beneficial to retrieve all hypernyms, which does not lead to low time efficiency since the increased number of hypernyms shrinks swiftly as the extraction depth rises. Finally, with all the synsets retrieved above, we extract all their derivationally related synsets.

By combining all the extracted synsets' gloss and examples, the extended gloss (bag of words) of a potential sense can be obtained. Algorithm 1 demonstrates the detailed procedure of synset retrieval and Fig. 4 illustrates the major concept of the relation exploitation method. Based on the obtained results, we calculate the similarity between the context of a word and the extended gloss of the senses to be disambiguated. Suppose $\overrightarrow{x_w} = (x_{w1}, \ldots, x_{wn})$ is the context vector of a word w, and $\overrightarrow{y_{w,s_i}} = (y_{w1,s_i}, \ldots, y_{wn,s_i})$ is the extended gloss vector of the ith sense of w, then the similarity of $\overrightarrow{x_w}$ and $\overrightarrow{y_{w,s_i}}$ can be calculated as their dot product, which is shown in formula (1).

$$Sim_{w,s_i} = Sim\left(\overrightarrow{x_w}, \overrightarrow{y_{w,s_i}}\right) = \overrightarrow{x_w} \cdot \overrightarrow{y_{w,s_i}} = \sum_{j=1}^{n} x_{wj} y_{wj,s_i} \quad (1)$$

Thus, we can rank all potential senses according to the values of Sim_{w,s_i}. These vectors $\overrightarrow{x_w}$ and $\overrightarrow{y_{w,s_i}}$ are acquired by utilizing the word representation learned from LSA with certain weighted sum strategies. For the vectors of context words, a uniform weighted scheme is adopted. As for the vectors of words in the extended glosses, each word's vector is weighted according to the distance between the original synset and the synset whose definition or examples contain this word, similar to the weighting scheme in [12] but only considers synset distance.

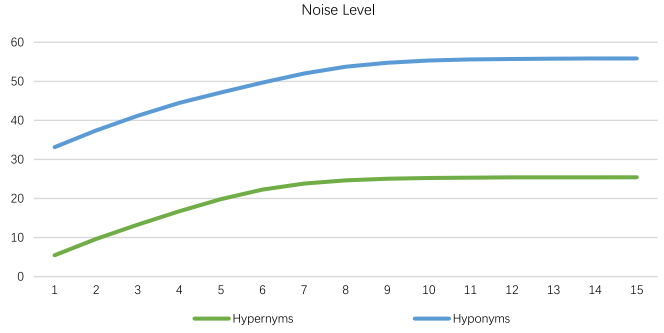

Fig. 3. Noise degree of extended hypernym and hyponym glosses.

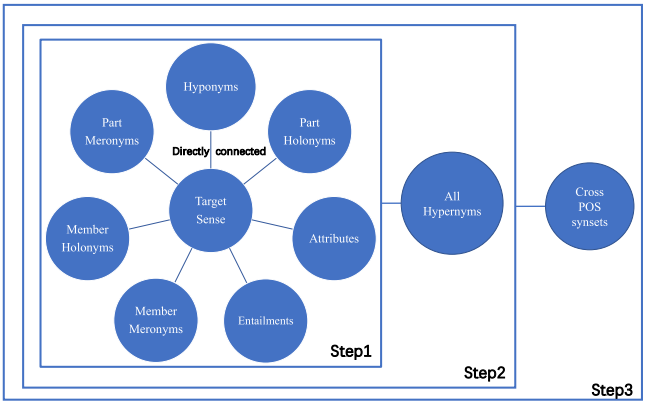

Fig. 4. Relation exploitation scheme.

The result shows that by discarding the senses which are not top 3, the ceiling performance only decreases by about 10% while noises are filtered in a much greater degree. This indicates that about 90% of the correct senses are ranked within top 3 at all datasets based on the above similarity ranking, which might contribute to the semantic path exploration by filtering a large number of noisy senses. Fig. 5 is a demonstration of the top 3 performances and filtered noise proportion over different datasets.

3.3. Semantic path exploration

Semantic path is common in a few tasks of NLP. For POS tagging task, the major objective is to explore the optimal POS path in a particular sentence. Semantic role labeling task aims at finding the optimal role sequence for a sentence. These tasks are commonly dealt with sequence models such as conditional random fields (CRF) in the past or recurrent neural networks in recent years. With a large pool of labeled corpus, a deep learning sequence model such as a Bi-LSTM with a CRF layer performs well and can achieve much better results than traditional models or a

Algorithm 1 Synset_retrieval()

Input: o_synset, relation_set = {'hyponyms', 'part_holonyms', 'part_meronyms', 'member_holonyms', 'member_meronyms', 'entailments', 'attributes'}
Output: extended_gloss

```
o_synset ← the starting synset of synset retrieval process
synset_list, hypernyms, hypernym_list, extended_gloss ← [], [], [], []
for relation in relation_set do
    new_synset ← synsets extracted by relation of o_synset
    synset_list ← synset_list + new_synset
End for
hypernym_list ← synset_list
while hypernym_list not empty do
    hypernyms ← []
    for synset in hypernym_list do
        hypernyms ← hypernyms + hypernyms of synset
        synset_list ← synset_list + hypernyms
    End for
    hypernym_list ← hypernyms
End while
for synset in synset_list do
    synset_list ← synset_list + derivationally related synsets of synset
End for
for synset in synset_list do
    extended_gloss ← extended_gloss + synset definition and examples
End for
```

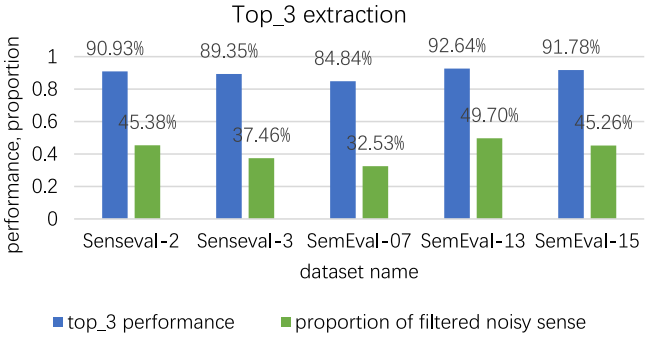

Fig. 5. Top 3 sense performance and proportion of filtered noise sense.

simple deep learning model such as LSTM or Bi-LSTM [50]. There are mainly two features of these tasks. First, the output of the model is a generalization of the input, whether from a word to a POS tag or from a word to a role label. In other words, the output pool is significantly smaller than the input vocabulary. Second, the final tag of a word in the sentence is dependent on most of other words, forming a sequence output as a whole.

Compared with the above tasks, however, WSD is a deeper semantic task, whose output space is much larger than the input space. In most cases, the sense of a given word is dependent on a few senses of the words in the sentence. This indicates the length of the semantic path might not be the same as that of the sentence. Also, the contribution of other senses to disambiguate a potential sense can vary notably. For example, in the following two sentences, the word 'bank' has different meanings. The sense of the first 'bank' mainly relies on senses from 'went', 'deposit' and 'money' while the latter requires the senses from 'went' and 'water' for disambiguation.

- Jon went to the bank to deposit some money yesterday.
- Lee went to the bank to fetch some water for the wounded horse.

The importance of each contributed word sense is different in the disambiguation process. Besides, connections between senses throughout the sentence are more similar to those in a graph rather than a sequence. In a word, the semantic path of WSD is better modeled in a network.

PageRank is a graph-based algorithm capable of determining the importance of graph nodes according to their connections to each other. The voting scheme of the algorithm is rather straightforward. A node's importance depends mainly on the number of nodes it connects to. More significantly, those nodes it connects to will raise the node's importance prominently if they are important in the network. This has shown the iterative process of the algorithm. Given a graph with K nodes, the general formula of PageRank is as follows:

$$\overrightarrow{Pr} = c\,M\overrightarrow{Pr} + (1-c)\,\vec{v} \tag{2}$$

where:

$\overrightarrow{Pr}$, K × 1 target rank vector, representing the probability of reaching each node by random walk, the final stable outcome can determine each sense's importance in WSD,

c, damping factor, a scalar ranging from 0 to 1, normally between 0.85 and 0.95 in WSD scenario,

M, K × K transition probability matrix, discussed in the following algorithm,

$\vec{v}$, K × 1 vector whose element reflects empirical importance of each node.

In our case, the graph is consisted of potential senses/synsets in the sentence being disambiguated, all the related synsets by different relations in WordNet. The connections of these graph nodes stem from two places. The connections between all the potential senses are modeled with their extended gloss vector 'similarity' (dot product). If the value is positive, a link is added between two senses. This process is illustrated in Fig. 6 from a

general view. The reason why we model potential sense connections with this 'similarity' is that most of them are not connected in WordNet since they are cross-POS connections. This type of connection is sparse in WordNet because it focuses on modeling paradigmatic relations such as hypernyms (is-a) and meronyms (part-of), paying little attention to syntagmatic relations such as the connection of 'eat' and 'food', 'bus' and 'drive', etc. It is worth mentioning that there is no connection between competing senses regardless of the similarity value because they might reinforce each other [16] during iteration and weaken other nodes' importance to a great extent.

In order to incorporate more background knowledge and expand this original graph, we employ the corresponding extended glosses (synsets) retrieved above. eXtended WordNet[2] (XWN) [51] is exploited to change all the words in definitions and examples into disambiguated senses. eXtended WordNet is a morphologically and semantically enhanced version of WordNet. It has disambiguated all the words in the definition and examples of the synsets. To solve the WordNet version compatibility issue, we use a sense mapping file from WordNet standoff files.[3] The link establishment of these nodes is more complicated since the synset relations are extracted from WordNet. Basically, connection exists if there is a link between two synsets in WordNet. Naturally, there are also links between a synset and the synsets from its gloss and examples. In addition, for those synsets from glosses and examples, they will be connected to each other if the synsets they belong to are connected in WordNet. This is aimed at enriching the connections throughout the constructed network, leading to better results [39]. All the connections mentioned above are either existed in WordNet or established by using eXtended WordNet.

The detailed disambiguation process is that, for each word w, we construct a graph $G_{w,C(w)}$ that contains the nodes (the related synsets) mentioned above and initialize $\vec{v}$ in Formula (2) with the empirical importance of corresponding nodes. Here, c(w) is the context of word w, which contains a group of words in a predefined sliding window. We use this graph to perform PageRank to calculate the importance of each sense (a node in the graph) corresponding to an ambiguous word. This implementation is similar to the process discussed in [16].

In order to avoid unnecessary calculation, isolated nodes are excluded from $G_{w,C(w)}$. Suppose the adjacency matrix of graph $G_{w,C(w)}$ is $X = (x_{ij})$, where x_{ij} (the element in ith row and jth column) is 1 if node i, j are linked, 0 otherwise. Based on the matrix X, a transition probability matrix $M = (m_{ij})$ is built by normalizing each column of X. The calculation of the normalization is shown in formula (3), where K is the number of nodes in $G_{w,C(w)}$.

$$m_{ij} = \frac{x_{ij}}{\sum_{k=1}^{K} x_{kj}}, i, j = 1, 2, \ldots, K \tag{3}$$

In Formula (2), the initial value of the vector $\vec{v}$ is critical, because a customized scheme can be adopted to replace the uniform distribution, which transforms the standard PageRank algorithm into a personalized one and improves the performance [15]. We initialize the vector $\vec{v}$ in the following way. Assume that a word w has p potential senses, $s_1, \ldots, s_p$, and the similarity between the context of the word w and its potential senses s_i $(k = 1, \ldots, p)$, can be represented as the vector $\overrightarrow{Sim}_w = (Sim_{w,s_1}, \ldots, Sim_{w,s_p})$ (here, the calculation for Sim_{w,s_i} is defined in formula (1) in Section 3.2). Then we can further normalize $\overrightarrow{Sim}_w$ to get the 'normalized similarity' between the context of the word w and its potential senses s_i as follows,

$$NSim_{w,s_i} = \frac{Sim_{w,s_i}}{\sum_{j=1}^{p} Sim_{w,s_j}}, i, j = 1, 2, \ldots, p \tag{4}$$

Then we initialize the component of $\vec{v}$ for the potential (competing) sense s_i with the value $NSim_{w,s_i}$. Besides, we initialize the components of $\vec{v}$ for other related (derived) synsets to 1/K. For words that appear in context more than once, we initialize their corresponding senses with the similarity from the word that is closer to the current word under disambiguation. As for the damping factor in Formula (2), we assign a common value of 0.85 used by previous researches. The initial value of Pr is a uniform distribution. Algorithm 2 shows the detailed procedure of the personalized PageRank algorithm.

Algorithm 2 Personalized PageRank()

Input: X, M, v, Pr, c, iteration, delta
Output: Pr

```
M=(m_ij) ← K×K transition matrix, column normalization of the graph connection
matrix X=(x_ij) without isolated nodes, x_ij is 1 if node i, j are linked, 0 otherwise
v⃗ ← K×1 node importance, v_i equals to similarity for potential senses, 1/K for others
P⃗r ← K×1 target rank, 1/K for all elements
c ← 0.85 damping factor
iteration ← 100
delta ← 0.0001
while iteration > 0 do
    Pr⃗′ = c * M * P⃗r + (1 − c) * v⃗
    iteration ← iteration - 1
    If sum(|Pr⃗′ - P⃗r|) <= delta Then
        Iteration = 0
    End If
    P⃗r = Pr⃗′
End while
```

4. Experiment settings

4.1. Datasets

There are several datasets for WSD system evaluation, with the first standard one tracing back to 2001. Their format varies because of specific conditions of the year when they were presented, including a combination with another task such as entity linking [26]. Due to format discrepancy, preprocessing of the datasets might have an effect on the performance comparison of different systems. Besides, the latest two datasets are annotated with three different sense inventories, namely WordNet, BabelNet and Wikipedia. This might also lead to comparison bias of different systems' performance. Therefore, an evaluation framework [7] is proposed to guarantee a fair comparison of different systems. The framework is composed of several WSD datasets [22–26] and is used to evaluate several state-of-the-art WSD systems in both supervised and knowledge-based category. All these datasets but Semeval-13 (only disambiguates nouns) are different from the official datasets since they do not require systems to disambiguate auxiliary verbs. In other words, the disambiguation word pool is changed. During the following experiments, we will utilize the datasets provided by the above framework and compare the results with some state-of-the-art systems from both dataset and POS perspective with identical settings.

4.2. Systems under comparison

IMS [33]: a supervised system that employs SVM as classifiers with features such as surrounding words, POS tags of surrounding

[2] http://www.hlt.utdallas.edu/~xwn/
[3] https://wordnet.princeton.edu/download/standoff-files

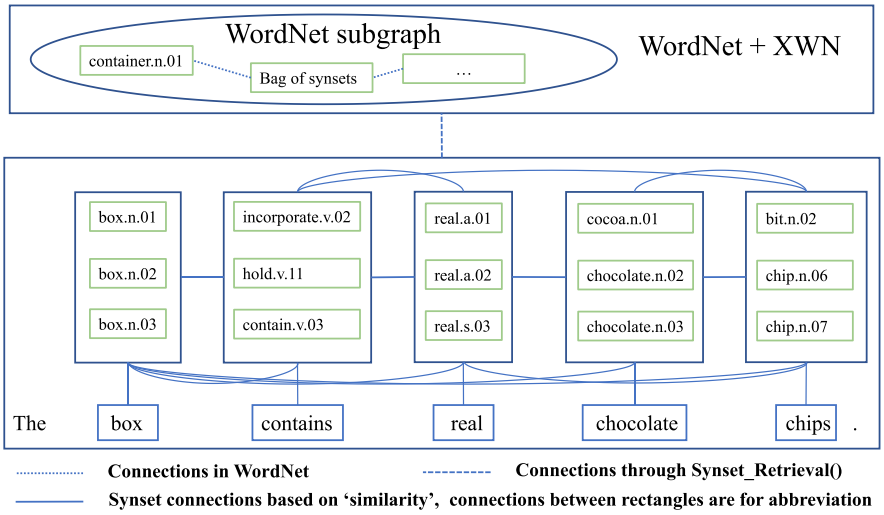

Fig. 6. Subgraph construction for semantic path exploration.

words, and local collocations. IMS+emb is another system that incorporates word embeddings as a new feature for disambiguation. IMS$_{-s}$+emb is a similar system as IMS+emb but it excludes the feature of surrounding words. For the latter two systems, word embeddings are utilized with a weight decay strategy. In detail, word weights decrease exponentially as their distances to the target word increase.

Context2Vec [35]: a system which uses Bi-LSTM to learn context embeddings for disambiguation as one of its many applications.

BLSTM [21]: different from the above two systems, it employs a single Bi-LSTM model rather than a series of word experts to disambiguate words.

Lesk$_{ext}$: an adapted version of Lesk [41], which has added relation employment to the system. The final choice of senses is dependent on TF–IDF similarity between the extended gloss and the context. Lesk$_{ext}$+emb [12] is an enhanced version of adapted Lesk. It incorporates latent semantic space into the system by learning semantic vector for words using LSA. Thus, it replaces the TF–IDF similarity with semantic vector similarity to decide which sense to pick.

UKB [15,16]: a graph-based system which performs personalized PageRank on WordNet with the context of the target word. UKB_gloss is an extended version of UKB by using eXtended WordNet to transform words in glosses into disambiguated synsets. This can enrich the connections within the network. UKB-g* is the latest version which uses sense distribution from SemCor and takes a context window of a minimum of 20 words for personalized PageRank of each target word.

Babelfy [42]: a graph-based system which performs random walks over a large semantic network integrating WordNet and other resources such as Wikipedia and Wiktionary. The best configuration considers the whole document as the starting nodes for personalized PageRank.

WSD-TM [19]: a graph-based WSD system which employs a variant of LDA to disambiguate a document with the whole document as context.

UMCC-DLSI [18]: a top-performing system in semeval-13 [25] dataset. The system performs personalized PageRank on an extended knowledge base that integrates several WordNet related resources such as eXtended WordNet, WordNet domain, etc. Also, it initializes the sense importance vector with sense frequency, which is shown to have improved the performance.

Nasarilexical [13]: a system that learns semantic representation for senses from explicit knowledge and corpus statistics in a multilingual setting. The sense representations can be applied to many language understanding tasks including word sense disambiguation.

Baselines: for supervised system, the most frequent sense heuristic (MFS) is presented, which selects the most frequent sense in the training corpus as the predicted sense for any target word. Also, a ceiling performance shows the proportion of annotated sense in test corpus that appears in the training corpus. In addition, WordNet 1st (WN 1st) sense is selected as a baseline for knowledge-based systems. Simply, for each target word, its first sense in WordNet is selected as the predicted sense.

NOTE: Throughout the following experiments, we have compared our results with those of several supervised and knowledge-based systems using the datasets provided by [7]. Most of the compared systems' results were reproduced by [7] with the datasets in its proposed evaluation framework, except BLSTM, WSD-TM, UMCC-DLSI and Nasarilexical. For BLSTM, WSD-TM, the authors of [7] used the same framework to evaluate them so we directly report their performance in this paper. For the latter two systems, UMCC-DLSI and Nasarilexical reported their performance on the official WSD datasets but not on the datasets in the framework by [7]. Since only the official SemEval-13 dataset is identical to the corresponding dataset in the framework, we include the performance of UMCC-DLSI and Nasarilexical on Semeval-2013 for a fair comparison.

4.3. Parameter settings

Next, we will introduce some parameter settings throughout our knowledge exploitation framework.[4] For domain knowledge, Wikipedia documents are selected to be a document source. Naïve IR is conducted against them to retrieve domain specific documents. The representation vector dimension for each word is chosen to be 200, high enough to incorporate sufficient topic factors. WSD documents are grouped by sentences during disambiguation since we will explore the semantic path within a sentence afterwards. For each ambiguous word, a context of 7 sentences is chosen, with 3 sentences on the left of the sentence where the ambiguous word locates and 3 on the right. Experiments have shown that using the whole document as context performs worse than using a window of sentences. In addition, window size of 3 is selected since increasing the number does

[4] https://github.com/lwmlyy/Knowledge-based-WSD

not improve the performance and a small window size results in loss of necessary information for disambiguation. The similarity Sim_{w,s_i} between the context of word w and the extended gloss of the ith sense of word w is obtained by calculating the dot product rather than the cosine similarity of the according vectors (the formula for Sim_{w,s_i} is in Section 3.2). This can enlarge the similarity value and enhance its effect. Before selecting the top 3 senses, sense frequency of each word from WordNet is utilized to weight the similarity. After attaining sense importance Pr_{w,s_i} (the importance of ith sense of word w, obtained from Algorithm 2) within the sentence network by personalized PageRank, we use a simple weighting scheme to combine it with the similarity Sim_{w,s_i} to determine which sense to be the final predicted sense. The final score of ith sense of word w under the context is calculated in formula (4), where the weight α for sense importance is fixed to be 0.1 since semantic path can only distinguish those highly related senses.

$$Score_{w,s_i} = \alpha \times Pr_{w,s_i} + (1-\alpha) \times Sim_{w,s_i} \quad (5)$$

5. Result analysis

5.1. Improvement factors illustration and error analysis

To illustrate the contribution of each factor proposed by our research, we have experimented with the latest WSD dataset (Semeval-15) under different settings. Table 2 is a demonstration of different experimental results. For each performance factor (each column), a fair comparison can be conducted on those two rows (experiments) with shaded cells in the same type of shadow. For example, a comparison of experiment 4 and 6 can reflect the influence of the TF–IDF factor. An overall conclusion can be drawn that the framework is relatively robust in terms of surpassing the WordNet 1st sense approach in the latest WSD dataset. Except for three aforementioned factors, a TF–IDF factor is included since one previous research [12] has shown its contribution under a varied WordNet relation extraction scheme. Basically, it treats each extended gloss as a document and calculates a TF–IDF score for all the words in the extended glosses of an ambiguous word, weighting each word vector similar to the distance weight calculated in Section 3.2. However, the performance decreases when we add this factor to the best scheme. This is probably due to a significant size change of the extended gloss.

As for the other factors, customized relation exploitation has achieved the greatest performance boost, 1.36%. This is a rather valid certification for its contribution. IR of domain knowledge documents has enhanced the performance by 0.97%, which shows the potential value of adding domain knowledge for disambiguation. Although semantic path exploration has not raised the performance as much as the above two factors, 0.2%, this factor is proven to be effective since the performance on all five datasets has witnessed a rise. Incidentally, a slightly larger margin, 0.4%, is achieved by semantic path exploration when TF–IDF is employed.

Fig. 7 is our system's performance on SemEval-15. It shows its high intention to regard WordNet 1st sense as the correct sense. This is due to an employment of the WordNet sense frequency to weight the calculated score for each sense while it is proven to be beneficial for the disambiguation process as a whole. Especially in the wrong prediction set, about 78% of words are falsely predicted to be their WordNet 1st sense. This indicates a customized utilization of sense frequency might contribute to the disambiguation process. Also, the average ambiguity (the number of senses that a word has) of the wrong prediction set (8.67) is much higher than that of the correct set (4.25). This is logical since words with more senses are harder to be disambiguated.

In a more specific perspective, some words in the wrong prediction set are semantically difficult to disambiguate. Often,

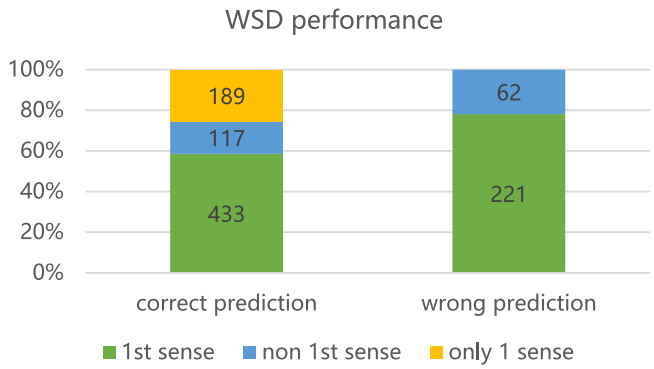

Fig. 7. WSD performance on SemEval-15.

some senses of a word are similar to each other if they are viewed from a statistical or even a shallow semantic angle. For example, the following sentences show the correct sense and the predicted sense of 'assessment' and 'prescription' in SemEval-15. There is a high overlap between the correct sense and the prediction. These words might be disambiguated by considering fine-grained semantics.

- Correct sense – prescription – 'written instructions from a physician or dentist to a druggist concerning the form and dosage of a drug to be issued to a given patient'
- Predicted sense – prescription – 'a drug that is available only with written instructions from a doctor or dentist to a pharmacist'
- Correct sense – assessment – 'the act of judging or assessing a person or situation or event'
- Predicted sense – assessment – 'the classification of someone or something with respect to its worth'

5.2. Performance on all datasets

An overall assessment of the proposed WSD framework on all datasets can further validate its robustness in disambiguation. Table 3 is an illustration of different systems' performance on five standard WSD datasets (Senseval-2 [22], Senseval-3 [23], SemEval-07 [24], SemEval-13 and SemEval-15), including both supervised and knowledge-based systems. The results show that the proposed system has achieved state-of-the-art performance on three datasets, with performance on the other two datasets approaching the best ones. On the combined dataset, the F-measure, 68.0, has surpassed all knowledge-based systems, 0.7 higher than the latest system. This margin almost equals to that between a deep sequence learning model (BLSTM) and an SVM with traditional features and word embeddings (IMS+emb).

The best performance is achieved on the latest two datasets (SemEval-13, SemEval-15), with the best knowledge-based systems outperforming most supervised systems. According to the ambiguity statistic provided in [7], these two datasets are relatively less ambiguous. Specifically, the number of all potential senses divided by the number of sense annotations is smaller than that of the other datasets.

From another perspective, ambiguity relates to the proportion of different POS throughout a dataset, especially the proportion of nouns and verbs, because of different ambiguity level of POS. In detail, SemEval-13 only disambiguates nouns, leading to a rather low ambiguity level. In this dataset, the best knowledge-based system, our system as well, can outperform the supervised ones with a relatively large margin. This has shown the tremendous potential of knowledge exploitation approaches for noun disambiguation. In the latest dataset, our system has outperformed all

Table 2
Performance factor illustration.

Experiment \ Factor	WN 1st	TF-IDF	IR	Customized Relation Exploitation	Semantic Path (PageRank)	F1
1	√					0.678
2			√			0.707
3			√	√		0.721
4			√	√	√	**0.723**
5			√	√	√	0.713
6		√	√	√	√	0.714
7		√	√	√		0.71

other state-of-the-art knowledge-based systems and most supervised systems, with a large margin (2 F-measure) over Babelfy and only 0.1 F-measure lower than the best deep sequence learning model. The dataset was originally proposed to jointly tackle WSD and entity linking although either task can be dealt with separately. This indicates a large pool of background information of those entities can be employed for disambiguation. In Babelfy, two tasks are jointly resolved by making use of a large integrated knowledge graph that incorporates a pool of other knowledge resources. In our system, it is achieved mainly by IR of domain knowledge documents and WordNet relation exploitation. This has contributed to narrow the performance margin between our system and the best supervised system to some extent. Additionally, a better performance than Babelfy indicates a customized exploitation of external knowledge is essential.

For those more ambiguous datasets, our system achieves the best result on the smaller one (SemEval-07) and reach near to the simple WN 1st sense scheme on the other (Senseval-3). Their high ambiguity mainly comes from a relatively larger proportion of verb, which is rather difficult to disambiguate. Especially in SemEval-07, there are more verbs than nouns to be disambiguated because of a large proportion of nouns, all adjectives and all adverbs are not annotated. This is rather different from the other datasets. Comparing the performance of supervised and knowledge-based systems, the biggest gap, 6.8, is obtained on the most ambiguous SemEval-07, with that of Senseval-3 being the second largest, 3.5. Our system has achieved the best performance on SemEval-07, attaining a gap of at least 1.3 F-measure between other well performing systems (UKB-g*, WSD-TM) regardless of the ambiguity. In addition, it has obtained the best result on Senseval-2 with a margin of 0.6 F-measure between the latest knowledge-based system. These results have illustrated a relatively stable performance of our system.

5.3. Performance on all POS

As regards the performance on POS, the proposed system has attained the best performance on nouns and verbs compared with the results of other state-of-the-art knowledge-based systems. Words in these two POS are the major components of most documents, which indicates a higher disambiguation capability of them usually leads to a more satisfactory performance on the whole. In particular, effectiveness of disambiguating nouns is much greater than the best system in knowledge-based category, obtaining a margin of 2.2 F-measure. This margin is significant since nouns are the key constituents in most documents or datasets, occupying about 59% of all POS in the combined dataset. Additionally, our noun disambiguation performance is equal to that of the best supervised system, showing our system's huge potential by knowledge exploitation.

Although the margin of disambiguating verbs is relatively small, it has also shown the effectiveness of our proposed framework. The reason is that WordNet relation exploiting scheme is more focused on hypernyms. This type of relation is only available for nouns and verbs. Another reason why performance on nouns can be significantly better than the past systems is that those domain knowledge documents are from Wikipedia. This knowledge resource is organized mainly based on nouns or entities and their relations. These results have thus illustrated the contribution of our proposed knowledge exploitation framework since they originate from the two factors we propose. In contrast, F-measure on adjectives is much lower than the best previous system, a margin of 2. This might be due to a relatively sparse relation of adjectives within WordNet. Also, the domain knowledge we use cannot provide much help in disambiguating adjectives or adverbs since the documents are from Wikipedia. The detailed information of performance over POS of different systems is shown in Table 4.

5.4. Transferability to entity linking

Similar to WSD, entity linking is another disambiguation task that focuses on disambiguating entities in unstructured data so as to enrich or update entities' information within a knowledge graph while it is in a coarse grain and sometimes referred to as name entity disambiguation [52]. In a more specific context, this task can dive into disambiguating certain entity types such as person name [53]. The commonness of both tasks indicates that solutions to one task can be conveniently transferred to deal with the other task. Furthermore, a joint scheme of coping with both tasks simultaneously might be beneficial to improve the performance on both tasks.

Although the fundamental concept of WSD and entity linking is relatively similar, a systematic transferring of systems from one to the other should be customized in different settings. Specifically, it is convenient to employ our knowledge exploiting framework to deal with entity linking issue on one hand; the performance on such task can be below average if certain details are not fully considered, on the other hand. For example, an entity identification process is required to determine all potential entities for disambiguation. In order to retrieve domain knowledge, entities in the documents under disambiguation should be utilized to retrieve corresponding Wikipedia documents. This is an expansion of the original domain knowledge retrieval. In addition, the knowledge base in our framework should be changed into other entity inventories (e.g. BabelNet) that includes sufficient entities for disambiguation since WordNet does not contain much entity-related information. In terms of the semantic path learning, entities and words in the sentence should be allowed to co-occur for a subgraph construction so that information is exploited to a greater extent.

Table 3
Performance over datasets.

	System	Senseval-2	Senseval-3	SemEval-07	SemEval-13	SemEval-15	ALL
Supervised	IMS	70.9	69.3	61.3	65.3	69.5	68.4
	IMS+emb	71.0	69.3	60.9	**67.3**	71.3	69.1
	IMS-s+emb	**72.2**	**70.4**	62.6	65.9	71.5	69.6
	BLSTM	72.0	69.4	**63.7**	66.4	**72.4**	**69.9**
	Context2Vec	71.8	69.1	61.3	65.6	71.9	69.0
	MFS	65.6	66.0	54.5	63.8	67.1	64.8
	Ceiling	91.0	94.5	93.8	88.6	90.4	91.0
Knowledge-based	$Lesk_{ext}$	50.6	44.5	32.0	53.6	51.0	48.7
	$Lesk_{ext}$+emb	63.0	63.7	56.7	66.2	64.6	63.7
	UKB-g*	68.8	66.1	53.0	**68.8**	70.3	67.3
	Babelfy	67.0	63.5	51.6	66.4	70.3	65.5
	UMCC-DLSI	–	–	–	64.7	–	–
	Nasarilexical	–	–	–	66.7	–	–
	WSD-TM	69.0	**66.9**	55.6	65.3	69.6	66.9
	WN 1st sense	66.8	66.2	55.2	63.0	67.8	65.2
	Ours	**69.6**	66.1	**56.9**	68.4	**72.3**	**68.0**

Nevertheless, certain aspects might influence entity linking performance negatively. For instance, in the phase of retrieving domain knowledge document, the sparsity of entities in documents can lead to their poor representation learning, which might damage the effectiveness of semantic capturing.

6. Conclusion and future work

In this paper, we present a comprehensive knowledge-based WSD framework. Our research regards WSD and its applications as interactive tasks in a mutually reinforcing relationship. Thus, WSD is performed with the assistance of information retrieval and knowledge graph, so as to achieve better knowledge exploitation. Experiment results on five standard WSD datasets have shown that the proposed approach outperforms all other knowledge-based systems on three datasets. From the perspective of POS disambiguation, the proposed approach performs excessively well on nouns and verbs, which are the major components of most documents. Its performance on noun disambiguation is even comparable to that of the best supervised system and its performance on verb disambiguation is better than that of all other knowledge-based systems.

The major focus of this research is to imitate the way human disambiguates words using a large pool of latent semantic factors and connections between senses. In order to incorporate latent semantic factors, LSA is employed to capture semantic space in a large corpus. Furthermore, some domain-specific documents are added to the large corpus so that those captured semantic factors are more coherent with the documents being disambiguated. These documents are acquired by a naïve information retrieval strategy with TF–IDF scores between Wikipedia documents and queries from documents under disambiguation or a simple employment of WordNet. As for semantic connections, different synset relation extraction schemes are explored over WordNet. After systematic analysis, a customized relation exploitation strategy is proposed to make full use of abundant knowledge carried by sense connections. To supplement synset connections on WordNet and capture semantic path within sentences under disambiguation, synset connections within a sentence are modeled by the vector similarity between extended glosses of synsets on top of the synset connections from WordNet. Then, a personalized PageRank is employed to determine sense importance while considering the semantic connections within the sentence-level graph. As demonstrated in factor analysis on the latest WSD dataset, all above factors are shown to have contributed to the final performance of the proposed approach.

Table 4
Performance over POS.

	System	Nouns	Verbs	Adjectives	Adverbs	All
Supervised	IMS	70.4	56.1	75.6	82.9	68.4
	IMS+emb	71.8	55.4	**76.1**	82.7	69.1
	IMS-s+emb	**71.9**	56.9	75.9	**84.7**	69.6
	Context2Vec	71.0	**57.6**	75.2	82.7	69.0
	BLSTM	71.6	57.1	75.6	83.2	**69.9**
	MFS	67.6	49.6	73.1	80.5	64.8
	Ceiling	89.6	95.1	91.5	96.4	91.5
Knowledge-based	$Lesk_{ext}$	54.1	27.9	54.6	60.3	48.7
	$Lesk_{ext}$+emb	69.8	51.2	51.7	80.6	63.7
	UKB	56.7	39.3	63.9	44.0	53.2
	UKB_gloss	62.1	38.3	66.8	66.2	57.5
	Babelfy	68.6	49.9	73.2	79.8	65.5
	WSD-TM	69.7	51.2	**76.0**	**80.9**	66.9
	WN 1st sense	67.6	50.3	74.3	80.9	65.2
	Ours	**71.9**	**51.6**	74.0	80.6	**68.0**

Despite its effectiveness, our framework has some points to be improved. First, the retrieval process of domain specific documents is time-consuming, taking much longer time than the WSD process itself. Second, although the semantic path exploration has achieved performance boost in all five datasets, the margin is not as large as the other two factors. This has shown the space for improvement.

For future work, we intend to investigate the multilingual property of our framework. Since all the resources employed in our framework are multilingual, it indicates a rather convenient adaptation to other languages. More importantly, there is an increasing demand for incorporating more abundant relations between senses. The current relation extraction source, WordNet, focuses mainly on paradigmatic relations rather than syntagmatic and others, which has limited the WSD performance to some extent. One possible direction is to employ a more comprehensive knowledge graph that combines various resources together, such as BabelNet, for the purpose of relation extraction after proper selection. In addition, in terms of time efficiency of the system, it might be better if different domain knowledge documents are attained in advance, which could eliminate the time for document retrieval while guarantee a satisfactory performance. This can be beneficial to upgrade the system into a real-time one. Finally, we will explore the transferability of our framework to a coarser grain of word sense disambiguation, entity linking or entity disambiguation, which is essential for knowledge graph construction and updating.

Acknowledgments

The authors would like to express thanks to the National Natural Science Foundation of China (under Project No. 61375053) and the graduate innovation fund of Shanghai University of Finance and Economics (under Project No. CXJJ-2019-395) for their financial supports.

References

[1] R. Navigli, Word sense disambiguation: A survey, ACM Comput. Surv. 41 (2) (2009) http://dx.doi.org/10.1145/1459352.1459355, Article 10.
[2] C. Hung, S.J. Chen, Word sense disambiguation based sentiment lexicons for sentiment classification, Knowl.-Based Syst. 110 (2016) 224–232, http://dx.doi.org/10.1016/j.knosys.2016.07.030.
[3] Z. Zhong, H. Ng, Word sense disambiguation improves information retrieval, in: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, ACL '12, 2012, pp. 273–282.
[4] C.D. Bovi, L. Telesca, R. Navigli, Large-scale information extraction from textual definitions through deep syntactic and semantic analysis, Trans. Assoc. Comput. Linguist. 3 (2015) 529–543.
[5] D. Xiong, M. Zhang, A sense-based translation model for statistical machine translation, in: Proceedings of the 52th Annual Meeting of the Association for Computational Linguistics, ACL '14, 2014, pp. 1459–1469. http://dx.doi.org/10.1016/j.euroneuro.2010.01.001.
[6] A. Raganato, C.D. Bovi, R. Navigli, Automatic construction and evaluation of a large semantically enriched wikipedia, in: Proceedings of the 25th International Joint Conference on Artificial Intelligence, IJCAI '16, 2016, pp. 2894–2900.
[7] A. Raganato, J. Camacho-Collados, R. Navigli, Word sense disambiguation: a unified evaluation framework and empirical comparison, in: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL '17, 2017, pp. 99–110.
[8] G.A. Miller, WordNet: a lexical database for English, Commun. ACM 41 (2) (1995) 39–41, http://dx.doi.org/10.1145/219717.219748.
[9] R. Navigli, S.P. Ponzetto, BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network, Artificial Intelligence 41 (2012) 217–250, http://dx.doi.org/10.1016/j.artin.
[10] S. Brin, L. Page, The anatomy of a large-scale hypertextual Web search engine, Comput. Netw. ISDN Syst. 30 (1998) 107–117.
[11] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent Dirichlet allocation david, J. Mach. Learn. Res. 3 (2003) 993–1022, http://dx.doi.org/10.1162/jmlr.2003.3.4-5.993.
[12] P. Basile, A. Caputo, G. Semeraro, An enhanced lesk word sense disambiguation algorithm through a distributional semantic model, in: Proceedings of the 25th International Conference on Computational Linguistics: Technical Papers, COLING'14, 2014, pp. 1591–1600. http://dx.doi.org/10.1024/1012-5302/a000007.
[13] J. Camacho-Collados, M.T. Pilehvar, R. Navigli, NASARI: Integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities, Artificial Intelligence 240 (2016) 36–64, http://dx.doi.org/10.1016/j.artint.2016.07.0.
[14] T.K. Landauer, S.T. Dumais, A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge, Psychol. Rev. 1M (2) (1997) 211–240, http://dx.doi.org/10.1037/0033-295X.104.2.211.
[15] E. Agirre, A. Soroa, Personalizing PageRank for word sense disambiguation, in: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, EACL '09, 2009. http://dx.doi.org/10.3115/1609067.1609070.
[16] E. Agirre, O.L. de Lacalle, A. Soroa, Random walks for knowledge-based word sense disambiguation, Comput. Linguist. 40 (1) (2014) 57–84, http://dx.doi.org/10.1162/COLI_a_00164.
[17] I. Iacobacci, M.T. Pilehvar, R. Navigli, Embeddings for word sense disambiguation: an evaluation study, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL '16, 2016 pp. 897–907. http://dx.doi.org/10.18653/v1/P16-1085.
[18] Y. Gutiérrez, S. Vázquez, A. Montoyo, Spreading semantic information by word sense disambiguation, Knowl.-Based Syst. 132 (2017) 47–61, http://dx.doi.org/10.1016/j.knosys.2017.06.013.
[19] D. Chaplot, R. Salakhutdinov, Knowledge-based word sense disambiguation using topic models, in: Proceedings of AAAI Conference on Artificial Intelligence, AAAI '18, 2018, pp. 5062–5069.
[20] A.M. Butnaru, R.T. Ionescu, F. Hristea, ShotgunWSD: An unsupervised algorithm for global word sense disambiguation inspired by DNA sequencing, in: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, EACL '17, 2017, pp. 916–926.
[21] A. Raganato, C. Delli Bovi, R. Navigli, Neural sequence learning models for word sense disambiguation, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP '17, 2017, pp. 1156–1167. http://dx.doi.org/10.1083/jcb.201007098.
[22] P. Edmonds, S. Cotton, SENSEVAL-2: overview, in: Proceedings of the 2nd International Workshop on Evaluating Word Sense Disambiguation Systems, SENSEVAL-2, 2001. http://dx.doi.org/10.1080/21565503.2016.1160413.
[23] B. Snyder, M. Palmer, The english all-words task, in: Proceedings of the 3rd International Workshop on the Evaluation of Systems for the Semantic Analysis of Text, SENSEVAL-3, 2004.
[24] S. Pradhan, E. Loper, D. Dligach, M. Palmer, SemEval-2007 Task 17: English lexical sample, SRL and all words, in: Proceedings of the Fourth International Workshop on Semantic Evaluations, SemEval '07, 2007, pp. 87–92.
[25] R. Navigli, D. Jurgens, D. Vannella, SemEval-2013 Task 12: Multilingual word sense disambiguation, in: Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval-2013), in conjunction with the Second Joint Conference on Lexical and Computational Semantics, SemEval/SEM '13, 2013, pp. 222–231. http://dx.doi.org/10.1016/S0044-328X(82)80082-2.
[26] Moro R. Navigli, SemEval-2015 Task 13: Multilingual all-words sense disambiguation and entity linking, in: Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval '15, 2015, pp. 288–297. http://dx.doi.org/10.18653/v1/S15-2049.
[27] R.J. Mooney, Comparative experiments on disambiguating word senses: An illustration of the role of bias in machine learning, in: Proceedings of the 1996 Conference on Empirical Methods in Natural Language Processing, EMNLP '96, 1996, pp. 82–91.
[28] G. Escudero, L. Marquez, G. Rigau, On the portability and tuning of supervised word sense disambiguation, in: Proceedings of the joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, EMNLP/VLC '00, 2000, pp. 172-180.
[29] G. Tsatsaronis, M. Vazirgiannis, I. Androutsopoulos, Word sense disambiguation with spreading activation networks generated from thesauri, in: Proceedings of the 17th International Joint Conference on Artificial Intelligence, IJCAI '07, 2007, pp. 1725–1730. http://dx.doi.org/10.1145/1459352.1459355.
[30] Y.K. Lee, H.T. Ng, An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation, in: Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing, EMNLP '02, 2002, pp. 41–48. http://dx.doi.org/10.3115/1118693.1118.
[31] R.E. Schapire, Y. Singer, Improved boosting algorithms using confidence-rated predictions, Mach. Learn. 37 (1999) 297–336, http://dx.doi.org/10.1023/A:1007614523901.
[32] G. Escudero, L. Marquez, G. Rigau, Boosting applied to word sense disambiguation, in: Proceedings of the 11th International Conference on Machine Learning, ICML '00, 2000, pp. 129–141.
[33] Z. Zhong, H.T. Ng, It makes sense: a wide-coverage word sense disambiguation system for free text, in: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL '10, 2010, pp. 78–83.
[34] S. Papandrea, R. Alessandro, D.B. Claudio, SUPWSD: A flexible toolkit for supervised word sense disambiguation, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP '17, 2017, pp. 103–108.
[35] O. Melamud, J. Goldberger, I. Dagan, context2vec: Learning generic context embedding with bidirectional LSTM, in: Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL '16, 2016, pp. 51–61. http://dx.doi.org/10.18653/v1/K16-1006.
[36] E.A. Corrêa, A.A. Lopes, D.R. Amancio, Word sense disambiguation: A complex network approach, Inform. Sci. 442–443 (2018) 103–113, http://dx.doi.org/10.1016/j.ins.2018.02.047.
[37] T. Wang, J. Rao, Q. Hu, Supervised word sense disambiguation using semantic diffusion kernel, Eng. Appl. Artif. Intell. 27 (2014) 167–174, http://dx.doi.org/10.1016/j.engappai.2013.08.007.
[38] T. Wang, W. Li, F. Liu, J. Hua, Sprinkled semantic diffusion kernel for word sense disambiguation, Eng. Appl. Artif. Intell. 64 (2017) 43–51, http://dx.doi.org/10.1016/j.engappai.2017.05.010.
[39] T. Pasini, R. Navigli, Train-O-Matic: Large-scale supervised word sense disambiguation in multiple languages without manual training data, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP '17, 2017, pp. 78-88.
[40] M. Lesk, Automatic sense disambiguation using machine readable dictionaries, in: Proceedings of the 5th annual international conference on Systems documentation, SIGDOC '86, 1986, pp. 24–26. http://dx.doi.org/10.1145/318723.318728.
[41] S. Banerjee, T. Pedersen, Extended gloss overlaps as a measure of semantic relatedness, in: Proceedings of the 13th International Joint Conference on Artificial Intelligence, IJCAI '03, 2003, pp. 805–810.

[42] A. Moro, A. Raganato, R. Navigli, Entity linking meets word sense disambiguation: A unified approach, Trans. Assoc. Comput. Linguist. 2 (2014) 231–244.
[43] R. Tripodi, M. Pelillo, A game-theoretic approach to word sense disambiguation, Comput. Linguist. 43 (1) (2017) 31–70, http://dx.doi.org/10.1162/COLI_a_00274.
[44] D. Weissenborn, L. Hennig, F. Xu, H. Uszkoreit, Multi-objective optimization for the joint disambiguation of nouns and named entities, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics, ACL '15, 2015, pp. 596-605.
[45] D. Chen, A. Fisch, J. Weston, A. Bordes, Reading wikipedia to answer open-domain questions, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL '17, 2017, pp. 1870–1879. http://dx.doi.org/10.18653/v1/P17-1171.
[46] M. Postma, R. Izquierdo, P. Vossen, VUA-background: When to use background information to perform word sense disambiguation, in: Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval '15, 2015, pp. 345–349. http://dx.doi.org/10.1111/jocd.12147.
[47] H. Toshitaka, F. Hamido, Sentence-level sentiment analysis using feature vectors from word embeddings, in: Proceedings of the New Trends in Intelligent Software Methodologies, Tools and Techniques, SoMeT '18, 2018, pp. 749–758. http://dx.doi.org/10.3233/978-1-61499-900-3-749.
[48] A. Agrawal, W. Fu, T. Menzies, What is wrong with topic modeling? And how to fix it using search-based software engineering, Inf. Softw. Technol. 98 (2018) 74–88, http://dx.doi.org/10.1016/j.infsof.2018.02.005.
[49] P. Resnik, Using information content to evaluate semantic similarity in a taxonomy, in: Proceedings of the 5th International Joint Conference on Artificial Intelligence, IJCAI '95, pp. 448-453.
[50] Y. Wang, M. Wang, Fine-grained opinion extraction from Chinese car reviews with an integrated strategy, J. Shanghai Jiaotong Univ. 23 (3) (2018) 1–7, http://dx.doi.org/10.1007/s12204-018-1961-6.
[51] R. Mihalcea, D.I. Moldovan, extended wordNet: progress report, in: Proceedings of the North American Chapter of the Association for Computational Linguistics Workshop on WordNet and Other Lexical Resources, NAACL '01, 2001, pp. 95–100.
[52] F. Wang, W. Wu, Z. Li, M. Zhou, Named entity disambiguation for questions in community question answering, Knowl.-Based Syst. 126 (2017) 68–77, http://dx.doi.org/10.1016/j.knosys.2017.03.017.
[53] A.D. Delgado, R. Martinez, S. Montalvo, V. Fresno, Person name disambiguation on the web in a multilingual context, Inform. Sci. 465 (2018) 373–387, http://dx.doi.org/10.1016/j.ins.2018.07.024.