# Measuring Semantic Relatedness with Knowledge Association Network

Jiapeng Li[1,2], Wei Chen[1], Binbin Gu[1], Junhua Fang[1], Zhixu Li[1,3], and Lei Zhao[1(✉)]

[1] School of Computer Science and Technology, Soochow University, Suzhou, China
trajepl@gmail.com, wchzhg@gmail.com, gu.binbin@hotmail.com, {jhfang,zhixuli,zhaol}@suda.edu.cn

[2] Institute of Electronic and Information Engineering of UESTC in Guangdong, Dongguan 523808, Guangdong, China

[3] IFLYTEK Research, Suzhou, China

**Abstract.** **Measuring semantic relatedness** between two words is a fundamental task for many applications in both databases and natural language processing domains. Conventional methods mainly utilize the latent semantic information hidden in lexical databases (WordNet) or text corpus (Wikipedia). They have made great achievements based on the distance computation in lexical tree or co-occurrence principle in Wikipedia. However these methods suffer from low coverage and low precision because (1) lexical database contains abundant lexical information but lacks semantic information; (2) in Wikipedia, two related words (e.g. synonyms) may not appear in a window size or a sentence, and unrelated ones may be mentioned together by chance. To compute semantic relatedness more accurately, some other approaches have made great efforts based on free association network and achieved a significant improvement on relatedness measurement. Nevertheless, they need complex preprocessing in Wikipedia. Besides, the fixed score functions they adopt cause the lack of flexibility and expressiveness of model. In this paper, we leverage DBPedia and Wikipedia to construct a **Knowledge Association Network (KAN)** which avoids the information extraction of Wikipedia. We propose a flexible and expressive model to represent entities behind the words, in which attribute and topological structure information of entities are embedded in vector space simultaneously. The experiment results based on standard datasets show the better effectiveness of our model compared to previous models.

**Keywords:** Semantic relatedness · Knowledge graph · Network embedding

## 1 Introduction

Computing semantic relatedness between two words is a fundamental task in many databases and natural language processing problems such as lexicon induction [17], Named Entity Disambiguation [7], Keyword Extraction [27], semantic


G. Li et al. (Eds.): DASFAA 2019, LNCS 11446, pp. 676–691, 2019.
https://doi.org/10.1007/978-3-030-18576-3_40

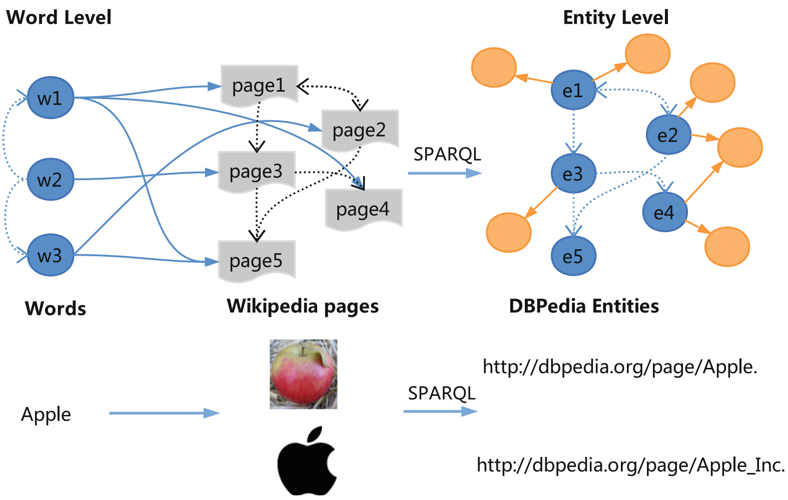


**Fig. 1.** Knowledge association network (Color figure online)

correspondences discovering [3] and Entity Matching [23]. In the aspect of spam problem [19] and image classification [10], semantic relatedness measurement plays a great role as well.

Due to its importance, plenty of efforts have been made on semantic relatedness measurement. The existing approaches can be roughly divided into three categories as below: (i) The *lexical-based* methods [16,25,28] measure the semantic relatedness between two words based on some lexical databases such as *WordNet* and *Wikitionary*. These methods mainly utilize fixed score functions, such as the path information between two words or the nearest parent common node which two words hold in a lexical tree. Apparently, they only employ pure lexical information but miss semantic information. (ii) The *co-occurrence-based* methods regard two words are related if they appear together in a fixed window size or a sentence. So far plenty of efforts [4,20,25] have applied this co-occurrence principle in the dumps of Wikipedia for semantic relatedness measurement. However, the co-occurrence principle does not always work well. Given that two words are semantically closed, such as synonyms, they do not necessarily appear together. Besides, two words that appear in the same sentence by chance may not necessarily be closely related in semantic space [5]. (iii) The *association network-based* methods propose that for a given word, the first word that comes into human mind is the most related one. To improve the co-occurrence-based methods, a more advanced approach builds an association network based on not only co-occurrences between words, but also the links and shared attributes between entities [5,26]. Based on the association network, some heuristic score functions are adopted to compute the semantic relatedness between entities [5,26]. In this way, they make a great improvement in measuring the semantic relatedness. However, the adopted heuristic score functions are not extensible and cause the lack of flexibility. In addition, to get the structured information of entities, they need significant preprocessing and data transformation efforts in Wikipedia.

To overcome the weaknesses of association network-based approaches mentioned above, we propose a *Knowledge Association Network (KAN)* to better capture the semantic features of words and entities, which consists of word-level and entity-level based on Wikipedia[1] and DBPedia[2]. The word-level leverages the co-occurrence relationship between words to capture the semantic features of words, and the entity-level exploits the semantic features of entities behind words to enhance the word-level relatedness measurement. As shown in Fig. 1, initially, for a word *apple*, we look for related Wikipedia pages *Apple* and *Apple_Inc*, where the semantic information of word *apple* could be captured by text analysis based on co-occurrence principle, and the entities *Apple* and *Apple_Inc* are utilized to reinforce the semantic information of *apple*. Then given that each Wikipedia page has a corresponding entity on DBPedia, we could further capture word-to-entity and entity-to-entity linking information on DBPedia for the input words. In the entity-level, attribute and topological structure space are utilized to represent semantic features of an entity. In Fig. 1, each orange node denotes an attribute of an entity, which constitutes the attribute space. And the relationships among entities form the topological space of entities, where the relationships are mapped from the original topological structure of the entity network on DBPedia.

In our model, we use two different strategies to perform the relatedness measurement in word and entity level respectively. At the word-level, word2vec [11] is carried out to compare the semantic information of words. At the entity-level, we firstly propose a novel entity embedding model by simultaneously considering the attribute space and topological structure space of entities. The attribute space captures the semantic information of attributes around an entity by minimizing a margin ranking loss function inspired by translation embedding on knowledge graph. The topological structure space utilizes random walk to generate sampled sequences and adopts Skip-gram model to get the entities embedding. Compared to existing association network-based methods [5,26], our method could avoid the significant preprocessing on the Wikipedia dump given the natural mapping relations between DBPedia entities and Wikipedia pages. Besides, the entity embedding model also works better than the heuristic score functions used in previous models [5,26].

The contributions made in this paper include:

1. We construct a knowledge association network to compute the relatedness of word-to-word, word-to-entity and entity-to-entity based on Wikipedia and DBPedia for better semantic relatedness measurement.
2. We propose a novel entity embedding model by simultaneously considering the attribute and topological structure space of entities, which works better than heuristic score functions.

[1] https://en.wikipedia.org/wiki/Main_Page.
[2] http://dbpedia.org.

3. Our experiments conducted on standard datasets for semantic relatedness measurement show that our approach outperforms several benchmarking methods.

The rest of the paper is organized as follows: We first cover the related work in semantic relatedness measurement in Sect. 5, and then give the definition and construction process of knowledge association network in Sect. 2. After that, in Sect. 3 we elaborate our approach for computing the semantic relatedness based on the KAN. Next we introduce our experiments in Sect. 4, then finally conclude this paper in Sect. 6.

## 2 Knowledge Association Network

In this paper, we consider the entities associated with words to enhance the relatedness measurement of word-to-word, and we build a Knowledge Association Network (KAN) to achieve this purpose. The symbols used in this paper are listed in Table 1.

**Table 1.** Symbols and their meanings

| Symbol | Meaning |
|---|---|
| $KAN$ | Knowledge Association Network |
| $G = (W, E, R)$ | A graph with word set $W$, entity set $E$ and edge set $R$ |
| $G_{\{attr,t\}}$ | Attribute space $G_{attr}$ and topological structure space $G_t$ |
| $R_{\{w,we,e\}}$ | Three types of edge set |
| $f_{\{w,we,e\}}$ | Three types of relatedness measurement |
| $W_{\{cnt,tf\_idf\}}$ | Two ways of weighting transition probability in $G_t$ |
| $e(w)$ | Entities set related to word $w$ |
| $\mathbb{R}_{\circlearrowleft}$ | Attribute space in vector space |
| $\mathbb{R}_{\approx}$ | Topological structure space in vector space |

**Definition 1 Knowledge Association Network (KAN).** *Knowledge association network is a graph $G = (W, E, R)$, where $W$ is the word set in vocabulary, $E$ is the entity set associated with the given words, and edge set $R$ denotes the relationships of word-to-word ($R_w$), entity-to-entity ($R_e$), and word-to-entity ($R_{we}$).*

There are many data resources that contain entities which are relevant to words such as Wikipedia, WordNet and DBPedia etc. WordNet provides precise lexical information but lacks adequate semantic information. Wikipedia is a large corpus where entities are described by natural language, that provides abundant unstructured semantic information. Recently, plenty of knowledge graphs are

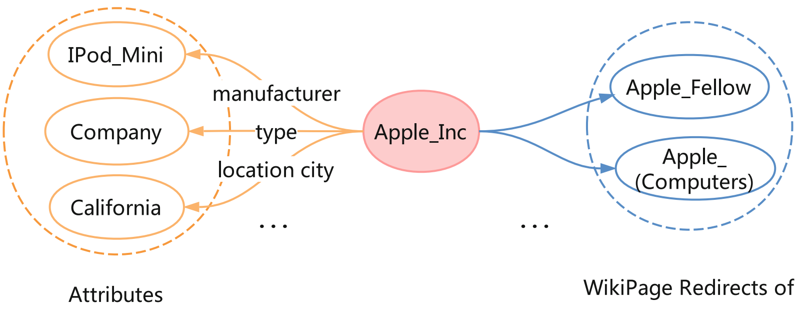


**Fig. 2.** The semantic features around an entity

established to hold structured knowledge. For example, DBPedia consists of a great number of entities and structured RDF format triples extracted from Wikipedia.

In this paper, we consider the DBPedia as entities database to avoid the significant preprocessing and data transformation efforts in Wikipedia [5,26]. To compute the relatedness of entity-to-entity, we consider two major factors in DBPedia: attributes information and topological structure. The attributes of an entity include the properties, categories, ontology information and some other information which enhance the entity itself. The topological structure reflects the relation between entities on the basis of a special predicate *WikiPageRedirectOf*, that means two entities appear in the same Wikipedia page.

As shown in Fig. 2, for the technology company Apple described as *Apple_Inc* in DBPedia, we get its attributes, that is, "Apple is the manufacturer of IPod Mini (properties)", "Apple is a company (categories)" etc. The relationship descriptions (e.g. "manufacturer", "is-a") are named on the basis of ontology language that contains affluent semantic information. In the aspect of links among other co-occurrence entities in the same Wikipedia page, there are *Apple_Fellow* and *Apple_(Computers)* in accordance with the special relationship *WikipageRedirectOf*.

To distinguish different semantic features of entities conveniently, we denote the attributes of an entity as attributes graph $G_{attr} = \{a_1, a_2, ..., a_j\}$, where $a_i$ denotes an attribute. We define topological structure as $G_t = G(E, R_{redirect})$, where $E$ is a set of entities connected by *WikiPageRedirectOf* (i.e. $R_{redirect}$).

In Wikipedia, a page and it's corresponding DBPedia entity describe the same entity. The predicate called *wikiPageID* reflects this mapping by one unique id, which can be obtained by the *Gensim*[3]. We can get the unique corresponding entity in DBPedia by *wikiPageID* and SPARQL endpoint[4]. For example, the id of Wikipedia page *Apple Inc* is 856, then we can use a simple query to get the corresponding entity name *Apple_Inc*:

[3] https://radimrehurek.com/gensim/wiki.html.
[4] http://dbpedia.org/sparql.

```
PREFIX dbo: <http://dbpedia.org/ontology/>
SELECT ?E WHERE {
    ?E dbo:wikiPageID 856.
}
```

## 3 Semantic Relatedness Measurement

We give an overview of our model in Fig. 3 where solid lines lead the flow of model and dotted lines demonstrate an additional function from source part to target part, which illustrates the construction of KAN and the relatedness measurement. (1) After an ordinary preprocessing in Wikipedia, for the words in vocabulary, we can get a mapping between words and pages in Wikipedia. (2) Then we query the unique entity by the page id by DBPedia SPARQL endpoint. (3) For the relatedness of entity-to-entity, we divide it into attribute and topological structure and adopt different models within it. Finally we combine three kinds of relatedness measurement word-to-word, word-to-entity and entity-to-entity to form the final semantic relatedness measurement.

### 3.1 Word-to-Word

The semantic relatedness in word level is mainly measured by (1) distributed vector representation such as word2vec [11] and GloVe [13] etc. (2) word co-occurrence [4,20], which means two words are relevant when they appear in a given window size. Experimental results prove that distributed vector representation works better in computing semantic relatedness [11]. Therefore in this paper, we abandon co-occurrence-based methods and adopt word2vec to train the Wikipedia corpus to product effective vector representation for each word. Formally, let $\overrightarrow{v}_i$ and $\overrightarrow{v}_j$ denote the vector representation of $w_i$ and $w_j$ which can be utilized to calculate the semantic relatedness $f_w(w_i, w_j)$ between $w_i$ and $w_j$ at the word-level based on cosine function, we have:

$$f_w(w_i, w_j) = cos(\overrightarrow{v}_i, \overrightarrow{v}_j) = \frac{\overrightarrow{v}_i \cdot \overrightarrow{v}_j}{\|\overrightarrow{v}_i\| \, \|\overrightarrow{v}_j\|} \tag{1}$$

The word2vec includes skip-gram and CBOW models, using either hierarchical softmax or negative sampling. The combination of skip-gram and negative sampling are used frequently and are effective experimentally. We choose this training program accordingly. The detailed parameters setting can be seen in experiments.

### 3.2 Word-to-Entity

In KAN, word-level and entity-level hold the one-to-many relationship. For a given word, several relevant entities will rise from KAN due to the word ambiguity. To measure the degree of association between a word ($w$) and an entity

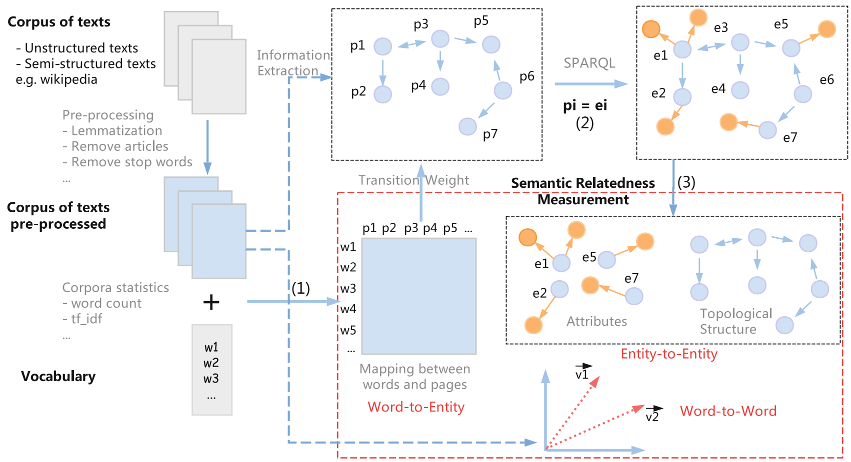


**Fig. 3.** Semantic relatedness measurement with KAN

($e$), (1) some researchers [15] take the co-occurrence times between $w$ and $e$ as the judgement of relatedness, which is insensitive for some common words like *this*, *that* and so on. (2) and some other works [5] consider $w$ and $e$ are closely related if $e$ is the only semantic meaning for word $w$. They compute the degree of strong connections between only anchor words and their out linked entities based on the *link popularity (LP)* equation,

$$LP(w,e) = \sum_{P} \sum_{w \in S} \frac{\sum_{w' \in S} tf\_idf(w',e)}{\sum_{e' \in e(w)} \sum_{w' \in S} tf\_idf(w',e')} \tag{2}$$

where $P$ indicates a page in Wikipedia, $S$ represents one sentence in $P$ that contains the word $w$, and $w'$ means every contextual word in $S$. $e(w)$ is a set of entities which are linked from anchor word $w$. This method just considers anchor word and out linked entities, but ignores the relevant pages that mention the word. In this paper, we extend the relevant entities $e(w)$ as:

$$e(w) = e_a(w) \cup e_m(w) \tag{3}$$

where $e_a(w)$ is the out linked entities set associated with $w$, and $e_m(w)$ contains entities that mention the word $w$ but not the out linked entities of $w$. So we have the *full popularity (FP)* that reflects the degree of connection between $w$ and $e$:

$$FP(w,e) = \begin{cases} LP(w,e) & e \in e_a(w) \\ \frac{tf\_idf(w,e)}{\sum_{e' \in e_m(w)} tf\_idf(w,e')} & e \in e_m(w) \end{cases} \tag{4}$$

Finally, we have the relatedness of word-to-entity defined as $f_{we}$:

$$f_{we}(w,e) = \frac{FP(w,e)}{\sum_{e' \in e(w)} FP(w,e')} \tag{5}$$

### 3.3 Entity-to-Entity

The knowledge association network at the entity-level is fundamentally a multi-relational graph where an entity is described by some discrete attribute and topological structure collectively. It is unreasonable to just consider either of the these information. Two entities may hold totally different attributes but they appear in the similar topological structure and vice verse. The part of attribute holds the detailed semantic information e.g. person A is the friend of B, person B is the member of organization C etc. The topological structure reflects the latent semantic information of co-occurrence relationship of entities. In our model, we adopt two different methods to obtain the vector representation of attribute and topological structure space.

**Attribute Space.** The straightforward method to embed a set of attributes around an entity is *one-hot*, where when one attribute appears in the attribute space of an entity, the corresponding vector position would be assigned 1, otherwise 0. Nevertheless, a surprisingly large number of attributes in DBPedia bring an insoluble problem for one-hot because of the excessive dimensions. Fortunately, there exists a kind of one-to-many relationship between entities and their attributes, which can be interpreted as a translation operation on the low-dimension entities embedding [2,21]. Suppose that there are N different attributes in our network and the attribute space is denoted as $\mathbb{R}_{\mathfrak{D}}{}^{|N|\times|d|}$, where $d$ is the dimension of vector for one attribute. We combine the relationships and entities to minimize a margin ranking loss over the attribute graph $G_{attr}$:

$$\mathcal{L} = \sum_{(a,b)\in G^{+}_{attr}} \sum_{b^{-}\in G^{-}_{attr}} [\ell + cos(a,b) - cos(a,b^{-})]_{+} \tag{6}$$

where $[x]_{+} = max(0,x)$, and $\ell$ is a margin hyperparameter. The $G_{attr}$ contains a set of $(h,r,t)$ triples, that is a head entity $h$, a relation $r$ and a tail entity $t$. We select uniformly at random to get positive sample $G^{+}_{attr}$ in two strategies: (i) $a$ consists of the bag of $h$ and $r$, while $b$ only consists of $t$; (ii) $a$ consists of $h$, $b$ consists of $r$ and $t$. Negative entities $b^{-}$ are sampled from the set of possible triples $G^{-}_{attr}$. We utilize a $k$-negative sampling strategy [11] to select $k$ negative pairs for each batch update. The optimization of this method inherits the strategy of stochastic gradient descent (SGD). Each SGD step is one sampling from $G^{+}_{attr}$ in the outer sum.

**Topological Structure Space.** The topological structure space ($G_t$) of an entity contains latent semantic information, for example, when somebody browses the Wikipedia page of *Apple_Inc*, there are lots of related entities contained in text description such as *Microsoft_Windows* and *Graphical user interface*, but they are not the attributes of *Apple_Inc*. To consider this latent semantic information, previous works [5,26] make lots of preprocessing in Wikipedia to extract the latent semantic features of entities. To avoid the extraction of link information in Wikipedia, we use DBPedia where a special relation named *WikiPageRedirectOf* connects two entities when one entity's anchor text is mentioned in the Wikipedia page of the other. Then we can get the topological structure space $G_t = G(E, R_{redirect})$, where $E$ is a set of entities and $R_{redirect}$ is the edge set formed by *WikiPageRedirectOf*.

It can be easily seen, $G_t$ is represented as a weighted graph model, where the edges in $R_{redirect}$ hold different transition weights. For instance, somebody is browsing the page of *Apple_Inc* in Wikipedia in which dozens of entities are linked. He wants to know more extended details about *Apple_Inc*, the most several related entities will draw his attention. So he will check the related out linked entities but ignore some other unrelated. It can be seen there are different transition weights from *Apple_Inc* to other entities. Moreover, the transition among different entities is directed, which means $G_t$ is a directed graph as well. Nevertheless, in DBPedia, the raw connections are represented as triples which are unweighted.

To get the weighted graph $G_t$, suppose that entity $e_i$ and $e_j$ are connected by $r_{ij}$, the most straightforward way to weight $r_{ij}$ is to consider the occurrence times of the anchor text of $e_j$ in the page of $e_i$. We regard the anchor text as a single term $t_i$ for $e_i$. Let $cnt(e_i, e_j)$ denote the co-occurrence times of appearance of $t_j$ in page of $e_i$. Formally we have the count-based transition weight $W_{cnt}(e_i, e_j)$ from $e_i$ to $e_j$:

$$W_{cnt}(e_i, e_j) = \frac{cnt(e_i, e_j)}{\sum_{e' \in P_i} cnt(e_i, e')} \tag{7}$$

where $P_i$ denotes the corresponding Wikipedia page of $e_i$. The $e^{'}$ is one out linked entity in $P_i$. However, just consider anchor text frequency would give some general frequent terms high degree of relatedness. In order to remedy this weakness, we calculate the *tf_idf*-based (Term Frequency–Inverse Document Frequency) transition weight $W_{tf\_idf}(e_i, e_j)$ from $e_i$ to $e_j$ as follows:

$$W_{tf\_idf}(e_i, e_j) = \frac{tf\_idf(e_i, e_j)}{\sum_{e' \in P_i} tf\_idf(e_i, e')} \tag{8}$$

After getting the weighted $G_t$, to make the entities are comparable in topological space, we need to embed the entities in expressive vector space. It is easy to understand that the related entities are close to each other in $G_t$ and they hold similar neighborhoods. It requires us to maximize the probability of observing neighborhoods for an entity. Formally, given an entity $e_i$, we predict its neighborhood entities $(e_0, e_1, ..., e_i, ...e_l)$ with the conditional probability $Pr$:

$$Pr((e_0, e_1, ..., e_{i-1}, e_{i+1}, ..., e_l)|e_i) \tag{9}$$

How to sample the neighborhood of an entity is widely studied in previous work [6,14]. In this paper, we adopt the randomized walk sampling strategy [6] to get the neighborhoods $N(e_i)$ around the entity $e_i$.

In order to maximize the probability of Eq. 9 in vector representation, we introduce a mapping function $\varPhi : e \in E \rightarrow \mathbb{R}_{\approx}^{|E| \times d}$ where E is the entity set of $G_t$. $\Phi$ is a $|E| \times d$ matrix of parameters which could be obtained by training. For each $e_i \in E$, we can get a $d$-dimension vector. And our goal is to minimize the following loss function:

$$minimize \ -logPr(N(e_i)|\Phi(e_i)) = -log \prod_{e^{'} \in N(e_i)} Pr(e^{'}|\Phi(e_i)) \tag{10}$$

where $Pr(e^{'}|\Phi(e_i))$ indicates how likely $e^{'}$ appears in neighborhoods of $e_i$. For each $e^{'} \in N(e_i)$, we adopt the softmax function to normalize the likelihood probability as each $e^{'}$ has a symmetric effect with $e_i$ in feature space [6], so we have conditional probability $Pr$:

$$Pr(e^{'}|\Phi(e_i)) = \frac{exp(\Phi(e^{'}) \cdot \Phi(e_i))}{\sum_{e_k \in N(e_i)} exp(\Phi(e_k) \cdot \Phi(e_i))} \tag{11}$$

Finally, We optimize function 10 using stochastic gradient descent (SGD).

**Relatedness of Entity-to-Entity.** We can get the embedding for an entity $e_i$, that consists of attributes embedding ($\overrightarrow{va}_i$) and topological space embedding ($\overrightarrow{vt}_i$). Formally, we formulate the relatedness of entity-to-entity as $f_e(e_i, e_j)$:

$$f_e(e_i, e_j) = \alpha cos(\overrightarrow{va}_i, \overrightarrow{va}_j) + (1 - \alpha) cos(\overrightarrow{vt}_i, \overrightarrow{vt}_j) \tag{12}$$

where $\alpha \in [0, 1]$ is to adjust the weights of two parts.

### 3.4 Word Semantic Relatedness Measurement *F*

The final semantic relatedness measurement has three parts including word-to-word, word-to-entity and entity-to-entity. We combine the word-to-entity and entity-to-entity as entity-level defined as $F_e(w_i, w_j)$:

$$F_e(w_i, w_j) = \sum_{e_i \in E_i} \sum_{e_j \in E_j} f_{we}(w_i, e_i) f_e(e_i, e_j) f_{we}(w_j, e_j) \tag{13}$$

where $E_i$ is the entities set associated with word $w_i$. And we denote the word-to-word relatedness as $F_w(w_i, w_j)$ that equals to $f_w(w_i, w_j)$. Finally, we can get the semantic relatedness measurement $F(w_i, w_j)$ in KAN:

$$F(w_i, w_j) = \varphi F_w(w_i, w_j) + (1 - \varphi) F_e(w_i, w_j) \tag{14}$$

where $\varphi \in [0, 1]$ trades off the weight of $F_w$ against $F_e$.

## 4 Experiments

In this section, we conduct extensive experiments on different datasets which contain the semantic relatedness measurement by human perceptions. We compute the Pearson correlation coefficient $\gamma$, Spearman correlation coefficient $\rho$ and harmonic mean coefficient $\mu = \frac{2\gamma\rho}{\gamma+\rho}$ between results of our experiment and scores of human judgement to evaluate the performance of our model.

### 4.1 Datasets

The Knowledge Association Network $KAN$ is constructed based on the dump of Wikipedia[5] and DBPedia[6]. The details about the basic datasets are shown in Table 2. The number of entities in DBPedia is larger than that in Wikipedia, since the entities set contain entities extracted from not only Wikipedia but also some other semantic datasets such as ontology language, YAGO and so on. It is necessary to preprocess the Wikipedia before constructing $KAN$. For each page in Wikipedia, we remove the stop words and punctuations, ignore the shorter pages whose words number less than 50 and some useless namespaces[7] such as *Category, File, Template* without introducing any entity.

**Table 2.** Wikipedia and DBPedia information

| | Entities | Date |
|---|---|---|
| Wikipedia | 5.5M | 2016–10 |
| DBPedia | 6.6M | 2016–10 |

### 4.2 Evaluation

A great number of datasets record the scores of human quantitative judgement for semantic relatedness. We evaluate $KAN$ on three frequently used datasets that are listed in Table 3. Based on the standard datasets, we compare our model with some existing models, containing (1) co-occurrence-based methods: ESA [4], SSA [8], word2vec [11] and SaSA [22]; (2) association network-based methods: AN [26] and HAN [5].

**Parameters Tuning.** In this paper, it is necessary to determine the following parameters:

- Recall word-to-word, we train word2vec in Wikipedia to get the vector representations for words. And we adopt *100 dimension, 30 window size, Skip-gram model and negative sampling* for word2vec.
- In the section of attributes space embedding, we set *margin* $\ell = 0.05$, *dimension* $d = 100$, *negative sampling number* $k = 50$, and we set the learning rate of SGD as 0.1 to optimize the margin ranking loss.
- In the section of embedding for topological structure space, the Skip-gram model is used for training the sequences of random walk, and we set the *100 dimension, 10 window size* as the basic parameters for training.
- $\alpha$ is proposed for the balance of attributes information and topological structure. $\varphi$ trades off the weight of word-level against entity-level.

[5] https://dumps.wikimedia.your.org/.
[6] https://wiki.dbpedia.org/downloads-2016-10.
[7] https://en.wikipedia.org/wiki/Wikipedia:Namespace.

**Table 3.** Word relatedness datasets information

| Datasets | Word pairs | Range of score | Reference |
|---|---|---|---|
| MC | 30 | [0, 4] | Miller and Charles (1991) |
| RG | 65 | [0, 4] | Rubenstein and Goodenough (1965) |
| WS353 | 353 | [0, 10] | Finkelstein et al. (2002) |

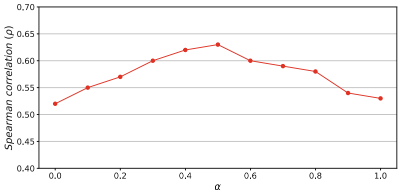


**Fig. 4.** $\alpha$ tuning on WS-Rel only considering Entity-to-Entity

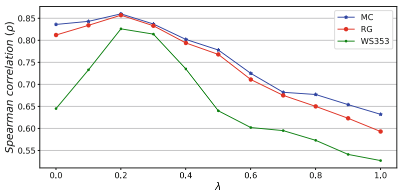


**Fig. 5.** Performance with value of $\lambda$

In order to get the optimal correlation, we pick *WS-Rel* [1] to tune the parameter $\alpha$, since there are not many comparison systems in literature report results on this dataset. *WS-Rel* contains 252 pairs of words along with relatedness judgement. We compute word semantic relatedness just on entity-to-entity part ($f_e$) to tune $\alpha$, as shown in Fig. 4, Spearman correlation ($\rho$) increases evidently when the importance of topological structure is raised. And we get the optimal values for $\alpha$ to be 0.5, which means attributes information and topological structure play the same role for semantic relatedness measurement.

Another parameter $\varphi$ trades off the weight of word-level relatedness $F_w$ against entity-level relatedness $F_e$. We set tuning rate as 0.1. Figure 5 shows the results w.r.t the multiple value of $\varphi$ and when $\varphi = 0.2$, we get the largest Spearman correlation ($\rho$). Obviously, $F_w$ has a leading role and our $F_e$ makes a great supplement for final semantic relatedness measurement.

**Comparions Results.** Evaluation results of word relatedness on different correlation coefficients are shown in Table 4. Recall embedding for topological structure of our network, there are two strategies to weight the relationship among entities: (1) $W_{cnt}(e_i, e_j)$ denotes the co-occurrence frequency of $e_j$ in page of $e_i$; (2) $W_{tf\_idf}(e_i, e_j)$ adopts $tf\_idf$ to judge how import an entity is to another. Based on these two weight strategies, we construct $KAN_{cnt}$ and $KAN_{tf\_idf}$ respectively. We can see that the $KAN_{tf\_idf}$ outperforms $KAN_{cnt}$ in different datasets and measurement coefficients, since $tf\_idf$ increases proportionally the number of times a term ($t$) appears in the page of an entity. And the value of $tf\_idf$ is offset by the number of pages in Wikipedia that contain the item $t$, which helps to adjust the weight for the fact that some items appear more frequently in general.

**Table 4.** Pearson-$\lambda$, Spearman-$\rho$, harmonic mean-$\mu$ on the word relatedness datasets

| Model | $\lambda$ | | | $\rho$ | | | $\mu$ | | |
|---|---|---|---|---|---|---|---|---|---|
| | MC | RG | WS353 | MC | RG | WS353 | MC | RG | WS353 |
| ESA | 0.588 | - - | 0.503 | 0.727 | - - | 0.748 | 0.650 | - - | 0.602 |
| SSA | 0.879 | 0.861 | 0.590 | 0.843 | 0.833 | 0.604 | 0.861 | 0.847 | 0.597 |
| word2vec | 0.852 | 0.834 | 0.633 | 0.836 | 0.812 | 0.645 | 0.844 | 0.823 | 0.639 |
| SaSA | 0.886 | 0.882 | 0.733 | 0.855 | 0.851 | 0.739 | 0.870 | 0.866 | 0.736 |
| $AN_{wiki}$ | 0.865 | 0.858 | 0.740 | 0.848 | 0.843 | 0.813 | 0.856 | 0.850 | 0.775 |
| $HAN_{wiki}$ | 0.886 | 0.884 | 0.772 | 0.860 | 0.857 | 0.826 | 0.873 | 0.870 | 0.798 |
| $KAN_{cnt}$ | 0.850 | 0.826 | 0.630 | 0.836 | 0.805 | 0.633 | 0.842 | 0.816 | 0.631 |
| $KAN_{tf\_idf}$ | **0.892** | **0.887** | **0.783** | **0.866** | **0.861** | **0.835** | **0.879** | **0.874** | **0.808** |

When compared with other methods shown in Table 4, our method performs better. $AN_{wiki}$ and $HAN_{wiki}$ get excellent performance on word semantic features relatedness on the idea of *free association network*, which improve the weakness of co-occurrence-based methods. In this paper, we adopt two different model to capture the semantic of attributes ($G_{attr}$) and topological structure ($G_t$) in $KAN_{tf\_idf}$ and make the model more flexible and expressive.

## 5 Related Work

Plenty of researchers have studied semantic relatedness between two words and made significant accomplishments, which include:

(i) The *lexical*-based methods measure the semantic relatedness between two words based on some lexical databases such as *WordNet* and *Wikitionary*. WordNet based methods [16] compute semantic relatedness for automatic speech recognition in meetings. However, they do not provide an individual result to reveal the efficiency of semantic relatedness measurements. Wikitionary [25] is introduced as an emerging lexical semantic resource that could be used as a substitute for expert-made resources in AI applications. Other lexical-based methods choose a path based approach [18], which can be utilized with any resource containing concepts connected by lexical semantic relations. Or they adopt a concept vector based approach [4], which is generalized to work on each resource that offers a textual representation of a concept.

(ii) The *co-occurrence*-based methods regard two words are related when they appear in a sentence or a fixed window in corpora texts such as Wikipedia. The initial model WikiRelate! [20] estimates relatedness based on categories in the articles of Wikipedia. Explicit Semantic Analysis (ESA) [4] represents the meaning of articles in a high-dimensional space. WikiRelate! and ESA only leverages texts in Wikipedia but does not consider links among articles. Another model WLM [12] scrutinizes incoming/outgoing links from/to articles instead of exploiting texts in Wikipedia articles. WikiWalk [24] extends the WLM by exploiting not only links that appear in an article but all links, to perform a random walk based on Personalized PageRank. However, those methods are faint to distinguish the different word senses. SensEmbed [9] leverages BabelNet[8] to annotate different word senses in the dump of Wikipedia, and exploits word2vec [11] to train the sense-annotated Wikipedia to get distributed representation of different word senses. Essentially this method is based on the large corpora and needs a significant pre-processing. The REWOrd [15] proposes an approach that exploits the graph nature of RDF and SPARQL query language to access knowledge graph. It not only obtains the comparable result with the state-of-art at that moment, but also avoids the burden of preprocessing and data transformations.

(iii) In order to improve the co-occurrence-based methods, *association network*-based methods is proposed to compute the semantic relatedness between two words utilizing *free association network*, that is, for a given word, the first word that appears in human mind intuitively is the most relevant one. AN [26] is proposed to build an association network based on not only co-occurrences between words, but also the links between Wikipedia pages of entities. Recently, HAN [5] constructs hierarchical association network to capture the association of word-to-word, word-to-entity and entity-to-entity. In this way, they make a great improvement in measuring the semantic relatedness. However, the adopted heuristic score functions are not reliable and cause the lack of flexibility. In addition, to get the semantic information of entities, they need significant preprocessing efforts in Wikipedia.

In this paper, we propose a *Knowledge Association Network* to measure semantic relatedness. Our model avoids the preprocessing of Wikipedia and considers the attribute and topological structure space simultaneously to capture the semantic features of entities. Experimental results show that our model outperforms the benchmarking models.

## 6 Conclusion

In this work, we focus on computing semantic relatedness to get an approximation to human judgement. We utilize the DBPedia which is derived from Wikipedia as background knowledge to construct a Knowledge Association Network. To measure the word semantic relatedness, we propose a flexible and expressive model to represent entities behind the words, where attribute and topological structure information of entities are embedded in vector space simultaneously. The experiments based on benchmarking datasets show that our model outperforms the state-of-the-art models.

**Acknowledgments.** This research is partially supported by National Natural Science Foundation of China (Grant No. 61572335, 61632016), the Natural Science Research

[8] http://babelnet.org.

Project of Jiangsu Higher Education Institution (No. 17KJA520003), and the Dongguan Innovative Research Team Program (No.2018607201008).

## References

1. Agirre, E., Alfonseca, E., Hall, K.B., Kravalova, J., Pasca, M., Soroa, A.: A study on similarity and relatedness using distributional and WordNet-based approaches. In: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, Boulder, Colorado, USA, 31 May–5 June 2009, pp. 19–27 (2009)
2. Bordes, A., Usunier, N., García-Durán, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: 27th Annual Conference on Neural Information Processing Systems, Lake Tahoe, Nevada, USA, 5–8 December 2013, pp. 2787–2795 (2013)
3. Fan, J., Lu, M., Ooi, B.C., Tan, W., Zhang, M.: A hybrid machine-crowdsourcing system for matching web tables. In: IEEE 30th International Conference on Data Engineering, Chicago, ICDE 2014, pp. 976–987 (2014)
4. Gabrilovich, E., Markovitch, S.: Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In: IJCAI, Hyderabad, India, 6–12 January 2007, pp. 1606–1611 (2007)
5. Gong, X., Xu, H., Huang, L.: HAN: hierarchical association network for computing semantic relatedness. In: AAAI, New Orleans, Louisiana, USA, 2–7 February 2018 (2018)
6. Grover, A., Leskovec, J.: Node2vec: scalable feature learning for networks. In: ACM SIGKDD, San Francisco, CA, USA, 13–17 August 2016, pp. 855–864 (2016)
7. Han, X., Zhao, J.: Structural semantic relatedness: a knowledge-based method to named entity disambiguation. In: ACL, Uppsala, Sweden, 11–16 July 2010, pp. 50–59 (2010)
8. Hassan, S., Mihalcea, R.: Semantic relatedness using salient semantic analysis. In: AAAI, San Francisco, California, USA, 7–11 August 2011 (2011)
9. Iacobacci, I., Pilehvar, M.T., Navigli, R.: SensEmbed: learning sense embeddings for word and relational similarity. In: ACL, Beijing, China, 26–31 July 2015, Volume 1: Long Papers, pp. 95–105 (2015)
10. Leong, C.W., Mihalcea, R.: Measuring the semantic relatedness between words and images. In: IWCS, Oxford, UK, 12–14 January 2011 (2011)
11. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR (2013)
12. Milne, D., Witten, I.H.: An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. In: AAAI Workshop on Wikipedia and Artificial Intelligence: An Evolving Synergy, pp. 25–30 (2008)
13. Pennington, J., Socher, R., Manning, C.D.: Glove: global vectors for word representation. In: EMNLP, Doha, Qatar, 25–29 October 2014, pp. 1532–1543 (2014)
14. Perozzi, B., Al-Rfou, R., Skiena, S.: DeepWalk: online learning of social representations. In: ACM SIGKDD, KDD 2014, pp. 701–710 (2014)
15. Pirrò, G.: Reword: semantic relatedness in the web of data. In: AAAI, Toronto, Ontario, Canada, 22–26 July 2012 (2012)
16. Pucher, M.: WordNet-based semantic relatedness measures in automatic speech recognition for meetings. In: ACL, Prague, Czech Republic, 23–30 June 2007 (2007)

17. Qadir, A., Mendes, P.N., Gruhl, D., Lewis, N.: Semantic lexicon induction from twitter with pattern relatedness and flexible term length. In: AAAI, Austin, Texas, USA, 25–30 January 2015, pp. 2432–2439 (2015)
18. Rada, R., Mili, H., Bicknell, E., Blettner, M.: Development and application of a metric on semantic nets. IEEE Trans. Syst. Man Cybern. **19**(1), 17–30 (1989)
19. Sandulescu, V., Ester, M.: Detecting singleton review spammers using semantic similarity. In: WWW, Florence, Italy, 18–22 May 2015, Companion Volume, pp. 971–976 (2015)
20. Strube, M., Ponzetto, S.P.: Wikirelate! computing semantic relatedness using Wikipedia. In: AAAI, Boston, Massachusetts, USA, 16–20 July 2006, pp. 1419–1424 (2006)
21. Wu, L.Y., Fisch, A., Chopra, S., Adams, K., Bordes, A., Weston, J.: StarSpace: embed all the things! In: AAAI, New Orleans, Louisiana, USA, 2–7 February 2018, pp. 5569–5577 (2018)
22. Wu, Z., Giles, C.L.: Sense-aware semantic analysis: a multi-prototype word representation model using Wikipedia. In: AAAI, Austin, Texas, USA, 25–30 January 2015, pp. 2188–2194 (2015)
23. Yang, J., Fan, J., Wei, Z., Li, G., Liu, T., Du, X.: Cost-effective data annotation using game-based crowdsourcing. PVLDB **12**(1), 57–70 (2018)
24. Yeh, E., Ramage, D., Manning, C.D., Agirre, E., Soroa, A.: WikiWalk: random walks on Wikipedia for semantic relatedness. In: Proceedings of the Workshop on Graph-based Methods for Natural Language Processing, Singapore, 7 August 2009, pp. 41–49 (2009)
25. Zesch, T., Müller, C., Gurevych, I.: Using wiktionary for computing semantic relatedness. In: AAAI, Chicago, Illinois, USA, 13–17 July 2008, pp. 861–866 (2008)
26. Zhang, K., Zhu, K.Q., Hwang, S.: An association network for computing semantic relatedness. In: AAAI, Austin, Texas, USA, 25–30 January 2015, pp. 593–600 (2015)
27. Zhang, W., Feng, W., Wang, J.: Integrating semantic relatedness and words' intrinsic features for keyword extraction. In: IJCAI, Beijing, China, 3–9 August 2013, pp. 2225–2231 (2013)
28. Zhu, G., Iglesias, C.A.: Computing semantic similarity of concepts in knowledge graphs. IEEE Trans. Knowl. Data Eng. **29**(1), 72–85 (2017)