Neural Processing Letters (2019) 50:1861–1875
https://doi.org/10.1007/s11063-018-9966-6

Measuring Entity Relatedness via Entity and Text Joint Embedding

Weixin Zeng[1] · Jiuyang Tang[1,2] · Xiang Zhao[1,2]

Published online: 17 December 2018

Abstract
As unique identifiers of objects and basic components of knowledge graphs, entities are crucial to many natural language processing related works, such as entity linking and question answering, in which the estimation of entity relatedness is required. Current entity relatedness measures either consider entities as words, which neglects the rich semantics entities contain, or are integrated into extrinsic applications, which fail to evaluate the intrinsic effectiveness. In this work, we propose E5, an effective entity relatedness measure taking into account of entity description text in a neural embedding manner. We first jointly map words and entities to the same high-dimensional vector space, the output of which is utilized as the input for the following joint entity and text embedding training. The well-trained entity and text embedding network is then leveraged to measure similarity between entities and entity descriptions, which in combination with a graph structure based method, constitute the eventual entity relatedness measure. The experimental results validate the usefulness of E5.

Keywords Entity relatedness · Joint embedding · Neural embedding network · Word embedding

1 Introduction

How similar are `Coldplay` and `Snow Patrol`? What are the closest entities to `Figure Skating`? Is `Oriental Pearl` related to `Beijing`? Nowadays, with the proliferation of knowledge graph (KG) and its vast applications, it is intuitive to hold doubts concerning the relatedness between entities—the basic components of KG. *Entities* are unique identifiers

✉ Xiang Zhao
xiangzhao@nudt.edu.cn

Weixin Zeng
zengweixin13@nudt.edu.cn

Jiuyang Tang
jiuyang_genius@sina.com

1 Key Laboratory of Science and Technology on Information System Engineering, National University of Defense Technology, Changsha, China

2 Collaborative Innovation Center of Geospatial Technology, Wuhan, China

of objects, which also serve as pivots connecting unstructured free-form texts with regularized KG. In many KG-related works, such as entity linking [1,2] and KG based document ranking [3], entity relatedness is considered as an indispensable part for enhancing overall performance.

Exploring the semantic relatedness of different entities is a routine yet deceptively complex task. Most of the existing entity relatedness measures are proposed and evaluated in *extrinsic* tasks, e.g., entity liking [4–6], while the *intrinsic* evaluation against human judgements of relatedness has been rare and confined mainly to word pairs instead of entity pairs [7]. Nonetheless, on the one hand, regarding entity relatedness as a sub-task of *extrinsic* problems might well render it task-specific and less applicable to other scenarios. On the other, the well-researched word similarity solutions cannot be directly applied to entity relatedness measurement due to the abundant semantic information entities contain. In consequence, the study of *entity* relatedness measure and *intrinsic* evaluation are of significance.

Existing entity relatedness measures can be mainly divided into two categories, methods based on *corpus text* and methods leveraging *graph structure*. The approaches harnessing *corpus text* model the textual description content of an entity with a vector of real numbers, which is then utilized to estimate relatedness with other vector-represented entities via traditional geometric measures. Meanwhile, in the line of works based on *graph structure*, entities are regarded as nodes and entity relatedness is in turn transformed to node-to-node similarity, which is characterized by the mutual neighbouring nodes or the whole graph structure. Figure 1 illustrates these two categories of approaches. Note that normally Wikipedia is leveraged as supporting KG due to its rich textual and graphical information.

The state-of-the-art *intrinsic* entity relatedness solution [7] implements representative existing solutions, among which the frontrunners are selected and constitute a two-stage

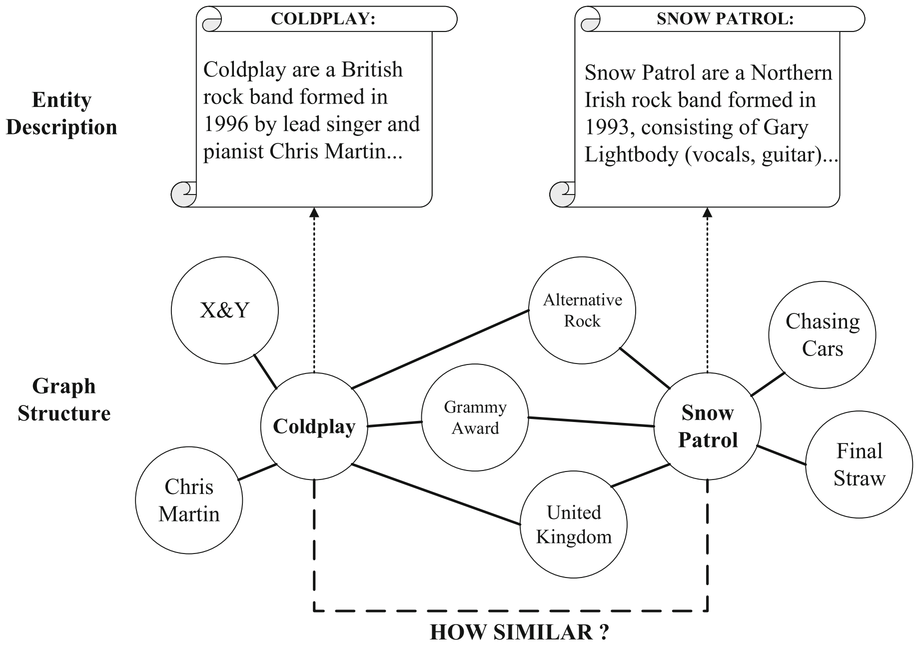

Fig. 1 Estimate entity relatedness via text corpus and graph structure

framework for computing entity relatedness with higher effectiveness and efficiency. Nevertheless, in the recommended configuration of the two-stage framework, merely *graph-based* approaches are harnessed, whereas the significance of *corpus text* is neglected. In addition, the *corpus text* based method reported with the best performance, albeit outperformed by *graph structure* approaches, is ENTITY2VEC, which extracts the latent semantic meanings of entities via embeddings. The inferiority can partially be attributed to the overlook of entity description text.

In short, the drawback of existing entity relatedness measures is two-fold:

- Most of the approaches are either adapted from word/document similarity measures, or driven by *extrinsic* tasks, which might overlook the semantics of entities and fail to cater to broader application scenarios; and
- The latent semantic meaning of entity description is yet neglected and *corpus text* information has not been taken fully advantage of.

In this work, we offer an *intrinsic* entity relatedness solution, E5, which measures Entity rElatedness via Entity and tExt joint Embedding. E5 comprises two methods. The primary contributor is the embedding similarity between entities and texts (entity descriptions) computed via the joint embedding network, which aims at making the most of the latent *corpus text* information. Additionally, *graph structure* is not overlooked and we adopt M&W [8], a method utilizing hyperlink structure of Wikipedia to characterize entity relatedness, the effectiveness of which has been embodied in many entity linkers. As a linear combination of these two measures, E5 achieves promising results in the experimental evaluation.

Furthermore, as input for joint entity and text embedding network, the result of joint entity and words embedding training affects the overall performance. Hence, we propose to enhance the embedding quality by introducing an expanded corpus and evaluate with parameter analysis and case study.

Contributions The main contributions of this article can be summarized into three ingredients:

- Joint entity and text embedding network is leveraged to measure the relatedness of entities in accordance to embedding similarities between entities and entity descriptions, which highlights and makes fully advantage of entity description text.
- We propose E5, an entity relatedness measure via entity and text joint embedding, which characterizes entity relatedness in terms of both *corpus text* and *graph structure*.
- The empirical results validate the usefulness of E5 regarding to the *intrinsic* evaluation of entity relatedness. Additionally, the joint embedding of entity and word using *expanded corpus* also proves to be effective.

Organization Section 2 overviews related work. Joint embedding of entity and text is elaborated in Sect. 3. Section 4 throws light on the experimental settings and entity relatedness evaluation results, followed by conclusion in Sect. 5.

2 Related Work

Entity Relatedness is a relatively new task and there is not much previous work directly devoted to measuring entity relatedness. Nonetheless, some existing solutions focused on calculating similarity between words and graph nodes can be adapted for measuring entity relatedness. Hence, we first overview relatedness measures focused on entities, which can further be divided into *intrinsic* and *extrinsic* evaluations. Then the extended relatedness methods are introduced, which are similar, but cannot be directly applied, to entity relatedness measurement.

***Intrinsic* entity relatedness methods** The difference between *intrinsic* and *extrinsic* entity relatedness evaluations lies in the motivations. While the former aims at developing general methods measuring entity relatedness that can cater to different downstream applications, the latter devises methods merely according to the requirement of specific tasks. Our work strives to offer an effective *intrinsic* entity relatedness measure.

The initial *intrinsic* entity relatedness measure could be traced back to [8] and [9], which centred on measuring relatedness between Wikipedia items. Especially, M&W [8] was established on the hypothesis that the semantic relatedness of two concepts can be defined by the number of incoming links they share. Nevertheless, in these *graph structure* based work, the conception of entity relatedness was not put forward and the focus was Wikipedia concept. The *intrinsic* entity relatedness task was formally proposed and defined in [10], in which the semantic meaning of an entity was represented by its distribution in the high dimensional concept space derived from Wikipedia. Zhao et al. [11] incorporated multiple types of relations to measure the semantic relatedness between Wikipedia entities and the task was transformed to completion of a sparse entity-entity association matrix. Still, the entity description text was yet not fully taken advantage of.

The state-of-the-art entity relatedness work [7] presented a thorough study of relatedness measures based on Wikipedia, offered an *intrinsic* evaluation dataset of entity relatedness, and devised a two-stage framework utilizing the existing entity relatedness measures with best performance. Despite that the method achieved promising results, the best configuration was still a combination of *graph structure* based approaches. In our work, we propose a *corpus text* based entity relatedness measure via joint embeddings, which in combination with a simple yet effective *graph structure* based method, can attain superior performance.

***Extrinsic* entity relatedness methods** Entity relatedness serves as a crucial part in many entity-related tasks such as entity linking and entity recommendation. Consequently, a large body of existing entity relatedness methods [4–6,12] were developed in those *extrinsic* tasks [13–19], where their *intrinsic* performances were not evaluated. Especially, Yamada et al. [4] proposed to measure entity relatedness via joint embedding of entity and word, which was similar to E5, but it did not directly model arbitrary-length text and neglected the contribution made by graph-based methods.

Extended relatedness measures There is a large number of studies devoted to measuring similarity between objects other than entities, such as words, documents, and graph nodes, which are similar to entity relatedness task. Existing methods can also be roughly clustered into two groups, relatedness based on *corpus text* and relatedness based on *graph structure*. The former models the textual content of a word/document/graph node with a real-number vector and outputs the relatedness by calculating cosine similarity between vectors. The representative approaches include Vector Space Model (VSM), Explicit Semantic Analysis (ESA) [20] and Latent Dirichlet Allocation (LDA) [21]. Meanwhile, the latter places targeted objects in a graph and computes relatedness via node-to-node similarity. Dominant methods comprise PPR+Cos [22], CoSIMRANK [23] and DEEPWALK [24].

In line with [7], we adapt aforementioned methods for measuring entity relatedness and the experiment results are reported and discussed in Sect. 4.

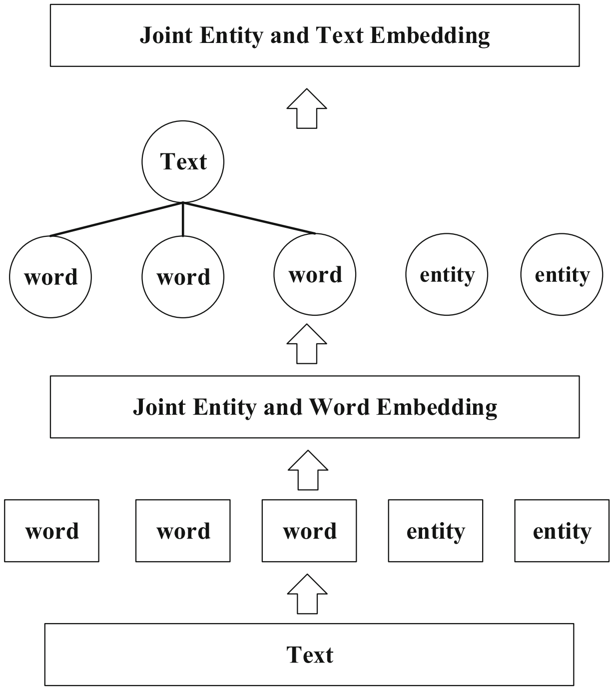

Fig. 2 Workflow of entity and text joint embedding

3 Methodology

Figure 2 illustrates the joint entity and text embedding process, which initiates from joint embedding of words and entities in text. The entity and word embeddings generated from the first stage are utilized as inputs for the second step. The second stage, joint entity and text embedding network, projects texts and entities into the same high-dimensional space, so that the relatedness scores between entities, entity description texts, entities and entity descriptions can be computed accordingly by utilizing cosine similarity. Eventually, E5 combines relatedness score generated from *corpus text* based joint embedding network, with a simple yet effective *graph structure* based method, M&W, to form overall entity relatedness measure.

3.1 Joint Embedding of Entity and Word

Embeddings are n-dimensional vectors of concepts that describe the similarities between these concepts using cosine similarity [25]. It is assumed that the concepts are similar if they frequently co-occur with the same other concepts. In literature, this has already been well researched for words [26] and documents [27]. Take word embedding as an instance. The word embedding vectors are designed to capture the semantic similarity between words when similar words are placed near one another in the vector space. Consequently, entities can also be projected to a relatively high dimensional vector space so as to better represent their semantic meanings.

Original Sentence:	Born in **Devon**, **Chris Martin** is the lead singer, guitarist, and pianist of **Coldplay**.
Entity Identifiers:	/wiki/Devon /wiki/Chris_Martin /wiki/Coldplay
Expanded 1:	Born in **/wiki/Devon**, **/wiki/Chris Martin** is the lead singer, guitarist, and pianist of **/wiki/Coldplay**.
Expanded 2:	**/wiki/Devon**, **/wiki/Chris Martin, /wiki/Coldplay**.

Fig. 3 Corpus Expansion

In line with recent work [4], we harness an embedding method that jointly embeds entities and words into the same vector space, where similar entities and words are placed in adjacent. Note that different from [4], we construct an expanded corpus for training, which yields embeddings with better quality, as reported in Sect. 4.

The joint embedding method stems from conventional skip-gram model [26] that learns word embedding, the training objective of which is to generate word representations that can predict context words given a specific word. Formally, let $w_1, w_2, ...w_N$ be a sequence of words, the model aims to maximize average log probability:

$$\Theta_w = \frac{1}{N}\sum_{i=1}^{N}\sum_{-c\leq j\leq c, j\neq 0} \log P(w_{i+j}|w_i). \tag{1}$$

where w_i represents the target word and w_{i+j} is a context word, c is the size of context window. The conditional probability is defined as:

$$P(w_{i+j}|w_i) = \frac{\exp({\upsilon'_{w_{i+j}}}^{\top}\upsilon_{w_i})}{\sum_{w\in W}\exp({\upsilon'_w}^{\top}\upsilon_{w_i})}. \tag{2}$$

where W represents the set of all words in the vocabulary, υ_w and υ'_w denote the 'input' and 'output' vector representations of word w. After training, the 'output' υ'_w is used for word embedding.

Then we extend the conventional skip gram model to joint embedding model. As for how to create the training corpus, specifically, the texts in Wikipedia pages consist of words and anchor texts and by utilizing the link associated with each anchor text, the entity identifier for the corresponding anchor text could be obtained. As is illustrated in Fig. 3, the expanded sentences (Expanded 1) for joint embeddings can be generated by replacing anchor texts with entity identifiers. Plus, entity identifiers from the original sentences are also extracted to form new inputs so as to better capture the relations between entities (Expanded 2).

Since entity identifier can be regarded as a special form of word, Eqs. 1 and 2 can be altered as follows:

$$\Theta_{ew} = \frac{1}{N}\sum_{i=1}^{N}\sum_{-c\leq j\leq c, j\neq 0} \log P(\tau_{i+j}|\tau_i). \tag{3}$$

$$P(\tau_{i+j}|\tau_i) = \frac{\exp({\upsilon'_{\tau_{i+j}}}^{\top}\upsilon_{\tau_i})}{\sum_{\tau\in\Gamma}\exp({\upsilon'_\tau}^{\top}\upsilon_{\tau_i})}. \tag{4}$$

where $\tau_1, \tau_2, \ldots \tau_N$ is a sequence of tokens (words or entity identifiers), τ_i and τ_{i+j} represent the target token and context token, respectively. Γ denotes the set of all tokens in the corpus, υ_τ and υ'_τ represent the 'input' and 'output' vector representations of token τ. After training, the 'output' υ'_τ is used for joint word and entity embedding.

3.2 Joint Embedding of Entity and Text

In spite of the fact that entity relatedness can be estimated by cosine similarity of entity embeddings derived from entity and word joint training process, it fails to take into account entity description information. Therefore, inspired by [28], we establish a neural network that jointly learns vector representations of texts and KB entities, so that the similarity between entities, entity description texts, an entity and a piece of of entity description text can be obtained accordingly, all of which can contribute to the estimation of entity relatedness.

Similar to the corpus for entity and word joint embedding, we train entity and text joint embedding on annotated Wikipedia pages. The target is to predict entities referred to by anchor links in Wikipedia text. Given a piece of text $t = \{w_1, w_2, \ldots w_N\}$, which contains entities $E_t = \{e_1, e_2, \ldots e_n\}$, Eqs. 1 and 2 can be transformed as follows to predict entities that appear in text:

$$\Theta_{et} = \sum_{(t,E_t)\in\Delta} \sum_{e\in E_t} \log P(e|t). \tag{5}$$

$$P(e|t) = \frac{\exp(\upsilon_e{}^\top \upsilon_t)}{\sum_{e^*\in E_K} \exp(\upsilon_{e^*}{}^\top \upsilon_t)}. \tag{6}$$

where Δ denotes a set of pairs, each of which comprises a text t, as well as the entities E_t contained in it. $P(e|t)$ represents the probability that a text t contains an entity e. All entities in KB are denoted by E_K and a random entity in E_K is represented by e^*. υ_e and υ_t are vector representations of entity e and text t respectively.

Noteworthy is that the vector representation υ_t of text $t = \{w_1, w_2, \ldots w_N\}$ is obtained by L_2 normalization of the sum of word embedding vectors in t:

$$\upsilon_t = W \frac{\sum_{m=1}^{N} \upsilon_{w_m}}{\| \sum_{m=1}^{N} \upsilon_{w_m} \|} + b. \tag{7}$$

where W is a weight matrix, b is a bias vector, which will be learned through the training process, and υ_{w_m} denotes the embedding vector of word w_m. Both word embedding υ_w and entity embedding υ_e are derived from joint embedding of entity and word.

3.3 Overall Entity Relatedness Measure

After obtaining the well-trained joint entity and text embedding network, given two entities e_i and e_j, with their Wikipedia description texts, d_i and d_j, the corpus text based entity relatedness can be measured by:

$$R_T(e_i, e_j) = \alpha_1 sim(\upsilon_{e_i}, \upsilon_{e_j}) + \alpha_2 sim(\upsilon_{e_i}, \upsilon_{d_j}) + \alpha_3 sim(\upsilon_{e_j}, \upsilon_{d_i}) + \alpha_4 sim(\upsilon_{d_i}, \upsilon_{d_j}) \tag{8}$$

where $sim(*)$ calculates the cosine similarity between two embedding vectors, and accordingly $sim(\upsilon_{e_i}, \upsilon_{e_j})$ denotes the similarity between the embeddings of two entities e_i and e_j, while $sim(\upsilon_{e_i}, \upsilon_{d_j}), sim(\upsilon_{e_j}, \upsilon_{d_i}), sim(\upsilon_{d_i}, \upsilon_{d_j})$ represent the similarity between e_i and d_j, e_j and d_i, d_i and d_j respectively, computed by utilizing our proposed joint text and entity embedding network. $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ are parameters balancing contributions made by different similarity measures, which are assigned with equal weights in our experiment.

Besides text corpus based method, as is pointed out in [7], a simple but effective graph structure based method, M&W, can achieve promising results. M&W is based on the assump-

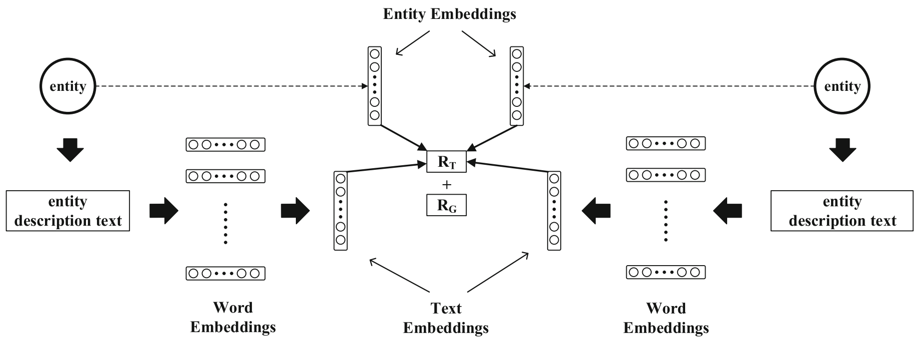

Fig. 4 Work flow of E5

tion that the semantic relatedness of two concepts can be defined by the number of incoming links they share. Formally, the M&W relatedness between entities e_i and e_j is:

$$R_G(e_i, e_j) = 1 - \frac{\log(\max\{|I(e_i)|, |I(e_j)|\}) - \log(|I(e_i) \cap I(e_j)|)}{\log(n) - \log(\min\{|I(e_i)|, |I(e_j)|\})} \tag{9}$$

where $I(e)$ denotes the incoming links of entity e, n represents the total number of entities in Wikipedia.

Hence, E5 generates the eventual relatedness between entities e_i and e_j as:

$$R(e_i, e_j) = \eta R_T(e_i, e_j) + \theta R_G(e_i, e_j) \tag{10}$$

where η and θ are two parameters balancing the importance of corpus text based and graph structure based measures, which are considered to be of equal significance and both assigned with 0.5 in our work.

Figure 4 further explains the work flow of E5. Specifically, given two entities and their description texts, the words in texts and entities are first transformed into embeddings via the well-trained network. Then we jointly embed the description texts and entities by harnessing the trained parameters. Accordingly, R_T, the corpus text based entity relatedness, can be obtained by combining the similarities between embeddings of texts and entities. In the end, we generate the eventual E5 relatedness by summing up R_T and the graph structure based entity relatedness R_G, where the specific definitions of R_T and R_G can be found above.

4 Experiment

In this section, we first elaborate the experimental settings, followed by evaluation results and analysis.

4.1 Experimental Settings

4.1.1 Additional Corpus Information

As is detailed in Sect. 3.1, the corpus for embedding training is composed of the original form and two expanded forms of each sentence, the punctuation of which should be removed before being forwarded to training. We use Wikipedia dump on 20-Mar-2018 [1] as text source.

[1] The Wikipedia dump could be downloaded from https://dumps.wikimedia.org/enwiki/20180320/.

The corpus for entity and text joint embedding is derived from DBpedia abstract corpus [29], which comprises the first introductory sections of all Wikipedia pages with links (entities) preserved. In contrast to the whole Wikipedia dump which might contain much noise, the first paragraph or section of each Wikipedia article generalizes the main topic and is of relatively higher quality. Plus, the first paragraph of entity's corresponding Wikipedia page is also considered as its entity description in our work.

4.1.2 Training Settings

For joint entity and word embedding training, we utilized word2vec implementation in Gensim [2]. The embedding dimension was set to 300, window size was assigned with 20, and iteration was 1.

As for learning the parameters, W and b, in entity and text joint embedding network, we initialized the dimension of the fully connected neural network layer to 300. The batch size was set to 128. To accelerate the training process, we utilized the negative sampling strategy [26] by generating the same number of irrelevant entities as negative samples. The number of negative samples for each positive pair was set to 10.

4.1.3 Dataset

Following the state-of-the-art work, we utilize WIRE [7] as the evaluation dataset, which consists of 503 pairs of named entities in Wikipedia and associated relatedness scores assigned by a group of human accessors. Entity pairs are of different levels of relatedness and many pairs are similar yet far in the Wikipedia graph structure, which renders KG distance a less effective entity relatedness measure.

Furthermore, we also consider WIKISIM [8] as an appropriate benchmark for the *intrinsic* evaluation of entity relatedness. It stems from WORDSIM-353 dataset comprising 353 word pairs, which is then manually annotated to their corresponding Wikipedia entities.

4.1.4 Evaluation Metric

The harmonic mean of *Pearson* correlation index and *Spearman* correlation index is utilized as the evaluation metric in our work. While the former, Pearson correlation index, highlights difference between predicted and correct results, the latter emphasizes the ranking order of the predicted relatedness values. The two indexes capture different aspects of the predicted results and are of equal significance to evaluating relatedness measures. In consequence, we adopt the harmonic mean of the two indexes as the final indicator, which can embody these two different features.

Specifically, suppose the relatedness scores generated by a specific method are denoted by $\mathbf{X}$, and the corresponding ground-truth relatedness scores are $\mathbf{Y}$. Then *Pearson* correlation coefficient is calculated by:

$$r_p = \frac{\sum \mathbf{XY} - \frac{\sum \mathbf{X} \sum \mathbf{Y}}{N}}{\sqrt{\left(\sum \mathbf{X}^2 - \frac{(\sum \mathbf{X})^2}{N}\right)\left(\sum \mathbf{Y}^2 - \frac{(\sum \mathbf{Y})^2}{N}\right)}}. \tag{11}$$

[2] https://radimrehurek.com/gensim/.

As for *Spearman* correlation index, both $\mathbf{X}$ and $\mathbf{Y}$ are first sorted in a ascending or descending order. Suppose the length of $\mathbf{X}$ and $\mathbf{Y}$ is n, and $d_i = \mathbf{X}_i - \mathbf{Y}_i$, $0 \le i < n$, then *Spearman* coefficient is calculated by:

$$r_s = 1 - \frac{6\sum_{i=1}^{n}(d_i)^2}{n(n^2-1)} \tag{12}$$

Then the harmonic mean of r_p and r_s is considered as the evaluation metric.

4.1.5 Competitors

We adopt the following measures, as well as the state of the art work [7], as competitors:

- Vector Space Model (VSM) measures entity relatedness by comparing the similarity of entity description texts, which are represented as sparse vectors over terms in text weighed by tf-idf.
- Explicit Semantic Analysis (ESA) [20] represents entities by their related Wikipedia articles with tf-idf weights. Then the relatedness is calculated by cosine similarity between sparse vectors of entities over all Wikipedia pages.
- Latent Dirichlet Allocation (LDA) [21] compares entities by cosine similarity between topic distribution weights of their description texts.
- ENTITY2VEC (E2V) [4] maps entities and words to the same embedding space and attains relatedness by entity embedding similarity score. This method is similar to our joint entity and word embedding, whereas we utilize the expanded corpus to improve embedding quality.
- PPR+Cos [22] measures the relatedness of two nodes in KG by cosine similarity between their PageRank vectors in the graph.
- CoSIMRANK [23] improves PPR+Cos by taking into account the fact that early meetings during the two separate random walks are of more value than later encounters.
- DEEPWALK (DW) [24] embeds the whole graph and measures the node relatedness via embedding similarity.
- M&W [8] is based on the assumption that the semantic relatedness of two concepts can be defined by the number of incoming links they share.
- Two-Stage Framework (TSF) [7] devises a two-stage framework, which creates a sub-graph of two entities by retrieving their most similar entities via M&W measure. The edge weights are computed by a linear combination of M&W and DW measures. Eventually entity relatedness is obtained by applying CoSIMRANK on the sub-graph.

4.2 Results and Analysis

4.2.1 Results Against Other Approaches

The full experiment results are presented in Tables 1 and 2. E5 achieves the best performance over three metrics and outperforms the runner-up by 2% on WIRE, and 1% on WIKISIM, in terms of harmonic value, which verifies the effectiveness of combining joint entity and text embedding network for measuring *corpus text* based relatedness, with M&W for evaluating *graph structure* based similarity.

Furthermore, to validate the superiority of our joint entity and word training process, we report the results of E5-, which merely utilizes the entity embedding obtained from the joint entity and word training to compute entity relatedness. Compared with E2V, which also

Table 1 Entity relatedness results on WIRE

Method	Pearson	Spearman	Harmonic
VSM	0.64	0.66	0.65
ESA	0.60	0.63	0.62
LDA	0.61	0.55	0.58
E2V	0.74	0.69	0.71
PPR+Cos	0.56	0.75	0.64
CoSIMRANK	0.56	0.76	0.64
DW	0.74	0.68	0.71
M&W	0.77	0.69	0.72
TSF	0.83	0.75	0.79
E5-	0.76	0.70	0.73
E5	**0.85**	**0.77**	**0.81**

Table 2 Entity relatedness results on WIKISIM

Method	Pearson	Spearman	Harmonic
VSM	0.43	0.55	0.48
ESA	0.62	0.72	0.67
LDA	0.58	0.57	0.58
E2V	0.67	0.66	0.66
PPR+Cos	0.20	0.72	0.31
CoSIMRANK	0.15	0.72	0.25
DW	0.71	0.70	0.70
M&W	0.62	0.65	0.63
TSF	0.74	0.75	0.74
E5-	0.69	0.67	0.68
E5	**0.76**	**0.75**	**0.75**

measures entity relatedness via entity embedding similarity, E5- improves the results by 2% on both datasets in terms of the Pearson correlation index and harmonic value. The superiority can be mainly attributed to the *expanded corpus*.

With regard to the comparison within corpus text based approaches and graph structure based methods, mapping entities or words to a higher dimensional space for measuring text-based similarity attains superior results since it can better capture the semantic meanings underneath text, while M&W, a simple method harnessing Wikipedia graph structure, surprisingly outperforms other graph-based solutions, which can be explained by the high quality of human-annotated Wikipedia links. TSF selects the best graph structure based methods to constitute a two-stage framework and further improves the results. Nonetheless, as a combination of two methods focused on different aspects, E5 attains the best overall performance.

4.2.2 Parameter Optimization

Noteworthy is that we assigned equal weights to the parameters in Eqs. 8 and 10, since there are no training/validation datasets and the annotation of entity relatedness scores between pairs of entities is both time and labour consuming.

Table 3 Parameter optimization results

	WIRE			WIKISIM		
	Pearson	Spearman	Harmonic	Pearson	Spearman	Harmonic
E5	0.85	0.77	0.81	0.76	0.76	0.75
E5+	0.85	0.79	0.82	0.77	0.77	0.77

Table 4 The most relevant entities/words to `Figure Skating` trained with E5- and E2V

E5-		E2V	
Name	Similarity score	Name	Similarity score
Figure Skater	0.739	Nordic skiing	0.681
Pair Skating	0.705	Curling	0.675
Ice Dancing	0.689	Triathlon	0.663
skater	0.685	Speed skating	0.661
Speed skating	0.680	Gymnastics	0.654

Nevertheless, to examine the effectiveness of parameter optimization on the final results, we conducted a fivefold cross validation by randomly splitting the WIRE and WIKISIM datasets, with 80% train and 20% test in each fold, and a linear RankSVM [30] was harnessed for parameter training as well as evaluation. The averaged result from the fivefold cross validation is denoted as E5+.

As is revealed in Table 3, performing parameter optimization can improve the final performance. However it would be unfair to compare E5+ with previous methods since the training set is obtained from the test datasets, i.e., WIRE and WIKISIM. Consequently, we merely aim to show that parameter optimization can improve overall results here and leave the construction of a large training dataset as future work, which might require the introduction of techniques such as distant supervision and crowdsourcing.

4.2.3 Time Consumption

It should be noted that the main time consumption of E5 comes from the two joint embedding processes, which can be trained in an off-line manner, while computing the relatedness score given two entities can be achieved within seconds by utilizing the well-trained framework. And since we have implemented the joint training process beforehand, the time consumption of E5, which can be represented by the time cost of the latter process, is relatively small.

4.2.4 Case Study

Table 4 presents the most relevant entities/words to entity `Figure Skating` trained with E5- and E2V. The bold items represent words while the rest denote entities. Compared with the results generated by E2V, which contain entities irrelevant to winter sports such as `Triathlon` and `Gymnastics`, our proposed training method with expanded corpus achieves superior performance.

In response to the question proposed in the beginning of the paper, 'How similar are `Coldplay` and `Snow Patrol`?', we present the most similar entities/words to entity

Table 5 The most relevant entities to Coldplay and Snow Patrol trained with E5-

Coldplay		Snow Patrol	
Name	Similarity score	Name	Similarity score
Keane (band)	0.833	Ash (band)	0.846
Kings of Leon	0.814	Paolo Nutini	0.840
Kasabian	0.814	Kaiser Chiefs	0.837
Arctic Monkeys	0.805	Biffy Clyro	0.836
The Killers	0.797	Arctic Monkeys	0.835

Coldplay and entity Snow Patrol in Table 5. It is evident that these two entities are closely related since their similar entities are all bands with indie/alternative style, and they share the same relative entity Arctic Monkeys. Nevertheless, if inspected carefully, the difference is also obvious. While the entities related to Coldplay are bands from different nations, the most similar entities of Snow Patrol are mainly Scottish or Northern Irish artists, indicating the wider popularity of Coldplay. The specific relatedness score between them calculated using E5 is 0.788.

5 Conclusion

Entities are unique identifiers of objects, which play an increasingly more significant role in many natural language processing related tasks. In those tasks, the estimation of entity relatedness is required. Current state-of-the art methods measure entity relatedness either by merely utilizing the graph structure of KG's, or by harnessing entity embeddings trained from text corpus, whereas the use of entity description text is yet neglected.

In this work, we propose E5, an effective entity relatedness measure which combines text corpus based and graph structure based approaches. The words and entities are first projected to the same high-dimensional vector space, and the outputs are utilized as inputs for the following joint entity and text embedding training. The well-trained entity and text embedding network can then be leveraged to measure similarity between entities and entity descriptions, which in combination with a graph structure based method, constitute the eventual entity relatedness measure. The experimental outcome not only verifies the effectiveness of E5, but also shows high quality of the word and entity embedding as an affiliated contribution.

Potential future research directions include applying the proposed measure on downstream tasks such as entity linking and entity recommendation, and creating a large entity relatedness training set by harnessing distant supervision or crowdsourcing.

Acknowledgements The authors would like to thank the anonymous reviewers for their insightful and constructive comments, which greatly contributed to improving the quality of the paper. This work was partially supported by NSFC under Grants Nos. 61872446 and 71690233.

References

1. Usbeck R, Röder M, Ngomo AN, Baron C, Both A, Brümmer M, Ceccarelli D, Cornolti M, Cherix D, Eickmann B, Ferragina P, Lemke C, Moro A, Navigli R, Piccinno F, Rizzo G, Sack H, Speck R, Troncy R, Waitelonis J, Wesemann L (2015) GERBIL: general entity annotator benchmarking framework. In:

Proceedings of the 24th international conference on world wide web, WWW 2015, Florence, Italy, pp 1133–1143. https://doi.org/10.1145/2736277.2741626
2. Zeng W, Zhao X, Tang J, Shang H (2018) Collective list-only entity linking: a graph-based approach. IEEE Access 6:16035–16045
3. Xiong C, Power R, Callan J (2017) Explicit semantic ranking for academic search via knowledge graph embedding. In: Proceedings of the 26th international conference on world wide web, WWW 2017, Perth, Australia, April 3–7, 2017, pp 1271–1279. https://doi.org/10.1145/3038912.3052558
4. Yamada I, Shindo H, Takeda H, Takefuji Y (2016) Joint learning of the embedding of words and entities for named entity disambiguation. In: Proceedings of the 20th SIGNLL conference on computational natural language learning, CoNLL 2016, Berlin, Germany, August 11–12, 2016, pp 250–259. http://aclweb.org/anthology/K/K16/K16-1025.pdf
5. Huang H, Heck LP, Ji H (2015) Leveraging deep neural networks and knowledge graphs for entity disambiguation. CoRR abs/1504.07678. http://arxiv.org/abs/1504.07678
6. Hoffart J, Seufert S, Nguyen DB, Theobald M, Weikum G (2012) KORE: keyphrase overlap relatedness for entity disambiguation. In: 21st ACM international conference on information and knowledge management, CIKM'12, Maui, HI, USA, October 29–November 02, 2012, pp 545–554. https://doi.org/10.1145/2396761.2396832
7. Ponza M, Ferragina P, Chakrabarti S (2017) A two-stage framework for computing entity relatedness in wikipedia. In: Proceedings of the 2017 ACM on conference on information and knowledge management, CIKM 2017, Singapore, November 06–10, 2017, pp 1867–1876. https://doi.org/10.1145/3132847.3132890
8. Witten IH, Milne D (2008) An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In: Proceedings of AAAI
9. Strube M, Ponzetto SP (2006) WikiRelate! computing semantic relatedness using wikipedia. In: Proceedings, the twenty-first national conference on artificial intelligence and the eighteenth innovative applications of artificial intelligence conference, July 16–20, 2006, Boston, Massachusetts, USA, pp 1419–1424. http://www.aaai.org/Library/AAAI/2006/aaai06-223.php
10. Aggarwal N, Buitelaar P (2014) Wikipedia-based distributional semantics for entity relatedness. In: AAAI fall symposium
11. Zhao Y, Liu Z, Sun M (2015) Representation learning for measuring entity relatedness with rich information. In: Proceedings of the twenty-fourth international joint conference on artificial intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25–31, 2015, pp 1412–1418. http://ijcai.org/Abstract/15/203
12. Ceccarelli D, Lucchese C, Orlando S, Perego R, Trani S (2013) Learning relatedness measures for entity linking. In: 22nd ACM international conference on information and knowledge management, CIKM'13, San Francisco, CA, USA, October 27–November 1, 2013, pp 139–148. https://doi.org/10.1145/2505515.2505711
13. Akhlaghi MI, Sukhov SV (2017) Knowledge fusion in feedforward artificial neural networks. Neural Process Lett 48:257–272. https://doi.org/10.1007/s11063-017-9712-5
14. Chandra R, Gupta A, Ong YS, Goh CK (2017) Evolutionary multi-task learning for modular knowledge representation in neural networks. Neural Process Lett 47:907–919. https://doi.org/10.1007/s11063-017-9718-z
15. Wu C, Shi X, Su J, Chen Y, Huang Y (2017) Co-training for implicit discourse relation recognition based on manual and distributed features. Neural Process Lett 46(1):233. https://doi.org/10.1007/s11063-017-9582-x
16. Zhao S, King I, Lyu MR (2017) Aggregated temporal tensor factorization model for point-of-interest recommendation. Neural Process Lett 47:975–992. https://doi.org/10.1007/s11063-017-9681-8
17. Jung C, Shen Y, Jiao L (2015) Learning to rank with ensemble ranking SVM. Neural Process Lett 42(3):703. https://doi.org/10.1007/s11063-014-9382-5
18. Zhao X, Xiao C, Lin X, Zhang W, Wang Y (2018) Efficient structure similarity searches: a partition-based approach. VLDB J 27(1):53. https://doi.org/10.1007/s00778-017-0487-0
19. Liu X, Zhu X, Li M, Wang L, Tang C, Yin J, Shen D, Wang H, Gao W (2018) Late fusion incomplete multi-view clustering. In: IEEE transactions on pattern analysis and machine intelligence
20. Gabrilovich E, Markovitch S (2007) Computing semantic relatedness using wikipedia-based explicit semantic analysis. In: IJCAI 2007, Proceedings of the 20th international joint conference on artificial intelligence, Hyderabad, India, January 6–12, 2007, pp 1606–1611. http://ijcai.org/Proceedings/07/Papers/259.pdf
21. Hoffman MD, Blei DM, Bach FR (2010) Online learning for latent Dirichlet allocation. In: Advances in neural information processing systems 23: 24th annual conference on neural information processing systems 2010. Proceedings of a meeting held 6–9 December 2010, Vancouver, British Columbia, Canada, pp 856–864. http://papers.nips.cc/paper/3902-online-learning-for-latent-dirichlet-allocation

22. Haveliwala TH (2002) Topic-sensitive PageRank. In: Proceedings of the eleventh international world wide web conference, WWW 2002, May 7–11, 2002, Honolulu, Hawaii, pp 517–526. https://doi.org/10.1145/511446.511513
23. Rothe S, Schütze H (2014) CoSimRank: a flexible & efficient graph-theoretic similarity measure. In: Proceedings of the 52nd annual meeting of the association for computational linguistics, ACL 2014, June 22–27, 2014, Baltimore, MD, USA, volume 1: long papers, pp 1392–1402. http://aclweb.org/anthology/P/P14/P14-1131.pdf
24. Perozzi B, Al-Rfou R, Skiena S (2014) DeepWalk: online learning of social representations. In: The 20th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '14, New York, NY–August 24–27, 2014, pp 701–710. https://doi.org/10.1145/2623330.2623732
25. Zwicklbauer S, Seifert C, Granitzer M (2016) Robust and collective entity disambiguation through semantic embeddings. In: Proceedings of the 39th international ACM SIGIR conference on research and development in information retrieval, SIGIR 2016, Pisa, Italy, July 17–21, 2016, pp 425–434. https://doi.org/10.1145/2911451.2911535
26. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems 26: 27th annual conference on neural information processing systems 2013. Proceedings of a meeting held December 5–8, 2013, Lake Tahoe, Nevada, United States, pp 3111–3119
27. Le QV, Mikolov T (2014) Distributed representations of sentences and documents. In: Proceedings of the 31th international conference on machine learning, ICML 2014, Beijing, China, 21–26 June 2014, pp 1188–1196. http://jmlr.org/proceedings/papers/v32/le14.html
28. Yamada I, Shindo H, Takeda H, Takefuji Y (2017) Learning distributed representations of texts and entities from knowledge base. TACL 5:397. https://transacl.org/ojs/index.php/tacl/article/view/1065
29. Brümmer M, Dojchinovski M, Hellmann S (2016) DBpedia abstracts: a large-scale, open, multilingual NLP, training corpus. In: Proceedings of the tenth international conference on language resources and evaluation LREC 2016, Portorož, Slovenia, May 23–28, 2016. http://www.lrec-conf.org/proceedings/lrec2016/summaries/895.html
30. Joachims T (2002) Optimizing search engines using clickthrough data. In: Proceedings of the eighth ACM SIGKDD international conference on knowledge discovery and data mining, July 23–26, 2002, Edmonton, Alberta, Canada, pp 133–142