Knowledge-Based Systems 164 (2019) 38–44

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

# Knowledge graph embedding with concepts

Niannian Guan, Dandan Song *, Lejian Liao

*School of Computer Science and Technology, Beijing Institute of Technology, China*

ARTICLE INFO

*Article history:*
Received 17 May 2018
Received in revised form 4 October 2018
Accepted 5 October 2018
Available online 15 November 2018

*Keywords:*
Knowledge graph embedding
Concept space
Knowledge graph completion

ABSTRACT

Knowledge graph embedding aims to embed the entities and relationships of a knowledge graph in low-dimensional vector spaces, which can be widely applied to many tasks. Existing models for knowledge graph embedding primarily concentrate on entity–relation–entity triplets, or interact with the text corpus. However, triplets are less informative, and the in-domain text corpus is not always available, making the embedding results deviate from the actual meaning. At the same time, our mental world contains many concepts about worldly facts. For human cognition, compared to knowledge that we learned, common-sense concepts are more basic and general, and they play important roles in human knowledge accumulation. In this paper, based on common-sense concepts information of entities from a concept graph, we propose a ***Knowledge Graph Embedding with Concepts (KEC)*** model that embeds entities and concepts of entities jointly into a semantic space. The fact triplets from a knowledge graph are adjusted by the common-sense concept information of entities from a concept graph. Our model not only focuses on the relevance between entities but also focuses on their concepts. Thus, this model offers precise semantic embedding. We evaluate our method on the tasks of knowledge graph completion and entity classification. Experimental results show that our model outperforms other baselines on the two tasks.



## 1. Introduction

As a collection of human knowledge, knowledge graphs have become important resources for many Artificial Intelligence and Natural Language Processing applications, such as question answering, web searching and semantic analysis. Among the relevant processes, knowledge embedding is a key step for knowledge representation and knowledge graph utilization, especially because of its compatibility with deep learning methods, which are currently becoming increasingly popular.

Knowledge graphs encode structured information of relational facts that are often represented in the form of a triplet (head entity, relation, tail entity) (denoted as $(h, r, t)$). The embedding of knowledge graphs is to learn continuous vector representations (embeddings) for entities and relations of a structured knowledge base (KB) such as Freebase [1] and Wordnet [2], which aims at providing a numerical computation framework for knowledge graphs. To accomplish this goal, many embedding methods have been explored, such as TransE [3], PTransE [4] and so on.

Among these methods, the translation-based methods, such as TransE and TransH [5], are simple and effective. They build entity and relation embeddings by regarding a relation as a translation from a head entity to a tail entity. These models assume embeddings of entities and relations that are in the same semantic space $R^k$. However, only the triplet information is deficient. As Fig. 1 shows, the head and tail entities can have multiple concepts, and various relations can focus on different concepts of entities, which makes only one semantic space insufficient for modelling.

The improved knowledge embedding methods with textual descriptions have achieved much success, such as DKRL [6], SSP [7], and Jointly(A-LSTM) [8], with deep neural networks. They employ supplementary textual descriptions of entities and relations in their embedding to discover semantic relevance. These methods have been successfully applied when there are many entity and relation-related textual descriptions. However, textual descriptions are not always available, especially for in-domain descriptions, while a cross-domain corpus would cause deviations in the embedding results.

At the same time, while our mental world contains many concepts about worldly facts, a ***concept*** is another important resource for human cognition. Compared to knowledge, common-sense concepts are more basic and general, and they play important roles in human knowledge learning and accumulation. Therefore, in our intuition, concept information would greatly benefit the knowledge embedding results.

Microsoft Concept Graph [9] maps text entry entities to different semantic concepts and is tagged with corresponding probability labels that depend on the context. For example, as Fig. 2 shows, the instance (i.e., entity) *"apple"* is mapped to *"fruit"*, *"company"* and other concepts, with the corresponding frequency tags

* Corresponding author.
*E-mail address:* sdd@bit.edu.cn (D. Song).

https://doi.org/10.1016/j.knosys.2018.10.008
0950-7051/

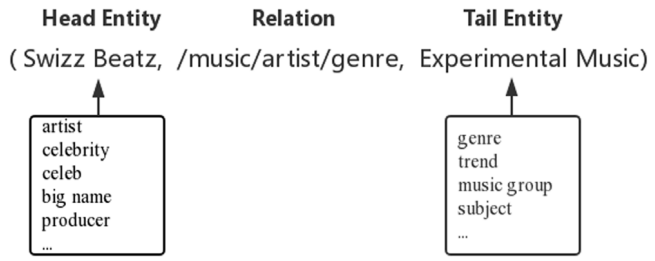


**Fig. 1.** Concept information for entities in a fact triplet.

| Instance | Concept | Relations |
|---|---|---|
| apple | fruit | 6315 |
| apple | company | 4353 |
| apple | food | 1152 |
| apple | brand | 764 |
| apple | fresh fruit | 750 |
| apple | fruit tree | 568 |
| apple | crop | 483 |
| apple | corporation | 280 |
| apple | manufacturer | 279 |
| apple | firm | 257 |

**Fig. 2.** *"apple"* in the Microsoft Concept Graph.

(defined as "Relations" by Microsoft). Conceptualization maps instances or short text into a large auto-learned concept space, which is a vector space with human-level concept reasoning. Therefore, we could construct a concept subspace with concepts with a head entity and tail entity in the concept graph for every triplet, which enables us to compute the meaningful concept relevance between entities in a principled way.

The incorporation of a concept graph into knowledge embedding could reveal the concept relevance between entities and contribute to precise semantic expression. Additionally, the concept relevance and precise semantic expression could be able to recognize true triplets. For example, for two triplets *(Apple, Developer, IPhone)* and *(Apple, Developer, Samsung Mobile)*, it is quite difficult to distinguish which is the true triplet that contains fact triplets only, because *"IPhone"* and *"Samsung Mobile"* both belong to mobile phones. However, in the concept graph, *"IPhone"* has a concept *"apple device"*, but *"Samsung Mobile"* does not. Thus, it is easy to infer the correct triplet by mapping *"IPhone"* to the *"apple device"* concept.

Compared to embedding methods with textual information, the embedding method with concept information is more general in its tasks, and it does not depend on the topic of the corpus. Specifically, when a corpus about technology is provided, embedding methods with technical textual descriptions could easily infer the fact *(Apple, Developer, IPhone)*, because the keywords *"hardware products"* and *"iPhone smartphone"* occur frequently in the textual description of *"Apple"*. However, it is difficult to infer the fact *(Apple, Taste, Sweet)*, which is irrelevant to textual descriptions of *"Apple"* about the specific topic of *"technology company"*.

In contrast, the concept information of *"apple"* contains related concepts such as *"fruit"*, *"taste"*, and *"food"*. These concepts can help to refine the topic of *"apple"* from *"technology company"* to *"fruit"*. Therefore, concept information is different from the domain-specific textual corpus, and it covers a variety of topics. Furthermore, textual descriptions must be extracted from knowledge bases or other resources, while the Microsoft Concept Graph has provided concept information that could be directly used in applications. Therefore, our method is relatively more efficient and has a lower labour requirement in the data acquisition.

In this paper, we propose a model for Knowledge Graph Embedding with Concepts (KEC), which constructs strong correlations between symbolic triplets and a concept graph by performing the embedding process in concept subspaces. Specifically, we measure the possibility of a triplet by projecting the loss vector onto a concept subspace such as a hyperplane that represents the concept relevance between entities. Thus, a fact triplet is always accepted as long as the 2-norm of its projected loss vector in the semantic concept space is sufficiently small.

We evaluate our model with the tasks of knowledge graph completion and entity classification on the benchmark datasets of Freebase and WordNet. Experimental results show that the KEC model achieves significant and consistent improvements compared to state-of-the-art models.

## 2. Related work

In recent years, many knowledge embedding methods have been developed, and they usually include three branches that are related to our work: *Knowledge Graph Embedding* models with only symbolic triplets, *Embedding with Textual Information* models, and *Embedding with Category Information* models.

### 2.1. Embedding with symbolic triplets

A knowledge graph is embedded into a low-dimensional continuous vector space while certain properties of it are preserved [10]. For example, the most famous TransE [3] model relationships occur by interpreting them as translations that operate on the low-dimensional embeddings of entities. TransE performs well in 1-to-1 relations, while it has issues for modelling 1-to-N, N-to-1 and N-to-N relations. Thus, many variants of TransE are assumed, and they would transform entities into different subspaces. For example, TransH [5] and TransR [11] allow a entity to have different representations under different relationships. There are also some other systems, such as RESCAL [12], SE [10] and LFM [13].

However, triplet information only is less-informative. Because there are other available sources of information (such as textual information in the following subsection, and concept information in this paper) that could be supplementary to augment the knowledge graph embedding, some improved models are proposed.

### 2.2. Embedding with textual information

Embedding with textual information attempts to use textual information to help knowledge graph representation learning. The studies in [14], [15] and [16] jointly embed knowledge and text into the same space by means of aligning methods. SSP [7] focuses on the stronger semantic interaction by projecting triplet embedding onto a semantic subspace such as a hyperplane. Jointly (A-LSTM) [8] uses a bidirectional long short-term memory network (LSTM) [17,18] with an attention mechanism [19] to model the text description.

The textual information is approved as effective in helping knowledge graph embedding. However, a text corpus, mainly text descriptions for entities, is required for these methods. However, text descriptions are not available or are not well-prepared for many long-tailed entities. If a cross-domain corpus is used, it would make the embedding results deviate.

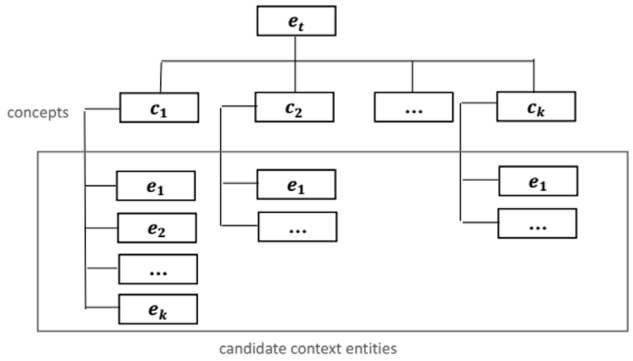


**Fig. 3.** An illustration of context entities.

### 2.3. Embedding with category information

There are some embedding methods that integrate hierarchical category information from large-scale knowledge bases. Entity Hierarchy Embedding [20] learns a distance metric for each category node and measures entity vector similarity under aggregated metrics. HCE [21] learns entity and category embeddings to capture the semantic relatedness between entities and categories. However, their goal is to extract semantics from plain text, which is different from our knowledge graph embedding target.

## 3. Methodology

Our Knowledge Graph Embedding with Concepts (KEC) model is to integrate concept graph information into knowledge graph embedding. Therefore, it is composed of two components: concept graph embedding and concept graph information enhanced knowledge graph embedding.

Inspired by the knowledge graph embedding with text description method SSP [7], our model performs concept space projection for knowledge graph embeddings. In other words, the loss vector of a triplet is projected onto a concept subspace as a hyperplane that represents the concept relevance between entities.

### 3.1. Concept graph embedding

In a Microsoft Concept Graph, as shown in Fig. 2, data is stored in the form of (Instance (i.e., entity), Concept, Relations (i.e., frequency)). It should be noted that the use of "Relations" here is different from the relation $r$ in a knowledge graph triplet $(h, r, t)$. Because the Microsoft Concept Graph contains the "IsA" relation in its current release, here, the "Relations" indicates the frequency of the instance mapped to the concept, which is measured by the total number of its occurrences.

To learn representations for concepts and entities that can capture their semantic relatedness, we use the skip-gram model [22] for embedding. The skip-gram model aims at generating word representations that are good at predicting context words that surround a target word in a sliding window. We extend the entity's context to be the entities with the largest frequencies that are conceptualized into the same concepts. Therefore, a set of entity pairs $D = (e_t, e_c)$ is acquired from the concept graph triplets as follows, where $e_t$ denotes the target entity, and $e_c$ denotes the context entity.

In a concept graph, as shown in Fig. 3, each entity $e_t$ can be conceptualized into one or more concepts $(c_1, c_2, \ldots, c_k)$, $k \geq 1$, and each concept $c_k$ contains one or more entities $(e_1, e_2, \ldots, e_{k_n})$, $k_n \geq 1$, which are treated as candidate context entities. To learn the embeddings of entities and concepts simultaneously, we adopt a method that incorporates the labelled concepts into the target entity when predicting its context entities, which is similar to the TWE-1 model [23], which combines topic information with words to predict context words.

For example, if $e_t$ is the target entity, then its labelled concepts $(c_1, c_2, \ldots, c_k)$ would be combined with $e_t$ to predict the context entities. For each target–context entity pair $(e_t, e_c)$, the basic Skip-gram formulation defines the probability of $e_c$ being the context of $e_t$ or $c_i$ using the softmax function:

$$P(e_c \mid e_t) = \frac{exp(e_c \cdot e_t)}{\sum_{e \in E} exp(e \cdot e_t)} \tag{1}$$

$$P(e_c \mid c_i) = \frac{exp(e_c \cdot c_i)}{\sum_{e \in E} exp(e \cdot c_i)} \tag{2}$$

where $e_t$, $e_c$, and $c_i$ are vector representations of the target entity, the context entity, and the concept of the target entity, respectively. $E$ denotes the set of entities, and $exp$ is the exponential function.

Our concept graph embedding object is to maximize the average log probability:

$$L = \frac{1}{|D|} \sum_{(e_c, e_t) \in D} [\log P(e_c \mid e_t) + \sum_{c_i \in C(e_t)} \log P(e_c \mid c_i)] \tag{3}$$

where $D$ denotes the set of target–context entity pairs, and $C(e_t)$ denotes the concept set of entity $e_t$. We adopt the stochastic gradient descent method to optimize the object. The initial learning rate is 0.025, which decreases with an increase in the training samples according to the following function:

$$\alpha = starting\_alpha \times (1 - count\_actual/(real)(iter \times total\_size + 1)) \tag{4}$$

where $\alpha$ is the learning rate, $count\_actual$ is the number of trained samples, $iter$ is the number of iterations, and $total\_size$ is the total number of training examples.

To handle large data sets, we use 12 threads in the training to train the model in ten minutes. To avoid overfitting, we adopt the "negative sampling" method for the optimization target.

### 3.2. Knowledge graph embedding

To characterize a strong correlation between the knowledge base triplets and concept information, we attempt to embed a specific triplet into the concept subspace. First, we construct a hyperplane with the normal vector $c$ as the concept subspace:

$$c = C(e_h, e_t) = \frac{e_h - e_t}{\|e_h - e_t\|_2^2} \tag{5}$$

where $C : R^{2d} \rightarrow R^d$ is the concept composition function, and $e_h$, $e_t$ is the head entity and tail entity semantic vectors that are acquired by the concept graph embedding, respectively. Since learning semantic embeddings with a skip-gram model can make the vectors that connect the pairs of entities of a certain relation almost parallel, e.g., vec("King") −vec("Queen") ≈ vec("Man") − vec("Women"), we suggest that the concept composition should take the subtraction form.

The TransE model defines the score function as $\|h + r - t\|_2^2$, which means that the triplet embedding depends on the loss vector $l = h + r - t$ . Therefore, we can compute that the component of the loss in the normal vector direction is $(c^T lc)$. Then, the other orthogonal component, which is projected onto the hyperplane, is $(l - c^T lc)$, as shown in Fig. 4. As a result, assuming that $e$ is length-fixed, the basic idea behind our model is to maximize the component inside the concept hyperplane, which is $\|l - c^T lc\|_2^2$.

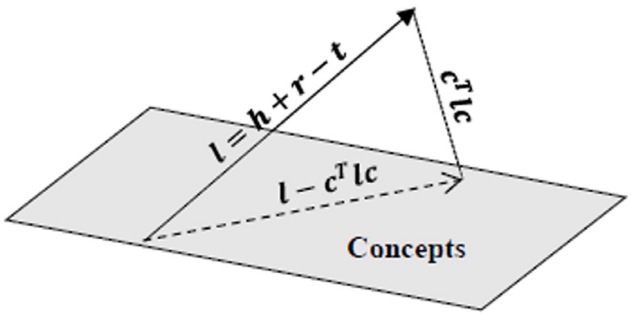


**Fig. 4.** Knowledge embedding loss vector projected onto the concept hyperplane.

We define the total score function as follows:

$$f_r(h,t) = -\lambda \|l - c^T lc\|_2^2 + \|l\|_2^2 \tag{6}$$

where $\lambda$ is a suitable hyper-parameter factor to balance the two parts. Moreover, the score function favours a lower value of the energy for a triplet in the training set than for a corrupted triplet. Furthermore, the projection part in our score function is negative, and thus, more projection means less loss.

### 3.3. Model interpretation

First, our model could discover the concept relevance. We aim at projecting the loss $h' - t$ onto the hyperplane, where $h' = h + r$ is the translated head entity. We then utilize the theorem that states that if a line lies on a hyperplane, then all of the points of that line lie on the hyperplane also. Thus, the entities that have relevant concepts always lay on a consistent hyperplane, and the loss vector between them $(h' - t)$ is also around the hyperplane. Based on this geometry, a corrupted triplet is far away from the hyperplane, which results in a large loss. In contrast, even if the loss of a correct triplet is large in terms of the knowledge embedding, it could be smaller (or better) after being projected onto the hyperplane. These considerations would imply that the concept information raises the concept relevance, which refines the knowledge embedding. For example, it is difficult to infer the possibility of the fact (*Christopher Plummer, /people/person/nationality, Canada*) only within the knowledge embedding. It ranks 11,831 out of 14,951 by TransE in link prediction, which would be classified as an impossible fact. However, in a concept graph, *"Christopher Plummer"* has the *"actor"* and *"star"* concepts, while *"Canada"* has the *"country"* and *"nation"* concepts. It is obvious that there exists the relationship *"/people/person/nationality"* between the concepts of the two entities. This consideration means that the two entities are relevant in the semantic concept space. As expected, the triplet ranks 49 out of 14,951 in our model after the concept projection, which means that it is more plausible.

Second, our model could offer a precise semantic expression. In TransE, if the loss vectors of the triplets are equal-length, then they are regarded as equivalent, and it is difficult to discriminate them. However, our model incorporates conceptual semantics to strengthen the discriminative ability. Specifically, we project the equal-length loss vectors onto the corresponding concept hyperplanes as the loss results, which makes a reasonable division of the losses. For an instance of the query about which the director made the film *"WALL-E"*, there are two candidate entities, which have the true answer *"Andrew Stanton"* and the negative answer *"James Cameron"*. The losses of two triplets are almost equal only in the knowledge embedding, and thus, it is difficult to discriminate them. However, in the concept subspace, we discover that *"WALL-E"* and *"Andrew Stanton"* both are very relevant to the *"Pixar"* concept, while *"James Cameron"* does not have this concept. In this way, both the query and the true answer lie in the *"Pixar"*-directed hyperplane, whereas the query and the negative answer do not co-occur in the corresponding associated concept hyperplane. Thus, the projected loss of the true answer could be much less than that of the false answer, which makes it easy for discrimination.

### 3.4. Objectives and training

We minimize a margin-based objective function as follows:

$$L = \sum_{(h,r,t)\in S} \sum_{(h',r,t')\in S'_{(h,r,t)}} [\gamma + f_r(h,t) - f_{r'}(h',t')]_+ \tag{7}$$

where $[x]_+$ denotes the positive part of $x$, $\gamma > 0$ is a margin hyper-parameter, and

$$S'_{(h,r,t)} = \{(h',r,t) \mid h' \in E\} \cup \{(h,r,t') \mid t' \in E\} \cup \{(h,r',t) \mid r' \in R\} \tag{8}$$

$S'_{(h,r,t)}$ is the set of corrupted triplets that are constructed according to the Bernoulli distribution introduced in [5]. It is composed of training triplets with the head or tail replaced by a random entity (but not both at the same time) or a relation replaced by a random relation.

The training procedure contains two steps: First, we pre-train the concept graph embedding model according to Eq. (3) and acquire entity vectors in concept space. Second, we apply these entity vectors to knowledge graph embedding to minimize the objective function over the training triplets. Both optimizations adopt the stochastic gradient descent method.

## 4. Experiments

### 4.1. Datasets

Our model is evaluated on benchmarked public knowledge graph datasets extracted from Wordnet and Freebase, denoted as WN18 and FB15K. We adopt the datasets provided by Microsoft Concept Graph [9] as the concept information. These datasets contain 5401,933 unique concepts, 12,551,613 unique instances, and 87,603,947 IsA relations. Because there are some entities that are currently not included in the concept graph, we remove entities not in the concept graph from WN18 and FB15K to construct additional datasets, denoted as WN18(filt.) and FB15K(filt.).

### 4.2. Knowledge graph completion

Knowledge graph completion aims at predicting the relations between the entities under supervision of the existing knowledge graph, which is the most important benchmarked task for knowledge graph embedding.

*Evaluation protocol.* For evaluation, we use the same ranking procedure as in TransE. For each test triplet $(h, r, t)$, the head $h$ (or the tail $t$) is removed and replaced by each entity $e$ in the knowledge graph in turn. Then, the scores of those corrupted triplets are computed by the score function $f_r(h, t)$. Finally, the rank of the correct triplet is stored after sorting these scores in ascending order. We report the average of the ranks, denoted as Mean Rank, and the proportion of test triplets ranked in the top 10 (as HITS@10) as evaluation metrics. The above is called the "Raw" setting. Additionally, the other setting is called "Filter". When some corrupted triplets end up being valid triplets, from the training set, for example, they can be ranked above the test triplet. However, these triplets are true and should not be counted as an error. Thus, in the "Filter" setting, we propose to remove entries from the list of corrupted triplets that exist in the training, validation or test set. In both settings, a higher HITS@10 and a lower Mean Rank mean better performance.

**Table 1**
Entity prediction results on FB15K and WN18.

| FB15K | Mean rank | | HITS@10 | |
|---|---|---|---|---|
| | Raw | Filt. | Raw | Filt. |
| TransE | 243 | 125 | 34.9 | 47.1 |
| TransH | 212 | 87 | 45.7 | 64.4 |
| DKRL(CNN) | 200 | 113 | 44.3 | 57.6 |
| Jointly(A-LSTM) | 167 | 73 | 52.9 | 75.5 |
| SSP(Joint.) | 188 | 85 | 53.5 | 77.1 |
| KEC | 180 | 81 | 53.9 | 77.3 |
| KEC(Con.) | **165** | **71** | **53.9** | **78.5** |
| WN18 | Mean rank | | HITS@10 | |
| | Raw | Filt. | Raw | Filt. |
| TransE | 263 | 251 | 75.4 | 89.2 |
| TransH | 318 | 303 | 75.4 | 86.7 |
| Jointly(A-LSTM) | **134** | **123** | 78.6 | 90.9 |
| SSP(Joint.) | 168 | 156 | 81.2 | 93.2 |
| KEC | 165 | 154 | 81.5 | 93.8 |
| KEC(Con.) | 158 | 143 | **82.8** | **94.3** |

**Table 2**
Entity prediction results on filtered datasets.

| FB15K(filt.) | Mean rank | | HITS@10 | |
|---|---|---|---|---|
| | Raw | Filt. | Raw | Filt. |
| SSP(Joint.) | 173 | 78 | 56.1 | 76.5 |
| KEC | 161 | 75 | 56.5 | 77.5 |
| KEC(Con.) | **126** | **50** | **60.6** | **80.0** |
| WN18(filt.) | Mean rank | | HITS@10 | |
| | Raw | Filt. | Raw | Filt. |
| SSP(Joint.) | 170 | 156 | 81.2 | 93.2 |
| KEC | 162 | 154 | 85.7 | 95.5 |
| KEC(Con.) | **134** | **121** | **87.8** | **95.6** |

*Baselines.* We train all baseline methods using the codes provided by the authors and select the best model using the parameters reported in their literatures.

*Implementation.* For the experiments of KEC, we have attempted several settings on the validation dataset to obtain get the best configuration. Under the "bern" sampling strategy, the optimal configurations are the following: (1) In concept graph embedding, embedding dimension $d = 100$, initial learning rate $\alpha = 0.025$, threads number $num\_threads = 12$, and iterations $iter = 5$, as the implementation details of skip gram model are similar to word2vec. (2) In knowledge graph embedding, embedding dimension $d = 100$, margin $\gamma = 1.7$, balance factor $\lambda = 0.3$ on FB15K, and embedding dimension $d = 100$, margin $\gamma = 5.0$, balance factor $\lambda = 0.3$ on WN18; initial learning rate $\alpha = 0.001$, which drops to its half every 2000 rounds for both datasets. And it stops learning when it converges, as the total loss stops decreasing.

To additionally utilize both the textual information and conceptual information, we use the concatenation of the output vector of KEC and output vector of SSP as the entity and relation representation, which is called the **Concatenation**(Con.) setting.

On FB15K and WN18, the embeddings in concept space of those entities not in the concept graph are initialized by averaging their head and tail entity embeddings in concept space based on symbolic triplets datasets.

*Results.* From the evaluation results shown in Tables 1–3, we can observe the following:

(1) Our KEC model outperforms the baseline methods, including TransE and TransH, significantly and consistently on all metrics, which demonstrates the effectiveness of our models and the correctness of our intuition analysis.

**Table 3**
Relation prediction results on FB15K.

| FB15K | Mean rank | | HITS@10 | |
|---|---|---|---|---|
| | Raw | Filt | Raw | Filt |
| TransE | 2.91 | 2.53 | 69.5 | 90.2 |
| TransH | 8.25 | 7.91 | 60.3 | 72.5 |
| DKRL(CNN) | 2.41 | 2.03 | 69.8 | 90.8 |
| SSP(Joint.) | 1.87 | 1.47 | 70.0 | 90.9 |
| KEC | 1.82 | 1.51 | 74.3 | 88.9 |
| KEC(Con.) | **1.35** | **1.02** | **78.8** | **90.9** |

(2) DKRL and SSP(Joint.) methods involve textual descriptions to improve the entity representations. Although SSP is also a hyperplane-based method, KEC adopts the hyperplane in a concept-specific way instead of a semantics-specific way. By focusing on the concept correlation between entities, our model outperforms them. This finding indicates that the concept information is more general and could be more beneficial to enrich knowledge embedding than textual information.

(3) On FB15K, our KEC(Con.) model outperforms Jointly (A-LSTM) model, although Jointly(A-LSTM) utilizes complicated deep neural nets and has a higher expense of computation. On WN18, KEC(Con.) is slightly worse than Jointly(A-LSTM) on the mean rank, but it achieves a better hits@10. The reason is most likely that there are some entities not involved in the current release of Microsoft Concept Graph, whose representations are not improved significantly and thus ranks high and hampers the mean rank. Additionally, our method obtains a more significant improvement on the filtered datasets where the entities with concepts are selected, which confirms this view.

(4) Because the concept and corpus are two valuable resources for entity embedding, we suggest using the two when an in-domain corpus is available; otherwise, we recommend our concept only model.

### 4.3. Entity classification

The task is a multi-label classification that aims to predict the entity types, which is crucial and widely used in many NLP tasks [24].

We adopt two datasets. One dataset, denoted as FB15K, is the same as DKRL, with 50 classes and 15K entities. The other dataset, denoted as Wiki13K, is extracted from the Wiki [25] dataset with 60 classes and 13K entities. In summary, this problem is a multi-label classification task, which means that for each entity, the method should provide a set of types rather than a single type.

*Evaluation protocol.* In the training process, similar to SSP, we use the concatenation of a conceptual vector and embedding vector $(c_e, e)$ as the entity representation, which is the feature for the front-end classifier. For a fair comparison, we also use Logistic Regression as the classifier and the one-versus-rest setting for multi-label classification, as in DKRL. For the subsequent evaluation we follow [26], and we apply the mean average precision (MAP).

*Implementation.* Because we use the same benchmarked datasets, we directly compare our models with the experimental results of several baselines reported in the literature. To obtain the best evaluation results, we train the model under the best configuration, following the above experiment, of the knowledge graph completion.

**Table 4**
Evaluation results on entity classification.

| Metrics | FB15K | Wiki13K |
|---|---|---|
| TransE | 87.8 | 86.6 |
| TransH | 88.2 | 87.2 |
| DKRL(CNN) | 90.1 | 89.5 |
| SSP(Joint.) | 94.4 | 90.8 |
| KEC | 94.8 | 91.5 |
| KEC(Con.) | **95.0** | **92.2** |

**Table 5**
Rank statistics of Link Prediction. Here, $r$ is the rank given by the corresponding models.

| | $KEC_{\#r\leq 100}$ |
|---|---|
| $TransE_{\#r\geq 500}$ | 610 |
| $TransE_{\#r\geq 1000}$ | 268 |
| $TransE_{\#r\geq 2000}$ | 81 |
| $TransE_{\#r\geq 3000}$ | 35 |

*Results.* Table 4 shows the evaluation results on entity classification. From this table, we can observe the following:

(1) In summary, our model KEC(Con.) achieves the state-of-the-art classification performance, which illustrates the capabilities of our model.

(2) The KEC model achieves a 8.0% and 7.5% improvement compared with TransE and TransH on FB15K dataset and 5.7% and 4.9% on Wiki13K dataset. As TransE and TransH only focus on the structural information, these improvements confirm the effectiveness of the concept integration.

(3) Compared to DKRL and SSP, the results show that incorporating the concept information into entity embeddings could contribute to entity classification more than textual information.

### 4.4. Concept relevance analysis

Modelling concept relevance is beneficial for correctly classifying the triplets that could not be discriminated by using only the structural information in the knowledge graph. To prove this aspect, we reported statistical results on link prediction in Table 5. The number in each cell means the number of triplets whose rank is larger than $m$ in TransE and less than $n$ in our model. Specifically, the triplet *(Moonraker, /award/award_nomination/nominated_for, For Your Eyes Only)* ranks 987 in TransE, while 4 in KEC. Looking into the concept graph, we discover that the two entities both have the *"bond movie"* concept. For another triplet, *(Simon Property Group, /organization/leadership/role, Chief Executive Officer)*, it ranks at 536 in TransE while 91 in KEC. The entity *"Simon Property Group"* has the concept *"mall reit"*, while the other entity has the concept *"management director"*. There exists the relation *"/organization/leadership/role"* between the two concepts. The statistical results show that many triplets benefit from the concept relevance offered by the concept graph. The experiments also demonstrate the effectiveness of our models and indicate the potential usage of the concept relevance.

### 4.5. Precise semantic expression analysis

As discussed above, precise semantic expression with concept information improves the performance of the discrimination. To justify this statement, we collected those triplets whose scores are slightly better than the golden triplets by TransE in link prediction (in other words, these are hard examples for TransE) as negative triplets, and then, we plotted the KEC score difference between each corresponding pair of the negative and golden triplets, as Fig. 5 shows. In the histogram, the $x$-axis indicates the score difference, where a larger value means better. The $y$-axis indicates the proportion of the corresponding triplet pairs. The right bars indicate that KEC makes a correct decision while TransE fails, and the left bars mean that both KEC and TransE fail. We can observe from the histogram that all of these triplets are predicted wrongly by TransE, but with the concept information, our model correctly distinguishes 82.5% of them. The experiments confirmed our intuition and demonstrated the effectiveness of our models.

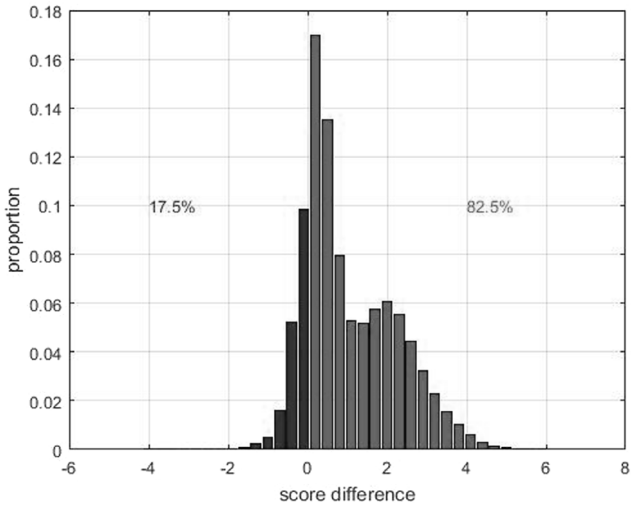


**Fig. 5.** Precise semantic expression analysis for KEC.

## 5. Conclusions

In this paper, we propose a Knowledge Graph Embedding with Concepts (KEC) model for representation learning of knowledge graphs with enhanced concept graph information. KEC could incorporate concept information into entity embeddings by characterizing semantic relatedness between concepts and entities. Concept information has a large impact on discovering concept relevance and then offering precise semantic expression. In our experiments, we evaluate our model on two tasks, including knowledge graph completion and entity classification. The experimental results show that our method achieves consistent and significant improvements compared to the state-of-the-art baselines.

Currently, our model is not based on deep learning but is instead simple and efficient. In the future, we will consider using deep learning methods to make improvements.

## Acknowledgements

This work was supported by National Key Research and Development Program of China (Grant No. 2016YFB1000902) and National Natural Science Foundation of China (Grant No. 61472040).

## References

[1] K.D. Bollacker, C. Evans, P. Paritosh, T. Sturge, J. Taylor, Freebase: A collaboratively created graph database for structuring human knowledge, in: Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008, 2008, pp. 1247–1250.

[2] G.A. Miller, WordNet: A lexical database for english, Commun. ACM 38 (11) (1995) 39–41.

[3] A. Bordes, N. Usunier, A. García-Durán, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational Data, 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, 2013, pp. 2787–2795.
[4] Y. Lin, Z. Liu, H. Luan, M. Sun, S. Rao, S. Liu, Modeling relation paths for representation learning of knowledge bases, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015, 2015, pp. 705–714.
[5] Z. Wang, J. Zhang, J. Feng, Z. Chen, Knowledge graph embedding by translating on hyperplanes, in: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada, 2014, pp. 1112–1119.
[6] R. Xie, Z. Liu, J. Jia, H. Luan, M. Sun, Representation learning of knowledge graphs with entity descriptions, in: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA, 2016, pp. 2659–2665.
[7] H. Xiao, M. Huang, L. Meng, X. Zhu, SSP: Semantic space projection for knowledge graph embedding with text descriptions, in: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA, 2017, pp. 3104–3110.
[8] J. Xu, K. Chen, X. Qiu, X. Huang, Knowledge graph representation with jointly structural and textual encoding, CoRR abs/1611.08661 (2016).
[9] J. Cheng, Z. Wang, J. Wen, J. Yan, Z. Chen, Contextual text understanding in distributional semantic space, in: Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19–23, 2015, 2015, pp. 133–142.
[10] A. Bordes, J. Weston, R. Collobert, Y. Bengio, Learning structured embeddings of knowledge bases, in: Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2011, San Francisco, California, USA, August 7–11, 2011, 2011.
[11] Y. Lin, Z. Liu, M. Sun, Y. Liu, X. Zhu, Learning entity and relation embeddings for knowledge graph completion, in: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25–30, 2015, Austin, Texas, USA, 2015, pp. 2181–2187.
[12] M. Nickel, V. Tresp, H. Kriegel, A three-way model for collective learning on multi-relational data, in: Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28–July 2, 2011, 2011, pp. 809–816.
[13] R. Jenatton, N.L. Roux, A. Bordes, G. Obozinski, A latent factor model for highly multi-relational data, in: Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3–6, 2012, Lake Tahoe, Nevada, United States, 2012, pp. 3176–3184.
[14] Z. Wang, J. Zhang, J. Feng, Z. Chen, Knowledge graph and text jointly embedding, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, 2014, pp. 1591–1601.
[15] H. Zhong, J. Zhang, Z. Wang, H. Wan, Z. Chen, Aligning knowledge and text embeddings by entity descriptions, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17–21, 2015, 2015, pp. 267–272.
[16] D. Zhang, B. Yuan, D. Wang, R. Liu, Joint semantic relevance learning with text data and graph knowledge, in: The Workshop on Continuous Vector Space MODELS & Their Compositionality, 2015, pp. 32–40.
[17] M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks, IEEE Trans. Signal Process. 45 (11) (1997) 2673–2681.
[18] A. Graves, J. Schmidhuber, Framewise phoneme classification with bidirectional LSTM and other neural network architectures, Neural Netw. 18 (5–6) (2005) 602–610.
[19] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, CoRR abs/1409.0473 (2014).
[20] Z. Hu, P. Huang, Y. Deng, Y. Gao, E.P. Xing, Entity hierarchy embedding, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers, 2015, pp. 1292–1300.
[21] Y. Li, R. Zheng, T. Tian, Z. Hu, R. Iyer, K.P. Sycara, Joint embedding of hierarchical categories and entities for concept categorization and dataless classification, 2016, pp. 2678–2688.
[22] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, 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, 2013, pp. 3111–3119.
[23] Y. Liu, Z. Liu, T. Chua, M. Sun, Topical word embeddings, in: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25–30, 2015, Austin, Texas, USA, 2015, pp. 2418–2424.
[24] A. Neelakantan, M. Chang, Inferring missing entity type instances for knowledge base completion: New dataset and methods, in: NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31–June 5, 2015, 2015, pp. 515–525.
[25] A. Abhishek, FgER: Fine-grained entity recognition, in: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, Louisiana, USA, February 2–7, 2018, 2018.
[26] K. Chang, W. Yih, C. Meek, Multi-Relational latent semantic analysis, in: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, a Meeting of SIGDAT, a Special Interest Group of the ACL, 2013, pp. 1602–1612.