Composing Knowledge Graph Embeddings via Word Embeddings

Lianbo Ma 1, Peng Sun 1*, Zhiwei Lin 2, Hui Wang 2

1 College of Software, Northeastern University, Shenyang, China
2 School of Computing, Ulster University, Newtownabbey BT37 0QB, U.K

Abstract

Learning knowledge graph embedding from an existing knowledge graph is very important to knowledge graph completion. For a fact \((h, r, t)\) with the head entity \(h\) having a relation \(r\) with the tail entity \(t\), the current approaches aim to learn low dimensional representations \((h, r, t)\), each of which corresponds to the elements in \((h, r, t)\), respectively. As \((h, r, t)\) is learned from the existing facts within a knowledge graph, these representations can not be used to detect unknown facts (if the entities or relations never occur in the knowledge graph).

This paper proposes a new approach called TransW, aiming to go beyond the current work by composing knowledge graph embeddings using word embeddings. Given the fact that an entity or a relation contains one or more words (quite often), it is sensible to learn a mapping function from word embedding spaces to knowledge embedding spaces, which shows how entities are constructed using human words. More importantly, composing knowledge embeddings using word embeddings makes it possible to deal with the emerging new facts (either new entities or relations). Experimental results using three public datasets show the consistency and outperformance of the proposed TransW.

Introduction

A knowledge graph \(K = (E, R)\) where \(E\) is a set of entities and \(R\) is a set of relations, contains linked information of facts, where each fact is a triple \((h, r, t)\) showing the relationship \(r \in R\) between the head entity \(h \in E\) and the tail entity \(t \in E\). However, with the growing volume of data on the Internet, the new facts emerge constantly and they need to be added into the existing knowledge graphs in order to complete the graphs. As such, knowledge graphs are far from complete. Adding the new facts manually is labor intensive and also makes it difficult to validate if the new fact should belong to the knowledge graph. One way around this is to learn knowledge graph embeddings for the entities and their relations (i.e., encoding both entities and their relations within facts into a continuous low-dimensional vector space) (Bordes et al. 2013; Cai, Zheng, and Chang 2018; Trouillon et al. 2017; Wang, Li, and Pan 2018). For a triple \((h, r, t)\), let \((h, r, t)\) be its representation in knowledge graph embedding, where \(h, t, r \in \mathbb{R}^n\). The existing approaches aim to train a model to ‘translate’ the head \(h\) to the tail entity \(t\) using the relation \(r\) with minimum loss, such as \(h + r \approx t\) in the TransE (Bordes et al. 2013) model.

However, as the representations of \((h, r, t)\) in knowledge graph embeddings are trained with the known entities from \(E\) and known relations from \(R\), the learned representations have critical issues for dealing with ‘unknown’ new entities and new relations (i.e., a new fact \((h, r, t)\) where \(h \not\in E\) or \(t \not\in E\) or \(r \not\in R\)). For example, Fig. 1 shows a sub-graph of the knowledge graph extracted from FB15K dataset (Bordes et al. 2013), in which there are 4 relations between “Agent 007” and “Daniel Craig”. There may be more than 4 relations between the two entities and the new relations may not exist in \(R\). Also, the title of the new “007” film (beyond 2020) is still new, and it would not exist in \(E\). Therefore, the current approaches can not deal with the unseen entities and relations.

With the fast growing volume of data on the Internet, adding new entities and relations is very important in scaling out knowledge graphs for enrichment. The current knowledge graph embedding approaches learn \((h, r, t)\) for a specific fact \((h, r, t)\) by ignoring the detail of the words within the facts. Such limitation makes them very difficult to generalize to known facts. This paper presents a new approach called TransW, which aims to address these issues by learning knowledge graph embeddings via the composition of word embeddings due to the fact that each entity or relation may contain multiple words. To the best of our knowledge, this is the first work aiming to enrich a knowledge graph by detecting unknown entities and unknown relations, using word embeddings.

From Fig. 1 it is very obvious that the multiple relations between two entities are very similar. For example, “/film/film/starring” and “/film/performance/character” are very similar as (1) on one hand, they all share the keyword “film”, and (2) on the other hand, “starring” is very close to “performance/character”. Each word can be represented in word embeddings and the words with semantic similarity tend to have very high similarity between their word embeddings. The similarity between two entities (or two relations)
Figure 1: An example of a sub-graph of the knowledge graph from FB15K dataset, where multiple relations exist between two entities

is measurable if a mapping function from word embedding to knowledge graph embedding can be defined. Let $\mathcal{W}$ be the word embedding space and $\mathcal{K}$ be the knowledge embedding space, and $g : \mathcal{W} \rightarrow \mathcal{K}$ be the mapping function from word embedding space to knowledge embedding space, the aim of this work is to learn such mapping function, so that for any new entities or new relations, their representations in knowledge graph embeddings can be constructed via this mapping function. This will be much flexible in dealing with the unknown entities and relations.

The contributions of this paper include (1) proposing to use word embeddings as the ingredients for learning knowledge graph embeddings; (2) introducing a new approach called TransW in order to create mappings from word embeddings to knowledge graph embeddings; (3) conducting experiments to show the proposed approach can deal with unknown facts.

**Related Work**

**Translation-based Models**

For a triplet $(h, r, t)$, the TransE [Bordes et al. 2013] treats the relation $r$ as a translation from a head entity $h$ to a tail entity $t$. Here, $(h + r)$ is close to $(t)$ and the score function is

$$f_r(h, t) = -\|h + r - t\|_2^2. \tag{1}$$

Despite its success in knowledge graphs with many 1-1 relations, it is not suitable for $1 - N, N - 1$ and $N - M$ relations. To address these issues, the TransH [Wang et al. 2014] transforms the entities using a norm vector $w_r$:

$$h_\perp = h - w_r^T h w_r \tag{2}$$
$$t_\perp = t - w_r^T t w_r. \tag{3}$$

before using Eq. (1).

The TransR [Lin et al. 2015b] defines a transformation matrix $M_r$ and its score function as

$$\|M_r h + r - M_r t\|_2^2 \tag{4}$$

The TransD [Ji et al. 2015] constructs a dynamic mapping matrix for each entity-relation pair. TransSparse [Ji et al. 2016] uses sparse matrices to model the relations. Similar approaches also involve TransM (Fan et al. 2014), TransG (Xiao et al. 2015) and ManifoldE (Xiao, Huang, and Zhu 2016). Recently, some new methods aim to deal with extra information, such as relation-type (Wang, Wang, and Guo 2015), paths with different confidence levels (PTransE) (Lin et al. 2015a), data uncertainty (KG2E) (He et al. 2015), and semantic smoothness of embedding space (Guo et al. 2017). Besides, several recent TransE variants such as TorusE (Ebisu and Ichise 2018) and ConvKB (Nguyen et al. 2018) also achieve the state-of-the-art performance.

**Other Models**

The structured embedding (Bordes et al. 2011) uses two relation-specific matrices $(M_{h,r}$ and $M_{t,r})$ for projecting head and tail entities, respectively, and defines a new score function as $f_r(h, t) = \|M_{h,r} h - M_{t,r} t\|_2$. Unstructured Model (UM) model (Bordes et al. 2012) is a naive version of TransE, where relation information is ignored and the score function is reduced to $f_r(h, t) = \|h - t\|_2^2$. Single Layer Model (SLM) (Socher et al. 2013) constructs a nonlinear neural network to represent the score function, which is defined as $f_r(h, t) = u_r^T g(M_{h,r} h + M_{t,r} t)$ where $M_{r1}$ and $M_{r2}$ are relation-specific weight matrices. Semantic Matching Energy (SME) (Glorot et al. 2013) aims to capture correlations between entities and relations via multiple matrix products and Hadamard product, defined as $f_r(h, t) = (M_1 h + M_2 r + b_1)^T (M_3 t + M_4 r + b_2)$ and $f_r(h, t) = (M_1 h \otimes M_2 r + b_1)^T (M_3 t \otimes M_4 r + b_2)$, where $M_1$, $M_2$, $M_3$ and $M_4$ are weight matrices, $\otimes$ is the Hadamard product, $b_1$ and $b_2$ are bias vectors. Latent Factor Model (LFM) (Jenatton et al. 2012) incorporates second-order correlations between entities using a quadratic form, and defines a bilinear score function as $f_r(h, t) = h^T W_r t$. Neural Tensor Network (NTN) (Socher et al. 2013) defines...
Table 1: Different translation-based models: the scoring functions $f_r(h, t)$ and the number of parameters. $n_e$ and $n_r$ are the number of unique entities and relations, respectively. $k$ is the dimension of embedding space. $n$ and $m$ are the number of words of each entity and relation.

| Model       | Score function $f_r(h, t)$                                                                 | Parameters                          |
|-------------|-------------------------------------------------------------------------------------------|-------------------------------------|
| TransE      | $\|h + r - t\|_2^2$                                                                     | $O(n, k + n_r k)$                    |
| TransH      | $(h - w_t^\top w_r + d_r - (t - w_r^\top tw_r))_2^2$                                    | $O(n_r k + 2n_r k)$                 |
| TransR      | $\|M_r h + r - M_r t\|_2^2$                                                             | $O(n_r k + n_r k)$ + $k$             |
| TransD      | $(w_r w_t^\top + 1) h + r - (w_r w_t^\top + 1) t)_2^2$                                 | $O(2n_r k + 2n_r k)$                |
| TranSparse  | $(M_r (\theta_r) h + r - M_r (\theta_r) t)_2^2$                                        | $O(n_r k + n_r (k + 1) k)$          |
| TransW      | $(\sum h_i \otimes w_{hi} + b_h) + \sum r_i \otimes w_{ri} - (\sum t_i \otimes w_{ti} + b_t))_2^2$ | $O((n + 1)n_r k + (m + 1)n_r k)$    |

an expressive score function to extend the SLM model, i.e., $f_r(h, t) = u_r^\top \tanh(h^\top W_r t + W_{r,1} h + W_{r,2} t + b_r)$, where $u_r$ is a relation-specific linear layer, $W \in \mathbb{R}^{d \times d \times d \times k}$ is a 3-way tensor. In addition of the above approaches, we will also compare with another common model RESCAL in the experiments, which is a collective matrix factorization model (Nickel, Tresp, and Kriegel 2011). More recently, adversarial learning has also been investigated for learning knowledge graph embeddings (Cai and Wang 2018; Wang, Li, and Pan 2018).

The TransW Approach

As the current knowledge embedding approaches ignore the fine-grained semantic information in the word space, and may be insufficient to deal with new facts with unknown entities or relations, this section presents the TransW approach which models entities and relations via word embeddings.

A motivating example

To begin with, using Fig. 1 as an example graph, the current approach to learn $(h, r, t)$ for the triple (“Agent 007”, “/film/film/starring”, “Casino Royale”), does not have semantic knowledge of how entity “Agent 007” is linked with “Casino Royale”. The vector $h$, $r$, and $t$ are only the vector representations without any meaning. This makes it unhelpful in predicting “/film/performance/character” as a relation between “Agent 007” and “Casino Royale”, even if “/film/performance/character” is semantically related to “/film/film/starring”.

Suppose in the word embedding space, if the embeddings for “film”, “performance”, “character” and “starring” are $h_f$, $h_p$, $h_c$, and $h_s$, by using a linear mapping function $g$ to combine the word embeddings for the relations:

$$g(h_f, h_s) \text{ for } /\text{film/film/starring}$$

$$g(h_f, h_p, h_c) \text{ for } /\text{film/film/starring}$$

it would be much easier to show the similarity between two relations in knowledge graph spaces. This will also make it possible to detect unknown facts in order to scale out a knowledge graph for enrichment.

TransW

In TransW, each entity or relation is represented in the form of linear combination of word embeddings. For a triple $(h, r, t)$ and its embedding $(h, r, t)$, suppose the numbers of words in $h$, $r$ and $t$ are $n$, $p$ and $m$ respectively, then $(h, r, t)$ can be represented with their words embeddings:

$$h = \sum_{i=0}^{n} h_i \otimes w_{hi} + b_h$$

$$t = \sum_{i=0}^{p} t_i \otimes w_{ti} + b_t$$

$$r = \sum_{i=0}^{m} r_i \otimes w_{ri} + b_r$$

where $h_i$, $r_i$, $t_i \in \mathbb{W}$ are the word embedding for the $i$-th word in the corresponding $h$, $r$ and $t$, respectively, $\otimes$ denotes Hadamard product, $w_{hi}$, $w_{ri}$, and $w_{ti}$ are the $i$-th connection vector for $h$, $r$ and $t$, respectively, $b_h$, $b_r$, $b_r \in \mathbb{R}^k$ are the bias parameters for entities $h$ and $t$, and relation $r$ respectively.

Similar to that of TransE, Eq. [1] is used as the score function for TransE. The score is expected to be lower for a golden triplet and higher for an incorrect triplet. Let $\Delta$ be the set of all triplets which are valid and $\Delta'$ be the set of all triplets which are not valid, the loss function is defined according to the translation approach:

$$L = \sum_{\xi \in \Delta} \sum_{\xi' \in \Delta'} [\gamma + f_r(\xi') - f_r(\xi)]_+$$

where $\gamma$ is margin.

Remark: From Eq. [5], it is obvious that each word embedding is assigned with a unique connection vector and an entity or relation is the sum of all transformed word embeddings. In this way, TransW represents each entity or relation by a unique combination of word embeddings. As illustrated in Fig. [2], given the fact about “/film/film/starring”, TransW is able to predict a new relation “/film/performance/character” according to the word-level semantic similarity between the two relations.
Table 2: Summary of the public datasets used in the experiments. Rel/Ent/Train/Valid/Test are the number of relations/entities/training triples/validation triples/testing triples.

| Dataset | Rel | Ent | Train | Valid | Test |
|---------|-----|-----|-------|-------|------|
| WN11    | 11  | 38,696 | 112,581 | 2,609 | 10,544 |
| FB13    | 13  | 75,043 | 316,232 | 5,908 | 23,733 |
| FB15K   | 1345 | 14951 | 483142 | 50000 | 59071 |

**Experiments**

This paper uses word embeddings in GloVe word embedding glove.6B.100d ([Pennington, Socher, and Manning 2014](#)). As this paper assumes the dimension of knowledge graph embeddings is 100, the word embeddings with dimension of 100 are used. The learning rate is 0.01.

Three public benchmark datasets, as summarized in Table 2, are used to evaluate both TransW. Usually, the knowledge graph embeddings are evaluated using link prediction and triple classification. In link prediction and triple classification, they assume the prediction is based on the pre-defined set of entities $E$ and the pre-defined set of relations $R$. The existing work, such as TransE, etc, can not be used to decide if a new fact $(h, r, t)$ belongs to a knowledge graph when the new fact contains either an unseen entity $h \notin E$ or $t \notin E$ or an unseen relation $r \notin R$. Therefore, in addition to link prediction and triple classification, this paper sets out a new task for detecting unknown facts with “new relations”.

As such, this section contains three tasks for evaluating the proposed TransW:

1. Detecting unknown facts: the relations in the test dataset are unknown (do not occur in the training part);
2. Link prediction: the relations and entities in the test dataset already exist in the training set;
3. Triple classification: the relations and entities in the test dataset already exist in the training set.

**Detecting Unknown Facts**

The first part of the experiments carried out is to detect unknown facts. Different from the existing approaches without using word embeddings, the TransW approach composes knowledge graph embeddings $(h, r, t)$ using word embeddings, which makes it possible to detect new facts $(h, r, t)$, where “unseen” entities $(h \notin E$ or $t \notin E)$ or an “unseen” relation $(r \notin R)$ exists. In this experiment, the primary focus will be evaluating the detection of new relations using FB15K dataset. The reason of not using WN11 and FB13 is that they have very limited number of relations (as shown in Table 2) there are only 11 relations in WN11 and 13 relations in FB13), which makes it hard to train a model using just 10 relations in order to make accurate predication for the remaining relations (if 10-fold cross validation is used).

This evaluation is conducted using 10-fold cross validation and the 1345 relations in FB15K are split into 10 folds, from which 9 folds are used to train a model and leave the relations in the remaining fold as “unseen” relations. This setting will make sure that the unseen facts will not only contain “unseen” relations ($r \notin R$) but may also contain “unseen” entities. The hypothesis is that the TransW models are able to tell if the “unseen” facts are part of the existing knowledge graph or not.

Table 3 shows the statistical information of each fold. For example, in the 1st fold, there are 1210 relations and 444,422 facts in total for training. There are also 77436 facts and 135 relations in the test set. However, as the numbers of facts for the relations in the test set are imbalanced (e.g., some relations in the test set only contains very few facts), for each relation, only 5000 facts are randomly selected for testing. This is repeated for 10 times in order to make sure if TransW can provide consistent results. The accuracy in the table is based on the average of the accuracies of these 10 times of testing subsets.

After training, a threshold $\sigma$ for determining if a relation is valid or not is set according to the score function of Eq. 1 using the training set in each fold. If the score (using Eq. 1) for a fact in the testing set of each fold is lower than $\sigma$, this fact is regarded as a true fact, otherwise, the fact is not a valid fact to the knowledge graph. The threshold $\sigma$ is shown in Table 3 for each fold, which is around 0.0863.

From Table 3, it is found that TransW performs very consistently in each fold, and the average accuracy for detecting unknown facts is very satisfying.

**Link Prediction**

Link prediction is to locate entities from $E$ for the following two cases:

1. $(?, r, t)$ when the head entity is missing;
2. $(h, r, ?)$ when the tail entity is missing;

where $h, t \in E$ and $r \in R$, in order to complete a triplet.

This experiment follows the settings in ([Bordes et al. 2011](#)) ([Bordes et al. 2013](#)), using WN11 and FB13 datasets.

In order to carry out the experiments, for each triple $(h, r, t)$ in the test set, a new set $T$ of data is created:

$$T = \{(e, r, t) | \forall e \in E\} \cup \{(h, r, e) | \forall e \in E\}.$$
Table 3: Prediction accuracy for unseen facts of FB15K using 10-fold cross validation.

| Fold | Train(facts/relations) | Test(facts/relations) | Average (%) | Bias(%) | σ |
|------|-------------------------|-----------------------|-------------|---------|---|
| 1    | 444422/1210             | 77436/135             | 61.18       | (-1.9,+1.14) | 0.0791 |
| 2    | 437977/1211             | 10942/134             | 54.62       | (-1.22,+0.78) | 0.0985 |
| 3    | 455313/1210             | 50144/135             | 55.07       | (-0.53,+0.53) | 0.0705 |
| 4    | 424743/1210             | 14430/135             | 54.54       | (-1.42,+0) | 0.0768 |
| 5    | 416196/1210             | 16322/135             | 52.4        | (-1.5,+1.0) | 0.0961 |
| 6    | 429286/1211             | 12952/134             | 55.76       | (-0.14,+0.29) | 0.0993 |
| 7    | 404590/1210             | 156120/135            | 53.82       | (-1.12,+0.84) | 0.0806 |
| 8    | 447139/1211             | 8652/134              | 55.63       | (-0.23,+0.27) | 0.0723 |
| 9    | 442557/1211             | 9955/134              | 55.27       | (-0.19,+0.45) | 0.1057 |
| 10   | 446054/1211             | 9168/134              | 56.69       | (-0.19,+0.41) | 0.0845 |
|      | Average                |                      | 55.5        | (-0.84,+0.48) | -  |

Table 4: Results on FB13 & WN11 for link prediction (%)

| Metric | FB13 (HIT@10) | FB13 (HIT@3) | FB13 (HIT@1) | WN11 (HIT@10) | WN11 (HIT@3) | WN11 (HIT@1) |
|--------|---------------|---------------|---------------|---------------|---------------|---------------|
| Rescal | 23.86         | 24.59         | 18.86         | 19.80         | 5.70          | 6.92          |
| TransE | 32.65         | 32.75         | 24.86         | 25.16         | 17.28         | 17.57         |
| TransR | 26.83         | 27.53         | 20.34         | 20.94         | 14.04         | 14.97         |
| TransH | 25.02         | 25.80         | 18.03         | 18.78         | 6.62          | 7.52          |
| TransD | 34.77         | 35.22         | 26.75         | 27.77         | 18.47         | 19.85         |
| DistMult | 16.72       | 17.82         | 14.12         | 14.80         | 6.27          | 7.03          |
| Complex | 16.51        | 17.60         | 14.21         | 14.80         | 6.80          | 7.53          |
| TransW | **45.12**     | **47.93**     | **35.71**     | **35.94**     | **29.77**     | **30.62**     |

Table 5: Accuracy for triple classification on FB13 and WN11

| Model            | FB13 (%) | WN11 (%) |
|------------------|----------|----------|
| SE (Bordes et al. 2011) | 75.2     | 53.0     |
| SME (Glorot et al. 2013) | 63.7     | 70.0     |
| SLM (Socher et al. 2013) | 85.3     | 69.9     |
| LFM (Jenatton et al. 2012) | 84.3     | 73.8     |
| NTN (Socher et al. 2013) | 87.1     | 70.4     |
| Rescal (Nickel, Tresp, and Kriegel 2011) | 70.7     | 61.0     |
| TransE (Bordes et al. 2013) | 78.2     | 75.7     |
| TransH (Wang et al. 2014) | 77.4     | 77.7     |
| TransR (Lin et al. 2015) | 82.5     | 85.5     |
| TransD (Ji et al. 2015) | 87.7     | 86.4     |
| DistMult (Yang et al. 2015) | 55.3     | 50.0     |
| Complex (Trouillon et al. 2016) | 56.2     | 60.0     |
| TranSparse (Ji et al. 2016) | 86.7     | 86.3     |
| ManifoldE (Xiao, Huang, and Zhu 2016) | 87.2     | **87.5** |
| TransW (this paper) | **87.5** | 81.1     |

Scores are calculated according to the score function in Eq. 1 for the “true” and “false” triples in T. The triples related to (? , r, t) will be ranked in descending order based on their scores. This is repeated for (h, r, ?).

To compare with the existing approaches, this paper uses the HITS@N approach, which calculates the proportion of correct predictions in the top N facts from the ranking lists. This means, the higher HITS@N, the better performance it has. In Table 4, this paper reports the results for HITS@10, HITS@3 and HITS@1 using FB13 and WN11 datasets.

From Table 4, TransW is significantly better than the state of the art approaches for FB13 dataset. Similar to FB15K, the FB13 comes from Freebase. As shown in Fig. 1, the relations between entities has very rich context with multiple words. The composition of knowledge graph embeddings using word embeddings makes use of the semantic information in word embeddings and as such the performance is significantly improved.

Whilst in WN11 dataset, which is from the WordNet, each
entity in WN11 is a word and the relations between entities do not have rich semantic information. Therefore, the results are not as good as what TransW achieves in FB13.

**Triple Classification**

Triple classification is to evaluate if a triple \((h, r, t)\) is a correct fact in a knowledge graph. False facts are also added to the 3 datasets, according to the settings in \([\text{Socher et al.} 2013] \) and \([\text{Wang et al.} 2014] \). Therefore, this triple classification is a binary classification with both “true” and “false” facts.

A relation-specific threshold \(\sigma_r\), is chosen based on the best classification accuracies on the validation dataset. For a triple \((h, r, t)\), if its score of \(f_r(h, t) \leq \sigma_r\), then \((h, r, t)\) is classified as positive, otherwise negative.

The result for TransW in FB13 is better than most of the existing approaches though it is 0.2% lower than TransD. TransW has a satisfying accuracy compared to the original TransE in WN11. The results for triple classification confirms the finding in the experiment for link predication that utilizing the semantic information would help to enrich knowledge graph embedding.

**Conclusions and Future Work**

This paper introduces a new approach, called TransW, which encodes entities and relations into a low dimensional space via word embeddings. Using such composition with word embeddings, the word-level semantic information hidden in the word embedding spaces can be utilized for detecting unknown facts, where entities or relations never occur in the existing knowledge graphs. Experimental results using public datasets validate the hypothesis. Future work includes 1) defining a better representation for entities and relations via word embeddings; 2) testing the approach using various score functions in order to improve the accuracy for detecting unknown facts.

**References**

[Bordes et al. 2011] Bordes, A.; Weston, J.; Collobert, R.; and Bengio, Y. 2011. Learning structured embeddings of knowledge bases. In Burgard, W., and Roth, D., eds., Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2011, San Francisco, California, USA, August 7-11, 2011. AAAI Press.

[Bordes et al. 2012] Bordes, A.; Glorot, X.; Weston, J.; and Bengio, Y. 2012. Joint learning of words and meaning representations for open-text semantic parsing. In Lawrence, N. D., and Girolami, M. A., eds., Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2012, La Palma, Canary Islands, Spain, April 21-23, 2012, volume 22 of JMLR Proceedings, 127–135. JMLR.org.

[Bordes et al. 2013] Bordes, A.; Usunier, N.; García-Durán, A.; Weston, J.; and Yakhnenko, O. 2013. Translating embeddings for modeling multi-relational data. In Burges et al. (2013). 2787–2795.

[Bordes et al. 2012] Bordes, A.; Usunier, N.; García-Durán, A.; Weston, J.; and Yakhnenko, O. 2013. Translating embeddings for modeling multi-relational data. In Burges et al. (2013). 2787–2795.

[Burgos et al. 2013] Burgos, B.; Bottou, L.; Ghahramani, Z.; and Weinberger, K. Q., eds. 2013. 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.

[Cai et al. 2018] Cai, L., and Wang, W. Y. 2018. KBGAN: adversarial learning for knowledge graph embeddings. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers), 1470–1480.

[Cai et al. 2018] Cai, H.; Zheng, V. W.; and Chang, K. C. 2018. A comprehensive survey of graph embedding: Problems, techniques, and applications. IEEE Transactions on Knowledge and Data Engineering 30(9):1616–1637.

[2018] Ebisu, T., and Ichise, R. 2018. Toruse: Knowledge graph embedding on a lie group. In McIlraith and Weinberger (2018), 1819–1826.

[2014] Fan, M.; Zhou, Q.; Chang, E.; and Zheng, T. F. 2014. Transition-based knowledge graph embedding with relational mapping properties. In Aroonmanakun, W.; Boonkwam, P.; and Supnithi, T., eds., Proceedings of the 28th Pacific Asia Conference on Language, Information and Computation, PACLIC 28, Cape Panwa Hotel, Phuket, Thailand, December 12-14, 2014, 328–337. The PACLIC 28 Organizing Committee and PACLIC Steering Committee / ACL / Department of Linguistics, Faculty of Arts, Chulalongkorn University.

[2013] Glorot, X.; Bordes, A.; Weston, J.; and Bengio, Y. 2013. A semantic matching energy function for learning with multi-relational data. In Bengio, Y., and LeCun, Y., eds., 1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings.

[2017] Guo, S.; Wang, Q.; Wang, B.; Wang, L.; and Guo, L. 2017. SSE: semantically smooth embedding for knowledge graphs. IEEE Trans. Knowl. Data Eng. 29(4):884–897.

[2015] He, S.; Liu, K.; Ji, G.; and Zhao, J. 2015. Learning to represent knowledge graphs with gaussian embedding. In Bailey, J.; Moffat, A.; Aggarwal, C. C.; de Rijke, M.; Kumar, R.; Murdock, V.; Selliis, T. K.; and Yu, J. X., eds., Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015, 623–632. ACM.

[2012] Jenatton, R.; Roux, N. L.; Bordes, A.; and Obozinski, G. 2012. A latent factor model for highly multi-relational data. In Bartlett, P. L.; Pereira, F. C. N.; Burges, C. J. C.; Bottou, L.; and Weinberger, K. Q., eds., 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., 3176–3184.

[2015] Ji, G.; He, S.; Xu, L.; Liu, K.; and Zhao, J. 2015. Knowledge graph embedding via dynamic mapping matrix. In Proceedings of the 53rd Annual Meeting of the Asso-
