GTransE: Generalizing Translation-Based Model on Uncertain Knowledge Graph Embedding

Natthawut Kertkeidkachorn[1(✉)], Xin Liu[1], and Ryutaro Ichise[1,2]

[1] National Institute of Advanced Industrial Science and Technology, Tokyo, Japan
{n.kertkeidkachorn,xin.liu}@aist.go.jp
[2] National Institute of Informatics, Tokyo, Japan
ichise@nii.ac.jp

Abstract. This is an extension from a selected paper from JSAI2019. Knowledge graphs are useful for many AI applications. Many recent studies have been focused on learning numerical representations of a knowledge graph in a low-dimensional vector space. Learning representations benefits the deep learning framework for encoding real-world knowledge. However, most of the studies do not consider uncertain knowledge graphs. Uncertain knowledge graphs, e.g., NELL, are valuable because they can express the likelihood of triples. In this study, we proposed a novel loss function for translation-based models, GTransE, to deal with uncertainty on knowledge graphs. Experimental results show that GTransE can robustly learn representations on uncertain knowledge graphs.

Keywords: Knowledge graph · Uncertainty · Knowledge representation

1 Introduction

A knowledge graph (KG) stores a set of triples. A triple (h, r, t), where h and t are entities and r is a relation directed from h to t, expresses a fact. For example, $(Tokyo,\ capital\_of,\ Japan)$ describes that Tokyo is the capital city of Japan. Currently, there are many available KGs, e.g. DBpedia, YAGO, and Freebase.

To deal with vast knowledge, automatic knowledge graph construction approaches [2,9] are introduced. Automated construction process often results in noisy KG containing inaccurate facts, whose validity can only be expressed by confidence. Consequently, we refer to such KG as an uncertain KG. An uncertain KG stores the facts with their confidence.

Learning representation of KG, known as KG embedding, aims to capture latent representations of entities and the relations. One of the popular models is the translation-based model. The translation-based models learn representations by computing a dissimilarity score of triples [1]. Nonetheless, the translation-based models are optimized by the margin loss function, where uncertainty is

Y. Ohsawa et al. (Eds.): JSAI 2019, AISC 1128, pp. 170–178, 2020.
https://doi.org/10.1007/978-3-030-39878-1_16

ignored. In uncertain KGs, reliability of a fact is represented by confidence. Ignoring such confidence may result in low-quality representations.

In this paper, we therefore introduce a novel approach for learning the representation of uncertain KGs by generalizing the translation-based model for knowledge graph embedding.

2 Problem Definition

In this section, we define the learning representation on uncertain KGs.

Definition 1. *An Uncertain KG is a directed graph denoted by $G = (E, R, Q)$, where E, R, and Q are the entity set, relation set, and fact set, respectively. A fact is represented by a quadruple $q = (h, r, t, s)$, where $h, t \in E$, $r \in R$, and $s \in \mathbb{R}_{[0,1]}$. It indicates that entities h and t are connected by a relation r with confident score s. A higher s means it is more likely h and t are connected by r.*

Definition 2. *Given an uncertain KG, the problem of learning representation is to learn numerical representations of an entity $\mathbf{e} \in \mathbb{R}^K$ for each $e \in E$ and a relation $\mathbf{r} \in \mathbb{R}^K$ for each $r \in R$ such that for each $(h, r, t, s) \in Q$; $L_A(h, r, t) \propto s$, where L_A is the loss function with respect to a learning algorithm A., i.e. the facts can be preserved in $\mathbb{R}^K$ while considering the score s.*

3 Related Work

KG embedding is classified into three models: tensor-factorization-based, neural-network-based and translation-based [5]. In this study, we mainly focused on a translation-based model.

The translation-based models learn embedding representations by considering the relation r from the head entity h to the tail entity t as distance or dissimilarity. They commonly use a margin-based loss function. Inspired by the skip-gram model [13], where surrounded words are used to learn the representation of the word, TransE [1] translates the triple (h, r, t) as $\mathbf{h} + \mathbf{r} = \mathbf{t}$. With this translation, it can capture the first-order rules. TransH [14] extends TransE by projecting entities on a hyperplane to a specific relation space in order to select embedding components for each relation. TransR [12] further generalizes TransH by using a linear transformation to map between relations and corresponding entities. TransD [7] improves on TransR by employing two vectors for representing an entity and a relation. TransG [15] addresses multiple relation semantics by introducing multiple vectors. TransA [8] introduce a proper way to find a suitable margin. pTransE [11] considers relation paths by taking the sum of relations on the path when computing the score. CKRL [16] improves TransE by computing triple scores on KG as the penalty when learning representations. TorusE [5] solves the regularization problem of TransE by embedding entities and relations into a torus space.

In an uncertain KG, the level of validity of a fact is expressed by a confidence s. In practice, we can omit the confidence of the facts and directly learn the embedding as the traditional methods. Nevertheless, without confidence as an indicator, noisy facts could degrade the quality of the embedding representations. So far, there is a study on uncertain knowledge graph embedding [3]. Nevertheless, the translation-based model for uncertain KGs has not been studied yet.

4 Generalizing Translation-Based Model

The translation-based model learns embeddings by computing a score representing the distance of a triple (h, r, t). In TransE, the main assumption is that $\mathbf{h} + \mathbf{r} = \mathbf{t}$, if $(h, r, t) \in T$, where T is a triple set in KG. To measure the fitness of an embedding, a scoring function $f(h, r, t)$ is computed. The main difference between the different translation-based model approaches is in how to compute the scoring function. In TransE, the scoring function f(h,r,t) is measured using the L-1 or L-2 norm of $\mathbf{h} + \mathbf{r} - \mathbf{t}$.

Because of its score function, TransE could learn a trivial solution, where all entity and relation embeddings are equal. To better learn the embeddings, negative examples are needed. KG usually stores only positive examples. To generate negative examples, TransE uses a negative sampling method that corrupts the head or tail entity of a positive triple by a random entity. The negative sampling method defines a set of negative triples as $T'_{(h,r,t)} = \{(h', r, t) | (h, r, t) \in T, h' \in E, h' \neq h\} \cup \{(h, r, t') | (h, r, t) \in T, t' \in E, t' \neq t\}$.

Like other translation-based models, TransE uses a margin-based loss function to learn the embedding representations. The loss function is as follows:

$$L = \sum_{(h,r,t) \in T} \sum_{(h',r,t') \in T'} [f(h, r, t) - f(h', r, t') + M]_+, \quad (1)$$

where $[x]_+$ is the positive part of x, $f(\cdot)$ is a score function, M is a margin, and (h', r, t') is a negative sample in T'. In TransE, the confidence s is not introduced to the method yet.

The generalized translation-based model (GTransE) improves the margin-based loss function in order to support an confidence on the quadruple q. As shown in Eq. 1, the margin-based loss function maximizes the discriminative margin between the positive examples and the negative examples such that $f(h', r, t')$ becomes larger with the margin M while $f(h, r, t)$ becomes smaller (ideally, 0). Intuitively, this means maximizing the distance between the positive entity and the negative entity of the head or tail entities with respect to the margin M. Suppose that the L-1 norm, $f(h, r, t) = |\mathbf{h} + \boldsymbol{r} - \mathbf{t}|$, is the score function. Then, the condition, $|\mathbf{e} - \mathbf{e}'| \approx M$ for each $(e, e') \in \{(h, h'), (t, t')\}$ and $e \in E$, holds on the margin-based loss function, as is verified in [6].

However, an uncertain KG contains confidence of quadruples. Due to the confidence, the influence of each quadruple is not uniform when learning the embeddings. Consequently, the global margin might not be able to capture the quadruple's influence. To capture the confidence of the quadruples, the margin M should be varied. On the usual KG, a margin M is set as a global constant

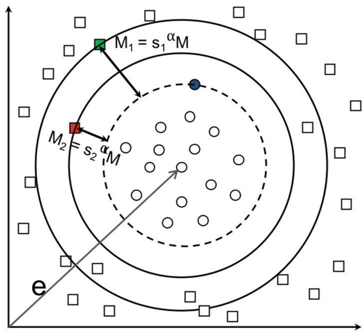

Fig. 1. The illustration of the entity embedding space with the different margin due to the uncertainty of quadruple. Circles and rectangles are positive and negative entities.

because all triples have the same confidence ($s = 1.0$). In an uncertain KG, each quadruple q denoted by (h, r, t, s) has its own confidence s. The condition $|\mathbf{e}-\mathbf{e}'| \approx M$ should be varied in accordance with the uncertainty of the quadruple s. Intuitively, the higher the uncertainty of the quadruple, the less margin should be used to keep the relation because (e, e') is likely to be noise. On each quadruple q, M should be then deducted when the uncertainty gets higher. This intuition behind is that if (h, r, t, s_i), (h, r, t, s_j) and $s_i \leq s_j$, then $s_i \cdot |\mathbf{h}-\mathbf{h}'| \leq s_j \cdot |\mathbf{h}-\mathbf{h}'|$ and $s_i \cdot |\mathbf{t}-\mathbf{t}'| \leq s_j \cdot |\mathbf{t}-\mathbf{t}'|$. The interpretation of confidence margins is motivated by the large and optimal margin [6,8]. In Fig. 1, we should maximize the larger margin of the high confidence quadruple (the green rectangle) rather than the low confidence quadruple (the red rectangle) because it is more likely that $|\mathbf{e}-\mathbf{e}'| \approx M$ holds on the green one. Therefore, the confidence margin-based loss function is derived as follows.

$$L = \sum_{(h,r,t,s)\in Q} \sum_{(h',r,t',s)\in Q'} [f(h, r, t) - f(h', r, t') + s^{\alpha} M]_+, \tag{2}$$

where $[x]_+$ is the positive part of x, $f(\cdot)$ is the score function for (h, r, t) of the quadruple q, M is the margin, $(h', r, t', 1.0)$ is a negative sample in Q' generated in the same way as T', s is the confidence of the quadruple, and α is a hyperparameter for adjusting the influence of confidence on the quadruple $\alpha \in \{x \in \mathbb{R} : x \geq 0\}$. As α becomes larger, the influence of confidence is amplified. When α equals 0, the influence of confidence is neutralized, and becomes the same as in the traditional translation-based model. Therefore, we can consider our loss function to be a generalization of the margin-based loss function.

5 Experiments and Results

To evaluate our approach, we conduct knowledge graph completion on uncertain KGs. Given an incomplete uncertain KG G, the task is to fill in G by predicting the set of missing quadruples $Q' = \{(h, r, t, \cdot) \mid h, t \in E, r \in R, (h, r, t, \cdot) \notin Q\}$. Note that it is difficult to measure confidence of quadruples. We thus decided not to try to predict the confidence of quadruples in this experiment.

5.1 Dataset

We constructed two new datasets: a synthetic dataset and a real-world dataset.

In the synthetic dataset, we derived an uncertain KG from FB15K-237. We selected FB15K-237, because it has no redundancy and has corrected the flaws of FB15K as reported in [4] and is more practical than WN18RR. We first sample a ρ proportion size triples and form a set T_c, where $T_c \subset K$ and $|T_c| = \rho|K|$. For each triple t; $t \in T_c$, we randomly generate a confidence score $s \sim uniform(0, 1)$ and form the quadruple (h, r, t, s); for the other triples t; $t \in T - T_c$, we associate them with a confidence score $s = 1.0$; as $(h, r, t, 1.0)$. Then, we try to corrupt each fact with the probability p. Specifically, (1) For each quadruple, we randomly generate probability $p \sim uniform(0, 1)$. (2) If $s < p$, we corrupt the triple by the negative sampling method, i.e., randomly replacing h, r, or t such that $h', t' \in E; h \neq h'; t \neq t'; r' \in R$ and $r \neq r'$ in order to create a quadruple (h', r', t', s); (3) If $s \geq p$, we leave the quadruple unchanged. Intuitively, the quadruple that has low confidence s are more likely to be corrupted, and the new created "negative" quadruple (h', r', t', s) has low confidence as our expectation. Statistically, $\int_0^1 p\, dp = 50\%$ of the quadruples in T_c will be corrupted. Note that, ρ plays a significant role in the uncertain KG because it indicates how uncertain KG is. As ρ increased, the likelihood, of which the uncertain KG contains noisy facts, increases. In uncertain KGs, we can regard ρ as the degree of uncertainty. We generate the KGs with the different ρ at 0.2, 0.4, 0.6, 0.8 and 1.0 to study uncertain KGs in various conditions.

In the real world dataset, we constructed the dataset from a real KG, NELL [2]. NELL provides a confidence score for each triple. To build our dataset, we first collected quadruples form NELL at the 995^{th} iteration. Then, we followed the cleaning process described in [17]. However, we did not add the inverse relation to the dataset as was done in that study. Adding the inverse relation would have made the dataset redundant. Therefore, our dataset is more difficult than the dataset in the study [17].

The statistical details of the two datasets are as follows. FB15K-237 contains 14,541 entities, 237 relations, 272,115, 17,536, 20,467 training, validation, and testing quadruples, respectively, while NELL comprises 75491 entities, 200 relations, 134,213, 10,000, 10,000 training, validation, and testing quadruples.

5.2 Experimental Setup

The evaluation protocol was similar to that of TransE [1]. In this task, a head or a tail entity of test triples is replaced by each entity in the knowledge graph and the score of each replaced triple is computed. After the calculation, the scores of the triples are used to rank the entities in what is referred to as a *raw ranking*.

Table 1. Results of KG completion on generated uncertain KGs from FB15K-237 with $\rho = 0.2, 0.4, 0.6, .08$ and 1.0

Method	FB ($\rho = 0.2$)			FB ($\rho = 0.4$)			FB ($\rho = 0.6$)			FB ($\rho = 0.8$)			FB ($\rho = 1.0$)		
	% Hit@		MR	%Hit@		MR	%Hit@		MR	%Hit@		MR	%Hit@		MR
	1	10		1	10		1	10		1	10		1	10	
$\text{TransE}_{s\geq 0}$	12.6	37.5	0.21	9.7	34.9	0.18	7.4	31.6	0.15	5.1	28.4	0.13	3.5	24.5	0.10
$\text{TransE}_{s\geq 0.2}$	13.0	37.6	0.21	11.2	35.5	0.19	9.3	33.4	0.17	6.9	30.4	0.15	4.9	27.3	0.12
$\text{TransE}_{s\geq 0.4}$	13.8	38.0	0.22	12.0	35.7	0.20	10.7	34.1	0.18	8.7	31.8	0.16	7.0	29.0	0.14
$\text{TransE}_{s\geq 0.6}$	13.8	37.7	0.22	12.8	35.7	0.20	11.6	33.8	0.19	10.1	31.7	0.17	9.0	26.9	0.15
$\text{TransE}_{s\geq 0.8}$	14.0	38.0	0.22	12.9	35.3	0.20	11.9	33.1	0.19	10.6	30.7	0.17	8.8	25.5	0.14
$\text{TransE}_{s=1.0}$	14.0	37.9	0.22	12.7	34.9	0.20	11.6	32.2	0.18	9.3	27.5	0.15	1.0	10.1	0.04
$\text{GTransE}_{\alpha=1}$	14.3	38.9	0.22	13.1	37.4	0.21	12.0	36.1	0.20	9.8	34.3	0.18	9.2	30.3	0.16
$\text{GTransE}_{\alpha=2}$	**14.5**	39.1	**0.23**	**13.6**	37.7	0.21	12.6	**36.4**	0.20	10.8	34.7	0.19	10.5	**31.4**	0.17
$\text{GTransE}_{\alpha=3}$	14.2	41.1	**0.23**	13.5	37.8	**0.22**	**13.3**	35.8	**0.21**	**12.9**	**35.0**	**0.20**	**10.8**	31.3	**0.18**
$\text{GTransE}_{\alpha=4}$	14.2	**41.2**	**0.23**	13.1	**38.0**	0.21	13.2	35.8	**0.21**	12.8	34.9	**0.20**	**10.8**	30.8	**0.18**

Table 2. Results of KG completion on NELL dataset

Method	% Hit@		MR
	1	10	
$\text{TransE}_{s\geq 0.90}$	10.44	30.15	0.18
$\text{TransE}_{s\geq 0.95}$	9.88	29.05	0.17
$\text{TransE}_{s=1.00}$	4.39	15.20	0.08
$\text{GTransE}_{\alpha=1}$	11.11	30.47	0.18
$\text{GTransE}_{\alpha=2}$	12.08	31.22	**0.19**
$\text{GTransE}_{\alpha=3}$	12.20	31.49	**0.19**
$\text{GTransE}_{\alpha=4}$	**12.21**	**31.81**	**0.19**

However, the *raw ranking* is not a fair measure of performance, as reported in [1]. We therefore evaluated the performance by using a filtered ranking [1]. A filtered rank re-ranks *raw rank* by eliminating entities whose corresponding triples are contained in the training, validation, and test datasets. After obtaining the filtered ranking, we employed three evaluation metrics: Hit@1, Hit@10 and mean reciprocal rank (MR). Hit@1 and Hit@10 are the percentages of test triples whose entities are ranked in the top 1 and 10, respectively. MR averages the inverses of the ranks of the correct entities.

The baseline in the experiment was chosen to be TransE, because the objective of the experiment was to evaluate the performance of the confidence margin-based loss function. Although GTransE can use any score functions for the translation-based models, we limited the score function to L-1 norm of TransE, in order to focus on the performance of the loss function. However, using TransE in the traditional way is not fair, because GTransE utilizes the confidence of the quadruples to learn the representation. We thus decided to implement an intuitive way for TransE utilizing confidence s. The idea is that quadruples whose s is less than the threshold are eliminated from the training data. The experiment set six thresholds for TransE: $s \geq 0.0$, $s \geq 0.2$, $s \geq 0.4$, $s \geq 0.6$, $s \geq 0.8$, and $s = 1.0$. Note that we conducted the experiment by setting the threshold for the NELL dataset at $s \geq 0.90$, $s \geq 0.95$ and $s = 1.0$ because the range of confidence scores in NELL lies between 0.9 and 1.0.

The implementations of TransE and GTransE both used the grid search algorithm to find appropriate parameters. The dimension was selected from {20, 50, 100, 200}. The search range for the margin M was set at {1, 5, 10, 50, 100}. The learning rate was selected from {0.1, 0.001, 0.0001}. The best parameters were selected by the best MR on the validation set. We used the **Bern** method [14] to generate the negative samples. Normalization was used as the regularization. TransE and GTransE were optimized with the Adam algorithm [10]. All methods were trained in 2000 epochs. For GTransE, we set the α parameter to 1, 2, 3, and 4 to study the effect of varying α.

5.3 Results

The experimental results are presented in Tables 1 and 2. They show that GTransE outperforms TransE on all datasets.

As α increases, the margin of quadruples with low uncertainty (high confidence s) becomes larger for learning the representations. GTransE then maintains performance, even though ρ, representing the degree of uncertainty, increases when compared with all TransE, as shown in Table 1. This confirms the effect of the large margin [6]. Regarding the results of TransE in this table, we found that when ρ is low, filtering quadruples that have low confidence tends to yield better results. This is true because it makes more sense to remove them from the training than to keep them. However, a ρ increases, the portion of low-confidence quadruples drastically increases; consequently, too many quadruples are removed. As a result, TransE does not have enough relationships to learn the representation. This is why $\text{TransE}_{s=1.0}$ gives the worst results when $\rho = 1.0$.

The uncertainty is lower in the NELL dataset, than in the synthetic dataset. Accordingly, $\text{GTransE}_{\alpha=1}$ yields similar results as $\text{TransE}_{s=0}$ in Table 2. However, an improvement over TransE appears as α increases. The reason is that the confidence of quadruples is amplified. Such amplification makes the margin become relatively larger for high confidence facts. GTransE then can better adjust the margin for such facts. The results clearly show the effect of α.

In summary, removing low-confidence quadruples enables TransE to learn a comparatively good representation; however, removing too many quadruples becomes a double-edged sword. GTransE simply uses all training data and directly uses the confidence of quadruples to determine the margin when learning embedding representations. GTransE thus does not suffer when the amount of training data is small and it still captures the effect of uncertainty.

6 Conclusion

We introduced learning representation on uncertain knowledge graphs. To deal with the confidence score, we proposed a new margin loss function for translation-based models. The experiment shows that our approach is promising to capture confidence in knowledge graphs. In the future, we plan to investigate other score functions in the other translation-based models.

Acknowledgment. This work was partially supported by the New Energy and Industrial Technology Development Organization (NEDO).

References

1. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Proceedings of Advances in Neural Information Processing Systems (NIPS), pp. 2787–2795 (2013)
2. Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka Jr, E.R., Mitchell, T.M.: Toward an architecture for never-ending language learning. In: Proceedings of the AAAI Conference, pp. 1306–1313 (2010)
3. Chen, X., Chen, M., Shi, W., Sun, Y., Zaniolo, C.: Embedding uncertain knowledge graphs. In: Proceedings of the AAAI Conference, vol. 33, pp. 3363–3370 (2019)
4. Dettmers, T., Minervini, P., Stenetorp, P., Riedel, S.: Convolutional 2D knowledge graph embeddings. In: Proceedings of the AAAI Conference, pp. 1811–1818 (2018)
5. Ebisu, T., Ichise, R.: TorusE: knowledge graph embedding on a lie group. In: Proceedings of the AAAI Conference, pp. 1819–1826 (2018)
6. Fan, M., Zhou, Q., Zheng, T.F., Grishman, R.: Large margin nearest neighbor embedding for knowledge representation. In: 2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), vol. 1, pp. 53–59, December 2015
7. Ji, G., He, S., Xu, L., Liu, K., Zhao, J.: Knowledge graph embedding via dynamic mapping matrix. In: Proceedings of Annual Meeting of the Association for Computational Linguistics, pp. 687–696 (2015)
8. Jia, Y., Wang, Y., Lin, H., Jin, X., Cheng, X.: Locally adaptive translation for knowledge graph embedding. In: Proceedings of the AAAI Conference, pp. 992–998 (2016)
9. Kertkeidkachorn, N., Ichise, R.: An automatic knowledge graph creation framework from natural language text. IEICE Trans. Inf. Syst. **101**(1), 90–98 (2018)
10. Kingma, D., Adam, J.B.: A method for stochastic optimization. In: Proceedings of International Conference on Learning Representations (2015)
11. Lin, Y., Liu, Z., Luan, H., Sun, M., Rao Liu, S.: Modeling relation paths for representation learning of knowledge bases. In: Proceedings of Conference on Empirical Methods in Natural Language Processing, pp.705–714 (2015)
12. Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. In: Proceedings of the AAAI Conference, pp. 2181–2187 (2015)
13. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of Advances in Neural Information Processing Systems (NIPS), pp. 3111–3119 (2013)

14. Wang, Z., Zhang, J., Feng, J., Chen, Z.: Knowledge graph embedding by translating on hyperplanes. In: Proceedings of the AAAI Conference, pp. 1112–1119 (2014)
15. Xiao, H., Huang, M., Zhu, X.: TransG: a generative model for knowledge graph embedding. In: Proceedings of Annual Meeting of the Association for Computational Linguistics, pp. 2316–2325 (2016)
16. Xie, R., Liu, Z., Lin, F., Lin, L.: Does William Shakespeare really write hamlet? Knowledge representation learning with confidence. In: Proceedings of the AAAI Conference (2018)
17. Xiong, W., Hoang, T., Wang, W.Y.: DeepPath: a reinforcement learning method for knowledge graph reasoning. In: Proceedings of Conference on Empirical Methods in Natural Language Processing, pp. 564–573 (2017)