A Knowledge Selective Adversarial Network for Link Prediction in Knowledge Graph

Kairong Hu, Hai Liu, and Tianyong Hao(✉)

School of Computer Science, South China Normal University, China
2018022615@m.scnu.edu.cn, namelh@gmail.com, haoty@m.scnu.edu.cn

Abstract. Knowledge Graphs (KGs) contain rich semantic information and are of importance to many downstream tasks. In order to enhance practical utilization of KGs, KG completion task, which is also called link prediction, is a newly emerging hot research topic. During KG embedding model training, negative sampling is a fundamental method for obtaining negative samples. Inspired by an adversarial learning framework KBGAN, this paper proposes a new knowledge selective adversarial network, named as KSGAN, using a knowledge selector for high-quality negative sampling to benefit link prediction. The performances of our model KSGAN are evaluated on three standard knowledge completion datasets: FB15k-237, WN18 and WN18RR. The results show that KSGAN outperforms a list of baseline models on all the datasets, demonstrating the effectiveness of the proposed model.

Keywords: Adversarial learning · Knowledge Graph · Link prediction · KSGAN.

1 Introduction

Knowledge Graphs (KGs) contain linked knowledge in the form of triples which describe relations between entities. Typical examples of KGs are Freebase [1], WordNet [2], Yago [3], etc. KGs consist of numerous facts by triples, i.e. (h, r, t), where h, r and t represent head entities, relations and tail entities respectively. Therefore, as its amount of valuable information, KG becomes the base of many research tasks such as information extraction, question answering and recommender systems [4]. However, open-domain KG is far from complete [5] due to its dramatic difficulty in incorporating all concepts that human ever had. Therefore, it is necessary to develop algorithms to predict missing entities or relations given head entities and relations (or relations and tail entities) or head entities and tail entities. Since the target information exists in the form of text, a variety of knowledge graph embedding(KGE) techniques that embed the triples of facts consisting of entities and relations in KG into a continuous vector space are proposed [4]. With the numeric representations of entities and relations, the similarities between entities or relations can be computed and measured. In order to model relations and entities in KG, positive and negative examples are frequently needed to train KG embedding models. Therefore, negative sampling [6] is widely used to acquire a great deal of negative examples (also called corrupted triples) when training a knowledge graph embedding model (e.g., TransE [6], TransD [7], DistMult [8], ComplEx [9]).

However, previous works such as TransE [6] generates corrupted triples uniformly. The generated corrupted triples consist of false corrupted triples (true facts) and true corrupted triples, in which the former denotes an accurate relation between head and tail entities while the latter fails. Moreover, true corrupted triples are composed of the triples that provide more semantic information (e.g. (*Beijing, IsA, province*)) and others that contain less (e.g. (*Beijing, IsA, car*)), since the former may be more reasonable and easy to mix up. Using true corrupted triples which provide less information may result in slowing training process down since the corrupted triples are obviously false and very likely to be distinguished from true facts. KBGAN [10] is a typical adversarial learning framework for link prediction, regarding semantic matching models (e.g., DistMult [8], ComplEx [9]) as a generator in GAN and translational distance models (e.g., TransE [6], TransD [7]) as a discriminator. This paper proposes a novel knowledge selective adversarial network (KSGAN) for link prediction task. In contrast to KBGAN and IGAN [11], KSGAN leverages a knowledge selector as filter to select corrupted triples from generator. The selected high-quality corrupted triples are used to help discriminator to avoid zero loss problem during training process. Three publically available datasets FB15k-237, WN18 and WN18RR are used to test the performance of our model. Through the comparison with a list of state-of-the-art baseline methods such as KBGAN, results show that KSGAN outperforms the baselines and achieves improvement (7.0% for MRR and 1.4% for Hits@10) on average.

In summary, the major contributions of this paper are three-fold: (1) Focusing on the negative sampling problem, a new knowledge selector to select high-quality negative triples for KG embedding model is proposed. (2) A novel knowledge selective adversarial network is proposed to predict missing entities for link prediction tasks. (3) Experiments on standard datasets illustrate the effectiveness of KSGAN using MRR and Hits@10 metrics.

2 Related Work

2.1 Knowledge Graph Embedding Models

Different knowledge graph embedding models explore diverse methods to embed triples into vector spaces. TransE [6] is a classic translational distance model, which represents entities and relations in d-dimensional vector $\mathbb{R}^d$ by modeling $\mathbf{h} + \mathbf{r} \approx \mathbf{t}$ given a true fact (h, r, t). Various variants such as TransH [12], TransM [13], TransR [14] and TransD [7] have been developed in recent years. These models focus on the drawbacks of TransE and introduce some effective strategies (e.g. using more reasonable scoring functions) to represent entities and relations. However, the aforementioned models simulate unique embedding in vector spaces for every entities and relations. Targeting at more flexible models, TransF [15] ensures that $\mathbf{t}$ (or $\mathbf{h}$) has the same direction with $\mathbf{h}+\mathbf{r}$ (or $\mathbf{t}-\mathbf{r}$) without enforcing strict magnitude constraints between them. ManifoldE [16] uses a manifold function to constrain $\mathbf{t}$ (or $\mathbf{h}$) within a sphere space with a center of $\mathbf{h}+\mathbf{r}$ (or $\mathbf{t}-\mathbf{r}$). As a generic model, GTrans [17] introduces eigenstate and mimesis to represent the features of entities and relations.

Different from the translational distance model, RESCAL [18] is a classic semantic matching model, which concentrates on capturing latent semantics between head entities and tail entities using a bilinear function as scoring function. DistMult [8] simplifies RESCAL by restricting interactions between heads and tails entities in a diagonal matrix. ComplEx [9] maps entity and relation embeddings to a complex space rather than a real space. SimplE [19] simplifies ComplEx by considering a different similarity scoring function. TuckER [20] is based on Tucker decomposition and the semantic matching models mentioned above such as RESCAL, DistMult, ComplEx and SimplE are all special cases of TuckER. Other semantic matching models such as NTN [21] and MLP [22], focus on neural network architectures and try to output scores from hidden layer of neural network which takes the vectors of entities and relations as input given facts (h, r, t).

The scoring functions of the models are investigated and summarized in Table 1. As shown in the table, $\mathbf{h}$, $\mathbf{r}$ and $\mathbf{t}$ represent a embedding vector of head entities, relations and tail entities. The vector related to hyperplane in TransH is denoted by w_r while w_h, w_r and w_t are mapping vectors in TransD. $w_r \in \mathbb{R}$ in TransM is the weight associated with specific relations. A radius of sphere in ManifoldE is denoted as D_r. In translational distance models such as TransR, the projection matrix used to map entities from entity space to relation space is denoted as M_r, while the matrix that contains interactions between heads and tails entities is denoted as M_r in semantic matching model like RESCAL. $\bar{\mathbf{t}}$ in ComplEx is the representation of the conjugate of a tail entity embedding vector $\mathbf{t}$ and $\mathbf{r}^{-1}$ in SimplE is the embedding vector of inverse relation. The relation-specific weight matrices are denoted by M_r^1 and M_r^2 in NTN as well as M^1, M^2 and M^3 are the weights in different layers in MLP. The tensor are denoted by $\underline{M}_r$ and $\mathcal{W}$. $Re(\cdot)$ means taking the real part of a complex vector and $\odot$ is element-wise product.

2.2 Negative Sampling Methods

The goal of training a knowledge graph embedding model is to tell the model how to distinguish right from wrong given negative triples and positive ones. Thus, negative sampling is necessary for training KG embedding model for the reason that both negative and positive triples are needed to be provided during training process. TransE [6] generates corrupted triples by replacing heads or tails in true triples randomly for each triple in mini-batch. It is possible that there exist some false negative examples which also make sense. For instance, a true fact (*Jackie Chan, profession, actor*) may turn into a negative example (*Jackie Chan, profession, director*), which is also true. In order to reduce false

Table 1. The score functions used in existing KG embedding models

Type	Model	Score function $f(h,r,t)$	Embedding
Translational distance model	TransE	$-\Vert\mathbf{h}+\mathbf{r}-\mathbf{t}\Vert_1$	$\mathbf{h},\mathbf{r},\mathbf{t}\in\mathbb{R}^d$
	TransH	$-\left\Vert(\mathbf{h}-w_r^\top\mathbf{h}w_r)+\mathbf{r}-(\mathbf{t}-w_r^\top\mathbf{t}w_r)\right\Vert_2^2$	$\mathbf{h},\mathbf{r},\mathbf{t},w_r\in\mathbb{R}^d$
	TransM	$-w_r\Vert\mathbf{h}+\mathbf{r}-\mathbf{t}\Vert_1$	$\mathbf{h},\mathbf{r},\mathbf{t}\in\mathbb{R}^d$
	TransR	$-\Vert M_r\mathbf{h}+\mathbf{r}-M_r\mathbf{t}\Vert_2^2$	$\mathbf{h},\mathbf{t}\in\mathbb{R}^d,\mathbf{r}\in\mathbb{R}^k,$ $M_r\in\mathbb{R}^{k\times d}$
	TransD	$-\left\Vert(w_rw_h^\top+I)\mathbf{h}+\mathbf{r}-(w_rw_t^\top+I)\mathbf{t}\right\Vert_2^2$	$\mathbf{h},w_h,\mathbf{t},w_t\in\mathbb{R}^d,$ $\mathbf{r},w_r\in\mathbb{R}^k$
	TransF	$(\mathbf{h}+\mathbf{r})^\top\mathbf{t}+(\mathbf{t}-\mathbf{r})^\top\mathbf{h}$	$\mathbf{h},\mathbf{r},\mathbf{t}\in\mathbb{R}^d$
	ManifoldE	$-(\Vert\mathbf{h}+\mathbf{r}-\mathbf{t}\Vert_2^2-D_r^2)^2$	$\mathbf{h},\mathbf{r},\mathbf{t}\in\mathbb{R}^d$
	GTrans	$-\Vert W_r\odot(\mathbf{h}+\mathbf{r}-\mathbf{t})\Vert_2^2$	$\mathbf{h},\mathbf{r},\mathbf{t},W_r\in\mathbb{R}^d$
Semantic matching model	RESCAL	$\mathbf{h}^\top M_r\mathbf{t}$	$\mathbf{h},\mathbf{t}\in\mathbb{R}^d,$ $M_r\in\mathbb{R}^{d\times d}$
	DistMult	$\langle\mathbf{h},\mathbf{r},\mathbf{t}\rangle$	$\mathbf{h},\mathbf{r},\mathbf{t}\in\mathbb{R}^d$
	ComplEx	$Re(\mathbf{h},\mathbf{r},\bar{\mathbf{t}})$	$\mathbf{h},\mathbf{r},\mathbf{t}\in\mathbb{C}^d$
	SimplE	$\frac{1}{2}\left(\langle\mathbf{h},\mathbf{r},\mathbf{t}\rangle+\left\langle\mathbf{t},\mathbf{r}^{-1},\mathbf{h}\right\rangle\right)$	$\mathbf{h},\mathbf{r},\mathbf{r}^{-1},\mathbf{t}\in\mathbb{R}^d$
	TuckER	$\mathcal{W}\times_1\mathbf{h}\times_2\mathbf{r}\times_3\mathbf{t}$	$\mathbf{h},\mathbf{t}\in\mathbb{R}^d,\mathbf{r}\in\mathbb{R}^k,$ $\mathcal{W}\in\mathbb{R}^{d\times d\times k}$
	NTN	$\mathbf{r}^\top tanh(\mathbf{h}^\top\underline{M}_r\mathbf{t}+M_r^1\mathbf{h}+M_r^2\mathbf{t}+b_r)$	$\mathbf{h},\mathbf{t}\in\mathbb{R}^d,\mathbf{r},b_r\in\mathbb{R}^k,$ $\underline{M}_r\in\mathbb{R}^{d\times d\times k},$ $M_r^1,M_r^2\in\mathbb{R}^{k\times d}$
	MLP	$w^\top tanh(M^1\mathbf{h}+M^2\mathbf{r}+M^3\mathbf{t})$	$\mathbf{h},\mathbf{r},\mathbf{t}\in\mathbb{R}^d$

negative triples, TransH [12] designs a strategy for replacing head entities or tail entities from a given true triple (h,r,t) with Bernoulli distribution. The aforementioned negative sampling methods are likely to be effective but not a reasonable way to generate high-quality corrupted triples that contain valuable information. Inspired by GAN [23], KBGAN [10] and IGAN [11] propose an adversarial learning framework for knowledge representation learning, which obtains high-quality negative samples effectively. KBGAN introduces a framework that uses one of semantic matching models (e.g. DistMult or ComplEx) as generator to generate high-quality negative samples from a candidate set. Meanwhile, discriminator, adopting one of the translational distance models (e.g. TransE or TransD), is trained given a positive sample and negative sample provided by generator. IGAN addresses a similar framework in which a generator corrupts true triples with the entire entity set and uses a different reward function. However, the performances of discriminator and generator from previous works can still be enhanced by adding a new knowledge selector proposed in this paper.

3 Model

3.1 Overall Framework

In order to obtain negative triples, most of previous works generate corrupted triples (h',r,t) or (h,r,t') with a certain probability distribution given a true fact triple (h,r,t). As observed in IGAN [11], training a translational distance

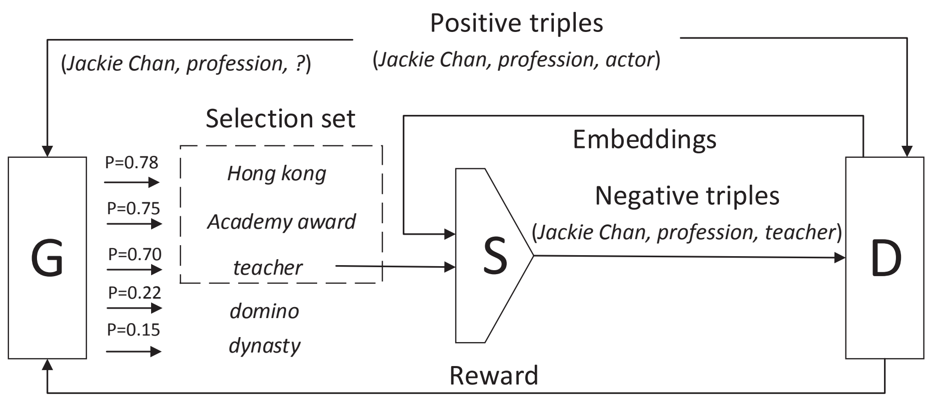

Fig. 1. The generator (G) computes probabilities of different triples. The knowledge selector (S) forms a selection set and then selects a semantically correct triple with the assist of the discriminator (D). The discriminator is trained with both positive and negative triples.

model with marginal loss function may cause zero loss problem. Whether the scores of those corrupted triples with highest probabilities are within the range of the margin in marginal loss function, is not unwarrantable during the training process. Using corrupted triples whose scores are not within the margin of the scores of positive triples may bring zero loss to the marginal loss function. The parameters of KG embedding models are not being updated due to vanishing gradient caused by zero loss. Thus, training with such corrupted triples, which have high scores in semantic matching models but low scores in translational distance models, may not push KG embedding models to converge effectively. Based on the framework proposed by IGAN and KBGAN [10], a new knowledge selective adversarial network KSGAN is therefore proposed to train KG embedding models with positive and negative triples to avoid the zero loss problem. Comparing with KBGAN and IGAN, KSGAN has a new component knowledge selector, which is a filter aiming to filter out obviously false triples and select semantic ones given positive training examples. The structure of KSGAN is illustrated in Fig. 1.

In KSGAN, the generator, which can be regarded as an agent in reinforcement learning, generates negative triples with probability distributions. The discriminator learns to adjust its parameters by minimizing loss function and calculates the rewards returned to the generator as interactions from environment to the agent in reinforcement learning. The goal of KSGAN is to train a discriminator with the negative triples generated by generator.

3.2 Triple Selection with Assist of Discriminator

Knowledge graph is composed by entities from a set of entities $\mathcal{E}$ and relations from a set of relations $\mathcal{R}$. $\mathcal{T}$ denotes a set of ground truth triples and $\mathcal{T}'$ denotes a set of corrupted triples (h', r, t') through the corruption of ground truth triples by replacing head entities or tail entities.

During training translational distance models such as TransE and TransD, a marginal loss function is applied, which is shown as Eq. 1:

$$L = \sum_{(h,r,t)\in\mathcal{T}} \sum_{(h',r,t')\in\mathcal{T}'} \max\left(0, f(h,r,t) - f(h',r,t') + \gamma\right) \tag{1}$$

where $f\left(h,r,t\right)$ is a scoring function in knowledge graph embedding models given a positive triple (h,r,t).

KSGAN is an adversarial network which consists of a generator, a knowledge selector and a discriminator. Following KBGAN, since small subset of entities shrink the search space of entities, corrupted triples are constructed based on the set of N_s candidate entities. The generator uses one of semantic matching models to represent different negative triples by calculating probabilities using a softmax function as Eq. 2:

$$p_i = \frac{\exp f_G(h'_i, r, t'_i)}{\sum_{j=1}^{N_s} \exp f_G(h'_j, r, t'_j)} \tag{2}$$

where N_s is the size of candidate set.

However, generated negative triples only based on a generator may be semantically false and cause the zero loss problem when training discriminator. Since entities closed to each other in the same vector space may express similar meanings or have semantically close information to some extent, the corrupted triples consisted of those entities may have similar scores. Training discriminator with those negative triples and their corresponding ground truth triples may cause higher loss value, where the parameters of negative triples should be updated since their scores are improperly similar to that of true triples. Therefore, aiming to avoid zero loss problem, a knowledge selector is designed to select S_s triples with relatively high probabilities from generator to form a selection set. Afterwards, a negative triple that has the closest distance to its ground truth triple is selected by a selector, based on the representation of entities and relations from KG embedding models in discriminator. The selector selects those triples with maximum scores from the selection set, referring to the embedding of KG embedding model in discriminator, which can be formulated as Eq. 3:

$$f_{sel}(h', r, t') = \max_{(h',r,t')\in{\mathcal{T}_s}'} \left(f_D(h', r, t')\right) \tag{3}$$

where the selection set is denoted by ${\mathcal{T}_s}'$, which is composed by S_s corrupted triples with high probabilities selected by the selector.

Thus, the selector selects negative triples with correct semantic information (high score in semantic matching models) and close distance (high score in translational distance models) to avoid the zero loss problem when training the discriminator. Suppose (h', r, t') is a negative sample selected from the selection set ${\mathcal{T}_s}'$ given a positive sample (h, r, t), one of the translational distance models is regarded as the discriminator and a objective function can be defined as follows:

$$L_D = \sum_{(h,r,t)\in\mathcal{T}} \max(0, f(h,r,t) - f_{sel}(h', r, t') + \gamma) \tag{4}$$

where $f_{sel}(h', r, t')$ is the score of a selected negative triple (h', r, t').

In reinforcement learning, S_s triples in the selection set selected by the selector are used to compute a reward by the discriminator embedding model. The reward function is formulated as follows:

$$R = f_D(h', r, t') \tag{5}$$

where $(h', r, t') \in \mathcal{T}_s{}'$ and $f_D(h', r, t')$ are the results from the discriminator through calculating the scores of corrupted triples (h', r, t) or (h, r, t'). In order to maximize the expectation of reward, the generator learns to follow a policy to generate more triples which have high semantic scores. The generator is formulated as Eq. 6:

$$R_G = \sum_{(h,r,t)\in\mathcal{T}} \sum_{(h',r,t')\in\mathcal{T}_s{}'} E_{(h',r,t')\sim p_G((h',r,t')|(h,r,t))}\left[f_D(h', r, t')\right] \tag{6}$$

In order to update parameters of the generator which can be viewed as a policy to generate triples with high probabilities, policy gradient [24] with baseline b is used to tune the parameters of semantic models in the generator. The policy gradient is:

$$\begin{aligned} \nabla_G R_G = &\sum_{(h,r,t)\in\mathcal{T}} \sum_{(h',r,t')\in\mathcal{T}_s{}'} \\ &E_{(h',r,t')\sim p_G((h',r,t')|(h,r,t))}\left[\Delta f_D(h', r, t') \nabla_G \log p_G((h', r, t')|(h, r, t))\right] \end{aligned} \tag{7}$$

where p_G denotes the policy for generator to generate negative samples as well as $\Delta f_D(h', r, t')$ is the difference between the score of negative triples and baseline b, which is nearly equal to the mean of rewards of corrupted triples in selection set $\mathcal{T}_s{}'$.

To avoid the zero loss problem, we further propose a strategy to exchange the models between generator and discriminator. In other words, one of the translational distance models is regarded as a generator while one semantic matching model acts as the role of discriminator. The generator generates the distribution of negative triples using Eq. 2, in which the score function belongs to translational distance model. The selector tends to select those triples that have relatively high scores in distance models and form a selection set. The triple with maximum value of semantic score in selection set is selected by selector. The logistic loss function used to train discriminator with selected negative triples and positive triples is defined as follows:

$$L_D = \sum_{(h,r,t)\in\mathcal{T}\cup\mathcal{T}_s{}'} \log\left\{1 + \exp\left[-l \cdot f_D(h, r, t)\right]\right\} \tag{8}$$

where l is a label used to distinguish positive($l = +1$) and negative($l = -1$) triples.

The reward computed by discriminator is returned to generator as feedback evaluating the quality of generated triples. Again, the generator updates the parameters of the KG embedding model through the policy gradient using Eq. 7.

Table 2. Statistics of the three standard datasets.

Datasets	#Entity	#Relation	#Training	#Validation	#Testing
FB15k-237	14,541	237	272,115	17,535	20,466
WN18	40,943	18	141,442	5,000	5,000
WN18RR	40,943	11	86,835	3,034	3,134

4 Experiments

4.1 Datasets

Three widely used standard dataset FB15k-237, WN18 and WN18RR for link prediction task are used to test our model. The dataset FB15k-237 [25] is a variant version of FB15k [6]. The dataset has been widely applied to KG completion tasks, such as link prediction and triple classification. It is constructed by removing redundant relations from the original dataset. In addition, to enhance the quality of evaluation, we further use WN18 [6] and its subset WN18RR [26]. The two datasets are the subsets of WordNet database, which consists of lexical relations (e.g. hypernym and hyponym) between words. The statistical characteristics of the three datasets are shown in Table 2.

4.2 Baseline methods

Our models are compared with following baseline methods:

- **TransE** is a classic translational distance model proposed in [6]. It captures latent representations through modeling translational distance between relations and entities in vector spaces.
- **TransD** is another KG embedding method proposed in [7] that projects entity vectors via a dynamic mapping matrix.
- **ComplEx** is a semantic matching model proposed in [9]. TransE, TransD and ComplEx are the pre-trained models used in KBGAN algorithm mentioned above and implemented using open-source code[1].
- **KBGAN(TransE+ComplEx)** is an adversarial learning model proposed in [10], using pre-trained model TransE as discriminator and ComplEx as generator.
- **KBGAN(TransD+ComplEx)** is the same adversarial learning model proposed in [10], taking pre-trained model TransD as discriminator and ComplEx as generator.

4.3 Evaluation Metrics

Following previous works such as TransE [6] and ComplEx [9], two commonly used metrics, filtered mean reciprocal rank (MRR) and hits at 10 (Hits@10), are used in the following experiments. We follow the similar filtered setting [6] in the

[1] https://github.com/cai-lw/KBGAN

experiments to avoid false corrupted triples (true facts) showing in evaluation process. The mean reciprocal rank MRR can be computed using Eq. 9:

$$MRR = \frac{1}{2 * |\mathcal{T}_t|} \sum_{(h,r,t) \in \mathcal{T}_t} \frac{1}{rank_h} + \frac{1}{rank_t} \tag{9}$$

where$\mathcal{T}_t$ is a set of test triples in link prediction task and the number of test triples is denoted as $|\mathcal{T}_t|$.

The Hits@10 is the proportion of correct entities ranked in top 10 after calculating the scores and ranking them in descending order:

$$Hits@10 = \frac{1}{2 * |\mathcal{T}_t|} \sum_{(h,r,t) \in \mathcal{T}_t} I(rank_h \leq 10) + I(rank_t \leq 10) \tag{10}$$

where $I(\cdot)$ is an indicator function representing that whether the ranks on head or tail entities are within 10 or not.

4.4 Results

Following KBGAN [10], our model KSGAN also utilizes pre-training models (e.g. TransE, TransD and ComplEx) as generator and discriminator in the adversarial learning network. In pre-training process, the aforementioned models are trained 1000 epochs, taking 100 training data as mini-batch. In each epoch, we generate corrupted triples by replacing head or tail entities from a given true triple (h, r, t) based on the average number of tails per head or heads per tail, similar to previous works (e.g. TransH). Using both true triples (h, r, t) and corrupted triples (h', r, t'), a knowledge graph embedding model is trained as well as carry out early-stop process every 50 epochs by testing the model on the validation dataset and recording MRR and hits@10. Following KBGAN, the dimension of embedding vectors is set to 50 and the value of margin γ in translational distance models (e.g. TransE and TransD) is 3 for the scoring function that use L_1 distance. The value of regularization λ in semantic matching models (e.g. ComplEx) is 1 for FB15k-237 and 0.1 for WN18/WN18RR. We use Adam [27] to update the model parameters in each epoch with their default settings $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 10^{-8}$.

In adversarial training process, the pre-trained models are loaded in KSGAN. Following the settings in KBGAN, the size of candidate entity set is set to 20 and the discriminator is trained 5000 epochs with 100 batches. Early-stop is carried out per 100 epochs evaluating the metrics such as MRR and hits@10 on validation sets. We regard translational distance models (e.g. TransE or TransD) as discriminator and semantic matching models (e.g. ComplEx) as generator as well as attempt to exchange the roles between them.

In KSGAN, two different types of models are considered and each type has two combinations in our experiments, (1)ComplEx is used as generator while TransE or TransD are as discriminator, named as KSGAN(TransE+ComplEx)

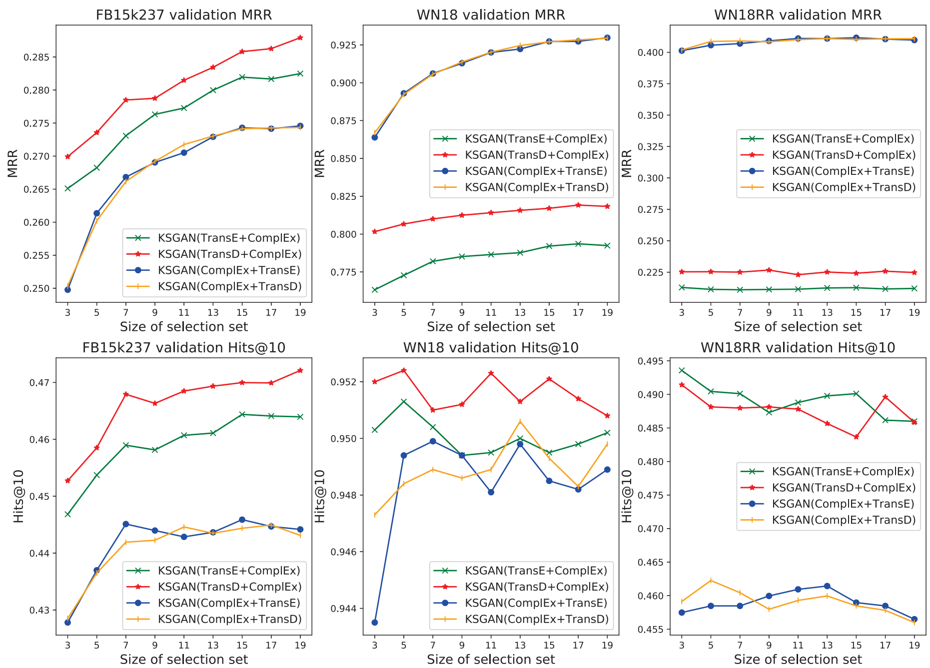

Fig. 2. The results of hyperparameter tunning using MRR and Hits@10.

or KSGAN(TransD+ComplEx) and (2) TransE or TransD acts as the role of generator while ComplEx is discriminator, denoted as KSGAN(ComplEx+TransE) or KSGAN(ComplEx+TransD). The number of selection set S_s not only affects the embedding in the discriminator but also affects rewards returned to the generator. Thus, to obtain the optimal value of the hyperparameter S_s, KSGAN is tested with different values $S_s = 3, 5, 7, 9, 11, 13, 15, 17, 19$ on the three datasets.

The results of hyperparameter S_s tunning on the validation dataset are shown in Fig. 2. The testing results compared with other baselines are shown in Table 3. The performances of KSGAN on the validation dataset, as shown in Fig. 2, improve when S_s increases from 3 to 15 and the results on MRR tend to be stable when S_s is larger than 15. We thus select 15 as the optimal value for the hyperparameter S_s. The testing results of KSGAN on the three datasets FB15k-237, WN18 and WN18RR with optimized S_s is displayed in Table 3.

The results show that KSGAN has improvements on the three datasets especially on WN18 (about 12.2% increasing for KSGAN(TransE+ComplEx) and 4.5% for KSGAN(TransD+ComplEx)), compared with KBGAN using the same evaluation metric MRR. However, when testing on FB15k-237, the results are equal to KBGAN on MRR but have slight improvements using Hits@10. The results show that KSGAN has a improvement on MRR (about 1% for KSGAN(TransE+ComplEx) and 2% for KSGAN(TransD+ComplEx)) and Hits@10 (about 2% for both models) on WN18RR. The performances of models KSGAN(ComplEx+TransE) and KSGAN(ComplEx+TransD) on Hits@10 demon-

Table 3. The performance comparision of models by setting $S_s = 15$.

	FB15k-237		WN18		WN18RR	
Models	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10
TransE(pre-trained))	24.2	42.2	43.3	91.5	18.6	45.9
KBGAN(TransE+ComplEx)	27.8	45.3	70.5	94.9	21.0	47.9
KSGAN(TransE+ComplEx)	**27.9**	**46.2**	**79.1**	**95.4**	**21.2**	**48.7**
TransD(pre-trained)	24.5	42.7	49.4	92.8	19.2	46.5
KBGAN(TransD+ComplEx)	27.7	45.8	77.9	94.8	21.5	46.9
KSGAN(TransD+ComplEx)	**28.0**	**46.5**	**81.4**	**95.2**	**22.0**	**47.9**
ComplEx(pre-trained)	26.4	43.6	76.1	92.3	37.2	45.3
KSGAN(ComplEx+TransE)	26.7	**44.0**	**92.8**	**95.0**	40.5	45.5
KSGAN(ComplEx+TransD)	**26.8**	**44.0**	**92.8**	**95.0**	**40.6**	**45.6**

strate the improvements on WN18 (about 2.9%) but the results on the rest datasets are almost equal to KBGAN indicating that TransE and TransD have little help to improve Hits@10. It is worth noting that KSGAN achieve a dramatic improvement on MRR on WN18 (about 21.9%) and WN18RR (about 9% for both models) but have slight improvements on MRR on FB15k-237 (about 1.1% for KSGAN(ComplEx+TransE) and 1.5% for KSGAN(ComplEx+TransD)), compared with the pre-trained model ComplEx.

5 Conclusions

This paper proposes a new model for negative sampling in knowledge graph embedding models to generate high-quality negative samples for avoiding the zero loss problem. Based on an adversarial learning framework, pre-trained models TransE, TransD and ComplEx are used as generator and discriminator in an exchanged way. Experiment results on three widely used datasets show that the performances of our proposed model have improvements compared with baseline methods when setting optimal hyperparameters, demonstrating that the performance of proposed adversarial learning network is effective for link prediction.

Acknowledgement. This work is supported by National Natural Science Foundation of China (No. 61772146) and Guangdong Natural Science Foundation (No. 2016A030313441, 2018A030310051).

References

1. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: a collaboratively created graph database for structuring human knowledge. ACM SIGMOD international conference on Management of data. pp. 1247-1250. (2008)
2. Miller, G.A.: WordNet: a lexical database for English. Communications of the ACM **38**(11), 39-41 (1995)
3. Suchanek, F.M., Kasneci, G., Weikum, G.: Yago: a core of semantic knowledge. International conference on World Wide Web. pp. 697-706. (2007)

4. Wang, Q., Mao, Z., Wang, B., Guo, L.: Knowledge graph embedding: A survey of approaches and applications. IEEE Transactions on Knowledge and Data Engineering **29**(12), 2724-2743 (2017)
5. Min, B., Grishman, R., Wan, L., Wang, C., Gondek, D.: Distant supervision for relation extraction with an incomplete knowledge base. NAACL. pp. 777-782 (2013)
6. Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. NIPS. pp. 2787-2795 (2013)
7. Ji, G., He, S., Xu, L., Liu, K., Zhao, J.: Knowledge graph embedding via dynamic mapping matrix. ACL and IJCNLP. pp. 687-696 (2015)
8. Yang, B., Yih, W.t., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases. arXiv preprint arXiv:1412.6575 (2014)
9. Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., Bouchard, G.: Complex embeddings for simple link prediction. ICML. pp. 2071-2080 (2016)
10. Cai, L., Wang, W.Y.: KBGAN: Adversarial Learning for Knowledge Graph Embeddings. NAACL. pp. 1470-1480 (2018)
11. Wang, P., Li, S., Pan, R.: Incorporating GAN for negative sampling in knowledge representation learning. AAAI. pp. 2005-2012 (2018)
12. Wang, Z., Zhang, J., Feng, J., Chen, Z.: Knowledge graph embedding by translating on hyperplanes. AAAI. pp. 1112-1119. (2014)
13. Fan, M., Zhou, Q., Chang, E., Zheng, T.F.: Transition-based knowledge graph embedding with relational mapping properties. PACLIC. pp. 328-337 (2014)
14. Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. AAAI. pp. 2181-2187. (2015)
15. Feng, J., Huang, M., Wang, M., Zhou, M., Hao, Y., Zhu, X.: Knowledge graph embedding by flexible translation. KR. pp. 557-560. (2016)
16. Xiao, H., Huang, M., Zhu, X.: From one point to a manifold: knowledge graph embedding for precise link prediction. IJCAI. pp. 1315-1321. (2016)
17. Tan, Z., Zhao, X., Fang, Y., Xiao, W.: GTrans: generic knowledge graph embedding via multi-state entities and dynamic relation spaces. IEEE Access **6**, 8232-8244 (2018)
18. Nickel, M., Tresp, V., Kriegel, H.P.: A three-way model for collective learning on multi-relational data. ICML. pp. 809-816. (2011)
19. Kazemi, S.M., Poole, D.: SimplE embedding for link prediction in knowledge graphs. NIPS. pp. 4284-4295 (2018)
20. Balažević, I., Allen, C., Hospedales, T.M.: TuckER: Tensor Factorization for Knowledge Graph Completion. arXiv preprint arXiv:1901.09590 (2019)
21. Socher, R., Chen, D., Manning, C.D., Ng, A.: Reasoning with neural tensor networks for knowledge base completion. NIPS. pp. 926-934 (2013)
22. Dong, X., Gabrilovich, E., Heitz, G.,et al.: Knowledge vault: A web-scale approach to probabilistic knowledge fusion. ACM SIGKDD. pp. 601-610 (2014)
23. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. NIPS. pp. 2672-2680 (2014)
24. Sutton, R.S., McAllester, D.A., Singh, S.P., Mansour, Y.: Policy gradient methods for reinforcement learning with function approximation. NIPS. pp. 1057-1063 (2000)
25. Toutanova, K., Chen, D., Pantel, P., et al.: Representing text for joint embedding of text and knowledge bases. EMNLP. pp. 1499-1509 (2015)
26. Dettmers, T., Minervini, P., Stenetorp, P., Riedel, S.: Convolutional 2d knowledge graph embeddings. AAAI. pp. 1811-1818 (2018)
27. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)