NEGATIVE SAMPLING IN KNOWLEDGE REPRESENTATION LEARNING: A MINI-REVIEW

Jing Qian[1, 2], Gangmin Li[1], Katie Atkinson[2] and Yong Yue[1]

[1]Department of Intelligent Science, School of Advanced Technology,
Xi'an Jiaotong-Liverpool University, Suzhou, Jiangsu Province, China
[2]Department of Computer Science, University of Liverpool,
Liverpool, United Kingdom

ABSTRACT

Knowledge representation learning (KRL) aims at encoding components of a knowledge graph (KG) into a low-dimensional continuous space, which has brought considerable successes in applying deep learning to graph embedding. Most famous KGs contain only positive instances for space efficiency. Typical KRL techniques, especially translational distance-based models, are trained through discriminating positive and negative samples. Thus, negative sampling is unquestionably a non-trivial step in KG embedding. The quality of generated negative samples can directly influence the performance of final knowledge representations in downstream tasks, such as link prediction and triple classification. This review summarizes current negative sampling methods in KRL and we categorize them into three sorts, fixed distribution-based, generative adversarial net (GAN)-based and cluster sampling. Based on this categorization we discuss the most prevalent existing approaches and their characteristics.

KEYWORDS

Knowledge Representation Learning, Negative Sampling, Generative Adversarial Nets.

1. INTRODUCTION

A knowledge graph (KG) is essentially a structural approach to tell facts. It refers to a network whose nodes are real entities or abstract concepts and edges are their in-between relations. Many KGs have gained steady development, such as NELL [1], Freebase [2] and YAGO [3]. They store and express ground-truth facts in the form of a triple (head entity, relation, tail entity) or (subject, predicate, object). Inspired by word embedding [4], people turned to distributed representation of entities and relations instead of one-hot representation that benefits the storage of triples but fails in capturing latent semantics.

Knowledge representation learning (KRL) is also known as knowledge graph embedding (KGE), it attempts to embed entities and relations in the KG into low-dimensional vector space. In recent years, a variety of KRL models have been successively proposed and deployed. Looking at conventional translational distance-based TransE [5], semantic matching-based RESCAL [6] or the state-of-the-art attention-based KBAT [7] and GAATs [8], they aim to learn better knowledge representations to serve knowledge graph completion tasks. KRL models define their own scoring functions on account of different embedding modes, which returns a score to measure the plausibility of the given triple. Mikolov et al. [4] simplifies noise contrastive estimation (NCE)

David C. Wyld et al. (Eds): ACITY, DPPR, VLSI, WeST, DSA, CNDC, IoTE, AIAA, NLPTA - 2020
pp. 229-238, 2020. CS & IT - CSCP 2020 DOI: 10.5121/csit.2020.101519

[9] to negative sampling with the aim of reducing computational complexity. KRL extends this strategy that ranks observed (“positive”) instances higher than unobserved (“negative”) ones [10]. As seen in translational distance-based models [5, 11-14], they are optimized through partitioning scores of positives and negatives with an adaptive margin. A large number of negative samples are required in training KRL models. However, most KGs only store ground-truth triples, for the sake of space efficiency. Negative sampling thus plays a pivotal role in the training process. The widely-used negative sampling method is uniform sampling[5, 12], which replaces the head or tail entity of the positive triple with an entity that is uniformly sampled from the entity set of the KG. Nevertheless, such generated negative triples are too obviously incorrect and contribute less as the training goes on, in most cases. Bernoulli sampling[11] applies different probabilities in head and tail replacement to alleviate the problem of false-negative triples. KBGAN [15] and IGAN [16] adversarially train the generator to provide better-quality negatives by applying a pre-trained KRL model as the discriminator. TransE-SNS [17] and NSCaching [18] carry out negative sampling in a more concentrated way. Furthermore, enlightened by CKRL [19], NKRL [20] puts forward a confidence-aware negative sampling method. Yang et al. [21] recently derives the general form of an effective negative sampling distribution, which is of pioneering significance. They are the first to deduce the correlation between positive and negative sampling distribution. Trouillon et al. [22] further studies the number of negatives generated for each positive triple, and elicits that fifty negative samples per positive is a good choice for balancing accuracy and training time.

In this review, we summarize current negative sampling methods and divide them into three categories, sampling from fixed distribution, sampling from GAN-based framework and sampling from custom cluster. Most KRL research focuses on proposing new embedding methods or their applications in downstream tasks, such as knowledge graph completion [23], question-answering [24] and recommendation [25]. Little attention is paid to negative sampling, although it is an influential and crucial step in KRL model training. In KRL surveys [26, 27], negative sampling is mentioned but only in a short space. To the best of our knowledge, this review is the first work to systematically and exhaustively overview existing negative sampling methods in KRL.

Around twelve negative sampling techniques applied in KRL are summarized in our work. Definitions and notations, as well as two necessary assumptions before modelling, are briefly covered in Section 2. Developments in KRL are presented in Section 3, in which we sort KRL models from four perspectives as is routine. Negative sampling is elaborated in Section 4 and this presents our main contribution. Finally, this review finishing with a conclusion and future research directions.

2. DEFINITIONS, NOTATIONS AND ASSUMPTIONS

In a standard KG, $\mathbb{E}$ represents the set of entities, $\mathbb{R}$ represents the set of relations. $\mathbb{D}^+$ and $\mathbb{D}^-$ are sets of the positive triples $\tau^+ = (h, r, t)$ and the counterpart negative triples respectively. The following formula sets out the components of the set $\mathbb{D}^-$. In general cases, one KRL model can be explained by its own-defined scoring function $f_r(h, t)$ where h and t belong to $\mathbb{E}$ and r belongs to $\mathbb{R}$. The relation r maps the head entity h to its tail entity t. The plausibility of each possible triple is measured by the scoring function. The higher the plausibility is, the more probability for the triple being a piece of truth.

$$\tau^- \in \mathbb{D}^-$$

$$\begin{aligned}\mathbb{D}^{-} = &\left\{(h',r,t)\middle|h' \in \mathbb{E} \bigwedge h' \neq h \bigwedge (h,r,t) \in \mathbb{D}^{+}\right\}\\ &\cup \left\{(h,r,t')\middle|t' \in \mathbb{E} \bigwedge t' \neq t \bigwedge (h,r,t) \in \mathbb{D}^{+}\right\}\\ &\cup \left\{(h,r',t)\middle|r' \in \mathbb{D} \bigwedge r' \neq r \bigwedge (h,r,t) \in \mathbb{D}^{+}\right\}\end{aligned}$$

KRL models are trained under the open world assumption (OWA) [28] or the closed world assumption (CWA) [29]. The CWA states that facts that are not observed in $\mathbb{D}^{+}$ are false, while the OWA is relaxed to assume that unobserved facts can be either missing or false. Most models prefer the OWA due to the incompleteness nature of KGs. The CWA has two main drawbacks, worse performance in downstream tasks and scalability issues caused by tremendous negative samples [26].

3. KRL Models

The goal of KRL is to embed triples (h,r,t) into a low-dimensional continuous vector space. A scoring function $f_r(h,t)$is manually defined to calculate the credibility score for the given triple. Different models embed semantic information into the vector representation of the KG in different ways [26]. By convention, there are mainly two types of KRL models, the Translational Distance-based and the Semantic Matching-based. In recent years, neural networks and additional information (entity type, path, text, etc.) have also been considered.

Translational distance-based models. The main idea of the translation-based models is to measure the distance between the head entity and the tail entity after triples in the KG are vectorized. Inspired by translation invariance in word vectors, TransE [5] considers the relation vector as a transition from the head to the tail, i.e. $h + r \approx t$. A short distance between $h + r$ and t reflects high credibility of the given triple. TransH [11] improves TransE to make it more applicable for modeling complex relations, like one-to-many and many-to-many. Some other variants extend TransE by projecting the embedding vectors of entities into various spaces, such as TransR [12], TransD [13] and TransG [14].

Semantic matching-based models. Compared to translational distance-based models, semantic matching-based models pay more attention to the latent semantics embodied in vectorized entities and relations. They are also called matrix decomposition models. RESCAL [6] is one of the earlier works that defines the scoring function based on semantic matching, in which, the relation vectors compose the mapping matrix $\mathrm{M_r}$ between the head and the tail, and the matrix product $h\mathrm{M_r}t$ is used to measure the plausibility of triples. DistMult [30] simplifies RESCAL by limiting $\mathrm{M_r}$ to be a diagonal matrix, while ComplEx [22] extends DistMult to the complex field to build an antisymmetric-relation model.

Neural network-based models. Applying neural networks in KRL has also seen steady progresses. MLP [31] feeds entities and relations into a fully-connected layer to encode semantic matching. ConvE [32] attempts to fit the scoring function using 2D convolution. By distinguishing relations and entities, RSN [33] introduces a recurrent skip mechanism. KG-BERT [34] is based on Transformer (BERT) to integrate KRL and language model pre-training. Referring to graph neural networks (GNNs), R-GCN [35] is the pioneer to encode relational data with the graph convolutional network framework.

Auxiliary-dependent models. Some work suggests incorporating additional information for improvement. Guo et al. [36] considers the entity type to be an extra piece of information and assumes that entities of the same type ought to be closer in vector representation. PTransE [37] attempts to describe multi-hop relations between entities through addition, multiplication and RNN rules so that the relation paths between entities can be represented by the vector calculations of relations. In addition, Wang et al. [38] introduces a joint model adding the text information in the embedding process, and Guo et al. [39] comes up with a rule-based KRL model combining some rule information.

All the above models require negative samples during training. Before explaining the necessity of negative sampling in KRL, its roles in word embedding ought to be mentioned. Both word embedding and KGE belong to the scope of unsupervised learning. The softmax function has conventionally served as the training objective that approximately maximizes its log probability by normalizing with respect to all words in the dictionary, which is highly inefficient and computationally expensive. Negative sampling is proposed to simplify the computation. Instead of estimating the probability distribution based on the whole dictionary, the final representations can be obtained through distinguishing the positive sample from a few negative samples that are generated by perturbing the positive one. In view of random walk over graphs, graph structured data is similar to natural language, where nodes are as words and links as context. KRL adopts negative sampling that is trained by discriminating from a preset number of negative samples, rather than modelling conditional on all nodes.

It is noticed that poor negative samples can be easily discriminated and helpless for training. However, most KRL studies center on embedding modes and simply apply uniform sampling to generate negative training triples [26]. At present, only a few works have been devoted to improving the quality of negatives. We outline these methods with the aim of gaining more attention to this field. Besides, conventional and the state-of-the-art KRL models, their applications and future trends, can be found in the representative surveys [26, 27, 40].

4. NEGATIVE SAMPLING

Negative sampling was first proposed in neural probabilistic language models and labelled as importance sampling [41]. Mikolov et al. [4] emphasizes it as a simplified version of NCE [9] to benefit the training of word2vec. Extended by word embedding, KGE also takes negative sampling as the prerequisite for the model training. Poor or too obviously false negatives are hard to capture for latent semantics in the KG and easily cause the zero loss problem as well. Conversely, generating better-quality negative triples will facilitate both the smooth running of the training and the learned embeddings getting desired performance in assessment tasks. Recognising the importance, benefit and standard of negative sampling, many methods have been proposed and many approaches have been tested. We survey the existing strategies and present them in the following schema.

4.1. Fixed distribution-based sampling

Negative sampling methods of this category are broadly used due to their simplicity and efficiency. However, the ignorance of changes over the negative sample distribution can easily result in the vanishing gradient problem and impede the model training.

4.1.1. Uniform sampling

Uniform sampling [5]is the earliest, easiest and most widely-used negative sampling method in KRL. It refers to constructing negative triples by replacing either the head h or the tail t of a positive triple with the entity randomly sampled from the entity set $\mathbb{E}$ according to uniform distribution. However, in most cases, the uniformly sampled entity is unrelated with the corrupted positive triple, then the formed negative triple is too wrong to facilitate the training. Taking the triple (*London, locatedIn, UnitedKingdom*) as an example, its tail entity *UnitedKingdom* needs to be replaced to produce counterpart negative triples. Under the uniform sampling schema, the generated negatives could be (*London, locatedIn, apple*) or (*London, locatedIn, football*). These low-quality triples will be easily discriminated by the KRL model merely in terms of different entity types, which can slow down the convergence [42]. Similarly, IGAN emphasizes the zero loss problem in the random sampling mode, and explains the little contribution made by the low-quality negatives. Translation-based KRL models prefer adopting a marginal loss function with a fixed margin to distinguish positive triples from negative ones. Unreliable negatives tend to be out of the margin, which easily results in zero loss. Another severe drawback of uniform sampling lies in false-negative samples. After replacing the head in (*DonaldTrump, Gender, Male*) with *JoeBiden*, *(JoeBiden, Gender, Male)* is still a true fact (false negative).

4.1.2. Bernoulli sampling

To alleviate the false negatives problem, Bernoulli negative sampling [11] suggests replacing head or tail entities with different probabilities according to the mapping property of relations. That is, to give more chance of replacing the head in one-to-many relations and the tail in many-to-one relations. *Gender* is a typical many-to-one relation. Replacing the tail in (*DonaldTrump, Gender, Male*) with high probability unlikely cause false negative triples. If setting constraints on entity type, it may generate high-quality negatives. Zhang et al. [43] extends Bernoulli sampling by considering relation replacement following the probability $\alpha = r/(r+e)$, here r is the number of relations and e is the number of entities. The rest $1-\alpha$ is divided by head entity replacement and tail entity replacement according Bernoulli distribution. Such changes enhance the ability of KRL models in relation link prediction.

4.1.3. Probabilistic sampling

Kanojia et al. [44]proposes probabilistic negative sampling to address the issue of skewed data that commonly exists in knowledge bases. For relations with less data, Uniform or Bernoulli random sampling fails to predict the missing part of golden triplets among semantically possible options even after hundreds of epochs of training. Probabilistic negative sampling speeds up the process of generating corrupted triplets by bringing in a tuning parameter β known as train bias that determines the probability by which the generated negative examples are complemented with early-listed possible instances. Kanojia et al. evaluates probabilistic negative sampling (PNS) over TransR in link prediction, and elicits that TransR-PNS achieves 190 and 47 position gains in Mean Rank on benchmark datasets WN18 and FB15K [5] respectively compared to TransR using Bernoulli sampling.

4.2. GAN-based sampling

GAN is short for Generative Adversarial Network [45]. In the GAN-based framework, the generator is responsible for providing negative samples and the discriminator is the target KRL model. Adversarial training is going on between the generator and the discriminator to optimize final knowledge representations. Reinforcement learning is required for training GAN [18]. The framework can be performed on various KRL models as it is independent of the specific form of the discriminator [16]. GAN is capable of modelling dynamic distribution, its generator has advantages in providing negative samples with better quality consistently. However, potential risks (training instability and model collapse) embodied in reinforcement learning should not be neglected.

4.2.1. KBGAN

KBGAN [15] is the first work to adapt GAN to negative sampling in KRL. It considers selecting one of two translational distance-based KRL models (DistMult [30], ComplEx [22]) as the negative sample generator and one of two semantic matching-based KRL models (TransE [5], TransD [13]) as the discriminator for adversarial training. The generator produces a probability distribution over a candidate set of negatives and selects the one with highest probability to feed into the discriminator. The discriminator minimizes the marginal loss between positive and negative samples to learn the final embedding vectors. KBGAN combines four Generator-Discriminator pairs that show better performance than baselines, which reflects the strength of the adversarial learning framework.

4.2.2. IGAN

Different from that of KBGAN [15]which considers probability-based, log-loss KRL models as the generator, IGAN [16] applied a two-layer fully-connected neural network as its generator to supply better quality negative samples. The discriminator is still the desired KRL model. The embedding vectors of the corrupted positive triple are fed into the neural network and followed by non-linear activation function ReLU. The softmax function is added after to calculate the probability distribution over the whole entity set $\mathbb{E}$ instead of a small candidate set in KBGAN. The quality of the formed negative is measured by the scoring function of the discriminator. IGAN can mine negative samples with relatively high quality during adversarial training but suffers from high computational complexity.

Comparison between GAN-based and self-adversarial sampling. Adversarial Contrastive Estimation (ACE) [46] introduces a general adversarial negative sampling framework for NCE that is commonly used in NLP. RotatE [47] thinks that such adversarial framework is difficult to optimize since it needs to train the discrete negative sample generator and the embedding model simultaneously, which costs a lot in computation. GAN-based sampling has no advantage in efficiency. In order to reduce the risk of training instability caused by reinforcement learning, both KBGAN and IGAN requires to be pre-trained, which gives rise to extra costs. Therefore, RotatE proposes a self-adversarial sampling method based on self-scoring function and avoids the requirement of reinforcement learning. Meanwhile, it outperforms KBGAN in link prediction.

4.3. Custom cluster-based sampling

Sampling from custom clusters means that the desired negative sample is selected from a handful of candidates rather than sampled from the whole entity set. For example, domain sampling [48] suggests to sample from the same domain, and affinity dependent sampling relies on the closeness of entities that are measured by cosine similarity. Two more sampling methods, TransE-SNS and NSCaching, are elaborated in this section. Reducing the sampling scope makes the target of negative sampling more clear, which gains efficiency. Because KGs grow rapidly and update frequently, the constant renewal of custom clusters is essential and skilled.

4.3.1. TransE-SNS

Qin et al. [17] puts forward entity similarity-based negative sampling (SNS) to mine valid negatives. Inspired by the observation that smaller distance between two entity vectors imply their higher similarity in the embedding space, the K-Means clustering algorithm [49] is used to divide all entities into a number of groups. An entity is uniformly sampled from the same cluster of the replaced head entity to complete the corrupted positive triple and when necessary, the tail entity is replaced in the same manner. The negatives generated in a such way should be highly similar to the given positive triple. Adapting SNS to TransE (TransE-SNS) and then evaluating in link prediction and triple classification, demonstrates that SNS enhances the ability of TransE.

4.3.2. NSCaching

High-quality negative samples tend to get high plausibility measured by scoring functions. Motivated by the skewed score distribution of negative samples, Zhang et al. [18]attempts to only track helpful and rare negatives of high plausibility with cache. NSCaching can be considered to be in the same group of GAN-based methods since they all parametrize the dynamic distribution of negative samples. To be precise, NSCaching is a distilled version of GAN-based methods, because it has fewer parameters, it does not need to be trained through reinforcement learning, and it also avoids the model collapse problem brought by GAN. After storing the high-quality negative triples in cache, NSCaching uniformly samples from the cache and applies importance sampling to update it. With more concentrated sampling and more concise training, NSCaching performs better than GAN-based methods in terms of efficiency and effectiveness.

4.4. Other novel approaches

We find that there are some novel negative sampling methods that cannot be simply classified into the above three categories, such as confidence-aware negative sampling [20] and Markov chain Monte Carlo negative sampling [21].

4.4.1. NKRL

Since human knowledge is innumerable and changeable, bypassing crowdsourcing and manual efforts in building KGs is the mainstream. Noise and conflicts are inevitably involved due to the auto-construction, explosive growth and frequent updates of typical KGs. Xie et al. [19] initially proposes a novel confidence-aware KRL framework (CKRL), and Shan et al. [20]extends this idea to negative sampling in noisy KRL (NKRL). CKRL detects noises but applies uniform negative sampling that easily causes zero loss problems and false detection issues. NKRL proposes a confidence-aware negative sampling method to address these problems, and the concept of negative triple confidence it introduces is conducive to generate plausible negatives by measuring their quality. NKRL also modifies the triple quality function defined in CKRL with the aim of alleviating the false detection problems and improving noise detection ability. Both CKRL and NKRL are performed on translation-based KRL models, and NKRL outperforms CKRL in link prediction task.

4.4.2. MCNS

Yang et al. [21] creatively derives that a nice negative sampling distribution that should be positively but sub-linearly correlated to the positive sampling distribution, and raises Markov chain Monte Carlo negative sampling (MCNS). In the proposed SampledNCE framework, the depth first search (DFS) algorithm is applied to traverse the graph to obtain the Markov chain of the last node, from which negative samples are generated. MCNS uses the self-contrast approximation to estimate positive sampling distribution, and the Metropolis-Hastings algorithm [50] to speed up negative sampling. Embedding vectors are updated by minimizing the hinge loss after inputting the positive sample and the generated negative sample into the encoder of the framework. The importance of negative sampling is proved in the formula derivation. Experiments exhibit that MCNS performs better than all baselines in downstream tasks and wins in terms of efficiency. The proposal of MCNS is based on graph structure models without limitation to KRL, which is a generic solution.

5. CONCLUSIONS

In this short paper we have reviewed negative sampling in KRL. We sketched out existing well known negative sampling methods in three categories. We aimed to provide a basis for selecting the proper negative sampling method to train a KRL model to its best. The majority of KRL studies focus on defining new scoring functions to model multi-relational data in KGs, thus simply selecting the random mode for negative sampling. Nevertheless, as another momentous perspective of KRL, negative sampling is of the same significance with positive sampling. We hope that this review can be of some help to those who are interested in negative sampling. Subsequent work lies in comparing the methods mentioned here by performing link prediction on benchmark datasets. Besides, proposing a new strategy for negative sampling is a challenging attempt but is also under consideration.

ACKNOWLEDGEMENTS

This work is supported by the AI University Research Centre (AI-URC) of Xi’an Jiaotong-Liverpool University through XJTLU Key Programme Special Fund KSF-P-02 and KSF-A-17. We appreciate their support.

REFERENCES

[1] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. Hruschka, and T. M. Mitchell, "Toward an Architecture for Never-Ending Language Learning," (in English), *Proceedings of the Twenty-Fourth Aaai Conference on Artificial Intelligence (Aaai-10),* pp. 1306-1313, 2010.

[2] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: a collaboratively created graph database for structuring human knowledge," in *SIGMOD Conference*, 2008.

[3] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: a core of semantic knowledge," in *WWW '07*, 2007.

[4] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," presented at the Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, Lake Tahoe, Nevada, 2013.

[5] A. Bordes, N. Usunier, A. Garcia-Durán, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," presented at the Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, Lake Tahoe, Nevada, 2013.

[6] M. Nickel, V. Tresp, and H.-P. Kriegel, "A three-way model for collective learning on multi-relational data," presented at the Proceedings of the 28th International Conference on International Conference on Machine Learning, Bellevue, Washington, USA, 2011.

[7] D. Nathani, J. Chauhan, C. Sharma, and M. Kaul, "Learning Attention-based Embeddings for Relation Prediction in Knowledge Graphs," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, 2019, pp. 4710-4723: Association for Computational Linguistics.

[8] R. Wang, B. Li, S. Hu, W. Du, and M. Zhang, "Knowledge Graph Embedding via Graph Attenuated Attention Networks," *IEEE Access,* vol. 8, pp. 5212-5224, 2020.

[9] M. U. Gutmann and A. Hyvärinen, "Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics," *J. Mach. Learn. Res.,* vol. 13, no. 1, pp. 307–361, 2012.

[10] B. Kotnis and V. Nastase, "Analysis of the Impact of Negative Sampling on Link Prediction in Knowledge Graphs," 08/22 2017.

[11] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," presented at the Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, Québec City, Québec, Canada, 2014.

[12] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning Entity and Relation Embeddings for Knowledge Graph Completion," in *AAAI*, 2015.

[13] G. Ji, S. He, L. Xu, K. Liu, and J. Zhao, "Knowledge Graph Embedding via Dynamic Mapping Matrix," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*

and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Beijing, China, 2015, pp. 687-696: Association for Computational Linguistics.

[14] H. Xiao, M. Huang, and X. Zhu, "TransG : A Generative Model for Knowledge Graph Embedding," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany, 2016, pp. 2316-2325: Association for Computational Linguistics.

[15] L. Cai and W. Y. Wang, "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, Volume 1 (Long Papers)*, New Orleans, Louisiana, 2018, pp. 1470-1480: Association for Computational Linguistics.

[16] P. Wang, S. Li, and R. Pan, "Incorporating GAN for Negative Sampling in Knowledge Representation Learning," in *AAAI*, 2018.

[17] S. Qin, G. Rao, C. Bin, L. Chang, T. Gu, and W. Xuan, "Knowledge Graph Embedding Based on Adaptive Negative Sampling," Singapore, 2019, pp. 551-563: Springer Singapore.

[18] Y. Zhang, Q. Yao, Y. Shao, and L. Chen, "NSCaching: Simple and Efficient Negative Sampling for Knowledge Graph Embedding," in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, 2019, pp. 614-625.

[19] R. Xie, Z. Liu, and M. Sun, "Does William Shakespeare REALLY Write Hamlet? Knowledge Representation Learning with Confidence," *ArXiv,* vol. abs/1705.03202, 2018.

[20] Y. Shan, C. Bu, X. Liu, S. Ji, and L. Li, "Confidence-Aware Negative Sampling Method for Noisy Knowledge Graph Embedding," *2018 IEEE International Conference on Big Knowledge (ICBK),* pp. 33-40, 2018.

[21] Z. Yang, M. Ding, C. Zhou, H. Yang, J. Zhou, and J. Tang, "Understanding Negative Sampling in Graph Representation Learning," *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining,* 2020.

[22] T. Trouillon, J. Welbl, S. Riedel, É. Gaussier, and G. Bouchard, "Complex embeddings for simple link prediction," presented at the Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48, New York, NY, USA, 2016.

[23] S. Kazemi and D. Poole, "SimplE Embedding for Link Prediction in Knowledge Graphs," in *NeurIPS*, 2018.

[24] X. Huang, J. Zhang, D. Li, and P. Li, "Knowledge Graph Embedding Based Question Answering," *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining,* 2019.

[25] X. Wang, X. He, Y. Cao, M. Liu, and T.-S. Chua, "KGAT: Knowledge Graph Attention Network for Recommendation," *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining,* 2019.

[26] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge Graph Embedding: A Survey of Approaches and Applications," *IEEE Transactions on Knowledge and Data Engineering,* vol. 29, no. 12, pp. 2724-2743, 2017.

[27] S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. S. Yu, "A Survey on Knowledge Graphs: Representation, Acquisition and Applications," *ArXiv,* vol. abs/2002.00388, 2020.

[28] L. Drumond, S. Rendle, and L. Schmidt-Thieme, "Predicting RDF triples in incomplete knowledge bases with tensor factorization," 03/26 2012.

[29] R. Reiter, "Deductive Question-Answering on Relational Data Bases," in *Logic and Data Bases*, H. Gallaire and J. Minker, Eds. Boston, MA: Springer US, 1978, pp. 149-177.

[30] B. Yang, W.-t. Yih, X. He, J. Gao, and L. Deng, "Embedding Entities and Relations for Learning and Inference in Knowledge Bases," *CoRR,* vol. abs/1412.6575, 2015.

[31] X. Dong *et al.*, "Knowledge vault: A web-scale approach to probabilistic knowledge fusion," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 08/24 2014.

[32] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "Convolutional 2D Knowledge Graph Embeddings," *ArXiv,* vol. abs/1707.01476, 2018.

[33] L. Guo, Z. Sun, and W. Hu, "Learning to Exploit Long-term Relational Dependencies in Knowledge Graphs," in *ICML*, 2019.

[34] L. Yao, C. Mao, and Y. Luo, "KG-BERT: BERT for Knowledge Graph Completion," *ArXiv,* vol. abs/1909.03193, 2019.

[35] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling, "Modeling Relational Data with Graph Convolutional Networks," Cham, 2018, pp. 593-607: Springer International Publishing.

[36] S. Guo, Q. Wang, B. Wang, L. Wang, and L. Guo, "Semantically Smooth Knowledge Graph Embedding," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Beijing, China, 2015, pp. 84-94: Association for Computational Linguistics.

[37] Y. Lin, Z. Liu, H. Luan, M. Sun, S. Rao, and S. Liu, "Modeling Relation Paths for Representation Learning of Knowledge Bases," *ArXiv,* vol. abs/1506.00379, 2015.

[38] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge Graph and Text Jointly Embedding," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, 2014, pp. 1591-1601: Association for Computational Linguistics.

[39] S. Guo, Q. Wang, L. Wang, B. Wang, and L. Guo, "Knowledge Graph Embedding with Iterative Guidance from Soft Rules," *ArXiv,* vol. abs/1711.11231, 2018.

[40] Y. Lin, X. Han, R. Xie, Z. Liu, and M. Sun, "Knowledge Representation Learning: A Quantitative Review," *ArXiv,* vol. abs/1812.10901, 2018.

[41] Y. Bengio and J. Senecal, "Adaptive Importance Sampling to Accelerate Training of a Neural Probabilistic Language Model," *IEEE Transactions on Neural Networks,* vol. 19, no. 4, pp. 713-722, 2008.

[42] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 815-823.

[43] Y. Zhang, W. Cao, and J. Liu, "A Novel Negative Sample Generating Method for Knowledge Graph Embedding," presented at the Proceedings of the 2019 International Conference on Embedded Wireless Systems and Networks, Beijing, China, 2019.

[44] V. Kanojia, H. Maeda, R. Togashi, and S. Fujita, "Enhancing Knowledge Graph Embedding with Probabilistic Negative Sampling," *Proceedings of the 26th International Conference on World Wide Web Companion,* 2017.

[45] I. J. Goodfellow *et al.*, "Generative adversarial nets," presented at the Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, Montreal, Canada, 2014.

[46] A. Bose, H. Ling, and Y. Cao, "Adversarial Contrastive Estimation," *ArXiv,* vol. abs/1805.03642, 2018.

[47] Z. Sun, Z.-H. Deng, J.-Y. Nie, and J. Tang, "RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space," *ArXiv,* vol. abs/1902.10197, 2019.

[48] Q. Xie, X. Ma, Z. Dai, and E. Hovy, "An Interpretable Knowledge Transfer Model for Knowledge Base Completion," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada, 2017, pp. 950-962: Association for Computational Linguistics.

[49] J. Hartigan and M. C. Wong, "Statistical algorithms: algorithm AS 136: a K-means clustering algorithm," 1979.

[50] N. Metropolis, A. W. Rosenbluth, M. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *Journal of Chemical Physics,* vol. 21, pp. 1087-1092, 1953.