Validation and Optimization of Link Prediction in Knowledge Graph Embeddings through Relationship Prediction

Yuhu Tao, Huimin Li*, Jianxiao Wang, Lulu Xue
School of Mathematics and Computer Science, Yunnan Minzu University, Kunming, China
**Corresponding Author.*

Abstract: Common inference tasks in knowledge graphs include link prediction, relation prediction, and entity alignment. Knowledge graph embedding (KGE) has demonstrated its effectiveness for these tasks, with numerous KGE models achieving significant results in this domain. Nevertheless, given the intricate relation patterns in knowledge graphs, KGE models frequently show constrained reasoning capabilities, especially in link prediction. Notably, most KGE models have demonstrated satisfactory performance in relation prediction tasks. Motivated by this observation, this paper evaluates and analyzes relation prediction results for two common KGE models and proposes a novel inference method. Our method leverages relation prediction scores to support and optimize KGE models' link prediction abilities. Comprehensive and rigorous experiments validate our methodology, achieving competitive results across both benchmark datasets.

Keywords: Knowledge Graph Embedding; Relation Prediction; Link Prediction

1. Introduction

Knowledge graphs (KGs) are collections of factual triples, each consisting of two entities and the relations connecting them. Notable instances of KGs include WordNet[1]and Freebase[2]. Currently, knowledge graphs are widely applied across numerous fields. However, knowledge graphs often contain missing relations, creating a need for more comprehensive knowledge representations. To address this, knowledge graph reasoning tasks, including relation prediction and link prediction, have emerged as major challenges in the field of KGs.

The Knowledge Graph Embedding (KGE) model provides an approach to this task, employing a specified scoring function to derive vector-based embeddings for entities and relations from a given set of triples, thereby enabling the prediction of missing links. Considerable success in predicting links within KGs has been achieved by several KGE models, including HAKE[3], TransE[4], PairRE[5], and Rotate[6], among others.

To evaluate the effectiveness of KGE models across various relational patterns, we initially examined the capability of various KGE models in relation prediction tasks. The experimental results indicated that these models exhibit high accuracy in relation prediction, demonstrating their ability to effectively identify the true relations between entities. Based on this observation, we propose our model, which leverages relation prediction to aid in the optimization of link prediction tasks.

Our model primarily aims to improve and refine the reasoning capability of KGE models by incorporating relation prediction results. When applied to link prediction with relaxed conditions—such as considering the prediction successful if the correct sample ranks within the top n among all candidates, where n is relatively large—KGE models have a high probability of successfully completing the link prediction.

However, when we limit n to 1 or 3, the prediction success rate significantly decreases, indicating that the model struggles to identify the correct sample from the top-ranked candidates. It is important to highlight that, within knowledge graphs, relations are vastly outnumbered by entities, and KGE models exhibit robust performance in relation prediction tasks.Therefore, ourmodel first utilizes the KGE model to filter the top-ranked samples, then predicts the relations of the candidate triplets, and incorporates the relation prediction score into the final ranking to enhance and optimize the link prediction outcomes. Experimental evaluations on the FB15K-237 and WN18RR confirm the efficacy and advantages offered by this model.

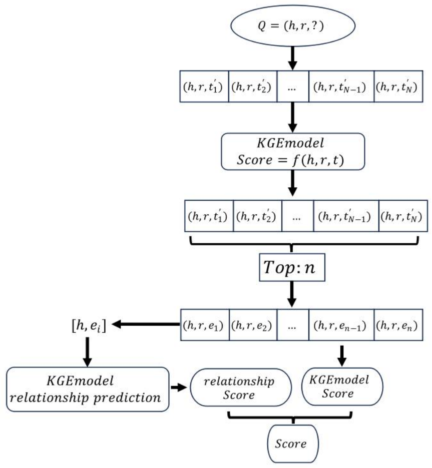

Figure 1. The Flowchart of Our Model

2. Task Definition

2.1 Link Prediction

The main goal of link prediction is to find absent entities in a given knowledge graph triple [7]. For a specified relation and known entity, all possible triples (h, r, e_i) are created by pairing each entity in the KGs with the chosen entity and relation. Each generated triple is subsequently assessed and ranked according to the scoring mechanism of the KGE approach. The triple achieving the highest score is selected as the model's predicted correct match.

2.2 Relation Prediction

Relation prediction is designed to infer the most probable relation that could connect a pair of specified entity pair of head and tail[8]. For a specified head-tail entity pair (h, t), all possible relations in the knowledge graph are paired with this entity pair to create candidate triples (h, r_i, t). These candidates are then evaluated and ranked based on a defined scoring function. The relation in the top-ranked candidate triple is considered the model's predicted outcome.

3. Method

3.1 Training KGE Model

Firstly, we use the scoring function and loss function defined by the KGE model to train vector representations of entities and relations. At this stage, we ulitize two classic KGE models, TransE and RotatE. TransE is one of the earliest and simplest KGE model. This approach posits that the interaction between two entities can be captured as a translation vector, such that for any given (h, r, t), the vector representation for entity t should approximate the result of adding the relation vector r to the representation for entity h. This model has high computational efficiency and performs well on datasets only with simple one-to-one relations datasets, but it may encounter difficulties on datasets containing complex structures such as one-to-many or many-to-many relations. The scoring functionof TransEcan be expressed as:

$$KGE\_\mathrm{Score} = \left| h + r - t \right|_p \tag{1}$$

The RotatE model interprets a relation as a rotation within a complex vector space. The model adeptly captures common properties like symmetry, anti-symmetry, inversion, and composition in knowledge graphs. Such an approach can efficiently manage various relation types and shows robust performance in modeling intricate knowledge graph structures. The scoring function employed by RotatE is defined as follows:

$$KGE\_\mathrm{Score} = \left| h \circ r - t \right|_p \tag{2}$$

where, $\circ$ denotes the vector rotation operation; $p = 1\ or\ p = 2$ indicates the use of $L1$ or $L2$ norm, respectively.

In this process, we apply the negative sampling loss function introduced by the RotatE model to train the relation and entity vector representations across both models. The formulation of the loss function is as follows:

$$Loss = \log \sigma(\gamma - KGE\_Score(pos\_triple)) + \sum_{i=1}^{n} p(h_i', r, t_i') \log \sigma(KGE\_Score(neg\_triple_i) - \gamma) \tag{3}$$

where, pos_triple and neg_triple denote the positive and negative triples, respectively; σ is the sigmoid function, γ is a constant.

3.2 Relation Prediction Score

Our study assesses the performance of two KGE models, TransE and RotatE, in the context of relation prediction using the specified datasets. The findings suggest that, given the comparatively limited set of relations relative to the extensive set of entities within the knowledge graph, KGE models perform effectively in relation prediction tasks. By capitalizing on the strong accuracy of relation prediction, we employ it as supplementary validation to improve link prediction, thus enhancing the effectiveness of link inference within the model.

Unlike existing KGE models, our model first scores and sorts all candidate entities based on a scoring function, and then uses relation prediction to validate the top n candidate triplets in the ranking. For the first n candidate triplets $[(h,r,e_1),(h,r,e_2),\dots,(h,r,e_n)]$, we extract the head and target entities (h,e_i) from each candidate to infer their relations, scoring the candidates based on the ranking (rank(r)) of relation r in the relation prediction outcomes. Specifically, the score assigned to each candidate triples in relation prediction is:

$$Relation\_Score = \begin{cases} 1\,, & if: \mathrm{rank}(r) \le k; \\ 0\,, & if:\ k \le \mathrm{rank}(x) \le m\ ; \\ -1\,, & if: \mathrm{rank}(r) \ge m\ . \end{cases} \quad (4)$$

where, k and m are positive integers that measure the strictness of the relation prediction score.

In the end, we use the link prediction score plus the relation prediction score as the final score for the first n candidate triplets, that is, the final scoring function is:

$$score = KGE\_\mathrm{Score} + Relation\_Score \quad (5)$$

where, KGE_Score and $Relation\_Score$ represent the link prediction score and relation prediction score of KGE model, respectively.

4. Experiments and Analysis

4.1 Datasets

To assess the performance of our model , we perform experiments using two widely recognized benchmark datasets, FB15K-237 and WN18RR. Obtained from Freebase[2], the FB15K-237 dataset is a foundational benchmark in KGs link prediction research. Its data structure is more aligned with real-world challenges, making it a realistic testbed for model evaluation. The WN18RR dataset, derived from WordNet [1], retains the semantic relations between words while removing simple inverse relations. This dataset is ideal for evaluating how well KG models generalize and capture complex relational patterns. Table1 summarizes the statistical data of these two datasets.

Table 1. Statistical Information of Datasets

Datasets	#entities	#relations	#training samples	#validation samples	#test samples
FB15K-237	14541	237	272115	17535	20466
WN18RR	40943	11	86835	3034	3134

represents the number.

4.2 Evaluation Metrics

In link prediction of KGs, the Mean Reciprocal Rank(MRR) and Hit@k metrics are frequently employed to assess the effectiveness of models. MRR computes the average of the reciprocal ranks for the correct answers across all test instances. The higher MRR values indicate that the model ranks the correct answer in a higher position. Hit@k evaluates the fraction of instances where the correct answer appears among the top k predicted results. A higher Hit@k value reflects superior predictive performance, while a smaller k signifies a more stringent evaluation. In this study, we use MRR, Hit@1, 3 and 10 as metrics to assess the model's performance on link prediction.

4.3 Relation Prediction Results

We first conduct relation prediction experiments on two classic KGE models, with results shown in Table 2, both models achieved strong performance across the two datasets. with RotatE notably outperforming TransE. In FB15K-237, RotatE achieves MRR of 0.965, and H@10of 0.992, exceeding TransE's MRR of 0.955 and H@10 of 0.983. On WN18RR, RotatE also shows a significant advantage with an MRR of 0.884 and a HIT@10 of 0.992, compared to TransE's MRR of 0.801 and HIT@10 of 0.998. Therefore, in subsequent experiments,we employ the relation and entity embeddings trained with the RotatE model to carry out link prediction tasks.

Table 2. Relation Prediction Results

FB15K-237				
Model	**MRR**	**H@1**	**H@3**	**H@10**
TransE	0.955	0.936	0.970	0.983
RotatE	0.965	0.946	0.982	0.992
WN18RR				
Model	MRR	H@1	H@3	H@10
TransE	0.801	0.739	0.791	0.998
RotatE	0.884	0.837	0.908	0.992

In addition, the RotatE model achieves Hit@3 metrics of 0.982 in FB15K-237 and 0.908 in WN18RR.This suggests that the model has a high probability of ranking the correct relation within the top three. Furthermore, the Hit@10 metric surpasses 99%, indicating that the model is highly confident in placing the correct entity within the top ten. As a result, for the relation prediction component of our model, the values of k and m in Equation(4) are assigned to 3 and 10, respectively. Specifically, if the relation of a candidate triple ranks within the top three for relation prediction, the relation_score is assigned a value of 1, indicating that the relation prediction confirms the candidate triple as correct.When the ranking falls between 4 and 10, the relation_score is set to 0, meaning the relation prediction does not influence the link prediction outcome. If the ranking exceeds 10, the relation_score is set to -1, indicating that the relation prediction suggests the current candidate sample should not be considered as the correct triple.

4.4 Experimental Results

We compare our model with the widely used baseline models for link prediction tasks. The baseline models include TransE [4], Rotate [6], HAKE [3] and PairRE [5]. The performance results for TransE and RotatE are derived from the models we trained, while the results for the other models are taken from their respective published works. Table 3 presents a summary of the experimental outcomes for these baseline models as well as our model across the two datasets.

Table 3. Performance of Different Models on the FB15K−237 and WN18RR Datasets

FB15K-237				
Model	**MRR**	**H@1**	**H@3**	**H@10**
TransE [4]	0.329	0.230	0.368	0.526
RotatE [6]	0.337	0.241	0.374	0.531
HAKE [3]	0.349	0.252	0.385	0.545
PairRE [5]	0.348	0.254	0.384	0.539
Ours	0.379	0.281	0.427	0.570
WN18RR				
model	**MRR**	**H@1**	**H@3**	**H@10**
TransE [4]	0.223	0.014	0.401	0.530
RotatE [6]	0.473	0.427	0.495	0.568
HAKE [3]	0.496	0.452	0.513	0.580
PairRE [5]	0.455	0.413	0.469	0.539
Ours	0.489	0.439	0.507	0.589

The experimental outcomes presented in Table3 reveal that our approach outperforms all comparison models on the FB15K-237 dataset, obtaining scores of 0.379, 0.281, 0.427, and 0.570 across the MRR, H@1, H@3and H@10,achieving the highest performance. On WN18RR, our method surpasses the performance of all baseline models for Hit@10, with scores of 0.589. However, for the other three metrics, our model score 0.489, 0.439 and 0.507, which are only slightly lower than HAKE’s scores of 0.496, 0.452 and 0.513, ranking second. Moreover, compared to the original RotatE model, our method demonstrates significant improvement on the four evaluation metrics:MRR, H@1, H@3 and H@10. Specifically, in the FB15K-237 dataset,our approach achieves enhancements of 4.2%, 4.0%, 5.1%, and 4.1%, respectively, across the four evaluation metrics. On the WN18RR dataset, our model shows improvements of 1.6%, 1.2%, 1.2%, and 2.1% in the corresponding evaluation metrics, respectively.These results indicate that our approach delivers strong performance in tackling the knowledge graph link prediction problem.

5. Conclusion

We introduce a novel model that enhances KGE tasks by incorporating relation prediction to refine and optimize the link prediction process. In our model, a KGE model first filters top-ranking candidate entities, and then applies relation prediction to verify these entities, generating a relation prediction score. The final ranking is determined by combining the scores from both the KGE model and relation prediction. The experimental findings indicate that our model delivers competitive results, consistently outperforming baseline models on most evaluation metrics.

Acknowledgments

This research is supported by Yunnan Province Graduate Quality Course Construction Project: Advanced mathematical statistics (No. Yun Degree [2022] 8) and the Research and Innovation Fund for Postgraduate Students of Yunnan Minzu University(No. 2024SKY128).

References

[1] Miller G A. WordNet: A Lexical Database for English[J]. Communications of ACM, 1995, 38:39-41.

[2] Bollacker K D, Evans C, Paritosh P K, et al. Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge[C]. Proceedings of the

SIGMOD Conference, 2008.
[3] Zhang Z, Cai J, Zhang, Y,et al. Learning Hierarchy-Aware Knowledge Graph Embeddings for Link Prediction[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(03): 3065-3072.
[4] Bordes A,Usunier N, Garcia-Durán A, et al. Translating embeddings for modeling multi-relational data[C]. Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, Lake Tahoe, Nevada, USA, 2013, 2787–2795.
[5] Chao L, He J, Wang T, et al. PairRE: Knowledge Graph Embeddings via Paired Relation Vectors[C]. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2021, 4360－4369.
[6] Sun Z, Deng ZH, Nie JY, et al. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space[C]. International Conference on Learning Representations, 2019.
[7] Zhu T, Tan H, Chen X Y, et al. A Transformer-based Knowledge Graph Embedding Model Combining Graph Paths and Local Neighborhood[C]. Proceedings of the 2024 International Joint Conference on Neural Networks (IJCNN), 2024, 1-9.
[8] Zhang W, Du T, Yang C, et al. A Multimodal Knowledge Graph Representation Learning Method Based on Hyperplane Embedding[C].2024 7th International Conference on Advanced Algorithms and Control Engineering (ICAACE), Shanghai, China, 2024, 434-437.