Role Semantics-Aware Multi-Relational Link Prediction Model

Anonymous ACL submission

Abstract

In the realm of knowledge graph completion, multi-relational link prediction presents a formidable challenge due to the intricate interactions and implicit information shared between entities. Many existing methodologies predominantly build upon binary relationship models, often overlooking the distinct roles that entities assume in multi-relational contexts and their subsequent interactions. To bridge this gap, we introduce a novel approach termed Role Semantic-Aware multi-relational link prediction (RSP). This model emphasizes the roles that entities assume within relationships, leveraging key-value pairs of entity roles to encapsulate the implicit information and profound semantics inherent in multi-relational data. By decomposing intricate relationships into multiple sub-relationships, RSP adeptly captures nuanced interactions between entities. Furthermore, it employs projection matrices to amalgamate role and entity embedding vectors, thereby effectively discerning the complex links between entities. Empirical evaluations on two publicly accessible multi-relational datasets reveal that RSP surpasses other benchmark models, particularly demonstrating superior performance when handling multi-relational data with heightened density.

1 Introduction

Knowledge graphs objectively describe facts in the real world through the form of triples (Ji Shaoxiong, et al.). In recent years, their application has expanded across various domains such as information retrieval, question-answering systems, and recommendation systems, thereby significantly enhancing system intelligence. However, despite their capacity to store vast amounts of structured data, knowledge graphs frequently encounter the issue of incomplete entity relations. This limitation can impede their expressiveness and potentially undermine the performance of subsequent tasks. To address this challenge, Knowledge Graph Completion (KGC) has emerged as a crucial research direction within the field.

Link prediction is a key task in Knowledge Graph Completion (KGC), aiming to uncover potential relationships between missing entities (Wang et al., 2021). Within a knowledge graph, incomplete triples are conventionally depicted as (?, r, t), (h, ?, t), or (h, r, ?). Through the examination of known entities and relations, link prediction is capable of capturing both direct interactions and implicit semantic connections, thereby facilitating the inference of concealed relationships. This not only replenishes missing information, thereby augmenting the graph's richness and accuracy, but also enhances its efficacy in applications such as intelligent question answering, recommendation systems, and information retrieval.

Early link prediction tasks primarily focused on binary relations, where independent triples are typically used to store entities and their relationships (Rosso et al., 2020). For instance, (Alice, friend, Bob), (Alice, friend, Carol) and (Bob, friend, Dave) are individual binary relations.

In this case, the link prediction task may predict a friendship between Carol and Dave. However, binary relations are inadequate for representing complex real-world facts that involve multi-entity relationships.

Existing multi-relational modeling methodologies predominantly extend binary relation modeling techniques to manage multi-relational data. For example, the m-TransH (Wen et al., 2016) builds upon the classic TransH (Wang et al., 2014) by embedding entities and relations onto a hyperplane for prediction. However, m-TransH falls short in sufficiently capturing the interconnections between different facts in multi-relational scenarios. The RAE (Zhang et al., 2018) improves upon m-TransH by incorporating fact correlation, but it does not account for the critical role of entity roles in the modeling process.

The GETD (Liu et al., 2020) extends the TuckER (Balažević et al., 2019), designed to encapsulate more comprehensive relational information by reconfiguring and streamlining the fundamental tensor. However, the high computational complexity associated with GETD can impede efficiency during data processing and training phases. Many existing methodologies tend to neglect the nuanced entity roles and intricate associations present in multi-relational contexts. This oversight restricts their capacity to fully grasp the profound semantics inherent in such datasets, thereby compromising prediction accuracy and overall model performance.

In order to confront these challenges, we introduce the Role Semantics-Aware Multi-Relational Link Prediction Model (RSP). This model effectively integrates implicit role information across diverse arities. By incorporating a projection matrix, RSP significantly improves the performance of multi-relational link prediction. The primary contributions of this paper are as follows:

- Starting from the attributes of entity roles in multi-relational contexts, we propose an entity role-centered multi-relational modeling method, RSP. This model represents the implicit information and deep semantics of multi-relational data through key-value pairs of entity roles, enhancing the accuracy of multi-relational modeling.
- To overcome the constraints of viewing relationships holistically and overlooking implicit entity roles across diverse relations, we meticulously model role attributes and their interplay across a spectrum of arities. Employing multiple sub-relation sets to depict comprehensive relationships augments the model's comprehension of multi-relational intricacies. Furthermore, the RSP scoring module incorporates a projection matrix, adeptly capturing both role and entity embeddings, thereby markedly elevating prediction precision.

2 Related work

Models involved in link prediction tasks are mainly divided into four categories: translation-based models, tensor decomposition-based models, neural network-based models, and key-value pair-based models.

2.1 Translational distance model

Translation-based models serve as a cornerstone for link prediction within knowledge graphs. These models embed entities and relations into a unified vector space, facilitating the detection of associations between entity and relation representations. The m-TransH model extends TransH by intertwining entity and relation embeddings, projecting them onto a hyperplane to manage multi-relational data. However, this approach is marred by information loss. To address the shortcomings of m-TransH, RAE incorporates a multi-layer perceptron (MLP) to bolster its precision in multi-relational link prediction. Nonetheless, RAE grapples with challenges stemming from inequality among entities in multi-relational scenarios.

2.2 Tensor decomposition model

Tensor decomposition models serve to simplify high-order tensors into lower-order ones, thereby enhancing the model's predictive capability. In the realm of multi-relational link prediction, the first method based on tensor decomposition was GETD. However, this approach necessitates segmented training for varying arities, thereby increasing complexity. To address this issue, S2S (Di et al., 2021) has been developed to enhance efficiency by partitioning entity and relation embeddings into segments that are shared across all arities.

Tensor decomposition techniques effectively model multi-relational data but require high-dimensional vector decomposition, leading to significant computational complexity. Additionally,

their expressiveness diminishes in sparse knowledge graphs, limiting the capture of complex high-order interactions.

2.3 Neural network models

Deep learning neural networks have been substantiated as an efficacious methodology for addressing the link prediction issue within knowledge graphs. These neural network techniques predominantly bifurcate into two categories: convolutional neural networks (CNNs) and graph neural networks (GNNs).

- **Convolutional Neural Networks**

ConvE (Dettmers et al., 2018) pioneered the application of multi-layer convolutional neural networks to link prediction tasks by conducting convolutions on both the head entity vector and the relation vector within a triple. However, this model falls short in fully capturing the interaction among all components of the triple. To address this limitation, ConvKB (Nguyen et al., 2017) enhances ConvE by incorporating the tail entity vector into the convolution process, thereby capturing the relational features of the entire triple more effectively. ConvR (Jiang et al., 2019) further refines this approach by using the relation vector directly as the convolution kernel and performing convolutions on the head entity vector. This modification significantly strengthens the model's ability to capture features of the entire triple. InteractE (Vashishth et al., 2020) optimizes the vector stacking process and increases the interaction between entity and relation vectors, thereby further enhancing the model's performance in link prediction tasks.

- **Graph Neural Networks**

Graph Neural Networks (GNNs) are specialized neural network models engineered to manage graph-structured data. The original GNN architecture has been enhanced for link prediction tasks through the application of relation embedding techniques. R-GCN (Schlichtkrull et al., 2018) pioneered the use of graph neural networks in knowledge graphs, employing a transformation matrix for each relation type to encapsulate relational information. However, as the number of relation types increases, so does the number of matrices, resulting in a parameter explosion. To counter this issue, CompGCN (Vashishth et al., 2019) amalgamates relation embeddings with node vectors, thereby reducing parameters and alleviating the scalability problem inherent in R-GCN. WGCN (Shang et al., 2019) further enhances expressiveness by conceptualizing the knowledge graph as multiple single-relation subgraphs.

2.4 Key-value pair model

The key-value pair model facilitates the transformation of multi-relational data into a series of key-value pairs, thereby supplying comprehensive contextual information for each relational fact. Through an in-depth analysis of these pairs, the model is capable of accurately ascertaining the validity of relational facts, thus presenting a more efficient method for multi-relational link prediction.

NaLP (Guan et al., 2021) represents entity roles in multi-relational data as key-value pairs to capture the complex relational structures among entities. HypE (Fatemi et al., 2019) represents entity roles in multi-relational data as key-value pairs to capture the complex relational structures among entities. HINGE (Rosso et al., 2020) represents multi-relational facts as a primary triple and a set of auxiliary key-value pairs, and uses the feature representations of these tuples to obtain final scores. NeuInfer (Guan et al., 2020) also considers the primary triple and the corresponding key-value pairs that describe it, and performs a weighted sum of the scores of the primary triple and the compatibility scores of the key-value pairs to generate the final score. StarE (Galkin et al., 2020) combines the first two entities and the relation to form a primary triple, while the remaining entities are represented as key-value pairs, and uses a Transformer (Vaswani et al., 2017) to evaluate the multi-relational facts.

Based on the key-value pair representation, we propose the multi-relational link prediction model RSP. RSP defines entities under different relational conditions as roles and analyzes the implicit information among different entity roles within the multi-relational data. By introducing projection matrices, RSP models and analyzes different entity roles, enabling it to handle varying relational arities and complex scenarios involving multiple relation types.

3 Method

The RSP model is composed of three key components: the Role Information Construction Module, the Relation Type Embedding Module, and the Model Scoring Module, with the model

architecture depicted in Figure 1. The core concept of RSP can be summarized in the following steps: First, for each relation, a role matrix M^r is constructed for the corresponding roles, and a global role embedding matrix U is introduced to capture the role information comprehensively. Second, a feature vector E_e is generated for each entity, and a feature vector S_r is generated for each relation. The vector S_r is used to capture and represent the overall relation feature R_r. Finally, the role embedding U, the projection matrix P_i^r, and the relation feature R_r are combined through an inner product operation, and a weighted fusion mechanism is employed to obtain a comprehensive representation of the relation, which is then used to compute the prediction score for multi-relational links. By following this process, the RSP model is able to effectively analyze and integrate role information, entity features, and relation characteristics, thereby improving the accuracy of multi-relational link prediction. The model design not only accounts for the interactions between roles but also enhances the richness of role representation through the global embedding matrix, thus boosting the model's ability to handle multi-relational link prediction tasks in knowledge graphs.

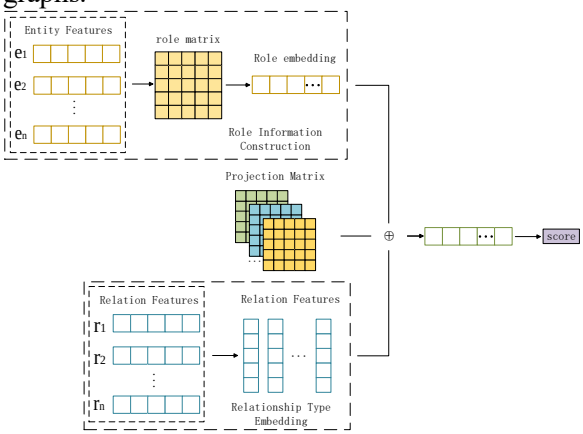


Figure 1: Architecture of the RSP Model.

3.1 Role Information Construction Module

In multi-relational representations, entities play different roles in various relationships. For each entity e, an entity feature space E is introduced, where each entity e has a corresponding feature vector E_e. For different relationships r, a role matrix $M^r \in \mathbb{R}^{|E|\times a_r\times d}$ is introduced to capture and distinguish the role information of entities in different relationships, where $|E|$ represents the number of entities, a_r denotes the number of roles in relation r, and d is the dimension of the embeddings. In the role matrix M^r, $M^r_{e,i,:}$ represents the embedding vector of entity e in the i-th role for relation r, where $1 \le i \le a_r, 1 \le e \le |E|$,:and : denotes all elements in that row, with d elements in total.

Additionally, a global role embedding matrix $U \in \mathbb{R}^{K\times d}$ is introduced to store global information related to roles, where K represents the total number of roles. For each relation r, the role embeddings for role i are computed by combining the role matrix M^r and the global embedding matrix U.

$$M^r_{e,i,:} = u_{r,i} = \sum_{k=1}^{K} \Phi(\alpha_i^r[k]) \cdot u_k, \quad i = 1,2,\dots,a_r \quad (1)$$

Here, $u_{r,i}$ represents the embedding vector of role r, and u_k is the k-th row in the global role embedding matrix, representing the global role embedding vector. K is the total number of global role embeddings. Φ is the activation function used to adjust the weights, and $\alpha_i^r[k]$ is the weight of the global role embedding u_k for role i in relation r.

3.2 Relation Type Embedding

In multi-relational data, many models treat a relationship as a whole, which overlooks the implicit information about the different roles played by entities within the relationship, potentially affecting the final prediction results. To address this issue, we treat a relation r as a collection of multiple sub-relations r_k, with each sub-relation containing specific implicit information. By extracting features from these sub-relations and effectively integrating them, a comprehensive relation feature vector can be generated. By exploring the connections between each sub-relation and the entities, as well as the connections between sub-relation features and entity features, the model's understanding of multi-relational data is enhanced. Specifically, for each sub-relation r_k, its feature extraction is represented as:

$$S_{r_k},\ D_{r_k} = \Psi(M_{r_k}, U_{r_k}) \quad (2)$$

The feature extraction function $\Psi()$ is responsible for extracting key features from the sub-relation r_k, forming a set of feature vectors S_{r_k}. Each vector in this set describes a specific aspect of the sub-relation r_k in detail. At the same time, the description D_{r_k} of the sub-relation r_k includes the relation type, connection patterns between entities,

and the distribution of roles. For the overall relation feature R_r , it can be specifically represented as:

$$R_r = \sum_{k=1}^{n} a_k \cdot S_{rk} \tag{3}$$

Here, R_r represents the overall feature vector of relation r. a_k is a weighting coefficient used to perform a weighted summation of the sub-relation features. A projection matrix P_i^r is introduced, where each element $P_i^r[j,k]$ represents the weight corresponding to the k-th dimension of the embedding vector for role i, used to map the embedding of role i to the j-th dimension of the entity feature space E_i.

3.3 Model Scoring Module

The entity role embedding vector $M_{e,i,:}^r$, the projection matrix P_i^r, and the relation features R_r are multiplied, and through weighted fusion, the comprehensive representation of the relation X_r is obtained, in the following form:

$$X_r = \sum_{i=1}^{a_r} \langle M_{e,i,:}^r, P_i^r \cdot R_r \rangle \tag{4}$$

Afterward, the relation's prediction score is modeled through a multi-layer perceptron (MLP), with the scoring function as follows:

$$\phi(r) = \sigma(\mathbf{W}_2 \cdot \mathrm{ReLU}(\mathbf{W}_1 \cdot X_r) + \mathbf{b}_r) \tag{5}$$

Where $\mathbf{W}_1$ and $\mathbf{W}_2$ are weight matrices, $\mathbf{b}_r \in \mathbb{R}^{1\times d}$ is the embedding representation of the corresponding bias vector, and σ is the activation function.

3.4 Random Model Training

In link prediction tasks, a scoring function is typically used to assign scores to each entity pair and relation to determine whether a link exists between them. During the training process, for each positive sample $x \in F$, a corresponding number of negative samples are generated to balance the ratio of positive to negative samples. For each positive sample, we identify its corresponding relation r and the number of roles a_r associated with that relation. Then, for each positive sample, a_r negative samples are generated, ensuring that these negative samples share the same relation r as the positive sample. This negative sampling strategy helps the model better learn the relationships between entities, improving the accuracy and performance of link prediction. The negative sampling process can be expressed as:

$$\bigcup_{i=1}^{a_r} N_x^{(i)} \equiv \bigcup_{i=1}^{a_r} \{M_1^r : e_1, \cdots, M_i^r : \bar{e}_i, \cdots, M_{a_r}^r : e_{a_r} \notin F \,|\, \bar{e}_i \in E, \bar{e}_i \neq e_i\} \tag{6}$$

Here, $N_x^{(i)}$ represents the negative sample set generated for the i-th role of the positive sample x. $\bar{e}_i$ denotes the i-th entity in the negative sample, which is different from the entity in the positive sample. F is the set of facts in the knowledge base. Through the negative sampling strategy, the model is exposed to more diverse data, improving its generalization ability and enhancing its robustness.

4 Experiments

In this section, we conduct experiments on two publicly accessible multi-relational datasets, namely JF17K and FB-AUTO, to assess the efficacy of the RSP model. We commence by introducing these datasets and the evaluation metrics employed in our experiments. Subsequently, we provide a comprehensive overview of the baseline models utilized for comparison. Finally, we present the experimental results to illustrate the performance of the RSP model in multi-relational link prediction tasks. Table 1 presents the statistical data of the two datasets. Tabel 1 presents the statistical data of the two datasets.

4.1 Datasets

- JF17K (Wen et al., 2016): This dataset is derived from Freebase through specific filtering conditions. JF17K removes entity facts with low occurrence in Freebase, while also excluding facts involving strings, numbers, and enumerated types.
- FB-AUTO (Schlichtkrull et al., 2018): This dataset is also based on Freebase and removes data that involve only single facts from Freebase. In terms of link order, it connects multiple related facts involving relevant relations and entities to form multi-relational groups.

Dataset	Entities	Relations	Train	Valid	Test	Arities
JF17K	28645	322	76379	15822	24568	2-6
FB-AUTO	3388	8	6778	2255	2180	2-5

Tabel 1: Statistical information of the datasets, including the number of entities and relations in the JF17K and FB-AUTO datasets. The datasets are split into training, validation, and test sets, and the arity of the multi-relational data is also provided.

4.2 Evaluation indicators

In multi-relational link prediction tasks, model evaluation typically uses two metrics: Mean Reciprocal Rank (MRR) and $Hits@k$.

- Mean Reciprocal Rank (MRR) is the arithmetic mean of the reciprocal ranks for all triples in the test set, defined as follows:

$$MRR = \frac{1}{|T_{\text{test}}|} \sum_{(h,r,t)\in T_{\text{test}}} \frac{1}{2}\left(\frac{1}{rank(h,r,t)}\right) \quad (7)$$

MRR has the property of smoothness, with its value range being $MRR \in (0,1]$. A higher value indicates better link prediction performance of the model.

- $Hits@k$ represents the proportion of triples in the test set whose rank is within the top k. It is defined as follows:

$$Hits@k = \frac{1}{|T_{\text{test}}|} \sum_{(h,r,t)\in T_{\text{test}}} \frac{1}{2}(\{rank_{r,t|h,r}(h) \leqslant k\} \quad (8)$$

A higher value of $Hits@k$ indicates better link prediction performance of the model. We use $Hits@1$, $Hits@2$, and $Hits@3$ as the performance evaluation metrics for the model.

4.3 Baseline model

Our experiment consists of two main parts. First, we evaluate the effectiveness of key-value pair representation in encoding multi-relational data, focusing on its impact on link prediction accuracy and efficiency. Second, to ensure fairness, we compare our model with baseline models that also utilize key-value pair representation.

- RAE (Zhang et al., 2018): This model introduces a multi-layer perceptron (MLP) based on m-TransH, improving the correlation assumption in the m-TransH model and enhancing its capability to handle link prediction tasks in knowledge hypergraphs.
- NaLP (Guan et al., 2021): As the first to extend neural networks to multi-relational link prediction, NaLP decomposes multi-relational facts into role-key-value pairs and applies convolution to extract features. An FCN network is then used to model these features for link prediction.
- HINGE (Rosso et al., 2020): This model represents multi-relational facts as a primary triple and a set of auxiliary key-value pairs. It learns relevant feature representations through convolution layers and computes the score of the facts.
- NeuInfer (Guan et al., 2020): This model represents multi-relational facts using a primary triple and auxiliary key-value pairs. The final fact score is computed as a weighted sum of the primary triple's trustworthiness score and the compatibility scores of the key-value pairs.
- HypE (Fatemi et al., 2019): This model designs different convolution kernels for multi-relational knowledge with varying characteristics, using unique filters to embed entities in specific positions, thus improving the expressive power of the model.

4.4 Validity analysis

The experimental results are shown in Tabel 2, which also includes the results of the baseline models for reference.

model	JF17K				FB-AUTO			
	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10	MRR
RAE	0.312	0.433	0.561	0.396	0.614	0.764	0.854	0.703
NaLP	0.239	0.334	0.450	0.310	0.611	0.712	0.774	0.672
HINGE	0.397	0.490	0.618	0.473	0.630	0.706	0.765	0.678
NeuInfer	0.373	0.484	0.604	0.451	0.700	0.755	0.805	0.737
HypE	0.421	0.550	0.669	0.507	0.774	0.824	0.856	0.804
RSP	0.465	0.579	0.696	0.543	0.782	0.845	0.879	0.819

Tabel 2: Comparison results between RSP and baseline models. $Hits@k$ indicates the ratio of triples ranked within the top k in the test set, while MRR represents the arithmetic mean of the reciprocal ranks of all triples in the test set. Higher values of $Hits@k$ and MRR indicate better model performance, with the best results highlighted in bold.

1

From the experimental results shown in Table 2, it can be observed that among the key-value-based models, all models except for NaLP outperform RAE on most evaluation metrics. This indicates that the key-value representation has advantages over traditional representation methods, suggesting that a reasonable representation form can significantly enhance the performance of multi-relational link prediction modeling. The RSP model achieves the best performance across all evaluation metrics on both multi-relational datasets. Specifically, on the JF17K dataset, compared to other key-value baseline models, RSP improves by 10.5%, 5.2%, 4%, and 7.1% on Hits@1, Hits@3, Hits@10, and MRR, respectively. On the FB-AUTO dataset, the RSP model achieves improvements of 1%, 2.6%, 2.7%, and 1.9% over the best baseline model results on Hits@1, Hits@3, Hits@10, and MRR, respectively. Thus, it is evident that the RSP model outperforms others in the set of key-value representation models.

4.5 Analysis of arities

In the realm of multi-relational data, a model's capacity to manage varying arity is a crucial determinant of its efficacy. Consequently, we conducted an in-depth analysis of the performance disparities of the RSP model across different arities within the FB-AUTO and JF17K datasets. The FB-AUTO dataset encompasses binary, ternary, and quinary relations, with Figure 2 illustrating the evaluation metrics of the RSP model for these three relation types. The JF17K dataset, on the other hand, includes five distinct arity types, with Figure 3 presenting the evaluation metrics of the RSP model for each arity.

We can see that the RSP model has a better performance as the arity of relations increases, which shows its effectiveness in dealing with multi-relational data. Besides, we find that the two factors that have the most influence on the performance of the RSP model when dealing with different arity data are the size of arity and the size of data sample. For example, in FB-AUTO, the RSP model performs poorly on quaternary relations due to the sparsity of samples, while for quinary relations with the largest data sample size, the RSP model achieves good performance.

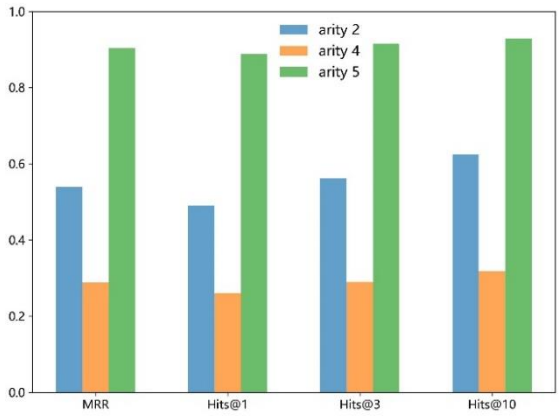


Figure 2: Comparison of various metrics across different arities in the FB-AUTO dataset.

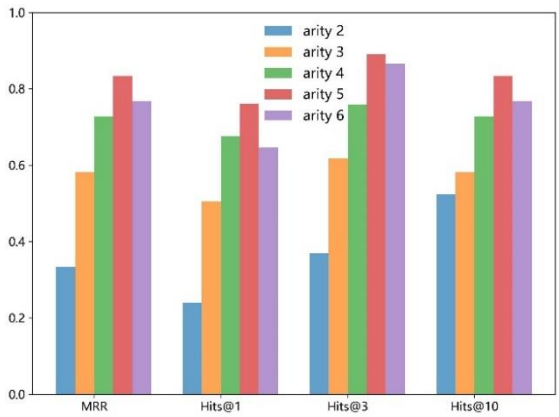


Figure 3: Comparison of various metrics across different arities in the JF17K dataset.

5 Conclusions

In addressing the oversight of entity roles' influence on prediction outcomes in multi-relational link prediction within knowledge graphs, we introduce the Role Semantic Perception (RSP) model. This model is designed to be centered around entity roles and has been rigorously tested through a series of experiments. The RSP model demonstrated superior performance on two publicly available multi-relational datasets across all key metrics, surpassing existing key-value pair-based models. This indicates its exceptional ability to manage complex multi-relational data. Notably, the RSP model exhibits particular proficiency in handling multi-relational data with a higher density.The task of multi-relational link prediction within knowledge graphs is currently a focal point and presents a significant challenge in the field. Future research endeavors should prioritize enhancing the accuracy of multi-relational data

modeling by emphasizing the interactions between entities, their comprehensive representation, and the equality among various relationships. These factors are frequently overlooked in existing multi-relational modeling methodologies.

Limitation

We identify the following limitations of our work.

- The RSP model has a few shortcomings in few-shot tasks. When the number of samples for an arity relation is small, it is difficult to learn and represent the relations well.
- Since our work is based on multi-relational modeling, the prediction performance on binary relational data is relatively poor. This may be due to the representation method we choose, as RSP performs poorly in analyzing binary data.
- Our method is evaluated on two public multi-relational datasets, which may not be able to cover all possible scenarios. Therefore, the performance of our method may vary on other datasets or real-world applications.

Acknowledgments

This work is supported by National Natural Science Foundation of China (62162056); Science Foundation of Ministry of Education of C hina (24YJA880007).

References

Ji Shaoxiong, et al. "A survey on knowledge graphs: Representation, acquisition, and applications." IEEE transactions on neural networks and learning systems 33.2 (2021): 494-514.

Wang Meihong, Linling Qiu, and Xiaoli Wang. "A survey on knowledge graph embeddings for link prediction." Symmetry 13.3 (2021): 485.

Rosso, Paolo, Dingqi Yang, and Philippe Cudré-Mauroux. "Beyond triplets: hyper-relational knowledge graph embedding for link prediction." Proceedings of the web conference 2020. 2020.

Wen, Jianfeng, et al. "On the representation and embedding of knowledge bases beyond binary relations." arXiv preprint arXiv:1604.08642 (2016).

Wang, Zhen, et al. "Knowledge graph embedding by translating on hyperplanes." Proceedings of the AAAI conference on artificial intelligence. Vol. 28. No. 1. 2014.

Zhang, Richong, et al. "Scalable instance reconstruction in knowledge bases via relatedness affiliated embedding." Proceedings of the 2018 world wide web conference. 2018.

Liu, Yu, Quanming Yao, and Yong Li. "Generalizing tensor decomposition for n-ary relational knowledge bases." Proceedings of the web conference 2020. 2020.

Balažević, Ivana, Carl Allen, and Timothy M. Hospedales. "Tucker: Tensor factorization for knowledge graph completion." arXiv preprint arXiv:1901.09590 (2019).

Di, Shimin, Quanming Yao, and Lei Chen. "Searching to sparsify tensor decomposition for n-ary relational data." Proceedings of the Web Conference 2021. 2021.

Dettmers, Tim, et al. "Convolutional 2d knowledge graph embeddings." Proceedings of the AAAI conference on artificial intelligence. Vol. 32. No. 1. 2018.

Nguyen, Dai Quoc, et al. "A novel embedding model for knowledge base completion based on convolutional neural network." arXiv preprint arXiv:1712.02121 (2017).

Jiang, Xiaotian, Quan Wang, and Bin Wang. "Adaptive convolution for multi-relational learning." Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). 2019.

Vashishth, Shikhar, et al. "Interacte: Improving convolution-based knowledge graph embeddings by increasing feature interactions." Proceedings of the AAAI conference on artificial intelligence. Vol. 34. No. 03. 2020.

Schlichtkrull, Michael, et al. "Modeling relational data with graph convolutional networks." The semantic web: 15th international conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, proceedings 15. Springer International Publishing, 2018.

Vashishth, Shikhar, et al. "Composition-based multi-relational graph convolutional networks." arXiv preprint arXiv:1911.03082 (2019).

Shang, Chao, et al. "End-to-end structure-aware convolutional networks for knowledge base completion." Proceedings of the AAAI conference on artificial intelligence. Vol. 33. No. 01. 2019.

Guan, Saiping, et al. "Link prediction on n-ary relational data based on relatedness evaluation." IEEE Transactions on Knowledge and Data Engineering 35.1 (2021): 672-685.

Fatemi, Bahare, et al. "Knowledge hypergraphs: Prediction beyond binary relations." arXiv preprint arXiv:1906.00137 (2019).

Rosso, Paolo, Dingqi Yang, and Philippe Cudré-Mauroux. "Beyond triplets: hyper-relational knowledge graph embedding for link prediction." Proceedings of the web conference 2020. 2020.

Guan, Saiping, et al. "Neuinfer: Knowledge inference on n-ary facts." Proceedings of the 58th annual meeting of the association for computational linguistics. 2020.

Galkin, Mikhail, et al. "Message passing for hyper-relational knowledge graphs." arXiv preprint arXiv:2009.10847 (2020).

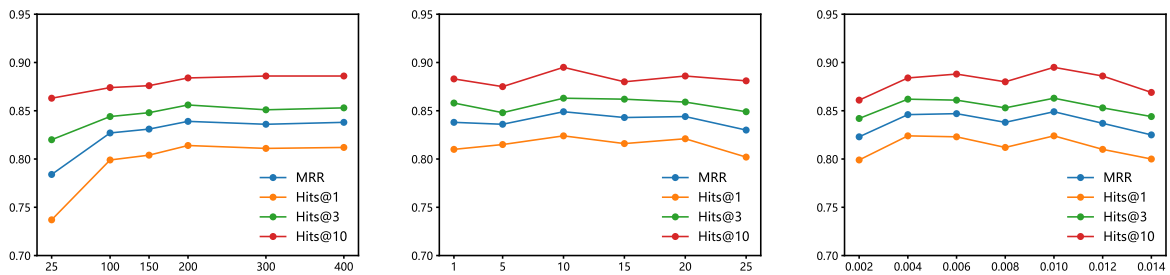
Vaswani, A. "Attention is all you need." Advances in Neural Information Processing Systems (2017).

A parameterization

This paper implements the RSP model using the PyTorch framework, with the following default parameter settings for the experiments: embedding dimension $d = 250$, projection matrix size $k = 10$, learning rate $lr = 0.01$, batch size of 64, number of training epochs set to 200, and decay rate of 0.99.

B Parameter sensitivity analysis

To gain deeper insights into the effectiveness of RSP, we conducted a series of parameter studies on the RSP model using the FB-AUTO dataset, focusing on the embedding dimension d, projection matrix size k, and learning rate lr. We selected the hyperparameters for the model as follows: embedding dimension $d \in \{25,50,100,150,200,250,300,400\}$, projection matrix size $k \in \{1,5,10,15,20\}$, and learning rate lr$\in \{0.002,0.004,0.006,0.008,0.01,0.12\}$.



(a) (b) (c)

Figure 4: Parameter sensitivity analysis on RSP model using FB-AUTOdataset. (a) The impact of embedding dimensio-ns on metrics. (b)The impact of projection matrices on metrics. (c) The impact of learning rate on metrics

On the embedding dimension. As shown in Figure 4(a):, we present the trend of various metrics of the RSP model with different sizes of embedding dimension d. The figure indicates that when $d < 200$, the metrics of the model exhibit an increasing trend. However, when $d > 200$, the metrics gradually stabilize as the embedding dimension increases, indicating that the influence of the embedding dimension on the model becomes negligible at this point.

On the projection matrix size. As shown in Figure 4(b):, we present the trend of various metrics of the RSP model with different sizes of projection matrix k. The figure indicates that when $k < 10$, the model metrics exhibit an increasing trend. However, when $k > 10$, the metrics show a declining trend, with the model achieving better results at $k = 10$.

On the learning rate. As shown in Figure 4(c): we present the trend of various metrics of the RSP model with different learning rates lr. The figure indicates that when $lr < 0.01$, the model metrics exhibit an increasing trend. Conversely, when $lr > 0.01$, the metrics show a declining trend, with the model achieving better results at $lr = 0.01$.

C Arity analysis

To gain deeper insights into the predictive capability of the RSP model across different arities, we analyze the JF17K dataset, which encompasses all relationships ranging from binary to 6-ary. This analysis allows us to

evaluate the RSP model's performance in predicting multi-relational data.

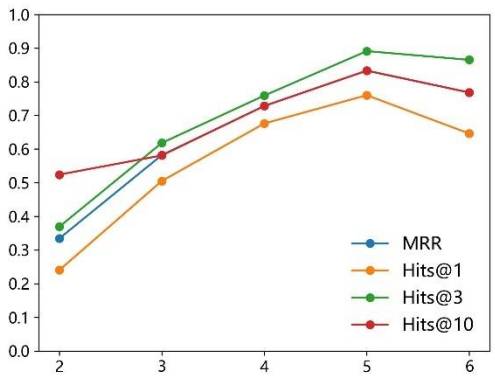


Figure 5: To gain a deeper understanding of RSP's modeling capability for multi-relational data, we conducted a series of studies using the JF17K dataset, which contains various types of relationships ranging from binary to 6-ary.

In multi-relational scenarios, the ability of a model to handle different arity is an important indicator for its performance. Figure 5 shows the prediction results of RSP on five kinds of relations with different arity from binary to 6-ary. We can observe that the link prediction performance of RSP improves significantly as the relation arity increases. This indicates that RSP can effectively utilize more complex relation information and capture deep semantic connections between entities, thus improving the prediction accuracy. The increasing predictive capacity further proves that RSP has the potential to model multi-relational datasets, which can adapt to different levels of relationship complexity and make more accurate predictions.