Restricted Neighborhood Knowledge Graph Embedding

Anonymous ACL submission

Abstract

Knowledge Graph Embedding (KGE) methods not only focus on the semantic information provided by triples, but also consider the structural information inherent in the graph structure, when modeling entities and relations within knowledge graphs. However, within the framework of the information transmission mechanisms, methods relying on adjacency tables cannot effectively balance the scale differences between neighborhoods, which directly leads to information disparities in modeling structural information. To address this imbalance, we present a novel KGE model called restricted neighborhood knowledge graph embedding (RNE). Firstly, it uses restricted neighborhood to aggregate neighborhood information. The restricted neighborhood formats the neighborhood space, normalizing the neighborhood of each entity to the same size. Secondly, to enrich the information within restricted neighborhood, we propose a restricted neighborhood position encoding method. Experimental results on two standard benchmarks demonstrate the significant potential of our proposals.

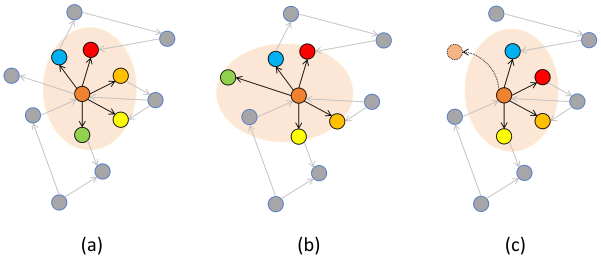

Figure 1: The composition of the restricted neighborhood. (a) The neighbor entities within the restricted neighborhood must meet the following constraints: they must be located within the first-order neighborhood of the anchor entity, and be pointed to by the anchor entity. (b) The process of selecting neighbor entities within the restricted neighborhood is in order. (c) If the number of the anchor entity's neighbors in the adjacent neighborhood does not meet the number of entities in the restricted neighborhood, an empty relation will be used to point from the anchor entity to a virtual entity composed of information from the anchor entity.

1 Introduction

Knowledge graphs are a type of graph-based data that contain two main elements: entities and relations (Wang et al., 2017). Knowledge graphs (e.g., Freebase (Bollacker et al., 2008) and WordNet (Miller, 1995)) are widely used in many downstream tasks, such as semantic search (Xiong et al., 2017), question answering (Kaiser et al., 2021), and recommendation systems (Wang et al., 2021a). Among many methods, KGE methods have been widely and deeply studied (Ge et al., 2023; Nayyeri et al., 2023).

The KGE method aims to use entities and relations embedded in a low-dimensional space as numerical representations to form expressions that conform to the semantics of triples, to evaluate the rationality of triples or infer missing information in knowledge graphs (Wang et al., 2017; Li et al., 2022a). Models such as TransE (Bordes et al., 2013), DistMult (Yang et al., 2015), and others impose constraints on the computational composition of triplet embeddings (Li et al., 2023) and calculate the rationality of the composition by defining a scoring function. These models focus on the semantic information provided by triples (Li et al., 2022a), but they do not consider the structural information provided by the topology of the graph. GNN-based models not only focus on the semantic information provided by triples, but also on the structural information of the knowledge graph structure (Cao et al., 2022). Models such as CompGCN (Vashishth et al., 2020) use message passing mechanisms to obtain new representations of entities using neighbor entities. Models such as MA-GNN (Xu et al., 2023) consider problems from both global and local levels when modeling structural information. These models leverage the adjacency relationship of graphs

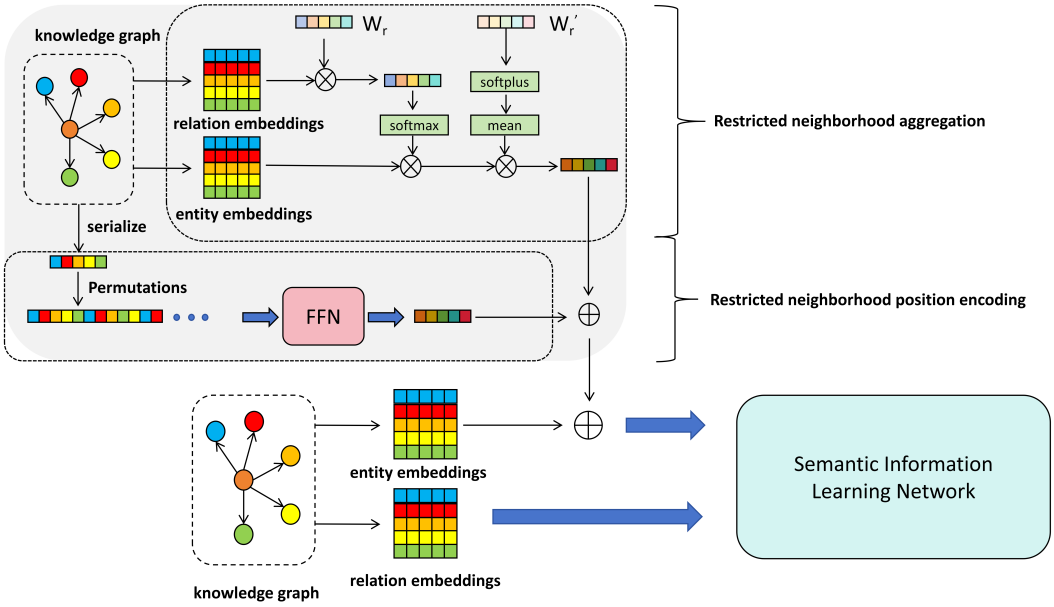

Figure 2: The architecture of RNE. RNE consists of two modules: structural information learning module and semantic information learning module. Among them, the structural information learning module includes two sub-modules: 1) restricted neighborhood information transfer, and 2) restricted neighborhood position encoding.

to model structural information onto head and tail entities.

However, the knowledge graph structure is inherently incomplete, and the size of the neighborhood of each entity is different. Therefore, using the original adjacency relationship to aggregate information in the neighborhood is unfair. The adjacency table usually considers the global structure, making it difficult to balance local differences. We start from modeling local structural information, placing the global perspective on the adjacent neighborhood and the local perspective within the adjacent neighborhood, further limiting the scope of local hierarchy coverage. We define a concept called restricted neighborhood. In the local perspective, we limit the structure of the restricted neighborhood at both the entity and relation levels to ensure that each entity is given the same amount of structural information; in the global perspective, we consider the importance differences of entities within the restricted neighborhood in the adjacent neighborhood, and adopt a restricted neighborhood position encoding to encode the semantic order of these entities. Our contributions can be summarized as follows:

•We propose a concept called restricted neighborhood, which uses a fixed number of neighbor entities to express structural information.

•We propose a new KGE model using restricted neighborhood. Theoretically, this model takes into account both the semantic information and the structural information.

•To supplement the expression of structural information in restricted neighborhood, we propose a encoding method based on restricted neighborhood.

•Our method has been fully tested on two standard benchmarks, proving the competitiveness of the model. This suggests that restricted neighborhood and its derived concepts have certain potential.

2 Related work

Although some works are designed based on broader tasks, they still affect knowledge graph representation learning and are applicable for comparison of KGE models. Based on the type of information used, KGE models can be divided into the following three categories (Rossi et al., 2021; Li et al., 2022a; Chen et al., 2023a; Zhang et al., 2023a): Non-GNN-based models, GNN-based models, and PLM-based models.

Non-GNN-based models focus on the semantic information provided by triples. For example, models such as TransE (Bordes et al., 2013), TransH (Wang et al., 2014), RotatE (Sun et al., 2019),

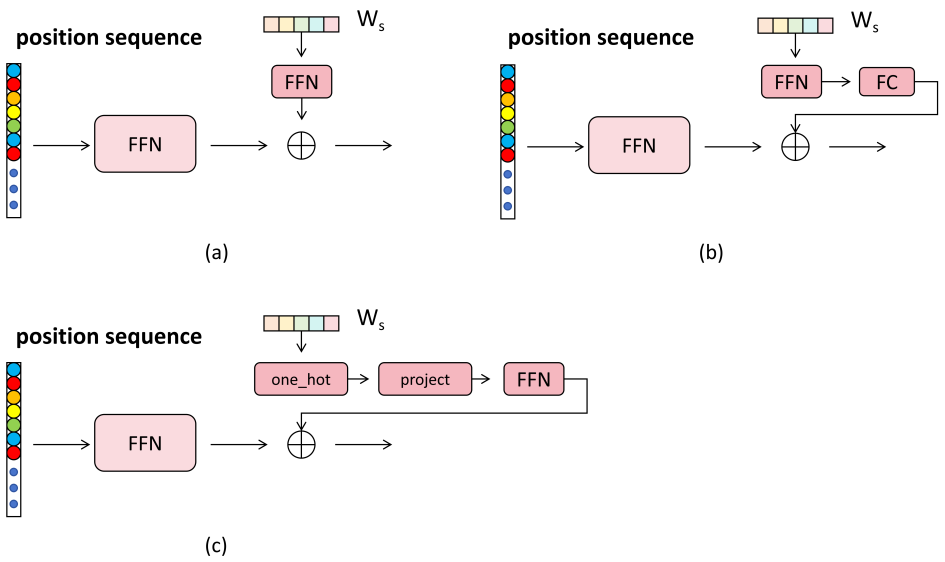

Figure 3: The way to encode information in a restricted neighborhood. (a) Linear transformation. (b) Deeper linear transformation. (c) Linear transformation mapped to the one-hot vector space.

RotH (Chami et al., 2020), and HousE (Li et al., 2022b) use the distance between the head entity, relation, and tail entity in the embedding space to measure the plausibility of the triple based on translation distance. Models such as RESCAL (Nickel et al., 2011), DistMult (Yang et al., 2015), ComplEx (Trouillon et al., 2016), and DualE (Cao et al., 2021) use the matching semantics between entities and relations in the embedding space to measure the plausibility of the triple. These models seek to fully utilize the semantic information provided by triples to model without using structural information as a supplement.

GNN-based models focus not only on the semantic information provided by triples, but also on the structural information of entities' positions in the graph. For example, CompGCN (Vashishth et al., 2020), MRGAT (Li et al., 2022c) and other models use message passing mechanisms to transfer adjacent entity information to head and tail entities. NBFNet (Zhu et al., 2021), RED-GNN (Zhang and Yao, 2022), A*Net (Zhu et al., 2023) and other models sample entities through paths to mine larger-scale structural information, and learn the influence of other entity information on the source and target entities in a inductive manner. These models rely on adjacency tables to express the data representation of graph structures and achieve modeling of structural information. However, adjacency tables are difficult to reflect the importance differences between relations. HRAN (Li et al., 2022d) designs a relation-aware attention mechanism based on the characteristics of relations, which considers the difference in the importance of relations when aggregating neighbor entity information. In addition, GraphSAGE (Hamilton et al., 2017) proposes a method for neighbor node sampling to fix the computational complexity of batch training. MA-GNN (Xu et al., 2023) proposes an attention mechanism for modeling structural information by building two-hop neighborhoods on the basis of entity sampling.

PLM-based models focus on the extended information related to the knowledge graph, in addition to semantic information and structural information. For example, models such as KG-BERT (Yao et al., 2019), SimKGC (Wang et al., 2022), and CSProm-KG (Chen et al., 2023b) convert the problem of triples into a sequence problem, using contextual information on the sequence to calculate the rationality score of the triple.

Whether it is based on graph structure to model structural information or based on an extension method to construct sequences that can produce contextual influence, these methods do not take into account the balance of information quantity. Many methods introduce larger-scale structural information modeling tools, such as paths(Zhang et al., 2023b), to blur the difference in their local information quantity by supplementing more information. This is one solution. However, we directly start from the balance of local information and fix the

information quantity of each entity by restricted neighborhood to balance the difference in neighborhood size.

3 Methodology

Our model is generally divided into two main modules. The first part is the structural information learning module built using the proposed concept, restricted neighborhood, and the second part is the semantic information learning module for triples. The semantic information learning module uses the GIE (Cao et al., 2022) framework. The structural information learning module mainly consists of two parts: restricted neighborhood information transfer and restricted neighborhood position encoding.

3.1 Discussion on restricted neighborhood

The incompleteness of the knowledge graph causes differences in the size of entities' neighborhoods, which directly leads to unequal participation of neighborhood information in the representation of entities. To obtain a more equitable representation of entities in terms of information participation, we propose restricted neighborhood. It fixes the neighboring entities participating in information transfer and limits the direction of the relations. This approach eliminates the support of the original graph structure, isolates the influence of entities on longer links, and focuses the model's attention on local scope with equal size.

We set the graph structure of knowledge graph as $G = (\mathcal{V}, \mathcal{E})$. For every entity u $\in \mathcal{V}$, its restricted neighborhood is:

$$\Phi_u = \{(v, < u, v >) : v \in \mathcal{V}, < u, v > \in \mathcal{E}\} \quad (1)$$

where $|\Phi_u| = N_r$ and N_r represents the fixed number of entities in the restricted neighborhood. For the convenience of discussion, we will define the central entity of the restricted neighborhood as the anchor entity, and the tail entity of the triplet with the anchor entity as the target entity. Information about the construction of restricted neighborhood can be found in Figure 1.

3.2 Restricted Neighborhood Information Transfer

Due to the simplicity of the structure, the restricted neighborhood can be seen as a supplement to the triplet information. The model should learn the influence of entities within the restricted neighborhood on the anchor entity, and also need to consider the influence of relations within the restricted neighborhood on the former. To avoid reconstructing triplet information within the restricted neighborhood or repeatedly learning triplet information in the datasets, the model only passes information of neighboring entities, and does not pass information of relations. The information transmission of relations is retained in the semantic information learning module.

When aggregating information of neighboring entities, our model calculates the embedding of relations as weights to help information aggregation. Besides, we calculate a weight in an element-wise style by using the relation pointed to a specific neighbor entity by the anchor entity. This weight is an indicator that determines how much influence this neighbor entity can have on the anchor entity during aggregation. We represent the embedding of relation as: $e_r = \{e_r^1, e_r^2, ..., e_r^d\}$, where d represents the embedding dimension. We obtain the formula for calculating the relation-aware weight:

$$W_r = \{W_r^1, W_r^2, ..., W_r^d\} \quad (2)$$

$$e_w = \alpha_w \sum_{i=1}^{d} e_r^i W_r^i \quad (3)$$

where α_w is a preset hyper-parameter, and W_r is learnable and related to the type of relation r. This can be regarded as a way to compute the attention of neighbor entity information aggregation with strong coupling to relation information. The information of empty relations (defined in Figure 1) is only considered as a weight to minimize its impact on the model.

To further obtain the influence of relation information on neighbor information aggregation, we also use a learnable parameter to help learn the aggregated weight. The weighted sum of neighboring entities can be expressed as:

$$e_n^{'} = \sum_{i=1}^{N_r} softmax(e_w) e_n^i \quad (4)$$

where N_r represents the number of entities in the restricted neighborhood, and e_n^i represents the embedding of the i-th entity in the restricted neighborhood. In this way, we can obtain the aggregation formula for neighborhood information:

$$\alpha_{agg}^{'} = \frac{1}{N_r} \sum_{i=1}^{N_r} softplus(\alpha_{agg}^i) \quad (5)$$

Models	MRR	H@1	H@3	H@10
Situation-1	0.492	0.441	0.513	0.590
Situation-2	0.498	0.441	0.525	0.605
Situation-3	0.492	0.440	0.517	0.587

Table 1: The comparison of different combinations of structural information learning modules and semantic information learning modules on WN18RR.

Models	MRR	H@1	H@3	H@10
Condition-1	0.552	0.486	0.590	0.671
Condition-2	0.488	0.437	0.509	0.586
Condition-3	0.528	0.463	0.561	0.645

Table 2: Comparison experiments on different position encoding combinations on WN18RR.

$$e_{agg} = \alpha^{'}_{agg} e^{'}_{n} \tag{6}$$

where α^{i}_{agg} represents the weight of the i-th restricted neighborhood entity e^{i}_{n} in information aggregation, and N_r represents the number of entities in the restricted neighborhood.

3.3 Restricted Neighborhood Position Encoding

Due to the fact that all entities in restricted neighborhood are pointed to by the anchor entity, the association between neighboring entities seems to be eliminated. However, in the original graph structure, there is a certain ordering, such as topological sorting. We believe that this ordering causes a difference in the order of entities in the restricted neighborhood. And we model this difference as the semantic order of entities in the restricted neighborhood. This involves our assumption that complex structural representations caused by external real structures can be mapped to the arrangement sequence of neighboring entities in this simple space. Further, the order of entities in the restricted neighborhood should be influenced by some entity ordering on the scale of the anchor entity adjacent neighborhood. The ordering of entities in the restricted neighborhood constitutes a semantic order that expresses the arrangement and structural information of the restricted neighborhood.

Learning this semantic order will complement the expression of structural information by the restricted neighborhood.Firstly, we need to initialize a sequence of position numbers: $p_a = \{p_1, p_2, ..., p_{N_r}\}$, where p_i represents the position of the restricted neighborhood entity e_i in the dictionary, and N_r represents the number of entities in the restricted neighborhood. The full permutation transformation for the position sequence can be expressed as: $s_a = Per(p_a)$, where $Per(\cdot)$ represents the full permutation transformation. We use linear layers to map the ordering to the same space as entity and relation embeddings. Specifically, we use a feed-forward neural network (FFN) to encode sequences of entities in a restricted neighborhood. After two linear transformations, we apply a normalization operation:

$$FFN(x) = LN((xW_1{+}b_1)W_2{+}b_2)W_3{+}b_3 \tag{7}$$

where W_1, W_2, W_3 are learnable, and b_1, b_2, b_3 are biases. Here we guarantee that after learning with FFN, the dimension of position encoding will be consistent with the embedding dimension of entities.

The reason why we do not emphasize that important entity information should be taken seriously is: we believe that the weight difference in the restricted neighborhood information transfer can already reflect this difference. What we want to add here is the natural difference in graph information expression caused by the original graph structure. According to the above transformation operation, we obtain the formula for restricted neighborhood position encoding:

$$v_s = s_a W_s \tag{8}$$

$$e_{pos} = FFN(v_s) \tag{9}$$

where W_s is learnable, and $FFN(\cdot)$ represents linear transformation.

3.4 Score function and loss function

After obtaining the restricted neighborhood information transfer formula and restricted neighborhood position encoding formula, the anchor entity with structural information added can be represented as:

$$e^{'}_{a} = e_a + e_{agg} + e_{pos} \tag{10}$$

With the new representation of the anchor entity, the scoring function is formulated as:

$$f(a, r, o) = -d(SE(e_a, e_r), e_o) + b_a \tag{11}$$

where a, r, o represents anchor entity, relation, object entity respectively, and $d(\cdot, \cdot)$ represents the distance function, $SE(\cdot, \cdot)$ represents the semantic

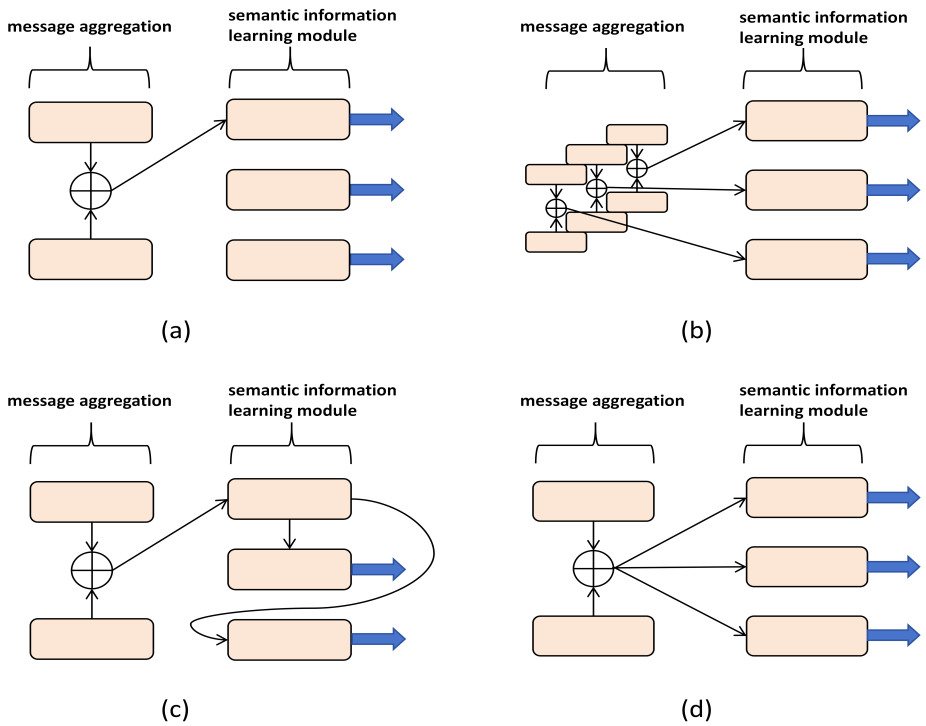

Figure 4: The combination of structural information learning module and semantic information learning module. (a) Calculate once and affect only one work route. (b) Calculate multiple times, affecting multiple work routes separately. (c) Calculate once and affect the remaining work routes through the first work route. (d) Calculate once and affect multiple work routes.

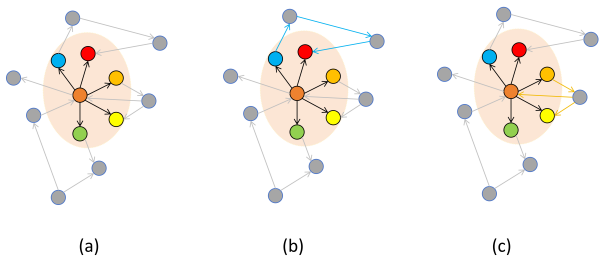

Figure 5: The basic ideas of restricted neighborhood position encoding. (a) Select entities within the adjacent neighborhood to form a restricted neighborhood. (b) In the original graph structure, entities in the restricted neighborhood may affect each other. (c) In the original graph structure, entities in the restricted neighborhood may pass their influence back to the anchor entity through a link.

learning network, and b_a represents the bias of the anchor entity (Balazevic et al., 2019). Our loss function follows that of the semantic information learning module:

$$L = \sum_{\{h,r,t\} \in \omega_P \cup \omega_N} log(1 + e^{(-Ef(h,r,t))}) \quad (12)$$

where $f(\cdot,\cdot,\cdot)$ represents the scoring function, $\omega_P \cup \omega_N$ represents the observed triplet set and the negative sampling triplet set, and E belongs to $\{1, -1\}$. When the triplet belongs to ω_P, it is 1, otherwise it is -1 (Cao et al., 2022).

4 Experiments

4.1 Datasets

Our work has been experimentally evaluated on two widely used benchmarks: WN18RR (Dettmers et al., 2018) and FB15K-237 (Toutanova and Chen, 2015). They are extracted from WordNet (Miller, 1995) and Freebase (Bollacker et al., 2008). Table 6 shows the detailed statistics of the two benchmarks. Meanwhile, we have conducted a statistical analysis on the neighborhood size of these two benchmarks for the needs of our work. The results are shown in Table 7.

4.2 Baselines

We divide the KGE models into three categories based on the types of information used. Among these three categories, some models are for path or sequence modeling, which should be distinguished from models that only use local information.

Based on these perspectives, we focus on non-path-dependent or non-sequence-dependent meth-

Models	WR18RR				FB15K-237			
	MRR	**H@1**	**H@3**	**H@10**	**MRR**	**H@1**	**H@3**	**H@10**
Non-GNN-based models								
TransE(2013)	0.223	0.014	0.401	0.529	0.330	0.231	0.369	0.528
ComplEx(2016)	0.448	0.427	0.485	0.554	0.323	0.229	0.353	0.513
RotatE(2019)	0.476	0.428	0.492	0.571	0.338	0.241	0.375	0.533
UltraE(2022)	0.501	0.450	0.515	0.592	0.351	**0.275**	0.400	0.560
HousE(2022b)	**0.511**	**0.465**	**0.528**	**0.602**	0.361	0.266	0.399	0.551
GIE(2022)	0.491	0.435	0.485	0.552	0.362	0.271	0.401	0.552
HRQE(2022)	0.505	0.454	0.524	0.601	**0.372**	**0.275**	**0.407**	**0.569**
CompoundE(2023)	0.491	0.450	0.508	0.576	0.357	0.264	0.393	0.545
Itô̂E(2023)	0.474	0.426	0.499	0.574	0.334	0.245	0.361	0.511
GNN-based models								
R-GCN(2018)	0.402	0.345	0.437	0.494	0.273	0.182	0.303	0.456
CompGCN(2020)	0.479	0.443	0.494	0.546	0.355	0.264	0.390	0.535
NBFNet(2021)	0.551	0.497	0.573	0.666	0.415	0.321	0.454	0.599
SE-GNN(2022a)	0.484	0.446	0.509	0.572	0.365	0.271	0.399	0.549
MRGAT(2022c)	0.481	0.443	0.501	0.568	0.358	0.266	0.386	0.542
MA-GNN(2023)	**0.565**	**0.507**	**0.592**	**0.679**	0.379	0.282	0.415	0.569
A*Net(2023)	0.549	0.495	0.573	0.659	0.411	0.321	0.453	0.586
KGCF(2023)	0.553	0.503	0.577	0.670	**0.421**	**0.326**	**0.461**	**0.603**
PLM-based models								
KG-BERT(2019)	0.216	0.041	0.302	0.524	-	-	-	0.528
StAR(2021b)	0.401	0.243	0.491	0.709	0.296	0.205	0.322	0.482
KG-S2S(2022)	0.574	0.531	0.595	0.661	0.336	0.257	0.373	0.498
SimKGC(2022)	0.666	0.587	0.717	0.800	0.336	0.249	0.362	0.511
CSProm-KG(2023b)	0.575	0.522	0.596	0.678	**0.358**	**0.269**	**0.393**	**0.538**
GHN(2023)	**0.678**	**0.596**	**0.719**	**0.821**	0.339	0.251	0.364	0.518
ours	0.552	0.486	0.590	0.671	0.383	0.297	0.420	0.549

Table 3: The results of link prediction on WR18RR and FB15K-237. The bold ones are the best performance, and the underlined ones are the second best performance. The results of TransE, ComplEx, RotatE are from (Xu et al., 2023). The results of R-GCN are from (Zhu et al., 2023). The results of KG-BERT are from (Wang et al., 2022). Other results are from published papers.

ods when selecting baselines. On this basis, we select some path- or sequence-dependent methods. We select three sets of baselines shown in Table 3.

4.3 Main results

Our experimental results are the average results of multiple experiments, and we did not specifically select those that were favorable to us to calculate the average.

As can be seen from Table 3, our model achieves competitive results among non-path-dependent or non-sequential-dependent methods, both compared to non-GNN-based models and GNN-based models. When we compare our model to GNN-based models, on the WN18RR, our model achieves or approaches the second best performance in all metrics except for the H@1 metric. On the FB15K-237, our model achieves the second-best performance, especially among GNN-based baselines that include path-dependent methods. Although our model is not as effective as PLM-based models on the WN18RR, it completely outperforms these PLM-based methods on the FB15K-237.

In particular, MA-GNN (Xu et al., 2023) is considered to be one of the path- or sequence-dependent methods because it uses second-order neighborhoods when modeling the neighborhood, while RNE only use first-order neighborhoods. Despite this, our model outperforms this work on metrics other than H@10 on the FB15K-237, and on the WR18RR, our model can approach the performance of this work on both H@3 and H@10 metrics. This proves that our proposed concept of restricted neighborhood enables the model to

Models	MRR	H@1	H@3	H@10
Base w/o RNPE	0.492	0.441	0.492	0.492
Base w/o RNA	0.486	0.438	0.505	0.575
Base w/o RNPE w/ GRNPE	0.498	0.441	0.525	0.605
Base w/ RNA w/ RNPE	0.552	0.486	0.590	0.671

Table 4: Comparison experiments on different RNE models on WN18RR.

achieve competitive performance on benchmarks using only first-order neighborhoods, facing challenges posed by the performance of works using second-order neighborhoods or even longer link relationships.

4.4 Ablation study

In Table 4, Base w/o RNPE represents the case where only restricted neighborhood information transfer is used; Base w/o RNA represents the case where only restricted neighborhood position encoding is used; and Base w/o RNPE w/ GRNPE represents the case where both modules are combined using the method shown in Figure 4(b) without using restricted neighborhood position encoding. As shown in Table 4, the restricted neighborhood information transfer and the restricted neighborhood position encoding can only achieve better results when used in combination.

When using restricted neighborhood information transfer alone, our model can only achieve near-best results in non-GNN-based models except for the H@1 metric; when using restricted neighborhood position encoding alone, the model cannot even outperform many non-GNN-based models on the benchmark. In theory, restricted neighborhood information complements the structural information not used by the semantic information learning module, which will improve the effect to a certain extent. However, using learnable parameters to aggregate is not optimal, as it does not express all the information of the restricted neighborhood. Therefore, our encoding method is very necessary.

When using only the encoding method, since the model does not combine effective embedding information, the addition of encoded values is meaningless, which indicates that the encoded values only play a guided role. Using the encoding method alone will not achieve good results due to lack of physical significance.

4.5 Partial results discussion

In Table 2, Condition-1 corresponds to Figure 3(a); Condition-2 corresponds to Figure 3(b); and Condition-3 corresponds to Figure 3(c). In Table 1, Situation-1 corresponds to Figure 4(d); Situation-2 corresponds to Figure 4(b); and Situation-3 corresponds to Figure 4(c).

As can be seen from Table 2, a deep network cannot encode the ordering of restricted neighborhood entities well, which is likely related to the semantic order of the modeling. The full permutation style cannot use deep-level encoding methods. As can be seen from Table 1, the combination method in Figure 4(b) does not significantly outperform over Figure 4(d), and may incur a greater computational burden in practice.

5 Conclusion and future work

The experimental results show that our proposed method has certain advantages, and the concept of restricted neighborhood proposed has certain reference significance. The performance of the model is not only competitive in non-path-dependent or non-sequential-dependent methods, but also does not completely fall behind in comparison with path- or sequence-based modeling methods.

The ablation experiment results show that the encoding method proposed in our work has a coupling relationship with the information of the restricted neighborhood when used, which directly helps the model to perform better on the benchmark.

In the possible future, related work should explore extensible representations of restricted neighborhoods, and the use of more effective encoding methods to help reduce the computational cost caused by expanding the restricted neighborhood. These encoding methods should fully consider the topological structure of the original graph. Our work proposes a possible attempt to combine structural information and semantic information for learning. A possible future is to build a better integrated network structure to learn both kinds of information.

6 Limitations

From the experiment, we found that our model has relatively significant fluctuations in performance. Although it can show some advantages under poor performance, this still indicates that our model's stability is not good enough. We did not change the structure of the semantic information learning module, which may be the reason for the model's instability, as it leads to the model's structure being insufficiently optimized, resulting in redundancy in the network structure.

In addition, our selection of the restricted neighborhood is absolute and non-extensible, which limits the model's ability to model more complex relationships. Moreover, our encoding method for the restricted neighborhood is very primitive. A possible future is to use more robust mathematical tools to map this encoding relationship.

In the end, we must admit that the computational complexity of our encoding method for the restricted neighborhood is exponentially related to the number of entities participating in the restricted neighborhood, which limits the number of entities covered by the restricted neighborhood and also limits the application of our encoding method on larger-scale knowledge graphs.

7 Ethic statement

Our work focuses on improving the performance of link prediction on benchmarks and has no particular ethical consideration.

References

Ivana Balazevic, Carl Allen, and Timothy M. Hospedales. 2019. Multi-relational poincaré graph embeddings. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019*, pages 4465–4475.

Kurt D. Bollacker, Colin Evans, Praveen K. Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008*, pages 1247–1250.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013.*, pages 2787–2795.

Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Xiaochun Cao, and Qingming Huang. 2021. Dual quaternion knowledge graph embeddings. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021*, pages 6894–6902.

Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Xiaochun Cao, and Qingming Huang. 2022. Geometry interaction knowledge graph embeddings. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI2022*, pages 5521–5529.

Ines Chami, Adva Wolf, Da-Cheng Juan, Frederic Sala, Sujith Ravi, and Christopher Ré. 2020. Low-dimensional hyperbolic knowledge graph embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 6901–6914.

Heng Chang, Jie Cai, and Jia Li. 2023. Knowledge graph completion with counterfactual augmentation. In *Proceedings of the ACM Web Conference 2023, WWW 2023*, pages 2611–2620.

Chen Chen, Yufei Wang, Bing Li, and Kwok-Yan Lam. 2022. Knowledge is flat: A seq2seq generative framework for various knowledge graph completion. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022*, pages 4005–4017.

Chen Chen, Yufei Wang, Aixin Sun, Bing Li, and Kwok-Yan Lam. 2023a. Dipping plms sauce: Bridging structure and text for effective knowledge graph completion via conditional soft prompting. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11489–11503.

Chen Chen, Yufei Wang, Aixin Sun, Bing Li, and Kwok-Yan Lam. 2023b. Dipping plms sauce: Bridging structure and text for effective knowledge graph completion via conditional soft prompting. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11489–11503.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18)*, pages 1811–1818.

Xiou Ge, Yun-Cheng Wang, Bin Wang, and C.-C. Jay Kuo. 2023. Compounding geometric operations for knowledge graph completion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023*, pages 6947–6965.

William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pages 1024–1034.

Magdalena Kaiser, Rishiraj Saha Roy, and Gerhard Weikum. 2021. Reinforcement learning from reformulations in conversational question answering over knowledge graphs. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 459–469.

Ren Li, Yanan Cao, Qiannan Zhu, Guanqun Bi, Fang Fang, Yi Liu, and Qian Li. 2022a. How does knowledge graph embedding extrapolate to unseen data: A semantic evidence view. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI2022*, pages 5781–5791.

Rui Li, Xu Chen, Chaozhuo Li, Yanming Shen, Jianan Zhao, Yujing Wang, Weihao Han, Hao Sun, Weiwei Deng, Qi Zhang, and Xing Xie. 2023. To copy rather than memorize: A vertical learning paradigm for knowledge graph completion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023*, pages 6335–6347.

Rui Li, Jianan Zhao, Chaozhuo Li, Di He, Yiqi Wang, Yuming Liu, Hao Sun, Senzhang Wang, Weiwei Deng, Yanming Shen, Xing Xie, and Qi Zhang. 2022b. House: Knowledge graph embedding with householder parameterization. In *International Conference on Machine Learning, ICML 2022*, pages 13209–13224.

Zhifei Li, Hai Liu, Zhaoli Zhang, Tingting Liu, and Neal N. Xiong. 2022d. Learning knowledge graph embedding with heterogeneous relation attention networks. *IEEE Trans. Neural Networks Learn. Syst.*, 33(8):3961–3973.

Zhifei Li, Yue Zhao, Yan Zhang, and Zhaoli Zhang. 2022c. Multi-relational graph attention networks for knowledge graph completion. *Knowl. Based Syst.*, 251:109262.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

Mojtaba Nayyeri, Bo Xiong, Majid Mohammadi, Mst. Mahfuja Akter, Mirza Mohtashim Alam, Jens Lehmann, and Steffen Staab. 2023. Knowledge graph embeddings using neural itô process: From multiple walks to stochastic trajectories. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7165–7179.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, pages 809–816.

Zile Qiao, Wei Ye, Dingyao Yu, Tong Mo, Weiping Li, and Shikun Zhang. 2023. Improving knowledge graph completion with generative hard negative mining. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5866–5878.

Andrea Rossi, Denilson Barbosa, Donatella Firmani, Antonio Matinata, and Paolo Merialdo. 2021. Knowledge graph embedding for link prediction: A comparative analysis. *ACM Trans. Knowl. Discov. Data*, 15(2):14:1–14:49.

Michael Sejr Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *The Semantic Web - 15th International Conference, ESWC 2018*, pages 593–607.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. In *7th International Conference on Learning Representations, ICLR 2019*.

Kristina Toutanova and Danqi Chen. 2015. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality, CVSC 2015*, pages 57–66.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016*, pages 2071–2080.

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha P. Talukdar. 2020. Composition-based multi-relational graph convolutional networks. In *8th International Conference on Learning Representations, ICLR 2020*.

Bo Wang, Tao Shen, Guodong Long, Tianyi Zhou, Ying Wang, and Yi Chang. 2021b. Structure-augmented text representation learning for efficient knowledge graph completion. In *WWW '21: The Web Conference 2021*, pages 1737–1748.

Liang Wang, Wei Zhao, Zhuoyu Wei, and Jingming Liu. 2022. Simkgc: Simple contrastive knowledge graph completion with pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022*, pages 4281–4294.

Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE Trans. Knowl. Data Eng.*, 29(12):2724–2743.

Xiang Wang, Tinglin Huang, Dingxian Wang, Yancheng Yuan, Zhenguang Liu, Xiangnan He, and Tat-Seng Chua. 2021a. Learning intents behind interactions with knowledge graph for recommendation. In *WWW '21: The Web Conference 2021*, pages 878–887.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, pages 1112–1119.

Bo Xiong, Shichao Zhu, Mojtaba Nayyeri, Chengjin Xu, Shirui Pan, Chuan Zhou, and Steffen Staab. 2022. Ultrahyperbolic knowledge graph embeddings. In *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2130–2139.

Chenyan Xiong, Russell Power, and Jamie Callan. 2017. Explicit semantic ranking for academic search via knowledge graph embedding. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017*, pages 1271–1279.

Hongcai Xu, Junpeng Bao, and Wenbo Liu. 2023. Double-branch multi-attention based graph neural network for knowledge graph completion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023*, pages 15257–15271.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *3rd International Conference on Learning Representations, ICLR 2015*.

Jinfa Yang, Xianghua Ying, Yongjie Shi, Xin Tong, Ruibin Wang, Taiyan Chen, and Bowei Xing. 2022. Learning hierarchy-aware quaternion knowledge graph embeddings with representing relations as 3d rotations. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022*, pages 2011–2023.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. KG-BERT: BERT for knowledge graph completion. *CoRR*, abs/1909.03193.

Yongqi Zhang and Quanming Yao. 2022. Knowledge graph reasoning with relational digraph. In *WWW '22: The ACM Web Conference 2022*, pages 912–924.

Yongqi Zhang, Zhanke Zhou, Quanming Yao, Xiaowen Chu, and Bo Han. 2023a. Adaprop: Learning adaptive propagation for graph neural network based knowledge graph reasoning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023*, pages 3446–3457.

Yongqi Zhang, Zhanke Zhou, Quanming Yao, Xiaowen Chu, and Bo Han. 2023b. Adaprop: Learning adaptive propagation for graph neural network based knowledge graph reasoning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023*, pages 3446–3457.

Zhaocheng Zhu, Xinyu Yuan, Mikhail Galkin, Sophie Xhonneux, Ming Zhang, Maxime Gazeau, and Jian Tang. 2023. A*net: A scalable path-based reasoning approach for knowledge graphs. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Zhaocheng Zhu, Zuobai Zhang, Louis-Pascal A. C. Xhonneux, and Jian Tang. 2021. Neural bellman-ford networks: A general graph neural network framework for link prediction. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021*, pages 29476–29490.

A Appendix

A.1 Setup

In experiment, a standard evaluation protocol is used in filtered setting (Bordes et al., 2013): all true triplets are filtered out during evaluation in the datasets. H@n (n=1, 3, 10) and the Mean Reciprocal Rank (MRR) are used to evaluate our model against baselines.

Our model follow a encoder-decoder manner. As shown in Figure 2, the framework of RNE contains two modules: the structural information learning module and the semantic information learning module. The former one present two decoder: restricted neighborhood information transfer and restricted neighborhood position encoding. The Latter one choose GIE (Cao et al., 2022) as decoder. It should be noted that our model is not GIE-specific. In practice, we selected several KGE models, such as TransE (Bordes et al., 2013) to be the decoder. On WN18RR, our method improves its performance (MRR, H@1, H@3, H@10) by (0.427, 0.353, 0.470, 0.555) from (0.262, 0.109, 0.374, 0.531), respectively. We chose GIE as the decoder because its framework structure is simple, its code is standardized, and its work is easily reproducible. Most importantly, our model achieved performance matching state-of-the-art works' performance when using GIE as the decoder.

We run our experiments on a single GPU. The optimal hyper-parameters of our model are determined by hyper-parameter search. The embedding size is searched in {250, 300, 350, 400}. Learning rate is searched in {0.001, 0.005, 0.01, 0.05, 0.1}. Batch size is searched in {250, 300, 350, 400, 450, 500, 550}.

A.2 Number of parameters

Our statistics of total parameters include statistics of the embedding layer. On WN18RR, the total number of parameters is 36.38M. On FB15K-237, the total number of parameters is 56.59M.

A.3 tables

Models	MRR	H@1	H@3	H@10
Situation-1	0.493	0.444	0.513	0.587
Situation-2	0.492	0.441	0.514	0.586
Situation-3	0.490	0.438	0.513	0.591

Table 5: The comparison experiments on different aggregation methods of neighborhood information on WN18RR.

Datasets	Entities	Relations	Train triplets	Valid triplets	Test triplets
WR18RR	40943	11	86835	3034	3134
FB15K-237	14541	237	272115	17535	20466

Table 6: Statistical data of WN18RR and FB15K-237.

	L<1	L<2	L<3	L<4	L<5
WN18RR-out	1428	18998	30874	36207	38471
WN18RR-in	9236	26792	32806	35714	37356
	L<1	**L<2**	**L<5**	**L<10**	**L<20**
FB15K-237-out	760	1505	3656	6325	10413
FB15K-237-in	1162	2117	5016	8365	11882

Table 7: Statistics of the number of entities in the neighborhood. WN18RR-out represents considering only the outgoing edges of the anchor entities; WN18RR-in represents considering only the incoming edges of the anchor entities. FB15K-237-out represents considering only the outgoing edges of the anchor entities; FB15K-237-in represents considering only the incoming edges of the anchor entities. L represents the neighborhood size.

Models	MRR	H@1	H@3	H@10
RNE-out w/o RNPE	0.492	0.441	0.513	0.590
RNE-out-in w/o RNPE	0.488	0.430	0.514	0.598

Table 8: On WN18RR, the comparison experiments between the case where both incoming and outgoing edges are selected and the case where only outgoing edges are selected. The model used is the first scenario in Table 5.

Length	MRR	H@1	H@3	H@10
100epoch				
10	0.481	0.425	0.506	0.586
5	0.487	0.431	0.512	0.592
2	0.487	0.432	0.511	0.591
150epoch				
5	0.492	0.441	0.513	0.590
2	0.490	0.435	0.513	0.593

Table 9: The comparison experiment on different size neighborhoods on WN18RR. Length refers to the size of the limited neighborhood. The model used is the first case in Table 5.