

Research Article

A Knowledge Representation Method for Question Answering Service in Mobile Edge Computing Environment

Rong Qian  and Xia Hou 

Beijing Information Science and Technology University, Beijing, China

Correspondence should be addressed to Xia Hou; houxia@bistu.edu.cn

Received 4 November 2021; Revised 11 January 2022; Accepted 20 January 2022; Published 17 February 2022

Academic Editor: Xiaolong Xu

Copyright © 2022 Rong Qian and Xia Hou. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Driven by the rapid development of mobile computing and the Internet of Things, the number of devices connected to Internet has increased dramatically in recent years. Such development has generated massive amounts of data and highlighted the importance and urgency of using the accumulated big data to improve frequently used services. Deploying the question answering service in the mobile edge computing environment is considered a good way to make efficient use of the data and improve user experiences. Powered by the breakthroughs of deep learning technologies, question answering system based on knowledge graph (KBQA) has flourished in recent years. Knowledge representation, as a key technology of KBQA, can express the knowledge graph as the vectors containing more semantic information and thereby improve the accuracy of the question answering system. This paper proposes a knowledge representation method that integrates more features than the traditional methods. In our method, knowledge is represented as a combination of a structured vector reflecting the target triple and the domain information around the entity. By representing richer semantic vectors, our method outweighs TransE, ConvE, and KBAT, in terms of link prediction.

1. Introduction

The explosive growth of smart phones and other mobile terminals and the emergence of many new applications have made a great impact on mobile and wireless networks [1–3]. The traditional centralized network cannot meet the needs of mobile users due to heavy load and long delay [4–6]. Therefore, a new architecture is proposed to open network capabilities from the core network to the edge network, i.e., mobile edge computing [7–9]. Mobile edge computing deploys the services and functions originally located in the cloud data center to the edge of the mobile network and provides computing, storage, network, and communication resources at the edge of the mobile network [10, 11].

Question answering system is an advanced form of information retrieval system, which can use accurate and concise natural language to answer users' questions. KBQA is different from other types of question answering systems in that KBQA uses knowledge graphs to provide a highly structured knowledge source for the question answering

system. Knowledge graph (KG) is a large scale multi-relationship graph, consisting of entities and their relationships. A few of the existing large knowledge graphs include Freebase, DBpedia, YAGO, and XLORE. Although these knowledge graphs are large in scale, they are far from complete. In order to address this problem, link prediction is proposed. The common way to solve this task is to learn a low-dimensional representation of all entities and relationships and use them to predict new facts, also known as knowledge representation learning [12].

In recent years, many knowledge representation models have been proposed. The most classic model is TransE [13], which has been inspired by Word2vec. TransE can only be used in one-to-one relationships and cannot express polysemous words. Based on TransE, researchers have proposed many extended models. TransH [14] introduces a specific relationship Hyperplane, so that different entities have different representations under different relationships. TransR [15] makes different relationships to have different semantic spaces. Neural networks have also been widely

used recently to build knowledge representation models. R-GCN [16] uses graph convolutional networks [17] (GCN) to represent relationships. ConvKB [18] combines the triple vector into a 3-column matrix and inputs it into the convolutional layer, which is represented by one-dimensional convolution. ConvE [19] uses two-dimensional convolution and multiple nonlinear features to model triples. Whether it is a traditional Trans model or a neural network-based model, each triple is processed independently, and the rich semantic information around the entity cannot be used.

In this paper, we propose KRDCG, a knowledge representation method that integrates multiple features. In order to learn the rich semantic information of nodes in the knowledge graph, KRDCG not only uses the structural information of the triple but also considers the neighborhood information around the entity. KRDCG uses TransD to represent structure information, so that different head and tail entities can be mapped to different relationship spaces according to their own attributes and relationship characteristics. The improved graph attention networks (GATs) can give different weights to adjacent entities according to different relationships between entities. We use improved GAT to obtain the neighborhood information within two hops of the entity. Finally, KRDCG uses the capsule network as a decoder. Our contributions in this paper are as follows:

- (1) We propose an end-to-end knowledge representation model KRDCG that combines the advantages of TransD and GAT.
- (2) We use the capsule network as a decoder to extract features of the same dimension in multiple feature maps.
- (3) We evaluate our KRDCG for link prediction on benchmark datasets FB15K-237 and WN18RR. KRDCG obtains the best mean rank and highest Hits@3.

The rest of the paper is organized as follows. Section 2 provides a review of background. In Section 3, we introduce the classic knowledge representation model. Section 4 reports experimental results and datasets' descriptions followed by our conclusion and future research directions in Section 5.

2. Background

Mobile edge computing provides content storage, computing, and distribution services near the mobile user side through in-depth cooperation with content providers and application developers [20, 21]. This enables applications, services, and content to be deployed in a highly distributed environment to better meet low latency and high bandwidth requirements [22–24]. For our system, the overall deployment framework is shown in Figure 1.

Recently, knowledge representation learning models can be divided into three categories: (1) using the structural information of the triple itself, (2) utilizing external information such as text, images, or rules, and (3) fusion of entity neighborhood information.

Models based on structured information can be divided into two categories, traditional translation models and neural network-based models. Traditional models based on translation include TransE, TransH, TransR, and TransD [25]. These models do not consider any information other than triples. TransH and TransR focus on the multiple representations of entities in different relations, improving the performance on knowledge completion and triple classification. However, both models only project entities according to the relations in triples, ignoring the diversity of entities. To address this problem, TransD proposes a novel projection method with a dynamic mapping matrix depending on both entities and relations, which takes the diversity of entities as well as relations into consideration. Inspired by the fact that concentric circles in the polar coordinate can naturally reflect the hierarchical structure, HAKE [26] was proposed. HAKE can effectively model the semantic levels in the knowledge graph and has a good performance in link prediction tasks.

Models based on neural networks include ConvKB and ConvE, as well as CapsE [27]. CapsE uses the capsule network to model triples. It has a “deep” architecture for modeling the entries in a triple at the same dimension. The number of interactions that ConvE can capture is limited, so InteractE [28] was proposed. InteractE increases the number of interactions between entities and relationships. It proves that increasing the number of interactions can improve the performance of link prediction.

DKRL [29] utilizes text information to model the corresponding triples and entity description information. TKRL [30] makes good use of the hierarchical information of entities. Compared with TransE and TransR, its performance has been improved by 11.3% and 6.2%, respectively. KALE [31] combines knowledge graphs and logic rules and then expresses and models them in a unified framework. This method can make better predictions outside the scope of pure logical reasoning, but the logic rules are more limited. IKRL [32] is the first attempt to combine images with knowledge graphs for KRL. Its promising performances indicate the significance of visual information for KRL. Although the use of information outside the knowledge graph can add semantics to entities, not all entities have access to additional information, and it is not universally applicable.

PTransE [33] is a path-based model. It combines multiple relationships of entities semantically to obtain a vector representation of the path. GAKE [34] defines three contexts with entities and relationships as subjects and uses these kinds of contextual information for modeling. TCE [35] improves on GAKE and proposes two types of context information, path context and neighborhood context. KBAT [36] is a novel attention-based feature embedding model that captures both entity and relation features in any given entity's neighborhood.

In order to make the model more effective, we refer to KBAT as the basic model. On the basis of making full use of entity neighborhood information, combined with triple structured information, the capsule network is used to extract more in-depth information.

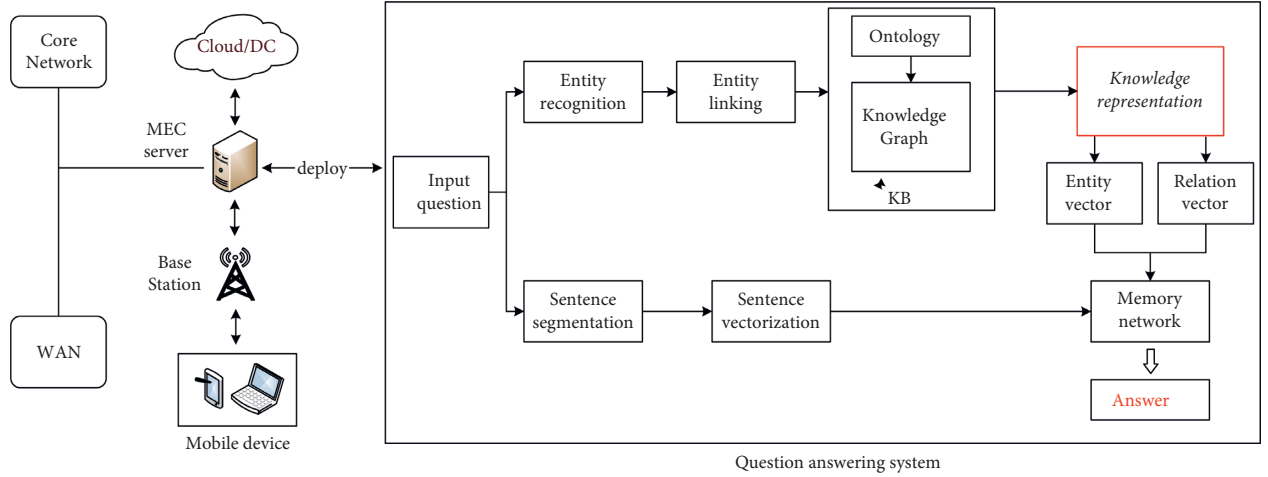


FIGURE 1: The overall structure and deployment of the question and answering system.

3. The Proposed KRDGC Model

In this section, we describe the proposed method. We first define notations. A knowledge graph $\mathcal{G} \in (\mathcal{E}, \mathcal{R})$ is a collection of valid factual triples in the form of (head entity, relation, tail entity) denoted as (h, r, t) such that $h, t \in \mathcal{E}$ and $r \in \mathcal{R}$ where \mathcal{E} is a set of entities and \mathcal{R} is a set of relations. Embedding model aims to define a score function giving a score for each triple, such that valid triples receive higher scores than invalid triples. The overall structure is shown in Figure 2.

3.1. Structured Features. We use TransD to model the structural features of triples. As illustrated in Figure 3, TransD sets up two projection matrices \mathbf{M}_{rh} and \mathbf{M}_{rt} that, respectively, project the head entity and the tail entity into the relational space. The specific definitions are as follows:

$$\begin{aligned} \mathbf{M}_{rh} &= \mathbf{I}_r \mathbf{I}_{h_p} + \mathbf{I}^{d \times k}, \\ \mathbf{M}_{rt} &= \mathbf{I}_r \mathbf{I}_{t_p} + \mathbf{I}^{d \times k}, \end{aligned} \quad (1)$$

where $\mathbf{I}_{h_p}, \mathbf{I}_{t_p} \in \mathbf{R}^d$, $\mathbf{I}_r \in \mathbf{R}^k$, and the subscript p represents that the vector is a projection vector. Therefore, the mapping matrices are determined by both entities and relations. Compared with other Trans models, TransD makes the two projection vectors interact sufficiently because each element of them can meet every entry coming from another vector.

3.2. Graph Attention Networks with Relations. In the KG, entities and relationships are not independent, and they all have an impact on each other. The local neighbor nodes of the entity contain a lot of important hidden semantic information. In this paper, we use GAT to extract hidden features in the neighborhood of an entity. The original GAT only considers entities, ignoring edge information. We use the method proposed by KBAT to redefine an attention layer.

The structure of the graph attention network is shown in Figure 4. In order to update the vector of entity e_i , a linear transformation layer is used to learn the vector representation of the combination of entities and relations in a

specific triple $t_{ijk} = (e_i, r_k, e_j)$. The corresponding vector after the combination is

$$c_{ijk} = \mathbf{W}_1 [e_i] [r_k] [e_j], \quad (2)$$

where \mathbf{W}_1 denotes the linear transformation matrix. Then, we obtain the absolute attention value of the triple through another linear transformation matrix \mathbf{W}_2 and the LeakyReLU nonlinearity.

$$b_{ijk} = \text{Leaky Rule}(\mathbf{W}_2 c_{ijk}). \quad (3)$$

We use softmax to normalize b_{ijk} to get the relative attention value.

$$\begin{aligned} \alpha_{ijk} &= \text{softmax}(b_{ijk}) \\ &= \frac{\exp(b_{ijk})}{\sum_{t \in N_i} \sum_{r \in R_{it}} \exp(b_{itr})}, \end{aligned} \quad (4)$$

where N_i is the neighborhood of entity e_i and R_{it} is the set of relations between entities e_i and e_j . The new vector representation of entity e_i is obtained by weighted summation of all neighborhood according to the relative attention value and is stabilized through the multihead attention mechanism.

$$e'_i = \sigma \left(\sum_{m=1}^2 \sum_{j \in N_i} \sum_{k \in R_{ij}} \alpha_{ijk}^m c_{ijk}^m \right), \quad (5)$$

where m represents the m -th attention head and σ represents any nonlinear function.

In order to keep the relationship dimension and entity dimension consistent, the relationship vector is updated through linear transformation in the GAT. In the last layer, the new vector and the original vector are linearly combined through the weight matrix to prevent the loss of the original information of the entity.

3.3. Capsule Network. After GAT training, we use improved CapsE [27] as a decoder in our model. It uses a three-column matrix to represent each triple. First, we use CNN to perform

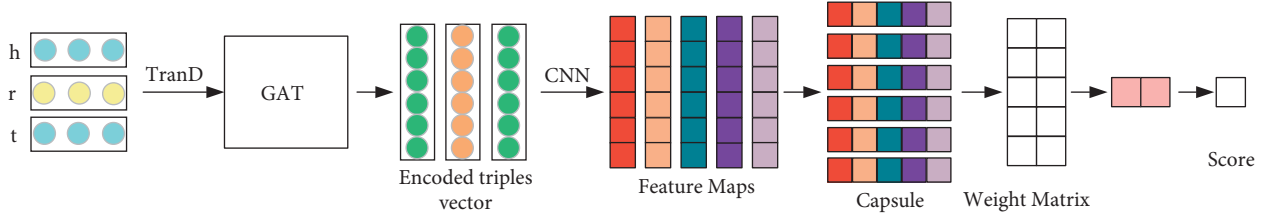


FIGURE 2: The overall structure of our model.

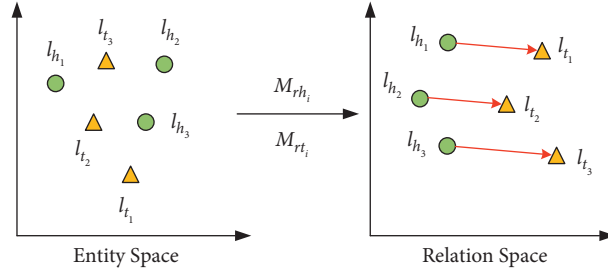


FIGURE 3: TransD model (entities and relationships' mapping matrix).

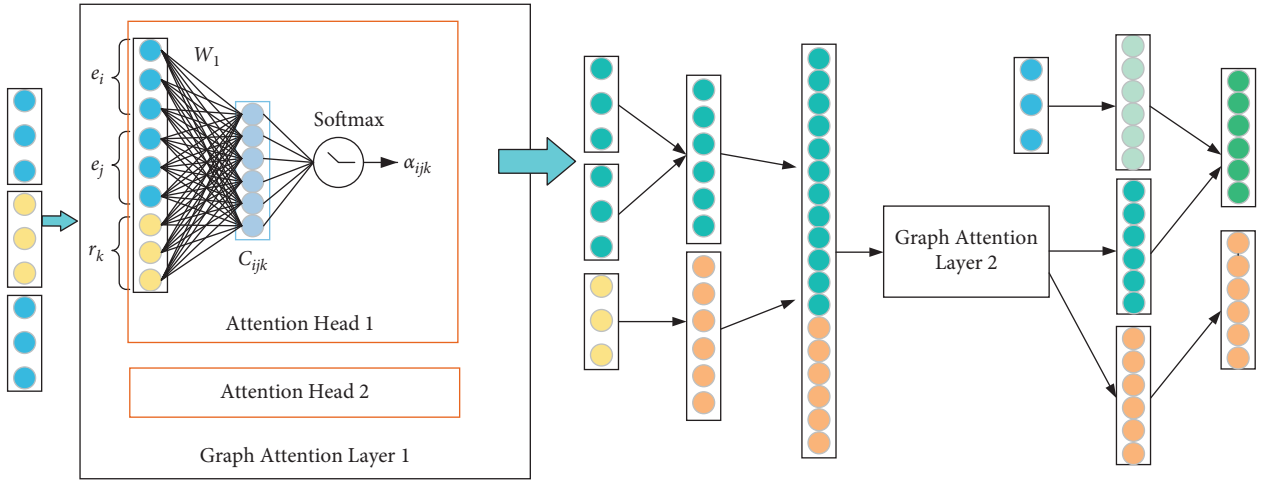


FIGURE 4: The structure of GAT with relations.

convolution operation on the triple vector to generate multiple different feature maps. All feature maps with the same dimension features are encapsulated into corresponding capsules. Therefore, each capsule can capture the different characteristics of the corresponding dimensions of the embedded triples. The products of these capsules and different weights generate smaller-dimensional capsules, and one continuous vector is obtained. The vector and the weight vector perform dot product operation again to obtain the corresponding score, and the result of the sum of all the scores is used to judge the correctness of the given triple. The score function for the triple is as follows:

$$f(h, r, t) = \|\text{caps}(g([v_h, v_r, v_t] * \Omega))\|, \quad (6)$$

where caps denotes a capsule network operator and g is an activation function. Because we use ReLU in this paper, Ω is the shared parameter in the convolution layer. The model is trained using the loss function as follows:

$$\mathcal{L} = \sum_{(h,r,t) \in \{\mathcal{T} \cup \mathcal{T}'\}} \log(1 + \exp((-t_{(h,r,t)} \cdot f(h, r, t))) + \lambda \|\omega\|_2^2, \quad (7)$$

in which

$$t_{(h,r,t)} = \begin{cases} 1, & \text{for } (h, r, t) \in \mathcal{T}, \\ -1, & \text{for } (h, r, t) \in \mathcal{T}'. \end{cases} \quad (8)$$

The construction of negative triples is to replace the head entity and tail entity of the correct triple with all the entities in the dataset.

4. Experiments and Analysis

4.1. Datasets. In our experiments, we use two widely used benchmark datasets FB15K-237 [37] and WN18RR [19] for evaluation of the performance of link prediction.

FB15K-237 is extracted from Freebase. It contains 14,541 entities and 237 relations. It is an improved version of FB15K dataset where all inverse relations are deleted to prevent direct inference of test triples by reversing train triples. WN18RR is created from WN18, which is a subset of WordNet. WN18 consists of 18 relations and 40,943 entities. Similar to FB15K dataset, all inverse relations are deleted to prevent direct inference of test triples by reversing train triples. WN18RR contains 40,943 entities and 11 relations. Details of the datasets are summarized in Table 1.

4.2. Link Prediction. In the link prediction task, the purpose is to predict a missing entity given a relation and another entity. In a specific experiment, for each triple in the test set, we remove the head or tail entity and then replace it with all the entities in dictionary in turn. We first compute scores of those corrupted triplets and then rank them by descending order; the rank of the correct entity is finally stored. The task emphasizes the rank of the correct entity instead of only finding the best one.

4.3. Evaluation Protocol. We use the filtered setting protocol, i.e., not taking any corrupted triples that appear in the KB into accounts. We rank the valid test triple and corrupted triples in descending order of their scores. We employ evaluation metrics: MR, MRR, Hits@1, Hits@3, and Hits@10 (i.e., the proportion of the valid test triples ranking in top 1, 3, and 10 predictions).

MR means mean rank, and its specific calculation method is as follows:

$$\text{MR} = \frac{1}{|S|} \sum_{i=1}^{|S|} \text{rank}_i = \frac{1}{|S|} (\text{rank}_1 + \dots + \text{rank}_{|S|}), \quad (9)$$

where S is a set of triples, $|S|$ is the number of triple sets, and rank_i refers to the link prediction ranking of the i -th triple. The smaller the indicator is, the better the performance is.

MRR means mean reciprocal ranking. The specific calculation method is as follows:

$$\text{MRR} = \frac{1}{|S|} \sum_{i=1}^{|S|} \frac{1}{\text{rank}_i} = \frac{1}{|S|} \left(\frac{1}{\text{rank}_1} + \dots + \frac{1}{\text{rank}_{|S|}} \right). \quad (10)$$

The symbols involved in the above formula are the same as those involved in the MR calculation formula. The bigger the indicator is, the better the performance is.

HITS@ n refer to the average proportion of triples that rank less than n in link prediction. The specific calculation method is as follows:

$$\text{HITS}@n = \frac{1}{|S|} \sum_{i=1}^{|S|} \mathbb{I}(\text{rank}_i \leq n), \quad (11)$$

where $\mathbb{I}(\cdot)$ is the indicator function.

Lower MR, higher MRR, or higher Hits@1,3,10 indicate better performance. Final scores on the test set are reported for the model obtaining the highest Hits@1,3,10 on the validation set.

4.4. Training Protocol. We first train TransD for 1000 and 3000 epochs on FB15K-237 and WN18RR, respectively. Then, we get a 200-dimensional entity and relationship vectors and use the vectors to initialize entity and relation embeddings in GAT. In the GAT, we use the following hyperparameters for training to select the optimal result. We use Adam learning rate $lr \in \{1e^{-4}, 5e^{-4}, 5e^{-3}, 1e^{-3}\}$, l_1 -norm or l_2 -norm, margin $\in \{1, 3, 5, 7\}$, and dropout $\in \{0.1, 0.2, 0.3\}$. The highest Hits@10 and Hits@3 scores and lower MR on the validation set are obtained when we use learning rate at $1e^{-3}$, l_2 -norm, margin = 1, and dropout = 0.3 for FB15K-237 and learning rate at $1e^{-3}$, l_2 -norm, margin = 5, and dropout = 0.3 for WN18RR.

After GAT, we train capsule network for 200 epochs. We set batch size to 128. We use the Adam optimizer with the initial learning rate $\in \{5e^{-6}, 1e^{-5}, 5e^{-5}, 1e^{-4}\}$. We monitor the MRR score after each training epoch and obtain the highest MRR score on the validation set when using the initial learning rate at $5e^{-5}$.

4.5. Results and Analysis. Table 2 compares the experimental results of our model with the common classic methods.

Table 3 shows the comparison between our model and some traditional methods on FB15K-237 and WN18RR. These methods only consider the structure vector of the triples and do not consider the semantic information around the triples.

Table 4 compares the experimental results of our model with previous published results of only using neural network methods.

Compared to the baseline method KBAT, our model outperforms it on FB15K-237 across all the metrics and on three metrics for WN18RR. Figures 5 and 6 show that KRDCG gains significant improvement of $0.527 - 0.518 = 0.009$ in MRR (which is 1.7% relative improvement) and $0.662 - 0.626 = 0.036$ in Hits@10 (which is 3.6% absolute improvement) on FB15K-237. We confirm previous findings that KBAT in fact is a strong baseline model, e.g., KBAT obtains better MRR and Hits@1 than KRDCG on WN18RR.

In Figure 7, KRDCG achieves better performance of $210 - 158 = 52$ (which is about 25% relative improvement) and $1940 - 1850 = 90$ (which is about 4.6% relative improvement) on MR for FB15K-237 and WN18RR, respectively.

In summary, combining structural information with neighborhood information can capture more semantic information and improve the effect of knowledge representation learning. The capsule network can extract more semantic information in the same dimension of the feature maps.

TABLE 1: Details of the datasets used.

Category	FB15K-237		WN18RR
#Entities	14,541		40,943
#Relations	237		11
#Train	271,115		86,835
#Valid	17,535		3034
#Test	20,466		3134

TABLE 2: Experimental result on FB15K-237 and WN18RR.

	FB15K-237					WN18RR				
	MR	MRR	Hits@10	Hits@3	Hits@1	MR	MRR	Hits@10	Hits@3	Hits@1
TransE	323	0.279	0.441	0.376	0.198	2300	0.243	0.532	0.441	0.043
RotatE	177	0.338	0.533	0.375	0.241	3340	0.476	0.571	0.492	0.428
TuckER	–	0.358	0.544	0.394	0.266	–	0.470	0.526	0.482	0.443
KBAT	210	0.518	0.626	0.54	0.46	1940	0.44	0.581	0.483	0.361
KRDGC	158	0.527	0.662	0.563	0.46	1850	0.424	0.584	0.489	0.34

TABLE 3: Comparison result with some traditional methods on FB15K-237 and WN18RR.

	FB15K-237					WN18RR				
	MR	MRR	Hits@10	Hits@3	Hits@1	MR	MRR	Hits@10	Hits@3	Hits@1
DistMult	254	0.241	0.419	0.263	0.155	5110	0.430	0.490	0.440	0.390
ComplEx	339	0.247	0.428	–	0.158	5261	0.440	0.510	–	0.410
HAKE	–	0.346	0.542	0.381	0.250	–	0.497	0.582	0.516	0.452
InteractE	172	0.354	0.535	–	0.263	5202	0.463	0.528	–	0.430
KRDGC	158	0.527	0.662	0.563	0.46	1850	0.424	0.584	0.489	0.34

TABLE 4: Comparison result with neural network methods on FB15K-237 and WN18RR.

	FB15K-237					WN18RR				
	MR	MRR	Hits@10	Hits@3	Hits@1	MR	MRR	Hits@10	Hits@3	Hits@1
CapsE	303	0.523	0.593	–	–	719	0.415	0.56	–	–
ConvKB	216	0.289	0.471	0.327	0.198	1295	0.265	0.558	0.445	0.058
ConvE	245	0.312	0.497	0.341	0.225	4464	0.456	0.531	0.47	0.419
SCAN	–	0.35	0.54	0.39	0.26	–	0.47	0.54	0.48	0.43
R-GCN	–	0.249	0.417	0.264	0.151	–	–	–	–	–
KRDGC	158	0.527	0.662	0.563	0.46	1850	0.424	0.584	0.489	0.34

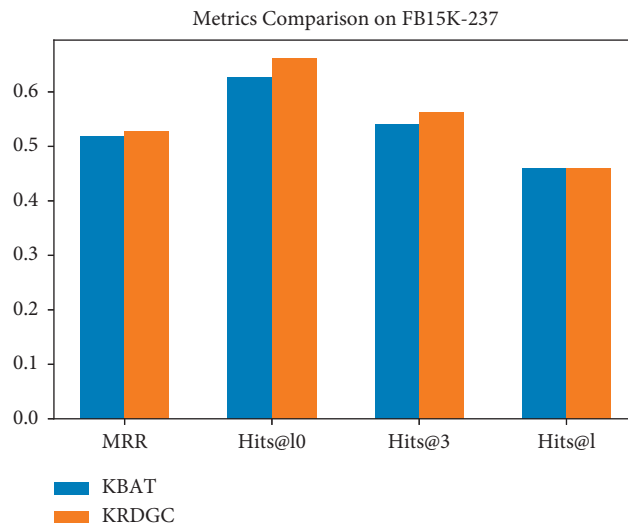


FIGURE 5: Comparison results of models KBAT and KRDGC on FB15K-237.

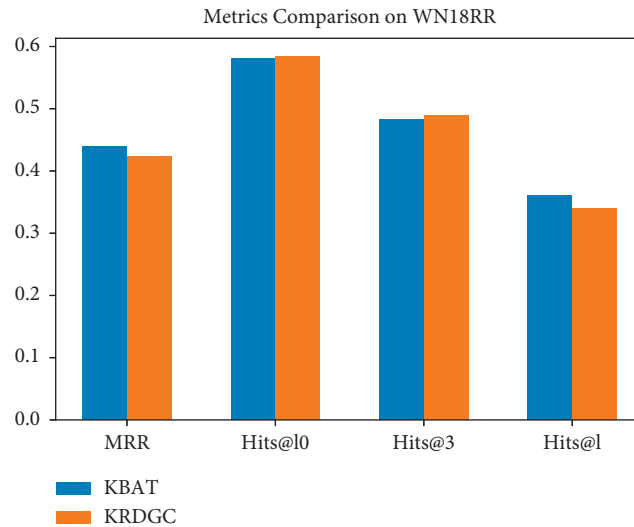


FIGURE 6: Comparison results of models KBAT and KRDGC on WN18RR.

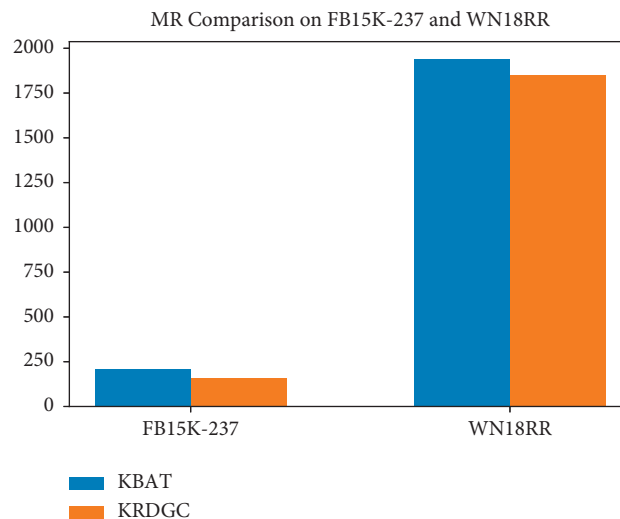


FIGURE 7: MR comparison on FB15K-237 and WN18RR.

5. Conclusion and Future Work

In this paper, we proposed a KG embedding model KRDGC which is able to take advantage of the structure and neighborhood information of the triple. Our method uses TransD to model structural information and GAT to model neighborhood information and finally extracts deep features through a capsule network. We evaluate our model on link prediction, and the experimental results show significant improvements over the major baselines. In the future, we would like to further represent the relation vector in GAT, incorporate our model into the KBQA system, and verify its response time in mobile edge computing.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (no. 61602044). The authors are grateful to the research of <http://export.arxiv.org/pdf/1808.04122> for providing new ideas.

References

- [1] Z. Tong, F. Ye, M. Yan, H. Liu, and S. Basodi, "A survey on algorithms for intelligent computing and smart city applications," *Big Data Mining and Analytics*, vol. 4, no. 3, pp. 155–172, 2021.

- [2] R. Bi, Q. Liu, J. Ren, and G. Tan, "Utility aware offloading for mobile-edge computing," *Tsinghua Science and Technology*, vol. 26, no. 2, pp. 239–250, 2021.
- [3] J. Huang, B. Lv, Y. Wu, Y. Chen, and X. Shen, "Dynamic admission control and resource allocation for mobile edge computing enabled small cell network," *IEEE Transactions on Vehicular Technology*, p. 1, 2021.
- [4] Y. Chen, N. Zhang, Y. Zhang, X. Chen, W. Wu, and X. S. Shen, "Toffee: task offloading and frequency scaling for energy efficiency of mobile devices in mobile edge computing," *IEEE Transactions on Cloud Computing*, vol. 9, no. 4, pp. 1634–1644, 2021.
- [5] D. Kim, J. Son, D. Seo, Y. Kim, H. Kim, and J. T. Seo, "A novel transparent and auditable fog-assisted cloud storage with compensation mechanism," *Tsinghua Science and Technology*, vol. 25, no. 1, pp. 28–43, 2020.
- [6] W. Zhang, X. Chen, and J. Jiang, "A multi-objective optimization method of initial virtual machine fault-tolerant placement for star topological data centers of cloud systems," *Tsinghua Science and Technology*, vol. 26, no. 1, pp. 95–111, 2021.
- [7] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: a survey," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450–465, 2018.
- [8] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: the communication perspective," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.
- [9] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge ai: intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Network*, vol. 33, no. 5, pp. 156–165, 2019.
- [10] Y. Liu, D. Li, S. Wan et al., "A long short-term memory-based model for greenhouse climate prediction," *International Journal of Intelligent Systems*, vol. 37, no. 1, pp. 135–151, 2021.
- [11] Y. N. Malek, M. Najib, M. Bakhouya, and M. Essaïdi, "Multivariate deep learning approach for electric vehicle speed forecasting," *Big Data Mining and Analytics*, vol. 4, no. 1, pp. 56–64, 2021.
- [12] S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. S. Yu, "A survey on knowledge graphs: representation, acquisition, and applications," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 2, pp. 494–514, 2022.
- [13] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," *Advances in Neural Information Processing Systems*, vol. 26, pp. 2787–2795, 2013.
- [14] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in, *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 28, no. 1, pp. 1112–1119, 2014.
- [15] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pp. 2181–2187, Austin, TX, USA, January 2015.
- [16] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *The Semantic Web, ESWC 2018, Lecture Notes in Computer Science*, vol. 10843, pp. 593–607, Springer, 2018.
- [17] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proceedings of the International Conference On Learning Representations (ICLR)*, Toulon, France, April 2017.
- [18] T. D. Nguyen, D. Q. Nguyen, and D. Phung, "A novel embedding model for knowledge base completion based on convolutional neural network," in *Proceedings of the Conference Of the North American Chapter Of the Association For Computational Linguistics: Human Language Technologies*, vol. 2, pp. 327–333, New Orleans, LA, USA, June 2018.
- [19] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "Convolutional 2d knowledge graph embeddings," in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*. AAAI, pp. 1811–1818, Toronto, Canada, February 2018.
- [20] J. Mabrouki, M. Azroul, G. Fattah, D. Dhiba, and S. E. Hajjaji, "Intelligent monitoring system for biogas detection based on the internet of things: mohammedia, Morocco city landfill case," *Big Data Mining and Analytics*, vol. 4, no. 1, pp. 10–17, 2021.
- [21] J. Huang, C. Zhang, and J. Zhang, "A multi-queue approach of energy efficient task scheduling for sensor hubs," *Chinese Journal of Electronics*, vol. 29, no. 2, pp. 242–247, 2020.
- [22] Y. Chen, F. Zhao, Y. Lu, and X. Chen, "Dynamic task offloading for mobile edge computing with hybrid energy supply," *Tsinghua Science and Technology*, 2021.
- [23] L. Qi, X. Wang, X. Xu, W. Dou, and S. Li, "Privacy-aware cross-platform service recommendation based on enhanced locality-sensitive hashing," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, pp. 1145–1153, 2021.
- [24] J. Huang, Y. Lan, and M. Xu, "A simulation-based approach of qos-aware service selection in mobile edge computing," *Wireless Communications and Mobile Computing*, vol. 2018, pp. 1–10, 2018.
- [25] G. Ji, S. He, L. Xu, K. Liu, and J. Zhao, "Knowledge graph embedding via dynamic mapping matrix," vol. 1, pp. 687–696, in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, vol. 1, Long Papers, Beijing, China, July 2015.
- [26] Z. Zhang, J. Cai, Y. Zhang, and J. Wang, "Learning hierarchy-aware knowledge graph embeddings for link prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 3065–3072, New York, NY, USA, February 2020.
- [27] D. Q. Nguyen, T. Vu, T. D. Nguyen, and D. Q. Nguyen, "A capsule network-based embedding model for knowledge graph completion and search personalization," vol. 1, pp. 2180–2189, in *Proceedings of the Conference Of the North American Chapter Of the Association For Computational Linguistics: Human Language Technologies*, vol. 1, Long and Short Papers, Minneapolis, MA, USA, June 2019.
- [28] S. Vashishth, S. Sanyal, V. Nitin, N. Agrawal, and P. Talukdar, "Interact: improving convolution-based knowledge graph embeddings by increasing feature interactions," in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 3009–3016, Beijing, China, April 2020.
- [29] R. Xie, Z. Liu, J. Jia, H. Luan, and M. Sun, "Representation learning of knowledge graphs with entity descriptions," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, pp. 2659–2665, 2016.
- [30] R. Xie, Z. Liu, and M. Sun, "Representation learning of knowledge graphs with hierarchical types," in *Proceedings of the IJCAI*, pp. 2965–2971, NY, USA, July 2016.
- [31] S. Guo, Q. Wang, L. Wang, B. Wang, and L. Guo, "Jointly embedding knowledge graphs and logical rules," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 192–202, Austin, TX, USA, November 2016.

- [32] R. Xie, Z. Liu, H. Luan, and M. Sun, "Image-embodied knowledge representation learning," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, Melbourne, Australia, August 2017.
- [33] Y. Lin, Z. Liu, H. Luan, M. Sun, S. Rao, and S. Liu, "Modeling relation paths for representation learning of knowledge bases," in *Proceedings of the 2015 Conference On Empirical Methods In Natural Language Processing*, pp. 705–714, Lisbon, Portugal, September 2015.
- [34] J. Feng, M. Huang, Y. Yang, and X. Zhu, "Gake graph aware knowledge embedding," in *Proceedings of the COLING the 26th International Conference on Computational Linguistics Technical Papers*, pp. 641–651, Osaka, Japan, December 2016.
- [35] J. Shi, H. Gao, G. Qi, and Z. Zhou, "Knowledge graph embedding with triple context," in *Proceedings of the 2017 ACM on Conference On Information And Knowledge Management*, pp. 2299–2302, Singapore, September 2017.
- [36] D. Nathani, J. Chauhan, C. Sharma, and M. Kaul, "Learning attention-based embeddings for relation prediction in knowledge graphs," in *Proceedings of the 57th Annual Meeting Of the Association For Computational Linguistics*, pp. 4710–4723, Florence, Italy, August 2019.
- [37] K. Toutanova, D. Chen, P. Pantel, H. Poon, P. Choudhury, and M. Gamon, "Representing text for joint embedding of text and knowledge bases," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1499–1509, Lisbon, Portugal, September 2015.