Modeling Long-distance Node Relations for KBQA with Global Dynamic Graph

Xu Wang[1], Shuai Zhao[1]*, Jiale Han[1], Bo Cheng[1], Hao Yang[2], Jianchang Ao[2], Zhenzi Li[2]
[1]State Key Laboratory of networking and switching technology,
Beijing University of Posts and Telecommunications, Beijing, China
[2] 2012 Labs, Huawei Technologies CO., LTD, Beijing, China
{wxx,zhaoshuaiby,hanjl,chengbo}@bupt.edu.cn,
{yanghao30,aojianchang,lizhenzi}@huawei.com,

Abstract

The structural information of Knowledge Bases (KBs) has proven effective to Question Answering (QA). Previous studies rely on deep graph neural networks (GNNs) to capture rich structural information, which may not model node relations in particularly long distance due to oversmoothing issue. To address this challenge, we propose a novel framework **GlobalGraph**, which models long-distance node relations from two views: 1) Node type similarity: GlobalGraph assigns each node a global type label and models long-distance node relations through the global type label similarity; 2) Correlation between nodes and questions: we learn similarity scores between nodes and the question, and model long-distance node relations through the sum score of two nodes. We conduct extensive experiments on two widely used multi-hop KBQA datasets to prove the effectiveness of our method.

1 Introduction

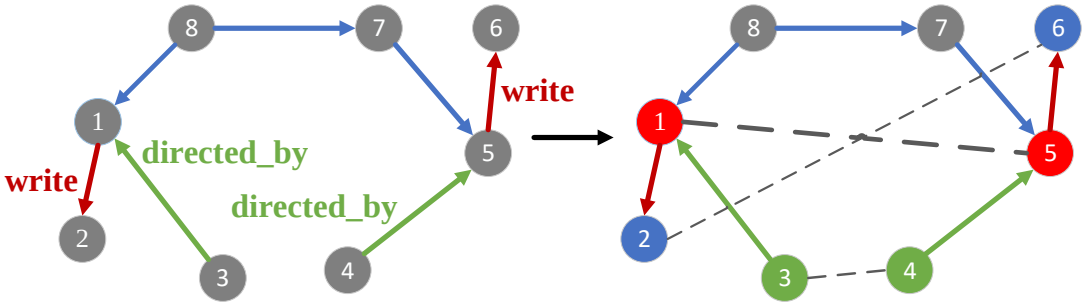
Knowledge bases have become critical resources in a variety of natural language processing applications. A KB such as Freebase (Bollacker et al., 2008) always contains millions of facts which are composed of subject-predicate-object triples, also referred to as a relation between two entities. Such rich structural information has proven effective in KB-based Question Answering (KBQA) tasks , which aim to find the answer entities to a factoid question using facts in the targeting KB (Zhou et al., 2018; Zhang et al., 2018).

Early studies on KBQA are mainly based on neural network models (Dong et al., 2015; Das et al., 2017), which simulate the similarity between the factoid question and the entities in the KB. Although these methods are effective, the structural information in the KB is not fully utilized, which is essential in the reasoning process (Sun et al., 2018). To address this limitation, recent studies (Sun et al., 2019; Xiong et al., 2019) focus on graph neural networks (GNNs), which update nodes by aggregating their neighbor information in graphs. This updated pattern allows GNNs to capture structural information. However, GNN is a special form of Laplacian smoothing (Li et al., 2018), stacking multiple GNN layers may oversmooth features of nodes and reduce the discriminative power of graph embedding. With this insufficiency, conventional GNNs are poor at modeling long-distance node relations, which is essential for GNN reasoning. (Wu et al., 2019).

In this paper, to address the above limitations, we propose a novel framework **GlobalGraph**, which models long-distance node relations from two views: **1)** Modeling node relations by predicting whether two nodes are of the same type label; **2)** Modeling node relations by predicting whether two nodes are all correlated with the question. For **the 1st view**, we assign global type labels for each node according to its neighbor relation information, and then model the long-distance node relations by their global label similarity. Relations contain node label information, and the relation information around the same type nodes should be similar. For example, as shown in Figure 1, there are two triples: (N_3, directed by, N_1) and (N_4, directed by, N_5). Based on the relation "directed by" in these two triples, we regard that

* Corresponding author

Proceedings of the 28th International Conference on Computational Linguistics, pages 2572–2582
Barcelona, Spain (Online), December 8-13, 2020



(a) Original Graph **(b) Modeling Long-distance Node Relations**

Figure 1: **a)** We display the original graph with different types (shown with different colored edges) of relations. Take a specific relation "directed by" as an example, we can infer that the type of N_1 and N_5, which are connected to this relation, is "person". So the two nodes are marked by the same color, indicating that they have the same label. It is the same to N_2 and N_6, and N_3 and N_4. Through this process, We assgin each node a global label through its neighbor relation set. It is worth noting that the larger the neighbor relation set, the more confident the model is to infer the similairty of two nodes' label. **b)** The confidence score is represented by the thickness of the dashed lines.

N_1 is a person and the same as N_5. Type labels of the two nodes are the same, so we connect the two nodes. Based on the new graph, GNN propagates information across long-distance nodes. For **the 2nd view**, only connect the same label nodes is insufficient, because a normal KB contains a huge number of long-distance node pairs with different labels, which are not utilized for GNN reasoning. For a specific node pair, although the node labels are not similar, if the two nodes are related to the question, the information propagation between them is also useful for reasoning. Based on this, we dynamically select nodes related to the current question, and construct a dynamic graph to connect these nodes through full connection. Finally, we implement GNN to perform information propagation and reasoning. By solving the two views, we model global node features through their long-distance nodes, and then combine them with the local node features of conventional GNNs to perform answer prediction.

The main contributions of this paper can be summarized as follows:

- We propose a novel idea to assign type labels to nodes based on their neighbor relation information, and introduce a novel model to enable GNNs to capture long-distance node information from two views: 1) node type similarity; 2) correlation between nodes and questions, which overcomes the shallow node representation in GNNs.
- We conduct extensive experiments on MetaQA (Zhang et al., 2018) and PQL (Zhou et al., 2018), and the results demonstrate the effectiveness of our model.

2 Related Work

2.1 Neural Network-based Question Answering

The KBQA based on the neural network can be divided into two categories: single-hop QA and multi-hop QA. Single-hop models (Bordes et al., 2014; Xu et al., 2016) predict the answer from one fact triple, which can be retrieved by judging the similarity between the question and relations in triples. Although these models have good performance in answer prediction, they are insufficient in multi-hop QA tasks. Because mutli-hop QA task contains complex questions, which requires reasoning across multiple triples to get answers. To perform reasoning, Jiang and Bansal (2019) proposes a self-assembling network to assemble the reasoning modules; Yavuz et al. (2017) considers a continuous checking mechanism to judge the correctness of answer evidence; Zhang et al. (2018) utilizes the variational learning algorithm for multi-hop reasoning; Wang et al. (2019b) explores additional knowledge bases to improve natural language inference; Mitra et al. (2019) translates the question and the KB to a logical representation and

then uses logical reasoning. However, these models lack considering graph structural information, which is important for multi-hop reasoning.

2.2 Graph Neural Networks based Question Answering

Supported with a number of studies on graph representation learning (Kipf and Welling, 2017; Schlichtkrull et al., 2018; Wang et al., 2019a), graph neural network (GNN) shows its powerful ability in graph analysis. A massive number of GNN-based algorithms are designed to perform graph reasoning, such as R-GCN (Schlichtkrull et al., 2018), GRAFT-Net (Sun et al., 2018), HGMAN (Wang et al., 2020) and BAG (Cao et al., 2019), in which nodes update themselves by aggregating the information of neighboring nodes. A node can capture the unconnected node information through multiple GNN layers. Since GNN is a special form of Laplacian smoothing, stack multiple GNN layers may oversmooth features of nodes from different clusters and reduce the discriminative power of graph embedding (Li et al., 2018). Therefore, most GNN models have less than two layers. Due to limited-layer information propagation, conventional GNNs suffer from bad performance in modeling long-distance node relations. Xiao et al. (2019) and Zhuang and Ma (2018) attempt to model long-distance node relations under the guidance of pre-defined node type labels. However, for most datasets of KBQA, pre-defined node type labels are not provided. which makes the above methods not applicable. Different from the previous work, we first assign a global label to each node by modeling its surrounding relation structure, and further gain the long-distance node relations based on the global labels.

3 Model

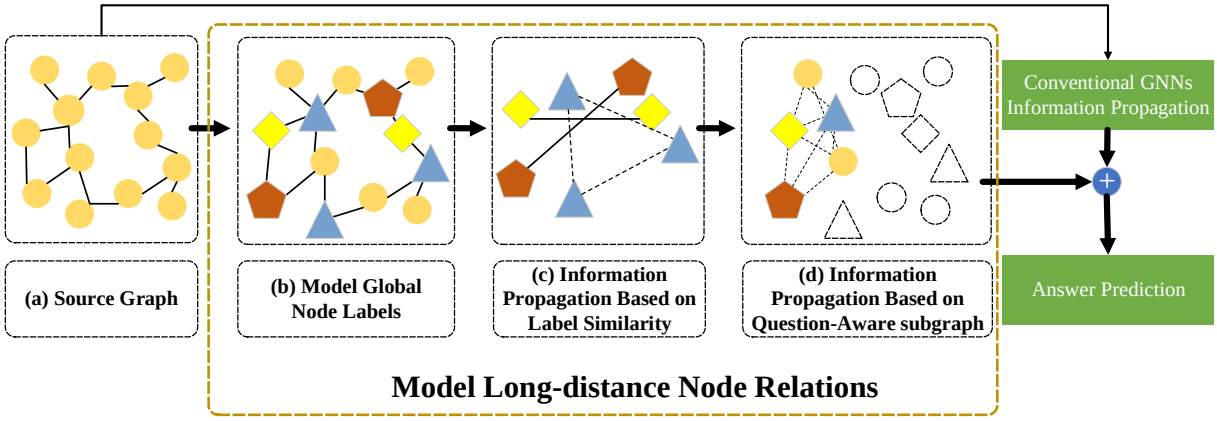


Figure 2: Overview of the model. **a)** The source graph without node labels. **b)** We assign a global label to each node based on the connected relations. **c)** The information propagation based on label similarity. **d)** The information propagation based on question-aware subgraph. Through (b, c, d), the model outputs the long-distance propagation results. We combine it with **Conventional GNNs Information Propagation** results to predict the answers.

3.1 Task Definition

Let $\mathcal{K} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$ denotes a knowledge graph, where $\mathcal{V}$ is the set of entities and $\mathcal{R}$ is the set of relations in KB. $\mathcal{E}$ consists of a set of triples (e_h, r, e_t), which represent the relation $r \in \mathcal{R}$ holds between $e_h \in \mathcal{V}$ and $e_t \in \mathcal{V}$. Given a natural language question $Q = (w_1, w_2, ..., w_{|q|})$, where w_i denotes the ith word, the model needs to extract its answer from $\mathcal{V}$, The overview of our models is shown in Figure 2.

The rest of the **Model Section** is organized as follows: Subsection 3.2 discusses how to encode the factoid question and knowledge graph. Subsection 3.3 describes the information propagation method of conventional GNNs. Subsection 3.4.1 and 3.4.2 discuss how to assign global type labels to each node and propagate information among nodes with similar labels. Subsection 3.4.3 explains the construction of question-aware dynamic graph. Finally, subsection 3.5 discusses the answer prediction.

3.2 Input Encoder

The input encoder initializes the given natural language question and all the candidate entities (in KB) to vector representation.

Question Initialization. We pass word sequence of the question Q to a long short-term memory network (LSTM) (Hochreiter and Schmidhuber, 1997):

$$q = LSTM(Q), \tag{1}$$

where $q \in \mathbb{R}^m$ is the last state of $LSTM$ output. m is the hidden state size. We use q to represent the question.

Node Initialization. Firstly, all of the nodes are represented by pre-trained word vectors or random initialized vectors. For node e_v, it is noted as $w_v \in \mathbb{R}^n$, where n is the embedding size. The seed nodes are nodes that can be connected to the question through entity linking (Sun et al., 2018). The input encoder also embeds the average distance from the node e_v to the seed nodes, as $d_v \in \mathbb{R}^n$. For simplicity, d_v can be represented with the embedding of words "0", "1", "2", etc. With the distance embedding d_v and word embedding w_v, the node e_v is represented as n_v, which is defined as:

$$n_v = [w_v; d_v]W^N, n_v \in \mathbb{R}^n, \tag{2}$$

where [;] is column-wise concatenation and $W^N \in \mathbb{R}^{2n \times n}$ is a learned parameter matrix. By adding distance information, nodes can better update themselves according to the number of hops needed to infer the answers to the current question.

3.3 Conventional GNNs Information Propagation

Similar to previous works (Sun et al., 2018; Xiong et al., 2019), we implement conventional GNNs methods to capture local information. A node catches its local information by aggregating the information of its real neighbors in the source graph.

To enable each node to capture the current question information, we concatenate each node representation n_v with the question q, which is defined as $h_v^0 = [n_v; q]$, and then the node updates itself by aggregating its neighbors' information, which is defined as:

$$u_v^{l+1} = \sigma\left(\sum_{r \in \mathcal{R}} \sum_{j \in N_v^r} \frac{1}{c_{v,r}} W_r^l h_j^l\right), \tag{3}$$

where N_v^r represents the set of neighbor indices of node v based on relation $r \in \mathcal{R}$. $c_{v,r}$ is a normalization constant that can be learned or set directly, such as $c_{v,r} = |N_v^r|$. $W_r^l \in \mathbb{R}^{d_{l+1} \times d_l}$ stands for a learnable parameter matrix. $0 \leq l < L$ and L is the number of layers in the model. h_j^l denotes the hidden state of node e_j at the lth layer.

A gate mechanism decides how much of the update message u_v^{l+1} propagates to the next layer. Gate levels are computed as:

$$a_v^{l+1} = \sigma\left(f_a\left([u_v^{l+1}; h_v^l]\right)\right), \tag{4}$$

where f_a is a linear function. Ultimately, the next layer representation h_v^{l+1} of the node e_v is a gated combination of the previous representation h_v^l and a non-linear transformation of the update information u_v^{l+1}:

$$h_v^{l+1} = \phi(u_v^{l+1}) \odot a_v^{l+1} + h_v^l \odot (1 - a_v^{l+1}), \tag{5}$$

where $\phi(\cdot)$ is any nonlinear function and $\odot$ stands for element-wise multiplication.

The model stacks such networks for L layers. Through L times' convolution operation, the node constantly updates its own state, which simulates the reasoning process. Finally, we get the node representation h_v^L. However, such GNNs can not propagate information between two long-distance nodes due to limited-layer. To overcome this challenge, in the next section, we introduce how to capture the long-distance node relations and propagate information based on them.

3.4 Model Long-distance Node Relations

3.4.1 Model Global Node Type Labels

In this section, we introduce how to build a global label for a node according to its connection relation information, which is based on the relation information implying the connected node type. For example, in the field of movies, for a specific triple (N_1, directed by, N_2) whose relation is "directed by", it can be retrieved that N_2 is a person and N_1 is a movie. It is the same for another triple (N_3, directed by, N_4). From the results, we get that N_2 and N_4 belong to the same type label. Similar to the above process, we first collect the connection relation set of each node e_v, which is defined as:

$$Set_v = (Set_v^{in}, Set_v^{out}), \tag{6}$$

where Set_v^{in} means the set of relations pointing to node e_v and Set_v^{out} represents the set of relations pointing out from node e_v. The reason we need to take into account the relation direction is that, with the above example, although N_1 and N_2 are both connected with relation "directed by", their labels are obviously different. Finally, we regard the Set_v as the global type label of node e_v.

3.4.2 Information Propagation Based on Label Similarity

With the global node type label, we calculate the similarity s_{ij} between two nodes, which is defined as:

$$s_{ij} = sim(Set_i, Set_j) \tag{7}$$

$$sim(Set_i, Set_j) = \frac{f(Set_i^{in}, Set_j^{in}) + f(Set_i^{out}, Set_j^{out})}{2} \tag{8}$$

$$f(Set_i^{in}, Set_j^{in}) = \frac{len(Set_i^{in} \cap Set_j^{in})}{min(len(Set_i^{in}), len(Set_j^{in}))} \tag{9}$$

where $* \cap *$ represents the intersection of two sets. $len(*)$ means the number of elements in the set. Finally, we get the node similarity matrix $S \in \mathbb{R}^{|V| \times |V|}$, where $|V|$ means the number of nodes.

Based on the node similarity matrix S, similar to Equation 3, we use graph convolutional network (GCN) to perform information propagation, which is defined as:

$$g_v^{l+1} = \sigma\left(\sum_{j=1}^{|V|} s_{vj} W^l t_j^l\right), \tag{10}$$

where t_j^l denotes the hidden state of node e_j at the lth layer and $t_j^0 = n_j$ (Equation 2). The update message g_v^{l+1} pass through the gate mechanism (similar to Equation 4,5) to get the current layer representation t_v^l. The model stacks such networks for K layers. Finally, we get the last layer representation t_v^K of the node e_v.

3.4.3 Information Propagation Based on Dynamic Question-aware Subgraph

In the above section, we consider that node pairs with higher label similarity have relations. However, although low label similarity, some node pairs are related to the current question. The information propagation between them can play a positive role in predicting answers. In this section, we first select the nodes related to the factoid question, and then link these nodes by full connection to construct a dynamic question-aware graph. With the dynamic graph, the model performs information propagation to capture question-related information.

We first get the representation of node e_v, which is defined as:

$$m_v^0 = t_v^K W, \tag{11}$$

where W stands for a learnable parameter matrix. The similarity between node e_v and question Q is calculated as:

$$sq_v^l = \sigma(m_v^l W q^l), \tag{12}$$

where $q^0 = q$ and it is updated by summing the seed nodes' vectors of the $(l-1)$th layer. $sq_v^l \in [0,1]$ represents the similarity confidence. We select the nodes whose sq_v^l is greater than the threshold t_q and then construct the node set. Then we connect the nodes in the collected set by full connection to construct the question-aware dynamic graph. In the dynamic graph, the edge weight between node e_i and node e_j is the average of sq_i^l and sq_j^l.

Similar to Equation 10, we perform GCN on the dynamic graph and stack such graphs for J layers. Finally, we get the last layer representation m_v^J of the node e_v.

3.5 Answer Prediction

We concatenate the entity representation of local propagation results h_v^L and global propagation results m_v^J and pass through a linear layer f_{out} to predict the answer distribution, which is defined as:

$$p_v = \sigma(f_{out}([h_v^L; m_v^J])), \tag{13}$$

where σ is the sigmod function. f_{out} converts the dimension to 1.

3.6 Loss

The training loss is binary cross-entropy loss of the final answers prediction, which is defined as:

$$L(\theta) = -\sum_{i=1}^{n}[y_i log(p_i) + (1-y_i) log(1-p_i)], \tag{14}$$

where θ represents the model parameters, y is the golden distribution over entities, and n is the number of nodes.

4 Experiments

4.1 Datasets

MetaQA (Zhang et al., 2018) is composed of three sets of question-answer pairs in natural language form (1-Hop, 2-Hop, and 3-Hop) and a movie domain knowledge base. It contains three versions of questions (Vanilla, NTM, and Audio). In our experiments, we use the "Vanilla" version and do performance analysis in three sets of different hops.

PQL (PathQuestion-Large) (Zhou et al., 2018) is a multi-hop KBQA dataset. The dataset consists of 2-Hop (PQL-2H) questions and 3-Hop (PQL-3H) questions.

Entity linking is performed on these two datasets. We follow Xiong et al. (2019) and utilize the simple surface-level matching to make fair comparisons. The statistics of the two datasets are shown in Table 1.

4.2 Baselines

We compare our proposed model with the following models:

(1) **Key-Value Memory Network** (KVMem) (Miller et al., 2016), an end-to-end memory network that can be used for KBQA. (2) **IRN** (Zhou et al., 2018), an interpretable reasoning model for knowledge graph question answering. (3) **VRN** (Zhang et al., 2018), an end-to-end variational learning algorithm, which not only addresses the noise in questions but also performs effective multi-hop reasoning. (4) **GraftNet** (Sun et al., 2018), a model which treats documents as a special genre of nodes in KB and utilizes graph convolution network to aggregate the information. (5) **SGReader** (Xiong et al., 2019), a model that aims to solve the incomplete knowledge graph by utilizing text information, applying a graph-attention to aggregate the information of each entity from its linked neighbors.

4.3 Training Details

We run the experiments on a P40 GPU with 24G memory. Throughout the experiments, for all of the baselines and the proposed model, we apply the 300-dimension TransE embeddings (Bordes et al., 2013) to initialize entity states and 300-dimension GloVE embeddings (Pennington et al., 2014) to initialize word states in questions. The hidden dimension of the LSTM is 300. The hidden dimension of all GCN

Datasets		Train	Dev	Test	Entity	Relation
MetaQA	1-Hop	96106	9992	9947	40128	9
	2-Hop	118980	14872	14872		
	3-Hop	114196	14274	14274		
PQL	2-Hop	1434	160	160	5035	364
	3-Hop	925	103	103		

Table 1: The statistics of the MetaQA and the PQL. We show the size of Train, Dev, and Test set of the two datasets, as well as the total number of entities and relations.

Model	MetaQA 1-Hop		MetaQA 2-Hop		MetaQA 3-Hop	
	Hits@1	F1	Hits@1	F1	Hits@1	F1
KVMem	0.958	-	0.760	-	0.489	-
VRN	0.978	-	0.898	-	0.630	-
SGReader	0.967	0.960	0.807	0.798	0.610	0.580
GraftNet	0.974	0.910	0.948	0.727	0.778	0.561
GlobalGraph	**0.990**	**0.976**	**0.955**	**0.830**	**0.814**	**0.624**

Table 2: Experimental results on the MetaQA datasets.

layers is 300. The layer number is 1 for all GCNs in our model, and the dropout after each GCN layer is set to 0.1. The Adam optimizer (Kingma and Ba, 2015) is used with the initial learning rate of 0.001.

To make fair comparisons, We follow Sun et al. (2018; Xiong et al. (2019) and apply the Personalized PageRank algorithm (Haveliwala, 2002) on the MetaQA dataset to pick the top N entities to get a smaller subgraph. After PageRank algorithm, for each subgraph on 1-hop, 2-hop and 3-hop datasets, there are an average of 6, 35, and 495 entities respectively

4.4 Main Results and Discussion

Table 2 depicts the comparisons with state-of-the-art models on the MetaQA dataset. As shown in Table 2, our model achieves the best Hits@1 and F1. Specifically, on the MetaQA 1-Hop, our model improves Hits@1 and F1 by 1.2% and 1.6% respectively, and on the MetaQA 2-Hop dataset, our model is 0.7% and 3.2% higher than the second best one on Hits@1 and F1 respectively. Similarly, our model has achieved the best performance on MetsQA 3-Hop.

We show the experimental results on the PQL dataset in Table 3. PQL dataset has the feature that each question has only one answer, so we only adopt Hits@1 for evaluation. On the Hits@1 metric, we observe that our model achieves the best results, improving 3.5% and 2.8% on 2-Hop and 3-Hop, respectively.

The reasons why our method performs well include: 1) Our method considers using graph neural network (GNN) to model the structural information of knowledge graph, which aims to enhance the reasoning ability; 2) Our idea can catch the long-distance node similarity by modeling the labels of each node, which is not considered in previous GNN-based KBQA models; 3) Our model captures more question-related information by constructing the question-aware dynamic graph.

4.5 Ablation Experiment

We compare our model with a few variants. R-GCN (Schlichtkrull et al., 2018) considers the influence of different types of connected relations when aggregating neighbors' information. GAT (Velickovic et al., 2018) implements the weight-based neighbor aggregation method. In the experiment, we combine R-GCN and GAT, and name it as R-GAT. R-GAT and R-GCN fail to consider the long-distance node relations, and only perform information propagation based on the real neighbors of nodes. As shown in Table 4, we can find that our model has achieved the best performance, and the biggest difference

Model	PQL 2-Hop	PQL 3-Hop
	Hits@1	Hits@1
KVMem	0.622	0.674
IRN	0.725	0.710
SGReader	0.719	0.893
GraftNet	0.707	0.913
GlobalGraph	**0.760**	**0.941**

Table 3: Experimental results on the PQL datasets.

Model	PQL 3-Hop	MetaQA 2-Hop
	Hits@1	Hits@1
R-GCN	0.859	0.920
R-GAT	0.890	0.941
GlobalGraph	0.941	0.955
- q-aware subgraph	0.932	0.950
- label similarity	0.920	0.943

Table 4: Ablation experiments of our model on PQL and MetaQA dataset.

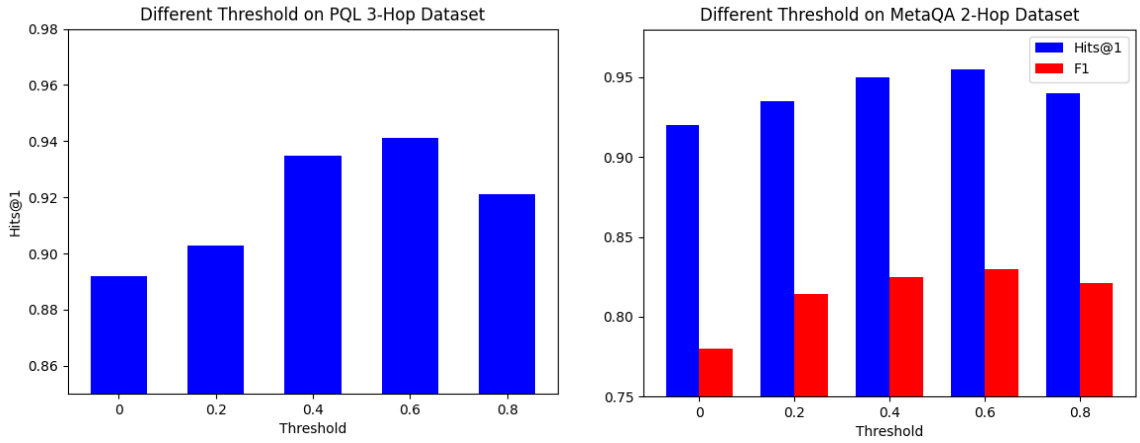


Figure 3: Performance using varying thresholds t_q on different datasets. The number of thresholds is ranging from 0 to 0.8.

between our model and these two models lie in considering long-distance node relations, which proves the effectiveness of our proposed model.

we conduct experiments to evaluate the performance of different components in our model. **GlobalGraph w/o q-aware subgraph** does not consider constructing the question-aware subgraph. **GlobalGraph w/o label similarity** does not consider propagating information between two nodes with the same label, which only performs local and question-aware information propagation. As shown in Table 4, without these components, the performance of the model has declined, which proves the effectiveness of these two components in our model.

4.6 Analysis of Question-Aware Graph

In the proposed model, we construct a question-aware dynamic graph to enhance the relevance between nodes and the given question. In this section, we analyze the effectiveness of this method by showing the model performance of different threshold values t_q. As shown in Figure 3, if the threshold is set too low (threshold=0), we can find that the model performance reduces, probably because there are too many question-irrelevant nodes in the graph. The information propagation between these nodes will reduce the reasoning performance. With the increase of threshold (from 0 to 0.8), the performance of the model is increasing, which proves the validity of the question-aware subgraph. If the threshold is too large (threshold=0.8), the performance of the model is also reduced because too many nodes are discarded, resulting in the information loss.

4.7 Case Study of Modeling Long-distance Node Similarity

In order to prove the validity of modeling long-distance node similarity based on the global labels, we give examples from PQL 3-Hop. As shown in Figure 4 (b), it contains the adjacency matrix of the real graph and the similarity matrix of node labels.

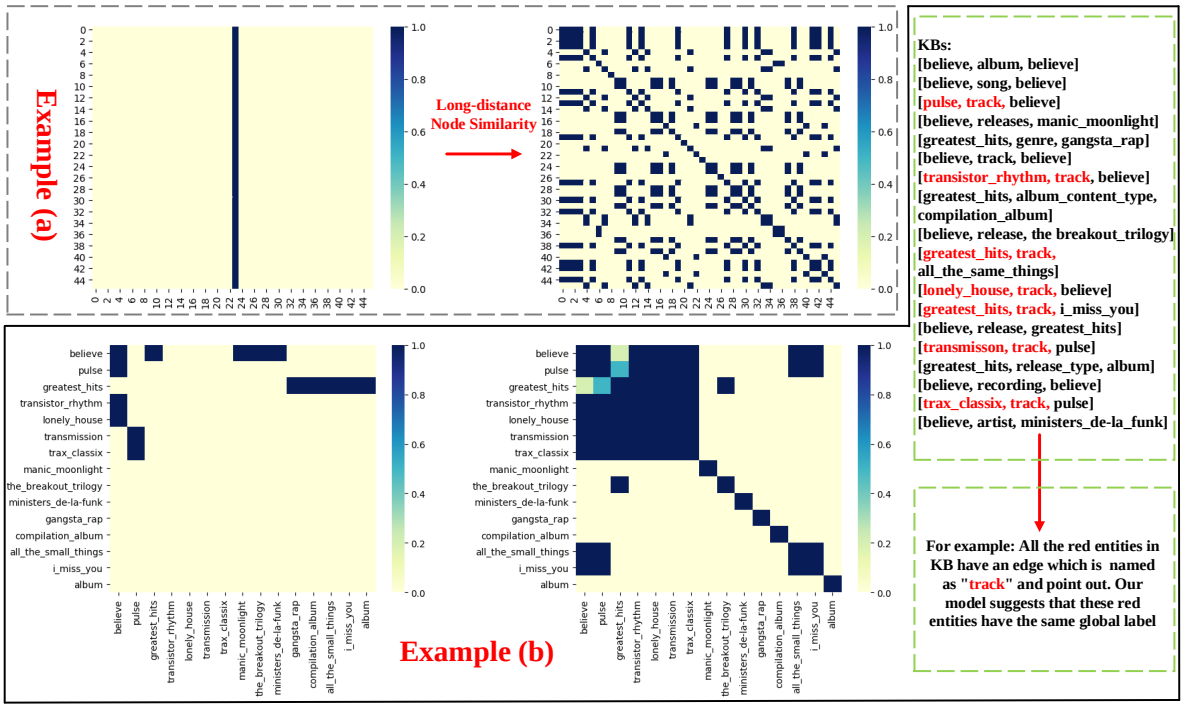


Figure 4: Example (a) and Example (b) display two KB examples respectively, in which the left heatmap is the original adjacency matrix, and the right heatmap is the constructed long-distance node relations. Deep color means a strong correlation between two nodes. From the comparison of two heatmaps in an example, we find that the constructed relation matrix can capture the long-distance node relations.

From the adjacency matrix of the real graph, we find that node "pulse" and node "i_miss_you" are not connected. However, the labels of them are the same, which can be obtained by their surrounding relations in the given KB. This relation is captured correctly by the similarity matrix, which proves the validity of our methods. Figure 4 (a) is a specific knowledge graph, because all nodes are connected to only one node. In this case, the relations between other nodes can not be captured. By using the constructed similarity matrix, we can capture more abundant relation information.

5 Conclusion

In this paper, we propose a novel KBQA model based on graph neural network, which can capture long-distance node relations by modeling the relation features of each node and further judge the feature similarity. Moreover, our model constructs a dynamic question-aware subgraph, retains the nodes related to the question, and propagates messages on these nodes to improve the reasoning ability. Experiments based on two open datasets demonstrate our model's ability on performing answer prediction. Ablation experiments prove the validity of each part of the model. Case study demonstrates our model's ability to capture long-distance node relations. In the future,we will explore other ways to capture the relation between distant nodes and improve the current proposed model.

Acknowledgements

This work is supported by Beijing Natural Science Foundation (Grant No. 4182042), Beijing Nova Program of Science and Technology (Grant No. Z191100001119031), National Key Research and Development Program of China (Grant No. 2018YFB1003804), and The Open Program of Zhejiang Lab (Grant No. 2019KE0AB03). Xu Wang is supported by BUPT Excellent Ph.D. Students Foundation under grant CX2019137. We thank Dr. Guoshun Nan for valuable discussions as well as insightful suggestions for improving our model. We also thank the anonymous reviewers who have given significant and constructive comments.

References

Kurt D. Bollacker, Colin Evans, Praveen 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, Vancouver, BC, Canada, June 10-12, 2008*. ACM.

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. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*.

Antoine Bordes, Jason Weston, and Nicolas Usunier. 2014. Open question answering with weakly supervised embedding models. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part I*. Springer.

Yu Cao, Meng Fang, and Dacheng Tao. 2019. BAG: bi-directional attention entity graph convolutional network for multi-hop reasoning question answering. In *NAACL*.

Rajarshi Das, Manzil Zaheer, Siva Reddy, and Andrew McCallum. 2017. Question answering on knowledge bases and text using universal schema and memory networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*. Association for Computational Linguistics.

Li Dong, Furu Wei, Ming Zhou, and Ke Xu. 2015. Question answering over freebase with multi-column convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*. The Association for Computer Linguistics.

Taher H. Haveliwala. 2002. Topic-sensitive pagerank. In *Proceedings of the Eleventh International World Wide Web Conference, WWW 2002, May 7-11, 2002, Honolulu, Hawaii, USA*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*.

Yichen Jiang and Mohit Bansal. 2019. Self-assembling modular networks for interpretable multi-hop reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *ICLR*.

Qimai Li, Zhichao Han, and Xiao-Ming Wu. 2018. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*.

Alexander H. Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-value memory networks for directly reading documents. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*. The Association for Computational Linguistics.

Arindam Mitra, Peter Clark, Oyvind Tafjord, and Chitta Baral. 2019. Declarative question answering over knowledge bases containing natural language text with answer set programming. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*. ACL.

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 *ESWC*.

Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William W. Cohen. 2018. Open domain question answering using early fusion of knowledge bases and text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*. Association for Computational Linguistics.

Haitian Sun, Tania Bedrax-Weiss, and William W. Cohen. 2019. Pullnet: Open domain question answering with iterative retrieval on knowledge bases and text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2380–2390. Association for Computational Linguistics.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S. Yu. 2019a. Heterogeneous graph attention network. In *WWW*.

Xiaoyan Wang, Pavan Kapanipathi, Ryan Musa, Mo Yu, Kartik Talamadupula, Ibrahim Abdelaziz, Maria Chang, Achille Fokoue, Bassem Makni, Nicholas Mattei, and Michael Witbrock. 2019b. Improving natural language inference using external knowledge in the science questions domain. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press.

Xu Wang, Shuai Zhao, Bo Cheng, Jiale Han, Yingting Li, Hao Yang, and Guoshun Nan. 2020. HGMAN: multi-hop and multi-answer question answering based on heterogeneous knowledge graph (student abstract). AAAI Press.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. 2019. A comprehensive survey on graph neural networks. *CoRR*.

Yuxin Xiao, Zecheng Zhang, Carl Yang, and Chengxiang Zhai. 2019. Non-local attention learning on large heterogeneous information networks. In *2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, December 9-12, 2019*. IEEE.

Wenhan Xiong, Mo Yu, Shiyu Chang, Xiaoxiao Guo, and William Yang Wang. 2019. Improving question answering over incomplete kbs with knowledge-aware reader. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*. Association for Computational Linguistics.

Kun Xu, Siva Reddy, Yansong Feng, Songfang Huang, and Dongyan Zhao. 2016. Question answering on freebase via relation extraction and textual evidence. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Semih Yavuz, Izzeddin Gur, Yu Su, and Xifeng Yan. 2017. Recovering question answering errors via query revision. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*. Association for Computational Linguistics.

Yuyu Zhang, Hanjun Dai, Zornitsa Kozareva, Alexander J. Smola, and Le Song. 2018. Variational reasoning for question answering with knowledge graph. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 6069–6076.

Mantong Zhou, Minlie Huang, and Xiaoyan Zhu. 2018. An interpretable reasoning network for multi-relation question answering. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*. Association for Computational Linguistics.

Chenyi Zhuang and Qiang Ma. 2018. Dual graph convolutional networks for graph-based semi-supervised classification. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*. ACM.