HGMAN: Multi-Hop and Multi-Answer Question Answering Based on Heterogeneous Knowledge Graph (Student Abstract)

Xu Wang,[1] Shuai Zhao,[1*] Bo Cheng,[1] Jiale Han,[1] Yingting Li,[1]
Hao Yang,[2] Guoshun Nan[3]
[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
[3]Singapore University of Technology and Design
{wxx, zhaoshuaiby, chengbo, hanjl, cindyting}@bupt.edu.cn,
yanghao30@huawei.com, nanguoshun@gmail.com

Abstract

Multi-hop question answering models based on knowledge graph have been extensively studied. Most existing models predict a single answer with the highest probability by ranking candidate answers. However, they are stuck in predicting all the right answers caused by the ranking method. In this paper, we propose a novel model that converts the ranking of candidate answers into individual predictions for each candidate, named heterogeneous knowledge graph based multi-hop and multi-answer model (HGMAN). HGMAN is capable of capturing more informative representations for relations assisted by our heterogeneous graph, which consists of multiple entity nodes and relation nodes. We rely on graph convolutional network for multi-hop reasoning and then binary classification for each node to get multiple answers. Experimental results on MetaQA dataset show the performance of our proposed model over all baselines.

Introduction

The knowledge base is composed of entities as nodes and relations as different types of edges. Knowledge base question answering (KBQA) aims to figure out the answer for a question from these entities.

However, Most end-to-end neural models only obtain a single answer when predicting. If there are multiple answers to a question, these models fail to figure out all the answers correctly. For multi-hop reasoning, previous models mostly consider how to propagate information between entities, and relations information are more used as the weight of entities information in the propagation, but it loses a lot of surface semantic information of relations.

To solve the above issues, in this paper, we propose a heterogeneous knowledge graph based multi-hop and multi-answer model (HGMAN). The question answering model is exempt from the limitation of predicting a single answer. To fuse the surface semantic information of the relations, we construct a heterogeneous knowledge graph by changing the relation edges to the relation nodes, which transforms information propagation between entities into information propagation between entities and relations. HGMAN transfers information between nodes in the graph and updates the current node's representation based on neighbor nodes and relations, which simulates the reasoning process. Let each node in the graph learn question-based node representation. Finally, HGMAN uses binary classification for each node to get multi-answer.

[*]Corresponding author

Model

Task Definition A knowledge graph is denoted as $\mathcal{K} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$, where $\mathcal{V}$ is the set of entities in KB, and $\mathcal{E}$ is the set of triples (e_1, r, e_2), where $e_1, e_2 \in \mathcal{V}$ are entities and $r \in \mathcal{R}$ is the relation between e_1 and e_2. Given a question $q = (w_1, ..., w_{|q|})$, where $|q|$ is the length of question, the model needs to extract its answers from entities set $\mathcal{V}$. We follow Sun et al. (2018) and only consider a subgraph for each question. The subgraph $\mathcal{K}_s = (\mathcal{V}_s, \mathcal{E}_s, \mathcal{R}_s)$ is retrieved by runing Personalized PageRank (PPR) (Haveliwala 2002).

Constructing Heterogeneous Knowledge Graph

Constructing the Reverse Relations. The relation $r \in \mathcal{R}_s$ in KB is directional, from e_1 to e_2. We expand the number of $\mathcal{R}_s$ by taking into account the reverse relation r' and the self-connection relation r_{self} as the node is connected to itself.
Adding Relations to Nodes. With subgraph $\mathcal{K}_s$, in the process of reasoning the answers, relations are added to the subgraph as nodes, defined by $e_{r_1}, e_{r_2}, ..., e_{r_n}$, where n is the number of relations in $\mathcal{K}_s$. The process is shown in Figure 1. We update the triples in the following way:

$$(e_1, r, e_2) \Rightarrow (e_1, r\_1, e_r) \cup (e_r, r\_2, e_2) \tag{1}$$

Where r_1 represent that the string r is connected to the string "_1" and the same as r_2.

Multiple Answers Prediction

Question-aware Node Initialization. For all nodes in the graph, we first use pre-trained word vectors or random initialization to represent them, as $w_v \in R^n$, where n is the embedding size. We also embed the question using an LSTM, and use q to represent the question, where q is the output

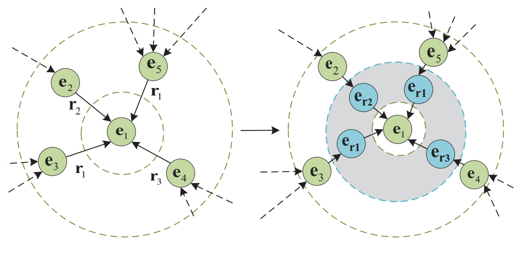

Figure 1: Explanation of transforming relations to nodes.

of the last element of the LSTM. Then we concatenate each node with the question, defined by $h_v^0 = [w_v; q]$.

Node Updates. We follow the R-GCN (Schlichtkrull et al. 2018) to update each node. For each node in the graph, it is updated by

$$h_v^{l+1} = \sigma\left(\sum_{r\in R^+}\sum_{j\in N_i^r}\frac{1}{c_{i,r}}W_r^{(l)}h_v^l\right) \tag{2}$$

where $0 \le l < L$ is the layer of graph convolution. $W_r^{(l)} \in R^{d_{l+1}\times d_l}$. $\sigma(\cdot)$ is the sigmoid function. N_i^r represents the set of neighbor indices of node v based on relation r. $c_{i,r}$ is set directly to a constant, such as $c_{i,r} = |N_i^r|$.

With the final representations h_v^L and question representation q, we convert each node into the input needed for the final training and prediction of the model, defined by

$$h_v^p = [h_v^L; q]W_p \in R^2 \tag{3}$$

where W_p is a learned parameter matrix. For vector $h_v^p \in R^2$, it is defined that if the value of the second dimension is higher than the first dimension, the node v is an answer. On the contrary, the node v is not an answer. Binary cross-entropy loss is used to train the model.

Model Predicting. At the predicting step, we mask the relation nodes and only predict on the entity nodes. Previous models mostly predict only one answer. In order to compare with previous models and demonstrate the performance of multi-answer prediction, we use the following methods:

Full: The model predicts all nodes individually. For a node, if the second dimension value is higher than the first dimension value, then the node is regarded as an answer.

Hits@1: First, The model uses **Full**'s method to get the predicted answer list, and then it selects the node from the list with the largest second dimension value as the final answer. If the predicted answer list is empty, then the model selects the node with the largest second dimension value as the final answer in the set of all entity nodes.

Experiments

Dataset. MetaQA (Zhang et al. 2018) is a multi-hop and multi-answer dataset for knowledge graph based question answering (KGQA). This dataset has 100,000 training data for each subset and a movie domain knowledge base. For entity linking, we use simple surface level matching.

Baselines. We compare with **KVMemNet**, **IRN** (Zhou, Huang, and Zhu 2018), **VRN** (Zhang et al. 2018), **GraftNet** (Sun et al. 2018) and **SGReader** (Xiong et al. 2019). For all the baselines, we expand them to multi-answer predictions. During testing, we pick the top k entities as candidates and k is the length of gold answers.

Training Details. We apply 768 dimensional BERT and TransE embedding for questions and knowledge nodes respectively. The layer number of GCN is 2 and the GCN hidden dimension is 400.

Model	1-Hop		2-Hop		3-Hop	
	Hits@1	Full	Hits@1	Full	Hits@1	Full
KVMem	0.958	-	0.251	-	0.101	-
VRN	0.975	-	0.898	-	0.625	-
SGReader	0.967	0.903	0.807	0.719	0.610	0.272
GraftNet	0.970	0.918	0.948	0.681	0.777	0.226
HGMAN	**0.991**	**0.976**	**0.970**	**0.862**	**0.856**	**0.275**
No Rel-nodes	0.971	0.950	0.937	0.826	0.810	0.271

Table 1: Experimental results on MetaQA datasets.

Main Results and Discussion

From Table 1, we can observe that: (1) On Hits@1 metric, unlike baselines, we use heterogeneous graphs to enhance the informativeness of relations, and use graph convolution to reasoning the answers. The validity of our model is proved by the results. (2) On Full metric, the performance of our model in multi-answer prediction is proved. The results validate the method of heterogeneous graph and node classification after graph convolution in multi-answer prediction. (3) We find that after removing the relation nodes (No Rel-nodes), the performance of our model decreases on Hits@1 and Full, which shows that it is important for the model to consider the surface semantic information of the relations.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (No. 2018YFB1003804)

References

Haveliwala, T. H. 2002. Topic-sensitive pagerank. In *WWW*.

Schlichtkrull, M. S.; Kipf, T. N.; Bloem, P.; van den Berg, R.; Titov, I.; and Welling, M. 2018. Modeling relational data with graph convolutional networks. In *ESWC*.

Sun, H.; Dhingra, B.; Zaheer, M.; Mazaitis, K.; Salakhutdinov, R.; and Cohen, W. W. 2018. Open domain question answering using early fusion of knowledge bases and text. In *EMNLP*.

Xiong, W.; Yu, M.; Chang, S.; Guo, X.; and Wang, W. Y. 2019. Improving question answering over incomplete kbs with knowledge-aware reader. In *ACL*.

Zhang, Y.; Dai, H.; Kozareva, Z.; Smola, A. J.; and Song, L. 2018. Variational reasoning for question answering with knowledge graph. In *AAAI*.

Zhou, M.; Huang, M.; and Zhu, X. 2018. An interpretable reasoning network for multi-relation question answering. In *COLING*.