Hypergraph Convolutional Network for Multi-Hop Knowledge Base Question Answering (Student Abstract)

Jiale Han, Bo Cheng, Xu Wang
State Key Laboratory of networking and switching technology,
Beijing University of Posts and Telecommunications
{hanjl, chengbo, wxx}@bupt.edu.cn

Abstract

Graph convolutional networks (GCN) have been applied in knowledge base question answering (KBQA) task. However, the pairwise connection between nodes of GCN limits the representation capability of high-order data correlation. Furthermore, most previous work does not fully utilize the semantic relation information, which is vital to reasoning. In this paper, we propose a novel multi-hop KBQA model based on hypergraph convolutional network. By constructing a hypergraph, the form of pairwise connection between nodes and nodes is converted to the high-level connection between nodes and edges, which effectively encodes complex related data. To better exploit the semantic information of relations, we apply co-attention method to learn similarity between relation and query, and assign weights to different relations. Experimental results demonstrate the effectivity of the model.

Introduction

The task of knowledge base question answering (KBQA) requires answering natural language question by reasoning on a knowledge graph and selecting candidate entities as answers. There are two variants of this task: single-hop and multi-hop. In this paper, we focus on multi-hop KBQA.

Recent years witness a growing development in graph convolutional network (GCN), some existing studies have applied GCN to KBQA system (Sun et al. 2018; Xiong et al. 2019). Despite the effectiveness of previous work, they still have several drawbacks. GCN assumes the connection between entities is in pairwise formulation, and the graph-based representation is insufficient for complex related data. Moreover, many existing models employ dot product operation to calculate the similarity between relation and query to fuse the relation into reasoning, which restricts the full utilization of the semantic relation information.

There has been some leading work (Feng et al. 2019) on hypergraph convolutional network (HGCN) for complex related data representation learning, which converts the general graph representation to a hypergraph representation. Hypergraph defines hyperedge that can connect more than two nodes, and focuses on the connection between nodes and edges indicated by the incidence matrix.

In this work, we propose the Hypergraph Convolutional Network for Multi-hop Knowledge Base Question Answering (HGCQA). We construct a hypergraph for each query based on the KB and regard entities as nodes and relations as hyperedges, which effectively exploits the representation of high-order data correlation. In order to make better use of the semantic relation information, we apply co-attention to learn the correlation between query and relations, and determine the importance of each relation in reasoning to assign different weights. HGCQA transforms the nodes' feature to the connected hyperedges, and then aggregates information from the hyperedges to update the status of nodes, finally predicts whether to be the answer on each node.

We experiment on PQL dataset (Zhou, Huang, and Zhu 2018) which is a multi-hop KBQA dataset. The experimental results illustrate the effectivity of the model.

Model

Task Definition. Given a question $q = q_1, q_2, ...q_{l_q}$, where l_q denotes the number of words in the question, and a knowledge base $\mathcal{K} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$, $\mathcal{E}$ consists of triples (e_s, r, e_o), which denotes a relation $r \in \mathcal{R}$ between subject $e_s \in \mathcal{V}$ and object $e_o \in \mathcal{V}$. The task is to pick answers from $\mathcal{V}$.
The overall framework of the model is shown in Figure 1.
Sub Hypergraph Retrieval. We obtain entities as candidate answers $c = c_1, c_2, ...c_{l_c}$ and relations $r = r_1, r_2, ...r_{l_r}$, where l_c and l_r represent the number of candidates and relations. Hypergraph is constructed based on KB triples by constructing the incidence matrix $\mathbf{H} \in \mathbb{R}^{l_c \times l_r}$. If the i-th candidate is connected with the j-th relation, H_{ij} is set to 1.
Query-aware Relation Weight. Let $\mathbf{X}_q \in \mathbb{R}^{l_q \times d}$, $\mathbf{X}_r \in \mathbb{R}^{l_r \times d}$ be the embedding matrices of question and relations, where d denotes the embedding dimension. First we employ bi-LSTM encoder to encode question and get question hidden states $\mathbf{H}_q \in \mathbb{R}^{l_q \times h}$. h denotes the hidden dimension of encoder, here we assume $h = d$. Then we apply co-attention to learn query-aware relation representations:

$$\mathbf{A}_{rq} = \mathbf{X}_r(\mathbf{H}_q^{\mathrm{T}}) \in \mathbb{R}^{l_r \times l_q} \tag{1}$$

$$\mathbf{C}_r = softmax(\mathbf{A}_{rq})\mathbf{H}_q \in \mathbb{R}^{l_r \times h} \tag{2}$$

$$\mathbf{C}_q = softmax(\mathbf{A}_{rq}^{\mathrm{T}})\mathbf{X}_r \in \mathbb{R}^{l_q \times d} \tag{3}$$

$$\mathbf{D}_r = lstm(softmax(\mathbf{A}_{rq})\mathbf{C}_q) \in \mathbb{R}^{l_r \times h} \tag{4}$$

$$\mathbf{R}_{co\_attn} = [\mathbf{C}_r; \mathbf{D}_r] \in \mathbb{R}^{l_r \times 2h} \tag{5}$$

where T represents matrix transposition, $softmax()$ is column-wise normalization, and $[;]$ is column-wise concatenation. Then we calculate weights for each relation as follows, where $Max()$ represents the maximum value by row.

$$\mathbf{a}_r = softmax(Max(\mathbf{R}_{co\_att})) \in \mathbb{R}^{l_r} \tag{6}$$

Query-aware Candidate Node Initialization. Let $\mathbf{X}_c \in \mathbb{R}^{l_c \times d}$ be the embedding matrix of candidates. We apply co-attention to get query-aware candidate representation $\mathbf{C}_{co\_attn} \in \mathbb{R}^{l_c \times 2h}$, then concatenate to $\mathbf{X}_c$ and pass through a linear network f_{in} to obtain the node initial states.

$$\mathbf{C}^{(0)} = f_{in}([\mathbf{C}_{co\_attn}; \mathbf{X}_c]) \in \mathbb{R}^{l_c \times h} \tag{7}$$

Reasoning over Hypergraph. We employ n-layer HGCN. In each HGCN layer, the features of nodes are transformed to the connected hyperedges by multiplying $\mathbf{H}^{\mathrm{T}}$, and then aggregated from the hyperedges by multiplying $\mathbf{H}$ to achieve better representation of high-order data correlation.

$$\begin{aligned} \mathbf{C}^{(l+1)} &= HGCN(\mathbf{H}, \mathbf{W}, \mathbf{C}^{(l)}) \\ &= \mathbf{D}_v^{-\frac{1}{2}}\mathbf{H}\mathbf{W}\mathbf{D}_e^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{D}_v^{-\frac{1}{2}}\mathbf{C}^{(l)}\mathbf{P} \end{aligned} \tag{8}$$

where $\mathbf{W}$ is a diagonal matrix $\mathbf{W} = diag(\mathbf{a}_r)$, represented the weights for each relation. $\mathbf{C}^{(l)}$ is the signal of hypergraph at l layer. $\mathbf{D}_v$ and $\mathbf{D}_e$ are the diagonal matrices of node and edge degree. $\mathbf{P}$ is the learnable parameter. Multi-head and skip connection are used to stabilize the reasoning process and improve the representation of networks. Finally, we concatenate the output of n-th layer and $\mathbf{C}_{co\_attn}$ and pass through a liner layer f_{out} to obtain final output.

$$output = f_{out}([\mathbf{C}_{co\_attn}; \mathbf{C}^{(n)}]) \in \mathbb{R}^{l_c \times 2} \tag{9}$$

Answer Selection. The final output is two-dimensional, where the first dimension represents the possibility of not being the answer, and the second represents to be. We output the candidate with the largest value of the second dimension.

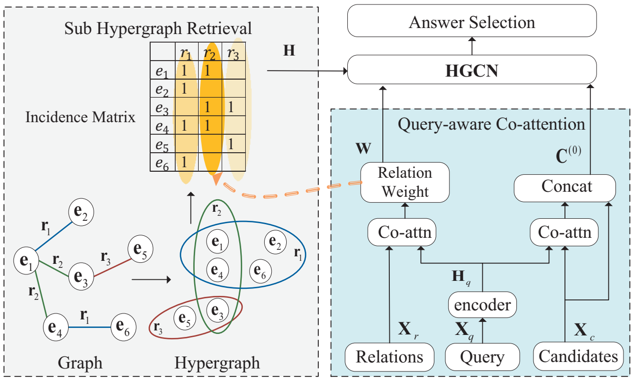
Experiment

Dataset. We use the PQL dataset which includes 1594 two-hop questions (PQL-2H), 1031 three-hop questions (PQL-3H), and a KB with 5035 entities and 364 relations.

Baselines. We compare with the following models: MemNet, KVMemNet, IRN (Zhou, Huang, and Zhu 2018), GraftNet (Sun et al. 2018), and SGReader (Xiong et al. 2019).

Training Details. We apply 768-dimensional TransE embeddings for entities and relations, and BERT embeddings for question words. The hidden size of LSTM and HGCN is 384 and 768. The HGCN layer is set to 3. The Adam optimizer is applied to minimize the binary cross-entropy loss with a learning rate of 0.001.

Results and Discussion. Accuracy metric is adopted to measure the performance. As shown in Table 1, HGCQA outperforms the baselines. The accuracy is 1.7% and 0.3% higher than the second best on the PQL-2H and the PQL-3H respectively. In particular, we compare HGCQA with Graftnet and SGReader, which employ GCN in their models. The results show that the accuracy of HGCQA is higher than theirs, and to some extent, prove the superiority of HGCN to represent high-order relation among data. To illustrate the validity of relation weight, we do the ablation experiment that assigns the same weight to relations. The results state the accuracy rises 1.3% and 2.0% respectively, proving that adding relation weight improves the performance.



	r_1	r_2	r_3
e_1	1	1	
e_2	1		
e_3		1	1
e_4	1	1	
e_5			1
e_6	1		

Figure 1: The overall framework of the model.

Model	PQL-2H	PQL-3H
MemNet	0.690	0.617
KVMemNet	0.722	0.674
IRN	0.725	0.710
GraftNet	0.707	0.911
SGReader	0.719	0.893
HGCQA(ours)	**0.742**	**0.914**
−relation weight	0.729	0.894

Table 1: Experimental results of accuracy on PQL dataset.

Acknowledgement

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

References

Feng, Y.; You, H.; Zhang, Z.; Ji, R.; and Gao, Y. 2019. Hypergraph neural networks. In *AAAI*.

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*.

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