Question Answering by Learning Transformation on KG Embeddings

Aditay Tripathi
IISc
Bengaluru
aditayt@iisc.ac.in

Apoorv Saxena
IISc
Bengaluru
apoorvsaxena@iisc.ac.in

Abstract

Question Answering (QA) is one of the major areas of research in machine learning. Large scale Knowledge Graphs are often used as a source of external knowledge to solve QA problems since they provide well-structured relational information between entities. However, it is challenging to build QA systems which can learn to reason over KG using question-answer pairs alone for 2 reasons. One, the same question may be phrased in multiple different ways. Secondly, many questions require multi-hop reasoning over the KG to retreive the answer. To address these challenges we propose a novel method that uses KG embeddings like TransE(Bordes et al., 2013), TransH(Wang et al., 2014) and a Bi-LSTM network to extract implied relationship between KG entities. Our method achieves competitive results on 1-hop QA tasks and promising results on 2-hop and 3-hop questions on the MetaQA dataset (Zhang et al., 2018)

1 Introduction

Question answering (QA) is one of the main challenges in the machine learning community. With the introduction of large scale Knowledge Graphs (KG's) like DBPedia (Auer et al., 2007) , Freebase (Bollacker et al., 2008), etc. well structured external facts can be used to answer natural language questions. People traditionally convert a natural language question into logical form using semantic parsers, and then using it to query over KG (Clarke et al., 2010) (Yih et al., 2015). More recently neural network based techniques has been to answer question based on KG's (Weston et al., 2014) (Miller et al., 2016). However, these techniques use KG as a big database of structured KG triplets, ignoring the structure of the graph which can help in logical reasoning. More recently, variational reasoning network (VRN)(Zhang et al., 2018) which tries to learn graph embeddings and multihop inference in end-to-end fashion.

In this project we propose a novel technique to answer questions using Knowledge Graphs. We have made the following contributions:

- We use KG embeddings like TransE (Bordes et al., 2013), TransH (Wang et al., 2014) to encode the structure of Knowledge Graph. This gives us an embedding for each entity.
- We propose a novel Bi-LSTM model that extracts the relation embedding(s) from a question.
- We then use this relation embedding along with the head embedding of the entity in the question to find the tail entity (answer to the question). This method is different depending on the KG embedding used (TransE or TransH).

Our method achieves competitive results on 1-hop questions while at the same time it gives promising results on multi-hop questions.

2 Related Work

2.1 QA with semantic parsers

Traditional approaches use semantic parsers to map the question to a specific logical form and then use it to query the KG(Clarke et al., 2010)(Yih et al., 2015). However, these approaches require domain specific rules or grammar.

2.2 Neural Network Approaches

Approaches like memory networks(Weston et al., 2014) (Miller et al., 2016) (Sukhbaatar et al., 2015) has achieved state-of-the-art results on various QA benchmark datasets. However, their multi-hop reasoning capabilities are limited.

2.3 Graph Embeddings

(Huang et al., 2019) proposed Knowledge Embedding based QA. They focused on answering simple questions(i.e.) 1-hop questions using KG embeddings like TransE (Bordes et al., 2013). KG embeddings, like TransE, learns the low-dimensional vector for entities in KG such that the relation between entities are preserved in the learned embeddings. (Huang et al., 2019) uses these learned embeddings to solve for the QA task for simple questions.

Our proposed methods utilizes the KG embeddings and tries to learn the relation between them in euclidean from the questions and annotated head and tail entities.

3 Model

3.1 Problem Statement

A Knowledge Graph (KG) is a directed multi-relational graph $(V(G), E(G))$ where, $V(G)$ are the vertices of the graph representing entities present in a KG and $E(G)$ are the edges in a graph representing the relations present in a graph. A KG entry is a triple $(e_i^h, r_i, e_i^t) \in E(G)$ where $(e_i^h, e_i^t) \in V(G)$. Given a KG $(V(G), E(G))$ and question answer pairs $\{(q_i, a_i)\}_{i=1}^N$, the problem is to output an entity $e \in V(G)$ that properly answers the question. In training data, head entity e_i^h is annotated.

Our hypothesis is that multi-hop relations can be viewed as a combination of single-hop relations. For example, in the case of TransE, we can view the relation embedding of a 2-hop question (*What genres are the movies directed by Gene Wilder in?*) as the vector sum of the relations $directed\_by^{-1}$ and has_genre. Our aim is to make the model learn such underlying embeddings.

3.2 KG embeddings

KG represents an entity $e \in V(G)$ as a low dimensional vector in Euclidean space. In particular it computes the k-dimensional vector for head entity e^h and tail entity e^t and calculated the scoring function $f_{\mathbf{r}}(e^h, e^t)$ that represents the plausibility of that triplet $(e^h, r, e^t) \in E(G)$ in $E(G)$. The scoring function $f_{\mathbf{r}}(e^h, e^t)$ is used to learn the transformation $\mathbf{r}$ that represents the relation r in (e^h, r, e^t).

In translation based methods i.e. TransE (Bordes et al., 2013) $f_{\mathbf{r}}(e^h, e^t) = ||e^h + r - e^t||_{l1/l2}$

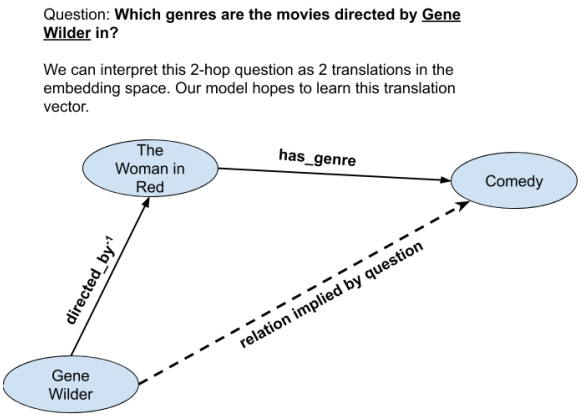

Figure 1: Learning multi-hop relations as combinations of single-hop relations

where $(r, e^t, e^h) \in \mathbb{R}^k$ and on translation using hyper-planes class of methods i.e. TransH (Wang et al., 2014) $f_{\mathbf{r}}(e^h, e^t) = ||(e^h - w_r^T e^h w_r) + d_r - (e^t - w_r^T e^t w_r)||_{l1/l2}$ where $(e^h, e^t, d_r, w_r) \in \mathbb{R}^k$.

In TransE, the transformation $\mathbf{r}$ is a simple translation vector $r \in \mathbb{R}^k$ in Euclidean space. In TransH, the embeddings $e^h \in \mathbb{R}^k$ and $e^t \in \mathbb{R}^k$ are projected in a hyperplane using $w_r \in \mathbb{R}^k$ and the projected e^h embedding is translated in the hyperplane using $d_r \in \mathbb{R}^k$ to get the projection of e^t on the hyperplane. These transformations are illustrated in Fig. 2.

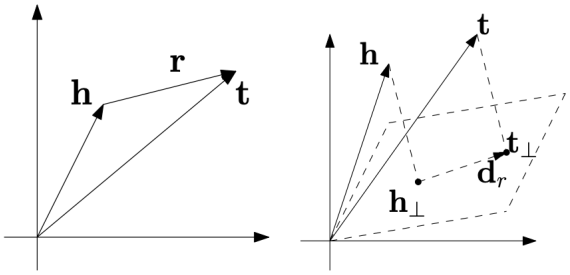

Figure 2: An illustration of TransE and TransH (Wang et al., 2014)

3.3 Answering Module

KG embeddings are largely used in link prediction in KG's. Given a head entity and a relation (e^h, r), the tail entity is predicted by first performing a transformation $f_{\mathbf{r}}(.)$ on e^h and then getting the nearest neighbor based on $||f_{\mathbf{r}} - e_i^t||_{l1/l2} \ \forall \ e_i^t \in V(G)$. The specific transformation $f_{\mathbf{r}}(.)$ depends on the type of embedding used which is explained in section 3.2.

However, in case of Question Answering (QA) problem, the relation r between the KG entities

is unknown. In this work, we tried to learn the implied relation between the head and tail entities in question directly from the question answer pairs. Specifically, we used a Bidirectional LSTM to infer the relation from the question. Let BiLSTM be represented as $f_{LSTM} : \mathbb{R}^t \rightarrow \mathbb{R}^k$, then the learned relation transformation $\mathbf{r} \in \mathbb{R}^k$ from $q \in \mathbb{R}^t$ is given by

$$r_i^{LSTM} = f_{LSTM}\left(q_i\right) \tag{1}$$

Then the tail entity (target entity) is computed by ranking all $e \in V(G)$ according to a scoring function $f_{\mathbf{r}}(e_i^h, e_i^t)$ and retrieving the entity with the best score, where e_i^t is given by:

$$e_i^t = f\left(e_i^h, r_i^{LSTM}\right)$$

The function $f(e, r^{LSTM})$ applies a transformation on head entity e^h and it is scored according to $f_{\mathbf{r}}(e^h, e^t)$.

In case of TransE,

$$f\left(e_i^h, r_i^{LSTM}\right) = e_i^h + r_i^{LSTM} = e_i^t \tag{2}$$

$$f_{\mathbf{r}}\left(e_i^h, e_i^t\right) = ||e_i^h + r_i^{LSTM} - e_i^t||_{l1/l2} \tag{3}$$

and in case of TransH,

$$e_i^t - \left(d_{w_i}\right)^T e_i^t\left(d_{w_i}\right) = e_i^h - \left(d_{w_i}\right)^T e_i^h\left(d_{w_i}\right) + d_{r_i}$$

where:

$$d_{w_i} = f_{LSTM_1}\left(q_i\right) \tag{4}$$

$$d_{r_i} = f_{LSTM_1}\left(q_i\right) \tag{5}$$

and

$$f_r(e_i^h, e_i^t) = ||\left(e_i^h - \left(d_{w_i}\right)^T e_i^h\left(d_{w_i}\right)\right) + d_{r_i} - \left(e_i^t\left(d_{w_i}\right)^T e_i^t(d_{w_i})\right)||_{l1/l2}$$

In TransH, by introducing the mechanism of projecting to the relation-specific hyperplane, it enables different roles of an entity in different relations/triplets.

To train the parameters of LSTM, margin based ranking loss is used which is described as follows:

$$L = \Sigma_{(e^h,q,e^t)}\Sigma_{(e_c^h,q,e_c^t)}\left[f_{\mathbf{r}}(e^h, e^t) + \gamma - f_{\mathbf{r}}(e_c^h, e_c^t)\right]_+, \tag{6}$$

where, (e^h, q, e^t) are the correct triples and (e_c^h, q, e_c^t) is the corrupted triplet in which either e^h or e^t is corrupted, γ is the margin which is fixed at 1 and $[x]_+ = max(x, 0)$.

The Adam (Kingma and Ba, 2014) optimizer is used to train the neural network. The negative sampling rate of 1 is used in our formulation. For every correct triplet, a randomly corrupted triplet is used in training. Either e^h or e^t is corrupted with uniform probability.

4 Experiments

4.1 Dataset Description

We are using MetaQA (MoviE Text Audio QA) dataset (Zhang et al., 2018) to evaluate our approach. It is a challenging Question answering benchmark dataset. It has more than $400K$ single hop, 2-hop and 3-hop questions with multiple answers per question. It provides 3 sets of dataset: vanilla set, paraphrased vanilla questions using NMT to introduce more variability and audio questions to make it even more challenging. However, in our experiments we are using only vanilla set to test our model. There are a total of 21 types of 2-hop questions and 15 types of 3-hop questions.

MetaQA also provides KG which contain $43K$ entities and $134K$ triplets. It has a total of 9 unique relations.

4.2 Model Description

Firstly we are learning 256 dimension TransE and TransH embeddings. The relation is learned directly from question according to eq.(6). A single layer BiLSTM network with hidden layer size of 256 is used to learn the relation. The model is trained minimize the loss given in eq. (6). In our experiments TransH and TransE embeddings are separately learned from KG triplets given with MetaQA and then these embeddings are frozen while learning the relation LSTM. The models are illiustrated in fig. 3 and fig. 4.

5 Results and Discussion

The MetaQA dataset provides training, dev and test sets for 1-hop, 2-hop and 3-hop questions separately. Our model performs best for 1-hop, 2-hop

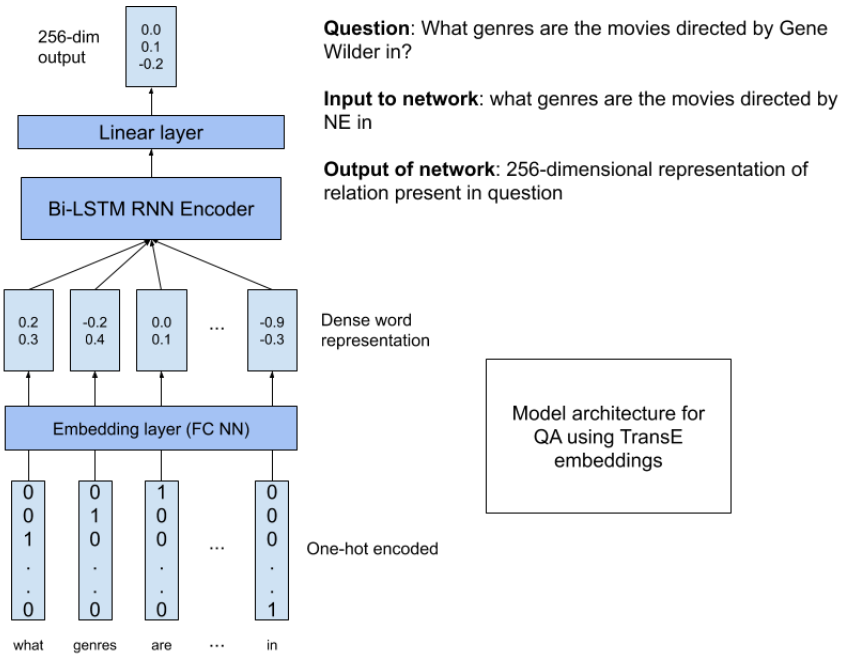

Figure 3: LSTM model to learn relations for TransE embeddings

	TH 1-hop	TH 2-hop	TH 3-hop	TE 1-hop	TE 2-hop	TE 3-hop
Hit@1	**94.06**	11.23	16.47	92.26	**12.3**	10.37
Hit@3	99.1	33.77	35.13	97.83	35.733	29.2
Hit@10	99.93	53.3	50.83	99.73	51.8	50.48

Table 1: TH represents model trained with TransH embeddings, TE represents model trained with TransE embeddings

	TH 1-hop	TH 2-hop	TH 3-hop	TE 1-hop	TE 2-hop	TE 3-hop
Hit@1	**86.15**	42.47	**42.37**	65.67	**44.2**	33.97
Hit@3	96.65	56	56.47	93.4	57.6	59.8
Hit@10	98.9	74.47	75.33	98.53	74.53	77.06

Table 2: TH represents model trained with TransH embeddings, TE represents model trained with TransE embeddings. Thses models are trained are trained with $l1$ norm loss without negative sampling and scored using cosine similarity.

	1-hop	2-hop	3-hop
VRN(Zhang et al., 2018)	**97.5**	**89.9**	**62.5**
Bordes et al.(Bordes et al., 2014)	95.7	81.8	28.4
KV-MemNN (Zhang et al., 2018)	95.8	25.1	10.1
Supervised embedding (Zhang et al., 2018)	54.4	29.1	28.9
Ours	*94.06*	*44.2*	*42.37*

Table 3: Our model performance compared with different baselines.

and 3-hop questions when these are trained on 1-hop , 2-hop and 3-hop set respectively. Our results are shown in Table 1 and Table 2 and all the values are in $\%$age.

Our model has obtained competitive results when compared with baselines provided by (Zhang et al., 2018) as shown in Table 3. Our model achieves more than 94% accuracy for 1-hop questions which shows that LSTM can infer the relation between head and tail entities successfully.

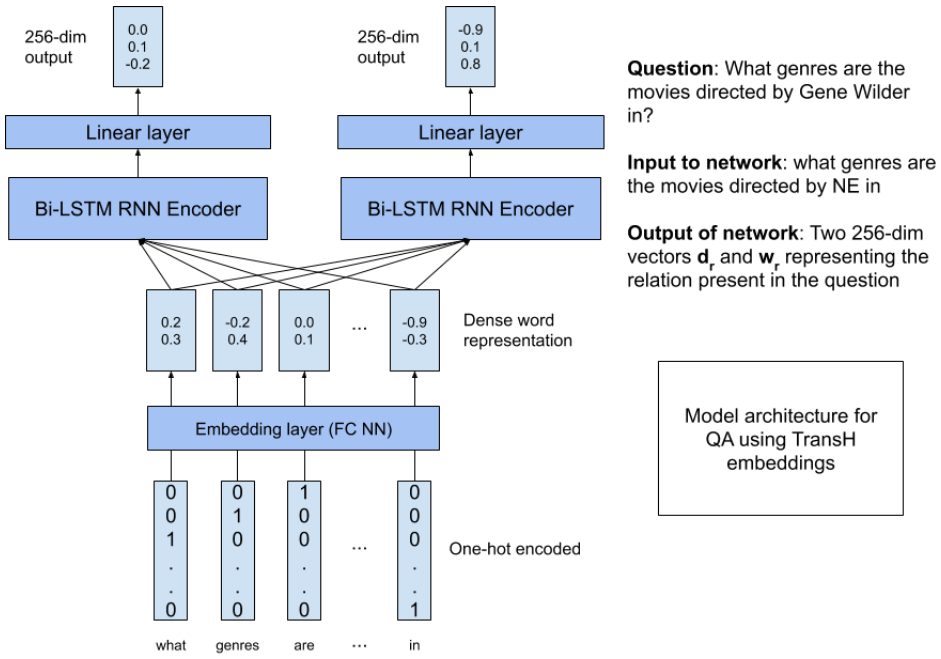

Figure 4: LSTM model to learn relations for TransH embeddings

However, the performance suffers for 2-hop and 3-hop questions. Our model is able to get good Hit@10 numbers for significant percent of questions.

We also trained our models without negative sampling using $L1$ norm as loss function. These models are evaluated using cosine similarity and results are given in Table 2. These models perform significantly better than the original model for 2-hop and 3-hop questions and outperforms two of the baselines given in Table 3.

Upon further analysis of our method on TransE embeddings, it has been observed that for 2-hop questions like

$$\text{"}which\ films\ have\ the\ same\ director\ of\ Harry\ Potter\ and\ the\ Chamber\ of\ Secrets?\text{"},$$

our Top-10 neighbors has 4 Harry Potter movies even though the director has directed only 2 Harry Potter movies and both of them were not in the Top-10 which suggests that TransE is learning to cluster similar movies together even though all of them have different directors. In another instance, we found that TransE has clustered horror movies together even though all of the them have different directors. This seems to be the limitation of current KG embedding methods which has negatively impacted our model for 2-hop and 3-hop questions.

6 Future Work

Our method performs competitively for 1-hop questions and it is promising for 2-hop and 3-hop questions. Since we are freezing the separately learned entity embeddings, the property of KG embedding to cluster similar entities together is negatively impacting out model. We believe that joint training of KG embeddings and relation extraction using LSTM on natural language questions will give better embeddings and will help our question answering model.

Better pre-trained KG embeddings like ConvE (Dettmers et al., 2018) can also help in our QA model. Incorporating ConvE embeddings in our framework would need to redesign of our model and we would explore it further.

In the current model, we are assuming that head entity is annotated in the question. However, such an assumption is not practical. We can not always extract head entity from the question by string matching. Incorporating entity linking into our framework is an interesting research problem.

7 Conclusion

We used KG embeddings like TransE and TransH and BiLSTM neural network to learn the implied multi-hop relation between the entities of KG directly from the question and use learned transformation on entity embeddings to retrieve the answer entity. Our model performs competitively for

1-hop questions and gives promising results for 2-hop and 3-hop questions. However, embeddings like TransE and TransH seems to cluster similar entities together which is negatively impacting our model.

References

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer.

Kurt 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 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. AcM.

Antoine Bordes, Sumit Chopra, and Jason Weston. 2014. Question answering with subgraph embeddings. *arXiv preprint arXiv:1406.3676*.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795.

James Clarke, Dan Goldwasser, Ming-Wei Chang, and Dan Roth. 2010. Driving semantic parsing from the world's response. In *Proceedings of the fourteenth conference on computational natural language learning*, pages 18–27. Association for Computational Linguistics.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Xiao Huang, Jingyuan Zhang, Dingcheng Li, and Ping Li. 2019. Knowledge graph embedding based question answering. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pages 105–113. ACM.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-value memory networks for directly reading documents. *arXiv preprint arXiv:1606.03126*.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Twenty-Eighth AAAI conference on artificial intelligence*.

Jason Weston, Sumit Chopra, and Antoine Bordes. 2014. Memory networks. *arXiv preprint arXiv:1410.3916*.

Scott Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. Semantic parsing via staged query graph generation: Question answering with knowledge base.

Yuyu Zhang, Hanjun Dai, Zornitsa Kozareva, Alexander J Smola, and Le Song. 2018. Variational reasoning for question answering with knowledge graph. In *Thirty-Second AAAI Conference on Artificial Intelligence*.