Retrieve and Re-rank: A Simple and Effective IR Approach to Simple Question Answering over Knowledge Graphs

Vishal Gupta
IIIT Hyderabad, India
vishal.gupta@research.iiit.ac.in

Manoj Chinnakotla
Microsoft, Bellevue, USA
manojc@microsoft.com

Manish Shrivastava
IIIT Hyderabad, India
m.shrivastava@iiit.ac.in

Abstract

SimpleQuestions is a commonly used benchmark for single-factoid question answering (QA) over Knowledge Graphs (KG). Existing QA systems rely on various components to solve different sub-tasks of the problem (such as entity detection, entity linking, relation prediction and evidence integration). In this work, we propose a different approach to the problem and present an information retrieval style solution for it. We adopt a two-phase approach: candidate generation and candidate re-ranking to answer questions. We propose a Triplet-Siamese-Hybrid CNN (TSHCNN) to re-rank candidate answers. Our approach achieves an accuracy of 80% which sets a new state-of-the-art on the SimpleQuestions dataset.

1 Introduction and Related Work

Knowledge Bases (KB) like Freebase (Google, 2017) and DBpedia[1] contain a vast wealth of information. A KB has information in the form of tuples, i.e. a combination of subject, predicate and object (s, p, o). SimpleQuestions (Bordes et al., 2015) is a common benchmark used for single factoid QA over KB.

Question answering (QA), both on KB (Lukovnikov et al., 2017; Yin et al., 2016; Fader et al., 2014) and in open domain (Chen et al., 2017; Hermann et al., 2015) is a well studied problem. Learning to rank approaches have also been applied successfully in QA (Agarwal et al., 2012; Bordes et al., 2014).

In this paper, we introduce an information retrieval (IR) style approach to the QA task and propose a Triplet-Siamese-Hybrid Convolutional Neural Network (TSHCNN) that jointly learns to rank candidate answers.

[1] http://dbpedia.org/

Many earlier works (Ture and Jojic, 2017; Yu et al., 2017; Yin et al., 2016) that tackle SimpleQuestions divide the task into multiple sub-tasks (such as entity detection, entity linking, relation prediction and evidence integration), whereas our model tackles all sub-tasks jointly. Lukovnikov (2017) is more similar to our approach wherein they train a neural network in an end-to-end manner. However, we differ in the fact that we generate candidate answers jointly (matching both subject and predicate using a single query) as well as the fact that we combine both the subject and predicate as well as the question before obtaining the similarity score. At no stage in our approach, do we differentiate between the subject and the predicate. Thus our approach can also be applied in other QA scenarios with or without KBs.

Compared to existing approaches (Yin et al., 2016; Yu et al., 2017; Golub and He, 2016), our model does not employ Bi-LSTMs, attention mechanisms or separate segmentation models and achieves state-of-the-art results. We also introduce a custom negative sampling technique that improves results significantly. We conclude with an evaluation of our method and show an ablation study as well as qualitative analysis of our approach.

2 Our System: IRQA

Our system which consists of two components is as follows: (1) the candidate generation method for finding the set of relevant candidate answers and (2) a candidate re-ranking model, for getting the top answer from the list of candidate answers.

2.1 Candidate Generation

Any tuple in Freebase (specifically, the object in a tuple is the answer to the question) can be an answer to our question. Freebase contains millions of tuples and the FB2M subset provided with

Proceedings of the First Workshop on Fact Extraction and VERification (FEVER), pages 22–27
Brussels, Belgium, November 1, 2018.

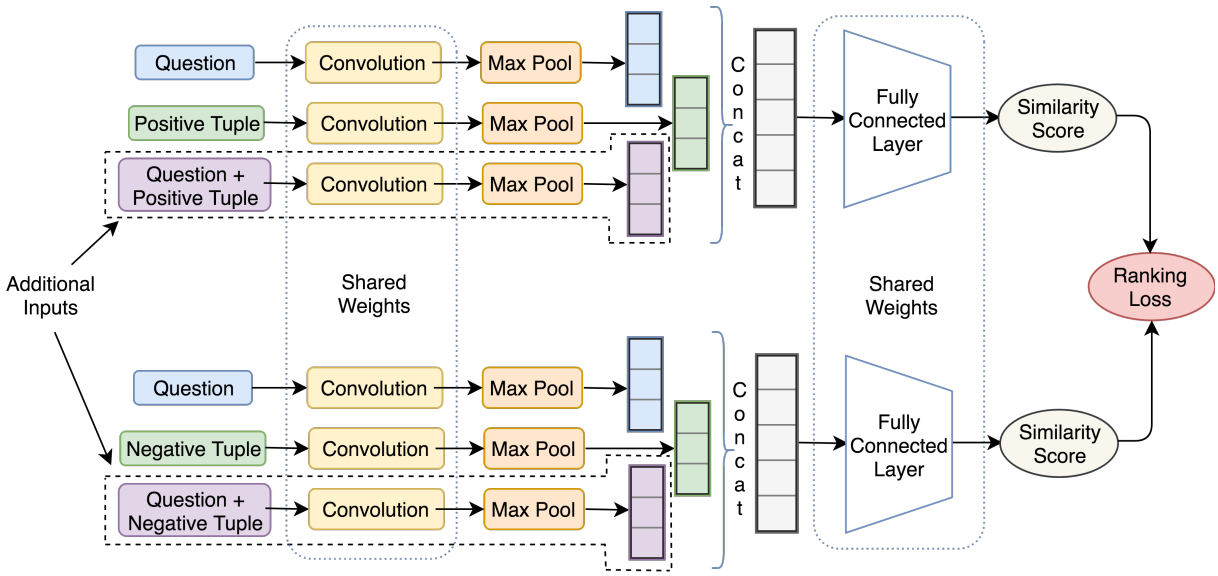

Figure 1: TSHCNN Architecture

SimpleQuestions contains 10.8 million tuples. As such, it is important to reduce the search space to make it feasible to apply semantic-based neural approaches. Thus, we propose a candidate retrieval system to narrow down our search space and focus on re-ranking only the most relevant candidates.

Solr[2] is an inverted index search system. We use Solr to index all our freebase tuples (FB2M) and query for the top-k relevant candidates providing a question as the input query. We adopt BM25 as the scoring metric to rank results. Our results demonstrate the effectiveness of the proposed method.

2.2 Candidate Re-ranking

We use Convolutional Neural Networks (CNN) to learn the semantic representation for input text (Kim, 2014; Hu et al., 2015; Zhang et al., 2015). CNNs learn globally word order invariant features and at the same time pick the order in short phrases. Thus, CNNs are ideal for a QA task since different users may paraphrase the same question in different ways. Siamese networks have shown promising results in distance-based learning methods (Bromley et al., 1993; Chopra et al., 2005; Das et al., 2016) and they possess the capability to learn a similarity metric between questions and answers.

Our candidate re-ranking module is motivated by the success of neural models in various image and text tasks (Vo and Hays, 2016; Das et al., 2016). Our network as shown in figure 1, is a Triplet-Siamese Hybrid Convolutional neural network (TSHCNN). Vo and Hays (2016) show that classification-siamese hybrid and triplet networks work well on image similarity tasks. TSHCNN can jointly extract and exchange information from the question and tuple inputs. We attribute it to the fact that we concatenate the pooled outputs of the question and tuple before input to the fully connected network.

All convolution layers are siamese and share weights in TSHCNN. The fully connected layers also share weights. This weight sharing guarantees that the question and its relevant answer are nearer to each other in the semantics space and irrelevant answers to it are far away. It also reduces the required number of parameters to be learned.

We provide additional inputs to our network which is the concatenation of both the input question and tuple. This additional input is motivated by the need to learn features for both the question and tuple.

2.2.1 Loss Function

We use the distance based logistic triplet loss (Vo and Hays, 2016), which Vo and Hays (2016) report exhibits better performance in image similarity tasks. Considering $\mathcal{S}_{pos}$ / $\mathcal{S}_{neg}$ as the score obtained by the question+positive tuple / question+negative tuple, respectively and $\mathcal{L}$ as the logistic triplet loss, we have:

$$\mathcal{L} = \log_e(1 + e^{(\mathcal{S}_{neg} - \mathcal{S}_{pos})}) \quad (1)$$

[2]http://lucene.apache.org/solr/

Table 1: Network Parameters

Parameter	*Value*
Batch Size	100
Non-linearity	Relu
CNN Filters & Width	90, 10 and 10 filters of width 1, 2 and 3 resp.
Pool Type	Global Max Pooling
Stride Length	1
FC Layer 1	100 units + 0.2 Dropout
FC Layer 2	100 units + 0.2 Dropout
FC Layer 3	1 unit + No Relu
Optimizer	Adam (default params)

Table 2: End-to-End Answer Accuracy for English Questions

Model	*Acc.*
Memory NN Bordes et al. (2015)	62.7
Attn. LSTM Golub and He (2016)	70.9
GRU Lukovnikov et al. (2017)	71.2
BiLSTM & BiGRU Mohammed et al. (2017)	74.9
CNN & Attn. CNN & BiLSTM-CRF Yin et al. (2016)	76.4
HR-BiLSTM & CNN & BiLSTM-CRF Yu et al. (2017)	77.0
BiLSTM-CRF & BiLSTM Petrochuk and Zettlemoyer (2018)	78.1
Candidate Generation (Ours)	68.4
Solr & TSHCNN (Ours)	80.0

Table 3: Candidate generation results: Recall of top-k answer candidates.

K	1	2	5	10	50	100	200
	68.4	75.7	82.3	85.6	91.4	92.9	94.3

Table 4: Candidate Re-ranking: Ablation Study. CQT: Additional inputs, concatenate question and tuple , SCNS: Solr Candidates as Negative Samples

CQT	*SCNS*	*Accuracy*
no	no	49.1
yes	no	68.2
no	yes	69.6
yes	yes	80.0

3 Experiments

We show experiments on the SimpleQuestions (Bordes et al., 2015) dataset which comprises 75.9k/10.8k/21.7k training/validation/test questions. Each question is associated with an answer, i.e. a tuple (subject, predicate, object) from a Freebase subset (FB2M or FB5M). The subject is given as a MID (a unique ID referring to entities in Freebase), and we obtain its corresponding entity name by processing the Freebase data dumps. We were unable to obtain entity name mappings for some MIDs, and removed these from our final set. Our resulting set contained 74,509/10,639/21,300 training/validation/test questions. As with previous work, we show results over the 2M-subset of Freebase (FB2M).

We use pre-trained word embeddings[3] provided by Fasttext (Bojanowski et al., 2016) and randomly initialized embeddings between [-0.25, 0.25] for words without embeddings.

3.1 Generating negative samples

In our experiments, we observe that the negative sample generation method has a significant influence on the results. We develop a custom negative sample generation method that generates negative samples similar to the actual answer and helps further increase the discriminatory ability of our network.

We generate 10 negative samples for each training sample. We use the approach in Bordes et al. (2014) to generate 5 of these 10 negative samples. These candidates are samples picked at random and then corrupted following Bordes et al. (2014). Essentially, Given $(q, t) \in D$, Bordes et al. (2014) create a corrupted triple $\acute{t}$ with the following method: pick another random triple $\acute{t}$ from K, and then, replace with 66% chance each member of t (left entity, predicate and right entity) by the corresponding element in $\acute{t}$.

Further, we obtain 5 more negative samples by querying the Solr index for top-5 candidates (excluding the answer candidate) providing each question in the training set as the input query. This second policy is unique as we generate negative samples closer to the actual answer thereby providing fine-grained negative samples to our network as compared to Bordes et al. (2014) who generate only randomly corrupted negative samples.

[3] https://fasttext.cc/

Table 5: Qualitative Analysis. CA: Correct Answer, PA: Predicted Answer

Examples
Example 1: CA (have wheels will travel, book written work subjects, family) Question: what is the have wheels will travel book about? Predicted Answer: (have wheels will travel, book written work subjects, adolescence)
Example 2: CA (traditional music, music genre artists, the henrys) Question: which quartet is known for traditional music? Predicted Answer: (traditional music, music genre albums, music and friends)

3.2 Evaluation

We report results using the standard evaluation criteria (Bordes et al., 2015), in terms of path-level accuracy, which is the percentage of questions for which the top-ranked candidate fact is correct. A prediction is correct if the system retrieves the correct subject and predicate. Network parameters and decisions are presented in Table 1. We use top-200 candidates as input to the re-ranking step.

4 Results

In Table 3, we report candidate generation results. As expected, recall increases as we increase k. This initial candidate generation step surpasses (Table 2) the original Bordes (2015) paper and comes close to other complex neural approaches (Golub and He, 2016; Lukovnikov et al., 2017). This is surprising since this initial step is an inverted-index based approach which retrieves the most relevant candidates based on term matching.

In Table 2, we present end-to-end results[4] of existing approaches as well as our model. There is a significant improvement of 17% in our accuracy after candidate re-ranking. We attribute it to our TSHCNN model. To obtain insights into these improvements, we do an ablation study (Table 4) of the various components in TSHCNN and describe them in more detail further.

SCNS: Using Solr Candidates as Negative Samples. The scores obtained using our custom negative sample generation method (described in section 3.1), were 17.3% and 41.8% higher as compared to using only 10 negative samples generated as per Bordes et al. (2014), with and without additional inputs respectively. This is a significant improvement in scores, and we attribute it to the reason that negative candidates similar to the actual answer increase the discriminatory ability of the network and lead to the robust training of our network.

CQT: Additional inputs, concatenate question and tuple. Compared to our model without additional inputs, we obtain an improvement of 14.9% and 38.9% in our scores when we provide additional inputs in the form of concatenated question and tuple, with and without our custom negative sampling approach respectively. One possible explanation for this increase is that this augmented network has 50% more features that help it in learning better intermediate representations. To verify this, we add more filters to our convolution layer such that the total features equalled that when additional input is provided. However, the improvement in results was only marginal. Another explanation for this improvement would be that the max pooling layer picks out the dominant features from this additional input, and these features improve the distinguishing ability of our network.

Combining both these techniques, we gain an impressive 62.9% in scores as compared to our model without neither of these techniques. Overall, we achieve an accuracy of 80%, a new state-of-the-art despite having a simple model.

In Table 5, some example outputs of our model are shown. Example 1 shows that the predicted answer is correct (subject and predicate match) but does not match the answer that comes with the question. Example 2 shows we can correctly predict the subject but cannot obtain the correct predicate owing to the high similarity between the correct answer predicate and the predicted answer predicate.

5 Conclusion

This paper proposes a simple and effective IR style approach for QA over a KB. Our TSHCNN model

[4](Ture and Jojic, 2017) reported a 86.8% accuracy but (Petrochuk and Zettlemoyer, 2018) and (Mohammed et al., 2017) have not been able to replicate their results.

shows impressive results on the SimpleQuestions benchmark. It outperforms many other approaches that use Bi-LSTMs, attention mechanisms or separate segmentation models. We also introduce a negative sample generation method which significantly improves results. Such negative samples obtained through Solr increase the discriminatory ability of our network. Our experiments highlight the effectiveness of using simple IR models for the SimpleQuestions benchmark.

References

Arvind Agarwal, Hema Raghavan, Karthik Subbian, Prem Melville, Richard D. Lawrence, David Gondek, and James Z Fan. 2012. Learning to rank for robust question answering. In *CIKM*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. 2015. Large-scale Simple Question Answering with Memory Networks.

Antoine Bordes, Jason Weston, and Nicolas Usunier. 2014. Open Question Answering with Weakly Supervised Embedding Models. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 8724 LNAI, pages 165–180.

Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. 1993. Signature verification using a "siamese" time delay neural network. In *Proceedings of the 6th International Conference on Neural Information Processing Systems*, NIPS'93, pages 737–744, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879. Association for Computational Linguistics.

Sumit Chopra, Raia Hadsell, and Yann LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 539–546, Washington, DC, USA. IEEE Computer Society.

Arpita Das, Harish Yenala, Manoj Kumar Chinnakotla, and Manish Shrivastava. 2016. Together we stand: Siamese networks for similar question retrieval. 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*.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014. Open question answering over curated and extracted knowledge bases. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 1156–1165, New York, NY, USA. ACM.

David Golub and Xiaodong He. 2016. Character-Level Question Answering with Attention.

Google. 2017. Freebase data dumps. https://developers.google.com/freebase/data.

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pages 1693–1701, Cambridge, MA, USA. MIT Press.

Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2015. Convolutional Neural Network Architectures for Matching Natural Language Sentences. *NIPS*, page 2009.

Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. pages 1746–1751.

Denis Lukovnikov, Asja Fischer, Jens Lehmann, and Sören Auer. 2017. Neural Network-based Question Answering over Knowledge Graphs on Word and Character Level. *Proceedings of the 26th International Conference on World Wide Web - WWW '17*, pages 1211–1220.

Salman Mohammed, Peng Shi, and Jimmy Lin. 2017. Strong Baselines for Simple Question Answering over Knowledge Graphs with and without Neural Networks.

Michael Petrochuk and Luke Zettlemoyer. 2018. SimpleQuestions Nearly Solved: A New Upperbound and Baseline Approach.

Ferhan Ture and Oliver Jojic. 2017. No Need to Pay Attention: Simple Recurrent Neural Networks Work! (for Answering "Simple" Questions). *Empirical Methods in Natural Language Processing (EMNLP)*, pages 2866–2872.

Nam N. Vo and James Hays. 2016. Localizing and orienting street views using overhead imagery. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9905 LNCS:494–509.

Wenpeng Yin, Mo Yu, Bing Xiang, Bowen Zhou, and Hinrich Schütze. 2016. Simple question answering by attentive convolutional neural network. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1746–1756. The COLING 2016 Organizing Committee.

Mo Yu, Wenpeng Yin, Kazi Saidul Hasan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2017. Improved Neural Relation Detection for Knowledge Base Question Answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 571–581, Stroudsburg, PA, USA. Association for Computational Linguistics.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NIPS*.