Question Answering Systems Approaches and Challenges

Reem Alqifari
King Saud University /Riyadh, Saudi Arabia
University of York / York, UK
ralgifary@ksu.edu.sa

Abstract

Question answering (QA) systems permit the user to ask a question using natural language, and the system provides a concise and correct answer. QA systems can be implemented for different types of datasets, structured or unstructured. In this paper, some of the recent studies will be reviewed and the limitations will be discussed. Consequently, the current issues are analyzed with the proposed solutions.

1 Introduction

Question answering (QA), are a type of systems in which a user can ask a question using natural language, and the system provides a concise and correct answer. A QA system is different from a search engine in that the user asks a question and the output is an accurate answer instead of a list of relevant documents. A considerable amount of literature has been published on QA, as it has been an object of research since the 1960s (Green et al., 1961).

There are three paradigms of question answering systems, which are:

- The information retrieval approach or free text QA, in which a question is analyzed to determine the answer type, and then Information Retrieval (IR) methods are performed to search a corpus for an answer (Tan et al. 2015; Feng et al. 2015).

- The knowledge base approach (KB-QA), where the question is reformulated as a predicate that has a semantic representation and the system will search datasets of facts. (Zhang et al. 2016; Hao et al. 2017; Yin et al. 2016).

- Hybrid paradigm, where the system combines free text with a KB. Therefore, the coverage of the system will be wider (the probability to find correct answer will be high)(Das et al., 2017).

Researchers have suggested different measures for evaluating a QA system, including precision and recall. The selection of evaluation metrics is mainly dependent on the QA application or track.

There are many types of questions, but they are generally classified into two types: factoid and non-factoid, also known as complex questions. In factoid questions, the question has a specific answer. In contrast, non-factoid questions are open-ended and may have a variety of possible answers (Cohen and Croft, 2016). Moreover, complex questions compromise multi-relations which means that the reasoning is essential.

Our goal is tackling a standard QA system over KB and free text in addition to a reading comprehension QA. To be specific, we will try to build a system that accurately provides answers for temporal questions. Based on my reading, extracting temporal relations is a challenging process that involves capturing the meaning of temporal prepositions, such as before or during. The main challenge in my research is determining how to overcome the complexity and difficulty of answering complex questions that involve reasoning. Furthermore, we will need to consider the domain dependency issue and the effectiveness of using deep learning approaches. The ultimate objective of this research is to improve the accuracy and efficiency of the state of the art.

2 Related Work

The first attempted question answering system was developed in the 1960s when (Green et al., 1961) built a baseball system, which is a sim-

Proceedings of the Student Research Workshop associated with RANLP-2019, pages 69–75,
Varna, Bulgaria, Sep 2–4, 2019.
https://doi.org/10.26615/issn.2603-2821.2019_011

ple closed domain system that answers a question asked in a natural language using a structured database. This study was followed by different systems with many limitations.

Research began to focus on open-domain questions when the Text REtrieval Conference (TREC) started a QA track in 1999. The TREC annual competition has encouraged many research projects in different languages. There are now some other competitions, such as the SQuAD leaderboard and MS MACRO leaderboard. All of that led to a proliferation of studies in QA.

QA can be applied to closed or open domains. In a closed domain, questions are focused on a particular domain, and the answer is extracted from datasets built for this domain only, such as the insuranceQA dataset (Feng et al., 2015). In contrast, in an open domain, the question can be on any subject, and the QA system uses a large corpus with a variety of topics, such as TREC QA.

There are various research areas and applications for QA, including:

- Standard question answering, where the answer comes from the KB or a free text.
- Dialog systems or chatbots are the system used for chatting with an agent. An example of such a system is Siri.
- Community question answering system, where the user asks or posts a question and receives a variety of answers from other users (community). The system has to validate the answers and choose the most relevant and accurate one.
- Multimedia QA, where the question is about an image or video. So, the system has to be able to capture and understand the features of the image or video.
- Reading comprehension QA (RC-QA), where the system is given a question with a passage, and the answer is selected from that passage.

Each path differs from the other in terms of challenges and problems and may depend on different techniques. However, the number of research is increasing in all applications. That because of the emergence and development of deep learning techniques and availability of datasets. The following sub-sections focus on two tracks that are the scope of the study.

2.1 Question Answering over Knowledge-Bases (KB-QA)

Recently, with the rapid growth of large-scale knowledge bases on the Web, such as DBpedia[1] and Freebase, knowledge bases have become very important resources and promising approaches for open-domain question answering. Three basic approaches are adopted in research into KB-QA (Hao et al., 2017):

- The semantic parsing based (SP-based) approach is focused on constructing a semantic parser that converts a question into structured expressions like logical forms (Yih et al., 2015). Semantic parser is also used to turn natural-language questions into structured queries (SPARQL). It is well known that semantic parser is not a straightforward task.
- The information retrieval or relation extraction (IR-based) approach searches for the answer from the KB based on the question. Ranking is used to select the best answers from the candidate list.
- Deep learning or embedding based: questions and answers are represented using semantic vectors (Hao et al., 2017). Then, a similarity matrix is applied to find the most similar answer. The crucial step is computing the similarity.

The recent methodology has three core stages, mentioned below:

- Topic entity: The goal of this step is to define the main topic of the question. Some researchers have used an API to extract the topic of the questions (Yih et al. 2015; Hao et al. 2017).
- Fact finding: Also known as relation extraction. This used to search for a relation with the defined topic entity, which is mentioned in the question, and then provide candidate knowledge triples. Knowledge completion can be used(Yih et al., 2015).
- Answer selection: This method is used to match the question and candidate triples into semantic vectors, and then calculate the semantic relevance score between them using

[1] http://dbpedia.org

the predefined similarity measure. Then, the most similar answer is chosen(Hao et al., 2017).

KB-QA are generally classified into two types: Multi-Relation questions and Single-Relation questions. Multi-Relation questions measure the ability of the system to answer multi-constrained questions. According to (Bao et al., 2016) there are six main constraints. There are three widely used datasets for KB-QA, and all of them are based on the Freebase KB. This means that all of the questions can be answered using Freebase. The SimpleQuestions dataset, introduced by (Hill et al., 2015) named simple because it is tackled the single-relation questions. On the other hand, WebQuestions (Berant et al., 2013) and ComplexQuestions (Bao et al., 2016) are based on multi-relations questions.

Despite the fact that KBs are very large, they are still quite incomplete, missing large percentages of facts. For the QA system, although the answer might not exist in the knowledge base, it can be discovered by using knowledge completion techniques. Some researches have adopted this track to improve the accuracy of QA systems (Toutanova et al., 2016).

The deep learning approach has been widely used in different NLP tasks, including QA. The DL can overcome some limitations, such as the complexity of feature extraction. Also, it can be beneficial for reducing dependency on the rule-based as in the existing SP-based KB-QA systems which affect the generalization. However, the DL methodology has not achieved human performance in QA applications. Hence, Some challenges remain to be tackled, including the lexical gap or vocabulary gap. This means that the user question has a different vocabulary than the KB does. So, the system requires to bridge the gap between the user question and the KB. The problem of the lexical gap can be minimized using word embedding. (Das et al., 2017) tried to overcome the incompleteness and lexical gap by combining text with the KB and using word embedding.

2.2 Reading Comprehension

Reading comprehension (RC) uses questions and answers to test the level of text understanding. Reading comprehension tests are normally used to test the reading level of language learners or children. When RC tests are used to test NL understanding by a computer, this is called machine comprehension. This task requires a machine to answer a question or set of questions from a given passage. The question can be either a multiple-choice or a short-answer question.

RC-QA is challenging, as it involves combination of multiple difficult tasks such as reading, processing, comprehending, reasoning, and finally providing the answer. One of the earliest systems designed to answer reading comprehension tests was QUALM, developed by Lehnert in 1977.

Reading comprehension has gained interest in recent research. There is also a gap between human and machine performance in answering questions because reading comprehension is not about word-based search and context matching. Challenges of machine comprehension QA arise mainly because of reasoning. They include:

- Synthesis: Answering a question requires integrating information distributed across multiple sentences in a passage.
- Paraphrasing: A single sentence in the article may entail or paraphrase the question. Paraphrase recognition may require synonymy and word knowledge.
- Inference: Some answers must be inferred from incomplete information in the passage.

Various deep learning techniques have been applied for reading comprehension. Generally, a variety of models of RNN and attention have been used in recent research such as: (Yu et al. 2018; Xiong et al. 2016; Seo et al. 2016) . Figure 1 reveals the common deep learning architecture that has been used for reading comprehension QA. The main components are:

- Embedding layer: The representation model of the input (the question and the passage), typically Word2Vec or GloVe.
- Encoding layer: The neural network model used for encoding the question and the passage separately. Usually, one of the RNN techniques applied.
- Attention layer: An attention mechanism is applied to capture the relation between the question and the passage.
- Output layer: Generating or finding the answer, depending on the answer type. if the

answer is a text span, the output will be the start and the end position of the answer in the passage. A pointer network can be used.

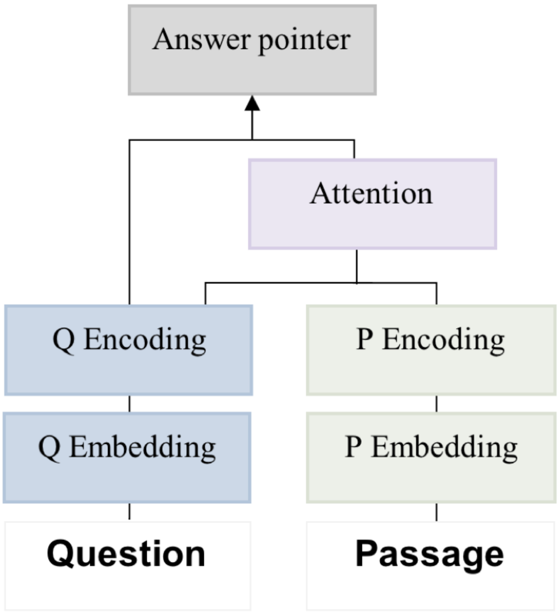

Figure 1: Common Architecture of RC-QA SYSTEM

2.3 Temporal Processing

Temporal language consists of time, event, and temporal relations. Events include occasions, actions, occurrences, and states (Derczynski, 2017). Temporal relations are categorized into three main categories (Pustejovsky et al., 2017):

- Temporal relation (TLINK): Represent the temporal relationship between two events, an event and a time or two times. For example: She submitted the report last week.
- Subordinate (SLINK): Used for modality, evidential and factual. For example: She refused to submit the report.
- Aspectual (ALINK): Only between two events, describing an aspectual connection. For instance: She finished writing the report.

Based on (Bethard et al., 2016), temporal relational extraction is the most difficult step of temporal representation. Temporal QA means the ability to answer any temporal-based question. This encapsulates extracting the temporal information and requires some reasoning. According to (Jia et al., 2018a) and (Bao et al., 2016), the Temporal question can be classified to four categories:

- Temporal answer, where the question asks about time or date.
- Explicit temporal, in which the question contains an explicit date, time, or event, such as: Who was the king of Saudi Arabia at 2014?
- Implicit temporal, in which the question has no explicit temporal term but contains a term such as before, after, or during.
- Ordinal constraints, in which the rank is needed to answer the question, such as What is the third largest continent?

(Jia et al., 2018a) has indroduced TemQuestions datasets. It has been extracted from three KB-QA datasets (Free917, WebQuestions, and ComplexQuestions), whose answer sets are based on Freebase. The released of this datasets has been followed by an implementation of Temporal QA system called TEQUILA by (Jia et al., 2018b). The main limitation of TEQUILA is that it is based on the rule-based approach.

3 Problem Statement and Proposed Contribution

Different types of temporal-based questions of various levels of complexity can be tackled. Temporal reasoning is challenging, and complicated because some events are vague. Also, extracting the temporal relations, which is essential step to answer a temporal question, is challenging. A brief summarization of the recent studies in different directions of KB-QA are provided in Table 1.

As previously mentioned in this paper, the extraction of temporal relations has not yet been solved. Temporal questions can found in KB-QA and RC-QA. Therefore, both tracks might be addressed and different issues might be considered. For example, as mentioned previously, KB-QA has many challenges including: lexical gap, scalability and complexity of understanding natural language questions. On the other hand, the main issue with the RC-QA is understanding the text and reasoning over multiple sentences.

3.1 Research Questions

Based on the explanation above, we must consider the following research questions:

Dataset	State of the art	Evaluation metrics	Methodology	Limitations
WebQuestions	(Yih et al., 2015)	F1=52.5%	SP-based..CNN for semantic similarity	Handcrafted features
WebQuestions	(Hao et al., 2017)	F1=42.9%	IR-based.. Bi-LSTM with cross attention model	
TempQuestions	(Jia et al., 2018b)	F1=36.7%	SP-based..	Hand-coded query templates

Table 1: Summarize some of the recent work on KB-QA.

- What methods can alleviate the out-of-vocabulary problem? How to bridge the lexical gap between the vocabulary of the natural language question and the KB or the context lexical?

- How to understand and model the semantic feature of the complex questions? What is the best method: Is it the semantic parsing or decomposition using reinforcement learning? How to minimize the reliance on the hand-coded rules?

- How to handle the reasoning over the complex questions. And what is the most efficient memory and attention mechanism that should be considered?

- How to represent the temporal questions without using the pre-defiend list of expressions that has been used in (Yih et al., 2015) and in and (Bao et al., 2016)?

3.2 Potential Contribution

Neural turing machines (NTMs) (Graves et al., 2014) together with reinforcement learning (RL) is expected to provide new mechanisms for handling long-term memory that is vital for QA. Hence, the application of NTMs and reinforcement learning for QA and RC will be studied. Using NTM in some problem such as question answering can improve the system because it can mimic the human memory. As NTM can save the information that is useful for answering the question. Although LSTM has internal memory stored in its hidden states, NTM has external memory. Hence, the system can have unlimited memory, and that effectively extended the capabilities of NN. Therefore, using NTM is promising for solving QA problems as QA systems require large and persistent memory. Moreover, considering the difficulty of temporal reasoning, a temporal linkage can be added to the NTM which is an inspiration from the DNC (Graves et al., 2016). Also, applying RL either for query graph generation or for the question decomposition is promising.

In order to overcome the out-of-vocabulary issue as well as the lexical gap, three main strategies will be applied: firstly, using character embedding for the question to overcome the misspelled word or typos. Secondly, using pre-trained word embedding for the question and the context. Also, combine the context with global knowledge, such as Wikipedia or commonsense knowledge.

The proposed approaches will be applied to different datasets (KB and RC) such as WebQuestions and MS MARCO or SQuAD. Additionally, answering temporal questions will be tackled.

4 Conclusion and Future Direction

Despite the promising results of applying deep learning for QA, there are some issued that need to be tackled. Therefore, we will try to handle some of the limitations. This encapsulate understanding of the complex questions, reasoning, and lexical gap. The importance and originality of this study are that it will explore the application of NTMs and reinforcement learning for the complex and temporal questions in KB-QA and RC-QA. Research questions that could be asked include whether one architecture can be applied for both KB-QA and RC-QA and provide high accuracy. Most research studies have considered them as two different problems, but they might not be, as the input text in RC can be seen as a KB.

Acknowledgments

I would like to thank my research supervisors, Dr. Suresh Manandhar, and Prof. Hend Al-Khalifa.Without their support and advice, this paper would have never been accomplished.

References

Junwei Bao, Nan Duan, Zhao Yan, Ming Zhou, and Tiejun Zhao. 2016. Constraint-Based Question Answering with Knowledge Graph. *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers* pages 2503–2514. https://aclanthology.info/papers/C16-1236/c16-1236.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic Parsing on Freebase from Question-Answer Pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Seattle, Washington, USA, pages 1533–1544. http://www.aclweb.org/anthology/D13-1160.

Steven Bethard, Guergana Savova, Wei-Te Chen, Leon Derczynski, James Pustejovsky, and Marc Verhagen. 2016. Semeval-2016 task 12: Clinical tempeval. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. pages 1052–1062.

Daniel Cohen and W. Bruce Croft. 2016. End to End Long Short Term Memory Networks for Non-Factoid Question Answering. In *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval*. ACM, New York, NY, USA, ICTIR '16, pages 143–146. https://doi.org/10.1145/2970398.2970438.

Rajarshi Das, Manzil Zaheer, Siva Reddy, and Andrew McCallum. 2017. Question Answering on Knowledge Bases and Text using Universal Schema and Memory Networks. *arXiv:1704.08384 [cs]* ArXiv: 1704.08384. http://arxiv.org/abs/1704.08384.

Leon R.A. Derczynski. 2017. *Automatically Ordering Events and Times in Text*, volume 677 of *Studies in Computational Intelligence*. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-319-47241-6.

Minwei Feng, Bing Xiang, Michael R. Glass, Lidan Wang, and Bowen Zhou. 2015. Applying Deep Learning to Answer Selection: A Study and An Open Task. *arXiv:1508.01585 [cs]* ArXiv: 1508.01585. http://arxiv.org/abs/1508.01585.

Alex Graves, Greg Wayne, and Ivo Danihelka. 2014. Neural Turing Machines. *arXiv:1410.5401 [cs]* ArXiv: 1410.5401. http://arxiv.org/abs/1410.5401.

Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, et al. 2016. Hybrid computing using a neural network with dynamic external memory. *Nature* 538(7626):471.

Jr. Bert F. Green, Alice K. Wolf, Carol Chomsky, and Kenneth Laughery. 1961. Baseball: An Automatic Question-answerer. In *Papers Presented at the May 9-11, 1961, Western Joint IRE-AIEE-ACM Computer Conference*. ACM, New York, NY, USA, IRE-AIEE-ACM '61 (Western), pages 219–224. https://doi.org/10.1145/1460690.1460714.

Yanchao Hao, Yuanzhe Zhang, Kang Liu, Shizhu He, Zhanyi Liu, Hua Wu, and Jun Zhao. 2017. An End-to-End Model for Question Answering over Knowledge Base with Cross-Attention Combining Global Knowledge. In *ACL*. https://doi.org/10.18653/v1/P17-1021.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015. The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations. *arXiv:1511.02301 [cs]* ArXiv: 1511.02301. http://arxiv.org/abs/1511.02301.

Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. 2018a. Tempquestions: A benchmark for temporal question answering. In *Companion Proceedings of the The Web Conference 2018*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, WWW '18, pages 1057–1062. https://doi.org/10.1145/3184558.3191536.

Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. 2018b. Tequila: Temporal question answering over knowledge bases. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. ACM, New York, NY, USA, CIKM '18, pages 1807–1810. https://doi.org/10.1145/3269206.3269247.

James Pustejovsky, Harry Bunt, and Annie Zaenen. 2017. Designing Annotation Schemes: From Theory to Model. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, Springer Netherlands, Dordrecht, pages 21–72. https://doi.org/10.1007/978-94-024-0881-2-2.

Min Joon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *ArXiv* abs/1611.01603.

Ming Tan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2015. LSTM-based Deep Learning Models for Non-factoid Answer Selection https://arxiv.org/abs/1511.04108.

Kristina Toutanova, Xi Victoria Lin, Scott Wentau Yih, Hoifung Poon, and Chris Quirk. 2016. Compositional Learning of Embeddings for Relation Paths in Knowledge Bases and Text. *Microsoft Research* https://www.microsoft.com/en-us/research/publication/compositional-learning-of-embeddings-for-relation-paths-in-knowledge-bases-and-text/.

Caiming Xiong, Stephen Merity, and Richard Socher. 2016. Dynamic Memory Networks for Visual and Textual Question Answering. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*. JMLR.org, New York, NY, USA, ICML'16, pages 2397–2406. http://dl.acm.org/citation.cfm?id=3045390.3045643.

Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. Semantic Parsing via Staged Query Graph Generation: Question Answering with Knowledge Base. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, Beijing, China, pages 1321–1331. https://doi.org/10.3115/v1/P15-1128.

Wenpeng Yin, Mo Yu, Bing Xiang, Bowen Zhou, and Hinrich SchACEtze. 2016. Simple Question Answering by Attentive Convolutional Neural Network. *arXiv:1606.03391 [cs]* ArXiv: 1606.03391. http://arxiv.org/abs/1606.03391.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. 2018. QANET: COMBINING LOCAL CONVOLUTION WITH GLOBAL SELF-ATTENTION FOR READING COMPRE- HENSION. page 16.

Yuanzhe Zhang, Kang Liu, Shizhu He, Guoliang Ji, Zhanyi Liu, Hua Wu, and Jun Zhao. 2016. Question Answering over Knowledge Base with Neural Attention Combining Global Knowledge Information. *arXiv:1606.00979 [cs]* ArXiv: 1606.00979. http://arxiv.org/abs/1606.00979.