최신 질의응답 독해모델의 정답유형 활용에 대한 분석

Analysis Answer Type Application Ability of State-of-the-Art Reading Comprehension Models for Question Answering Task

Abstract

Recently, the research community has paid attention to the development of Question Answering (QA) systems, thanks to the development of large datasets and benchmarks. In this work, we analyzed the causes of the high performance of Reading Comprehension (RC) models and proposed a new research line to improve the state of the art. We argue that RC models can achieve close to human performance on several benchmarks, due to the implicit use of Lexical Answer Type (LAT) and Semantic Answer Type (SAT). However, if the question does not have any of them, RC models struggle to answer correctly. We also discuss the necessity to create better models that tackle questions without LAT or SAT since that would actually improve the quality of text understanding of RC models.

1. Introduction

Natural Language Understanding has drawn the attention of the research community with the creation of many large datasets and benchmarks like GLUE [1], SQuAD [2], TriviaQA [3], etc. There have been many approaches to these tasks; for example, in the case of Question Answering (QA), there are Reading Comprehension (RC) models, Contextual Graphs (CG) models, and Knowledge Base (KB) models. The current state of the art uses RC models. We hypothesize that these models implicitly learn how to identify Lexical Answer Types (LAT), Semantic Answer Types (SAT) and the types of the entities of the passage to select the answer. We analyzed BiDAF [4], Document QA [5], and BERT [6] fine-tuned for QA on the SQuAD dataset to validate our hypothesis.

There are some previous works on the explicit use of LAT in QA systems [https://arxiv.org/pdf/1703.04816.pdf]. In the case of KBQA, previous work shows high-performance improvement when using explicit LAT information [7, 8]. However, as far as we know, we are the first one analyzing if RCQA models intrinsically use LAT features.

2. Related Work

In Knowledge-Based Question Answering (KBQA), there have been attempts to explicitly utilize the answer type of a question. Since knowledge bases (KB) provide their own entity types, KBQA models try to predict the entity type information of an expected answer given set of entity types inside the KB. In [7], the authors guessed the KB entity type of the answer using the question context. First, they generalized the question to extract a pattern for questions with similar answer types. Next, a BiLSTM model predicted the entity type of the main entity in the question. Then, the main entity of the question was modified by the entity type and the question was transformed into imperative form. This modified question was used by another BiLSTM model to get the final entity type for the answer. The system achieves 2.9% F1 improvement from the baseline [9], which does not consider the answer type explicitly.

In [8], they used the expected answer type when they formulate the query to the KB and when ranking the answer candidates. First, they decomposed the question in multiple simple questions. For each simple question, a context-aware hierarchical classifier predicted the expected answer type. They used a scoring function to find the most probable expected answer type pair. The scoring function considers the granularity and confidence of the answer type to finalize the expected answer type. This answer type was added to the query and also used for ranking the answer candidates which led to an improvement of 5% in the recall with respect to the baseline model [10, 11].

3. Method

There are many possible approaches to create a QA system. The state-of-the-art is based on RCQA (SQuAD, TriviaQA) [2, 3]. We hypothesize that their superior performance is due to the implicit use of Lexical Answer Type (LAT) and Semantic Answer Type (SAT) features. We think that LAT and SAT have a high discriminative power to filter out wrong candidate answers, especially in datasets like SQuAD and TriviaQA because most of their questions have an explicit LAT or SAT. We randomly sampled 20 questions from SQuAD and checked the results from BiDAF [4], Document QA [5], and BERT [6]. Next, we analyzed the reasons why each model fails in some cases. In order to do that, we analyzed a random sample of 20 questions that all models fail, and a sample of 20 questions that only each model fails, i.e.: 20 questions that only BiDAF fails, 20 questions that only Document QA fails, and 20 questions that only BERT fails. All of them were selected randomly. In this way, we can analyze the weak points of each model.

4. Results

4.1 BiDAF, BERT, DocQA common failure cases

In this sample, 8 out of 20 questions do not have an explicit LAT or SAT. The three RCQA models have similar errors and difficulties when answering questions without LAT or SAT. These questions usually ask about the definition, reasoning or explanation of something, and often need a long phrase to answer. Even in some questions with LAT, they do not do well when the LAT is ambiguous. Answer types like ‘experience’, ‘work’ are too general and abstract to select an answer with them. Like other datasets, SQuAD [2] also has a limitation on listing up all possible answers. Sometimes ‘an apple’ is considered a wrong answer even though the right answer is ‘apple’. In this sample, we found nine cases has this problem. To get a more accurate evaluation result, F1 or BLEU scores for long answers are needed. Finally, in many cases, RCQA models fail to retrieve the right answer even though the proposed answer has the right entity type, which clearly states that RCQA models take into account the answer type to select a candidate answer.

4.2 Cases where Only BERT fails

BERT is the model that performs the best among all RCQA models. The current leaderboard of SQuAD [2] reports that all top 10 models are based on BERT. However, BERT also reports a similar pattern failure. 8 out of 20 questions in our sample has no answer type in the question. Among the questions with LAT or SAT, BERT [6] performs well. Even when the answer is wrong, its entity type matches with the LAT or SAT of the question, which shows that BERT takes into account entity-type information. However, there is a question with an abstract LAT, “health problem.” In this case, even though it is more difficult than normal LAT, the model could provide a reasonable prediction. The right answer type for the question is a name of a ‘disease’ but the model answers the ‘reason of death’. Even though it is wrong, entity types are related. It is an indication that BERT also fails when there is insufficient specification of LAT or SAT in questions, suggesting a weakness of the current RCQA approaches. Furthermore, 11 questions in our sample can be evaluated as correct but due to an incomplete golden answer list, they were classified as wrong.

4.3 Cases where Only DocQA fails

Out of 20 questions in our sample, DocQA [5] was able to retrieve a candidate answer with the right entity type in nine questions. This shows that also DocQA is able to use this entity-type information to select candidate answers. In our sample, only 4 of the 20 questions have no LAT or SAT. The remaining questions were actually responded correctly by the model. However, the candidate answer was not included in the answer list.

4.4 Cases where Only BiDAF fails

Among the sample of 20 questions that only BiDAF [4] responds incorrectly, four does not have LAT and seven has a problem with it. These problems are mainly that the LAT is too abstract or it is too difficult to find an answer with that type. For example, there are LATs like entity, faction or conditions, which are too abstract, and also too difficult LATs like "member nations of the Holy Roman Empire." In two questions, the candidate answer provided by BiDAF had a correct type and in another two questions, the type of the answer did not match the LAT of the question. The other wrong answers are due to the incomplete answer list.

4.5 Overall Analysis for RCQA models

We created a simple heuristic to extract LATs from the questions. Extracting only the first noun phrase next to WH word (what, which), we could get a lexical answer type in 93.68% accuracy in TriviaQA dataset [3]. In addition, we also used some WH words (when, who, where) that also mention the answer type (date, person, place, etc.). Using this heuristic, we were able to extract from the dev set 2,212 questions without LAT, i.e., 20% of the dev set. From those questions, BiDAF [4] could not reply properly to 44% of them, and DocQA to 42% of them. This clearly shows that these models have problems with this kind of questions. On the other hand, in the case of BERT [6], this percentage decreases to 28%. This is due to the fact that BERT is actually a language model, rather than an RCQA model, so it is able to understand the language itself better than BiDAF and DocQA [5]; and in questions without LAT the understanding of the question and document is more important to respond correctly than in LAT questions.

Furthermore, only 20% of the questions of the training set of SQuAD do not have LAT or SAT. From this, we can see that most questions do have LAT. Thus, the models get biased towards LAT questions. They can actually learn how to identify entities and filter them out by entity types. However, if there is no LAT, the system struggles in finding an answer. We also analyzed questions in which the type of the answer provided by the models matches the LAT or SAT, but then, the answer is wrong due to the context. For example, in the question 'Which Doctors were in Project: Lazarus?', the document mentions many different doctors who participated in different projects. However, the QA models retrieve the name of a doctor who is not related to "Project: Lazarus." The reason why RCQA models can consider the answer type is on the architecture of the model and the inputs. Since RCQA models utilize word embeddings, it can capture the WH-word (what, when, where, etc.) information in the question. Some RCQA models also consider position embeddings, so they have information about the absolute and relative position of words. Since we already showed that our simple heuristic can extract LATs and SATs, neural models can also easily get this information with word and position embeddings. Lastly, RC models compare the similarity between each word in the question and the document. In this way, they can compare the type of the answer with the LAT or SAT. In the case of BERT, it performs better than others because it considers the semantics of the words on several levels using multi-head attention.

5. Conclusions and Future Work

In this work, we showed that RCQA models learn how to identify Lexical Answer Type (LAT) and Semantic Answer Types (SAT) and how they filter candidate answers using them. We also showed that RCQA struggles when dealing with questions that do not have LAT or SAT. Thus, we propose to the research community to focus on models that can answer questions without LAT or SAT, which are more difficult to respond than questions with any of them. In this way, we can ensure that the model is actually understanding what it reads rather than identifying the type of the entities and filtering out by the LAT or SAT of the question.

Reference

[1] Wang, Alex, et al. "Glue: A multi-task benchmark and analysis platform for natural language understanding." arXiv preprint arXiv:1804.07461 (2018).

[2] Rajpurkar, Pranav, et al. "Squad: 100,000+ questions for machine comprehension of text." arXiv preprint arXiv:1606.05250 (2016).

[3] Joshi, Mandar, et al. "Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension." arXiv preprint arXiv:1705.03551 (2017).

[4] Seo, Minjoon, et al. "Bidirectional attention flow for machine comprehension." arXiv preprint arXiv:1611.01603 (2016).

[5] Clark, Christopher, and Matt Gardner. "Simple and Effective Multi-Paragraph Reading Comprehension." Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). (pp. 845-855). (2018).

[6] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[7] Yavuz, Semih, et al. "Improving semantic parsing via answer type inference." Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 149-159. (2016).

[8] Ziegler, David, et al. "Efficiency-aware Answering of Compositional Questions using Answer Type Prediction." Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers). pp. 222-227. (2017).

[9] Berant, Jonathan, and Percy Liang. "Imitation learning of agenda-based semantic parsers." Transactions of the Association for Computational Linguistics 3. pp. 545-558. (2015).

[10] Bast, Hannah, and Elmar Haussmann. "More accurate question answering on freebase." Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. pp. 1431-1440. (2015).

[11] Bao, Junwei, et al. "Constraint-based question answering with knowledge graph." Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. pp. 2503-2514. (2016).