COSE474-2024F: Final Project Proposal
Dialogue-Based Question Answering: Evaluating NLP Models on Multi-Participant Conversations

Jeongmin Moon

1. Introduction

These days, it is important to provide concise summaries of long texts. Many people often seek summaries of news articles, academic papers and books, often using large language models (LLMs) like ChatGPT to generate those summaries. As a result, Natural Language Processing (NLP) models such as BERT (Devlin et al., 2019) have shown great performance on Question Answering (QA) tasks.

However, most of QA datasets such as Stanford Question Answering Dataset (SQuAD (Rajpurkar et al., 2016)) include document-style input texts. This means NLP models are good at understanding those texts and answering questions based on them. But it is unclear how well the models perform on QA datasets containing different styles of input texts. For example, if conversational texts from a group chat involving multiple participants are given, can the model answer to the question asking about the content of the conversation?

Therefore, it is important to address that uncertainty. By using pre-trained model that shows high performance on document-based QA datasets, I analyzed its capability on conversation-based QA datasets.

2. Related Works

Question Answering (QA) is a key task in NLP that has gained significant attention due to its broad applicability in areas such as information retrieval, customer service, and chat bot system. This section explores existing QA tasks, highlighting diverse approaches and datasets.

Traditional QA Systems Early QA systems usually relied on rule-based approaches, leveraging manually created knowledge bases and linguistic patterns. However, these approaches lacked scalability, as they required extensive domain-specific knowledge and engineering.

Extractive QA The introduction of large-scale datasets, such as SQuAD, marked a turning point for extractive QA. These tasks involve identifying a span of text from a given passage that directly answers the question. Transformer-based models, particularly BERT and its variants like RoBERTa (Liu et al., 2019), have shown outstanding performance in those tasks by leveraging self-attention mechanisms to understand contextual relationships.

Conversational QA Conversational QA focuses on multi-turn interactions including multiple participants where conversational context is maintained across sequential queries. Datasets like CoQA (Reddy et al., 2019) and QuAC (Choi et al., 2018) evaluates conversational capabilities. Transformer-based approaches have shown great performance in understanding conversational context.

3. Method

3.1. Problem Definition

Given a conversational dataset D which includes several conversations between two speakers in chatting system, I created a question-answer pair D_{QA} for each conversations using LLM: $D_{QA} = LLM(D_C)$.

Then, given D and D_{QA}, I created a conversation-based QA dataset D_{conv} and a document-based QA dataset D_{doc} using a rule-based program P: D_{conv}, D_{doc} = $P(D, D_{QA})$.

Then, given D_{conv}, D_{doc} and a pre-trained NLP model M, the performance for each dataset was calculated and compared.

3.2. Conversation-based & Document-based QA

In a QA dataset, a paragraph is typically given to find an answer to the given question. In my case, each conversation is treated as a paragraph. Each conversation in D is structured as follows:

```
Human 1: Hi
Human 2: Any plans for the weekend?
Human 1: my friends are gonna visit me this
     weekend. we might go hiking!
...
```

To generate the conversation-based QA dataset, I simply concatenated each sentence in the conversation. On the other hand, to generated the document-based QA dataset, I transformed each sentence into a third-person narrative style as shown below:

```
Human 1 said 'Hi!'
Human 2 said 'Any plans for the weekend?'
Human 1 said 'my friends are gonna visit me
    this weekend. we might go hiking!'
...
```

After this transformation, I concatenated the sentences. These concatenated texts were then used to build D_{conv} and D_{doc}, each paired with corresponding question and answer sets.

3.3. Generating Questions & Answers

To create questions and answers based on the conversations from D, I employed OpenAI's GPT-4o model. The generation process focused on ensuring that both questions and answers adhered strictly to the information contained within the provided conversation. One question-answer pair was generated in each conversation, and the methodology ensured that no question or answer introduced information outside the scope of the given conversation.

Question Generation Given a conversation as input, GPT-4o was prompted to generate questions that directly referenced the content of the conversation. These questions were designed to test comprehension of specific details within the conversation. Each question focused on either the explicit facts mentioned or the underlying intent of the conversational participants, ensuring relevance to the provided context.

Answer Generation For each generated question, GPT-4o was tasked with producing concise answers that aligned strictly with the information present in the conversation. To ensure brevity, the answers consisted of only a few words.

One-shot Prompting To improve the generation accuracy, a one-shot prompting technique was applied. In this setup, an example of a conversation, along with its corresponding question and answer, was provided as part of the prompt. This single example served as a guide for GPT-4o, enabling it to better understand the desired structure and constraints for question and answer generation. By including a high-quality example in the prompt, the model was able to produce more accurate and precise QA pairs. The example of my prompt is as follows:

```
Based on the following conversation,
    generate a question and an answer. Each
     question should inquire about one
    specific detail within the conversation
    . The answers must be brief and concise
    , and the information should be
    explicitly contained in the
    conversation. Both question and answer
    should strictly adhere to the given
    conversation. An example is as follows:
{
Conversation:
A: What was your breakfast?
B: I ate sandwich.
A: Cool!

Question:
What did A eat for breakfast?

Answer:
Sandwich
}
Now, I will provide the conversation.
```

4. Experiment

4.1. Dataset

I utilized Human Conversation training data [1] from Kaggle. The dataset contains several conversations between two people.

4.2. Model

4.3. Evaluation

5. Results

5.1. Quantitative Results

5.2. Qualitative Results

6. Discussion

6.1. Limitations

7. Problem definition & challenges

I will utilize a pre-trained model M, a dialogue-based QA dataset D, and a document-based QA dataset D', which is produced by preprocessing the dialogue-style input texts of D into a document-style.

Then, the performance of M on the QA task when using D and D' will be compared. Additionally, the performance of M after fine-tuning on D will be also compared.

The main challenges are as follows: it is not difficult to find conversational datasets, but most of them doesn't include any QA labels for the conversations in the datasets. Furthermore, converting D into the document-style D' is another difficulty.

[1] https://www.kaggle.com/datasets/projjal1/human-conversation-training-data

8. Datasets

It is planned to use a conversational dataset from AI-Hub. Because it doesn't include QA labels, proper questions and answers about the conversations should be generated to use it as a dataset D. Alternatively, it can be another choice to find a dialogue-based QA dataset.

9. State-of-the-art methods and baselines

BERT outperformed not only previous state-of-the-art (SOTA) models but also human performance on QA tasks with SQuAD dataset. The performance of BERT in QA task with D and D' will be compared.

10. Schedule & Roles

October 26 $\sim$ November 8: Construct and preprocess dataset D.

November 9 $\sim$ November 22: Evaluate the performance of M on QA task using D and D'.

November 23 $\sim$ November 29: Evaluate the performance of M after fine-tuning on D.

November 30 $\sim$ December 6: Write the report.

References

Choi, E., He, H., Iyyer, M., Yatskar, M., tau Yih, W., Choi, Y., Liang, P., and Zettlemoyer, L. Quac : Question answering in context, 2018. URL https://arxiv.org/abs/1808.07036.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized bert pretraining approach, 2019. URL https://arxiv.org/abs/1907.11692.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. Squad: 100,000+ questions for machine comprehension of text, 2016. URL https://arxiv.org/abs/1606.05250.

Reddy, S., Chen, D., and Manning, C. D. Coqa: A conversational question answering challenge, 2019. URL https://arxiv.org/abs/1808.07042.