Assignment 2

Michael Magdy Nasr Zaki Ghaly, Simona Scala and **Sara Vorabbi**
Master's Degree in Artificial Intelligence, University of Bologna
{ michael.magdynasr, simona.scala6, sara.vorabbi }@studio.unibo.it

Abstract

This report introduces the NLP task of abstractive question answering (QA) on the Conversational Question Answering (CoQA) dataset released by Stanford NLP in 2019. It uses the Hugging Face library to implement and evaluate an encoder-decoder transformer. Two models are warm-started and fine-tuned with pre-trained language model checkpoints.

- QA Transformer
- QA with Dialogue Transformer

In particular, the checkpoints DistilRoBERTa and BERT-tiny are examined using three different seeds. Overall, this report provides a useful implementation and evaluation of an abstractive QA system on the CoQA dataset using transformer-based models warm-started with pre-trained language model checkpoints.

1 Introduction

Question answering (QA) is a fundamental task in natural language processing (NLP) that involves responding to a question posed in natural language. There are two main approaches to QA: extractive QA extracts a response from a pre-defined list of options, such as a document or a database; abstractive QA involves generating a response from scratch, using a combination of understanding the question and its context.

There are several standard approaches for question answering, which can generally be categorized into three main groups: rule-based methods involve using a set of rules (regular expressions) to extract the answer from a text; fact-based approaches consists of using a knowledge base to find the answer to a question; neural-based approaches involve using machine learning models, such as deep neural networks, to learn how to answer questions, which is the approach taken in the project.

The most recent and state-of-the-art approach is pre-trained transformer models such as BERT and its variants (RoBERTa, ALBERT, etc.). These models are pre-trained on a vast amount of data and fine-tuned on a smaller dataset to perform question-answering tasks. In this report, we take advantage of these pre-trained models to warm-start the original encoder-decoder transformer architecture published by Google (Vaswani et al., 2017) to implement and evaluate an abstractive QA conversational system on the CoQA (Reddy et al., 2018) dataset. It contains 127k questions with answers obtained from 8k conversations about text passages from seven diverse domains.

We fine-tuned two models on three seeds: the first QA model takes a context and a question and returns an answer; the second considers the dialogue history when returning it. The experimentation involved warm-starting the encoder-decoder model with the pre-trained DistilRoBERTa and BERT-tiny. These smaller models are chosen for our limited computational resources. The results obtained from our experimentation show that this approach can be used effectively to create a conversational QA system.

2 System description

The pipeline implemented in this work consists of the following steps:

1. Download the CoQA dataset as two JSON files (train and validation).

2. The validation set is used for testing, and the 20% of the train set is used for validation.

3. Load the dataset from the JSON files into Pandas dataframes, removing unanswerable questions.

4. The dataframes are cast as a Hugging Face dataset dictionary of dataset objects. This is useful for efficient processing of large

amounts of data. Due to our limited computational resources, only a portion of the dataset is selected.

5. Tokenization: datasets are tokenized, batched, and prepared for the transformer model.

6. Two encoder-decoder models are created and warm-started with DistilRoBERTa and BERT-tiny. Weight sharing is enabled to improve the training efficiency.

7. In order to train the models with our limited resources, we cut off the context at 256 tokens (which is half the maximum length of the BERT architecture). This allowed us to increase the batch size during the training process which quickened it considerably.

8. Evaluate the models: the SQuAD-F1 is computed for the validation and test sets.

3 Experimental setup and results

We adopted a sequence-to-sequence (Seq2Seq) trainer to extend Hugging Face Transformer's Trainer for encoder-decoder models. It allows using the $generate()$ function during evaluation, which is necessary to validate the performance of encoder-decoder models on most sequence-to-sequence tasks. It also allows one to specify a seed for training. For both models, we performed three trainings on three different seeds $(42, 2022, 1337)$ for three epochs each. Their performances were evaluated on the validation set using the SQuAD Macro F1-score. The results are shown in Table 1.

Finally, we evaluated the performance of the models on the test set, using the SQuAD F1-score as well. The results are reported in Table 2.

4 Discussion

The results of the evaluations show how the models without history perform better, both on the validation and on the test set. Overall the results are not very satisfying as we can see from the F1-score and in the predicted answers. This is also corroborated by the error analysis in which the five worst predictions per sources are shown.

This is due to the fact that we are unable to use the entire dataset and the models we are using (DistilRoBERTa and BERT-tiny) are smaller transformers in size because of limitations in computational resources.

QA Model	seed	SQuAD F1
Shared DistilRoBERTa	42	0.15
Shared BERT-tiny	42	0.12
Shared DistilRoBERTa	2022	0.16
Shared BERT-tiny	2022	0.12
Shared DistilRoBERTa	1337	0.15
Shared BERT-tiny	1337	0.12
QA Model with History	**seed**	**SQuAD F1**
Shared DistilRoBERTa	42	0.11
Shared BERT-tiny	42	0.11
Shared DistilRoBERTa	2022	0.13
Shared BERT-tiny	2022	0.12
Shared DistilRoBERTa	1337	0.13
Shared BERT-tiny	1337	0.07

Table 1: SQuAD F1-scores of the models on the validation set.

QA Model	seed	SQuAD F1
Shared DistilRoBERTa	42	0.19
Shared BERT-tiny	42	0.21
Shared DistilRoBERTa	2022	0.21
Shared BERT-tiny	2022	0.21
Shared DistilRoBERTa	1337	0.20
Shared BERT-tiny	1337	0.21
QA Model with History	**seed**	**SQuAD F1**
Shared DistilRoBERTa	42	0.12
Shared BERT-tiny	42	0.14
Shared DistilRoBERTa	2022	0.13
Shared BERT-tiny	2022	0.21
Shared DistilRoBERTa	1337	0.12
Shared BERT-tiny	1337	0.09

Table 2: SQuAD F1-scores of the QA models on the test set.

5 Conclusion

In conclusion, this report provides a useful implementation and evaluation of an abstractive question-answering (QA) system on the CoQA dataset using transformer-based models warm-started with pre-trained language model checkpoints. Due to the limitations of Google Colaboratory notebooks, we were not able to train the models on the entire dataset. Surely, this had a considerable negative impact on the training performance. However, this is a viable approach to building conversational QA systems.

References

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2018. Coqa: A conversational question answering challenge.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.