Assignment 2: question answering with transformers on CoQA

Mohammadreza Hosseini, Parvaneh Soleimanybaraijany, Fatemeh Rajbaran and **Shakiba sadat Mirbagheri**
Master's Degree in Artificial Intelligence, University of Bologna
{ mohammadrez.hossein2, p.soleimanybaraijny, fatemeh.ranjbaran, shakiba.mirbagheri }@studio.unibo.it

Abstract

This report consists of showing methodologies used to address a common NLP task called question answering, which is concerned with building systems that automatically answer questions posed by humans in a natural language. What we are asked in this assignment is to implement two transformers structures containing DistilRoBERTa and BERTTiny to answer generation with text passage P and question Q, Once with dialogue history H, once without it. The used dataset is CoQa, a novel dataset for building Conversational Question Answering in which the questions are conversational (meaning that without a knowledge of previous questions and answers; it is not possible to answer the following questions), and the answers are free-form text with their corresponding evidence highlighted in the passage. The overall procedure we followed is first to split train, validation and test sets, and then we have done some precess on data to remove unanswerable QA pairs from it. We analyze CoQA in depth by implementing 12 different models both in size and input while performing multiple train/evaluation seed runs and showed that conversational questions have challenging phenomena not present in existing reading comprehension datasets, e.g., coreference and pragmatic reasoning. In order to evaluate the results, we used f1-score.

1 System description

We fine-tune two BERT based models DistilRoBERTa and BERTTiny to perform this task as follows:

Feed the context and the question (and a history of two previous questions and answers in case the model considers also the history) as inputs to the models. Tokenizer takes the aforementioned inputs and returns a two layers input and a three layers input respectively for DistilRoBERTa and BERTTiny. Take two vectors S and T with dimensions equal to that of hidden states in BERT. Compute the probability of each token being the start and end of the answer span. The probability of a token being the start of the answer is given by a dot product between S and the representation of the token in the last layer of BERT, followed by a Softmax over all tokens. The probability of a token being the end of the answer is computed similarly with the vector T. Fine-tune models and learn S and T along the way.

Based on problem definition, we train each two models with three different seeds 42, 2022 and 1337, all of them with and without considering the history. As a result, we would have $2 \times 3 \times 2 = 12$ models to consider.

2 Data

In this experiment, we use fifty percent of CoQa as our dataset. To prepare the data to be ready for further manipulation, we removed 'unknown' answers from the dataset. Then, we use BERT and RoBERTa tokenizers on our data in order to tokenize each question and context pair. This process includes setting a max-length for the input tensor, as well as finding a mapping between start and end spans of context and tokenized context. As the input present in the data are of various lengths (either lower or higher than max-length), we need to both pad short inputs and truncate long ones. This fixed length, however, is a hyperparameter. In our architectures, the maximum length is set to 512. We tried higher fractions of data and lengths of tensors, but despite running the code on Colab, we got out of memory. As expected, there is a trade-off between, on one hand, the fraction of data loaded and the length of input tensor and on the other hand, the accuracy of the models. The former decreases the latter decreases. Decreasing max-length of input tensor, particularly affects models trained with history due to a longer part of context being truncated; which in turn increases the probability of the

Model	Random seed	F1 score
Bert-tiny	42	0.21
	1337	0.16
	2022	0.16
Bert-tiny-history	42	0.14
	1337	0.13
	2022	0.13

Table 1: Bert-tiny performance on the test data

Model	Random seed	F1 score
distilroberta	42	0.48
	1337	0.47
	2022	0.47
distilroberta-history	42	0.54
	1337	0.53
	2022	0.54

Table 2: RoBERTa performance on the test data

answer being truncated. This phenomenon results in a drastic vanish of the advantage expected to be gained by considering history.

During the inspection of the dataset we found some integral issues. They are so big that can change the training objectives. However, it is not possible to see how many of them are present due to the enourmous amount of data rows. For instance, one of the questions starts with "IS is ..." instead of "IS it ...", which can teach the model the wrong lesson that is, following "is" there could occur another "is". As another example we can take the span texts that neither include the answer nor a reason to conclude the answer. We found these examples by looking for instances that their answer is in the context but not in the span text or those which their answer is longer than their span text.

3 Experimental setup and results

For the experiments the hyper parameters are set as follows: loss function is SparseCategoricalCrossentropy. The models are optimized using Adam which works fine for the purposes of this task, with learning rate equals to 5e-5. we use SQuAD F1-score of word overlap as our main evaluation metric. Tables 1 and 2 present the results of each models on the test data. Considering the results, the Bert-tiny with history performs the worst, and DistilRoBERTa with history performs the best as expected, with regard to the number of its trainable params which are about 20 times more than Bert-tiny.

4 Error analysis

The performance of a question answering system is tightly coupled with the complexity of questions asked and the difficulty of answer extraction. In order to analyze the errors generated by the models, we selected the most promising model to consider which is DistilRoBERTa with history and random seed set as 42 (f1-score = 0.54). All of the sources almost made the same errors which are as follows:

1. As stated also in the main paper, all models suffer from pragmatic reasoning and coreference.

2. The questions for which the main parts of the answer are distributed far from each other in the context, are hard to be answered.

3. In most cases our model is predicting the correct answer, however the answer proposed by the dataset is wrong.

4. number of histories included in the input tensor is not enough for the model to find what is a pronoun in the question refering to.

5. In some cases, there are more than one possible answer. For instance, the answer is "the president Lugo", and other acceptable answers are "the president" and "Lugo". Due to the dataset, providing only one of these three answers, the accuray decreases only because of the absence of all different possible true answers.

6. In some cases that pragmatic reasoning is needed to answer a question, despite using it, yet there could be multiple possible answers. For instance one of the questions is "Has she done this before?", for which there are two possible answers present in the context. One is "It is not her first visit." and the other is "This has become an almost-daily practice" which both make sense even considering a long history.

5 Conclusion

In this experiment we implemented two bert based models for question answering task. These models are Bert-tiny and DistilRoBERTa which the latter performs better on test test.