Encoder-Decoder Models for Question Answering on CoQA

Mohammad Amin Nazerzadeh, Davide Baldelli, Mohammad Reza Ghasemi Madani
Master's Degree in Artificial Intelligence, University of Bologna
{ davide.baldelli4, mohammad.nazerzadeh, mohammadreza.ghasemi}@studio.unibo.it

Abstract

In this report we observed the performance of four different encoder-decoder models on a task of question-answering. We studied the effect of the model architecture and specification, the effect of providing the history of the dialogues to the models, possible causes for the errors and shortages, and patterns of mistakes observed in the models.

1 Introduction

There are two main approaches to tackling question answering. One is Extractive QA, in which the model extracts the answer from a context and provides it directly to the user. The second is Generative QA, in which the model generates free text directly based on the context. Moreover, QA systems differ in where answers are taken from. In Open QA, the answer is taken from a context while in Closed QA, no context is provided and the answer is completely generated by the model. In this assignment, we developed an Open Generative QA model. To do so, the CoQA dataset was selected as it provides discourses of question-answer pair dialogues over a context. We selected an encoder-decoder architecture as our baseline and tried to train the model on the dataset with two different variants of BERT (BertTiny, DistilRoberta) to evaluate their performances. Moreover, models could/-couldn't know about the history of previous dialogues in a discourse. In this way we studied how providing the history affects the performance of the models.

2 System description

Our architecture is based on (Rothe et al., 2019), an Encoder-Decoder model with pre-trained encoder and decoder checkpoints (Bert-Tiny (Bhargava et al., 2021), and DistilRoberta (Sanh et al., 2019)) to skip the costly pertaining. The **DistilRoberta-base** is a distilled version of Roberta-base. It has 6 layers, 768 hidden dimensions, and 12 heads, totaling 82M parameters. The **TinyBert** is a distilled version of Bert-base. The model has only 2 layers, 128 hidden dimensions, and 12 heads totaling 4M parameters. The main pipeline of the code is based on :Leveraging Pre-trained Language Model Checkpoints for Encoder-Decoder Models and Fine-tune a warm-started encoder-decoder model (BERT2BERT).

3 Experimental setup and results

Our four models were Encoder-Decoder models where both the encoder and the decoder were initialized from *'distilroberta-base'* or *'prajjwal1/bert-tiny'* checkpoints on Huggingface with/without considering the history of dialogues.

We tested three randomized seeds (42, 2022, 1337) for each of the four models to be able to observe the performance of the models independently of random factors. Models were trained for three epochs. The optimizer was *AdamW* with *lr = 5e-5*. The metric for evaluating the performance of the models is the squad metric version 1 f1 score as described in (Rajpurkar et al., 2016). For the training, we utilized a cross-entropy loss function. We labeled contexts that did not have the answer in their span (with the aid of rationale which is provided in the dataset) as "UNANSWERABLE". This helped to inject into the model the knowledge of whether the answer is present in the given context or not. Moreover, to generate a final unique output for each context-question pair that could have been split into multiples during preprocessing, we discarded the splits that model generated an "UNANSWERABLE" answer, and chose the answer with the lowest perplexity (The answer about which the model was the most confidence) as the final answer of the model. Due to (Reddy et al., 2018), restricting history to only few previous turns is a good choice. They state that all

Table 1: Evaluation F1 scores of the models

	Validation		Test	
	Conversational	Non Conversation	Conversation	Non Conversation
DistillRoberta	0.37	0.31	0.36	0.31
TinyBert	0.15	0.17	0.13	0.16

models suceeded at leveraging history but the gains were little beyond one previous turn.

We performed hyper-parameter tuning on three parameters of the generation method of the decoder model. Namely, number of beams, top k, and top p parameters were tuned and analyzed.

The evaluations results of the four models on the validation and the test set are shown in Table 1.

4 Discussion

The hyper-parameter tuning resulted in only a 0.1 f1 score increase for all models with respect to the greedy approach (You can see the results in the notebook). Hence, we opted for the greedy generation of the outputs.

As shown in table 1, by injecting history to the DistilRoberta model, the f1 score increased by 5 points but it was the reverse for the TinyBert model-the score decreased by 3 points. This could be due to the fact that TinyBert is 40 times smaller than DistilRoberta, prohibiting it to follow the scaling laws for natural language models. In fact, we think that the model loses the capacity to distinguish between the related and unrelated parts of the context to the answer, as the length of the context increases. It could also be explained that by injecting history into the contexts, we may enter a regime of diminishing return as explained in (Kaplan et al., 2020).

As the performance of the model did not depend on the choice of the seed it was trained on (you can find the scores in the notebook), we considered analyzing the five worst model predictions per context source trained with seed 42. The followings were observed (you can find the exhaustive list in the notebook): 1- It can happen that the provided answer of the model is semantically correct but the f1 score is equal to zero. This could be considered a drawback of the evaluation metric for asessing performance. 2- *DistilRoberta* may hallucinate and provide grammatically coherent answers but semantically not present in the context. To overcome the issue, more training computation and data should be provided. 3- *TinyBert* outputs stereotypical answers with regard to specific question types. This can be explained as the model may only capture basic and low-level statistics of some question-answer types and therefore not a proper understanding of the context. 4- *TinyBert* when provided with the history of the dialogue, shows more tendency to output "UNANSWERABLE". This could also explain the drop in performance after providing the history of dialogues to the model. 5- F1 score compensates short-length answers more. As a result, most of the low-scored answers correspond to yes/no question types. 6- Both models are specifically bad with math and mathematical reasoning.

5 Conclusion

In this work, we assessed the performance of four Bert based encoder-decoder models on a task of question answering based on the CoQA dataset.We obsereved that DistilRoberta performed better than TinyBert. Also adding history of the dialogues to the model, increased the performance of the DistilRoberta but not the TinyBert. Then we analyzed worst-case answers of the models and observed general patterns of mistakes of the models. We observed that the models may hallucinate, provide stereotypical answers which are not dependant on the context, and could be bad at math and math reasoning. As a future direction, an extractive-generative approach could be followed to observe its effect on the aforementioned issues.

References

Prajjwal Bhargava, Aleksandr Drozd, and Anna Rogers. 2021. Generalization in nli: Ways (not) to go beyond simple heuristics.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *CoRR*, abs/2001.08361.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. *CoRR*, abs/1606.05250.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2018. Coqa: A conversational question answering challenge. *CoRR*, abs/1808.07042.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. 2019. Leveraging pre-trained checkpoints for sequence generation tasks. *CoRR*, abs/1907.12461.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.