

Stanford University

BERT for Question Answering on SQuAD 2.0 Zhaozhuo Xu, Yuwen Zhang CS 224N

Problem

- Machine reading comprehension and question answering is an essential task in natural language processing. It is always challenging since it requires a comprehensive understanding of natural languages and the ability to do further inference and reasoning.
- Recently, **Pre-trained Contextual Embeddings** (PCE) models like ELMo and BERT have attracted lots of attention due to their great performance in a wide range of NLP tasks.
- In this project, we picked up BERT model and tried to fine-tune it with additional task-specific layers to improve its performance on Stanford Question **Answering Dataset (SQuAD 2.0)**

Data

- We used Stanford Question Answering Dataset (SQuAD 2.0) to train and evaluate our models.
- Samples in this dataset include (question, answer, context paragraph) tuples.
- The paragraphs are from Wikipedia. The questions and answers were crowdsourced using Amazon Mechanical Turk.
- We have around **150k** questions in total, and roughly half of the questions are not answerable.
- If a question is answerable, the answer is guaranteed to be a continuous span in the context paragraph.

It was only the orbit of the planet Mercury that Newton's Law of Gravitation seemed not to fully explain. Some astrophysicists predicted the existence of another planet (Vulcan) that would explain the discrepancies; however, despite some early indications, no such planet could be found. When Albert Einstein formulated his theory of general relativity (GR) he turned his attention to the problem of Mercury's orbit and found that his theory added a correction, which could account for the discrepancy. This was the first time that Newton's Theory of Gravity had been shown to be less correct than an alternative.

What planet seemed to buck Newton's gravitational laws? Ground Truth Answers: Mercury Mercury Mercury Mercury Prediction: <No Answer>

What planet did astrophysisist predict to explain the problems with **Mercury**? Ground Truth Answers: Vulcan Vulcan Vulcan Vulcan Prediction: Vulcan



texture level reconstruction.

✓ Modules on Top of BERT

BERT	BERT is a multi-layer designed bidirectional representations that we		
Encoder/Decoder	RNN Encoder (Bi-L steps of the output tol CNN Encoder: 2D-C the relationship of nea		
Self-Attention	On top of CNN or LS of sentence embeddin and including that po different positions in		
Highway	Optimization of network a neural network can without attenuation. I to adaptively copy or		
Output	BERT Output Layer BiDAF Output Layer model.		

Results and Analysis



Our main idea is to add an encoder-decoder architecture on top of the BERT model. This idea comes from the computer vision area. For multi-view synthesis task, we can use a general auto-encoder to generate the sketch of other views and an additional auto-encoder for

> er bidirectional encoder. By training deep transformers on a carefully al language modeling task, we can get the pre-trained BERT ve are going to fine-tune later with additional output layers.

LSTM, GRU): try to integrate temporal dependencies between timekenized sequence better.

Convolution on the dimension of seq len and hidden state. Extract arby word embeddings in the sequence.

STM to introduce extra source of information to guide the extraction ings. Each position of the output token attends to all positions up to osition. This can help for better interpreting the inference between the output sequence.

vorks with virtually arbitrary depth. By applying a gating mechanism, have paths along which information can flow across several layers It serves as multi-layer state transitions in RNN to allow the network transform representations.

er: A simple linear output layer

ver: the modeling layer + the output layer discussed in BiDAF

✓ Results Summary

ID	Architecture on Top of BERT	F1	EM	
1	BERT-base PyTorch Implementation	76.70	73.85	
2	BERT-base Tensorflow Implementation	76.07	72.80	
3	GRU Encoder + Self-attention + GRU Decoder + BERT-SQUAD-Out	73.59	69.87	
4	BiLSTM Encoder + BiDAF-Out	76.37	73.05	
5	CNN Encoder +Self-attention +BERT-SQUAD-Out	76.49	73.23	
6	CNN Encoder + BERT-SQUAD-Out	76.56	73.64	
7	GRU Encoder + GRU Decoder + BERT-SQUAD-Out	76.85	73.77	
8	CNN Encoder + BiLSTM Decoder + Highway + BERT-SQUAD-Out	77.07	73.87	
9	BiLSTM Encoder + Highway + BERT-SQUAD-Out	77.41	74.32	
10	BiLSTM Encoder + Highway + BiLSTM Decoder + BERT-SQUAD-Out	77.66	74.87	
11	BiLSTM Encoder + BiLSTM Decoder + Highway + BERT-SQUAD-Out	77.96	74.98	
12	Ensemble of 11 and 7	78.35	75.60	
13	Ensemble of 11 and 7 and BERT large case model	79.44	76.966	
Table 1: F1 and EM scores for different architectures (All our implementations are done in PyTorch).				

✓ Error Analysis Example

Questions	model 1	model 3	model 11
Who made fun of the Latin language?	No Answer	Geoffrey Chaucer	No Answer
Who led Issacs troops to Cyprus?	No Answer	Guy de Lusignan	No Answer
Who began a program of church reform in the 1100s?	No Answer	the dukes	No Answer

Conclusion

- We added several components on top of the BERT model as task-specific layers and analyzed their performance compared to BERT baseline model in great details.
- Our best model so far implements BiLSTM Encoder + BiLSTM Decoder + Highway + BERT-SQUAD-Out as the output architecture on BERT uncased base model, and it achieves an F1 score of 77.96 on the Dev set.
- With ensemble technique, we finally achieved an F1 score of **79.44** on the Dev Set and 77.827 on the Test Set, ranked 12th on the 224N leaderboard.

Reference

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pretraining of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

[2] GitHub. https://github.com/huggingface/pytorch-pretrained-BERT, 2018.

