Approach

We have adapted BERT for choosing an answer among multiple candidates. A detailed descriptioin of the bert model can be found in the original paper (Devlin, 2018). On top of the final layer of BERT, we pool the output (into (batch_size, hidden_size)), add a dropout regularizer, add a linear layer (config.hidden_size, 1), and finally a softmax upon the results of all the 5 choices to select the best answer (answer_id from [0, 4]).

Validation Results

hyper-parameters

pretrained model	bert-large-uncased
batch size	32
activation function	gelu
hidden dropout probability	.1
learning rate	1.57513e-05
max_sequence_length	48

57.66% accuracy on the validation dataset.

Instructions on Running

1. (optional) create a Python virtual environment
2. `pip install -r requirements.txt`
3. `python3 pytorch_reimpl.py`

   To opt out of fine-tuning, simply replace the ranges with the hyper-parameter choices above.

Trained models

PyTorch checkpoints are here: https://drive.google.com/drive/folders/1-R5c0KTHsuL0iLA7NpWSW6IK2rE_Xb7G?usp=sharing