

DOCUMENTATION

The approach leveraged to tackle this task of Grammatical Error Detection is by fine-tuning pretrained BERT models.

We see that the no. of positive samples and negative samples in training data are equal (9999 of each) with a total of 19998 examples in the training set. The average length of a sentence is about 12 words, with max length sentence being 319 words.

The model framework is as follows:

1. The input sentences are tokenized and mapped to their token ID's to allow them to be inputted to the BERT model.
2. Start and end tokens are added at each sentence, with PAD tokens also added to pad the sentences to 128 tokens. Sentences greater than 128 tokens are truncated. Attention masks are returned to mask out the PAD tokens while training and evaluating the model's performance.
3. A dataset is created from the token id, attention masks and labels. The training dataset is passed to a RandomSampler to sample the sentences with a batch size of 16. The validation dataset is meanwhile sampled sequentially with the same batch size.
4. AdamW optimizer is used to perform updates on the model parameters. This optimizer uses weight decay to regularize the model (Reference: <https://arxiv.org/abs/1711.05101>).
5. A Sequential BERT classifier (The traditional BERT + a single fully-connected layer) is used as our model. The model is trained for 2 epochs, with a learning rate of $2e-5$.

The model achieves a training loss of 0.589 and a validation loss of 0.69, which increases with the number of epochs, which indicates the model might be overfitting. It achieves a precision of 0.56 and a F1-score of 0.66 on the validation set.



References: <https://arxiv.org/abs/1711.05101>

https://huggingface.co/transformers/v2.2.0/model_doc/bert.html#bertforsequenceclassification

<https://arxiv.org/abs/1810.04805>

<https://gist.github.com/sayakmisra/dbb06efec99e760cf9e5d197175ad9c5#file-grammar-checker-bert-ipynb>

