Project Title: COVID-19 Fake News Detection Using BERT

Yuxiang Wang; Yongheng Zhang; Xinyao Yu; Xuebo Li

1 Introduction

Background From the past year, the whole world has gone through the COVID-19 pandemic. Twitter, Facebook, Instagram and many other social platforms update news on pandemics every day. In this project, we will use and fine tune the pre-trained Bidirectional Encoder Representations from Transformers (BERT) model to train social media news posts, which are already known for truth or fake, for better recognizing those possible false news that may appear in the future.

Related Work Gundapu [1] introduces an ensemble of three transformer models (BERT, ALBERT, and XLNET) to detect fake news. Sun [2] investigates different fine-tuning methods of BERT on text classification task and provides a general solution for BERT fine-tuning.

2 Methods

Dataset We have the balanced training data, which contains 6,420 data entries with variable id, tweet and label. We also have balanced testing data which contains 2,140 data entries with variable id and tweet. There are three main variables in our training dataset: 'id' indicates the id number of the tweet; 'tweet' means the actual context of the tweet/post; lastly, 'label' describes whether the news is real or fake. We combine those two datasets together and randomly split data into training set (90%) and test set (10%).

Setup Data pre-processing is essential for feeding the data into BERT. The pre-processing steps can be summarized as the following steps:

- Load the dataset
- Tokenization and Encoding
  We use BertTokenizer and our own tokenizer to tokenize the tweets. [SEP] and [CLS] tokens need to be added at the end and beginning of every sentence. Then, we map tokens to ids.
- Pad and Truncate
  BERT requires that all sentences must have the same fixed length and the max length of 512 tokens per sentence. We found out that only 10 out of 8560 rows has length that is over 512. Therefore, we set up our max length to 512.
- Attention Masks
  The purpose of adding the masks is to not incorporate the padded tokens into the interpretation of the sentences.

Training and Evaluation Our training and evaluation procedure can be summarized as the following steps:

- Apply the BertForSequenceClassification model.
- Fine tune the BERT model.
- Add additional layers after the fine-tuned model, including CNN and Bidirectional LSTM, for both with and without freezing the parameters in the fine-tuned model.
- Training, hyperparameter tuning, and testing.
- Investigate key words that affect the authenticity of the news.

3 Results

Hyperparameters The learning rate for all of our models is 5e-5. The epochs for BERT finetune(with frozen parameters)+LSTM(2 layers) model are 10, for BERT finetune(without frozen parameters)+LSTM(1 layer) model are 6. The epochs of other models are 4. Besides, BERT finetune+CNN models has 2 CNN layers with 3 as output channels, (1,768) and (2,768) as kernel sizes. For the loss function, we use cross entropy loss for all of our models.

Evaluation Criteria To test our classifiers' prediction results on fake news dataset, we use the following metrics:

- Test accuracy (primary)
  In our task, the test accuracy is the number of news which are correctly classified divided by the total number of news in the test dataset.

- ROC AUC score
  The ROC AUC stands for the area under the curve of ROC. The range of ROC AUC score is 0 to 1, and a large auc value for a model indicates a good performance of the prediction.

- F1 score
  F1 score, ranges from 0 to 1, is calculated from the precision and the recall of our test results. The higher the score, the better performance it indicates.

$$F1 \text{ score} = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

Performance Analysis From the results, the BERT Finetune (with frozen parameters) + LSTM model has the highest test accuracy, ROC AUC and F1 score. The performance of BERT Finetune (with frozen parameters) +CNN model is better than that of the BERT fine-tuned model as well.

Table 1: Model Evaluation

Model	Test acc	Train loss	ROC AUC	F1 score
BERT Finetune	0.9579	0.0036	0.9586	0.9607
Finetune (with frozen parameters) + CNN	0.9591	0.0200	0.9589	0.9622
Finetune (without frozen parameters) + CNN	0.9439	0.0211	0.9449	0.9474
Finetune (with frozen parameters) + LSTM	0.9614	0.0197	0.9607	0.9646
Finetune (without frozen parameters) + LSTM	0.9346	0.0227	0.9351	0.9389

Keywords in Fake News

- Word count
  We count and sort the words in sentences which are classified as fake news by our best model to obtain keywords. For example, excluding some commonly used prepositions, some of the keywords are: cases, corona, news, deaths, tests, today, confirmed, reported, states, total.

- Frequent words and model performance
  We delete those top frequent words listed above in our inputs, and see if the model performance changes after removing those words. As a result, the model performance does not change. This indicates that top frequent words do not usually solo contribute to the overall performance.

4 Discussion

We realize that the performance of models with frozen parameters in the fine-tuned model improves, and the performance of models without frozen parameters in the fine-tuned model does not improve. The reason could be that the size of the dataset does not have enough support to learn those architectures without frozen parameters.

In the future, we might want to try the combination of not pre-trained model with additional layers, as well as exploring ways to find keywords in fake news. Pre-trained model is representative for general tasks but might not for this specific case.

References

[1] Sunil Gundapu and Radhika Mamidi. Transformer based automatic COVID-19 fake news detection system. *CoRR*, abs/2101.00180, 2021.

[2] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. How to fine-tune BERT for text classification? *CoRR*, abs/1905.05583, 2019.