1 Extension Description

We used the BERT[1] (Bidirectional Encoder Representations from Transformers) model for the extension. BERT is a pre-trained language model based on the Transformer[2] architecture, introduced by Google in 2018. Its principles mainly involve two key concepts: pre-training and fine-tuning. In the pre-training phase, BERT learns deep language representations through large-scale language model tasks such as masked language modeling and next-sentence prediction. In the fine-tuning phase, the BERT model can be fine-tuned based on annotated data for specific tasks to adapt to particular application scenarios.

Compared to simple baseline and strong baseline models, BERT has significant advantages in the task of fake news detection. BERT is a Transformer-based model that utilizes attention mechanisms to capture global information in the text, allowing for a better understanding of contextual relationships. This enables BERT to handle long-distance dependencies more effectively, which is crucial for fake news identification as texts often have complex contextual structures. Additionally, the pre-training mechanism of the BERT model allows it to learn universal language representations from large-scale text data, providing strong generalization capabilities. In contrast, traditional models like LSTMs often require more labeled data for training and are prone to overfitting. In the context of fake news detection, where genuine fake news data is typically limited, BERT's generalization performance makes it excel in small-sample scenarios.

In this extension, we used the "bert-base-uncased" model from HuggingFace. Considering model training time, complexity leading to overfitting, and GPU memory constraints, we set the number of BERT model layers to 2. When constructing the model, we used the AutoTokenizer and AutoModelForSequenceClassification from the transformers library to load the tokenizer and model, respectively. For model training, we employed the AdamW[3] optimizer, which introduces weight decay to address potential overfitting issues with Adam[4] in certain cases. The learning rate was set to 2e-5, and weight decay was set to 0.01. Additionally, we used a learning rate scheduler, starting with a small learning rate in the early training stages, gradually increasing it, and then maintaining a relatively stable learning rate in the later stages. This helps improve the model's stability in the early training phase and accelerates convergence in the later stages. We set the warm-up steps to 0.1 times the total steps.

To show the training effectiveness, we used mask prediction from the bert-base-uncased" model. The model achieves **41%** accuracy and **0.52** f1 score on the test set. The results of the fine-tuned BERT-based extension are as follows: accuracy is **68.82%**, and F1-score is **0.51**. This shows an improvement compared to both the simple baseline and the strong baseline. Due to the uneven distribution of samples, leading to insufficient learning of positive class data, future considerations may involve adopting data augmentation to expand positive class data.

2 Github

https://github.com/kevin00hu/CIS-5300-final-project

3 Reference

[1] Devlin, J., Chang, M. W., Lee, K., Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[2] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).

[3] Loshchilov, I., Hutter, F. (2017). Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101.

[4] Kingma, D. P., Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.