Rumour Detection on Covid-19 Related Information

Zhang, Cheng Chuan
Xu, Yuanchen
Zhuo, Ran
Wong, Man Chun
Liu, Yuxi

Abstract

In this report, we propose our solution for Constraint@AAAI2021 - COVID19 Fake News Detection in English challenge, where we rank in the 3rd position with F1 score of 0.9860 on the test set against a top score of 0.9869. In our method, we first propose a baseline LSTM model, and then an ensemble model[1] consisting of different pre-trained language models such as BERT, RoBERTa, etc. We also apply various training strategies such as domain training, learning rate warm up and cosine decay.

1 Introduction

Rumour is defined as an unverified account of an event or source of information. Rumour detection, as illustrated in our project, refers to the use of Machine Learning to discern and verify rumours as either True or False. While rumours can also be classified as unverified (Qazvinian et al., 2011), for our project, we are verifying unverified sources of information on Covid-19 against verified sources.

Rumour detection has been growing in importance as there appears to be a global increase in the need to safeguard the integrity of online information. Sources of unverified information can range from social media users to less-known media channels.

2 Related Work

In this section, we will discuss some of the research that has been done previously surrounding Rumour Detection.

2.1 RNN Models

The RNN model used for rumour detection was first proposed by Jing Ma et al. in 2016 (Ma et al., 2016), which aims at learning the hidden representations that capture the variation of contextual information of relevant posts over time. Through modeling the social context information of an event as a variable-length time series, this model proved effective to capture the dynamic temporal signals characteristic of rumor diffusion and even achieved high accuracy for the early detection of rumours.

Another Recurrent Neural Network with an attention mechanism was published in 2017 (Jin et al., 2017), which was designed to fuse multimodal features for effective rumor detection. In this end-to-end network, image features are incorporated into the joint features of text and social context, which are obtained with an LSTM (Long-Short Term Memory) network, to produce a reliable fused classification. The neural attention from the outputs of the LSTM is utilized when fusing with the visual features.

2.2 Pre-trained Language Models

Pre-training of Language Models in rumour detection has seen recent research, done by Slimi et al in 2021(Slimi et al., 2021). Although their domain of interest lies only in the social media platform Twitter, their methods in pre-trained language models can be adapted and used. Specifically, they made use of domain-specific knowledge to further tune the existing language models - they used RoBERTa (Liu et al., 2019). Their knowledge of the domain allowed them to adjust weights in the optimizers, apply weight decays and maximum sequence lengths since tweets are limited in length. In this way, the fine-tuned RoBERTa pretrained language model(RoPLM) managed to provide rumour-sensitive word embedding for tweets, which improved the performance in recognizing rumour-propagating tweets.

[1] Trained model file can be downloaded here https://drive.google.com/drive/folders/1mddVJaCqdhC2q0vl-bCEM7ElEbBIl1ab?usp=sharing

3 Dataset

The dataset (Patwa et al., 2021) for Constraint @AAAI2021 - COVID19 Fake News Detection in English challenge was provided by the organizers on the competition website[1]. It consists of data that has been collected from various social media and fact checking websites, and the veracity of each post has been verified manually. The news items labelled with "real" were collected from verified sources which give useful information about COVID-19, while those labelled with “fake” were collected from tweets, posts and articles which make speculations about COVID-19 that are verified to be false. The distribution of datasets is shown in Table 1. 52.34% are real samples and 47.66% are fake samples. The dataset is then further split into train, test and validation sets in the ratio of 6:2:2.

Label	Train	Test	Val
Real	3360	1120	1120
Fake	3060	1020	1020
Total	6420	2140	2140

Table 1: Distribution of datasets

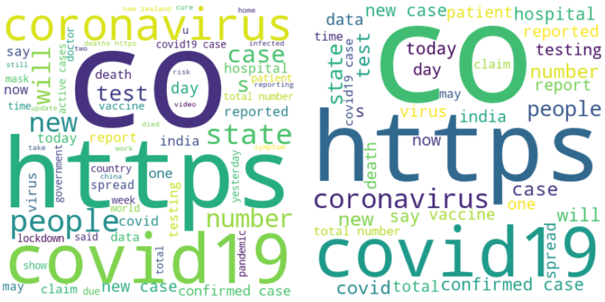
In addition, we analyzed the characteristics of the dataset from the word level. We calculated the word frequencies of the training and validation dataset to generate the corresponding word cloud, as shown in Figure 1. We can see from figure 1 that ’covid19’, ’https’ and ’co’ occur most frequently. The occurrence of "http" and "co" indicates that the dataset contains many url links, possibly due to shortened social media links. We choose to delete these in the data preprocessing stage as they do not provide useful information.

4 Methodology

We propose two models here: one is the LSTM model, and the other is the ensemble model based on pretrained language models.

4.1 LSTM Model

Long-Short Term Memory model have achieved good performance in sequence prediction problems including text sequences data. In LSTM, using a multiple word string to find out the class to which it belongs makes it effective in memorizing important information. As shown in Figure 2, we use a stacked bidirectional LSTM model to obtain a single prediction output with Sigmoid activation function and determine the category using the threshold.



(a) Train word cloud (b) Validation word cloud

Figure 1: Word cloud

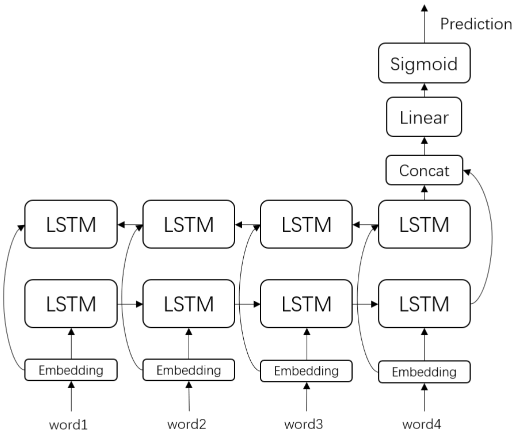


Figure 2: LSTM model

4.2 Ensemble Model

Pretrained language models like BERT have achieved significant results on downstream tasks such as text classification. We use a variety of pretrained language models for fake news detection. In each model, an additional linear layer is stacked on top of the backbone model to obtain the prediction probability for each category. We use four language models, including BERT, RoBERTa (Liu et al., 2019), COVID-SciBERT (pretrained on SciBERT (Beltagy et al., 2019)) and COVID-Twitter-BERT (Müller et al., 2020).

As shown in Figure 3, we use the prediction of each model to obtain our final classification result. Ensemble models can help reduce the generalization error of the prediction. We adopt soft voting to obtain final prediction. In this approach, we take the highest accuracy of each model on the validation dataset as the ensemble weight. The final

[1] https://competitions.codalab.org/competitions/26655

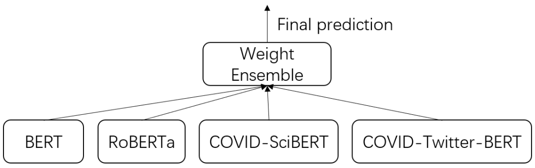


Figure 3: Ensemble model

prediction is given by

$$prediction_{final} = \sum_{i} weight_i \times prediction_i$$

where i is one model. The category with the highest average probability will be selected as the final prediction category.

5 Experiments

5.1 Experimental Settings

- **LSTM** The batch size is 64. The epoch is set to 15. Number of stacked layers is 2. Packing is used for the variable-length text. Threshold is set to 0.5. Embedding dimension is set to 150.

- **Enemble Model** The batch size is 32. The epoch is set to 15. The maximum length of the text is set to 140 and we use the following training strategy.

5.2 Training Strategy

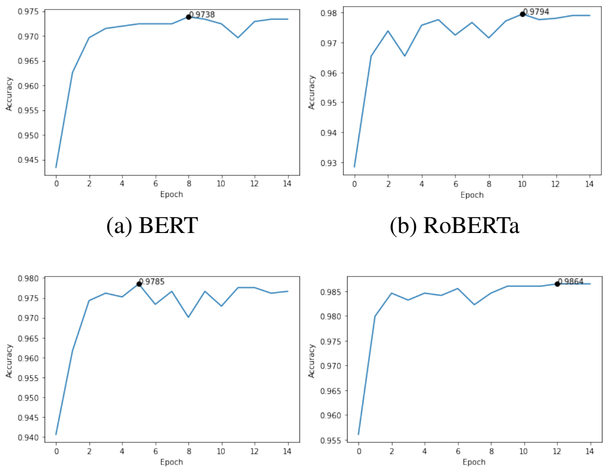
- **Learning Rate Warm Up and Cosine Decay** Learning rate warm-up (He et al., 2016) uses relatively small step sizes at the beginning of training. The learning rate increases linearly or non-linearly to a specific value in the first few epochs. Because the weights are first initialized randomly when training the model, a large learning rate may cause the model to oscillate. After the model is relatively stable, we can choose a preset learning rate for training, which allows the model to converge faster and the model to work better. In our experiments, the learning rate gradually increases to 1e-5 in the 1st epoch.

  In addition, when training deep networks, it is often helpful to anneal the learning rate over time because at the later training stage, if the learning rate is too high, the model may start to oscillate. As a result, after the learning rate warm up phase, we will reduce the learning rate. We use cosine decay (Loshchilov and Hutter, 2016) as our training strategy. In our experiments, after reaching a maximum value of 1e-5, the learning rate will eventually decrease to 0 in the final epoch.

- **Domain Pretraining** Language models such as BERT trained on general domain corpora have obtained impressive gains on various NLP tasks. (Sun et al., 2019) and (Gururangan et al., 2020) show that second phase of pretraining in-domain leads to performance gains. Therefore, we adopt Covid-Twitter-BERT(Müller et al., 2020) which is pretrained on a large corpus of Twitter messages on the topic of COVID-19.

5.3 Results

The weights for ensemble model is given in figure 4. The point annotated in the figure is the highest accuracy achieved by the corresponding model during the training process.



(a) BERT

(b) RoBERTa

(c) COVID-SciBERT

(d) COVID-Twitter-BERT

Figure 4: Validation accuracy during training

In table 2, we present the results of different models on the test dataset.

Method	Accuracy	F1-score
LSTM	0.8752	0.8753
BERT	0.9799	0.9799
RoBERTa	0.9780	0.9780
COVID-SciBERT	0.9776	0.9776
COVID-Twitter-BERT	0.9855	0.9855
Ensemble	**0.9860**	**0.9860**

Table 2: Results of different models on the test dataset

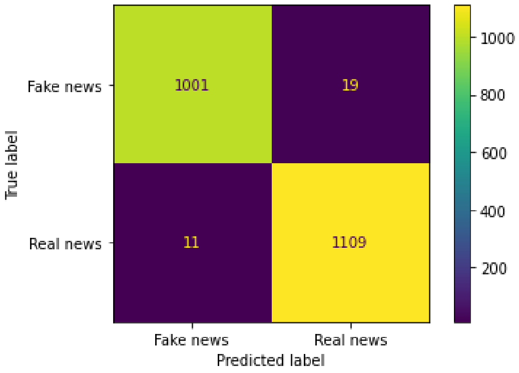


Figure 5: Confusion matrix of ensemble model

As can be seen Figure 4, the model COVID-Twitter-BERT which uses domain pretraining indeed performs better than other models pretrained on general corpora. The model COVID-SciBERT also outperforms the baseline BERT model. The ensemble model using soft voting ends up outperforming all other models. From Table 2, this ensemble model has been able to achieve an accuracy and f1-score of 0.9860. It is able to rank 3rd in the leaderboard, against the leaderboard top score of 0.9869. As can be seen in Figure 5, the model is more likely to predict a given fake news text as real news than to predict real news as fake news.

6 Conclusion

In summary, apart from the LSTM model based on RNN, we have implemented and explored four other types of models including BERT, RoBERTa, COVID-Sci-BERT and COVID-Twitter-BERT, and ensembled them for the rumor detection task.

Given Figure 4 and Table 2, pretrained language models generally proved more efficient than the LSTM model, since all of them could achieve a high validation accuracy over 0.95 in the first epoch of training. These halfway results outperform the final accuracy of LSTM. Besides, we managed to improve the pretrained language models by using them in an ensemble and attain a final accuracy of 0.9860, which is quite close to the top score of leaderboard 0.9869.

As for the future work, we plan to use generative models such as T5 (Roberts et al., 2019) to generate labels directly, further enhancing the predicted results. We will also be more careful with data labeling since some news taken from verified news sources labeled real may prove fake later. Besides, although our trained result can only be applied to detect Covid-19 related rumours, the methodology can be practiced in other general fields.

References

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Zhiwei Jin, Juan Cao, Han Guo, Yongdong Zhang, and Jiebo Luo. 2017. Multimodal fusion with recurrent neural networks for rumor detection on microblogs. In *Proceedings of the 25th ACM international conference on Multimedia (MM '17)*, pages 795–816. Association for Computing Machinery.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2016. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.

Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, J. Bernard Jansen, Kam-Fai Wong, and Meeyoung Cha. 2016. Detecting rumors from microblogs with recurrent neural networks. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*, pages 3818–3824. Research Collection School Of Computing and Information Systems.

Martin Müller, Marcel Salathé, and Per E Kummervold. 2020. Covid-twitter-bert: A natural language processing model to analyse covid-19 content on twitter. *arXiv preprint arXiv:2005.07503*.

Parth Patwa, Shivam Sharma, Srinivas Pykl, Vineeth Guptha, Gitanjali Kumari, Md Shad Akhtar, Asif Ekbal, Amitava Das, and Tanmoy Chakraborty. 2021. Fighting an infodemic: Covid-19 fake news dataset. In *International Workshop on Combating Online Hostile Posts in Regional Languages during Emergency Situation*, pages 21–29. Springer.

Vahed Qazvinian, Emily Rosengren, Dragomir Radev, and Qiaozhu Mei. 2011. Rumor has it: Identifying misinformation in microblogs. pages 1589–1599.

Adam Roberts, Colin Raffel, Katherine Lee, Michael Matena, Noam Shazeer, Peter J. Liu, Sharan Narang, Wei Li, and Yanqi Zhou. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. Technical report, Google.

Hamda Slimi, Ibrahim Bounhas, and Yahya Slimani. 2021. Adapting pre-trained language models to rumor detection on twitter. *JUCS - Journal of Universal Computer Science*, 27:1128–1148.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *China national conference on Chinese computational linguistics*, pages 194–206. Springer.