A BERT based Two-stage Fake News Spreaders Profiling System

Notebook for PAN at CLEF 2020

Shih-Hung Wu and Sheng-Lun Chien

Department of Computer Science and Information Engineering,
Chaoyang University of Technology
Taichung, Taiwan (R.O.C)
shwu@cyut.edu.tw, s10727614@cyut.edu.tw

Abstract. This paper describes our two-stage classification approach to the CLEF 2020 lab: Profiling Fake News Spreaders on Twitter. The task can be briefly defined as: Given a Twitter feed, determine whether its author is keen to be a spreader of fake news. Our approach is to adopt the pretrained model BERT as a tweet classifier and to spot potential spreaders whose tweets are strongly suspected as fake news. The performance of our approach can reach 0.71 on the English data set during developing the system. However, the performance drop to 0.56 in the final PAN at CLEF 2020 shared task.

1 Introduction

A great amount of fake news and rumors are propagated in online social networks. According to the experience on developing anti-spam techniques, it is a good approach to spot the source instead of trying to check the content one-by-one. The aim of profiling fake news spreaders task at PAN-2020 is to know if it is possible to discriminate authors who have posted some fake news in the past from those who have never done it before [1].

The organizers propose the task from a multilingual perspective, and provide data set in English and Spanish, and recommend the participants to take part in both languages. The uncompressed dataset consists in a folder per language (en, es). Each folder contains an XML file per author (Twitter user) with 100 tweets and the filename of these XML files corresponding to the unique author IDs. There are also a separate truth.txt file with the list of authors and the ground truth of whether they are fake news spreaders or not. The performance of a system will be ranked by accuracy in discriminating between the two classes.

However, due to the limitation of time and resource, we just build a system only for tweets in English based on the content analysis and skip the tweets in Spanish. The decision process of our system is a two-stage classification approach to the

Profiling Fake News Spreaders on Twitter task. Our system adopted the pre-trained bidirectional transformer language model, known as BERT [2] as our NLP tool for content analysis. During the training phrase, we fine-tune of the pretrained model BERT as a tweet classifier and use it to classify each tweet as potential fake news or not. Then our system spot a spreader by checking the percentage of each author's tweets that is classified as fake news. If the percentage is higher than a threshold, then we consider the author is a fake news spreader.

2 The BERT Pre-trained Model

The system flow is shown in the following figures. Figure 1 shows the BERT model and classifier architecture. The core of our system is the pretrained language model "BERT". The BERT model is a bidirectional transformer pre-trained using a combination of masked language modeling (MLM) objective and next sentence prediction on a large corpus comprising the Toronto Book Corpus and Wikipedia. BERT stands for Bidirectional Encoder Representations from Transformers. BERT is designed to pre-train deep bidirectional representations from unlabeled text. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create models for new tasks. The implementation of BERT that we use is BERT for sequence classification from Hugging face library[1]. The pretrained model is "bert-base-uncased", that required all English text to be in lower case. The hyper-parameter in the training phrase: Hidden size = 768, Learning r= 6.0e-5, and Vocab = 30522. We train the model 10 epochs in each experiment setting.

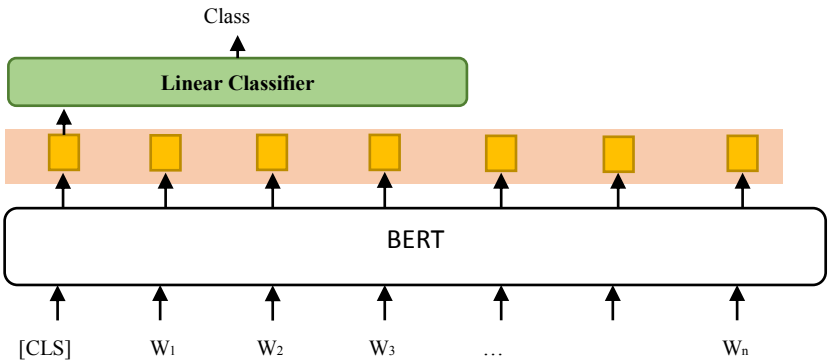

Figure 1: BERT model and classifier.

Figure 2 shows how our system do the training. In the training phrase, we have done some our training data preprocessing. We extracted every tweet and added the truth data for each XML file. Each XML file contains 100 tweets, and will be associated with the same label. All the non-English characters are filtered, only

[1] https://huggingface.co/transformers/model_doc/bert.html

English characters are kept for training data. Then we combine all data into a training dataset for the model to let it learn which tweet may be telling the fake news or not.

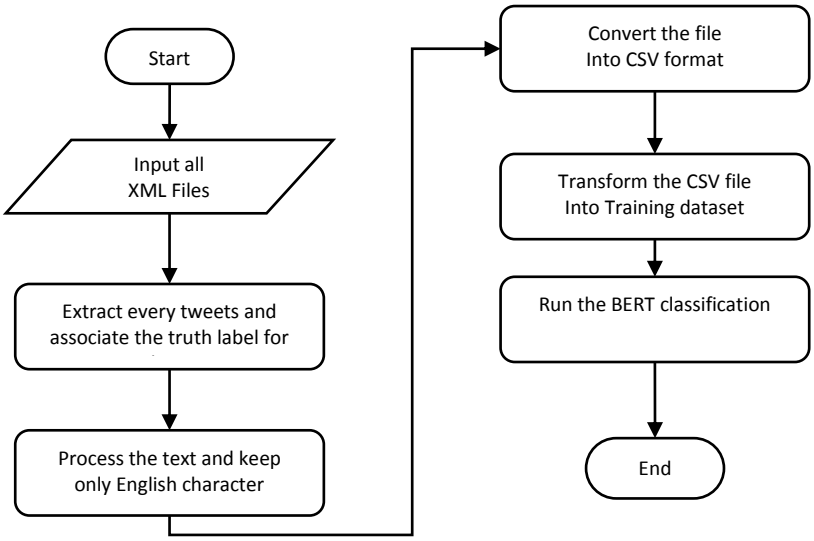

Figure 2: system architecture.

3 System Development

There are 300 authors and each has 100 tweets in the training set. We find that half of them are spreaders however we do not know whether each tweet is a fake news or not. However, we assume that all tweets belong to the spreaders are potential fake news and all tweets belong to the non-spreaders are real news. Thus, we trained a classifier that can classify the news into potential fake ones and real ones.

We know this assumption is imprecise, the classifier cannot spot the fake news well. So when we need to use it to spot a spreader, we set a threshold mechanism to prevent overly identify too many authors as spreaders. Only if the percentage of an author's tweets passed the threshold he/she will be labelled as spreader. An author with only a few tweets that are classified as fake news will not be labelled as a spreader. The decision is made by an empirical threshold. We divide the training set into two parts and use this developing set to find the best threshold, where 70% of the data used as training set and 30% of the data used as test set. Figure 3 shows the accuracy vs. threshold result, where the threshold range from 60% to 90%. The system can get a 0.71 accuracy value with a threshold 74%. The threshold is selected manually and used in our system.

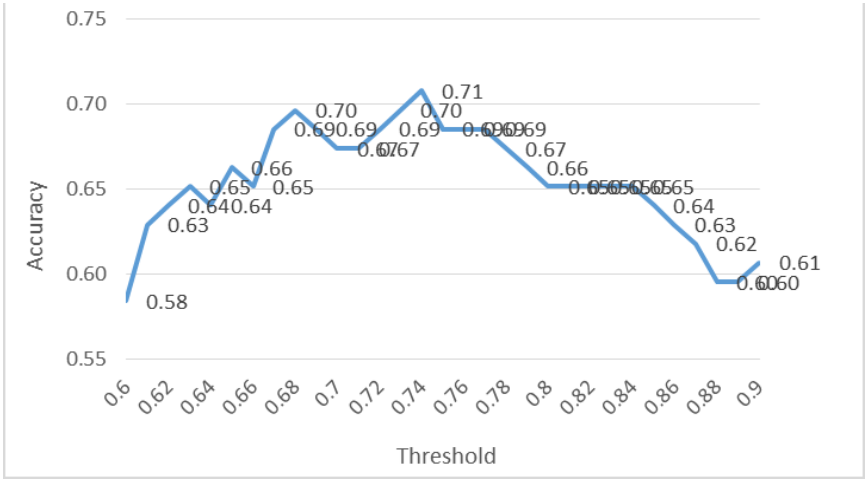

Figure 3: Threshold vs. accuracy result on training set.

To know how the system might perform, we conduct several similar experiments on the training set. Table 1 shows the test results. The accuracy value is around 0.65 to 0.71 given enough training data, i.e. 60% to 80% of the data in training set. We expect that our system can get similar result in formal test.

Table 1: System performance during develop, we divide the English training set as training part and test part with the threshold 74%

Training data to test data ratio	Accuracy
Training50% test50%	0.47
Training60% test40%	0.66
Training70% test30%	0.71
Training80% test20%	0.65

Figure 4 shows how our system do the test. Before testing the data, we also do the data preprocessing first. We extracted every tweet for the one Author file (XML file) and used the model to predict every tweet. After all tweets of one author were labeled 1 or 0, we have a threshold mechanism to make decisions on whether the author is a spreader or not by checking the percentage of 1 exceeded 74% or not. Then our system will put the final answer with author id to a XML file and finish the task.

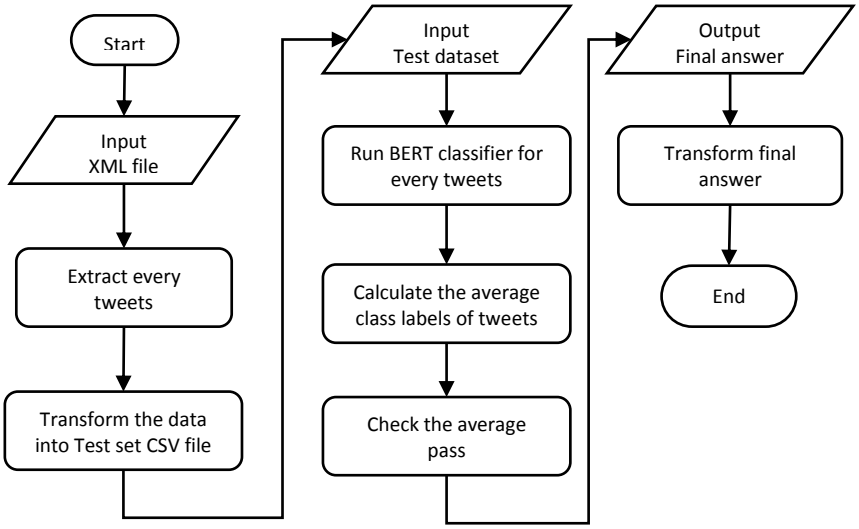

Figure 4: Our system flow.

4 Test Result and Discussion

The test part is taken on the virtual machine provided by the organizers, we met some technical error. In the training phrase, all the non-English characters are filtered, only English characters are kept for training data, but we omitted this part during the test phrase. This is one of the reasons that our system performance decreased. Table 2 shows our system official final test result vs. some benchmarks. The accuracy value of our system is 0.560, which is equal to the LSTM benchmark but lower than our best result on the development set.

Table 2: Our system performance of the final result vs. benchmarks.

Test runs	Accuracy
SYMANTO (LDSE) [5]	0.745
EIN [6]	0.640
LSTM	0.560
Pan20-author-profiling-test-dataset-2020-02-23	0.560
RANDOM	0.510

5 Conclusion

This paper describes our two-stage classification approach to Profiling Fake News Spreaders on Twitter task. The performance of our approach can reach 0.7 on the development set. However, the performance drop to 0.56 in the final PAN evaluation at CLEF 2020 shared task.

As future work, we intend to investigate what are the other information that might help to detect fake news spreaders [3]. For example, in addition to news content and labels, fake news articles in some datasets also provide information on social network of Twitter which contains Twitter users and their following relationships, i.e., user-user relationships, and how the news has propagated (tweeted/re-tweeted) by users, i.e., news-user relationships [4].

Acknowledgements

This study was supported by the Ministry of Science and Technology under the grant number MOST 109-2221-E-324-024

Reference

1. Rangel F., Giachanou A., Ghanem B., Rosso P. Overview of the 8th Author Profiling Task at PAN 2020: Profiling Fake News Spreaders on Twitter. In: L. Cappellato, C. Eickhoff, N. Ferro, and A. Névéol (eds.) CLEF 2020 Labs and Workshops, Notebook Papers. CEUR Workshop Proceedings.CEUR-WS.org
2. Devlin, j., Chang, M., Lee, K., and Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, arXiv:1810.04805v2 [cs.CL] (2018)
3. N. Ruchansky, S. Seo, and Y. Liu. CSI: A Hybrid Deep Model for Fake News Detection. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pages 797-806. ACM, (2017)
4. K. Shu, S. Wang, and H. Liu. Beyond News Contents: The Role of Social Context for Fake News Detection.In WSDM, (2019)
5. Rangel F., Franco-Salvador M., Rosso P. A Low Dimensionality Representation for Language Variety Identification. In: Postproc. 17th Int. Conf. on Comput. Linguistics and Intelligent Text Processing, CICLing-2016, Springer-Verlag, Revised Selected Papers, Part II, LNCS(9624), pp. 156-169 (arXiv:1705.10754)
6. Ghanem, B., Rosso, P., and Rangel, F. (2020). An Emotional Analysis of False Information in Social Media and News Articles. ACM Transactions on Internet Technology (TOIT), 20(2), pp. 1-18.