NLP North at WNUT-2020 Task 2: Pre-training versus Ensembling for Detection of Informative COVID-19 English Tweets

Anders Giovanni Møller
IT University of Copenhagen
Rued Langgaards Vej 7
2300 Copenhagen
agmo@itu.dk

Rob van der Goot
IT University of Copenhagen
Rued Langgaards Vej 7
2300 Copenhagen
robv@itu.dk

Barbara Plank
IT University of Copenhagen
Rued Langgaards Vej 7
2300 Copenhagen
bapl@itu.dk

Abstract

With the COVID-19 pandemic raging worldwide since the beginning of the 2020 decade, the need for monitoring systems to track relevant information on social media is vitally important. This paper describes our submission to the WNUT-2020 Task 2: Identification of informative COVID-19 English Tweets. We investigate the effectiveness for a variety of classification models, and found that domain-specific pre-trained BERT models lead to the best performance. On top of this, we attempt a variety of ensembling strategies, but these attempts did not lead to further improvements. Our final best model, the standalone CT-BERT model, proved to be highly competitive, leading to a shared first place in the shared task. Our results emphasize the importance of domain and task-related pre-training.[1]

1 Introduction

The amount of COVID-19 pandemic cases is rapidly approaching 25M world wide, with almost 1M people who have lost their life to the merciless disease, according to worldometer.[2] This paper exploits the capabilities of Natural Language Processing (NLP) techniques to extract informative tweets, and is a participation in the WNUT-2020 Task 2: Identification of informative COVID-19 English Tweets (Nguyen et al., 2020).

Social media is a useful medium for rapid access to information about the pandemic - but along with all the informative tweets comes an even larger amount of non-informative information. Being able to extract what is informative, and hereby leave out all the non-informative posts, is vital in monitoring and tracking the development of COVID-19. In the shared task, informative tweets were defined as to contain information about COVID-19 cases such as statistics, locations or travel history. Figure 1 shows two examples from the training data.

Informative
Oklahoma's first confirmed case of coronavirus is in Tulsa County <URL>#SmartNews
Uninformative
Trump could cure Coronavirus 19, AIDS, and Cancer in the same day and the media would say he wasn't doing anything.

Figure 1: Examples of INFORMATIVE and UNINFORMATIVE tweets from the training data.

The introduction of neural networks has led to an increase in performance for many natural language processing tasks (Manning, 2015). However, previous work on classification showed that SVMs with character and/or word n-grams often still outperform neural networks (Zampieri et al., 2017; Medvedeva et al., 2017; Çöltekin and Rama, 2018; Basile et al., 2018). Neural network approaches can elegantly exploit raw data, by pre-training word embeddings using a language modeling objective. Recently, more powerful contextual embeddings were introduced (Peters et al., 2018; Devlin et al., 2019), which base each word embedding on its context. These contextual embeddings are generally pre-trained on huge amounts of raw data, and then fine-tuned on the target task. This leads to the question: *How do the three types of classification models viz. SVM, neural models with pre-trained embeddings and various contextual models compare and perform in this classification task?* (**RQ1**)

Neural networks as well as transformer-based models can directly exploit additional raw data by

[1] source code is available on: https://github.com/AGMoller/noisy_text/

[2] https://www.worldometers.info/coronavirus/

pre-training. This pre-training often leads to superior performance, depending mainly on the size and distribution of the pre-training data. Although no additional annotation effort is necessary for pre-training, it often comes with huge computational cost and exhaustive training time. Recent work has shown that selecting data which matches the target domain better (domain-specific or task-specific) is important for transformer-based pre-training (Gururangan et al., 2020; Gu et al., 2020). This leads to the question: *How important is task-specific pre-training for detection of informative COVID-19 tweets?* (**RQ2**)

Finally, we are interested in the supplementary of the three different architectures. Even though one model outperforms the other two models, it can still be that they have different strengths, and combining them can thus lead to superior performance. Our last question is: *Can we ensemble SVM, neural network and BERT-based models to improve robustness?* (**RQ3**)

2 Methodology

Below we will discuss our implementations of each of the classifiers and the ensemble models.

2.1 SVM

We used the linear SVM classifier with default parameters from Scikit-learn (Pedregosa et al., 2011) as basis for our implementation. We experimented with n-grams on a variety of levels. Besides the standard word and character n-grams, we also evaluate wordpiece n-grams (Schuster and Nakajima, 2012).[3] For each granularity (character, word piece, word), we systematically evaluated each range of n between 1-7. We found the optimal range of n to be 1-2 for words, 5-6 for characters, and 1-2 for word pieces. When combining all features, and ablating one group, we found that the highest score was obtained with word and character n-grams which was used in the final model. We found that adding word pieces led to a small performance decrease.

2.2 Neural Networks

We experimented with two different neural architectures, a multi-layer perceptron (MLP) and a 1-dimensional convolutional neural network (Conv1d). The text input was embedded using GloVe embeddings (Pennington et al., 2014), pre-trained on 2B English tweets with 27B tokens and a vocabulary of 1.2M words. These embeddings are chosen, because they are trained on Twitter data, even though this data was sampled before the COVID-19 pandemic. The embeddings were not further tuned during training but were kept static.

The MLP is a two-layer neural net using ReLU as activation in the hidden layers. The layers consist of 1024 and 512 neurons respectively. Between the two layers a dropout with a rate of 0.5 is applied. The Conv1d model consists of a single layer of 1-dimensional convolution with 64 filters and a kernel size of 5, max-pooling with a pool-size of 2, and a dropout layer with a rate of 0.5. Both architectures apply a sigmoid function in the final output layer.

2.3 BERT

Three different pre-trained transformer models were used and evaluated to investigate the impact of diverse pre-training domains and task-specific fine-tuning. All transformer models were fine-tuned on 4 epochs and optimized using the AdamW optimizer (Loshchilov and Hutter, 2017) with a learning rate of 2×10^{-5} and an epsilon value of 1×10^{-8}. We used the following transformers:

- BERT base (uncased) (Devlin et al., 2019): pre-trained on the BookCorpus dataset (Zhu et al., 2015) consisting of 800M words and English Wikipedia with 2.5B words.
- RoBERTa base (Liu et al., 2019): similar to BERT base, but has extended the training data with CC-News (Nagel, 2016), OpenWebText (Gokaslan and Cohen, 2019) and Stories (Trinh and Le, 2018), a total amount of 160GB of text.
- Covid-Twitter BERT (CT-BERT) (Müller et al., 2020): based on BERT-Large, but has been trained further on a collection of 22.5M corona related tweets collected from January 12 to April 16, 2020. The data consisted of 40.7M sentences and 633M tokens.

Where BERT-base is trained on $\sim$3B words unrelated to COVID-19, RoBERTa is trained on much more data, and the CT-BERT training data is similar in size as BERT-base, but matches the domain of our task.

2.4 Ensembling

In an attempt to achieve better performance, different ensembling experiments were carried out,

[3] We used the mBERT word piece vocabulary.

Model	F1	Note
SVM	83.64	word 1-2 grams, char 5-6 grams
SVM	83.54	word 1-2 grams, char 5-6 grams, word-piece 1-2
MLP	78.05	200d Twitter GloVe embeddings
Conv1d	75.52	200d Twitter GloVe embeddings
BERT-base	89.86	
RoBERTa-base	89.59	
CT-BERT	**92.19**	
Ensemble Model	**F1**	**Note**
CT-BERT, RoBERTa, BERT-base	88.19	Soft voting
CT-BERT, RoBERTa, BERT-base	89.99	Hard voting
CT-BERT, SVM	92.19	Thresholding
Random Forest Classifier	**91.67**	Stacking

Table 1: Model results evaluated on the development data using weighted F1 score as metric.

which included majority voting, stacking and thresholding.

Majority: Majority voting was used among the BERT-models, both with hard and soft voting. In hard voting classification, each transformer model would provide a predicted label, and the majority label would be the final prediction. In soft voting, each model produces a probability for each class using a sigmoid function. The final prediction is the class with the highest average probability. All three models were weighted equally.

Stacking: Our second ensembling approach is stacked generalization (Wolpert, 1992), where we trained a meta-classifier which takes the predictions of all other models as input as well as their confidence. Confidence being the probability for each class. We tuned this step in a 10-fold setup on the development data. As classifier, we chose a random forest classifier (Breiman, 2001), because it can model different types of features (binary and continuous), and can model feature interactions intrinsically.

Thresholding: We test whether CT-BERT outputs can be replaced with SVM predictions whenever the confidence score of CT-BERT is below a certain threshold. Here, we use SVM as second system instead of the better performing BERT models because SVM in nature is very different compared to CT-BERT, and is thus more likely to give a complementary perspective.

3 Evaluation

3.1 Data

The data used in this work is provided in connection with the shared task (Nguyen et al., 2020) of the 2020 W-NUT workshop. The training data consists of 7,000 tweets, the validation data 1,000 tweets and the final test data of 12,000 tweets of which 2,000 where annotated and used for the final scores and ranking. All the data has a close to equal class distribution being either INFORMATIVE or UNINFORMATIVE.

3.2 Individual Model Evaluation on Development Data

As found in Table 1, CT-BERT had the overall best performance on the validation data scoring a weighted F1 of 0.92185. Compared to the other transformer models, the domain-specific pretraining appears to be crucial in this specific classification task, with a relative difference in weighted F1-score of absolute 2.6% higher than BERT-base and 2.9% higher than RoBERTa (**RQ2**). When comparing BERT-base and RoBERTa one can observe that they achieve almost similar results, despite being pre-trained on datasets covering different domains. This shows that the domain on which the embeddings are trained is more important compared to the size (RoBERTA is trained on more general text compared to BERT-base, whereas CT-BERT is trained on domain-specific data).

The two neural networks using GloVe embeddings achieved the lowest scores among the tested

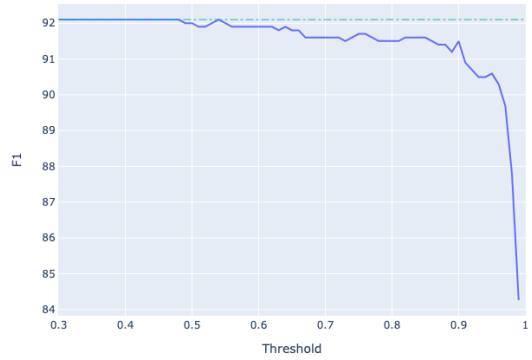

Figure 2: Thresholding: CT-BERT predictions with SVM replacement

models with an F1 of 78.05 for the MLP model, and 75.52 for the Conv1d model. They both suffered from lack of fine-tuning of parameters and non-contextualized embeddings.

The SVM achieved an F1 score of 83.64, being in the middle when comparing the three types of models. Our initial ablation study in the selection of n-gram features allowed for an increase in performance of $\sim 2\%$. This, however, shows that an SVM with sparse n-gram input can outperform neural models with pre-trained embeddings, confirming previous work (Section 1) (**RQ1**). It should be noted that an SVM does not require any pre-training, and is much faster and cheaper to train, so in specific situations it could in fact be the preferred solution.

3.3 Ensembling on Development Data

Majority: Neither hard voting nor soft voting managed to overcome the performance of standalone CT-BERT. Soft voting, which was based on the probabilities of the two labels, achieved an F1 score of 88.19. Using hard voting, where each transformer model contributes with a single predicted label, an F1 score of 89.99 was achieved.

Stacking: The stacking model proved to be the best ensemble model and was used as alternative model to our standalone CT-BERT in the official shared task submission, which allowed for two final submissions. On development data, our 10-fold development setup achieved an F1 score of 91.67. This is a relative difference of -0.55% compared to CT-BERT (**RQ3**). The standalone model is more accurate and more efficient, and hence the preferred solution over ensembling.

Model	F1
Ensemble (Random Forest)	90.54
CT-BERT (ours)	**90.96**
Highest (team NutCracker)	**90.96**

Table 2: Results on the test data, we evaluate our best individual model, best ensemble model and the highest score achieved in the shared task. According to F1, our system shares the first place with team NutCracker.

Thresholding: CT-BERT proved to perform the best compared to the other models. In an attempt to assist CT-BERT when the confidence score of a prediction was below a certain threshold, the non-neural SVM model would provide its prediction on the given input tweet and replace the CT-BERT prediction.

Figure 2 shows the F1 score when testing different confidence thresholds. The dashed line indicates standalone CT-BERT. We found that replacing CT-BERT does not at a single point obtain better F1 score than the standalone model. When the threshold surpasses a certain lower boundary, all predictions are solely from CT-BERT. This is the reason why the maximum F1 score achieved is equal to standalone CT-BERT, and why it is not considered as the best ensemble model. In the other end when the threshold approaches 1, all predictions are provided by the SVM.

3.4 Test data

Results in Table 2 confirm that ensembling is not beneficial over using the output of CT-BERT directly. It appears from the evaluated scores on both the development and test data that task-specific pre-training is crucial in this particular classification task, and that complementing CT-BERT with ensembling did not improve the performance. Furthermore, the performance of CT-BERT is confirmed by comparing to the other participants of the shared task, where it ranked 1st out of 98 submissions from 55 teams (according to official F1 score ranking; we rank 2nd if we consider both F1 and accuracy). Our CT-BERT model shares the first place with the submission by team NutCracker.

4 Conclusion

In this paper we have presented our winning participation for the shared task of WNUT-2020 on Identification of Informative COVID-19 English Tweets. We evaluated three types of models; SVM,

neural networks with pre-trained embeddings, and transformer models. We found that the transformer-based covid-related CT-BERT model performed the best, achieving an F1 score of 90.96 on the hidden test data **(RQ1)**. Evaluating our models on the development data, we found that the CT-BERT model, which was pre-trained on domain and task-related data, performed better than BERT-base and RoBERTa pre-trained on data unrelated to the shared task **(RQ2)**. Different types of ensembling approaches were tested in an attempt to improve robustness. This included majority voting, stacking and thresholding. We found stacking to be most competitive, albeit it still underperformed compared to standalone CT-BERT (-0.55%) **(RQ3)**.

Acknowledgement

We would like to thank the organizers for this shared task. Part of this research is supported by a grant from Danmarks Frie Forskningsfond (9063-00077B).

References

Angelo Basile, Gareth Dwyer, Maria Medvedeva, Josine Rawee, Hessel Haagsma, and Malvina Nissim. 2018. Simply the best: Minimalist system trumps complex models in author profiling. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 143–156, Cham. Springer International Publishing.

Leo Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

Çağrı Çöltekin and Taraka Rama. 2018. Tübingen-oslo at SemEval-2018 task 2: SVMs perform better than RNNs in emoji prediction. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 34–38, New Orleans, Louisiana. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Aaron Gokaslan and Vanya Cohen. 2019. Openwebtext corpus. http://Skylion007.github.io/OpenWebTextCorpus.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2020. Domain-specific language model pretraining for biomedical natural language processing. *arXiv preprint arXiv:2007.15779*.

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. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

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. *CoRR*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

Christopher D. Manning. 2015. Computational linguistics and deep learning. *Computational Linguistics*, 41(4):701–707.

Maria Medvedeva, Martin Kroon, and Barbara Plank. 2017. When sparse traditional models outperform dense neural networks: the curious case of discriminating between similar languages. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 156–163, Valencia, Spain. Association for Computational Linguistics.

M. Müller, Marcel Salathé, and P. Kummervold. 2020. Covid-twitter-bert: A natural language processing model to analyse covid-19 content on twitter. *ArXiv*, abs/2005.07503.

Sebastian Nagel. 2016. https://commoncrawl.org/2016/10/news-dataset-available/.

Dat Quoc Nguyen, Thanh Vu, Afshin Rahimi, Mai Hoang Dao, Linh The Nguyen, and Long Doan. 2020. WNUT-2020 Task 2: Identification of Informative COVID-19 English Tweets. In *Proceedings of the 6th Workshop on Noisy User-generated Text*.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152. IEEE.

Trieu H. Trinh and Quoc V. Le. 2018. A simple method for commonsense reasoning. *CoRR*, abs/1806.02847.

David Wolpert. 1992. Stacked generalization. *Neural Networks*, 5:241–259.

Marcos Zampieri, Shervin Malmasi, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, Jörg Tiedemann, Yves Scherrer, and Noëmi Aepli. 2017. Findings of the VarDial evaluation campaign 2017. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 1–15, Valencia, Spain. Association for Computational Linguistics.

Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *CoRR*, abs/1506.06724.