

# EdinburghNLP at WNUT-2020 Task 2: Leveraging Transformers with Generalized Augmentation for Identifying Informativeness in COVID-19 Tweets

Nickil Maveli

ILCC, School of Informatics  
University of Edinburgh  
n.maveli@sms.ed.ac.uk

## Abstract

Twitter and, in general, social media has become an indispensable communication channel in times of emergency. The ubiquitousness of smartphone gadgets enables people to declare an emergency observed in real-time. As a result, more agencies are interested in programmatically monitoring Twitter (disaster relief organizations and news agencies). Therefore, recognizing the informativeness of a Tweet can help filter noise from the large volumes of Tweets. In this paper, we present our submission for *WNUT-2020 Task 2: Identification of informative COVID-19 English Tweets*. Our most successful model is an ensemble of transformers, including RoBERTa, XLNet, and BERTweet trained in a Semi-Supervised Learning (SSL) setting. The proposed system achieves an  $F_1$  score of **0.9011** on the test set (ranking 7<sup>th</sup> on the leaderboard) and shows significant gains in performance compared to a baseline system using FastText embeddings.

## 1 Introduction

In late December 2019, there was an identification of an outbreak of a novel coronavirus causing COVID-19.<sup>1</sup> Due to the rapid spread of the virus, the World Health Organization declared a state of emergency. Among several social media platforms, Twitter provides a powerful lens for identifying people’s behavior, decision-making, and sources of information before, during, and after wide-scope events, such as natural disasters (Becker et al., 2010). Due to the low signal-to-noise ratio, identifying relevant information in Tweets is a challenging task.

The basic goal of WNUT-2020 Task 2 (Nguyen et al., 2020) is to automatically identify whether a COVID-19 English Tweet is Informative or

not. Such Informative Tweets provide information about recovered, suspected, confirmed, and death cases as well as the location or travel history of the cases. About 4M COVID-19 English Tweets are daily being posted on Twitter, most of which are, however, not informative. In many scenarios, it is not always clear whether a person’s Tweet is announcing a disaster response. Consider an example of an UNINFORMATIVE Tweet from the dataset as shown in Figure 1.

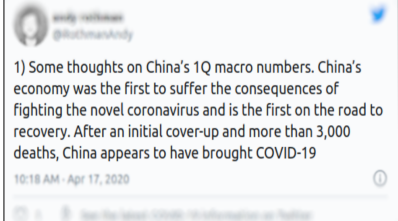
Text	Label
 1) Some thoughts on China's 1Q macro numbers. China's economy was the first to suffer the consequences of fighting the novel coronavirus and is the first on the road to recovery. After an initial cover-up and more than 3,000 deaths, China appears to have brought COVID-19 10:18 AM · Apr 17, 2020	0

Figure 1: A hard to classify Tweet.

However, this observation is near inscrutable examining only the vocabulary used; the Tweet contains a variety of top frequent informative words (“*coronavirus*”, “*covid-19*”, “*deaths*”). This example hints that in order to reach meaningful results, we may have to examine contextual linguistic features, model the annotator’s bias, introduce adversarial examples, and so on (Geva et al., 2019; Goodfellow et al., 2015).

In this paper, we build an ensemble of Transformer (Vaswani et al., 2017) models to leverage its strength in capturing contextual information. The data used to train these models is an augmented version carefully designed to alleviate the confirmation bias and thereby improve generalization. The final inference result is the majority voting of the class from all the constituent models through optimal thresholding as a post-processing step. Our best model (ensemble) achieves  $F_1$  scores of **0.9248**

<sup>1</sup><https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7159299/>

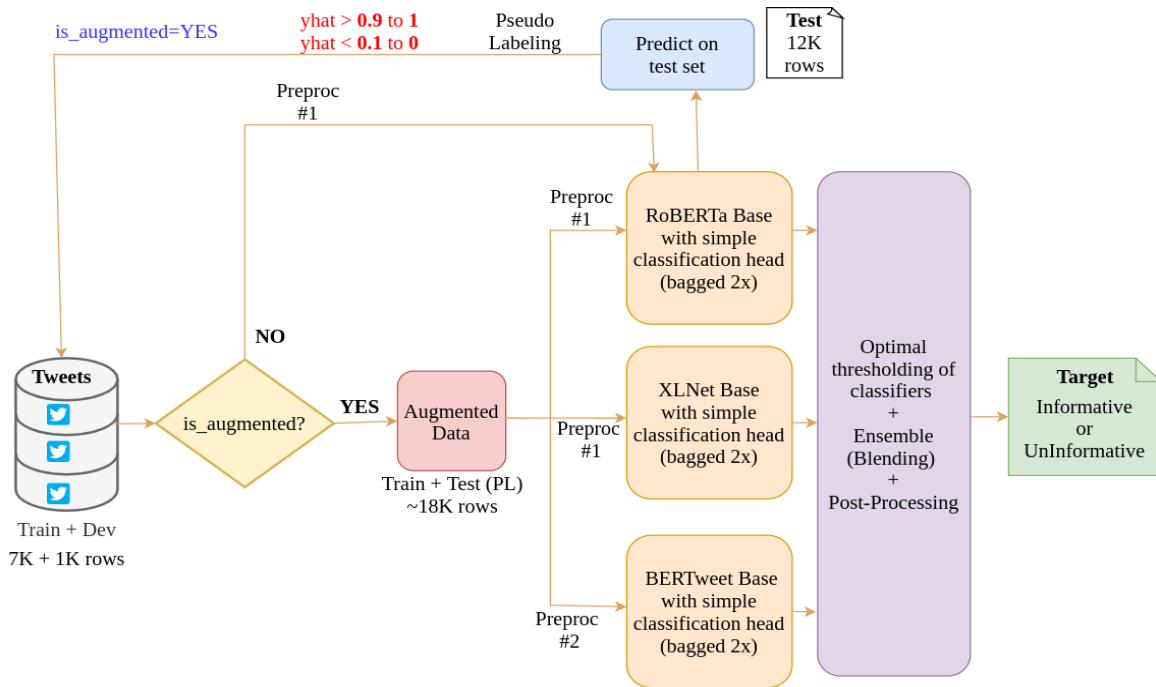


Figure 2: Our proposed model architecture. A RoBERTa model does the classification on the 12K test-set, while being trained using 7K train-set. Later, 11K most confident predictions are appended to the train-set. The new concatenated data is fed to the ensemble models leading to better generalization and improved model performance.

and 0.9011 on the development and test set respectively.

## 2 Related Work

Recently, research has started to investigate the use of deep learning in the area of disaster response. For instance, Caragea et al. (2016) use CNN to detect informative messages in data from flood-related disasters and report significant improvements in performance over SVM and fully connected Neural Networks. Nguyen et al. (2017) use CNNs to capture the most salient n-gram information on situational awareness crisis data and note the improvements over conventional algorithms. Lazreg et al. (2016) use LSTM network to learn a model from crisis Tweets to generate snippets of information for summarizing the Tweets. Wang and Lillis (2019) classify crisis-related Tweets using ELMo contextual word embeddings, whereas Ma (2019) use a monolingual BERT-based model for Tweets classification problem in the disaster management field.

Text classification generally consists of two processes — an encoder that converts textual inputs to numerical representations and a classifier that estimates hidden relations between these representations and class labels. The text representations are generated using N-gram statistics (Wang and

Manning, 2012), word embeddings (Joulin et al., 2017; Wang et al., 2018). More recently, powerful pre-trained models for text representations, e.g. BERT (Devlin et al., 2018), have shown state-of-the-art performance on text classification tasks using only the simple classifier of a fully connected layer.

## 3 System Description

We formulate the task of identifying informativeness in Tweets as a binary text classification problem with INFORMATIVE and UNINFORMATIVE as the class names. As shown in Figure 2, the framework of our Informativeness classification model consists of three modules — Transformer and BERTweet ensemble learning, generalized augmentation via pseudo-labeling, and optimal thresholding via post-processing to adjust the distribution of class labels in target.

### 3.1 Data Preprocessing

The preprocessing pipeline consists of the following two strategies:

- **Preproc #1:** We lowercase the Texts and remove the Non-ascii letters, urls, @RT:[NAME], @[NAME]. Furthermore, we break apart common single tokens; Eg:

RoBERTa makes a single token for "...", so we convert all single [...] tokens into three [...] tokens. We split "!!!" in the same manner. All Transformer models use this preprocessing strategy.

- **Preproc #2:** We normalize the Texts using TweetTokenizer.<sup>2</sup> Some of the normalization steps are — Expand text contractions (“can’t” to “cannot”, “M” to “million”, etc), text normalization (“p . m .” to “p.m.”, etc). All BERTweet models use this preprocessing strategy.

### 3.2 Model

We train 6 models in total — 2 each of RoBERTa-base, XLNet-base-cased, and BERTweet-base respectively on a 5-fold setup to find the optimal epoch. Its performance is evaluated on the validation set after every epoch. Later, it is trained on the complete dataset.

#### 3.2.1 RoBERTa

The meaning of words vary subtly across different contexts, and RoBERTa generates contextualized word representations to capture the context-sensitive semantics of words (Liu et al., 2019). The use of word representations from RoBERTa results in the state-of-the-art performance in a wide variety of language understanding tasks. Given a sentence  $s$  consisting of  $n$  words  $\{w_1, \dots, w_n\}$ , RoBERTa model generates their contextualized representations  $\{\mathbf{v}_{s,w_1}^c, \dots, \mathbf{v}_{s,w_n}^c\}$ .

#### 3.2.2 XLNet

XLNet is an auto-regressive language model which is based on the transformer architecture with recurrence (Yang et al., 2019). It outputs the joint probability of a sequence of tokens. The training objective calculates the probability of a word token conditioned on all permutations of word tokens in a sentence, as opposed to just those to the left or to the right of the target token.

#### 3.2.3 BERTweet

It is the first public large-scale language model pre-trained for English Tweets that is trained using a 80 GB corpus of 850M English Tweets (Dat Quoc Nguyen and Nguyen, 2020). It uses the same architecture as BERT-base, which is trained with a Masked Language Modeling (MLM) objective

<sup>2</sup><https://github.com/VinAIResearch/BERTweet/blob/master/TweetNormalizer.py>

Parameter	Version 1	Version 2
Max Sequence Length	128	192
Epochs	4	4
Batch Size	16	16
Learning Rate	2e-5	3e-5
Optimizer	Adam	AdamW (0.01)
FGM	no	yes

Table 1: Training Hyperparameters. To perform bagging, Version 1 and Version 2 are used.

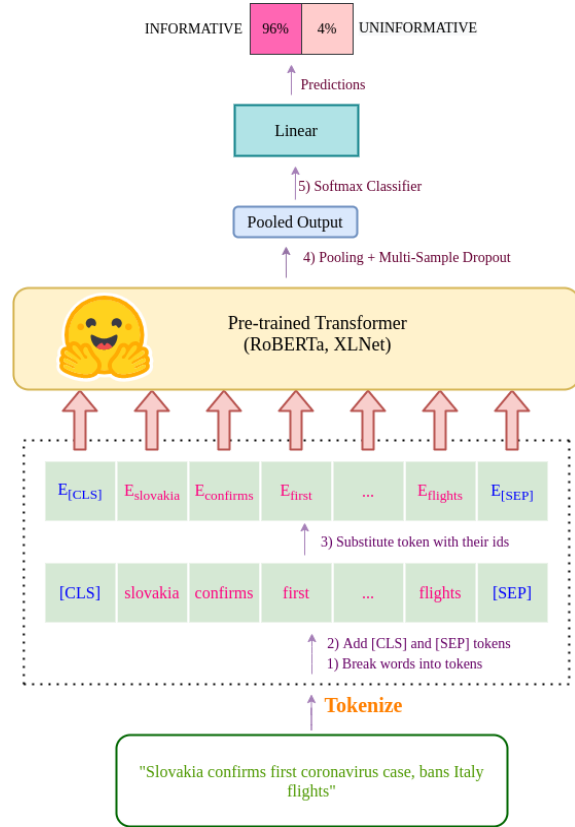


Figure 3: Pre-trained Transformer model architecture for informativeness classification.

(Devlin et al., 2018). BERTweet-base model claims to do better than RoBERTa-base and outperforms previous SOTA models on three downstream Tweet NLP tasks of POS tagging, NER, and text classification.

#### 3.2.4 Loss

Training Loss, Binary Cross Entropy Loss is defined as follows:

$$\text{BCE} = \begin{cases} -\log(f(s_1)) & \text{if } t_1 = 1 \\ -\log(1 - f(s_1)) & \text{if } t_1 = 0 \end{cases}$$

where,  $f()$  is the sigmoid function and  $s_1$  and  $t_1$  are the score and the ground truth label for the class  $C_1$ , which is also the class  $C_i$  in  $C$ .

Model	Without Augmentation			With Augmentation		
	Precision	Recall	$F_1$	Precision	Recall	$F_1$
RoBERTa-base <sub>1</sub>	0.8652	0.9386	0.9004	0.9619	0.8833	0.9209
RoBERTa-base <sub>2</sub>	0.8760	0.9280	0.9012	<b>0.9640</b>	0.8818	0.9211
XLNet-base <sub>1</sub>	0.8583	0.9364	0.8956	0.9619	0.8798	0.9190
XLNet-base <sub>2</sub>	0.8580	0.9343	0.8945	0.9619	0.8731	0.9153
BERTweet-base <sub>1</sub>	0.8630	0.9343	0.8973	0.9534	0.8858	0.9184
BERTweet-base <sub>2</sub>	0.8483	<b>0.9597</b>	0.9006	0.9449	0.8974	0.9206
<b>Ensemble</b>	0.8790	0.9386	0.9078	0.9513	0.8998	<b>0.9248</b>

Table 2: Results on the Development set.

Model	P	R	$F_1$
Baseline FastText	0.7730	0.7288	0.7503
RoBERTa-XLNet-BERTweet-Ensemble	0.8768	0.9269	0.9011

Table 3: Results on the Test set.

## 4 Experimentation

We rely on the dataset provided by the organizers to perform our experiments. Overall, there are a total of 10K Tweets split in the ratio of 70/10/20 parts into train/development/test set respectively. However, for the final evaluation, 12K unlabeled noisy Tweets are provided, out of which 2K test Tweets are the actual ones the models are finally evaluated upon.

### 4.1 Setup

We install the huggingface transformers library (Wolf et al., 2019). Pretrained RoBERTa-base and XLNet-base-based models with a single linear layer which is simply a feed-forward network that acts as a classification head, are used. Figure 3 shows a high-level overview of the architecture.

To speed up the training, sequence bucketing by removing unnecessary padding is employed (Khomenko et al., 2017). To improve the robustness of neural networks, and improving resistance to adversarial attacks, Fast Gradient Method (FGM) is used at the end of the Transformer models (Miyato et al., 2017). Multi-Sample Dropout (Inoue, 2019) is used when using dropout before the last layer with  $p = 0.5$ , as it seem to converge the loss faster. We pass the output of each dropout layer to a shared weight fc layer. Lastly, we take the average of the outputs from fc layer as the final output. Table 1 lists the chosen hyperparameters during model training.

For the BERTweet-base model, we normalize and tokenize Tweets with a CNN-Dropout layer

for the inference part.<sup>3</sup> Through a bunch of hyperparameters experimented from a finite sample space, we set the batch\_size = 16, epochs = 5, max\_seq\_len = 128, learning\_rate =  $3e - 6$ , along with Learning Rate Schedulers (Loshchilov and Hutter, 2017).

### 4.2 Augmentation

We augment the data carefully with the help of pseudo labeling which is the process of adding confident predicted test data to the training data. Inorder to make the Cross Validation (CV) scheme less over-optimistic, we exclude the pseudo labels from the validation folds. In other words, once we retrieve the labels, we run the Kfold technique on only the original data points with real labels, and then add the labels to train exactly at training time. That way, the CV isn’t biased by easy and artificially noiseless targets. It is augmented using the criteria:

$$y_{\text{new}}^{\hat{}} = \begin{cases} 1 & \text{if } y_{\text{roberta}}^{\hat{}} \geq 0.9 \\ 0 & \text{if } y_{\text{roberta}}^{\hat{}} < 0.1 \end{cases}$$

where,  $y_{\text{roberta}}^{\hat{}}$  is the meta-prediction on the 12K test-set using RoBERTa-base and  $y_{\text{new}}^{\hat{}}$  is the new label associated with it. These are then concatenated back to the train set, making an augmented data of 18.915K Tweets to develop the final model. In other words, 11.915K out of 12K Tweets in the test-set are identified as confident predictions after pseudo labeling. The thresholds are decided based on several optimization ranges so as to maximize the  $F_1$  score on the holdout development set.

### 4.3 Post-Processing

The idea here is to make the distribution of labels in development/test set to match corresponding distribution of labels in the train set so as to maintain the class ratio. Hence, the probabilities from all the 6 models are added and a majority voting cutoff value of 4 is found out by fine-tuning that maximize the  $F_1$  score on the holdout development set.

$$p = \sum_{i=1}^6 \vec{p}_i$$

$$p_{\text{out}} = \begin{cases} 1 & \text{if } p \geq 4 \\ 0 & \text{if } p < 4 \end{cases}$$

where,  $\vec{p}_i$  is the probability vector calculated by the 6 models  $i \in \{1, \dots, 6\}$ .  $p$  is the ensemble output, whereas  $p_{\text{out}}$  is the final prediction.

<sup>3</sup><https://www.kaggle.com/christofhenkel/setup-tokenizer>

## 5 Results and Error Analysis

We conduct ablation analysis to compare the performance of our model variants. We evaluate the effect of contextual features by comparing our model with and without augmentation. Table 2 summarizes the performance on the development set.

Without the augmentation, we notice a situation of high Recall, low Precision. Our classifier thinks a lot of Tweets as belonging to the INFORMATIVE class. This likely leads to a higher number of False Positive measurements, and a lower overall accuracy. For the BERTweet-base<sub>2</sub> model that gives the highest recall, 81 False Positive and 19 False Negative cases are identified. Whereas with the augmentation, a situation of low Recall, high Precision is observed. This makes sense as the model has access to more positive training samples and is able to make better decisions. Our classifier is very picky, and does not think many Tweets are INFORMATIVE. For the RoBERTa-base<sub>2</sub> model that gives the highest Precision, 61 False Positive and 17 False Negative cases are identified. Ideally, in the real-world scenario, the high Recall case would be more favourable as we want the model to label everything that could potentially be an INFORMATIVE Tweet, because a human personnel will most likely then interpret these results.

Understandably, the fine-tuned RoBERTa model did outperform every other experimented models. Bagging the models also lead to lower variance and robust predictions. Table 3 shows the final results on the Test set, wherein our model improves the organizer’s baseline by 20%. The effect of augmentation in the final ensemble is drastic as the  $F_1$  score increases by about 1.87%. Moreover, the idea of summing the probabilities of single models while ensembling worked better in comparison to choosing the most common label after finding different cutoff points that maximize the  $F_1$  score of individual models.

The confusion matrix of our best model is as shown in Figure 4. We look through the examples where our model made misclassification, and summarize the patterns of these error examples along with their attention visualization (Vig, 2019).

- **Inaccurate interpretation of context:** In the sentence, “*Writing 101: don’t put 2 numbers side by side. The punctuation is easy to miss. I first read this as being 51,385 people have died in Ontario from Covid.*”, much of the attention weights are focused on the latter part.

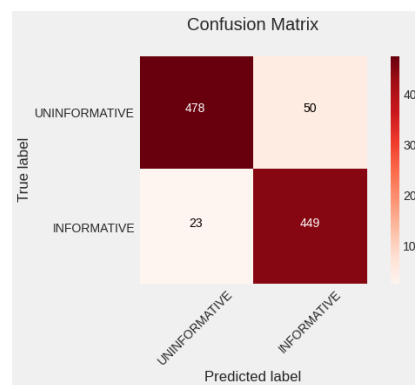


Figure 4: Confusion Matrix.

Our model may not capture this shift correctly given the long-distance dependency, which results in a False Positive prediction. See Figure 5 in Appendix A for attention visualization.

- **Misinformation due to ambiguity and subjectivity:** In the sentence, “*I just remember this news recently China keeping two sets of coronavirus pandemic numbers? “Leaked” infection numbers over 154,000; deaths approach 25,000*”, it could be well evident that some events may not really happen as the source of the news lacked credibility. This could have prompted inter-annotator disagreement. See Figure 6 in Appendix A for attention visualization.

## 6 Conclusion

We adopt an ensemble approach to reduce the variance of predictions and improve the model performance. The empirical results show the effectiveness and robustness of our model. Additionally, we perform a linguistic error analysis to gain insights into the model behavior. In the near future, we would like to combine user-related Tweet features (followers, friends, favorite counts) and Tweet-related meta-features (Retweets, creation date, sentiment) along with the contextual representation. Moreover, extending to multilingual Tweets (Chowdhury et al., 2020) is a potential future direction to pursue.

## Acknowledgements

We want to thank the anonymous reviewers for their time and comments which have helped make this paper and its contribution better.

## References

- Hila Becker, Mor Naaman, and Luis Gravano. 2010. Learning similarity metrics for event identification in social media. In *WSDM*, pages 291–300. ACM.
- Cornelia Caragea, A. Silvescu, and A. Tapia. 2016. Identifying informative messages in disaster events using convolutional neural networks. In *ICIS 2016*.
- Jishnu Ray Chowdhury, Cornelia Caragea, and Doina Caragea. 2020. Cross-lingual disaster-related multi-label tweet classification with manifold mixup. In *ACL (student)*, pages 292–298. Association for Computational Linguistics.
- Thanh Vu Dat Quoc Nguyen and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English Tweets. *arXiv preprint*, arXiv:2005.10200.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.
- Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. In *EMNLP/IJCNLP (1)*, pages 1161–1166. Association for Computational Linguistics.
- Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *ICLR (Poster)*.
- Hiroshi Inoue. 2019. Multi-sample dropout for accelerated training and better generalization. *CoRR*, abs/1905.09788.
- Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *EACL (2)*, pages 427–431. Association for Computational Linguistics.
- Viacheslav Khomenko, Oleg Shyshkov, Olga Radyvonenko, and Kostiantyn Bokhan. 2017. Accelerating recurrent neural network training using sequence bucketing and multi-gpu data parallelization. *CoRR*, abs/1708.05604.
- Mehdi Ben Lazreg, Morten Goodwin, and Ole-Christoffer Granmo. 2016. Information abstraction from crises related tweets using recurrent neural network. In *AIAI*, volume 475 of *IFIP Advances in Information and Communication Technology*, pages 441–452. Springer.
- 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. SGDR: stochastic gradient descent with warm restarts. In *ICLR (Poster)*. OpenReview.net.
- Guoqin Ma. 2019. Tweets classification with bert in the field of disaster management.
- Takeru Miyato, Andrew M. Dai, and Ian J. Goodfellow. 2017. Adversarial training methods for semi-supervised text classification. In *ICLR (Poster)*. OpenReview.net.
- 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*.
- Dat Tien Nguyen, Kamla Al-Mannai, Shafiq R. Joty, Hassan Sajjad, Muhammad Imran, and Prasenjit Mitra. 2017. Robust classification of crisis-related data on social networks using convolutional neural networks. In *ICWSM*, pages 632–635. AAAI Press.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008.
- Jesse Vig. 2019. A multiscale visualization of attention in the transformer model. In *ACL (3)*, pages 37–42. Association for Computational Linguistics.
- Congcong Wang and David Lillis. 2019. Classification for crisis-related tweets leveraging word embeddings and data augmentation. In *TREC*, volume 1250 of *NIST Special Publication*. National Institute of Standards and Technology (NIST).
- Guoyin Wang, Chunyuan Li, Wenlin Wang, Yizhe Zhang, Dinghan Shen, Xinyuan Zhang, Ricardo Henao, and Lawrence Carin. 2018. Joint embedding of words and labels for text classification. In *ACL (1)*, pages 2321–2331. Association for Computational Linguistics.
- Sida I. Wang and Christopher D. Manning. 2012. Baselines and bigrams: Simple, good sentiment and topic classification. In *ACL (2)*, pages 90–94. The Association for Computer Linguistics.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface’s transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.
- Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*, pages 5754–5764.

## A Appendix

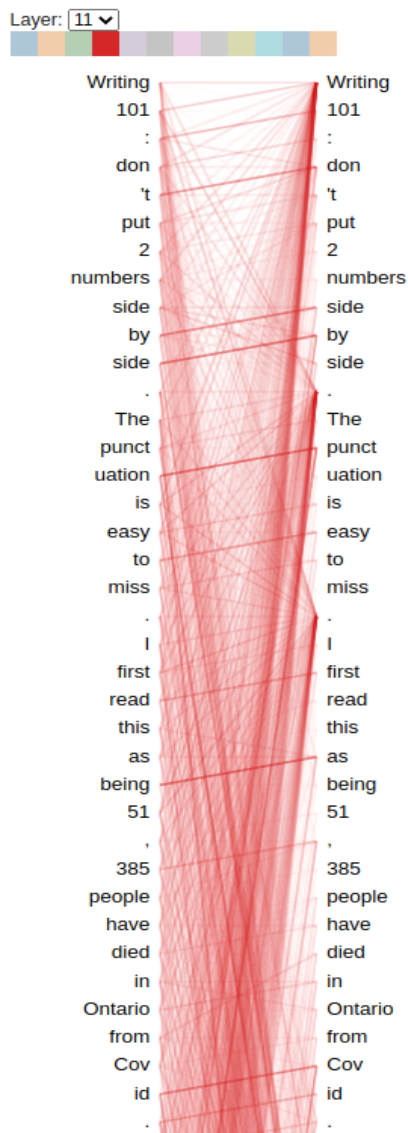


Figure 5: Attention-head view for the last layer of RoBERTa-base showing attention to other words predictive of word. In this pattern, attention seems to be directed to other words that are predictive of the source word, excluding the source word itself. In the example below, most of the attention from “id” is directed to “Cov”, whereas most of the attention from “Cov” is not focused on “id”.

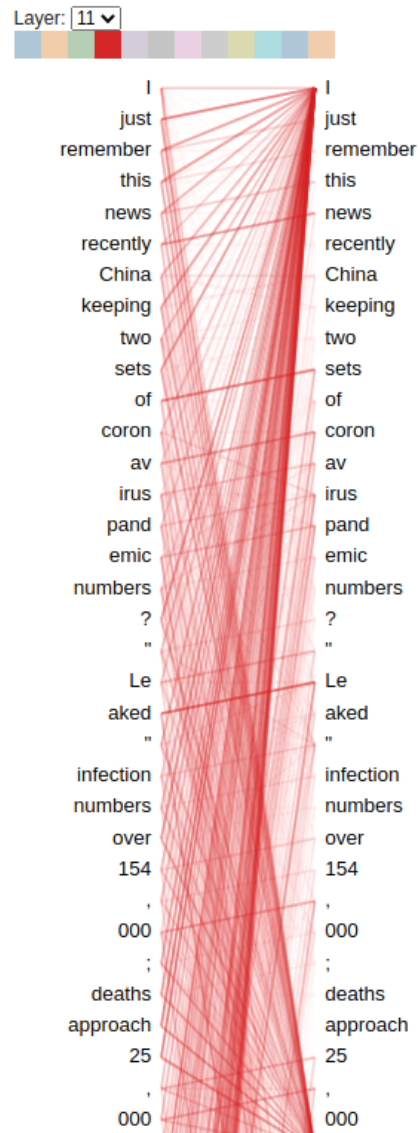


Figure 6: Attention-head view for the last layer of RoBERTa-base showing attention to either the previous or the next token in the sentence. For instance, most of the attention for “China” is directed to the previous word “I”. Considering a different example, most of the attention for “coron” is directed to the next word “irus” skipping “av” in between.