Identifying Fake News
Final Report

EECS 498-010
Group 17

Authors:
Joshua Fink, Alex Mahlon, Mathieu Noguier, and Bradford Williams

April 19, 2022

1. Introduction

Our group is attempting to solve a problem of increasing concern - fake news. **We aimed to create a tool that can accurately label a piece of writing as "fake".** We believe that this will be important due to the increase in propaganda and misinformation being dispersed not only in the U.S. but around the world. When we first thought of this project, we mostly thought of this as a much-needed tool to help fight against the misinformation being spread about COVID, the recent presidential election, and upcoming midterm elections. Now though, we can see how something that is able to identify fake news can make a difference in a theater of war due to the Russian invasion of Ukraine and how the Russian government is using misinformation to justify its invasion.

We have created a model that we hope can be used to fight against modern misinformation. At this point in time, it is clear how fast information can move on the internet and many social media sites are designed to be echo chambers. More so, sharing links amongst friends and family is easier than ever. Not to mention more and more people are getting their news from social media and less so from traditionally trusted news sources. Thus, it is imperative that a viable solution to the increasingly pervasive issue of fake news is found, especially due to the lack of development in this area in terms of data science and machine learning.

With its increasing popularity, there have been efforts to fight the proliferation of fake news. The issue though is that doing so takes an immense amount of time and energy. Not to mention a large workforce that would be marking articles as true or fake by hand.

Therefore, in this paper we present the data we found that gave us the most dynamic results, as well as the methods and algorithms we used to create a robust model given the information we can attain.

After experimenting with a few PyTorch classifiers, we found that our first rendition of a Bi-LSTM RNN worked well with our data; however, we believed that the model was overfitting. We then tried a BERT model, which resulted in better accuracy. Given that we do not have "a lot" of data, it was imperative that we do as much as we can with the given data; therefore, we worked vastly on feature extraction. In the end, we created a resilient model that can help evaluate news, even though it may be hindered in certain cases. We conclude with what implications arise from this model from an ethics standpoint and share our findings.

2. Related Work

There is not much related work on "fake news" as its popularity is still relatively new. We looked through a very in-depth survey that sifted through tweets to determine what was fake or not (Sharma et al., 2019). They worked with a dataset of tweets that also included the responses so they can take "into account their stance towards the content that is being discussed". These are some of the results that they gathered that most similarly reflect ours (does not include any metric relating to responses from other users):

Table 1: Classification accuracies by content-based methods. (Sharma et al., 2019)

Methods	Weibo [61]	Twitter [89]
Ott et al. 2011 - LIWC	66.06	62.13
Ott et al. 2011 - POS	74.77	70.34
Ott et al. 2011 - n-gram	84.76	80.69
Wang 2017	86.23	83.24
Qian et al. 2018	89.84	88.83

This survey also makes note of other interesting additional features that we did not take into

consideration, such as the inherent design of social media sites as "echo chambers" for ideas. The survey notes: "Quattrociocchi et al. 2016 similarly studied polarization by scientific v/s conspiracy narratives in Facebook users and observed that although both types of information were consumed similarly overall, 76.79% of all users who interacted on scientific pages and 91.53% of all users who interacted with conspiracy posts had 95% of their likes on either science or conspiracy posts" (Sharma et al., 2019).

There is another interesting study that looked at how people use deception to pass fake news as real. Some of their main features were Absurdity, Humor, Grammar, Negative Affect, and Punctuation) as well as recognizing sarcasm and irony. The authors found that a decent portion of the population eventually falls into the trap of thinking articles from *The Onion* or snippets from Stephen Colbert are entirely true and miss the subtle cues of humor. One thing that this study certainly lacks is a large dataset. Their initial iteration of it consisted of merely 360 articles. An interesting thing they did with this dataset, though, is that they would pair "each satirical piece… to a legitimate news article that was published in the same country, and as closely related in subject matter as possible" (Victoria Rubin et al., 2016). Their baseline model achieved 82% accuracy. While interesting to see a correlation between satire and fake news, this study did not address the more dangerous form of fake news that we address: one that is stern and assertive in tone and not humorous.

3. Data

Our main fake and real news dataset comes from Kaggle and contains 17,903 fake news articles and 21,192 real news articles. This was the most thorough dataset we could obtain. Attaining complete datasets was one of the biggest issues we came across and, unfortunately, does not allow our model to be as robust as possible. For example, this dataset is mainly centered around mid 2015-2018; because of this, our predictor likely would not perform well on articles pre-2015 and its accuracy will progressively decrease as time moves on. We believed that scraping the web and creating our own dataset would likely be too much given the timeframe in which this project needed to be completed. It would also not have been feasible for us to hand label enough examples during this semester to ensure a thorough and robust model. As a result, we assumed that every article labeled "True" is 100% true and the same for false articles. It's almost a certainty that there are false positives and false negatives, but it was not feasible for us to sift through every article and fact-check it.

Even if we were to sift through the dataset and manually verify each article's contents, there is a high probability that a group member will unknowingly use their own bias to access the content. Furthermore, journalists can frame articles in a way where they are not telling a lie, per se, but are not being neutral in regard to the content. This could bring about ethical issues and designating an article like this as true or false would likely not be agreed upon.

We made adjustments to the dataset in regard to pre-processing. For example, we removed "REUTERS", author names, and other pieces of information that could point to the source of the given article. Before doing this, the word "Reuters" was only found in "True" articles and led to extremely high levels of "accuracy". One Random Forest classifier that we ran before we adjusted the data resulted in 99% accuracy, which obviously raises questions about overfitting. "Reuters" is still featured in the word cloud provided in *Figure 1*, but that is only because the word cloud represents the data before pre-processing.

An interesting thing we found is how prevalent Getty Images are in the fake dataset, as you can see in Figure 1. At first, this was alarming to the group; however, it is our belief that reputable news sources have the resources to buy images through those on the frontline, whereas fake news producers likely do not have the funds to acquire images and use ones that are free to use instead. As our models do not use the images themselves, "Getty Images" is likely obtained from the caption, whereas reputable sites most likely place the name of the photographer they bought it from or no one if the photographer works for that media outlet.

Our exploratory analysis found no differences in the summary statistics for reading comprehension levels or overall TextBlob sentiment polarity. Though there obviously are more factors to explore, when overall vocabulary level and polarity scores are virtually the same (see *Table 2* and *Table 3*), having success to that extreme with a simple Random Forest Classifier raises many red flags. This is illustrated by the prominence of "Reuters" in the word cloud generated below in *Figure 2*.

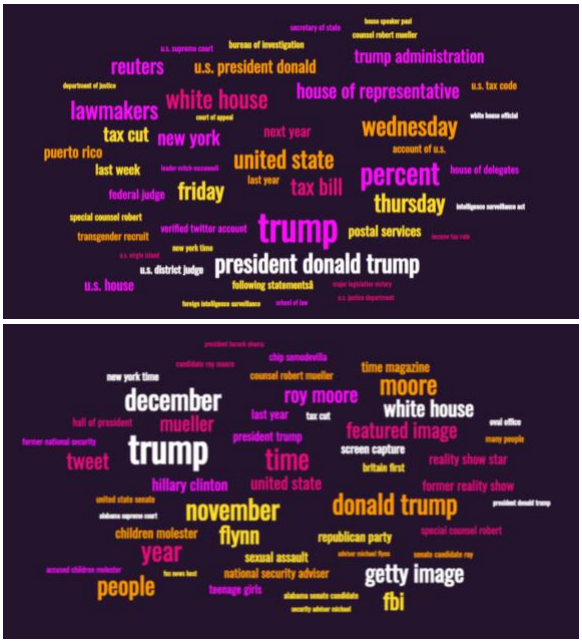
Table 2: Vocabulary Level Summary Statistics (Normals for Both True and Fake Samples). (Bisaillon, 2020)

	Dale-Chall Score	Flesch Reading Ease	Gunning-Fog
True	10.84	30.92	19.98
Fake	11.09	40.36	17.08

Table 3: TextBlob Sentiment Analysis (Bisaillon, 2020)

	Mean	Median
True	0.05	0.041
Fake	0.03	0.02

Figure 1: WordCloud from True.csv (top) and False.csv (bottom) (Bisaillon, 2020), "worldnews" Category



Though we decided to pull away from the original Kaggle dataset (Bisaillon, 2020) for a different dataset (Kaggle, 2018), we think that data right now is still causing issues. In the coming week, one way we are looking to enhance the model is via Stratified K-Fold Cross-Validation.

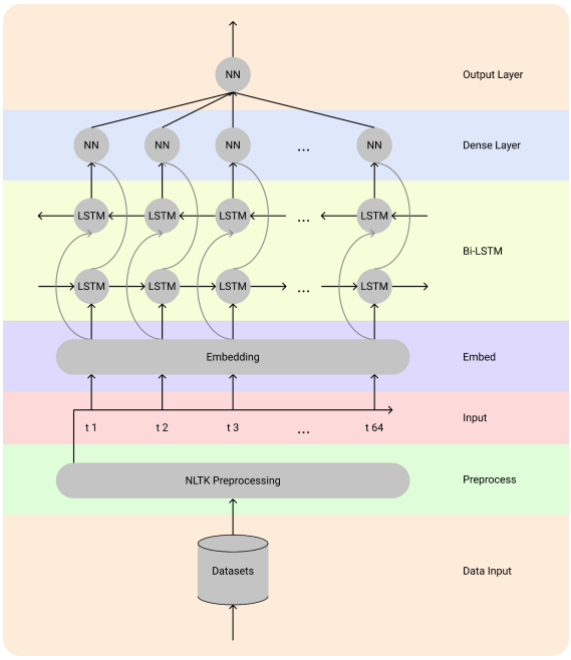
4. Methods & Algorithms

4.1 First Relatively Robust Model: Bi-LSTM RNN

The first fake news classifier was based on a bi-direction long short-term memory recurrent neural network (Bi-LSTM RNN). A Bi-LSTM

RNN was chosen as they are very performant in detecting anomalies in sequential data. The model has four main layers: an input layer, a Bi-LSTM RNN, a 64-node neural network layer, and a single node neural network output. The model's architecture can be seen below in *Figure 2*.

Figure 2: Bi-LSTM RNN Classifier Architecture



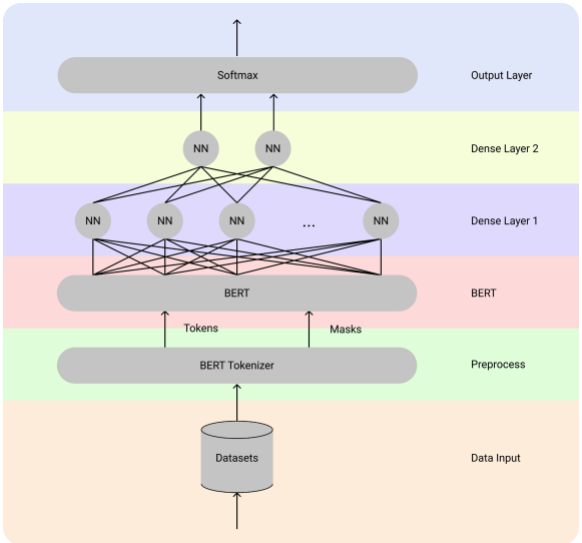
The bidirectional layer is composed of two, 64 cell LSTM RNNs. The first LSTM RNN layer receives the input forward and the second receives the input backward. The dense NN layer is composed of 64 neurons. These neurons are fully connected to the single-neuron output layer. The classifier uses binary cross-entropy as the loss function and Adam as the optimizer.

The data needs to be processed before it is run through the network. For our feature extraction and modeling approach, the NLTK library is used to preprocess the inputs. First, all stop words are removed from the input text. Then, the text is run through a stemmer, where each token has any unnecessary suffixes removed. The tokens are also set to lowercase in this step. All stemmed tokens are then collected in a corpus data structure. Afterward, the script uses the corpus to encode the preprocessed text articles into integer vectors.

4.2 Final Model: BERT Base

The current model is based on Google's bidirectional encoder representations from transformers (BERT) technique. BERT often offers better performance than other language classification legacy methods. The BERT-based classifier has 5 main layers: an input layer, a BERT layer, a 512 node neural network layer, a 2 node neural network layer, and a softmax layer. This architecture can be seen below in *Figure 3*.

Figure 3: BERT Classifier Architecture



The current classifier uses the BERT Base model. BERT Base is the smaller of the two original BERT models. BERT Base has 12 encoders, 12 bidirectional attention heads, and a 768-dimension output. This model was chosen as it offers similar performance with significantly less resources to train. Further, unlike the Bi-LSTM RNN used in the first

model, BERT is pre-trained.

The data also needs to be preprocessed before it is used by the BERT model. However, this step is simpler than the preprocessing required by the Bi-LSTM RNN-based model. The Transformers library from Hugging Face comes with a BERT-specific tokenizer to process the input data. Preprocessing the data is as easy as feeding the tokenizer the input data. Then, this data can be used to train and evaluate the model.

5. Results and Discussion

For long texts, Bi-LSTM and BERT are both models classically used for text classification. However, the unfortunate reality of these models is that when it comes to interpretability, they are effectively black boxes. Though we can look at summary statistics, we cannot determine what features the models specifically extract to make the predictions.

According to one study from the University of Calgary, Bi-LSTM RNN-based classifiers can reach high accuracies on combined corpora (Khan et al., 2021). Our Bi-LSTM RNN is performing adequately for the fake news classification. However, we suspect that we are quickly overfitting the training data. *Figure 4* shows the training statistics from our Bi-LSTM RNN. Here, you can see that validation loss continues to increase rapidly after epoch 3, a clear sign that the model is overfitting.

Table 4 shows the testing statistics at the best epoch. Epoch 3 was chosen as it was right before the model began to overfit. Here we see that the classifier has high overall performance metrics. However, the fact that after only epoch 3 the data overfits is rather alarming. For that reason, we began to look at other models.

Figure 4: Training Summary for the Bi-LSTM RNN on the Kaggle Fake News Community Competition Dataset (Kaggle, 2018).

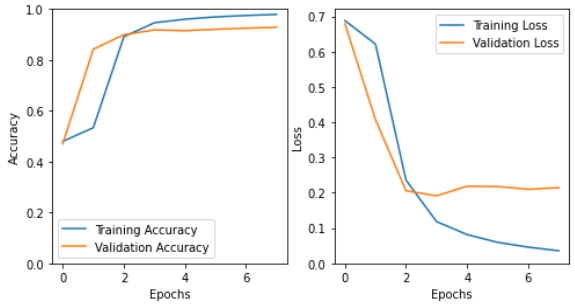


Table 4: Testing statistics for the Bi-LSTM RNN at epoch #3 on the Kaggle Fake News Community Competition Dataset (Kaggle, 2018).

Statistic	Value
Accuracy	0.85
Precision	0.86
Recall	0.85
F1	0.85

From the same University of Calgary study, BERT-based classifiers performed the best of all the compared classifiers. Similarly, our BERT based classifier outperforms our Bi-LSTM RNN based classifier in most regards. *Figure 5* shows the training statistics from the BERT based classifier. There is little indication of overfitting, thus the classifier was stopped when the validation accuracy began to plateau. Loss could be minimized by training the model for longer, however, more training did not increase the overall testing performance of the model.

Originally, the BERT model was fine-tuned to detect fake news. The retraining was limited to 3 epochs to reduce the chances of overfitting and

the risk of ruining the pretrained weights. Overall, pretraining had little effect in increasing the scores of the classifier. Additionally, it was very expensive to retrain the BERT model. Thus, fine-tuning was not used in the final BERT based model.

Over the course of training, it is also apparent that the model has a higher validation accuracy than training accuracy. This is an effect of the high dropout rate of the model. When training, the model disables many neurons. This makes it artificially harder for the model to make accurate predictions. However, dropout decreases the overall reliance on specific paths in the neural network layers. It also reduces the likelihood of overfitting as a result.

Table 5 below shows the testing statistics at the model's best epoch. For this model, epoch 10 was chosen. This model performed slightly higher than the Bi-LSTM RNN based classifier. Further, when tested on separate datasets, the BERT based model also outperformed the Bi-LSTM RNN based model. Thus, the BERT based classifier appears to generalize better as well.

Figure 5: Training Summary for the BERT Classifier on the Kaggle Fake News Community Competition Dataset (Kaggle, 2018).

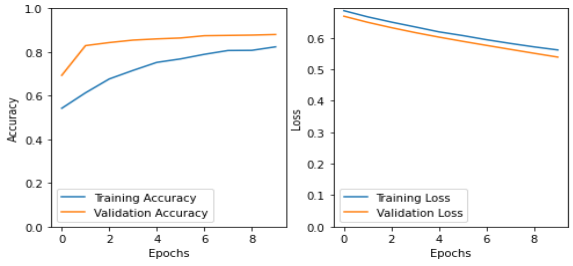


Table 5: Various testing statistics for the BERT Classifier at epoch #10 on the Kaggle Fake News Community Competition Dataset (Kaggle, 2018).

Statistic	Value
Accuracy	0.86
Precision	0.87
Recall	0.86
F1	0.86

6. Ethical Implications

6.1 Our Definition of Truthfulness

As a group, we did not feel it was appropriate for us to define truthfulness. People have their own "truths" which can easily be seen in a religious context (believing in a God could be how someone truly sees the world while someone else may not believe that whatsoever). Therefore, the definition of truthfulness that most accurately reflects our project is that of journalistic integrity. By this, we mean that our own inherent biases are not reflected whatsoever in the dataset or model, only the content of each true/fake article in our dataset is. Therefore, it is the cumulative journalistic integrity of each author from the true dataset that has become our de facto definition of truthfulness.

It would be nearly impossible to address any inherent biases that a given author exhibited in a given article. We do not have the workforce to verify every article as true and unbiased. Even if we were able to do this, it is possible that our own inherent biases are reflected in what we deem as a biased article or not. The best way we could address this is by finding neutral or at least perceived neutral news sources for both true and fake. This could be solved by limiting the number of articles that come from perceived

partisan news sources such as Fox News and MSNBC.

Even then, there will be people who fundamentally disagree with this method as well as every other. This isn't to say we aren't responsible for this, but a disclaimer. By using the Kaggle dataset, which in turn originated from the University of Victoria in Canada, we restrict our input to the analysis rather than defining this input to avoid this ethical issue.

6.2 Censorship

Once the model was complete there was the interesting question of what to do with it. Do you use that information provided from the model to take the article off the internet, not allow it to be shared on social media, or sent online to friends and family? We know the damage that fake news can cause and how powerful it can be, but barring it from being shared seems questionable. Slander and libel laws already protect individuals from personalized attacks and freedom of speech is one of the most highly regarded rights in America.

One strong implication is that with continued work, one could pinpoint specific accounts or news sources that are responsible for distributing mostly false information. However, it is an ongoing discussion if barring those users from posting violates their rights; as a group, we are unsure of how to address this ethical implication.

6.3 Misuse

Being able to score an article on truthfulness and label it as fake news seems great, but similar to the definition of truthfulness, who is in control of the model is also complicated. Russia recently passed a new law that can put anyone sharing "fake news" in prison for up to 15 years. The issue is that the Kremlin defines fake news as anything that openly criticizes their government or President Vladimir Putin. This goes beyond the issue of defining truth. The Kremlin could use the reputation of a credible model and manipulate it behind the scenes so that a true, anti-Kremlin article gets labeled as "Fake" when a user tries to use the tool. One solution would be to use an open-source model, which is popular in the crypto space and online gambling (to ensure fairness).

7. Future Extensions

If we were to continue working on this project, we would develop a website where the user can paste/submit an article and output an accuracy prediction from our model in a clear and intuitive fashion. For instance, we explored showing our truthfulness score as a traffic light, speedometer, compass, and a simple spectrum. Unfortunately, time caught up with our initial ambitions for this project and that part remains incomplete.

It would be interesting to see how much we could improve this model in the future by adding features relating to humor, a dataset containing true and false tweets, branching into replies and comments to extract sentiment, or even feeding history textbooks to a single model. This seemingly would create a much more robust model than the one we produced, and it would be interesting to see how a model like this would deal with issues such as journalists framing stories in different ways, since, in this case, the model knows facts from fiction based on textbooks.

In the week leading up to the final presentation, we also stumbled upon a model known as CogLTX, which specifically looks at BERT models with long texts. Rather than purely splitting the texts into blocks via a sliding window approach and feeding them into the model, the CogLTX framework employs a judge

and a reasoner model to better simulate how the human mind relates portions of texts together. The text first uses a MemRecall algorithm that employs multi-step reasoning to recall key phrases, then subsequently trains a few BERT models later in the pipeline. This model exhibited substantial performance improvements on known datasets, indicating that this model may translate well (Ding et al., 2020).

8. Conclusion

While we were able to make a model that found and outputted a predicted accuracy for a given article's truthfulness, we are aware that we had a limited dataset that has not made the model as robust as it can be for a topic of this complexity. We still cannot provide a metric that incorporates how diverse people's definition of truth is and what they are willing to believe. In all, solving the fake news crisis is a very complex issue and this model only solves it at a rudimentary level. This project introduced us to how many variables are needed when modeling real life, and we did not realize how complex this topic is when we first embarked on this project.

9. References

Bisaillon, C. (2020, March 26). Fake and real news dataset. Kaggle. Retrieved March 10, 2022, from https://www.kaggle.com/clementbisaillon/fake-and-real-news-dataset/version/1

Ding Ming, Zhou Chang, Yang Hongxia, & Tang, Jie. (2020). CogLTX: Applying BERT to Long Texts. In Advances in Neural Information Processing Systems (pp. 12792–12804). Curran Associates, Inc..

Fake News. Kaggle. (2018). Retrieved March 10, 2022, from https://www.kaggle.com/c/fake-news/data

Junaed Younus Khan, Tawkat Islam Khondaker, Sadia Afroz, Gias Uddin, Anindya Iqbala. (2021, March 24). *A benchmark study of machine learning models for online fake news detection*.Machine Learning with Applications. Retrieved February 3, 2022, from https://www.sciencedirect.com/science/article/pii/S266682702100013X

Karishma Sharma, Feng Qian, He Jiang, Natali Ruchansky, Ming Zhang, & Yan Liu. (2019). Combating Fake News: A Survey on Identification and Mitigation Techniques. *ACM Trans. Intell. Syst. Technol.*, *10*(3). doi:10.1145/3305260

Victoria Rubin, Niall Conroy, Yimin Chen, and Sarah Cornwell. (2016). Fake News or Truth? Using Satirical Cues to Detect Potentially Misleading News. In Proceedings of the Second Workshop on Computational Approaches to Deception Detection, pages 7–17, San Diego, California. Association for Computational Linguistics