HIGH SCHOOL EDITION
Journal of Student Research
Volume 12 Issue 4 (2023)

High Performing Explanatory Fake News Classification on Longer Texts

Chelsea She[1] and Clayton Greenberg[#]

[1]Adlai E. Stevenson High School
[#]Advisor

ABSTRACT

After misinformation became prevalent in 2020, the research community started prioritizing creating state of the art (SOTA) fake news detectors. However, these models did little in changing user attitudes towards misinformation. Therefore, we try to increase trust between users and AI fake news detectors by implementing an explanatory moderator. We started with two research questions: (1) can long texts like normal news articles perform well in current fake news detectors meant for short texts, and (2) can we create a fake news detector that can achieve comparable high performances to SOTA fake news detectors while representing its classifications in explainable visualizations. To fulfill our first research question, we picked WELFake, a dataset containing news articles from four different news platforms. In order to create a comparable, SOTA fake news detector performance, we ran preliminary models of Majority Class Baseline, Random Forest Classifier with bag of words, and the third-place model from the AAAI 2021 Shared Task: COVID-19 Fake News Detection in English competition with WELFake. Lastly, we fulfilled our second research question by making a manually fine-tuned BERT model to access attention masks that we could visualize through BertViz. Our manually fine-tuned BERT model outperformed our comparable, SOTA Two-Fold Four-Model ensemble with a 99.99% test accuracy. We made conclusions that current SOTA fake news detectors made for short texts can perform the same level of accuracy with long texts and explanatory fake news detectors can be comparable to current SOTA models.

1. Introduction

With fake news becoming more prevalent in many social and information platforms, using AI models to combat misinformation has emerged as a very popular topic within the research community. From human centered countermeasures like warning or flagging posts to creating fake news detectors, many new technologies and solutions are developed to help users avoid misinformation. However, without a solution to effectively create trust between users and fake news detectors, humans will still fall vulnerable to believing fake news. A paper evaluating user preferences to various countermeasures revealed that the warning flag providing a link for further information was most attractive to users because it provided further explanations of its classification (Kirchner and Reuter, 2020). In addition, when the paper evaluated the effectiveness of each method, the warning flag was also the most effective in letting users be aware of the misinformation content (Kirchner and Reuter, 2020). Users perceive explanations of fake news detectors as crucial to their trust in changing their opinions and detecting misinformation.

Unfortunately, current SOTA news detectors do not provide interpretable explanations. This was mainly due to the majority of the highly accurate fake news detectors using Deep Learning methods, that contain processes not understandable for humans to interpret. Therefore, we addressed this need for more explanatory fake news detectors by asking the research question: can we create a fake news detector that can achieve the same high performance of SOTA fake news detectors while representing its classifications in explainable visualizations.

In addition, recent research on these fake news detectors was heavily concentrated towards short texts, similar to those on social media platforms. Detection of long text news articles on news sites like Reuters and New York

Times are still relatively under-researched, even though those platforms have just as great of an effect towards current day information distribution and extremism. Therefore, we had another research question asking if long texts like normal news articles can perform just as well in current fake news detectors meant for short texts. This was achieved by utilizing WELFake, a news article fake news classification dataset that contains some of the longest news articles in a dataset, and a SOTA short text classification model from Verma et al. (2021).

2. Background

While majority of the current papers presented models that solely classify fake news, a few papers had already attempted creating explanatory fake news detectors.

For instance, Szczepański et al. (2021) created an add-on model to current existing fake news detectors for the sole function of creating linear, understandable explanations to the detector's conclusions. After adding the explanatory model on BiLSTM and BERT, they maintained the same high precision, recall, and f1 scores above 90%. Their explanatory model's architecture contains 2 parts: LIME and Anchor. Under the assumption that every complex model is linear at the local scale, LIME samples each explained prediction and alters them to train an inherently interpretable linear model (Szczepański et al., 2021). Anchors is a model-agnostic explanation algorithm based on 'if-then' rules, where an "anchor" is a rule applied to each local prediction and will not be affected by any other feature changes (Szczepański et al., 2021). This allows Anchor to make constant predictions. With their two explanatory model extensions, Szczepański et al. (2021) was able to create a plug-in explanatory model that can be applied to any of the already existing models and successfully give easily understandable explanations. However, some limitations of their study include LIME and Anchor explanations overlapping whereas using various techniques could be further studied to highlight different patterns, Anchors sometimes not able to provide an explanation, and only analyzing short headlines over full news articles.

Another paper from Shu et al. (2019) created an explanatory model called dEFEND. The dEFEND model will find the most check-worthy sentences in news and comments that help with fake news detection. There are 4 parts of their model: (1) news content encoder (including word encoder and sentence encoder), (2) user comment encoder, (3) sentence-comment co-attention component, and (4) fake news prediction component (Shu et al., 2019). dEFEND used a dataset called FakeNewsNet, which included social engagement features to each datapoint, enabling their access to user comments and engagement (Shu et al., 2019). Their results were also promising: dEFEND not only outperformed current leading detectors during that time in accuracy, precision, F1, and Recall, but also was proven to be more explainable after being tested against HPA-BLSTM by recruiting Amazon Mechanical Turk workers to choose between the two lists of explained comments/sentences (Shu et al., 2019). HPA-BLSTM is a neural network model that learns news representation through a hierarchical attention network on word-level, post-level, and sub-event level of user engagements on social media (Shu et al., 2019). This allowed them to use HPA-BLSTM as a baseline for user comment explainability since they can learn attention weights for news sentences and user comments, respectively (Shu et al., 2019). However, some limitations of dEFEND that we noticed included how low accuracies were compared to the present updated fake news detectors, and how heavily dependent it was on comments to provide explanations,

However, the model most similar to our desired model was the Hierarchical Multihead Attentive Network for Fact-Checking (MAC) model. MAC's framework consists of four main components: (1) embedding layer, (2) multi-head word attention layer, (3) multi-head document attention layer and (4) output layer (Vo and Lee, 2021). Vo and Lee (2021) compared their MAC model's performance to two different type of baseline model types: models using only the claim's text (includes BERT, LSTM-Avg, LSTM-Last, and TextCNN) and models using both claim and article text (includes HAN, NSMN, and DeClare). Out of all the models, MAC outperformed all models for true and fake news alike on Snopes and Politico datasets (Vo and Lee, 2021). Some limitations of this model were that it

was tested on false claim datasets instead of normal news article datasets, making it dependent on claims. Its performance, although improved from other evidence-aware fake news detection models, is still a lot lower than current SOTA fake news detectors.

Overall, three common themes were seen in current explanatory fake news detectors: worse performance compared to non-explanatory fake news detectors, dependence of other features not available on normal news article cites (like claims or user comments), and not interpreting the full texts of news articles. This paper will attempt to create a model solving all three of these shortcomings.

3. Dataset

3.1 Data source

There currently exist a lot of fake news detection datasets. These datasets are used for a variety of purposes, including fake news classification, stance classification, false short claim verification, rumor verification, modeling social engagement, and modeling explanations. For the purposes of this paper, we decided to look for fake news classification datasets that labeled a broad domain of news as either real or fake. We referenced the Kaggle website to acquire popular, openly accessible fake news classification datasets. Top contenders we considered included COVID-19 Fake News Dataset (Patwa et al., 2021), FakeNewsNet (Shu et al., 2020), Fake and real news dataset (Ahmed et al., 2018), and WELFake (Verma et al., 2021).

The COVID-19 Fake News Dataset was from the Constraint@AAAI - COVID19 Fake News Detection in English shared task. Although it was a heavily referenced fake news classification dataset, it extracted data from social media platforms like Facebook and Twitter, so the texts were shorter than optimal for the goal of highlighting specific sections within a longer text. Its domain was also limited to COVID-19 related information, which was too specific for our holistic study on fake news classification. FakeNewsNet consists of both Politifact and Gossipcop data, which allowed a broader range of misinformation in both entertainment and political domains (Shu et al., 2020). However, since it targeted modeling social engagement dataset in addition to fake news classification, it only included the source URL, instead of the full text, meaning we would need to extract each datapoint information with an API. In contrast, both Fake and real news dataset and WELFake already extracted the full text of each news article, making them easier to use compared to FakeNewsNet. The defining factor that led us to use WELFake was its size compared to Fake and real news dataset. Fake and real news had 20826 real news and 17903 fake news compared to WELFake having 35028 real news and 37106 fake news. WELFake authors merged multiple popular news datasets like Kaggle, McIntire, Reuters, BuzzFeed Political in order to prevent overfitting of similar domains of information and compile a larger dataset (Verma el. al, 2021). With a larger dataset to train with and a more even distribution of real and fake news, WELFake was the dataset we decided to use.

3.2 Data Analysis

To have a better understanding of WELFake before we implement it into our model, we did some exploratory analyses on the dataset.

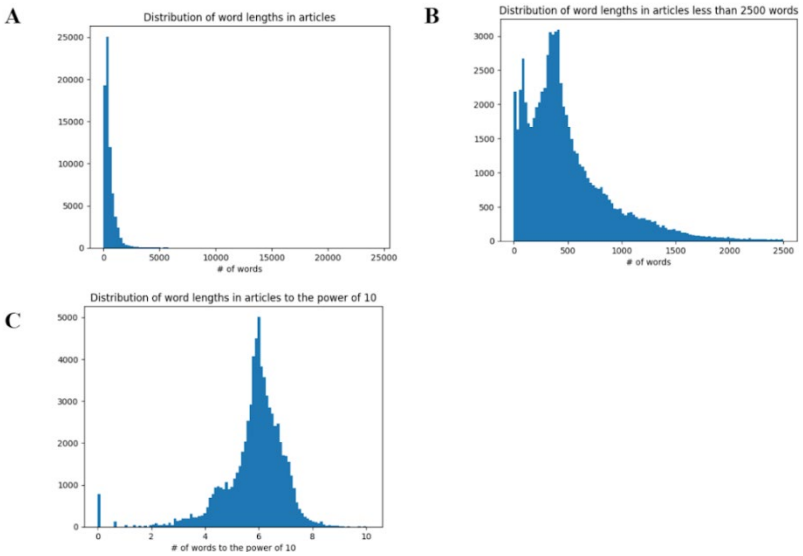

Figure 1. Distribution of WELFake article word lengths. As seen in Figure 1A, maximum word lengths reach up to 20000 words. As seen in figure 1B, most word lengths are between 0-1000 words. As seen in figure 1C, WELFake contains many news articles with large word lengths.

We first explored the distribution of word length in the article bodies. Our initial reaction was the dataset had much longer news articles than we thought. While most seem to fall between the 0-1000 word length, there were still a few that were as long as 24000 words. The top 5 longest news articles were all over 20000 words, with surprisingly one of them being in Russian, instead of English. Although our aim was to analyze full news articles, we would later learn that such long lengths of text needed much more computing power and would get truncated down to 200 words or less when we applied them in LLMs like BERT to obtain attention masks for an explanatory fake news detector. Another reaction we noticed was how 783 articles had 0 words in their article body. Even without a body, most of the articles did have a title, so we kept most of these articles for our model after merging the titles and article texts together. Interestingly, both the top 5 longest and shortest articles were all labeled as real news.

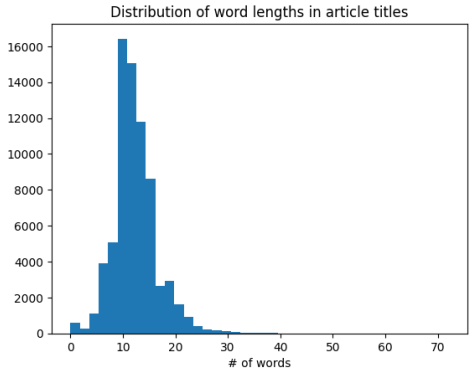

Figure 2. Distribution of WELFake article title word lengths

Next, we did a preliminary analysis on WELFake's article titles. Similar to the longer article lengths, titles were also abnormally longer than expected. The average title length was 12.17 while the longest title was 72 words. Interestingly, both the top 5 shortest and longest article titles were also all classified as real news. Just like how there were

some articles without a text body, there were 558 articles without a title. So, to ensure we had no datapoint with neither a title nor article text, we checked and eliminated any empty data prior to implementing the dataset to our model.

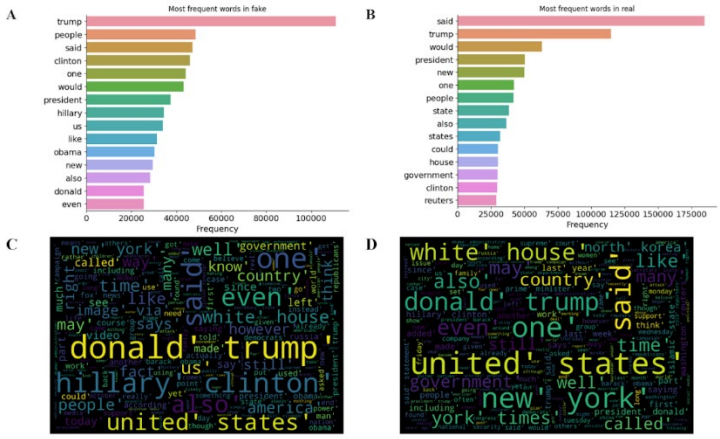

Figure 3. Frequency of words in fake and real news articles of WELFake. As seen in figure 3A and 3B, many words overlap between the fake and real news articles. As seen in figure 3C and 3D, prominent American political figures are frequently used.

After we preprocessed the text by lowercasing letters and eliminating stop words to every article, which is the article title and body merged together, we also looked at the frequency of the most common words used based on fake or real labeled news articles. Both labels had prominent American political proper nouns like Hillary Clinton, Donald Trump, White House, and president listed which reveals how WELFake might predominantly be an American political news dataset. Our exploration did not detect any robust differences in the distributions for real articles and fake articles in WELFake.

4. Models

4.1 Preliminary models

To ensure we have a model that maintains just as high of an accuracy to current SOTA models, we decided to implement three machine learning algorithms for text classification. We analyzed Majority Class Baseline, Random Forest Classifier over a bag of words, and an adapted Two-Fold Four Model ensemble from a Five-Fold Five-Model ensemble. The Five-Fold Five-Model Ensemble was created in Li's et al. (2021) third place paper in the AAAI 2021 Shared Task: COVID-19 Fake News Detection in English competition. For all models, we used the same training, validation, and test data from WELFake, splitting training and testing data 80-20 from WELFake, and then training data again with validation data 80-20. There were 46165 training data, 11542 validation data, and 14427 testing data. Due to confusion from WELFake labeling fake news as 1 and real news as 0, we switched the labels as fake news being 0 and real news being 1.

4.1.1 Majority Class Baseline

Wanting a comparable baseline that all classification models can be comparable to, we analyzed our test data with all false predictions, since there were more false labels than real labels in WELFake. Our results were 51.46% test accuracy, 70020 false negatives, and 0 false positives.

4.1.2 Random Forest Classifier with bag of words

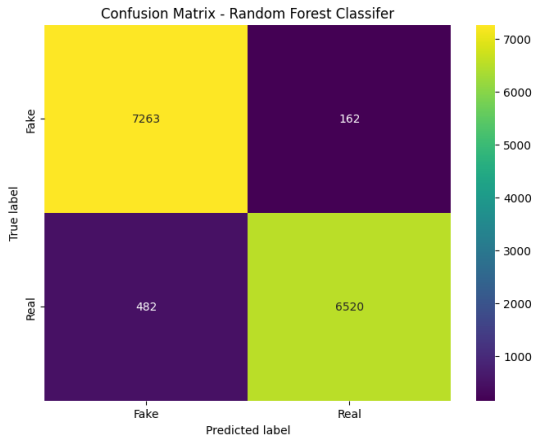

Figure 4. Results of Random Forest Classifier with bag of words

Our bag of words model considered the 1000 most frequent words in both our training and test data. Since Random Forest Classifier does not need validation data, we decided to train the model with both the universally split training and validation data. After feeding the vectors through a basic Random Forest Classifier, we surprisingly reached an accuracy score of 95.53%, precision score of 97.58%, Recall score of 93.12%, and F1 score of 95.29%. The distribution of errors were 162 false positives compared to 482 false negatives. We interpreted more false negatives as worse than having more false positives, as we believed there was a larger consequence invalidating real work as fake compared to not catching articles that were fake.

4.1.3 Two-fold Four-model Ensemble

Wanting to make sure we could get a higher precision than our CountVectorizer Random Forest Classifier model to have a comparable current SOTA fake news detector, we decided to use the third ranked, premade model from the Constraint@AAAI 2021 Shared Task: COVID-19 Fake News Detection in English competition (Li et al., 2021). In this paper, they proposed two models: Text-RNN and Text-transformers. While deep learning LSTM has already been proven to be effective, its shortcoming of depending on previous text contextualization made the researchers use a bidirectional LSTM model for Text-RNN (Li et al., 2021). For Text-transformers, they created a five fold structure that could be built in three ways: in a five-fold single model ensemble, five-fold five model ensemble, or five-fold five model ensemble in addition to a pseudo label algorithm (which augmented test data and used test data with 95% accuracy for training data due to their own extracted dataset being too small) (Li et al., 2021). All text transformer models Li et al. (2021) used were large language models, including BERT, RoBERTa, Ernie, X1-net, and Electra. However, a limitation the model had was that the shared task dataset was solely short text based on social media platforms like Twitter, making the model's accuracy uncertain for larger textual information like our news article dataset. It was interesting to see how well their models would work with WELFake.

Out of all the methods they explored, Li et al. (2021) achieved the best performance with their five-fold five model. Even though their pseudo label algorithm further improved their model's performance, we decided to not use their pseudo label algorithm because we did not have a shortage of data like they did with their much smaller external datasets. As we implemented their five-fold five model method, we also faced large consumption of storage and processing energy, due to the length of text we had to process compared to Li's et al. (2021) short texts. Therefore, we decided to eliminate the BERT model to create a five-fold four model configuration.

For our first attempt in using this model, we decided to not take the traditional approach of using the universal WELFake train, validation, and test datasets, but used Li's et al. (2021) already embedded training and validation datasets, only utilizing WELFake for the testing data. This was to help answer our first research question of whether models made and trained with short texts can perform just as well with longer texts. While our training losses and validation accuracies were similar to Li's et al. (2021) paper, which was expected since we trained with the exact data they had except for leaving out their pseudo and external datasets, our test results were very poor, generating a test accuracy of 47.72%. Although the model performed well classifying fake news, it performed poorly in classifying real news. This answered our first question by revealing models trained with short texts should not be used to classify long texts.

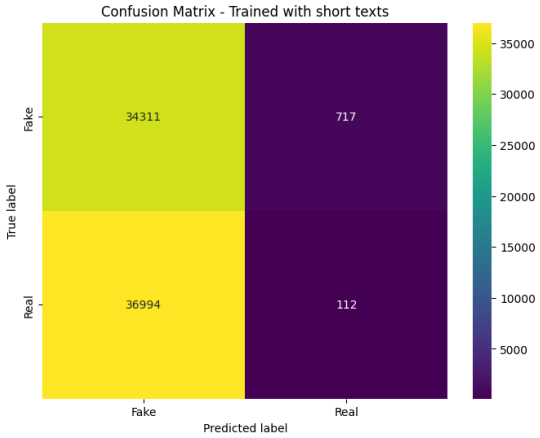

Figure 5. Results of Five-Fold Four Model Ensemble

However, we still wanted to see if the inaccuracies were due to the model architecture overall or because it was trained on a short text dataset. So instead of using training and validation data from the shared task, our second attempt in using the model architecture used the universally split train, validation, and testing WELFake data. Surprisingly, the model performed a lot better during training with the first epoch getting a training loss of 3.31%, accuracy of 99.96%, and F1 score of 99.96%. So instead of doing all five folds, we decided only two folds were necessary.

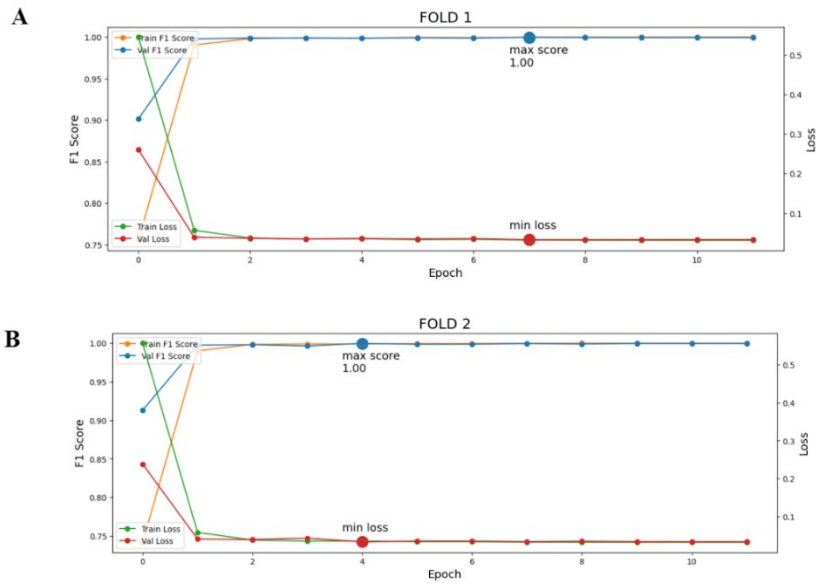

Figure 6. Training results for Two-Fold Four Model Ensemble. As seen in figure 6A and 6B, training losses and F1 scores were so close that it was unnecessary to do more folds.

Once our model was fully trained, we tested our model and got a test accuracy of 99.92% with only 4 false positives and 8 false negatives. Our precision was 99.94%, Recall was 99.88%, and F1 score was 99.99%. This was surprising, as the model trained on WELFake performed so well that it outperformed the original paper's results of 98.5%, 98.6%, 98.5%, and 98.5% accuracy, precision, recall, and F1 score, respectively. We concluded this happened due to classification being much easier if examined on a longer text compared to a shorter twitter post. This relieved our concerns about our first research question, as it proved that not only were high performing fake news detectors made for short texts able to be implemented on long news articles, but those same models might perform even better with long news articles.

In the end, we concluded that while it drastically decreases a SOTA's performance when training it with a different dataset, it performs very well if trained and tested with the same dataset. A SOTA model's architecture can perform well under any length of text. We will be comparing this model's performance with our actual model's performance to ensure we achieve the same standard performance in our explanatory model compared to other highly accurate models.

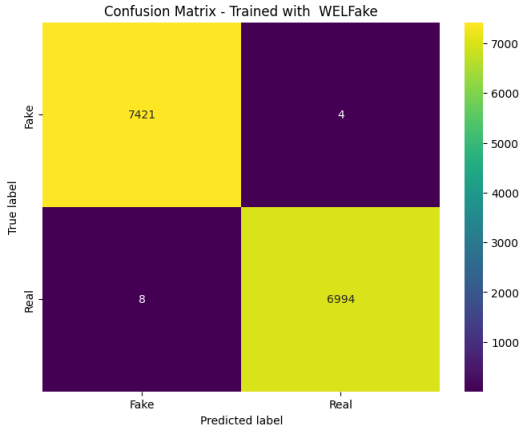

Figure 7. Test results for Two-Fold Four Model Ensemble.

4.2 Manually Fine-tuned BERT model

After creating comparable results, we started implementing our own model. In order to create an explanatory fake news detector, we needed an attention mask to explain the relationship each word had in comparison to their label of either real or fake. The current models that have attention masks were large language text transformers. Therefore, we decided to implement a manually fine-tuned BERT model to have access to each data's attention mask.

Before we can use our model, BERT needs texts to be preprocessed in a specific way. This includes adding special tokens like [CLS] and [SEP] to separate each datapoint, splitting up proper nouns to words in the BERT dictionary with special tokenization, and padding to maintain constant lengths of all data points. During this process, we also decided to attach the labels of each datapoint at the end of each article text to allow the model to train correctly. We did this by separating the labels from the actual article's text with an [SEP] tag. Due to how much computing energy it will take to process the full, long articles of WELFake, we decided to pad and truncate sequences to 200 tokens. Since most articles are much longer than 200 words, we decided to also truncate each article to 100 words prior to tokenizing and padding so that the label we added at the end of each article text would not get truncated during padding.

We used the same split training, validation, and testing data as our preliminary models. When training our model, we sampled our training and validation data randomly. In the end, we reached a surprising average training loss of 1%, validation loss of 0%, and validation accuracy of 100% after 1 epoch. Therefore, we thought it was unnecessary to have more than 1 epoch, since our model seemed well trained with WELFake after 1 epoch.

5. Results

In this section we highlight our model's results and the results of our visualizations that made our model explanatory.

5.1 Manually Fine-tuned BERT Model Results

When we tested our model, we decided to test the data sequentially. In the end, we had a 99.99% test accuracy with 0 false positives and 1 false negative. Precision score was 100%, Recall score was 99.98%, and F1 score was 99.99%. With such high results, we were able to fulfill the first part of our research question, achieving a model that can have just as high of an accuracy in comparison to SOTA models. Our high accuracies may have come from how easy WELFake is to classify, and testing if our model can perform just as comparatively well in other datasets might make this study even more conclusive.

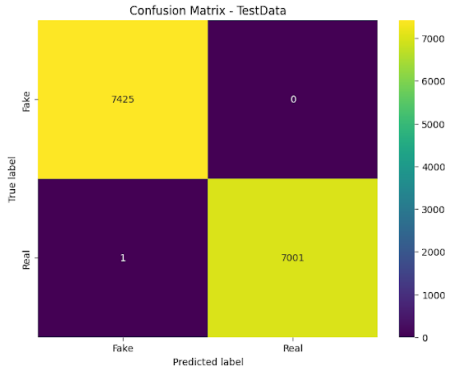

Figure 8. Results for manually fine-tuned BERT model

5.2 BertViz Visualization

To have a visual understanding of attention masks to create an understandable explanation, we decided to use BertViz, which is an open-source tool for visualizing multi-head self-attention in the BERT model (Vig, 2019).

BertViz is an open-source tool for visualizing self-attention in the BERT language representation model (Vig, 2019). It can visualize attention in three different modes: the attention-head level, the model level, and the neuron level (Vig, 2019). The attention-head view visualizes the attention patterns produced by one or more attention heads in a single transformer layer while model view gives visualizations for all transformer layers between texts (Vig, 2019). The neuron view visualizes the individual neurons in the query and key vectors and shows how they interact to produce attention scores (Vig, 2019). Out of the three levels, we decided to use attention-head level so we can see the relationship of every word towards the article's label more specifically.

In BertViz's attention-head level, it needs two sentences that are tokenized and separated into attention masks by token type ids. After trying to send full news articles of WELFake through BertViz, our code crashed due to overconsumption of computer storage. We decided to limit each article instead to only 100 characters for our baselines, and 28 words for our test articles.

We decided to first test the attention-head level model out with the baseline, unprocessed real and fake data points in WELFake as the first sentence, and their associated label numbers for the second sentence. After running the texts through BertViz's head model function, we decided to only look at Sentence B to Sentence A attention, since we wanted to see every relationship between the label and the article texts. Some common patterns we saw were more attention put towards punctuation, prepositions, and BERT's special [CLS] and [SEP] tokens.

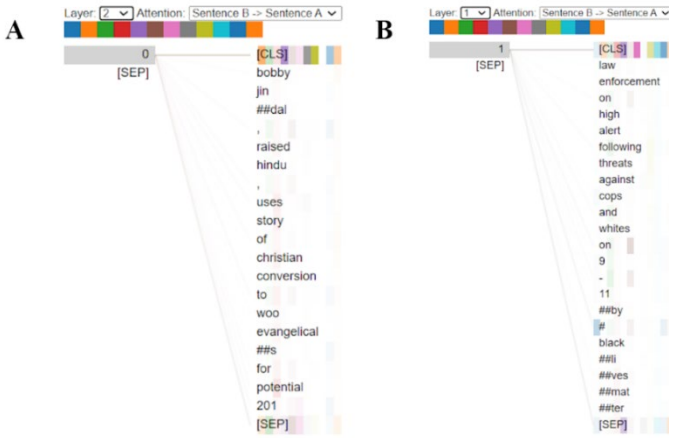

Figure 9. Baseline visualizations with BertViz. As seen in figure 9A and 9B, labels are more drawn to special tokens and punctuation.

Second, we tried to access text from the test data to see the relationship between WELFake articles and the labels it classified. When we use Bertviz, we use preprocessed text that is tokenized and encoded to create an attention mask. However, when we accessed our test data, they were already converted to ids. So, we converted them back to tokens to encode and get their attention masks that way.

We analyzed correctly classified fake and real news and falsely classified fake and real news. While there were common patterns of more attention to special BERT tokens and punctuation like the baseline visualizations above, what was interesting was that the model paid less attention to prepositions. Another interesting observation we made about the false negative classification we tested was it had the separator and label added at the end of 100 words of each article cut off. This could be due to the article being truncated at 200 tokens even though it did not use up all 100 words before adding the separator label pair. Further research might be needed to determine if adding the label at the end of the news articles impacts the performance of the model.

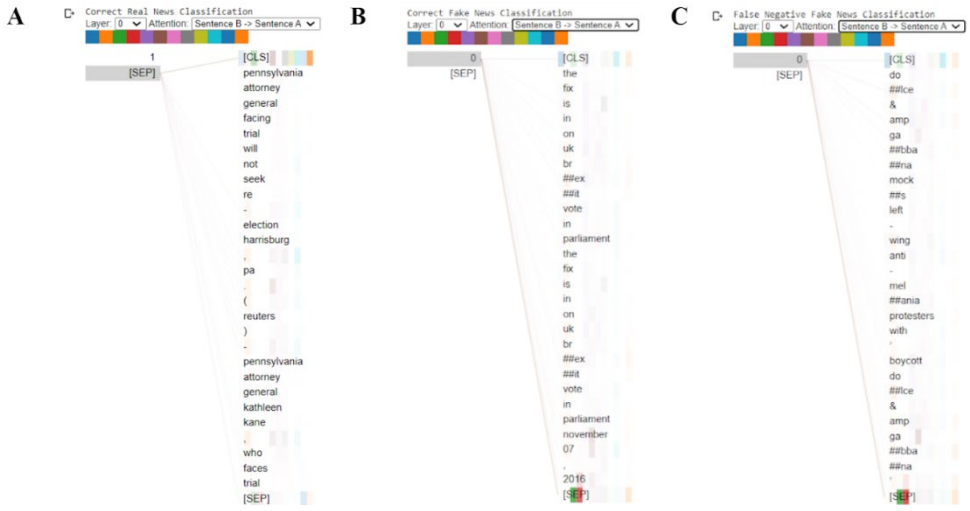

Figure 11. Visualizations of model classified text. figure 11A and 11B are correct classifications whereas figure 11C is a false negative.

5. Discussion & Future Work

In this paper, we reached two answers for our two research questions. Current SOTA fake news detectors made for classifying short texts can maintain their high performance when trained and tested on longer texts. It was also possible to have an explanatory fake news detector that had comparable performance to non-explanatory fake news detectors. However, while we were able to achieve a high performing explanatory fake news detector, it can be contested whether our visualizations are actual explanations understandable to users. BertViz reveals which words are most connected to the model's classification labels, but it is still uncertain if users can interpret the visualization as an explanation to why an article might be fake news. BertViz also uses the tokenized versions of text, which forces it to split up proper nouns to be words that exist in BERT's vocabulary, creating tokens that make it less interpretable. Lastly, the length limits of BertViz, while can be avoided by calling the method multiple times, can also hinder usability of the model. Therefore, further research may need to be done to improve the interpretability of high performing explanatory fake news detectors.

On a broader scale, both long text fake news classification and explanatory fake news detectors need to be further researched. There are an overwhelming number of datasets with short text data extracted from social media platforms, but little datasets created from full text length articles. Even as social media platforms become more prevalent, many users still utilize news outlets to get information. An effort on all kinds of text is necessary to effectively combat misinformation. Similarly, as current fake news detectors reach the highest accuracies, most of those models cannot explain their classifications. Prioritizing both high performance and explainability will increase the effectiveness of these models.

6. Conclusion

We started our research with two research questions: (1) can we create a fake news detector that can achieve the same high performance of SOTA fake news detectors while representing its classifications in explainable visualizations, and (2) can long texts like normal news articles can perform just as well in current fake news detectors meant for short texts. To answer these questions, we used the full length, long news article dataset called WELFake. We first ran baseline models of Majority Class Baseline and Random Forest Classifier with bag of words. To ensure we had a comparable performance metric of a SOTA model tested with WELFake, we used the third-place model from the AAAI 2021 Shared Task: COVID-19 Fake News Detection in English competition (Li et al., 2021). Although performance was poor when testing the model with WELFake, after making adaptations of Li's et al. (2021) model into a Two-Fold Four Model ensemble and training the model beforehand with WELFake, test accuracy reached a surprising 99.92%. This answered our first research question, revealing current SOTA detectors used for short text can perform just as well with long texts.

Lastly, we presented our second research question by making a manually fine-tuned BERT model to access its attention models that we could use to visualize attention masks through BertViz. Our manually fine-tuned BERT model maintained the same accuracies as our SOTA model with a 99.99% test accuracy, answering our second research question by creating an explanatory fake news detector maintaining accuracies comparable to current SOTA models. We then implemented our BERT model's attention masks to BertViz and saw the relationships between a BERT's classification label and the first 28 words of that news article, allowing our model to be explanatory.

Acknowledgements

I would like to thank the Inspirit AI program for letting me work with a research community that supports helping high school students draft their own research papers.

References

Ahmed, H., Traore, I., & Saad, S. (2018). Detecting opinion spams and fake news using text classification. *Security and Privacy*, *1*(1), e9. https://doi.org/10.1002/spy2.9

Kirchner, J., & Reuter, C. (2020). Countering fake news: A comparison of possible solutions regarding user acceptance and effectiveness. *Proceedings of the ACM on Human-computer Interaction*, *4*(CSCW2), 1-27. https://doi.org/10.1145/3415211

Li, X., Xia, Y., Long, X., Li, Z., Li, S. (2021). Exploring Text-Transformers in AAAI 2021 Shared Task: COVID-19 Fake News Detection in English. In: Chakraborty, T., Shu, K., Bernard, H.R., Liu, H., Akhtar, M.S. (eds) Combating Online Hostile Posts in Regional Languages during Emergency Situation. CONSTRAINT 2021. Communications in Computer and Information Science, vol 1402. Springer, Cham. https://doi.org/10.1007/978-3-030-73696-5_11

Nguyen Vo and Kyumin Lee. (2021). Hierarchical Multi-head Attentive Network for Evidence-aware Fake News Detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 965–975, Online. Association for Computational Linguistics. 10.18653/v1/2021.eacl-main.83

Patwa, P. *et al.* (2021). Fighting an Infodemic: COVID-19 Fake News Dataset. In: Chakraborty, T., Shu, K., Bernard, H.R., Liu, H., Akhtar, M.S. (eds) Combating Online Hostile Posts in Regional Languages during Emergency Situation. CONSTRAINT 2021. Communications in Computer and Information Science, vol 1402. Springer, Cham. https://doi.org/10.1007/978-3-030-73696-5_3

Shu, K., Cui, L., Wang, S., Lee, D., & Liu, H. (2019, July). defend: Explainable fake news detection. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 395-405). https://doi.org/10.1145/3292500.3330935

Shu, K., Mahudeswaran, D., Wang, S., Lee, D., & Liu, H. (2020). FakeNewsNet: A Data Repository with News Content, Social Context, and Spatiotemporal Information for Studying Fake News on Social Media. *Big data*, *8*(3), 171–188. https://doi.org/10.1089/big.2020.0062

Szczepański, M., Pawlicki, M., Kozik, R. *et al.* (2021). New explainability method for BERT-based model in fake news detection. *Sci Rep* 11, 23705. https://doi.org/10.1038/s41598-021-03100-6

Verma, P. K., Agrawal, P., Amorim, I., & Prodan, R. (2021). WELFake: word embedding over linguistic features for fake news detection. *IEEE Transactions on Computational Social Systems*, *8*(4), 881-893. 10.1109/TCSS.2021.3068519

Vig, J. (2019, May). BertViz: A tool for visualizing multihead self-attention in the BERT model. In *ICLR workshop: Debugging machine learning models* (Vol. 23).