Fake News Detection

DS4A: Women's Summit

Rabiya Noori
Jerri Zhang
Iris Yoon
Renee G. Reynolds
Hannah Mei

Introduction

Disinformation and manipulation in the media have always existed, but with the advent of online news outlets and social media platforms, there has been an increase in the amount of misinformation spread, the ability of anyone to contribute false information, and the speed at which this information can be spread to larger audiences. Fake news has even drawn more attention as it is widely used to sway opinions in the past and upcoming elections.

We understand that being able to understand the credibility of the news plays an important role for us to make informed decisions and judgements in daily life. Because of that, we aim to build a web app/API that can differentiate between real and fake news. We will use publicly available news to train a deep learning model and deploy the model on a website or as an API as a fake new detector, which can help prevent the spread of misinformation and its harmful consequences on society.

Data Analysis & Computation

Datasets + Data Wrangling & Cleaning

Dataset

We used the fake news dataset from Kaggle, which contained about 23,000 fake news articles and 21,000 real news articles.

The following table shows an example of a fake and real article from the dataset. We highlighted structural differences between the fake and real articles that were repeatedly observed.

- Data source: Fake articles displayed data sources at the end of the article, while real articles displayed data sources at the beginning.
- Punctuation: Fake articles were missing apostrophes.

Fake article	Real article
Title: Donald Trump Just Visited The 9/11 Museum And P*ssed EVERYONE Off (IMAGES/TWEETS) **Text:** Native New Yorker and Republican front runner Donald Trump made his first-ever visit to the 9/11 Memorial ... Trump doesn t give a damn about the victims, survivors or families affected by 9/11 he only cares about making himself look better. Featured image via Josh Andew	**Title:** As U.S. budget fight looms, Republicans flip their fiscal script **Text:** WASHINGTON (Reuters) - The head of a conservative Republican faction in the U.S. Congress, who voted this month for a huge expansion of the national debt to pay for tax cuts, called himself a “fiscal conservative” ... The Senate has not yet voted on the aid.

Instagram	

Preprocessing

We performed text preprocessing with the goal of (1) reducing bias in our dataset and (2) preparing our dataset for language models.

1. **Reducing bias in dataset**
    - Removed data source: We used regex to remove data sources at the end of fake articles and the beginning of the real articles. This step was necessary because we expect our final product to be used on articles in which the data source is absent. By removing the data sources, we prevented our fake news detector from relying on explicit data sources.
    - Added apostrophes: We used regex to add apostrophes when we encountered strings ‘s’, ‘t’, ‘ve’, ‘ll’, ‘m’, separated by a space. In particular, this would transform the text “doesn t” to “doesn’t”, “Trump s” to “Trump’s”, and so on. This step was necessary because the apostrophe was missing from every fake article. We did not want our fake news detector to base its decision on the absence of apostrophes. Moreover, we expect our final product to be used on articles with correct punctuations.
2. **Preparation for language models**
    - Removed url, twitter handles, hashtags: We removed these items because many of the fake articles came from social media platforms, including twitter. We did not want our model to learn that any article from twitter was fake.
    - Removed stopwords, punctuation, symbols, numbers: We removed these items as they do not provide much information for language models. This step reduces the feature size as well.
    - Convert all text to lowercase & stemming: Stemming reduces inflection in words to their root forms. We performed these steps to standardize the vocabulary in our dataset.

Note: Different analysis tools require different amounts of preprocessing. For example, we performed fewer preprocessing steps before applying BERT, because BERT utilizes its own tokenizer which performs many of the preprocessing steps. In particular, we did not remove stop words because they may provide contextual information for BERT. Furthermore, we did not stem the words because stemming would remove syntactic nuances.

Vectorizing Articles

We explored vectorizing news articles via tf-idf and doc2vec.

- Tf-idf encodes a document by numeric statistics that show how important each word is to the specific document. In our case, each document was encoded into a sparse vector of dimension 160,000
- Doc2vec computes a vector representation of a document through a model very similar to word embeddings. In our case, each document was encoded into a vector of dimension 300.

After vectorizing the articles through tf-idf and doc2vec, we visualized 700 random articles via PCA. As one can see in the following figures, dov2vec seems to have a better separation between fake and real articles than tf-idf. Moreover, doc2vec vectors are only 300-dimensional, while the tf-idf vectors have a much higher dimension. For such reasons, doc2vec seems like a better method for vectorizing articles than tf-idf.

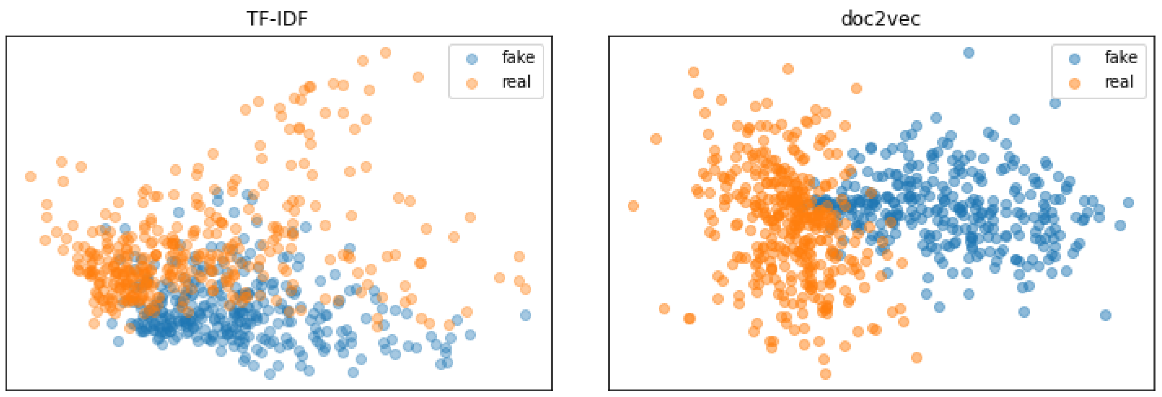


Figure 1: Visualizations of document embedding via TF-IDF and doc2vec

Limits of The Dataset

While we did our best to eliminate the structural differences in the fake and real articles, there may be remaining inherent differences in the dataset. Many of these differences probably stem from the fact that the fake news articles tend to be prevalent on social media platforms. Our model may thus learn to detect fake news articles based on these structural or syntactic cues of the writing.
In order to understand the extent of these issues, one can perform follow up studies on our final model. One can evaluate the performance of our model on real news from social media platforms and on fake news articles written in the style and formality of a real news article.

Exploratory Data Analysis

As mentioned in the previous section, our raw dataset from Kaggle contains approximately 23,000 fake news articles and 21,000 real news articles. Below we show a barchart of these exact frequencies, namely 23,481 fake articles and 21,417 real articles.

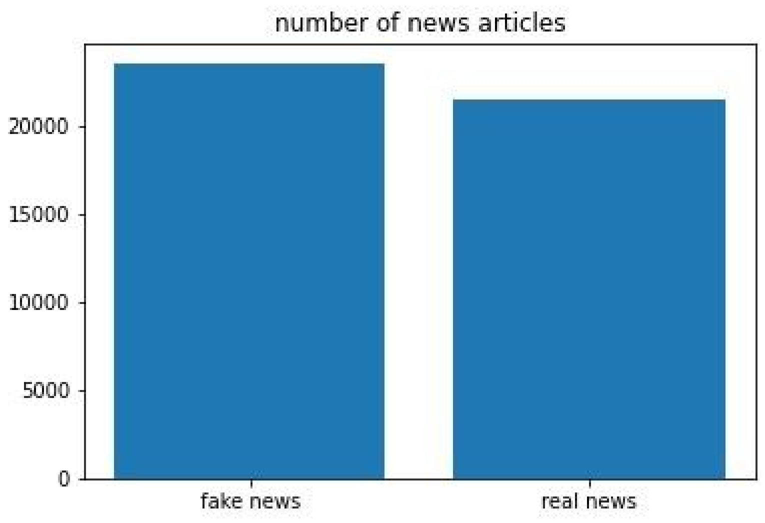


Figure 2: Barchart displaying number of fake and real news articles in Kaggle dataset

The median length of the raw fake news articles is 2,166 characters with a standard deviation of 2,532.88 characters, while the real news articles have a median length of 2,222 characters with a standard deviation of 1,684.84 characters. We notice that real and fake news have similar median lengths; however, fake news has higher variability in news length.

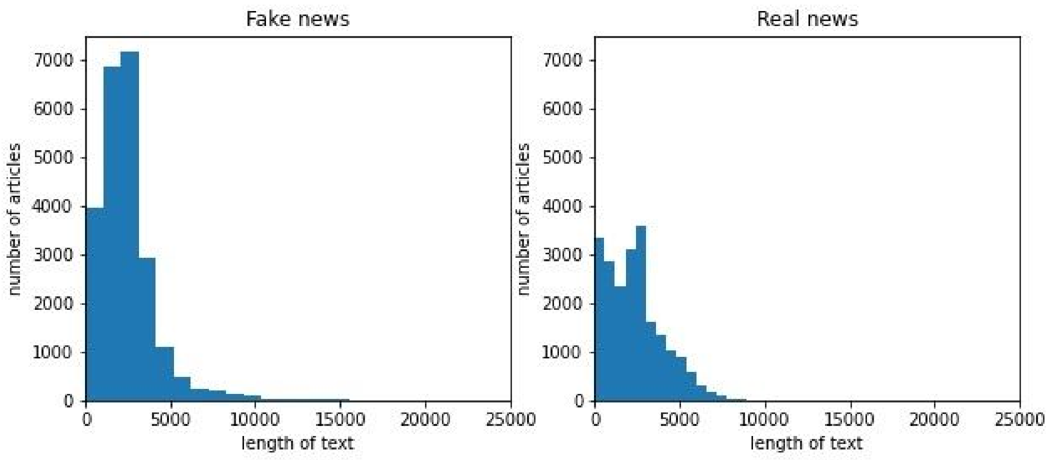


Figure 3: Length of fake and real news articles

To get a sense of the prominent words, names, and topics that appear in our data, we plot the top non-stopwords in each of the news titles only. We exclude stopwords – words such as ‘be’, ‘is’, ‘of’, ‘in’, ‘to’ – as these are some of the most common words in the English language and do not contribute to our understanding of the core themes present in the data. Below we display the top stopwords in the real and fake news article titles, rather than the text, since we conduct our exploratory data analysis on our raw data prior to additional cleaning/processing, and we believe that the news title captures the subject of the text. Additionally, the top non-stopwords on the raw text include punctuation and tags such as ‘(Reuters)’, which are not as informative.

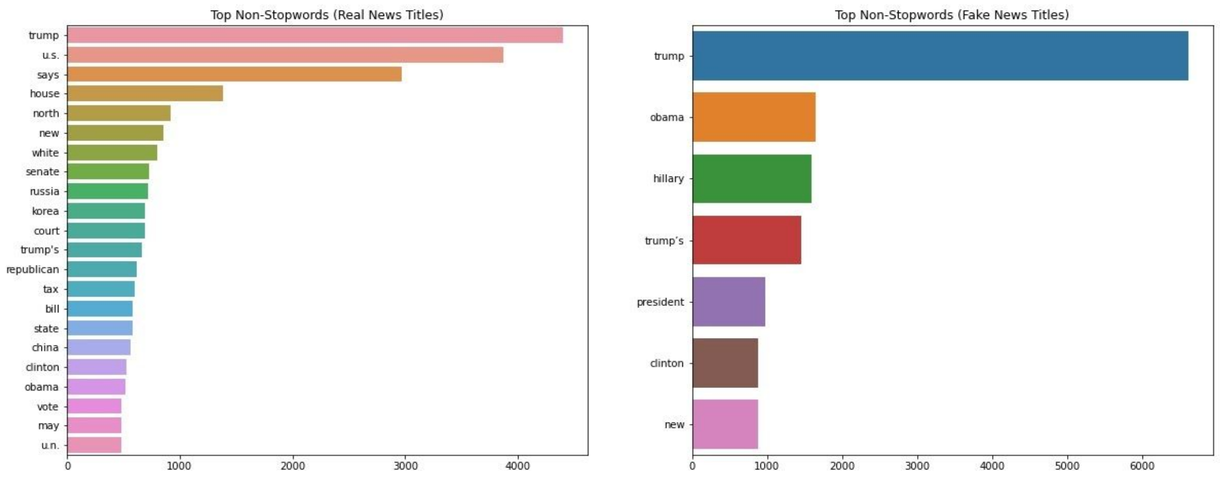


Figure 4: Top non-stopwords that appear in the real and fake news titles

We notice that the fake news titles seem to focus heavily around the outgoing president and incoming presidential nominees, as we see 'Trump', 'Obama', 'Hillary', 'president', and 'Clinton' appear most frequently. This is to be expected as fake news involving Trump, Obama, and Clinton is believed to have proliferated during the 2016 US presidential election. The top non-stopwords in the real news titles appear to focus more broadly on political issues including not only Trump, Clinton, and Obama, but also the senate, the U.S., North Korea, Russia, China, and a Republican tax bill. In short, the real news titles appear to focus on general topics relevant to the election, while the fake news titles focus only on the individuals involved. We also find it interesting that the most common word that appears in both real and fake news titles is 'Trump'.

Similarly, we look at the most common bigrams and trigrams in the real and fake news titles. Bigrams and trigrams are pairs and trios of consecutive words, used to understand some common phrases that appear in the data, rather than singular words as in Figure 3.

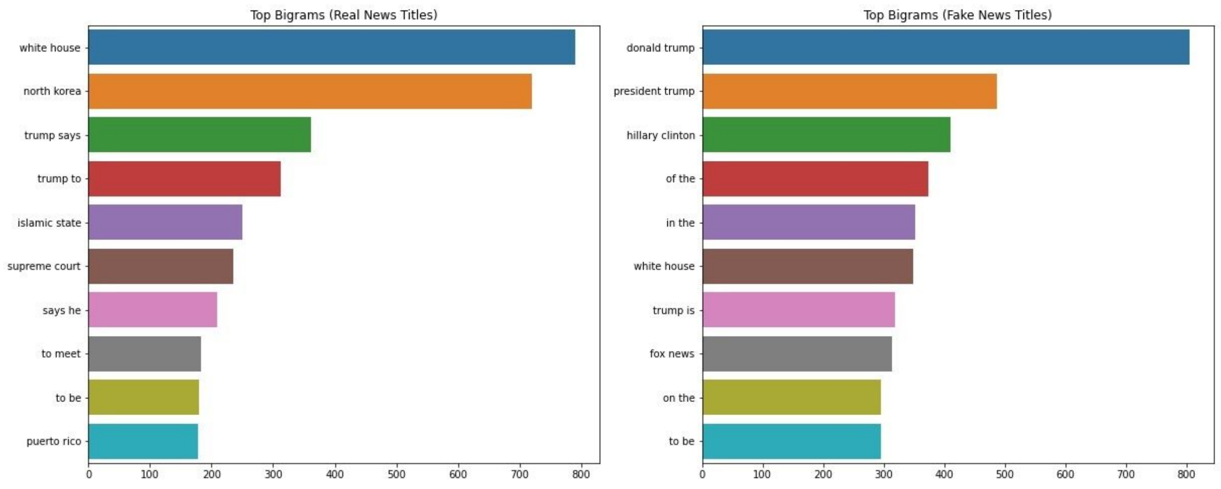


Figure 5: Top bigrams that appear in the real and fake news article titles

As we see in Figure 3, the top bigrams for the fake news titles also include 'Donald Trump', 'Hillary Clinton', as well as the 'White House' and 'Fox News'. For the real news titles, we see 'White House', 'North Korea', 'Trump says', 'Islamic State', and 'Supreme Court'. We gather similar themes as from Figure 3 – fake news focuses particularly on the presidential candidates as well as a right-leaning news source, and real news focuses on a few main issues surrounding the election.

In addition, we also look at the top trigrams.

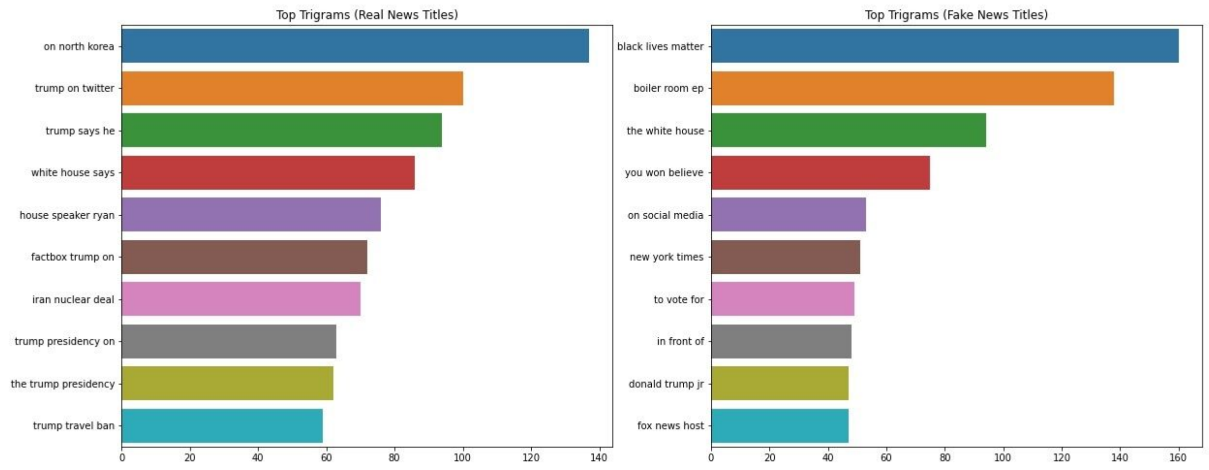


Figure 6: Top trigrams that appear in the real and fake news article titles

Now, we start to see a shift from the 2016 election to more recent events. The fake news titles center around the Black Lives Matter movement, the white house, social media, the New York Times, Donald Trump Jr., and Fox News hosts, while the real news titles focus on North Korea, Trump's claims, twitter and travel ban, the White House's claims, House Speaker Ryan and the Iran nuclear deal.

We also show word clouds below of the raw fake and real news article text to visualize common words found in the text itself, rather than just the titles. In the fake news text, similar to the titles, we see Donald Trump, Hillary Clinton, the United States, the White House, America, Obama, and Twitter are heavily featured.

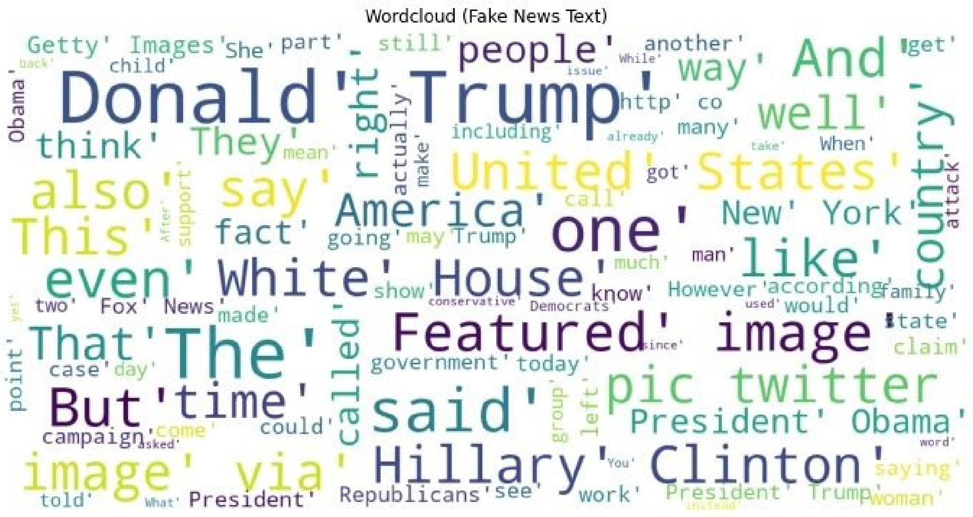


Figure 7: Wordcloud of fake news article text

Words commonly found in the real news articles include Washington and New York (the location tags of many of the articles), Reuters (the source of many of the articles), Donald Trump, the United States, the White House, Prime Minister, North Korea, and the Islamic State.

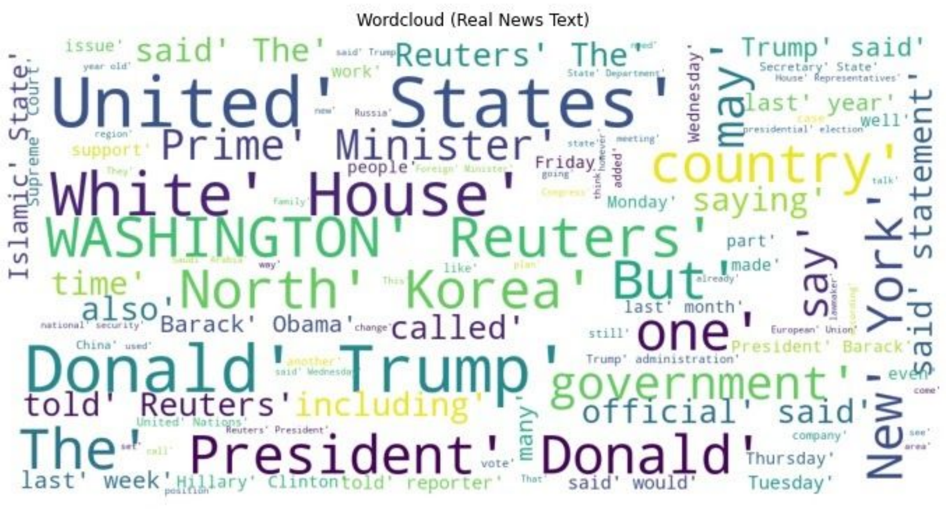


Figure 8: Wordcloud of real news article text

Statistical Analysis & Machine Learning

Baseline Model I - Document Encoding & Linear Classifier

Recall from the data preprocessing step that encoding news articles via doc2vec led to the following visualization of our dataset. Since the visualization shows a nice separation between the fake and real articles, we decided to train a linear classifier on the encoded dataset.

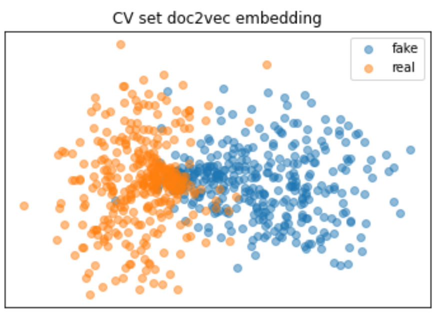


Figure 9: Visualization of documents encoded via doc2vec

Each document was encoded into a 300-dimensional vector via doc2vec. We then performed logistic regression on the training set to build a linear classifier. We report the performance of the model on the test set. We achieved an accuracy of 97%, with an F1 score of 0.97.

```
              precision    recall  f1-score   support

           0       0.97      0.96      0.96      3165
           1       0.96      0.97      0.97      3570

    accuracy                           0.97      6735
   macro avg       0.97      0.96      0.96      6735
weighted avg       0.97      0.97      0.97      6735
```

Figure 10: Classification report on test set

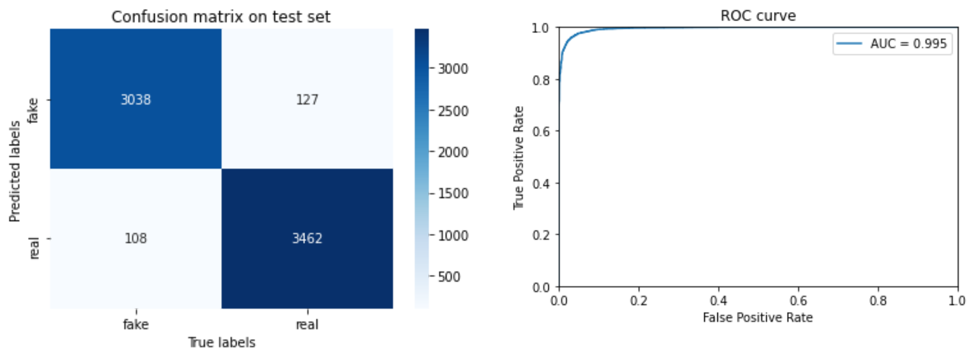


Figure 11: Confusion matrix and ROC curve on test set

While the model performed well on the given test set, we caution the reader regarding the model performance. As mentioned in the Dataset section, we found some structural differences between the fake and real news articles in our dataset. While we did our best to remove such differences during the preprocessing step, there may be additional biases that we failed to remove. We plan to evaluate the performance of this model on a separate dataset.

Furthermore, while this model has a good performance, it may not be the ideal model to deploy in an application, as the model does not provide any reasoning for its classification. In a later section, we explore the use of LIME to provide explanations for a RNN-based classifier.

Baseline Model II - Recurrent Neural Networks (RNN)

Many classification and regression tasks involve data that are assumed to be independently and identically distributed. Various modeling techniques have been developed to handle these types of data, such as regression, decision tree and deep learning models like feed-forward neural networks.

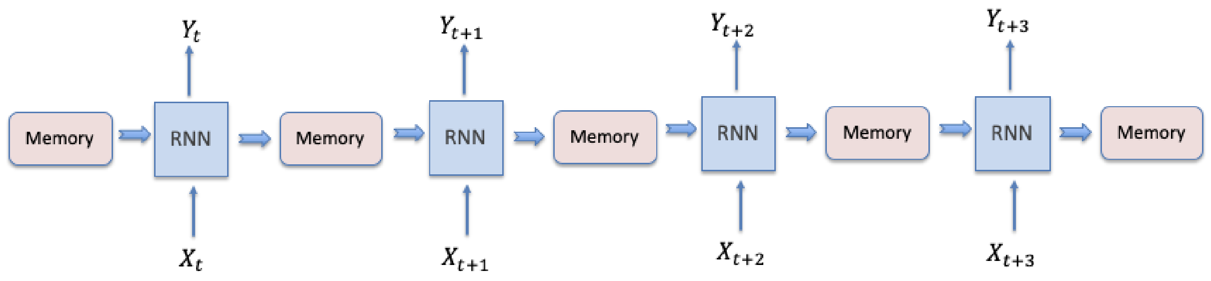
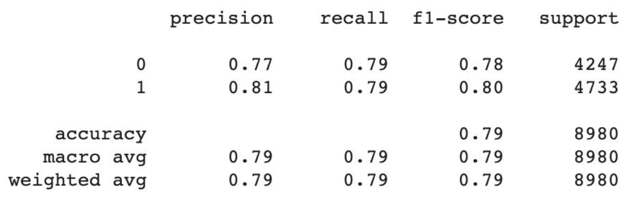


Figure 12. Recurrent neural network (unfolded)

However, much of our data today is also inherently sequential. Recurrent neural networks recognize the data's sequential characteristics and use patterns to address classification and regression tasks. Each recurrent unit in the network contains value composed by input at time t and memory from previous input at time t-1. For each input sentence or article of length T, the network unfolds for a sequence of recurrent state of length T and updates its memory with each word. During training, the network applies backpropagation through the recurrent state and improves the network by optimizing the loss function.

We constructed a RNN model as our baseline model, which yields a test accuracy of 0.79.



```
              precision    recall  f1-score   support

           0       0.77      0.79      0.78      4247
           1       0.81      0.79      0.80      4733

    accuracy                           0.79      8980
   macro avg       0.79      0.79      0.79      8980
weighted avg       0.79      0.79      0.79      8980
```

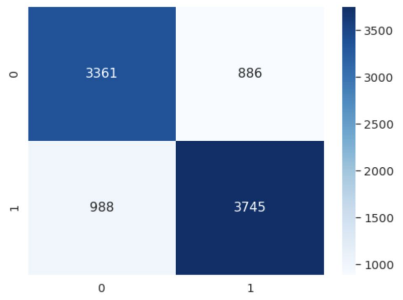


Figure 13. Test Accuracy for Baseline RNN model

Transfer Learning with BERT

Transfer learning is commonly applied to natural language processing problems given the complex nature of the training samples and the model architecture. Since the amount of data needed to accurately train a language model isn't readily available for many language problems it's helpful to use a pre-trained model. For our problem we apply transfer learning on the BERT (Bidirectional Encoder Representations from Transformers) model. There are three ways we can train using BERT: train the entire BERT model using our data, freeze some layers and weights and train the rest or freeze all the layers and weights of BERT and add extra layers to train. For our problem we chose the last option. We froze the entire BERt architecture with pre-trained weights and added two dense layers. We trained these last two layers over 5 epochs; because the BERT model was pre-trained so well, it does not require many epochs to train the extra two layers with our data.

In order to feed data into the BERT model, it needs to undergo another preprocessing step known as tokenization. In this step the words in the input sample are separated into 'tokens' meaning the input string is split into individual words (sometimes the words are further split into sub-words). Each token is also assigned a corresponding numerical ID. When the original BERT model was trained, each word in the vocabulary was assigned a unique ID, so it's important that we assign those same words in our dataset with the appropriate IDs. The maximum length of tokens that can be used as input to BERT is 512, so we had to truncate our articles to be 512 tokens in length.

The final trained model performed very well on the test set, achieving an AUC score of 0.9319.

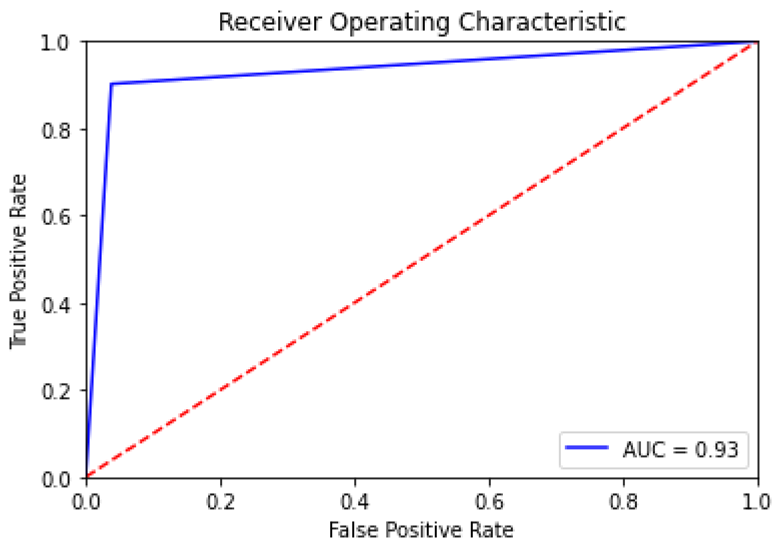


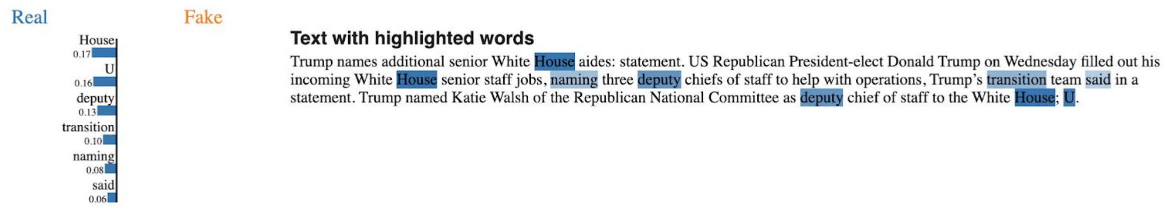
Figure 14: ROC curve of classification on test data with BERT model

	Actual: True	Actual: Fake
Predicted: True	3090	123
Predicted: Fake	345	3177

Table 1: Confusion Matrix of classification on test data with BERT model

Model Interpretability - LIME

Local Interpretable Model-Agnostic Explanations (LIME) allows us to better understand how our model is making the prediction decision, what happens when the model is making the correct/incorrect prediction, and which words have the most weight during decision making. Below we present three case studies of the baseline RNN model where in one case the model makes a correct prediction, in one case the model makes an incorrect prediction, and in another case the model makes a correct classification but for the wrong reason.

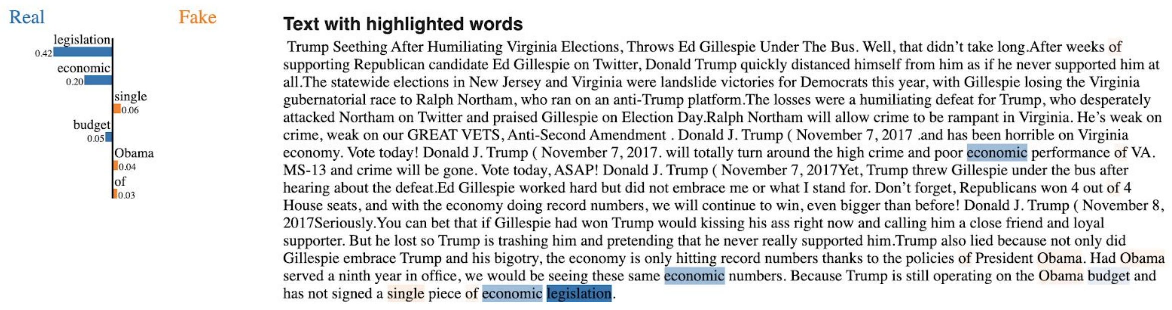


Text with highlighted words

Trump names additional senior White House aides: statement. US Republican President-elect Donald Trump on Wednesday filled out his incoming White House senior staff jobs, naming three deputy chiefs of staff to help with operations, Trump's transition team said in a statement. Trump named Katie Walsh of the Republican National Committee as deputy chief of staff to the White House; U.

Figure 14. Case study for correct classification

The above news is labeled as Real, and the model determines the news to be real as well, which means the model is making a correct prediction. From the LIME interpret plot, you can see that the model is making the predictions based on words such as 'House', 'deputy' and 'transition', which provides some insights on which words are associated with real news.

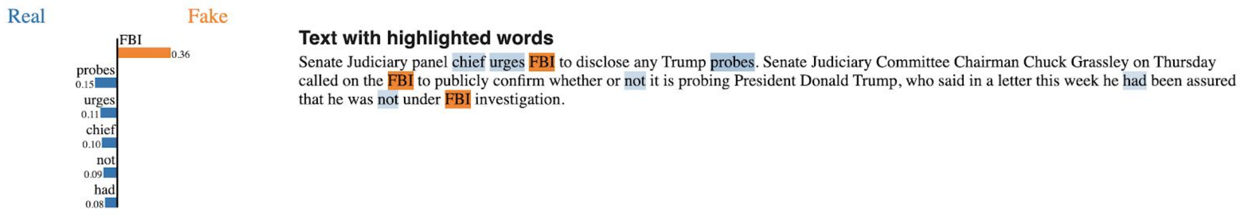


Text with highlighted words

Trump Seething After Humiliating Virginia Elections, Throws Ed Gillespie Under The Bus. Well, that didn't take long.After weeks of supporting Republican candidate Ed Gillespie on Twitter, Donald Trump quickly distanced himself from him as if he never supported him at all.The statewide elections in New Jersey and Virginia were landslide victories for Democrats this year, with Gillespie losing the Virginia gubernatorial race to Ralph Northam, who ran on an anti-Trump platform.The losses were a humiliating defeat for Trump, who desperately attacked Northam on Twitter and praised Gillespie on Election Day.Ralph Northam will allow crime to be rampant in Virginia. He's weak on crime, weak on our GREAT VETS, Anti-Second Amendment . Donald J. Trump (November 7, 2017 .and has been horrible on Virginia economy. Vote today! Donald J. Trump (November 7, 2017. will totally turn around the high crime and poor economic performance of VA. MS-13 and crime will be gone. Vote today, ASAP! Donald J. Trump (November 7, 2017Yet, Trump threw Gillespie under the bus after hearing about the defeat.Ed Gillespie worked hard but did not embrace me or what I stand for. Don't forget, Republicans won 4 out of 4 House seats, and with the economy doing record numbers, we will continue to win, even bigger than before! Donald J. Trump (November 8, 2017Seriously.You can bet that if Gillespie had won Trump would kissing his ass right now and calling him a close friend and loyal supporter. But he lost so Trump is trashing him and pretending that he never really supported him.Trump also lied because not only did Gillespie embrace Trump and his bigotry, the economy is only hitting record numbers thanks to the policies of President Obama. Had Obama served a ninth year in office, we would be seeing these same economic numbers. Because Trump is still operating on the Obama budget and has not signed a single piece of economic legislation.

Figure 15. Case study for incorrect classification (fail to detect fake news)

The above news is labeled as Fake, but the model determines the news to be real, which indicates the model is making an incorrect prediction. From the LIME interpret plot, you can see words such as 'legislation' and 'economic' are positively related to the 'Real' class, and that's why the model made a wrong judgement. In other words, if the new article contains such words, the chance the model fails to detect it as fake is lower.



Text with highlighted words

Senate Judiciary panel chief urges FBI to disclose any Trump probes. Senate Judiciary Committee Chairman Chuck Grassley on Thursday called on the FBI to publicly confirm whether or not it is probing President Donald Trump, who said in a letter this week he had been assured that he was not under FBI investigation.

Figure 16. Case study for incorrect classification (mistake real news for fake)

The above news is labeled as Real, but the model determines the news to be fake, which indicates the model is making an incorrect prediction. From the LIME interpret plot, you can see that the model made the wrong conclusion because 'FBI' is highly related to fake news from our training corpus. We can interpret it as that many fake news have the FBI related content, hence when a real news article mentions the FBI, our model still classifies it as fake.

External Validation of Transfer Learning Model

Because we were concerned with the biased nature of our training dataset, we evaluated the performance of our model on Kaggle's fake news dataset from 2016. The data consists of around 10,000 articles labeled 'bs' by the BS Detector chrome extension. Assuming that all data are, in fact, fake articles, we obtain an accuracy of 82%. We examined some articles in the dataset that were classified as being real articles, and noticed that some articles were, indeed, based on factual information. This may arise because the dataset collected via the BS Detector may include some false positives, that is, real articles labeled as 'bs'.

We then tested our model on 8 of the 10 most-viewed fake news stories from 2019. We made sure that these 8 articles were not part of the training data for our model. Our model correctly identified 4 out of the 8 articles as being fake news. The following table shows the titles of the fake news articles and the classification according to our model.

Fake news title	Model classification
Trump's grandfather was a pimp and tax evader	fake
Pelosi Diverts $2.4 Billion From Social Security To Cover Impeachment Costs	fake
Ocasio-Cortez Proposes Nationwide Motorcycle Ban	fake
Trump Is Now Trying To Get Mike Pence Impeached	fake
Omar Holding Secret Fundraisers With Islamic Groups Tied to Terror	real
BREAKING: Nancy Pelosi's Son Was Exec At Gas Company That Did Business In Ukraine	real
Dems Vote To Enhance Med Care for Illegals Now, Vote Down Vets Waiting 10 Years for Same Service	real
Joe Biden Calls Trump Supporters the "Dregs of Society"	real

Table 2: popular fake news from 2019 and model classification

This work should be validated further via external validation involving fake and real news data from various sources. In particular, our future work would involve evaluating our model on fake and real articles scraped from known sources.

Conclusions and Future Work

From our research and analysis, we can conclude that there are intrinsic differences between the real and fake news and we are able to implement a Natural Language Processing model to detect these two categories.

However, there are a few limitations that we need to consider when making the conclusion. Our main limitation is due to the inherent bias within the dataset. Even though we managed to obtain Kaggle dataset and tried to preprocess it to remove distinctive non-semantic signals that were not part of our research scope, there could still be some bias remaining in the fake news dataset that might lead to inaccurate results. Another limitation is due to insufficient fake news dataset from various sources. Since most fake news is usually taken down relatively quickly, we are only able to use the Kaggle data set that was gathered from social media platforms. Therefore, we believe the accuracy of our Fake News Detection mechanism will be greatly improved when incorporating datasets via diverse sources and platforms. The accuracy of the model will also improve when we retrain the model with more latest news instead of using 2016 election data.

In addition, we are also interested in running the model against news in fields other than politics such as healthcare and environmental issues to cross validate the accuracy of our current model. There could be new findings among these different categories of news and will therefore, inform us on the transitivity and applicability of our model based on political news to other fields. We would also like to incorporate news in other fields for model training so that we could expand our service scope and increase our validity. Eventually, we hope to implement a web application that could provide fake news detection service through the platform for daily use.

notes:
Ronnie Ghose20:09
https://slidesgo.com/

Ronnie Ghose20:11
https://thenounproject.com/

flaticon

https://medium.com/analytics-vidhya/how-to-deploy-simple-machine-learning-models-for-free-56cdccc62b8d

https://arxiv.org/pdf/1812.00315.pdf

- describe the impact of fake news
- try to convince VC why they should fund your project

‘consumer trust’
human fact checking vs machine fact checking investments