Fake news identification using text classifier

Aman Kumar Nayak (amana551)

732A92

Abstract

In recent years deceptive news articles are becoming dangerous prospects for online users, and with most of the news being published as online articles, it is becoming difficult to track the origin and validate information based on the source. Using fake news for political and economic gain is a rising trend in online articles. Presently, two ways are dominating in the detection of fake news. The first approach is to use human analysts as fact-checkers and is limited to analyst's knowledge to detect misinformation, the second is to use machine learning-based natural learning processing to detect fake news. In this project, three machine learning based models were compared using Kaggle's Fake News dataset and evaluated against their performance to distinguish the articles. The Bidirectional Encoder Representations from Transformers (BERT) is used in this project as a Transformer based machine learning algorithm, Logistic Regression based Classifier is used as a baseline model and paired with Random forest as Ensemble learning method. Logistic Regression based classifier is selected as the final model for its low requirement of training time and computation resources with a higher recall score.

Introduction

Spreading fake news to gain political and economical gain is an age-old practice and its instance can be seen in ancient Rome, where Octavian spread false information about Mark Antony's character and his relationship with Queen Cleopatra[i] for personal gains. The rapid growth of social media platforms like Facebook and Twitter and their adoption by news agencies to share news in near real-time with the users have removed the limitation to having a physical network of distributors to spread the news. It is easy for the end-user to access news in real-time with few clicks on a cellphone. Facebook alone accounts for 70% of the traffic to news websites[ii]. These social media platforms are extremely powerful and liberating for consumers as they allow people to share their belief about political, economical, and other personal issues but it has been seeing that certain group use these platforms to spread their agenda for personal gains[iii] and in other cases to formulate a biased opinion, manipulating readers mindset and spreading fear. This practice of spreading misinformation is commonly known as fake news.

Study shows that our capability to take the decision is affected by the news we consume[iv] which make fake news more severe problem than just another news article. Hence, the requirement of validating news articles is one of the major challenges in the current state. Using human analyst to perform this task is extremely costly and require subject matter expertise to judge the information which is hard to obtain. This makes using a machine learning based approach important, anomalies detection can be done by using Natural Language Processing (NLP) to separate articles that are deceptive in content from the rest.
To detect fake news, this project uses Kaggel Fake News Dataset[v]. The dataset contains labeled news articles where label 1 marked is unreliable i.e., fake news, and label 0 is assigned to it reliable news i.e., not fake news. The study aims to explore the data and evaluate the accuracy and performance of different classifiers namely Logistic Regression, Random Forest, and Bidirectional Encoder Representations from Transformers (BERT) on the same data.

Theory

Vectorization

Word vectorization is a method to map words or phrases from vocabulary to a corresponding vector of real number which can be used to find similarity in word. Term Frequency-Inverse Document Frequency (**TFIDF**) is a vectorization technique for textual data with many possible variations. Using TFIDF, two documents are considered to be similar if they share rare but informative words. In TFIDF, every term is considered a different dimension orthogonal to all other dimensions. Each term is represented by a weight that is positively correlated with its occurrence in the current document but is negatively correlated to its occurrence in all other documents in the corpus. This is done to downgrade the importance of the terms which is common across many documents considering these terms have less information specific to the objective text. One way to assign weights to term *t* in document *d* is given by:

$$TFIDF_{i,d,f} = TF_{t,d} * log \frac{|D_t| + 1}{DF_{t,D_t} + 1}$$

Here: |D| represents the total number of documents in the corpus, $TF_{t,d}$ represents the total number of occurrences of term t in document d and $DF_{t,D}$ is total number of documents where term t occurs.[i]

Logistic Regression

Logistic Regression can classify text based on a wide feature set obtained from TFIDF with a binary output of 0 and 1. This model is used as it provides an intuitive equation to classify articles into two classes. Mathematically, the logistic regression function can be defined as:

$$h_{\theta}(X) = \frac{1}{1+exp\big(-(\beta_0+\beta_1(X))\big)}$$

Here $\beta_0,\ \beta_1$ are class weights.

Logistic regression uses a sigmoid function to produce output class probability by minimizing the cost function to achieve an optimal probability.

$$Cost(h_{\theta}(x), y) = \begin{cases} log\big(h\theta(x)\big), & at\ y = 1 \\ -\,log\big(1 - \ h\theta(x)\big), & at\ y = 0 \end{cases}$$

Random Forest

Random forest (RF) is selected from ensemble learner class as it is an advanced form of decision trees which also uses a supervised learning model. RF consists of a large number of decision trees working individually to predict an outcome of a class where the final prediction is selected by majority voting. We have impurity measures in a random forest, namely Gini Index and Entropy.

Gini Index: $1 \ - \sum_{i\ =1}^{c}(P_i)^2$

Here P_i is probability of each class.

Binary cross entropy: $-(x\ log(p)\ +\ (1-x)\ log(1-p)\ )$

Here p is probability of predicted class.

Bidirectional Encoder Representations from Transformers

Bidirectional Encoder Representations from Transformers (BERT) is a pre-trained deep bidirectional representation from the unlabeled text by jointly conditioning on both left and right context in all layers.[vii]

To apply pre-trained language representations to downstream other strategy uses two major approached: feature based and fine-tuning. The feature based approach such as ELMo[viii] and fine-tuning approach, such as Generative Pre-trained Transformer[ix] use unidirectional language model to learn general language representation.

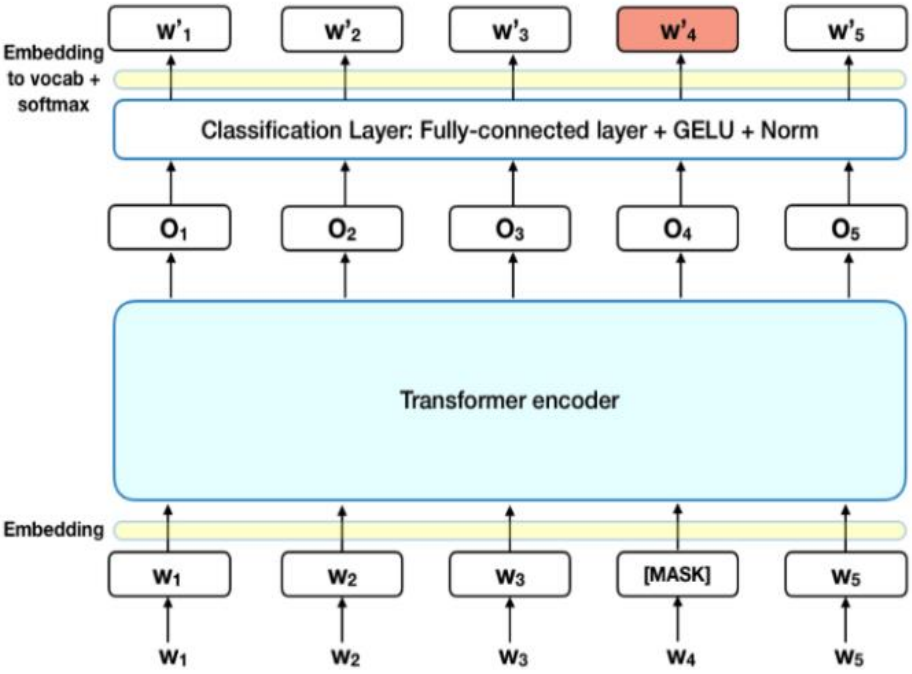
In BERT transform encoder reads an entire sequence of the statement at once thus it is bidirectional, unlike directional models which statement as the input sequence. This allows the model to learn the context of the word based on all its surrounding i.e., left and right of the word.

BERT uses two training strategies:

Masked LM (MLM):

Before feeding words sequence into BERT, 15% of words in each sequence are replaced with a MASK (token). The model then tries to predict the original value of masked words based on the context provided by the remaining non-masked words in a sequence. This is accomplished in 3 steps[x]:

1. Adding a classification layer on top of the encoder output
2. Multiplying the output vectors by the embedding matrix, transforming them into the vocabulary dimension.
3. Calculating the probability of each word in the vocabulary with softmax.


W'₁
W'₂
W'₃
W'₄
W'₅
Embedding to vocab + softmax
Classification Layer: Fully-connected layer + GELU + Norm
O₁
O₂
O₃
O₄
O₅
Transformer encoder
Embedding
W₁
W₂
W₃
[MASK]
W₅
W₁
W₂
W₃
W₄
W₅

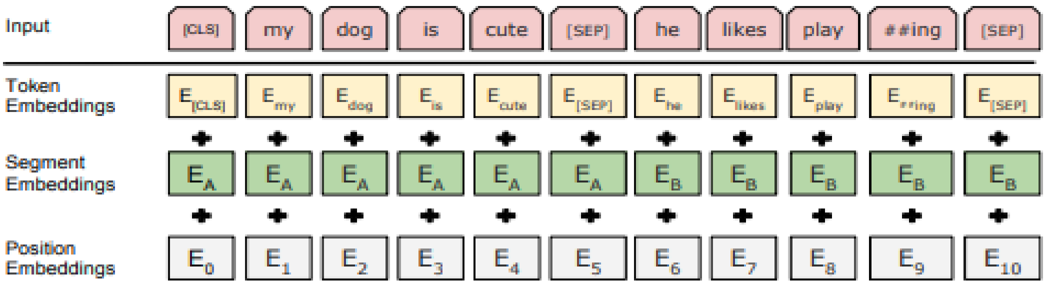
The BERT loss function only considers predictions made for masked words and ignores the prediction of non-masked words which increases model convergence time when compared to directional models, but it increases context awareness in the BERT model.

Next Sentence Prediction:

In the training process, the model receives two pair of sentences as input and learn to predict if the second sentence in the pair is following sentence in the original document. During training, 50% of sentences are randomly chosen to be the first sentence and while the remaining 50% are chosen to be following the second sentence.

To distinguish between two sentences during training, the following steps are performed:

1. A [CLS] token is inserted at the beginning of the first sentence and [SEP] token is inserted in the end of each sentence.
2. Segment embedding indicating sentence A and sentence B is added to each token.
3. A positional encoding is embedded to each token to indicate its position in the sequence.



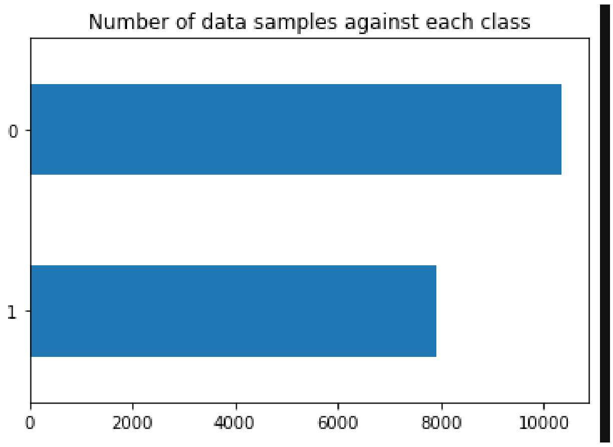
BERT input representation as shown in original paper[ii]

To perform a classification task an additional classification layer is placed on the top of the transformer output for the [CLS] token.

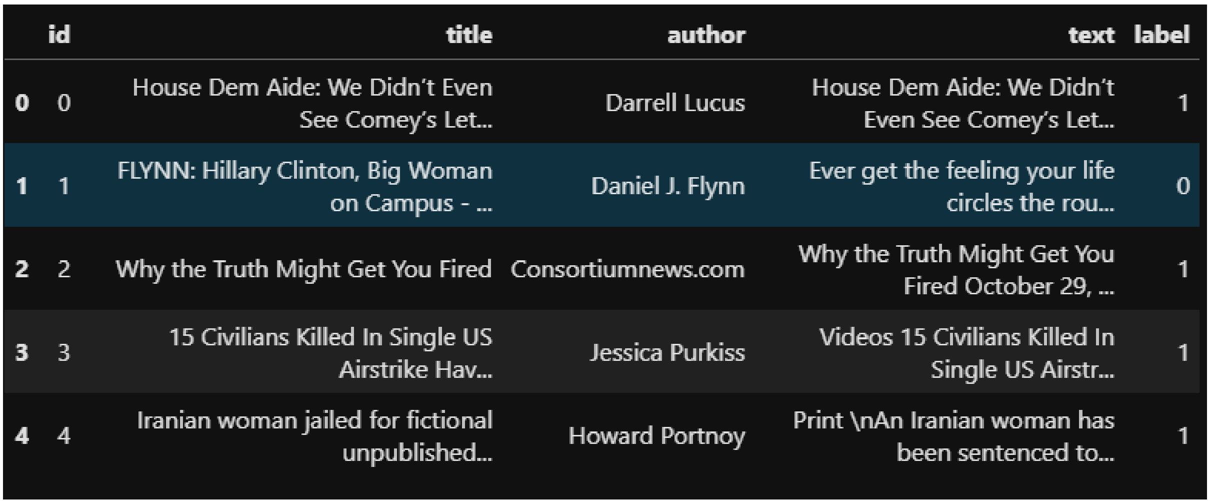
Data

The dataset used to perform classification consists of 18285 unique training samples with label 1 and 0, label 1 marks fake news, and 0 for not-fake news. Additionally, the test set contains 4500 samples.

In the training set, we nearly have 40% labels for label 1 and 60% labels for label 0 which makes this dataset fairly balanced for this task.

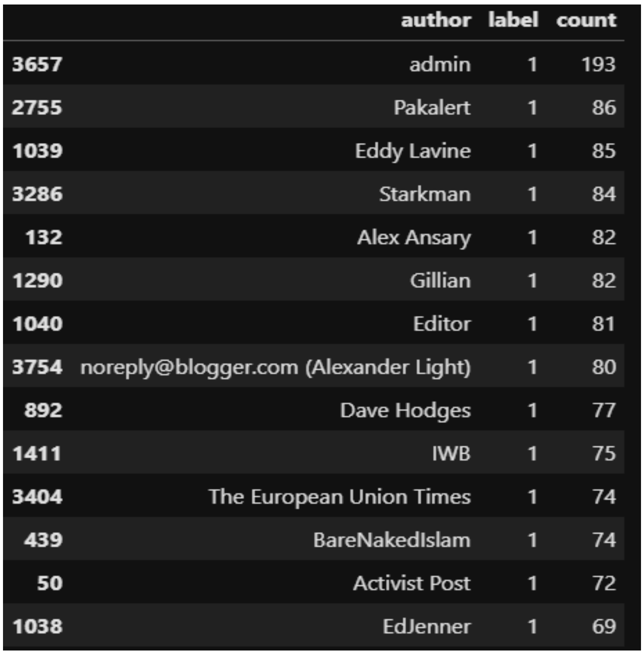


We have three features, namely, title as article title, author as the name of the author who has written the article, and text which is the body of the article. Below is the sample snippet of the dataset:



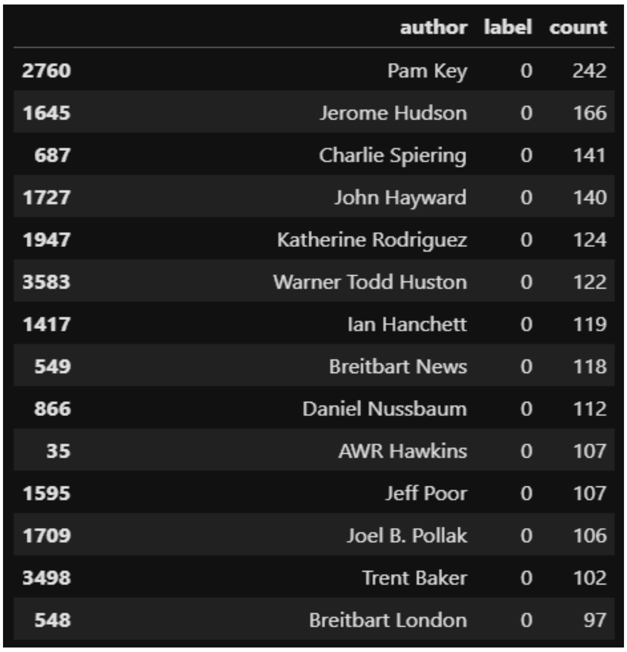
	id	title	author	text	label
0	0	House Dem Aide: We Didn’t Even See Comey’s Let...	Darrell Lucus	House Dem Aide: We Didn’t Even See Comey’s Let...	1
1	1	FLYNN: Hillary Clinton, Big Woman on Campus - ...	Daniel J. Flynn	Ever get the feeling your life circles the rou...	0
2	2	Why the Truth Might Get You Fired	Consortiumnews.com	Why the Truth Might Get You Fired October 29, ...	1
3	3	15 Civilians Killed In Single US Airstrike Hav...	Jessica Purkiss	Videos 15 Civilians Killed In Single US Airstr...	1
4	4	Iranian woman jailed for fictional unpublished...	Howard Portnoy	Print \nAn Iranian woman has been sentenced to...	1

During analysis, it can be seen that the author's name associated with articles with label 1 i.e. fake news is mostly ambiguous in the dataset and mostly alias names were used against the author's name like dmin, Pakalert, Editor, IWB, -NO AUTHOR- and wmw_admin while considering not-fake news(label 0), names of authors were quite clear.



	author	label	count
3657	admin	1	193
2755	Pakalert	1	86
1039	Eddy Lavine	1	85
3286	Starkman	1	84
132	Alex Ansary	1	82
1290	Gillian	1	82
1040	Editor	1	81
3754	noreply@blogger.com (Alexander Light)	1	80
892	Dave Hodges	1	77
1411	IWB	1	75
3404	The European Union Times	1	74
439	BareNakedIslam	1	74
50	Activist Post	1	72
1038	EdJenner	1	69

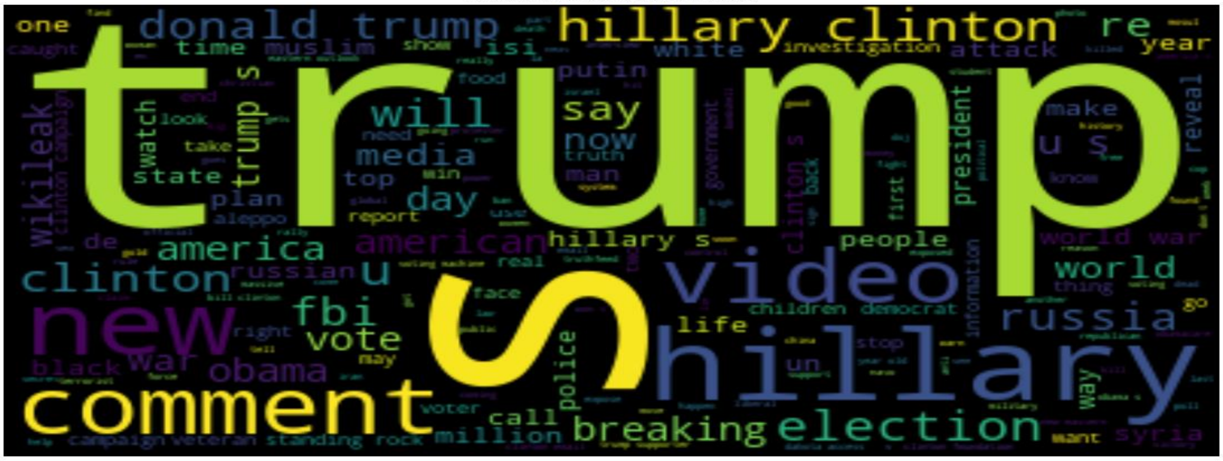
2Author Name in Fake News

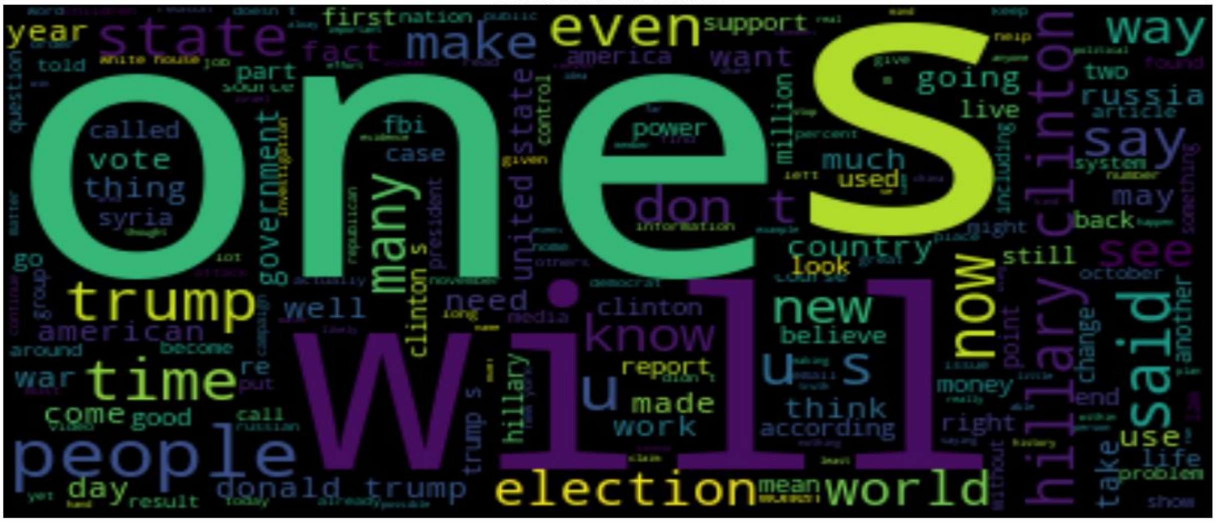


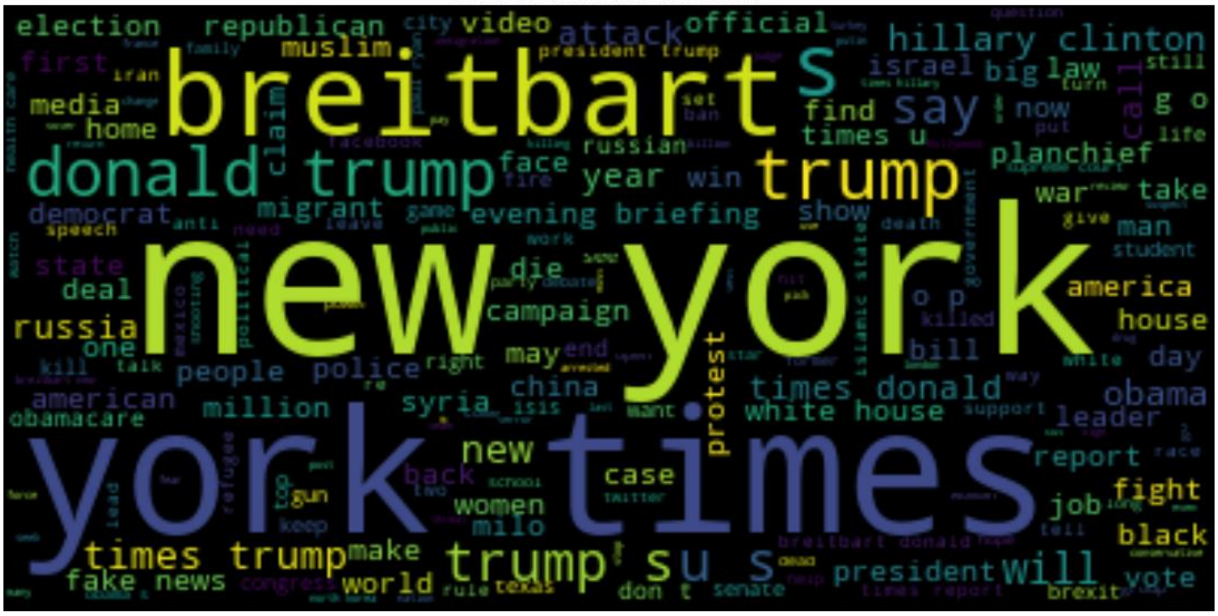
	author	label	count
2760	Pam Key	0	242
1645	Jerome Hudson	0	166
687	Charlie Spiering	0	141
1727	John Hayward	0	140
1947	Katherine Rodriguez	0	124
3583	Warner Todd Huston	0	122
1417	Ian Hanchett	0	119
549	Breitbart News	0	118
866	Daniel Nussbaum	0	112
35	AWR Hawkins	0	107
1595	Jeff Poor	0	107
1709	Joel B. Pollak	0	106
3498	Trent Baker	0	102
548	Breitbart London	0	97

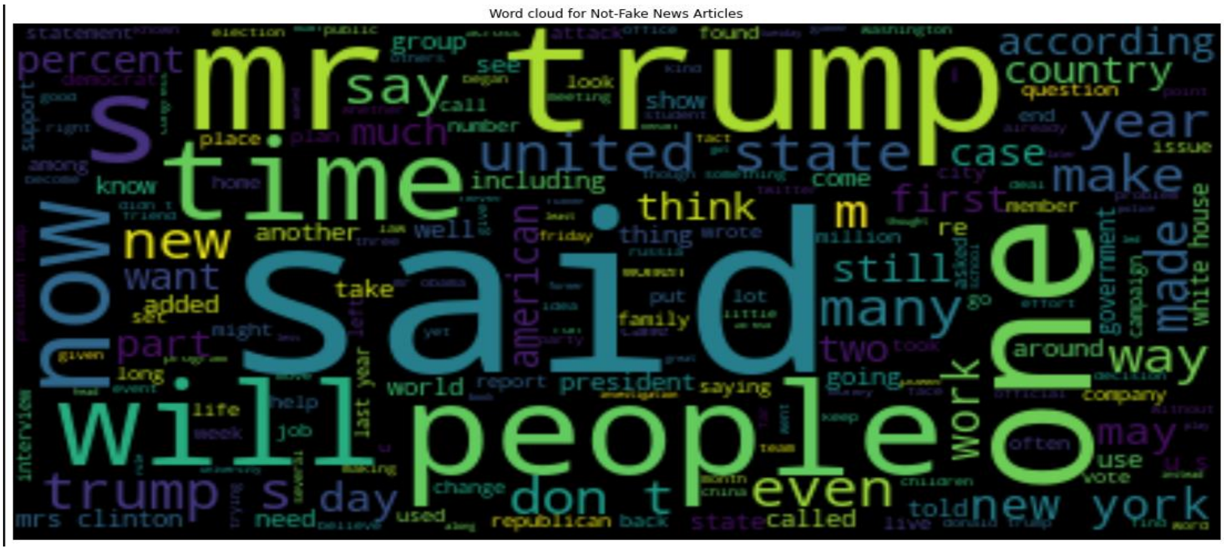
1Authors Name in Not-Fake News

From word cloud, we can see that what these articles are about. Below is the word cloud for articles headline and text (marked as articles in plot label) plotted with respective label values.


Word Cloud for Fake News Articles Headline


Word Cloud for Fake News Articles


Word Cloud for Not Fake News Articles Headline


Word cloud for Not-Fake News Articles

Looking at a word cloud of articles that are labeled not fake are comparatively have less name-calling (only trump's mentions is there in terms of a recognizable name in the word cloud) looks more like a discussion on future promises of things which can be seen in terms of words like make, want, world, etc. As it has been seen previously while looking at author's who have written fake news articles, mostly with them we do not have a clear name, so we do not have a meaningful endpoint associated with them which looked like gave them the freedom to write about anything which helps them in spreading their agenda.

From the word cloud of the dataset, it looks like articles were written during the 2016 United States presidential election.

Method

As a primary step, Null and NA values was filtered from the dataset. Three feature columns, namely, author, title, and text were combined as training data. For Logistic Regression and Random Forest based classifiers, during pre-processing word tokens were generated and grid search with 5-fold cross-validation is used for hyperparameter tuning.

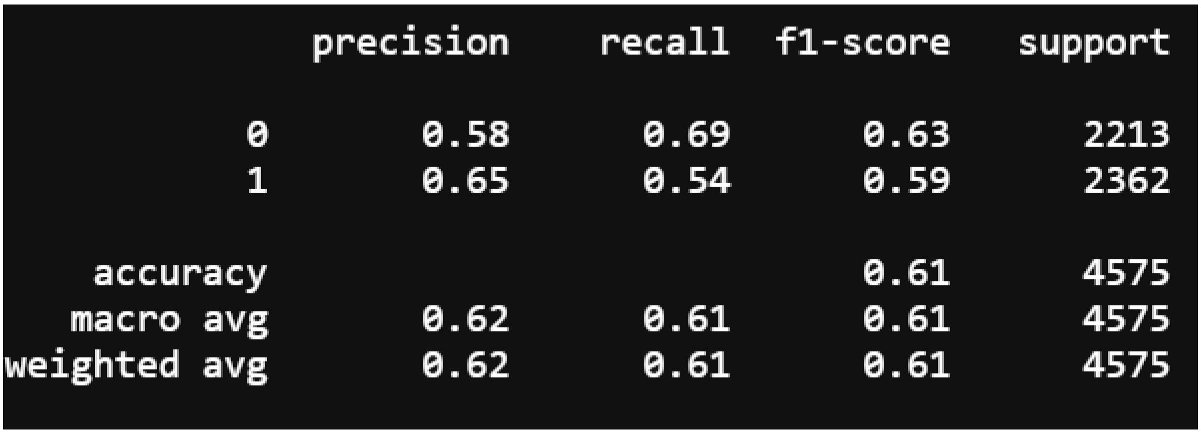
For BERT, during pre-processing non-alphabetic characters were removed and a complete sentence was used for training which was tokenized and embedded using a simple transform[xi] pipeline.

For models, the comparison was made by calculating Precession, Recall, and F1 Score (F-measure).

Results

For the result of this project, the accuracy, precision, recall, and F1 Scores were calculated for each model. Logistic regression has the highest recall score of 0.54 (which is measure of how accurately model is able to detect all true positives from total actual true positives) for fake news class labelled as 1 and lowest training time.

Logistic Regression model with parameter C : 5, max_iteration : 5000 and solver : newton-cg

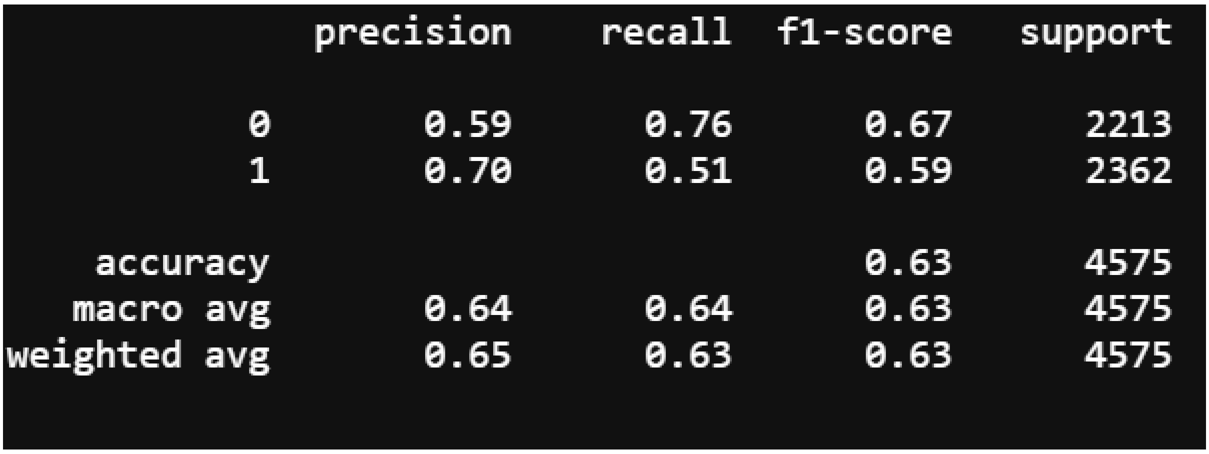


```
              precision    recall  f1-score   support

           0       0.58      0.69      0.63      2213
           1       0.65      0.54      0.59      2362

    accuracy                           0.61      4575
   macro avg       0.62      0.61      0.61      4575
weighted avg       0.62      0.61      0.61      4575
```

Random Forest model with parameter ccp_alpha : 0, criterion : gini and n_estimator : 300

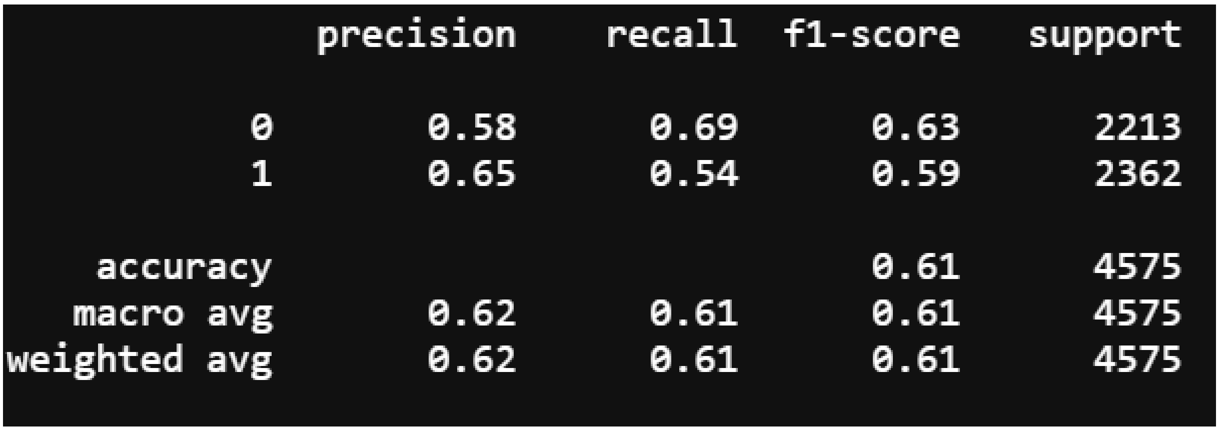


```
              precision    recall  f1-score   support

           0       0.59      0.76      0.67      2213
           1       0.70      0.51      0.59      2362

    accuracy                           0.63      4575
   macro avg       0.64      0.64      0.63      4575
weighted avg       0.65      0.63      0.63      4575
```

BERT with sliding_window estimation as False:

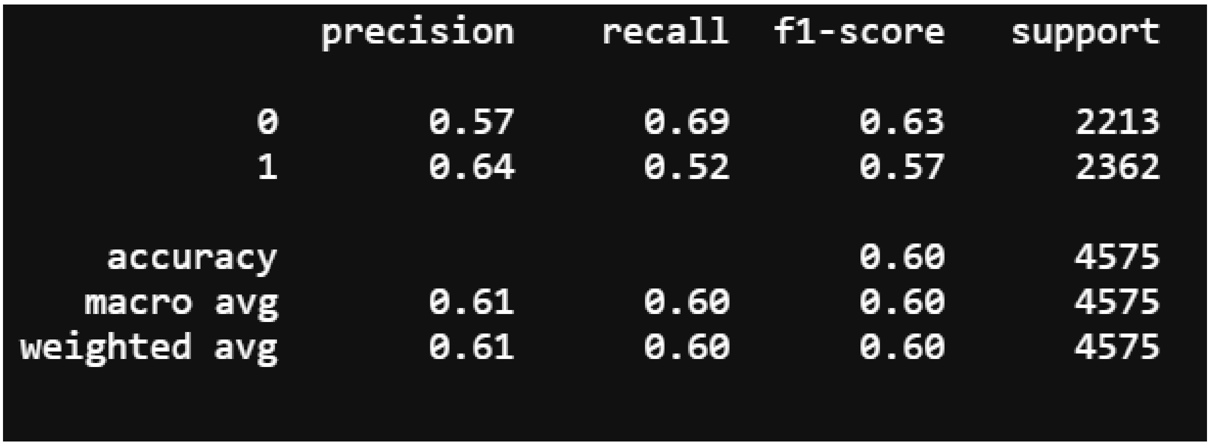


```
              precision    recall  f1-score   support

           0       0.58      0.69      0.63      2213
           1       0.65      0.54      0.59      2362

    accuracy                           0.61      4575
   macro avg       0.62      0.61      0.61      4575
weighted avg       0.62      0.61      0.61      4575
```

BERT with sliding_window estimation as True as we have long sentences and if it is False Simple transformation pipeline truncate longer sequences in our case articles are quite long.



```
              precision    recall  f1-score   support

           0       0.57      0.69      0.63      2213
           1       0.64      0.52      0.57      2362

    accuracy                           0.60      4575
   macro avg       0.61      0.60      0.60      4575
weighted avg       0.61      0.60      0.60      4575
```

Discussion:

With hyperparameter tuning, all three models' accuracy is close to each other but for Logistic Regression, the recall score is highest, also the time required to train logistic regression was lowest while the time required to train Random Forest was highest. BERT's small model was used in this project because of memory and device capability and yet its performance is comparable to Logistic Regression based model.

In terms of model accuracy, for the same dataset people have obtained around 90% accuracy[xii] but most of those models only use two features namely author and article title while ignoring the main section which is article text. Such models will fail in the case when reliable author name and the copied title is used to promote different underline content. Also, the use of recall score is not shown which is extremely useful in such cases as the objective is to identify fake news in the given set of news.

One import factor to consider while developing such model is that change in news trend is extremely fast and also using larger pre-trained models like BERT still need additional final layer training which requires large computation power, considering multiple models have to be developed in real-time scenarios using geo-local information and news. Thus model while selecting a final model recurring training costs also has to be included in the evaluation metrics.

Conclusion:

The project aimed to find the simplest model while producing reliable results. Comparison among three different machine learning model's capability to predict fake news was done and the finding of the project suggests that simple yet powerful logistic regression performance is better when compared with the state of the art deep learning framework BERT (small module) in terms of recall score and time required to train. It is not to claim that Deep learning capability is limited, such a model might perform better with more data or using BERT big module in case of BERT but, under current consideration and data, Logistic Regression is found to outperform other applied methods for the given dataset.

Codes

Code for this project can be found at: https://github.com/amannayak/Fake_News_Classifier.git

References

[i] K. Shu, A. Sliva, S. Wang, J. Tang and H. Liu, "Fake news detection on social media: A data mining perspective", *SIGKDD Explor. Newsl.*, vol. 19, no. 1, pp. 22-36, Sep. 2017, [online] Available: http://doi.acm.org/10.1145/3137597.3137600.

[ii] Wong, J. "Almost all the traffic to fake news sites is from facebook, new data show." *The Medium* (2016).

[iii] Lazer, David MJ, et al. "The science of fake news." *Science* 359.6380 (2018): 1094-1096.

[iv] https://www.bbc.com/future/article/20200512-how-the-news-changes-the-way-we-think-and-behave

[v] Fake News | Kaggle

[vi] https://arxiv.org/pdf/1810.00664.pdf

[vii]https://arxiv.org/pdf/1810.04805.pdf

[viii] https://arxiv.org/abs/1802.05365

[ix] https://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf

[x] https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270#:~:text=How%20BERT%20works,%2Dwords)%20in%20a%20text.&text=As%20opposed%20to%20directional%20models,sequence%20of%20words%20at%20once.

[xi] Simple Transformers

[xii] Fake News | Kaggle