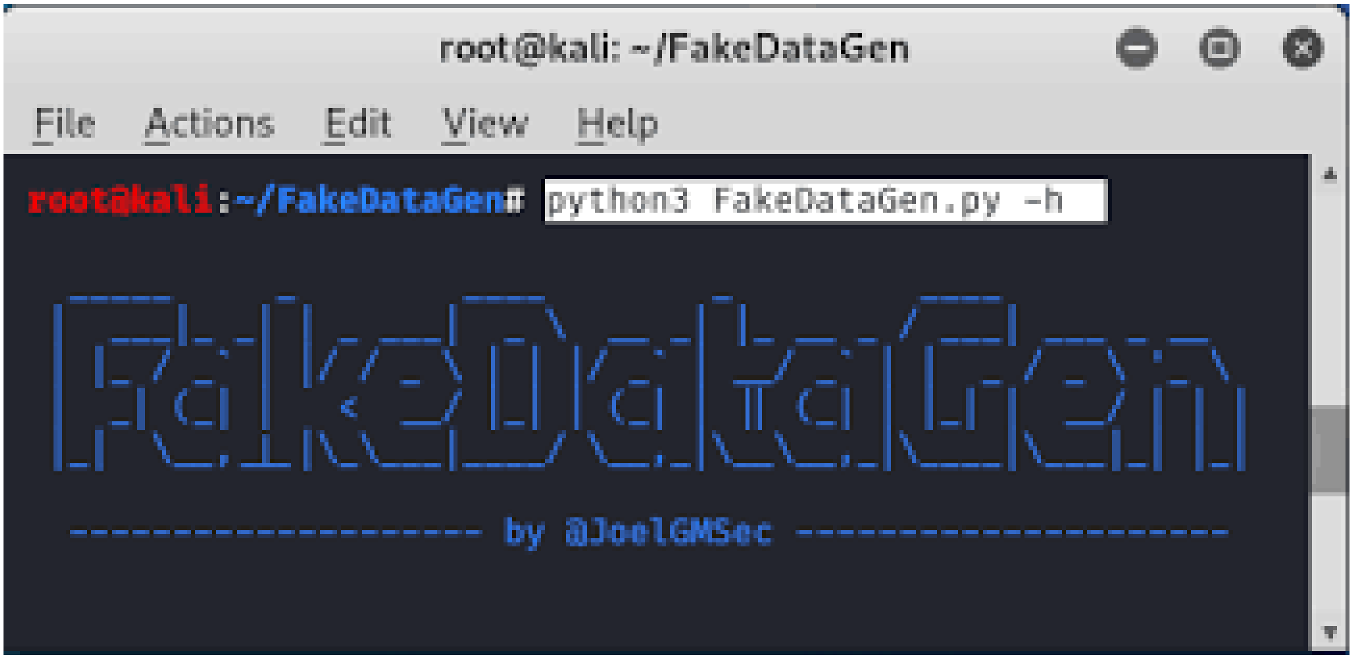
DS 3000 Group Seven Project Proposal

Julian Savini
juliansavini0@gmail.com
Northeastern University
Boston, Massachusetts, USA

Jingxuan Liu
liu.jingx@northeastern.edu
Northeastern University
Boston, Massachusetts, USA

Cici Ling
cicilinggg0502@gmail.com
Northeastern University
Boston, Massachusetts, USA

Miles Vollmer
vollmer.m@northeastern.edu
Northeastern University
Boston, Massachusetts, USA



```
root@kali:~/FakeDataGen# python3 FakeDataGen.py -h

--------------------- by @JoelGMSec ---------------------
```

Figure 1: One of the many sources to help us with creating our own program to detect fake data

ABSTRACT

As social media is continuing to become more immersed into people's lives, so is misinformation and fake news. It's a challenge for journalists to relay information that does not have their own bias in it, and the digital era allows for misinformation to be shared with millions of people within a single day. It is challenging to identify what is not and is fake news, so if a machine learning model could accurately predict whether a news article is reliable or unreliable, that would be very beneficial for what media one chooses to engage with. Referring to DataFlair, DataFlair provides a dataset containing a variety of news articles, including detecting a fake/real news article. In order to create the machine learning model, we will use the materials we have been learning in DS 3000. We want to read in data, and properly clean the data so it is ready for processing. Then, we will explore the data by creating visualizations that will help us further understand the data. Finally, we would like to use a machine learning library, such as sklearn or PyCaret, to create a model which can predict whether news is real or fake. Our deliverables will include information on our data and model on November 7 and 21. Then the poster, data and source code, discussion board posts, and video will be submitted by December 8.

KEYWORDS

Datasets, Machine Learning, Fine-Tuning, Misinformation, False Dataset

Conference'17, July 2017, Washington, DC, USA
© 2022 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

ACM Reference Format:
Julian Savini, Jingxuan Liu, Cici Ling, and Miles Vollmer. 2022. DS 3000 Group Seven Project Proposal. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

The problem we are looking to research is identifying what news is fake and what is real. This is an issue you probably run into everyday if you use social media or browse news on the internet. Recently we have seen some high profile examples of the damage that fake news can cause from the fake news smear campaigns aimed to influence the 2016 and 2020 elections to the swath of misinformation being published during the covid-19 epidemic, you may have heard of people using Ivermectin, a drug commonly used on livestock, to try and treat covid or even prevent it. This was proven to be completely ineffective in treating covid especially when compared with acetaminophen or ibuprofen, but due to fake news articles supporting its use people still took it. We believe using fake news data to create a fake news classifier would be a way to combat this issue. We hope to create a classifier that can accurately predict whether an article is fake or real. With this we hope to provide a simple way to differentiate fake news from real news to give people protection from being tricked by fake news.

2 RELATED WORKS/BACKGROUND

Manzoor, Singla, and Nikita's article, titled "Fake News Detection Using Machine Learning approaches: A Systematic Review" is a paper review on the various machine learning approach models for fake news detection[3]. For example, there are Naive Bayes, Decision trees, SVM, Neural Networks, Random Forest, and XG Boost. It analyzes previous research and identifies types of data in social media posts as well as types of fake news. The paper concludes that FakeDetector should address two main components: representation feature learning and credibility label inference.

Ahmad, Muhammad Yousaf, Suhail Yousaf, and Ovais Ah Ahmad's article, "Fake News Detection Using Machine Learning Ensemble Methods", discusses fake news classification using machine learning models and ensemble techniques [1]. In the end, it mentions open issues that are interesting to explore: identifying key elements involved in the spread of news as well as real-time fake news identification in videos.

DataFlair is a guide to help build a project to test if a piece of the news article is real or fake. DataFlair explained code terminology like TfidfVectorizer, which is able to convert a collection of raw data into TF-IDF features. In the end, DataFlair created a machine-learning model with a 92.82 percent accuracy in magnitude [2].

Piero Paialunga's "Fake News Detection with Machine Learning" discusses how to code Fake News detection using BERT, TensorFlow, and PyCaret [4]. "Fake News Detection with Machine Learning, Using Python" goes over easier and faster Machine Learning to a pre-trained neural network, fine-tune it and obtain state of art results on a dataset. Rendering the power of deep learning, this website was able to obtain an 88 percent accuracy, 88 percent recall and 89 percent precision.

PythonCode is a website that provides a collection of resources and tutorials on python topics to beginner and intermediate programmers. This specific article on "Fake News Detection in Python" by Rockikz and Payong allows readers to explore fake news datasets and build a fake news detector using a transformer library [5]. In the end, they were able to create a model with 99.78 percent accuracy on private and public news.

Lastly, this research paper on "weakly supervised learning for fake news detection on twitter" by Helmstetter and Paulheim developed a machine learning model on fake news detection on twitter with a F1 score of up to 0.9, They used manually collected datasets using twitter API, DMOZ and worked with large datasets. The algorithms they used include Naive Bayes, Decision Trees, and Support Vector Machines. They also utilized two ensemble methods - Random Forest and XG-Boost. In conclusion, they showed that despite using an inaccurate dataset, it is possible to detect fake news with an F1 score of up to 0.9.

There are many more related sources to this project, where all of thse sources can help shape our Fake News Detector program project. With all of these information, hopefully we can code a program that would help detect at least 98 percent of accuracy in differentiating fake and real news in that dataset that we obtain from the article "Detecting Fake News with Python and Machine Learning".

3 PROPOSED TIMELINE

Phase 3 Model selection, Training, and Evaluation - Due 11/21

- Finish data wrangling and find solutions to threats found in phase 2 - by 11/11
- Write up data wrangling - by 11/14
- Find suitable tests and have plans for future data analysis - by 11/18
- Write up for future plans and explanation of tests choice - by 11/21

Phase 4 Results, Documentation, and Potential Impacts - Due 12/9

- Update previous sections if necessary - by 11/25
- Add more details to all sections as more work is done - by 11/29
- Wrap up analysis and report findings - by 12/4
- Create a concise poster - by 12/9

4 METHODOLOGY

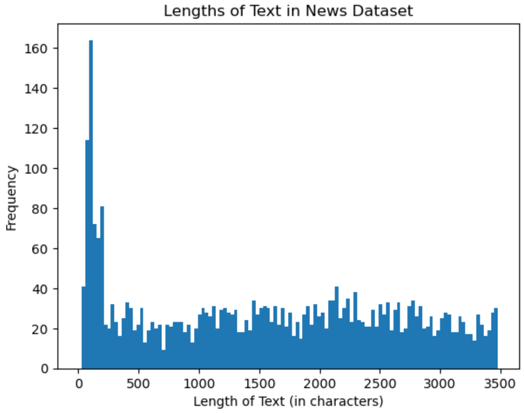
We obtained our dataset from an article called "Detecting Fake News with Python and Machine Learning" by DataFlair. This data source file is called news.csv and has a shape of 7796x4. The dataset contains the title of news articles, the text of news articles, and whether each news article is real or fake. Furthermore, there is a News ID column which has corresponding numbers relating to each news source.

This data source directly relates to the problem because we want to design a machine learning model that will identify a news source that is either real or fake. The dataset identifies which articles were real or fake. So therefore, we can use classification to design our machine learning model.

To summarize the data preparation, we looked at a lot of different websites. Since there has been other research and attempts to do this project, there were a few choices to choose from. First, we wanted a list of the different title articles. Second, we needed the text from the articles or else we would not be able to see if the article was real or fake. We think these are the most important requirements needed for our data set and new.csv provided these requirements in the simplest form.

50.41% of the news articles are real. 49.59% of the news articles are fake. The mean number of characters in each article is 4420.26 The median is 3518, and the standard deviation is 4103.73. The maximum is 32509, and the minimum is 50.

Null values were removed from the dataset, and any articles that contained less than fifty words were also removed, because having more text data in an article improves a model's accuracy. Removing all null values from the dataset reduced the number of rows in our dataset from 7795 to 6754. After removing text with less than fifty characters, the size of the dataset reduced to 6432 rows.



One concern with this dataset is that in our EDA phase, we may have difficulty creating informative visualizations. Since the news website sources are represented by an identification number, we cannot know the names of the news sources, so we cannot make interesting visualizations with that. The next two columns contain the title and text, so we could work with common words or lengths of the text. In addition, another concern is determining the accuracy level of each news article since we do not necessarily know how DataFlair determined if the news was real or fake. Going through about 8,000 articles is a lot of data and at times, the labeler may have miscalculated if the article is fake or real. For future research, we would like to work alongside information analysts, who could create our data set, so we have an understanding of what goes into the process of determining an article is real or fake. Thus, we would have full confidence if our articles were properly labeled.

In order to conduct text classification, we need to convert the text columns into numerical feature vectors. We can use the bags of word model and segment texts into words and count the number of times each word occurs in each news article title as well as assigning each word to an integer id. To begin with, we will use CountVectorizer to count the words inside of our dataset news column and return us with a Document-Term matrix as n_samples and n_features. Then, to reduce the weightage of common words, we will use TF and TF-IDF. Lastly, for the algorithms, we will be using Naïve Bayes, Support Vector Machines, and Random Forest. The Naïve Bayes model is simple to use and good for larger data sets. It calculates each tag's probability for a given text and outputs the one with the highest chance. The Random Forest Model consists of multiple decision trees and predicts an outcome by taking the decision trees' mean output. As we have a sufficient amount of data, the accuracy of prediction would increase because of that. The Support Vector Machines Models have great speed and they calculate prediction using an expensive five-fold cross-validation.

The model will be trained using sklearn's train_test_split function, and then we will test 10% of the data, and will make sure to not randomize the training and testing data while testing different models, so results are consistent. We will evaluate the models by using the score function, which will tell us the accuracy of the model. The model with the highest accuracy will be used for hyperparameter testing.

In our dataset, the parameters that will be tuned are 'vect_ngram_range', 'tfidf_use_idf', and 'clf-svm_alpha'. As we are using hyperparameter tuning and cleaning our dataset, we expect our accuracy level to be greater than DataFlair's accuracy level of 92.8%. We expect our dataset to be much easier to read since the information is cut down.

5 CITATIONS

A paginated journal article [1, 4**?**], World wide web resource [5, 6], Online citations: [2, 5]

ACKNOWLEDGMENTS

This assignment is for Professor Albert Lionelle.

REFERENCES

[1] Iftikhar Ahmad, Muhammad Yousaf, Suhail Yousad, and Muhammad Ovais Ahmad. 2020. Fake News Detection Using Machine Learning Ensemble Methods. 2020, 2 (2020), 11 pages. https://www.hindawi.com/journals/complexity/2020/8885861/

[2] DataFlair 2021. *Detecting Fake News with Python and Machine Learning.* Retrieved Oct 15, 2022 from https://data-flair.training/blogs/advanced-python-project-detecting-fake-news/

[3]]Paulheim Stefan Helmstetter and Heiko Paulheim year = 2018 title =. [n. d.]. ([n. d.]).

[4] S.I. Manzoor, J. Singla, and Nikita. 2019. Fake News Detection Using Machine Learning approaches: A systematic Review. 1 (2019), 230–234. https://doi.org/10.1109/ICOEI.2019.8862770

[5] Piero Paialunga. 2022. *Fake News Detection with Machine Learning, Using Python.* Retrieved Oct 14, 2022 from https://towardsdatascience.com/fake-news-detection-with-machine-learning-using-python-3347d9899ad1

[6] Abdou Rockikz and Adrien Payong. 2022. *Fake News Detection in Python.* Retrieved Oct 15, 2022 from https://www.thepythoncode.com/article/fake-news-classification-in-python