Fake News Detection Using Machine Learning

1st Sujoy Paul Dakkumalla 2nd Venkata Sai Varun Mooraboina 3rd Pranavi Guttikonda 4th Sai Kaushik peesari

***Abstract*—A lot of things read online, especially in the social media feeds, may appear to be true, often is not. Fake news is news, stories or hoaxes created to deliberately mislead or deceive readers. Usually, these stories are created to either influence people's views, push a political agenda or cause confusion and can often be a profitable business for online publishers. The dissemination of fake news in today's digital world has effected beyond a specific group. Mixing both believable and unbelievable information on social media has made the confusion of truth. That is, the truth will be hardly classified. This paper comes up with the machine learning techniques for discerning the 'fake news', which is the misguiding news that is being published through unknown sources**

***Index Terms*—Machine Learning, Fake news, Natural Language Processing, online publishers, confusion of truth**

I. Introduction

Fake news has quickly become a society problem, being used to propagate false or rumour information in order to change peoples behaviour. It has been shown that propagation of fake news has had a non - negligible influence of 2016 US presidential elections[2]. A few facts on fake news in the United States: • 62% of US citizens get their news for social medias[3] • Fake news had more share on Facebook than mainstream news[4]. Fake news has also been used in order to influence the referendum in the United Kingdom for the "Brexit". In this paper I experiment the possibility to detect fake news based only on textual information by applying traditional machine learning techniques[5, 6, 7] as well as bidirectionalLSTM[8] and attention mechanism[1] on two different datasets that contain different kinds of news. In order to work on fake news detection, it is important to understand what is fake news and how they are characterized. The following is based on Fake News Detection on Social Media: A Data Mining Perspective[9]. The first is characterization or what is fake news and the second is detection. In order to build detection models, it is need to start by characterization, indeed, it is need to understand what is fake news before trying to detect them.

Fake news definition is made of two parts: authenticity and intent. Authenticity means that fake news content false information that can be verified as such, which means that conspiracy theory is not included in fake news as there are difficult to be proven true or false in most cases. The second part, intent, means that the false information has been written with the goal of misleading the reader.

A. News Content Models

1) Knowledge - based models: : Now that fake news has been defined and the target has been set, it is needed to analyse what features can be used in order to classify fake news. Starting by looking at news content, it can be seen that it is made of four principal raw components:

- Source: Where does the news come from, who wrote it, is this source reliable or not.
- Headline: Short summary of the news content that try to attract the reader.
- Body Text : The actual text content of the news.
- Image/Video: Usualy, textual information is agremented with visual information such as images, videos or audio

Features will be extracted from these four basic components, with the mains features being linguistic - based and visual - based. As explained before, fake news is used to influence the consumer, and in order to do that, they often use a specific language in order to attract the readers. On the other hand, non - fake news will mostly stick to a different language register, being more formal. This is linguistic - based features, to which can be added lexical features such as the total number of words, frequency of large words or unique words.

II. Related work

There are two main categories of state of the art that are interesting for this work: previous work on fake news detection and on general text classification. Works on fake news detection is almost inexistent and mainly focus in 2016 US presidential elections or does not use the same features. That is, when this work focus on automatic features extraction using machine learning and deep learning, other works make use of hand - crafted features[12, 13] such as psycholinguistic features[14] which are not the goal here.

Current research focus mostly on using social features and speaker info rmation in order to improve the quality of classifications. Ruchansky et al.[15] proposed a hybrid deep model for fake news detection making use of multiple kinds of feature such as temporal engagement between n users and m news articles over time and produce a label for fake news categorization but as well a score for suspicious users.

Tacchini et al.[16] proposed a method based on social network information such as likes and users in order to find hoax information. Thorne et al.[17] proposed a stacked

https://github.com/joy1302/FakeNewsDetection
https://github.com/SaiVarun1125/Fake-News-Detection
https://github.com/Tejaswi521/ML-Final-Project
https://github.com/SXP42750/Fake$_{N}ews_{d}etection$

ensemble classifier in order to address a subproblem of fake news detection which is stance classification. It is the fact of finding if an article agree, disagree or simply discus a fact. Granik and Mesyura[18] used Na ıv - Bayes classifier in order to classify news from buzzfeed datasets. In addition to texts and social features, Yang et al.[19] used visual features such as images with a convolutional neural network . Wang et al.[20] also used visual features for classifying fake news but uses adversarial neural networks to do so.

III. Motivation

In the era of news in our lives, it is the people's responsibility not to share any misleading information as there are many sources available nowadays. Fraud news such as spam messages, funding news, or any false information falls out or reaches people we consider a serious issue although it is extremely complicated to find out which is fraud and which is not a fraud profile or users in social media, they replicate the information as the original one. As the technology evolved and machine intelligence has came into existence everyone tends to use available sources for creating and disseminating fraudulent news. People who are illiterate might be new to digital media as they are inexperienced, so they are the ones who believe that fraudulent news easily and makes it practical in their lives. To a minimum, we have deviled simple weblitin which statistically detects false information, and also real news.

IV. Objectives

This paper is focused on evaluating different Machine Learning Algorithms for the Detection of Fake News using the supplied dataset.

- To preprocess the data effectively in order to train the machine learning models, such as treating the missing values or null values.
- To implement and evaluate the classification models available in machine learning for the considered fake news dataset.

V. Proposed Frame Work

A good starting point for the analysis is to make some data exploration of the data set. The first thing to be done is statistical analysis such as counting the number of texts per class or counting the number of words per sentence. Then it is possible to try to get an insight of the data distribution by making dimensionality reduction and plotting data in 2D.

A. Dataset Collection

The dataset is Liar, Liar Pants on Fire dataset[30], which is a collection of twelve thousand small sentences collected from various sources and hand labelled.

It should be noted that this one differs from the two other datasets is it is composed only on short sentences, and thus it should not be expected to have very good results on this dataset for models trained on Fake News Corpus which is made of full texts. In addition, the texts from the latest dataset are more politically oriented than the ones from the first one.

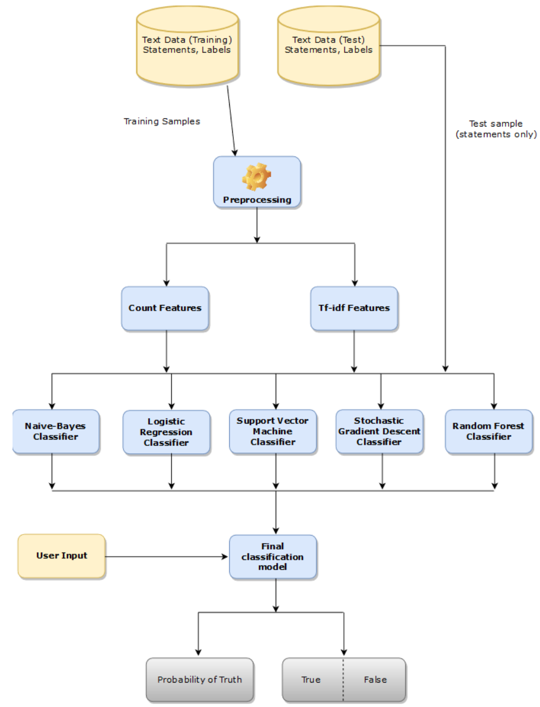

Fig. 1.

B. Analyzing the Data

In order to visualize the data, it is needed to transform text in a numerical way and eventually reduce the dimension in order to allow it to be plotted on a 2D or 3D plot. Here TF-IDF (term frequency, inverse document frequency[6, 33]) is used. How it works will be details later on. This produces a sparse matrix with each document being represented as an array, each value of the array being a value for one term.

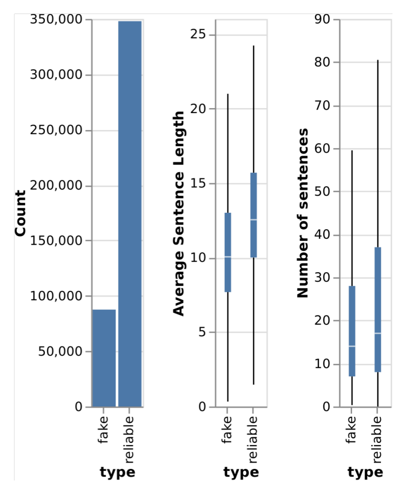

Fig. 2.

C. Text to vectors

As explained before, text needs to be represented in a way that gives more meaningful information than a simple

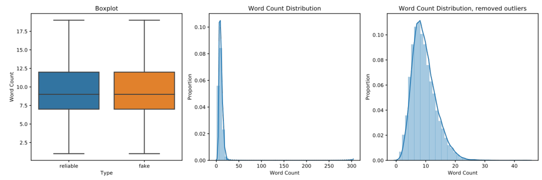

Fig. 3.

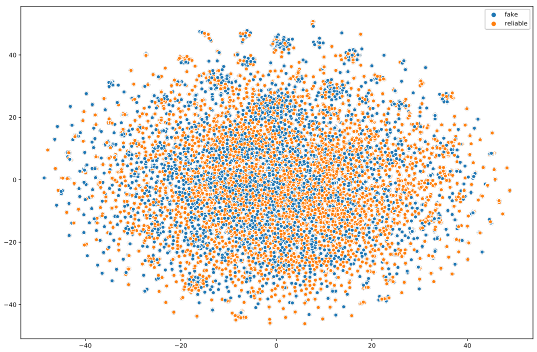

Fig. 4. t-SNE plot for liar dataset.s

sequence of bits, which have additional drawbacks such that for a given word, the sequence of bits representing it depends on the coding. The first and simplest coding that comes to mind is a one-hot encoding: a matrix M of size number of texts × number of words where Mij = 1 if the word j is present in the text i and 0 in the other case. But this is still not enough as each word is given the same weight, no matter how often it appears in the text.

Creating the tf - df frequency after bagging the words to summarize the tf-idf ngram:

```
#create tf-df frequency features
#tf-idf
tfidfV = TfidfTransformer()
train_tfidf = tfidfV.fit_transform(train_count)

def get_tfidf_stats():
    train_tfidf.shape
    #get train data feature names
    print(train_tfidf.A[:10])


#bag of words - with n-grams
#countV_ngram = CountVectorizer(ngram_range=(1,3),stop_words='english')
#tfidf_ngram  = TfidfTransformer(use_idf=True,smooth_idf=True)

tfidf_ngram = TfidfVectorizer(stop_words='english',ngram_range=(1,4),use_idf=True,smooth_idf=True)
```

Fig. 5.

Now the POS tagger is applied on the training sentences:

```
#POS Tagging
tagged_sentences = nltk.corpus.treebank.tagged_sents()

cutoff = int(.75 * len(tagged_sentences))
training_sentences = DataPrep.train_news['Statement']

print(training_sentences)

#training POS tagger based on words
def features(sentence, index):
    """ sentence: [w1, w2, ...], index: the index of the word """
    return {
        'word': sentence[index],
        'is_first': index == 0,
        'is_last': index == len(sentence) - 1,
        'is_capitalized': sentence[index][0].upper() == sentence[index][0],
        'is_all_caps': sentence[index].upper() == sentence[index],
        'is_all_lower': sentence[index].lower() == sentence[index],
        'prefix-1': sentence[index][0],
        'prefix-2': sentence[index][:2],
        'prefix-3': sentence[index][:3],
        'suffix-1': sentence[index][-1],
        'suffix-2': sentence[index][-2:],
        'suffix-3': sentence[index][-3:],
        'prev_word': '' if index == 0 else sentence[index - 1],
        'next_word': '' if index == len(sentence) - 1 else sentence[index + 1],
        'has_hyphen': '-' in sentence[index],
        'is_numeric': sentence[index].isdigit(),
        'capitals_inside': sentence[index][1:].lower() != sentence[index][1:]
    }
```

Fig. 6.

D. Model building

Here in this section we used sklearn module from python to implement the machine learning algorithms.

1) Naive-Bayes: The basic idea of Naïve-Bayes model is that all features are independent of each other. This is a particularly strong hypothesis in the case of text classification because it supposes that words are not related to each other. But it knows to work well given this hypothesis. Given an element of class y and vector of features X = (x1,...,xn).

2) Linear SVM: Linear SVM is a method for large linear classification. Given pairs of features-label (xi,yi),yi 1,1, it solves the following unconstrained optimization problem.

3) Decision Tree: Decision tree works by recursively selecting features and splitting the dataset on those features. These features can either be nominal or continuous.

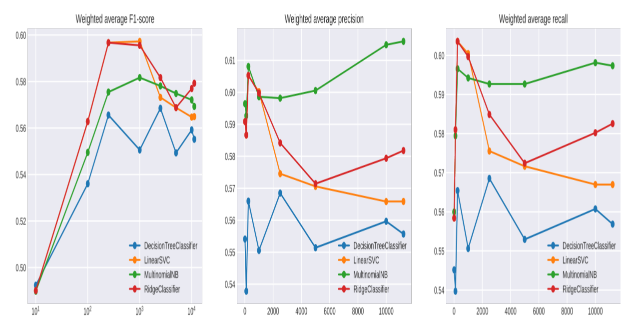

Fig. 7. Model comparison

E. Outputs

Here are the output screenshots for our machine-learning fake news prediction system in the sequence:

The command to check the news fake or not is-“ python -W ignore prediction.py ”

Then, Now, user gives the input to check whether it is fake news or not.

The resultant looks like “ The given statement is true” likewise we can execute and the whether it is fake news or not.

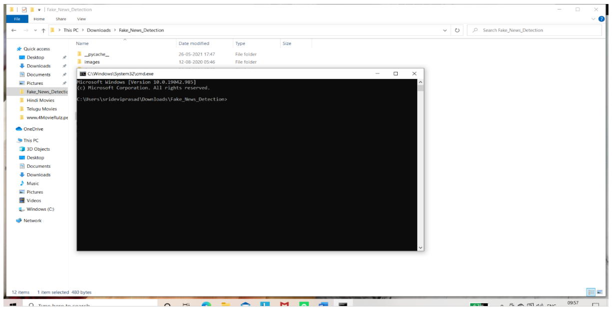

Fig. 8.

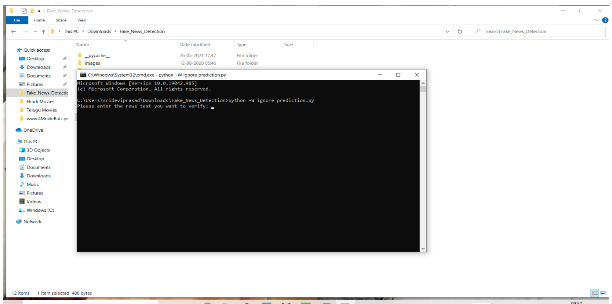

Fig. 9.

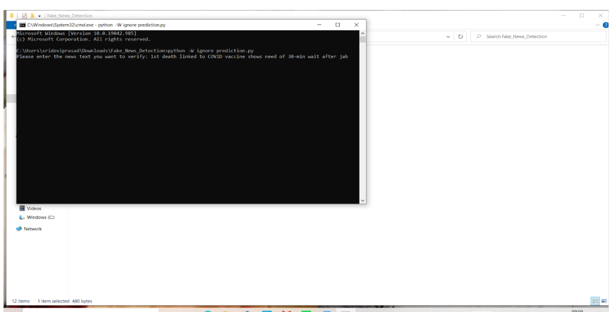

Fig. 10.

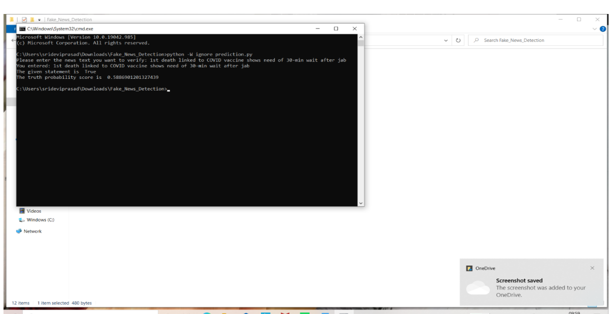

Fig. 11.

Conclusion

Some hypotheses can be made on why same models works very well on one dataset and does not work well on the other one. The first thing we can think of is that the original hypothesis on different styles of writing between fake and reliable news is only verified in one dataset, the Fake News Corpus, and it is the most logical one, as these texts are coming from online newspapers (or pretending to be), and thus capitalize on advertisements for making money. The dataset, Liar-Liar Corpus is described by its authors as a collection a short sentence coming from various contexts such as political debate, interviews, TV ads and so on, thus it induces a lot of variety in writing style. For instance, it contains a transcription of vocal messages, which have in essence a different style from written one. The data exploration chapter had already given an insight about this fact, as 2D data projection of the Liar-Liar Corpus shows no clear sign of separation, when Fake News Corpus shows one at the first look.

References

[1] [1]Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. In Journal of Economic Perspective, volume 31, 2017.

[2] Jeffrey Gottfried and Elisa Shearer. News Use Across Social Medial Platforms 2016. Pew Research Center

[3] Craig Silverman and Lawrence Alexander. How teens in the balkans are duping trump supporters with fake news. Buzzfeed News,

[4] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. J. Mach. Learn. Res., 9:1871–1874,

[5] Stephen Robertson. Understanding inverse document frequency: On theoretical arguments for idf,

[6] [6]Harry Zhang. The Optimality of Naive Bayes. page 6.

[7] Sepp Hochreiter and Ju¨rgen Schmidhuber. Long short-term memory. Neural Computation, 9:1735–1780.

[8] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. ACM SIGKDD Explorations Newsletter, 19(1):22–36,

[9] WWF. Wwf 10yearschallenge,

[10] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. Journal of machine Learning research, 3(Jan):993–1022

[11] Julio CS Reis, Andr´e Correia, Fabr´ıcio Murai, Adriano Veloso, Fabr´ıcio Benevenuto, and Erik Cambria. Supervised learning for fake news detection. IEEE Intelligent Systems, 34(2):76–81,

[12] Natali Ruchansky, Sungyong Seo, and Yan Liu. Csi: A hybrid deep model for fake news detection. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pages 797–806. ACM,

[13] Ahmad M., Aftab S., Ali I. Sentiment Analysis of Tweets using SVM. Int. J. Comput. Appl. 2017;177:25–29. doi: 10.5120/ijca2017915758. [CrossRef] [Google Scholar]

[14] [15]Eugenio Tacchini, Gabriele Ballarin, Marco L. Della Vedova, Stefano Moret, and Luca de Alfaro. Some like it hoax: Automated fake news detection in social networks.

[15] James Thorne, Mingjie Chen, Giorgos Myrianthous, Jiashu Pu, Xiaoxuan Wang, and Andreas Vlachos. Fake news stance detection using stacked ensemble of classifiers. In Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism, pages 80–83,

[16] Mykhailo Granik and Volodymyr Mesyura. Fake news detection using naive bayes classifier. In 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), pages 900–903. IEEE

[17] Yang Yang, Lei Zheng, Jiawei Zhang, Qingcai Cui, Zhoujun Li, and Philip S. Yu. Ti-cnn: Convolutional neural networks for fake news detection

[18] [23]Takeru Miyato, Andrew M. Dai, and Ian Goodfellow. Adversarial Training Methods for Semi-Supervised Text Classification. arXiv:1605.07725 [cs, stat], May 2016. arXiv: 1605.07725

[19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12:2825–2830

[20] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space