FAKE NEWS DETECTION

MAJOR REPORT

Submitted by:
Harshit Ahuja (9917103210)
Akshay Sharma (9917103224)
Chirag Khera (9917103231)

Under the supervision of:
Dr. Shailesh Kumar

Submitted in partial fulfilment of the degree of
Bachelor in Technology
in
Computer Science Engineering

Department of Computer Science Engineering/ Information Technology
Jaypee Institute of Information Technology

December 2020

DECLARATION

We hereby declare that this submission is our own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgement has been made in the text.

Place:

Date:

Signature

Harshit Ahuja

9917103210

Akshay Sharma

9917103224

Chirag Khera

9917103231

CERTIFICATE

This is to certify that the work titled **"Fake News Detection"** submitted by **"Harshit Ahuja", "Akshay Sharma"**, and **"Chirag Khera"** in partial fulfilment of the degree B.Tech of Jaypee Institute of Information Technology, Noida has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

Signature of supervisor

Name of Supervisor: Dr Shailesh Kumar

Designation: Assistant Professor

Date:

ACKNOWLEDGEMENT

We would like to express our deepest appreciation to Dr Shailesh Kumar, Assistant Professor, Jaypee Institute of Information Technology, Noida whose contribution in simulating suggestions and encouragement, helped us to coordinate our project.

Furthermore, we would also like to acknowledge with much appreciation the crucial role of Dr Devpriya Soni and Mr. Bansidhar Joshi, Department of CSE/IT, Jaypee Institute of Information Technology, Noida for the extended help provided to us. A special thanks to my team mates and my classmates for their valuable suggestions.

Harshit Ahuja

9917103210

Akshay Sharma

9917103224

Chirag Khera

9917103231

SUMMARY

Fake news spreads like a wild fire. People unknowingly share news that they find different without knowing the truth. The agenda behind creating such news can be monetary, political or economic. When someone or something such as a bot try to impersonate someone or some reliable source to spread false information, this can also be considered as fake news.

From our empirical research we have observed and analysed that an automated system sometimes identifies fake news articles better than human do. The automated systems can play an important tool for identifying fake stories, articles, blogs, clicks which manipulate public opinion on social networking sites.

The research in this area is based on two categories of algorithms, text-based, image-based and web scraping based.

Taking into need for the development of such fake news detection system, in this paper we have identified news articles as fake or real by using supervised machine learning classifiers such as Linear Regression (LR), Decision Tree (DT), Random Forest (RF). The feature extraction techniques, Term Frequency–Inverse Document Frequency (TF–IDF), Count-Vectorizer (CV) are used to extract feature vectors from the textual content of articles. The effectiveness of a set of classifiers is observed when they predict the labels of the testing samples after learning the training data using these features extraction techniques. Various supervised ML models are extensively used for categorization of textual data as fake or real.

In order to test a machine learning algorithms, we define two different datasets: training dataset(80%) and testing dataset(20%)

We successfully implemented a machine learning and natural language processing model to detect whether an article was fake or fact. We a total of 73205 articles scraped from our web crawler for which we trained our models. Our models provides accuracy as Logistic Regression (LR) 97.7 %, Decision Tree(DT) 93.64% and Random Forest (RF) 95.2% .When doing such a classification, it is important to check that we limit the number of false positives as they can cause facts to be marked as fake. Text analytics and NLP can be used to work with the very important problem of fake news. We have seen the big impact they can have on people's opinions, and the way the world thinks or sees a topic.

LIST OF FIGURES

LIST OF SYMBOLS AND ACRONYMS

- LR – Logistic Regression
- DT – Decision Tree
- RF – Random Forest
- TF – Term Frequency
- IDF – Inverse Document Frequency
- ML – Machine Learning
- CV – Count Vectorizer
- BoW – Bag of Words

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

1.1 General Introduction

In the modern age, the power of social media and the internet cannot be neglected. The traditional newspapers are getting replaced by websites and mobile applications. The world is open to sharing information. But with so much power and ease of sharing, it is getting increasingly difficult to spot fake news. Fake news is the news that is fabricated to benefit the author, the articles that present unverified facts just to get page views. Fake news articles are clickbait. The headlines are written in such a way that they seem irresistible.

1.2 Problem Statement

Fake news spreads like a wild fire. People unknowingly share news that they find different without knowing the truth. The agenda behind creating such news can be monetary, political or economic. When someone or something such as a bot try to impersonate someone or some reliable source to spread false information, this can also be considered as fake news.

People believe fake news too easily. They do not always check the facts and sources of such information. This leads to a major problem that needs to be solved. The motto of the hacker/ bot is fulfilled when someone clicks the news or if someone shares them. News has a major impact on political and economic issues. A false news can make or break the image of a political party. It also has the power to impact the economy.

1.3 Significance/Novelty of problem

Companies, hackers and bots create fake news to benefit themselves. They try and influence people. People can become victim to cyber-crimes, frauds and can end up losing money. Sometimes, they even lose their right to choose. Fake news can influence the taste of the people and their choices. Companies are investing heavily in detecting fake news. They do not want someone to use their names to commit frauds. This brings negative impact in the company's reputation. Also, Facebook, Instagram, Twitter and other social media platforms are becoming a hub to fake news. If a fake news can be detected by a machine then it can automatically pop-up a warning against opening such websites or web-links. Upon clicking on such links, the imposter might earn by monetizing through the website or YouTube video but, the consumer of the information gets adversely affected. They can fall prey to the false information and end up losing data privacy or money.

The fake news detection is a way that can protect people from acquiring false information and can safeguard their privacy and hard-earned money.

1.4 Empirical Study

From our empirical research we have observed and analysed that an automated system sometimes identifies fake news articles better than human do. The automated systems can play an important tool for identifying fake stories, articles, blogs, clicks which manipulate public opinion on social networking sites.

1.5 Brief Description of the solution

Taking into need for the development of such fake news detection system, in this paper we have identified news articles as fake or real by using supervised machine learning classifiers such as Linear Regression (LR), Decision Tree (DT), Random Forest (RF). The feature extraction techniques, Term Frequency–Inverse Document Frequency (TF–IDF), Count-Vectorizer (CV) are used to extract feature vectors from the textual content of articles. The effectiveness of a set of classifiers is observed when they predict the labels of the testing samples after learning the training data using these features extraction techniques. Various supervised ML models are extensively used for categorization of textual data as fake or real. Our contribution to this paper is as follows:

- Building a web-crawler for collection of dataset from various fake news websites as well as genuine websites for real news articles. Scrapped data then merged with data we get from Kaggle.
- Statistical analysis of collected datasets with negative and positive instances.
- Three ML models are evaluated using TF-IDF and CV feature extraction techniques to retrieve the best model based on performance metrics.
- A comparative study is performed to show the effectiveness of proposed models.

CHAPTER-2: LITERATURE REVIEW

2.1 Summary of papers studied

Fake/ deceptive news can be classified into three categories, namely:

1. Serious fabrications (Type A, Figure 1A)
2. Large-scale hoaxes (Type B, Figure 1B)
3. Humorous fakes (Type C)

The circulation of fake news not only jeopardizes news industry but puts a negative impact on the user's mind and they tend to believe all the information they read online. Numerous researches are being carried out to help detect fake news so that it isn't spread amongst a wider audience. Social media is the home to such news. Facebook came up with a flag feature to highlight the information that is not verified and chances are that it is false.

The research in this area is based on two categories of algorithms, text-based, image-based and web scraping based.

Figure 1. Three Types of Fake News Form Three Sub-Tasks in Fake News Detection: A) exposed fabrications (Shingler, 2015); B) large-scale hoaxes (Matt, 2015); C) news satire (Fan Duel, 2015).

2.1.1 Grammatical incorrectness of news articles

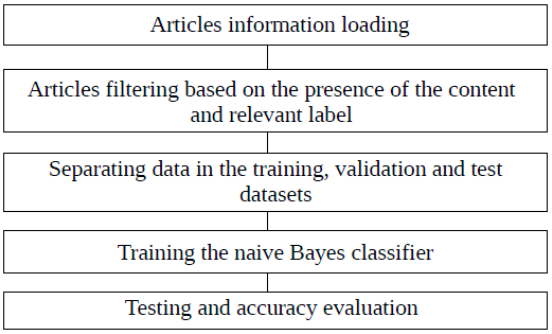

Figure 2. Workflow for identifying spam messages/ fake news using Naïve Bayes Classifier

The main idea is to treat each word of the news article independently. The formula for calculating the conditional probability of the fact, that news article is fake given that it contains some specific word looks as following:

$P(F|W) = P(W|F) \cdot P(F) / (P(W|F) \cdot P(F) + P(W|T) \cdot P(T))$,

where:

P(F|W) – conditional probability, that a news article is fake

given that word W appears in it;

P(W|F) – conditional probability of finding word W in fake news articles;

P(F) – overall probability that given news article is fake news article;

P(W|T) – conditional probability of finding word W in true news articles;

P(T) – overall probability that given news article is true news article.

The next step probabilities to get the probability of the fact, that given news article is fake.

The classification accuracy for true news articles and false news articles is roughly the same, but classification accuracy for fake news is slightly worse. This may be caused by the skewness of the dataset: only 4.9% of it are fake news.

2.1.2 Features for Fake News Detection

Language Features – Sentence-level features, including bag of words approach, 'n-gram' and parts of speech(POS tagging), number of words and syllables per sentence are evaluated.

Lexical Features – Typical lexical features are character. unique words and their frequencies, pronouns, verbs and punctuation count.

Psycholinguistic Features – Word count is a dictionary-based text mining software. Use of persuasive and biased language.

Semantic Features – The features that capture semantic aspects of texts can help determine patterns from data.

Subjectivity – The sentiment scores of the text can help us know the emotion invoked by the news article.

2.1.3 Deception linguistic features and profiling

A popular approach which gained prominence in the mid-20 0 0s was the use of linguistic cues for detecting deception in written narratives. Experiments performed by psychologists in cooperation with linguistics experts and computer scientists, revealed that the potential deceivers use certain language patterns, such as small sentences lots of phrasal verbs, certain tenses etc. The experiments concluded to specific linguistic cues classes that could reveal the deceiver. The features proposed is separated into four categories, namely Grammatical Complexity, Vocabulary Complexity, Quantity and Specificity/Expressiveness.

Similar experiments have also aimed at a multivariate linguistic profile of deception. More specifically, they proposed a set of five out of twenty-nine linguistic cues which is an

intersection of the most significant predictors of deception.

2.1.4 Text – based detection

The text from the news articles is raw and requires a lot of pre-processing according to our needs. One of the approaches is to create a Bigram Term Frequency- Inverse Document Matrix. Feature generation can be done using Python libraries such as NLTK and Spacy. NLTK is a primitive python package but is still very useful while Spacy is relatively faster as compared to NLTK.

We remove any mention of the name of the source from the news articles because the reliable/unreliable classification is determined at the source level, this step is necessary to ensure the model does not just learn the mappings from known sources to labels. The first feature set is vectorized bigram Term Frequency Inverse Document Frequency. This is a weighted measure of how often a particular bigram phrase occurs in a document relative to how often the bigram phrase occurs across all documents in a corpus.

2.1.5 Image-based detection using web scrapping

The text from the article is extracted using OCR. The entity extractor then extracts all the entities from the text and then it goes for text cleaning, which includes, striping all non-alphabetic characters, removing multiple occurrences of words, checking whether the word is a valid English word, checking each word for spelling errors, removal of any media house name or newspaper name to remove bias. Then the string of extracted entities is looked upon Google and then the relevant search results are gathered. The last and foremost step is to classify the news into genuine or fake.

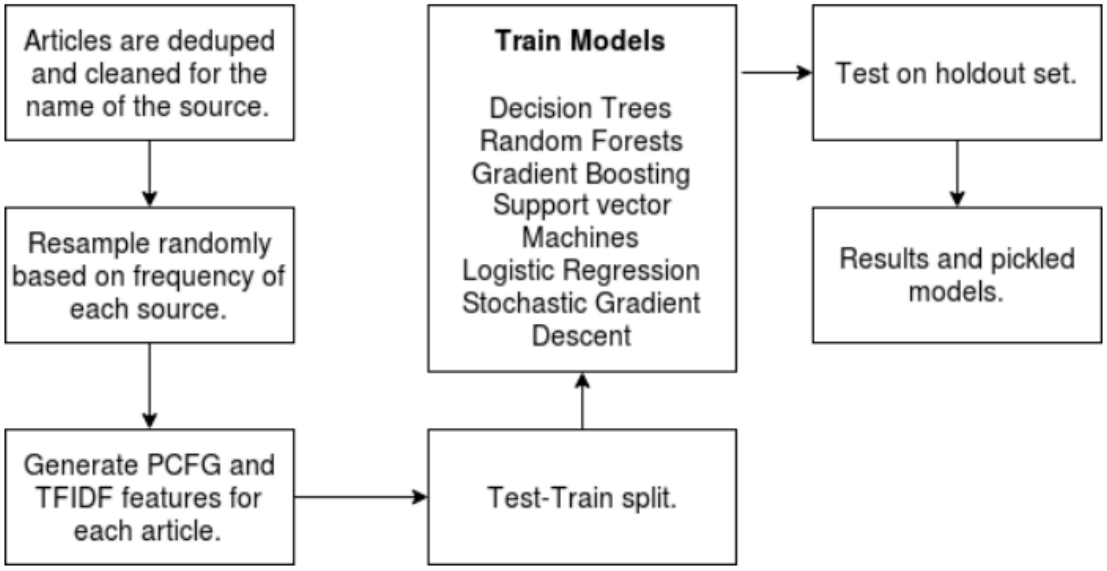

Figure 3. Architecture of fake news detection using text-based approach.

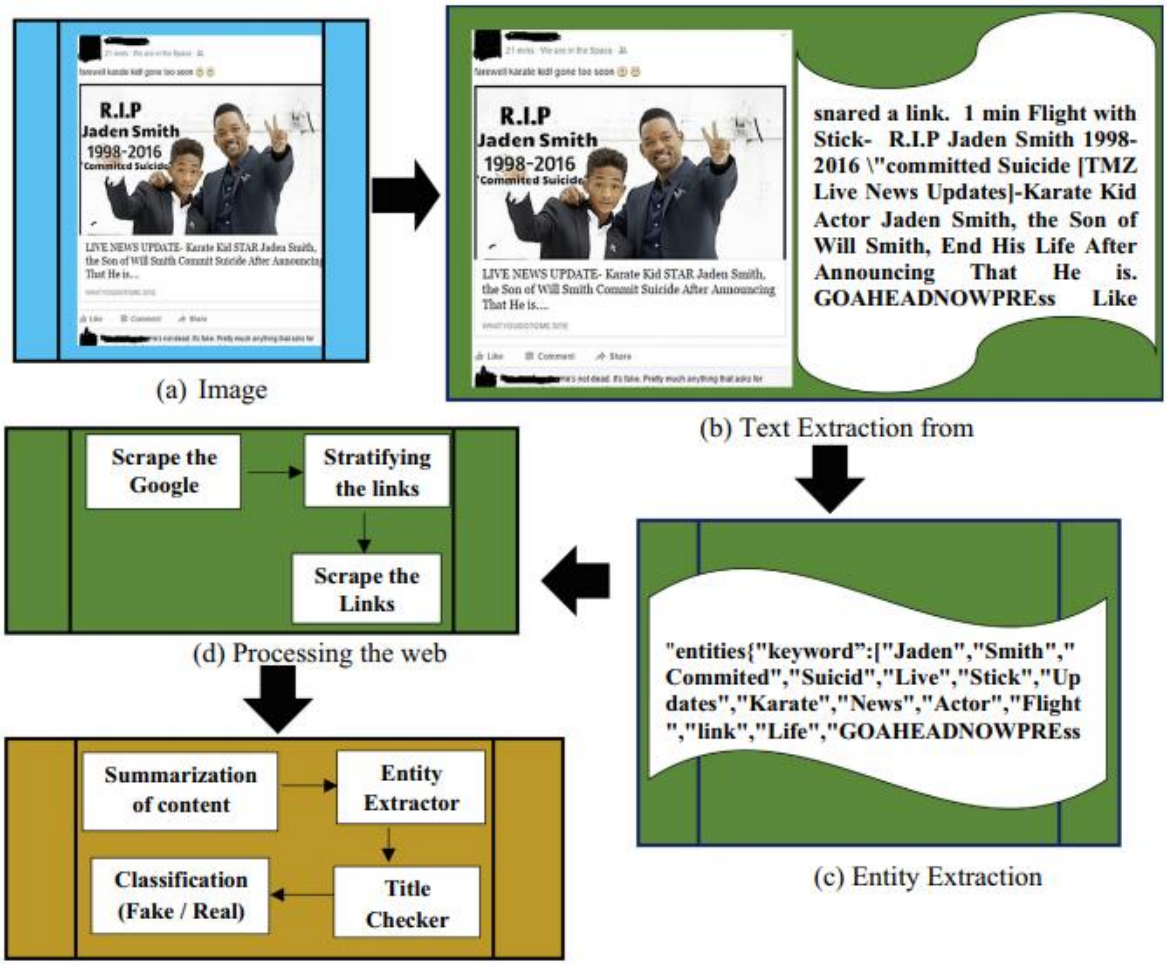

Figure 4. Architecture of fake news detection using image-based approach

2.1.6 Detecting fake sites

The first step is to locate a credible clickbait's database, and then compute the attributes. And the second step is to gather URLs in a file, a python script computed the attributes from the title and the content of the web pages. Then the attributes are ranked based on several algorithms to choose the most relevant to increase the accuracy and decrease the training time.

- Info Gain Attribute Eva evaluates the worth of an attribute by measuring the information gain with respect to the class. $InfoGain(Class, Attribute) = H(Class) - H(Class \mid Attribute)$.

CorrelationAttributeEval evaluates the worth of an attribute by measuring the correlation (Pearson's) between it and the class. Nominal attributes are considered on a value by value basis by treating each value as an indicator. An overall correlation for a nominal attribute is arrived at via a weighted average. So, an indicator for the value of a nominal attribute is a numeric binary attribute that take on the value of 1 when the value occurs in an instance and 0 otherwise.

2.1.7 Web scraping method

ALGORITHM FOR SCRAPING THE CONTENT
FROM XML USING BS4

1. Import the necessary libraries for scraping such as bs4 to parse the data returned from the website
2. Fetch the links of the url using urllib2 library and
save it in a variable called page
3. For each link do
    a. Parse the XML in the page variable and store it in a BeautifulSoup format.
    b. For each data in the item tag do
        i. Scrap the title, published date,
creator/author, link, description and image.
4. Save the scraped content into the database
5. Set the apscheduler to schedule jobs to run
periodically at fixed times, dates, or intervals.

2.1.8 Machine Learning approach

The general principle is to train an algorithm on a large sample of manually analyzed web pages. The machine learning is based on geographical indicators of the text blocks on the page: statistical measure is done where the main text block is located compared to the other text blocks. The

machine is then able to deduce by itself where the text is usually located. The larger the sampling, the more accurate the algorithm is.

2.2 Integrated summary of the literature studied

What is Fake News?

A type of yellow journalism, fake news encapsulates pieces of news that may be hoaxes and is generally spread through social media and other online media. This is often done to further or impose certain ideas and is often achieved with political agendas. Such news items may contain false and/or exaggerated claims, and may end up being viral by algorithms, and users may end up in a filter bubble.

Types of Fake News

Not all new misleading news articles are created with the same purpose. The following are the major categories

1. Fake news or hoax: Websites that post articles with the purpose of intentionally deceiving readers, possibly mimicking existing real news websites. e.g. RealNewsRightNow.com, ABCnews.com.co

2. Satire/parody: Websites intended to be funny with no direct intention of deceiving readers but may still lead to spread of misinformation. e.g. TheOnion.com

3. Bias/conspiracy theories: Websites consisting of a mix of biased opinions and made up conspiracy theories. We speculate that the content in these websites differ distinctively from genuine news sources in their style of writing and choice of words. We believe that a machine learning model should be able to accurately model this difference based solely on the text of the articles. For the purpose of this paper, we have classified news articles belonging to any of the 3 categories above as “Fake” since our primary goal is only to red-flag an article if it is not deemed genuine and truthful.

CHAPTER 3: REQUIREMENT ANALYSIS AND SOLUTION APPROACH

3.1 Overall description of project

This project is defined around the concepts of data science and machine learning. The goal here is to identify whether a "news" article is fake or real. There are many sources of fake article generation such as Facebook and Twitter, which are used as a trading platform to disseminate fake news. We have created a web crawler using Scrapy framework for collection of vast data from fake news websites such as www.abcnews.com and real news websites such as www.nytimes.com. Also, we have Kaggle (https://www.kaggle.com/clmentbisaillon/fake-and-real-news-dataset) dataset with similar attributes like title, text, date, subject and we have merged it with the dataset collected from web-crawler for a more unbiased and genuine dataset.

We will take this dataset of labelled news articles and apply classification techniques with frequency vectorizer. We can later test the model for accuracy and performance on unclassified public-messages. Similar techniques can be applied to other NLP applications like sentiment analysis etc.

3.2 Requirement analysis

To meet the expectations of the model and to complete the project, we would require the following hardware specifications and hardware.

Specific Requirements: -

1. Laptop/ Desktop PC with minimum 8 GB RAM
2. Intel-i5 Processor
3. Windows / Linux Operating Systems
4. Stable Internet Connection

Non-Functional Requirements

1. **Usability:** The method can be used to classify the news articles into the two categories, real and fake.
2. **Maintainability**: The code developed in the project is easily readable as it consists of well-defined functions and variables. This also facilitate in debugging the code at later stages. Also, any further improvement in the code is facilitated.
3. **Scalability:** The code will be able to handle a single news article at a time

3.3 Solution approach

Detection Of news articles whether they are “fake” or “real” has been done using feature extraction techniques, Term Frequency–Inverse Document Frequency (TF–IDF), Count-Vectorizer (CV) that are used to extract feature vectors from the textual content of articles.

This phase involves converting the text articles to a numerical fixed-length vector representation interpretable by machine learning models. The following techniques are frequently used for vector space models:

1. <u>Bag-of-words (BoW)</u> : The bag-of-words model is a simple and intuitive model, but works surprisingly well for methods involving calculation of a similarity measure between documents. It is commonly used in document clustering tasks. The idea behind the model is to generate a vocabulary based on the unique words collected from documents (or articles) in the corpus, and then projecting each article/document into a vectorized representation by counting the occurrence of each word of the vocabulary in the document/article. Either the entire vocabulary or a subset of most commonly occurring words can be used.

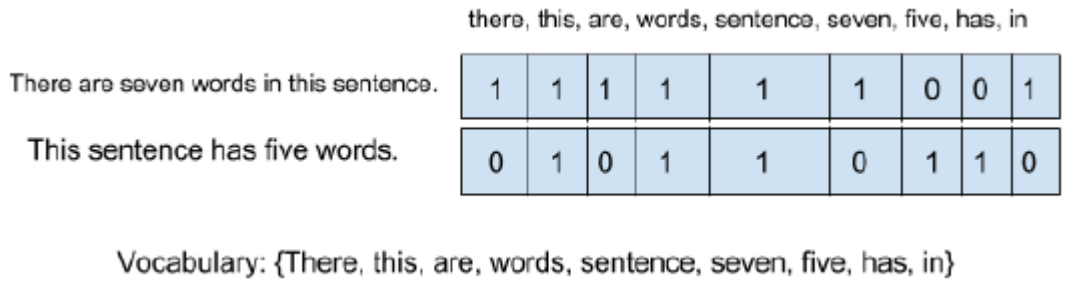

Fig 5. Example for bag-of-words model

2. <u>Term-frequency Inverse document frequency (tfidf)</u>: tfidf combines the frequency of occurrence of each term with the inverse of its document frequency (the number of documents in the corpus it shows up in). This dampens the high frequency scores of words that appear frequently in all documents such as pronouns and prepositions (“this”, “an”, “but”, etc) Given a corpus of documents D, the tfidf score for a term t appearing in document d is defined as:

$$tfidf(t, d, D) = tf(t, d).idf(t, D)$$

Where tf(t, d) is the number of times term t appears in document d, and idf(t, D) is the logarithmically scaled inverse of the fraction of documents containing t. A high weight in

tf–idf is reached by a high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents.

3. N-gram models: N-grams models solve the problem of incorporating the ordering of words into vector representations. It extends the BoW and tfidf models by using not only individual words to form the vector representation but also groups of N continuous words extracted from the corpus of documents. For instance, given the corpus of sentences we used to illustrate the Bag-of-words model

"There are seven words in this sentence"

"This sentence has five words"

A 2-gram model would extract the following 9 groups as part of the vocabulary other than the 9 unique words seen in the corpus:

(there are), (are seven), (seven words), (words in), (in this), (this sentence),

(this sentence), (has five), (five words)

This creates a vector representation of size 18 for each sentence (or document).

4. Latent semantic analysis: Latent semantic analysis, also known as latent semantic indexing, solves the problem of synonymy and polysemy in language vocabularies. In other words, the vocabulary represented by a vector space model usually contain words that are similar in meaning (synonymy) or contain words that have multiple interpretations (polysemy). Latent semantic analysis takes the term-document matrix (from tfidf or bag of words model) and transforms it into a lower rank matrix, preserving the rows (denoting the documents) while reducing the number of columns (the number of topics) by compressing words with similar meanings into a single topic.

CHAPTER 4: MODELING AND IMPLEMENTATION DETAILS

4.1 Design diagrams

Web Crawler

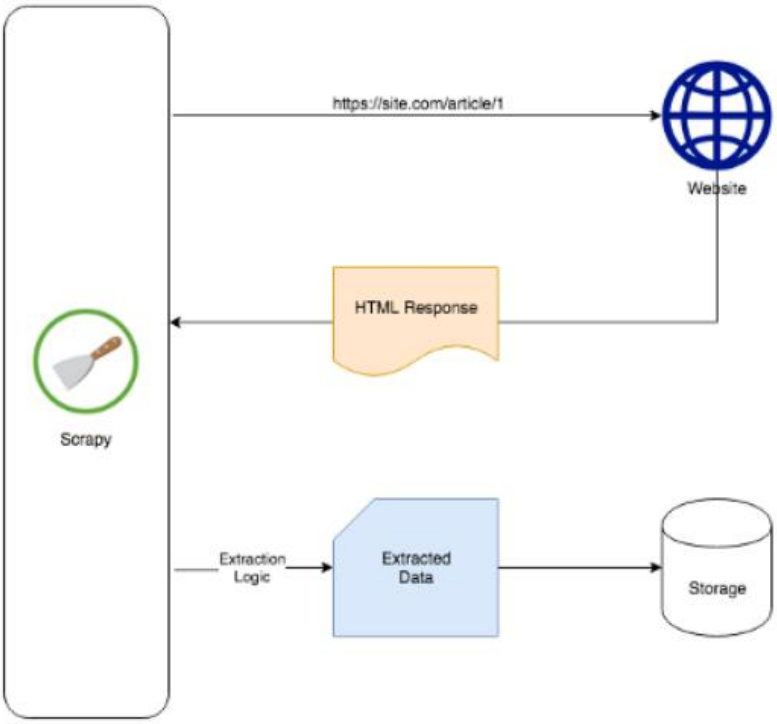

Fig 6. Overview of scraping news articles from website using Scrapy

Model

We use TfIdf Vectorizer to convert our text strings to numerical representations and initialize a Logistic Regression Classifier to fit the model.

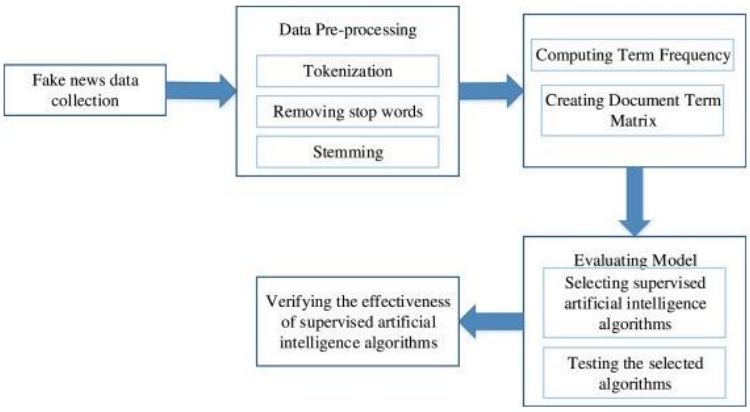

Fig 7. Architecture of Fake News Detection ML model

4.2 Implementation details and issues

Collection of Data

In order to assess the performance/reliability of a machine learning system in identifying fake news, hoaxes and satire from genuine news articles, a robust dataset of news articles is a prerequisite and building this dataset is no trivial task. We expect that the data to satisfy the following requirements:

1. Topic diversity**:** Having news articles from a plethora of categories such as politics, business, sports, entertainment, etc. increases the generalization capability of our trained model. We try to maintain a balanced topic distribution as far as possible but this is not easy due to the political bias of most fake news sites.

2. Source diversity**:** Different news websites and sources have different styles of writing, preference of word usage, etc. To factor in such differences in our model, articles need to be scraped from a variety of different sources.

3. Good class distribution: The class distribution should ideally be balanced, i.e. roughly equal number of instances for Fake and Real classes.

For articles categorized as fake news, hoaxes and satire, the data was gathered by scraping the following websites:

- www.theonion.com (Satire)
- www.politicops.com (Fake/Hoax/Bias)
- www.realnewsrightnow.com (Fake/Hoax/Bias)
- www.enduringvision.com (Fake/Hoax/Bias)
- www.civictribune.com (Fake/Hoax/Bias)
- www.newsbiscuit.com (Fake/Hoax/Bias)

A total of 3313 articles were gathered from these websites which have been assigned the class "Fake" regardless of the category being satire, fake, hoax or bias. Note that since most of these websites focus largely on political content, the topic of the articles collected might be biased towards the politics category.

In addition, an openly available fake news dataset on Kaggle was used, which had 12403 articles in English belonging to the time period October 2016 and November 2016, and categorized as being fake, unreliable, untrustworthy or junk. We added these articles to our dataset, coalescing them under the "Fake" class. This resulted in a total of **15716** fake news

articles. For authentic news, we used a collection of articles from "The New York Times" and "The Guardian" since these are the popular and reliable news sources, and also have active developer APIs for mining news articles. A publicly available dataset of news articles from "BBC news" was also appended to the list of real articles, adding up to a total of **12591** real news articles.

```
target
fake    39197
true    34008
Name: text, dtype: int64
```

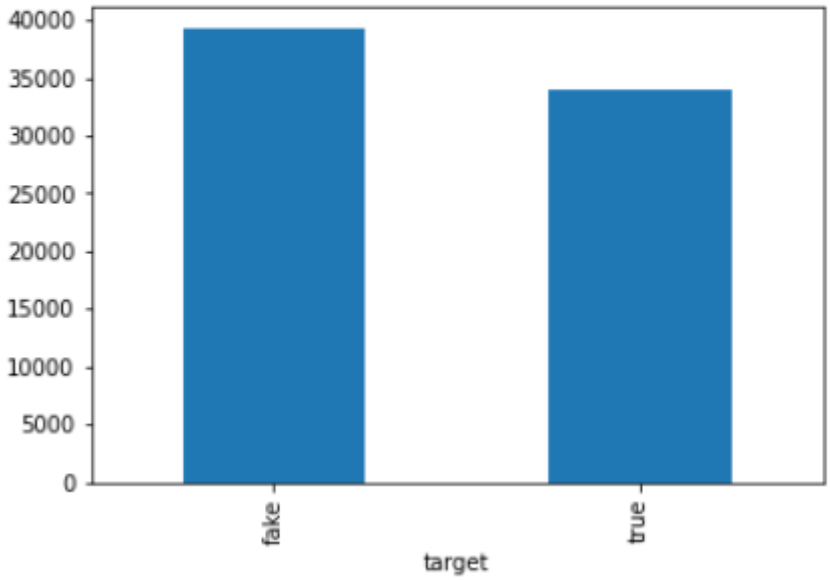

Fig 8. Analysis of Final Dataset

Term Frequency (Tf) — Inverse Document Frequency (Idf) Vectorizer

Tf-Idf Vectorizer is a common algorithm to transform text into meaningful representation of numbers. It is used to extract features from text strings based on occurrence.

We assume that higher number of repetitions of a word would mean greater importance in the given text. We normalize the occurrence of the word with the size of the document and hence call it term-frequency. Numerical definition:

tf(w) = doc.count(w) / total words in the doc

While computing term-frequency, each term is given equal weightage. There may be words which have high occurrence across the documents and hence would contribute less in deriving the meaning of document. Such words for example 'a', 'the' etc. might suppress the weights of more meaningful words. To reduce this effect, Tf is discounted by a factor called inverse document frequency.

idf(w) = log(total_number_of_documents / number_of-documents_containing_word_w)

Tf-Idf is then computed by taking a product of Tf and Idf. More important words would get a higher tf-idf score. tf-idf(w) = tf(w) * idf(w)

Developing the Model

- The architecture of the proposed fake news detection system is shown in Fig. 3 .To train the system, dataset have been collected from a web crawler built using scrapy framework.
- In the pre-processing phase, stop words and duplicate text from news articles are removed. The missing values, i.e., not available (NA) values, are collected and cleaned in the next step.
- The data retrieved are then split into two parts, training (0.80) and testing (0.20) sets.
- The feature extraction phase is then carried out to retrieve meaningful features from the textual data. The features are extracted from the articles such as Term Frequency-Inverse Document Frequency (assigns weights according to the importance of the terms in the document), Count-Vectorizer (counts the frequency of the terms in a document) have been applied.
- The features retrieved are then fed to the classification algorithm chosen in next step. The various ML models such as Logistic Regression, Decision Tree, and Random Forest Classifier are chosen to learn and identify the patterns and outcomes from them.
- The models are then evaluated based on performance metrics to achieve an efficient classifier.

CHAPTER 5: TESTING

5.1 Testing plan

In order to test a machine learning algorithms, we define two different datasets:

training dataset(80%) and testing dataset(20%)

Once the data set is defined, we will begin to train the models with the training dataset. Once this training model is done, then we performs to evaluate the models with the testing dataset. This is iterative and can embrace any tweaks/changes needed for a model based on results that can be done and re-evaluated.

5.2 Component decomposition and type of testing required

Cross Validation

Cross-validation is a technique where the datasets are split into multiple subsets and learning models are trained and evaluated on these subset data. One of the widely used technique is the ***k-fold cross-validation*** technique. In this, the dataset is divided into k-subsets(folds) and are used for training and validation purpose for ***k iteration*** times. Each subsample will be used at least once as a validation dataset and the remaining (*k-1*)as the training dataset. Once all the iterations are completed, one can calculate the average prediction rate for each model.

Evaluation Techniques:

1. Confusion Matrix

It's a square matrix table of N*N where N is the number of classes that the model needs to classify. It's best used for classification models that categorizes an outcome into a finite set of values. These values are known as labels. One axis is the label that the model predicted and the other is the actual label.

2. Classification Accuracy

It's the most basic way of evaluating the learning model. It's a ratio between the positive(TN+TP) predictions vs the total number of predictions. If the ratio is high then the model has a high prediction rate. Below are the formulas to find the accuracy ratio.

$$\text{Accuracy} = \frac{Total\ Positive\ Prediction}{Total\ Number\ of\ Prediction}$$

(OR)

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+FP+FN+TN)}$$

5.3 Limitations of the solution

1. A major flaw of the bag-of-words model is that it does not account for the ordering of words in a document, and as such any random permutation of words of a document would end up having the same representation. For example, the following sentences, although different semantically, would have the same representation under the BoW model:

The quick brown fox jumps over the lazy dog

The quick dog jumps over the lazy brown fox

2. Another issue with the BoW model is its inability to put adequate weights on important, representative words in the corpus vs frequently occurring words in the language like prepositions and pronouns. For example, using the BoW model, words like “the”, “a”, “an” which occur very frequently in English text would always end up with high counts.

3. One issue with the N-grams models is that the term-document matrix created is extremely sparse and the sparsity of the increases with N.

CHAPTER 6: FINDINGS, CONCLUSIONS AND FUTURE WORK

6.1 Findings

We list below the performance of the best model and the associated parameters of the model for each vectorization method used. Confusion metrics are over the validation set and averaged over 4 cross-validation iterations.

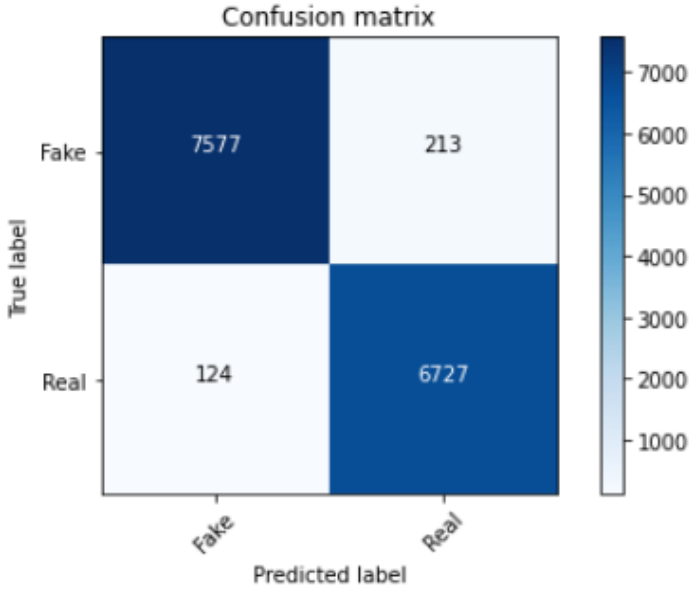

Fig.9. Confusion matrix for Logistic Regression Classifier
Accuracy 97.7 %

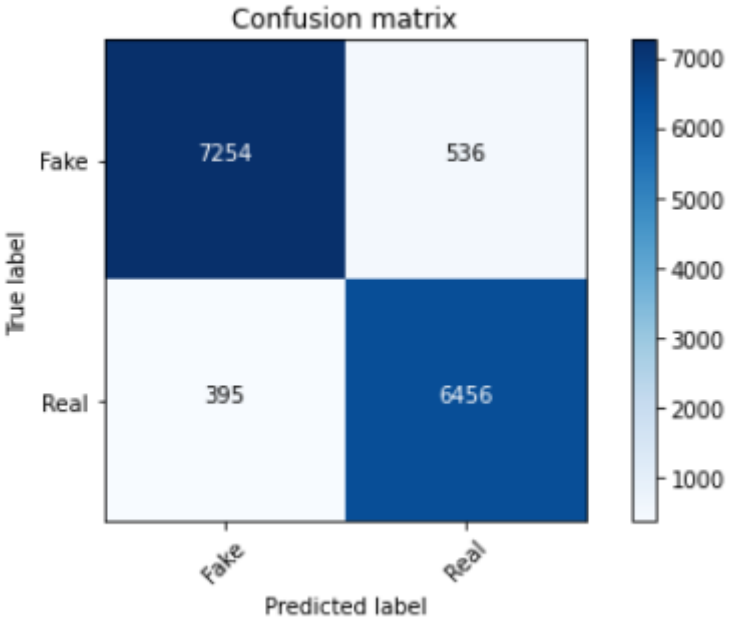

Fig 10. Confusion matrix for Decision Tree Classifier
Accuracy 93.64 %

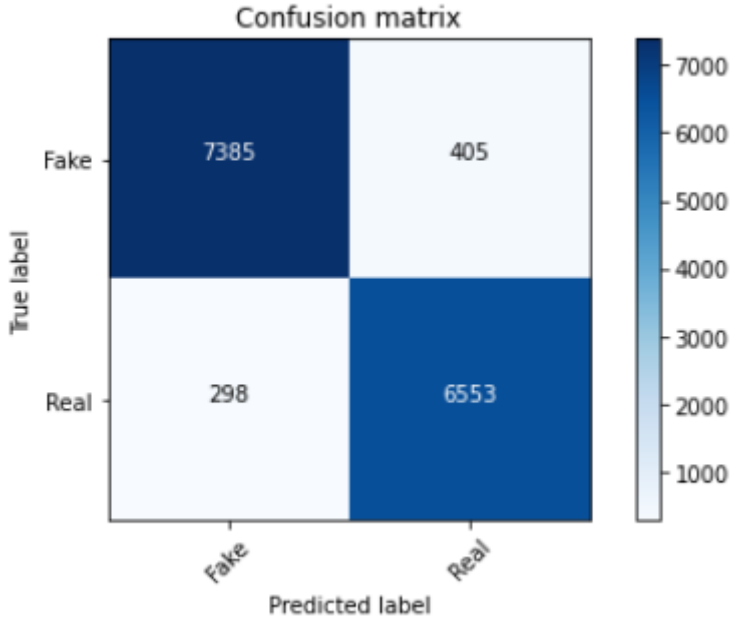

Fig 11. Confusion matrix for Random Forest Classifier
Accuracy 95.2 %

6.2 Conclusion

We successfully implemented a machine learning and natural language processing model to detect whether an article was fake or fact. We a total of 73205 articles scraped from our web crawler for which we trained our models. Our models provides accuracy as Logistic Regression (LR) **97.7 %**, Decision Tree(DT) **93.64%** and Random Forest (RF) **95.2%** .When doing such a classification, it is important to check that we limit the number of false positives as they can cause facts to be marked as fake. Text analytics and NLP can be used to work with the very important problem of fake news. We have seen the big impact they can have on people's opinions, and the way the world thinks or sees a topic.

6.3 Future Work

1. We intend to expend this project by adding a graphical user interface (GUI) where one can paste any piece of text and get its classification in the results. This can be done by evaluating our ml model as a Google extension.
2. Currently, we are only evaluating text of the news articles. In future, we can evaluate titles of the news also.
3. Fake link such as the hyperlinks that redirect you to a different web page can also be detected.

REFERENCES

[1] Aldwairi, M., & Alwahedi, A. 2018. Detecting Fake News in Social Media Networks. Procedia Computer Science, 141: 215–222. https://doi.org/10.1016/j.procs.2018.10.171

[2] A. Choudhary and A. Arora, "Linguistic Feature Based Learning Model for Fake News Detection and Classification", Expert Systems with Applications, p. 114171, 2020.

[3] R. DIOUF, E. SARR, O. SALL, B. BIRREGAH, M. BOUSSO and S. MBAYE, "Web Scraping: State-of-the-Art and Areas of Application", 2019, pp. 6040-6042.

[4] N. El Abbadi, A. Mohamad Hassan and M. Mohammed AL-Nwany, "Blind Fake Image detection", Ijcsi.org, 2013. [Online]. Available: https://ijcsi.org/papers/IJCSI-10-4-1-180-186.pdf.

[5] P. Faustini and T. Covões, "Fake News Detection Using One-Class Classification," 2019 8th Brazilian Conference on Intelligent Systems (BRACIS), Salvador, Brazil, 2019, pp. 592-597, doi: 10.1109/BRACIS.2019.00109.

[6] P. Faustini and T. Covões, "Fake news detection in multiple platforms and languages", Expert Systems with Applications, vol. 158, p. 113503, 2020.

[7] S. Gilda, "Notice of Violation of IEEE Publication Principles: Evaluating machine learning algorithms for fake news detection," 2017 IEEE 15th Student Conference on Research and Development (SCOReD), Putrajaya, 2017, pp. 110-115, doi: 10.1109/SCORED.2017.8305411.

[8] S. Girgis, E. Amer and M. Gadallah, "Deep Learning Algorithms for Detecting Fake News in Online Text", 2019, pp. 93-97.

[9] M. Granik and V. Mesyura, "Fake news detection using naive Bayes classifier," 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), Kiev, 2017, pp. 900-903, doi: 10.1109/UKRCON.2017.8100379.

[10] G. Gravanis, A. Vakali, K. Diamantaras and P. Karadais, "Behind the cues: A benchmarking study for fake news detection", Expert Systems with Applications, vol. 128, pp. 201-213, 2019.

[11] S. Hakak, M. Alazab, S. Khan, T. Gadekallu, P. Maddikunta and W. Khan, "An ensemble machine learning approach through effective feature extraction to classify fake news", Future Generation Computer Systems, vol. 117, pp. 47-58, 2020.

[12] R. Kaliyar, A. Goswami, P. Narang and S. Sinha, "FNDNet – A deep convolutional neural network for fake news detection", Cognitive Systems Research, vol. 61, pp. 32-44,

2020.

[13] S. Kaur, P. Kumar and P. Kumaraguru, "Automating fake news detection system using multi-level voting model", Soft Computing, vol. 24, no. 12, pp. 9049-9069, 2019.

[14] H. Ko, J. Hong, S. Kim, L. Mesicek and I. Na, "Human-machine interaction: A case study on fake news detection using a backtracking based on a cognitive system", Cognitive Systems Research, vol. 55, pp. 77-81, 2019.

[15] P. Lara-Navarra, H. Falciani, E. Sánchez-Pérez and A. Ferrer-Sapena, "Information Management in Healthcare and Environment: Towards an Automatic System for Fake News Detection", International Journal of Environmental Research and Public Health, vol. 17, no. 3, p. 1066, 2020.

[16] Y. Liu and S. Xu, "Detecting Rumors Through Modeling Information Propagation Networks in a Social Media Environment," in IEEE Transactions on Computational Social Systems, vol. 3, no. 2, pp. 46-62, June 2016, doi: 10.1109/TCSS.2016.2612980.

[17] A. Pathak, A. Mahajan, K. Singh, A. Patil and A. Nair, "Analysis of Techniques for Rumor Detection in Social Media", Procedia Computer Science, vol. 167, pp. 2286-2296, 2020. Available: 10.1016/j.procs.2020.03.281.

[18] D. Radovanovic and B. Krstajic, "Review spam detection using machine learning", 2018 23rd International Scientific-Professional Conference on Information Technology (IT), 2018. Available: 10.1109/spit.2018.8350457.

[19] J. Reis, A. Correia, F. Murai, A. Veloso, F. Benevenuto and E. Cambria, "Supervised Learning for Fake News Detection", IEEE Intelligent Systems, vol. 34, no. 2, pp. 76-81, 2019.

[20] Victoria L. Rubin, Yimin Chen, and Niall J. Conroy. Deception detection for news: three types of fakes. Proceedings of the Association for Information Science and Technology, 52(1):1--4, 2015.

[21] K. Shu, D. Mahudeswaran and H. Liu, "FakeNewsTracker: a tool for fake news collection, detection, and visualization", Computational and Mathematical Organization Theory, vol. 25, no. 1, pp. 60-71, 2018.

[22] R. Silva, R. Santos, T. Almeida and T. Pardo, "Towards automatically filtering fake news in Portuguese", Expert Systems with Applications, vol. 146, p. 113199, 2020.

[23] K. Sundaramoorthy, R. Durga and S. Nagadarshini, "NEWSONE- AN AGGREGATION SYSTEM FOR NEWS USING WEB SCRAPING METHOD", in International Conference on Technical Advancements in Computers and Communications, 2017, pp. 137-140.

[24] B. Suri, S. Taneja, S. Aggarwal and V. Sharma, "Fake news detection tool (FNDT): Shield against sentimental deception", Journal of Information and Optimization Sciences, pp. 1-12, 2020.

[25] M. Umer, Z. Imtiaz, S. Ullah, A. Mehmood, G. Choi and B. On, "Fake News Stance Detection Using Deep Learning Architecture (CNN-LSTM)", IEEE Access, vol. 8, pp. 156695-156706, 2020.

[26] D. Vishwakarma, D. Varshney and A. Yadav, "Detection and veracity analysis of fake news via scrapping and authenticating the web search", Cognitive Systems Research, vol. 58, pp. 217-229, 2019.

[27] G. Wang, S. Xie, B. Liu and P. Yu, "Identify Online Store Review Spammers via Social Review Graph", ACM Transactions on Intelligent Systems and Technology, vol. 3, no. 4, pp. 1-21, 2012. Available: 10.1145/2337542.2337546.

[28] Zhang, C., Gupta, A., Kauten, C., Deokar, A. and Qin, X., 2019. Detecting fake news for reducing misinformation risks using analytics approaches. European Journal of Operational Research, 279(3), pp.1036-1052.

[29] X. Zhang and A. Ghorbani, "An overview of online fake news: Characterization, detection, and discussion", Information Processing & Management, vol. 57, no. 2, p. 102025, 2020.

[30] Zhou, Z., Guan, H., Moorthy Bhat, M., & Hsu, J. (2019). Fake News Detection via NLP is Vulnerable to Adversarial Attacks. Proceedings of the 11th International Conference on Agents and Artificial Intelligence, doi:10.5220/0007566307940800