JAIN DEEMED-TO-BE UNIVERSITY | FACULTY OF ENGINEERING AND TECHNOLOGY

"*FAKE NEWS DETECTION USING MACHINE LEARNING.*"

BACHELOR OF TECHNOLOGY (HONOURS)

IN

COMPUTER SCIENCE (DATA SCIENCE)

Submitted by

Prabal Dhar - 21BTRCD056.

Yogendra Naidu – 21BTRCD028.

Amen H. Asfaw – 21BTRCD058

Sajjad Ali Dhuniya – 21BTRCD064

Chandan Kumar Roy - 21BTRCD057

Under the guidance of

Prof. Narasimiah

Faculty of Engineering & Technology

Jain (Deemed-To-Be University)

FAKE NEWS DETECTION USING MACHINE LEARNING.

Introduction

The technological era brought with its innovations like radio, the internet, and television; information outlets like written newspapers were left behind, and the world opened up to new ways to learn about current events. While most news content on television and radio is reviewed and controlled, that of the internet is hardly supervised. As a result, false news has gained ground in today's society. Despite the fact that it is impossible to provide a formal definition of false news, they are manufactured in such a manner that they successfully mimic some of the most renowned newspapers and news networks.

Related Work

Many earlier studies have been conducted from various viewpoints, targeted in various ways, and employed various approaches, but ultimately all strive to counteract misinformation. Some of these Studies will be provided below because the subject addressed is so relevant in our information age.

In Gerardo Ernesto Rolong Agudelo et al. (2018) they make a study of Raising a Model for Fake News Detection Using Machine Learning in Python. For the suggested method of identifying false news in open data sets, machine learning methods such as "CountVectorizer," "TfidfVectorizer," a Naive Bayes Model, and natural language processing are used. The study's final limitation was that, while utilising a conventional method for text classification, categorising news is a difficult endeavour because it has so many different qualities that might be assessed.

On the other hand, Akshay Jain et al. (2018) they work on social media fake news detection using Machine Learning. Their proposed method of picked Facebook as a starting point to curb this problem. Their method faced limitations due to social media fake news detection is unable to identify stories that receive few shares or likes. Social media users must spread the news in order to be detected. While it cannot identify all false news, it can identify a significant portion of social media news.

In Z Khanam et al. (2021) they inquire into fake news detection using machine learning approaches. The paper proposed detecting fake news from different social media websites by Neural Network, Naïve Bayes, and Support Vector Machine

(SVM) which resulted in 96.08% for detecting fake messages. The failing of the paper was that the results were not accurate for the untruthful sources. Detection of Twitter spam senders, with accuracy rated from 70% to 71.2%.

Feyza Altunbey and OzbayBilalAlatas (2019) worked on fake news detection within online social media using supervised artificial intelligence algorithms. A two-step method for identifying fake news in social media was proposed. An experimental evaluation of the intelligent classification methods was performed. Three actual data sets are used to test methods using various evaluation measures.

M. Granik and V. Mesyura (2017) they present on the study of a smart system for fake news detection using machine learning. The suggested approach combines semantic analysis, support vector machines, and Naive Bayes classifier. Instead of using computations that can't accurately reflect subjective capacities, the suggested solution is entirely constructed out of artificial intelligence approaches, which are fundamental to distinguish between the real and the fake. According to the study's findings, the prototype's effectiveness and accuracy can be improved to some extent, and the user interface of the suggested model can also be improved.

Objective

To successfully stop the spreading of fake news, which has increased in volume and has led to more and more misinformation, leading to various forms of property harm and fatalities. Fake news detection is a classic text classification problem with a straightforward proposition. Building a model that can distinguish between "Real" and "Fake" news is necessary. In order to create a precise and useful prototype that also has a nice user interface, this study investigates numerous textual features that can be utilized to distinguish between authentic and fraudulent information.

Problem Statement

The use of the internet for news consumption has pros and downsides. On the one hand, individuals seek out and consume news online due to its low cost, ease of access, and quick transmission of information. On the other side, it facilitates the widespread dissemination of "fake news," or poor news containing purposefully incorrect content. The widespread dissemination of false news has the potential to have very detrimental effects on people and society.

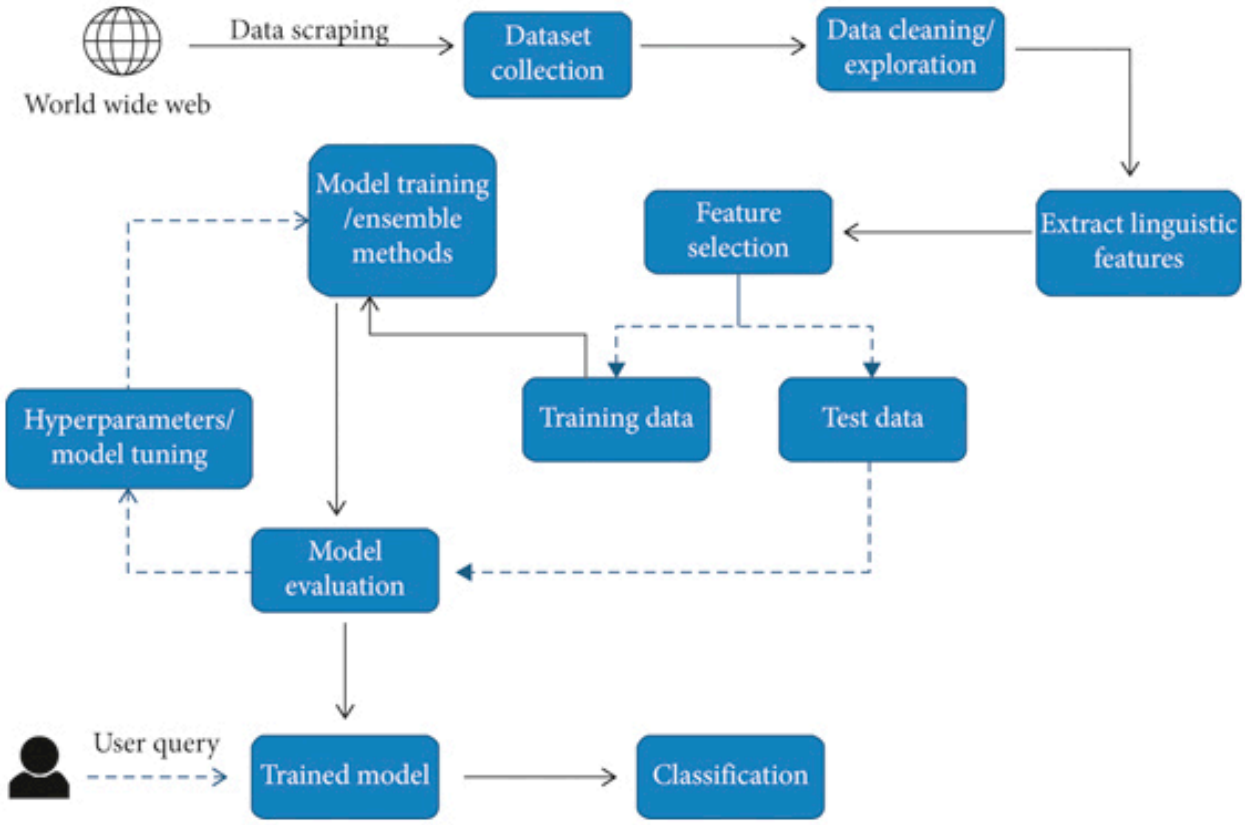
Our main problem of how fake news be effectively categorized as real or false utilizing the existing data and supervised machine learning algorithms can be done by integrating multiple processes together. This might include web scraping for data collection, user-generated data, pre-existing classified data, Database systems for data

storing, different machine learning algorithm implementations for classification, and finally effective distribution of the acquired insight to the user.

We propose developing an interface that smoothly integrates into users' news sources as a solution to the next issue of proper distribution of the acquired insights. Browser add-ons or other workable solutions may be used for this.

Proposed Method

- **ARCHITECTURE**



The architecture of the suggested approach consists of several parts that interact to produce the desired outcome. The architecture seen above has three primary parts. These include gathering data, processing it, and conveyance of insights or visualizing them.

Data collection is a crucial step in this process. We use pre-classified false news data sets from open-source projects and web-scraping with manual labelling to train a

prototype machine learning model. We may leverage user-generated content after prototyping to further train our model.

A model is trained using the machine learning algorithms and techniques described below during the data processing phase, which also involves feature selection from the obtained data beforehand.

How we share the acquired insight with the end user will determine how well we can stop the spread of fake news. Users need a seamless classification of their news as fake or real. We suggest a user interface that is completely integrated with users' social media and news sources. This interface can effortlessly let the user know in real time if the news is reliable.

- **ALGORITHM & TECHNIQUE**

To assess the effectiveness of false news detection classifiers, we combined our suggested methodology with the learning algorithms listed below. The learning algorithms are trained with a variety of hyperparameters in order to maximise accuracy for a specific dataset while preserving the appropriate balance between variance and bias. Each model is trained several times using a range of different parameters in order to optimise it for the best outcomes. It costs a lot of processing power to discover the optimal parameters using a grid search. To prevent the models from either overfitting or underfitting the data, however, precautions are taken.

We utilize this techniques and algorithms to effectively classify and detect fake news:

1. Logistic Regression
2. Support Vector Machine
3. Multilayer Perceptron
4. *K*-Nearest Neighbours (KNN)

Implementation

The desired outcome is a system for categorizing fake news that, through seamless integration with users' news sources, successfully stops fake news from spreading over time. This is accomplished by using a model with a high degree of confidence and a user interface that fits the users' routines for consuming news.

Importing the necessary libraries and packages into our workspace is the first step in achieving this goal so that we may train our model using the methods described above.

In [ ]:

```
import pandas as pd
import numpy as np
import re
import nltk
import sklearn
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
import matplotlib.pyplot as plt
import seaborn
```

After importing our pre-classified data set, we clean it using various libraries, extract features using tokenizers, and then divide the data set into a train set and a test set before executing algorithms to develop our model.

Feature Extraction using Count Vectorizer

In [ ]:

```
cv = CountVectorizer(max_features = 2000)
x = cv.fit_transform(preprocessed_train).toarray() #preprocessed tweets in the form of a sparse matrix
```

In [ ]:

```
#Define the dependent variable (prediction column)
y = dataset.iloc[:,2:3].values
```

Split the dataset into Train and Test sets

In [ ]:

```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.2, random_state = 0)
```

In [2]:

```
print(x_train.shape)
print(y_train.shape)
print(x_test.shape)
print(y_test.shape)
```

Then, in order to compare and contrast an effective way to categorise fake news, we apply various machine learning (ML) algorithms. To achieve this, we train our model using the train set, and then test it against the test set to compare it to various algorithmic approaches.

Logistic Regression

In [ ]:

```
#Using Logistic Regression
from sklearn.linear_model import LogisticRegression
lr=LogisticRegression()
lr.fit(x_train,y_train)
lr_ypred=lr.predict(x_test)
lr_acc = sklearn.metrics.accuracy_score(lr_ypred,y_test)
print("Logistic Regression Accuracy: ")
print(lr_acc*100,"%")
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/utils/validation.py:760: DataConversionWarning: A column-vector y was
passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

```
Logistic Regression Accuracy:
92.7570093457944 %
```

Naive Bayes Classifier ¶

In [ ]:

```
from sklearn.naive_bayes import GaussianNB
nb=GaussianNB()
nb.fit(x_train,y_train)
nb_ypred=nb.predict(x_test)
nb_acc = sklearn.metrics.accuracy_score(nb_ypred,y_test)
print("Naive Bayes Accuracy: ")
print(nb_acc*100,"%")
```

```
Naive Bayes Accuracy:
87.14953271028037 %
```

Support Vector Machine

In [ ]:
```
from sklearn.svm import SVC
svm=SVC(kernel="linear",C=0.025,random_state=None)
svm.fit(x_train,y_train)
svm_ypred=svm.predict(x_test)
svm_acc = sklearn.metrics.accuracy_score(svm_ypred,y_test)
print("Support Vector Machine Accuracy: ")
print(svm_acc*100,"%")
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/utils/validation.py:760: DataConversionWarning: A column-vector y was
passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

```
Support Vector Machine Accuracy:
92.13395638629284 %
```

K-Nearest Neighbour

In [ ]:
```
from sklearn.neighbors import KNeighborsClassifier
knn=KNeighborsClassifier(n_neighbors=3)
knn.fit(x_train,y_train)
knn_ypred=knn.predict(x_test)
knn_acc = sklearn.metrics.accuracy_score(knn_ypred,y_test)
print("K-Nearest Neighbour Accuracy: ")
print(knn_acc*100,"%")
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:3: DataConversionWarning: A column-vector y was passed w
hen a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  This is separate from the ipykernel package so we can avoid doing imports until
```

```
K-Nearest Neighbour Accuracy:
77.10280373831776 %
```

Finally, we may choose an accurate model that is effective and on which we can base our fake news classifier in order to create a user-friendly interface that will effectively address the fake news issue we are attempting to address.

Reference

[1]. Gerardo Ernesto Rolong Agudelo et al. (2018): Raising a Model for Fake News Detection Using Machine Learning in Python. https://link.springer.com/chapter/10.1007/978-3-030-02131-3_52#Sec3

[2]. Akshay Jain, Amey Kasbe (2018) : social media fake news detection using Machine Learning. https://ieeexplore.ieee.org/abstract/document/8546944

[3]. Z Khanam (2021) : Fake News Detection Using Machine Learning Approaches https://iopscience.iop.org/article/10.1088/1757-899X/1099/1/012040

[4]. Feyza Altunbey , OzbayBilalAlatas (2019): Fake news detection within online social media using supervised artificial intelligence algorithms https://www.sciencedirect.com/science/article/abs/pii/S0378437119317546

[5]. M. Granik and V. Mesyura (2017): A Smart System For Fake News Detection Using Machine Learning https://www.researchgate.net/publication/339022255_A_smart_System_for_Fake_News_Detection_Using_Machine_Learning