A Mini Project Report on

Fake News Detection

Submitted in partial fulfilment of the requirement for
Degree in T.E. (VI-Semester) in Computer Engineering

By

Sibin George (101819)
Radley Roy (101849)
Harshul Raina (101851)
Vishadh Sawant (101855)

Guided By

Mrs. Dakshayani R.

Department of Computer Engineering

Fr. Conceicao Rodrigues Institute of Technology,

Sector 9A, Vashi, Navi Mumbai - 400703

University of Mumbai

2021

CERTIFICATE

This is to certify that the Mini project entitled

Fake News Detection

Submitted by

Sibin George
Radley Roy
Harshul Raina
Vishadh Sawant

In partial fulfilment of degree of T.E. (VI-Semester) in Computer Engineering for term work of the Mini Project is approved.

External Examiner	**Internal Examiner**
External Guide	**Internal Guide**
Head of Department	**Principal**
Date: -	**College Seal**

ACKNOWLEDGEMENT

Success of a Mini project like this involving high technical expertise, patience and massive support of guides, is possible when team members work together. We take this opportunity to express our gratitude to those who have been instrumental in the successful completion of this project. We would like to show our appreciation to **Mrs. Dakshayani R.** for their tremendous support and help, without them this project would have reached nowhere. We would also like to thank our project coordinator **Ms. Shweta Tripathi** for providing us with regular inputs about documentation and project timeline. A big thanks to our HOD **Dr. Lata Ragha** for all the encouragement given to our team. We would also like to thank our principal, **Dr. S. M. Khot**, and our college, Fr. C. Rodrigues Institute of Technology, Vashi, for giving us the opportunity and the environment to learn and grow.

Abstract

Fake news and hoaxes have been there since before the advent of the Internet. The widely accepted definition of Internet fake news is: fictitious articles deliberately fabricated to deceive readers”.

The advent of the World Wide Web and the rapid adoption of social media platforms (such as Facebook and Twitter) paved the way for information dissemination that has never been witnessed in the human history before. With the current usage of social media platforms, consumers are creating and sharing more information than ever before, some of which are misleading with no relevance to reality

Social media and news outlets publish fake news to increase readership or as part of psychological warfare. In general, the goal is profiting through clickbaits. Clickbaits lure users and entice curiosity with flashy headlines or designs to click links to increase advertisements revenues. This exposition analyses the prevalence of fake news in light of the advances in communication made possible by the emergence of social networking sites.

The purpose of our fake news detection is to come up with a solution that can be utilized by users to detect and filter out sites containing false and misleading information. We use simple and carefully selected features of the title and post to accurately identify fake posts.

Contents

Introduction

As we spend more and more time interacting on social media platforms, more and more people tend to seek out and consume news from social media rather than traditional news organizations. However, because it is inexpensive to provide news online and much faster and easier to disseminate through social media, large volumes of fake news, i.e., those news articles with intentionally false information, are produced online for a variety of purposes, such as financial and political gain.

The widespread spread of false news can have a significant negative impact on individuals and society. First, fake news can break the authenticity balance of the news ecosystem. For example, it is evident that the most popular fake news was even more widely spread on Facebook than the most popular authentic mainstream news during the U.S. 2016 president election. Second, fake news intentionally persuades consumers to accept biased or false beliefs [1]. Fake news is usually manipulated by propagandists to convey political messages or influence. For example, some report shows that Russia has created fake accounts and social bots to spread false stories [2]. Third, fake news changes the way people interpret and respond to real news. For example, some fake news was just created to trigger people's distrust and make them confused, impeding their abilities to differentiate what is true from what is not [3] .

The Fake News Detection System is designed to help to keep track of many news articles and also provides the means to segregate fake and genuine news and completely removing any fake news. Users of the system can access genuine news without getting misled by the fake news.

To help mitigate the negative impact caused by fake news–both to benefit the public and the news ecosystem, we have created a project on fake news detection system. This project intends to prevent any spread of fake news by detecting and removing any fake news encounter.

Literature Survey

Fake news is not a new concept. It existed since prehistoric times. News was spread in the form of gossips, rumors, etc. In this day and age with the evolution of technology yellow journalism has gained more popularity through social media especially. The nature of social media makes it easy to spread fake news, as a user potentially sends fake news articles to friends, who then send it again to their friends and so on. Comments on fake news sometimes fuel its 'credibility' which can lead to rapid sharing resulting in further fake news [4]. Engaging the attention of the reader through clickbait is another form of fake news. Clickbait is an advertising tool used to get the attention of users. Fortunately, tools have been developed for detecting fake news. For example, a tool has been developed to identify fake news that spreads through social media through examining lexical choices that appear in headlines and other intense language structures [5]. Another tool, developed to identify fake news on Twitter, has a component called the Twitter Crawler which collects and stores tweets in a database. When a Twitter user wants to check the accuracy of the news found they can copy a link into this application after which the link will be processed for fake news detection. This process is built on an algorithm called the NER (Named Entity Recognition) [6].

Supervised Learning for Fake News. Detection.:

Reis et al. [7] has used feature engineering to generate hand-crafted features like syntactic features, semantic features etc. This problem discussed how supervised learning models can be used to assist fact-checkers in evaluating digital content and reaching warranted conclusions. The problem was then approached as a binary classification problem where these features were fed into conventional Machine Learning classifiers like K-Nearest Neighbour (KNN), Random Forest (RF), Naïve Bayes, Support Vector Machine (SVM) and XGBOOST (XGB), where RF and XGB yielded results that were quite favourable. The results show that the prediction performance of proposed features combined with existing classifiers has a useful degree of discriminative power for detecting fake news.

CSI: A Hybrid Deep Model for Fake News Detection.:

Ruchansky et al. [8] has proposed a novel CSI(Capture-Score-Integrate) framework that uses a Long Short-term Memory (LSTM) network to capture the temporal spacing of user activity and a doc2vec representation of a tweet, along with a neural network-based user scoring module to classify the tweet as real or fake. CSI is composed of three modules: Capture, Score, and Integrate. The first module is based on the response and text; it uses a Recurrent Neural Network to capture the temporal pattern of user activity on a given article. The second module learns the source characteristic based on the behaviour of users, and the two are integrated with the third module to classify an article as fake or not. Experimental analysis on real-world data demonstrated that CSI achieved higher accuracy than existing models, and extracts meaningful latent representations of both users and articles. It emphasizes the value of incorporating all three powerful characteristics in the detection of fake news: the tweet content, user source, and article response.

Multi-Source Multi-Class Fake News Detection.

Karimi et al. [9] has proposed a Multi-Source Multi-class Fake News Detection framework that can do automatic feature extraction using Convolution Neural Network (CNN) based models and combine these features coming from multiple sources using an attention mechanism, which has produced much better results than previous approaches that involved hand-crafted features. This combines information from multiple sources to discriminate between different degrees of fakeness, and proposes a Multi-source Multi-class Fake news Detection framework MMFD, which combines automated feature extraction, multi-source fusion and automated degrees of fakeness detection into a coherent and interpretable model.

There are many available approaches to help the public to identify fake news and this project aims to enhance understanding of these by developing a fake news detection system which detects fake news.

Existing Systems

1. **Oigetit Fake News Filter:** [10]
    - It is an award-winning news app that uses its proprietary AI-powered fake news filtering technology to deliver trusted news.
    - You can access your news from approximately 1,00,000 news sources.
    - Each and every day, OIGETIT follows approximately 1M news sources from all over the world in order to provide the most comprehensive, verified and trusted news on the web.
    - Using very sophisticated algorithms, every article is quickly processed through our artificial intelligence system based on facts, and then is assigned a trust rating so that you know how trustworthy the article is right now.
    - Oigetit's AI-powered fake news filtering technology compares the facts within an article against a large fact database that is derived from historical articles; then, it statistically calculates the levels of bias.
    - Oigetit then assigns a reliability % score and coloured shield based on these factors.

2. **OwlFactor:** [11]
    - Here, you can set your political leaning, find under-reported topics or those with the most coverage, and choose a grade for the article.
    - These grades are based on several factors like quality of sources, opinionated tone, author and outlet credibility, etc.
    - Machine Learning engine that has now rated 2 million articles on dimensions like author expertise, diversity of references, factual writing style, and more.
    - OwlFactor scans 10,000 stories from 200+ sources daily, automatically identifying the most credible stories on the hottest topics using the OwlFactor Grade.
    - Each article's OwlFactor Grade is calculated based on diversity of references, factual tone of writing, author expertise on topic, and historical site scores.
    - OwlFactor's algorithm has evaluated 2 million articles and has ratings for over 2,000 news sources and 30,000 journalists.

3. **WhatsFarzi:** [12]
    - The application helps to identify the fake news and helps to authenticate the content by checking it on the internet.
    - A lot of news gets circulated on social media like Facebook, Twitter, and WhatsApp. But very few are actually validated both by the senders and receivers.
    - The application uses a particular logarithm that helps to verify the content for its authenticity.

- The WhatsFarzi application not only verifies news but also images/photographs with the help of image tampering algorithm.
- They extract information from various locations, products, and people that are available on the internet and they use a knowledge graph for this.
- They update the graph with original news and save it on a secure database. This process helps them to verify the real news.

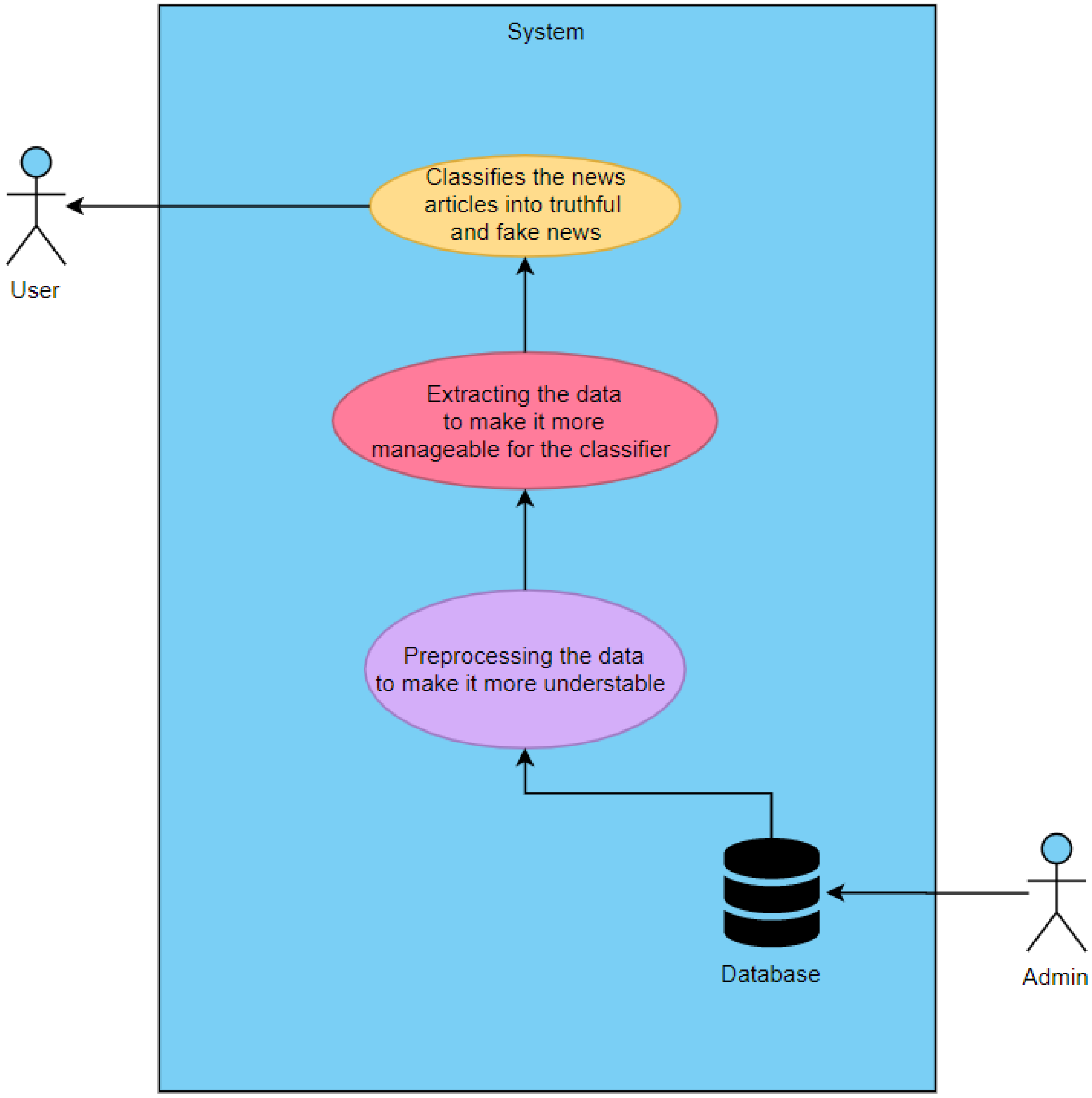
4. **NewsCop:** [13]

- If a user uploads an image for verification, then NewsCop will use SIFT Algorithm for fast image Comparison between already verified images from our Server and the user uploaded image. If it still doesn't find a match then will use Cloud Functions (have deployed ML for Image Comparison.
- Similar to Image Comparison if someone posts a news to get verified then NewsCop will use JaroWinkler distance algorithm to search if we have an already verified report of that particular news.
- App even provides a daily list of many viral but fake news.
- Cloud-based Artificial Intelligence uses a web of the algorithms to find the origin of the fake news, spoilers etc. It uses Machine Learning to make itself better every day.
- If someone posts news with an image to verify then this program will search all over the internet for its original source and will provide a set of data (i.e. links, references, image data, first-time online appearance etc.). These data will be later used by an expert to write a reliable report.

Oigetit Fake News Filter	OwlFactor	WhatsFarzi	NewsCop
OIGETIT follows approximately 1M news sources from all over the world in order to provide the most comprehensive, verified and trusted news on the web.	OwlFactor's algorithm has evaluated 2 million articles and has ratings for over 2,000 news sources and 30,000 journalists.	The app will track content on the internet and check it with authentic websites to curb spread of misinformation otherwise known as fake news	NewsCop is a community based and non-profitable platform which lets you verify news, spoilers etc. It uses a variety of methods to provide you with the best conclusion possible. It uses mainly 3 methods to determine any conclusion or result of

			the uploaded post. They are: - a) Poll Method b) Expert Review c) Cloud-based AI Reference finder
They say they use a database of historical articles, but how do we know their database is accurate? Someone has to fact check the database, which is open to personal bias.		It turned out that it marks everything as fake. I think this is a serious flaw from developer side. Hope to see a better app in future. Showing all the trusted and authentic news sources as fake.	Essentially based on other users' consensus as to whether or not something is real. You cannot search anything.

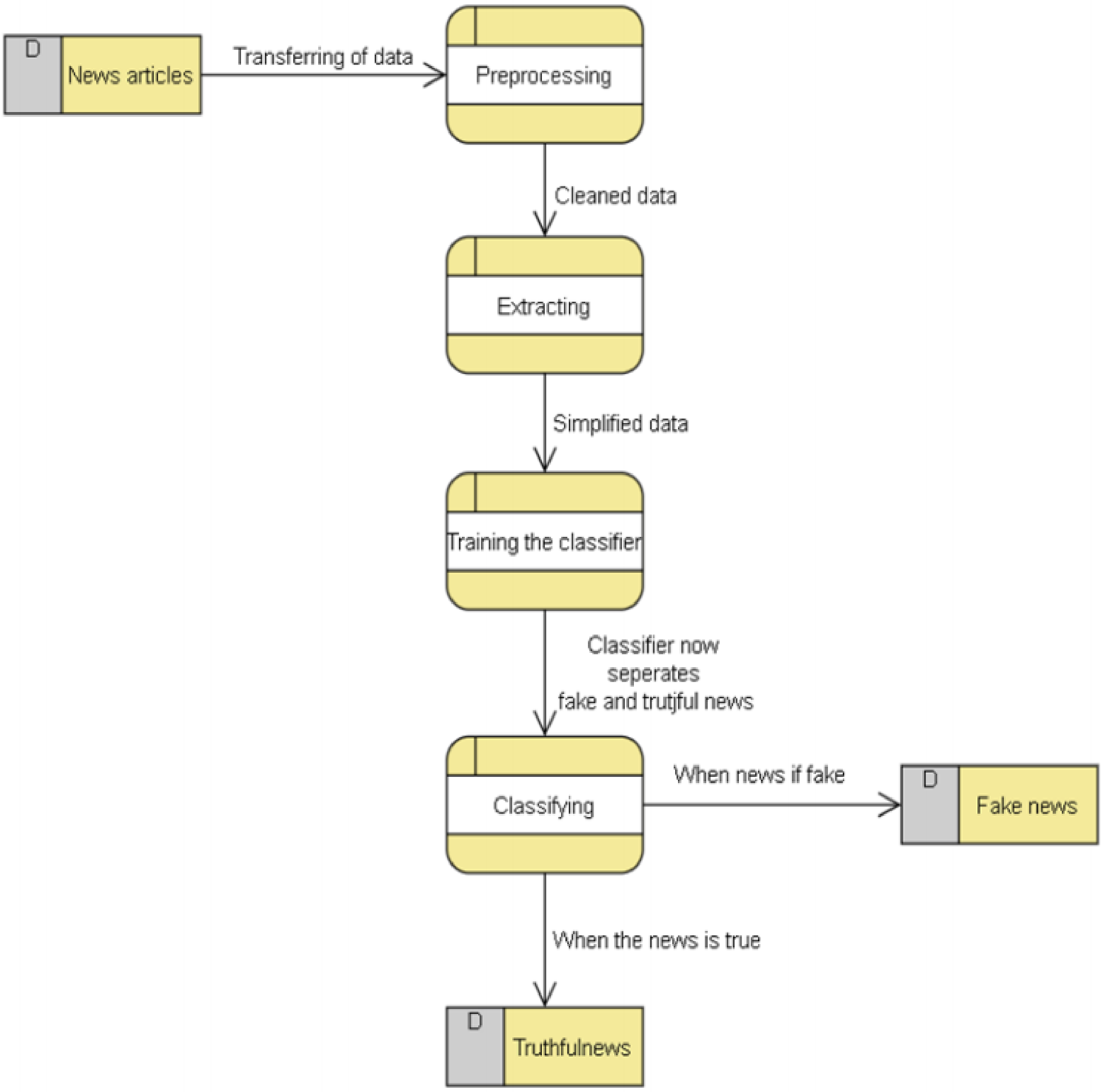
Use Case Diagram


System
Classifies the news articles into truthful and fake news
User
Extracting the data to make it more manageable for the classifier
Preprocessing the data to make it more understable
Database
Admin

Explanation: -

- In the above Use Case Diagram, the data taken from the database is preprocessed.
- Then all this data is simplified (manageable) by removing all the unnecessary data.
- It is then fed to the classifier which classifies the given data (news article) into real or fake news and shows it to the user.

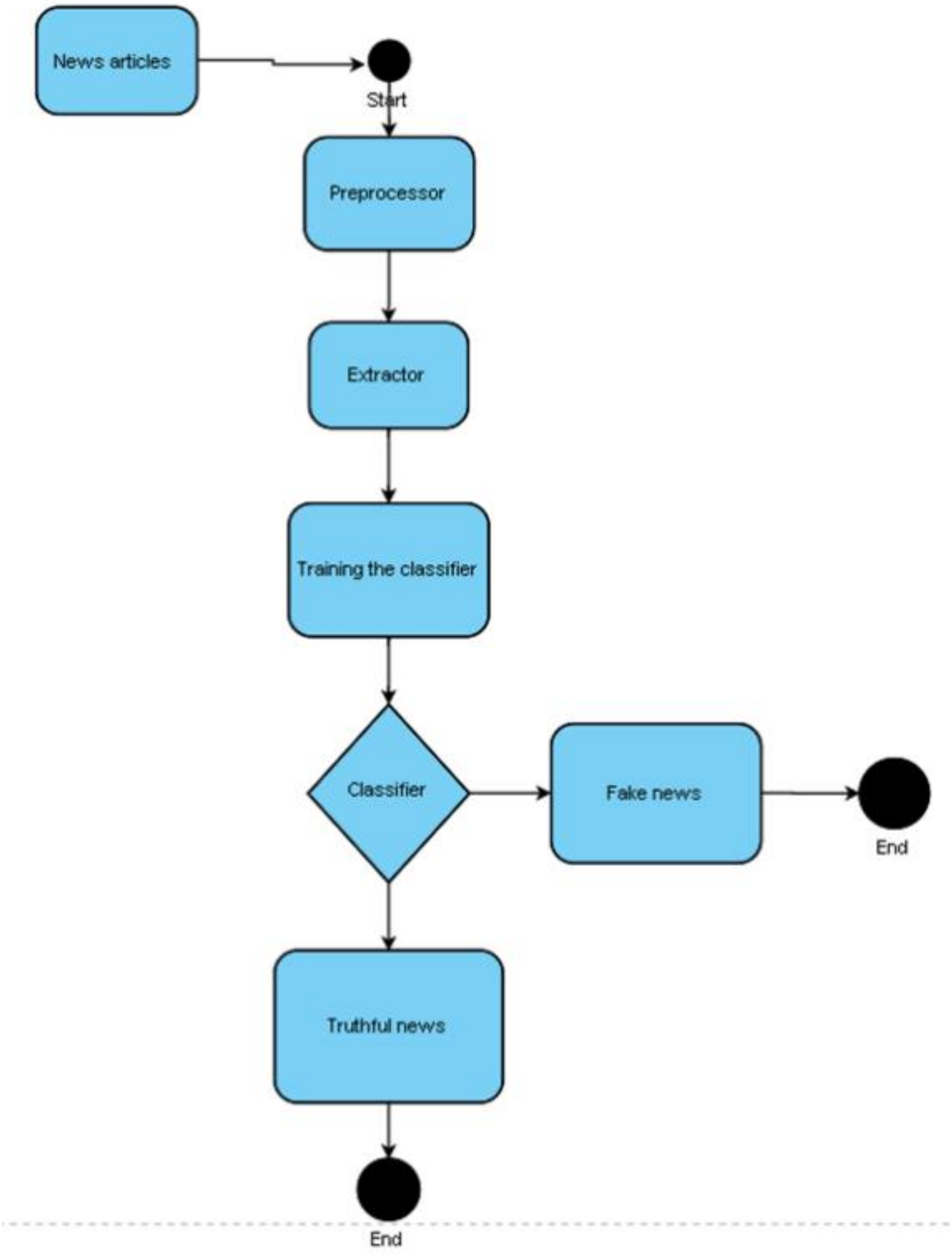
DFD Diagram



Explanation: -

- In the above Data Flow Diagram, the data taken from the database is preprocessed where it is cleaned.
- Then all this cleaned data is simplified by removing all the unnecessary data during extracting.
- Using this data, the classifier is trained.
- Which then is able to separate fake news articles from truthful articles using different classification techniques and shows it to the user.

Activity Diagram



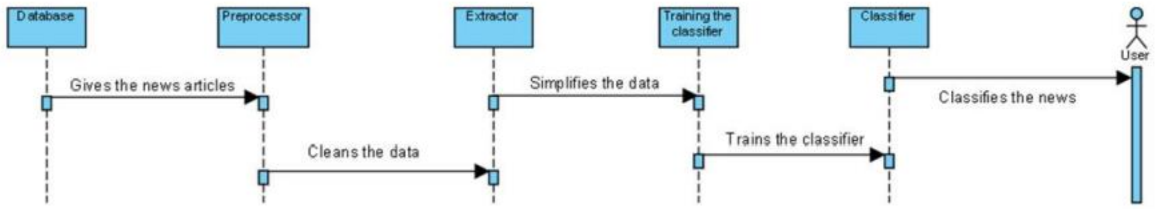
Explanation: -

- In the above Activity Diagram, the data taken from the database is preprocessed in the preprocessor where it is cleaned.
- Then all this cleaned data is simplified by removing all the unnecessary data during extracting in the extractor.
- Using this data, the classifier is trained.
- Then the classifier uses various classification techniques classifies the news as: -
  1. If the news if real news it will classify it as "Real news" and show it to the user.
  2. If the news if fake news it will classify it as "Fake news" and show it to the user.

Data Dictionary Diagram

S.no	Name	Data Type	Length
1	Id (PK)	Int	5
2	Title	Varchar	100
3	Text	Varchar	5000
4	Label	Boolean	1
5	URL	Varchar	300

Sequence Diagram

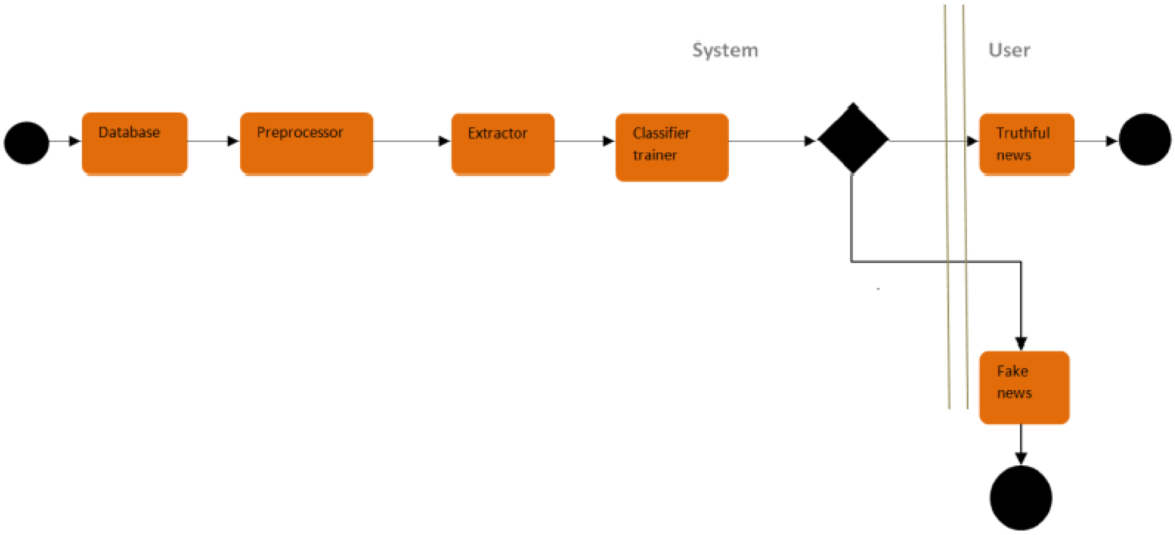


Explanation: -

The above Sequence Diagram shows the sequence of classifying the news: -

- The data taken from the database is preprocessed in the preprocessor where it is cleaned.
- Then all this cleaned data is simplified by removing all the unnecessary data during extracting in the extractor.
- Using this data, the classifier is trained.
- Then the classifier uses various classification techniques classifies the news as Real or Fake news and shows it to the user.

State Transition Diagram



Explanation: -

The above State Transition Diagram shows how the news is classified in their given states.

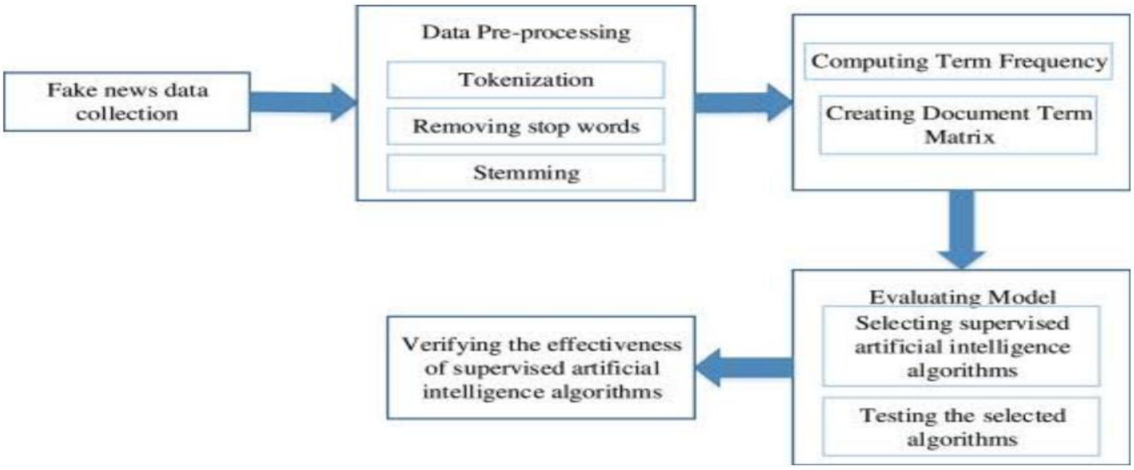
a) In the system state: -

- The data taken from the database is preprocessed in the preprocessor where it is cleaned.
- Then all this cleaned data is simplified by removing all the unnecessary data during extracting in the extractor.
- Using this data, the classifier is trained.
- Then the classifier uses various classification techniques classifies the news as Real or Fake news.

b) In the user state: -

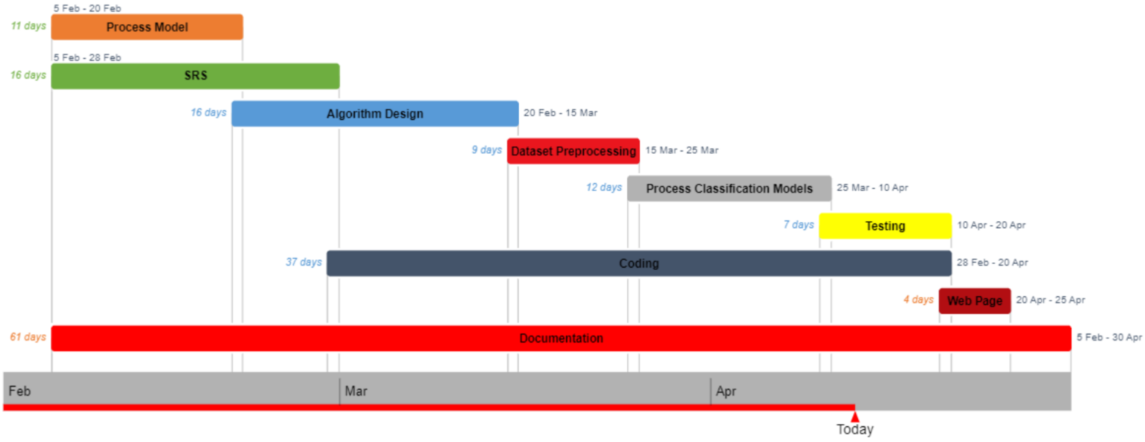
- The classified Real or Fake news is then shown to the user.

Workflow Diagram


Fake news data collection
Data Pre-processing
Tokenization
Removing stop words
Stemming
Computing Term Frequency
Creating Document Term Matrix
Evaluating Model
Selecting supervised artificial intelligence algorithms
Testing the selected algorithms
Verifying the effectiveness of supervised artificial intelligence algorithms

Timeline Chart

Fake News Detection

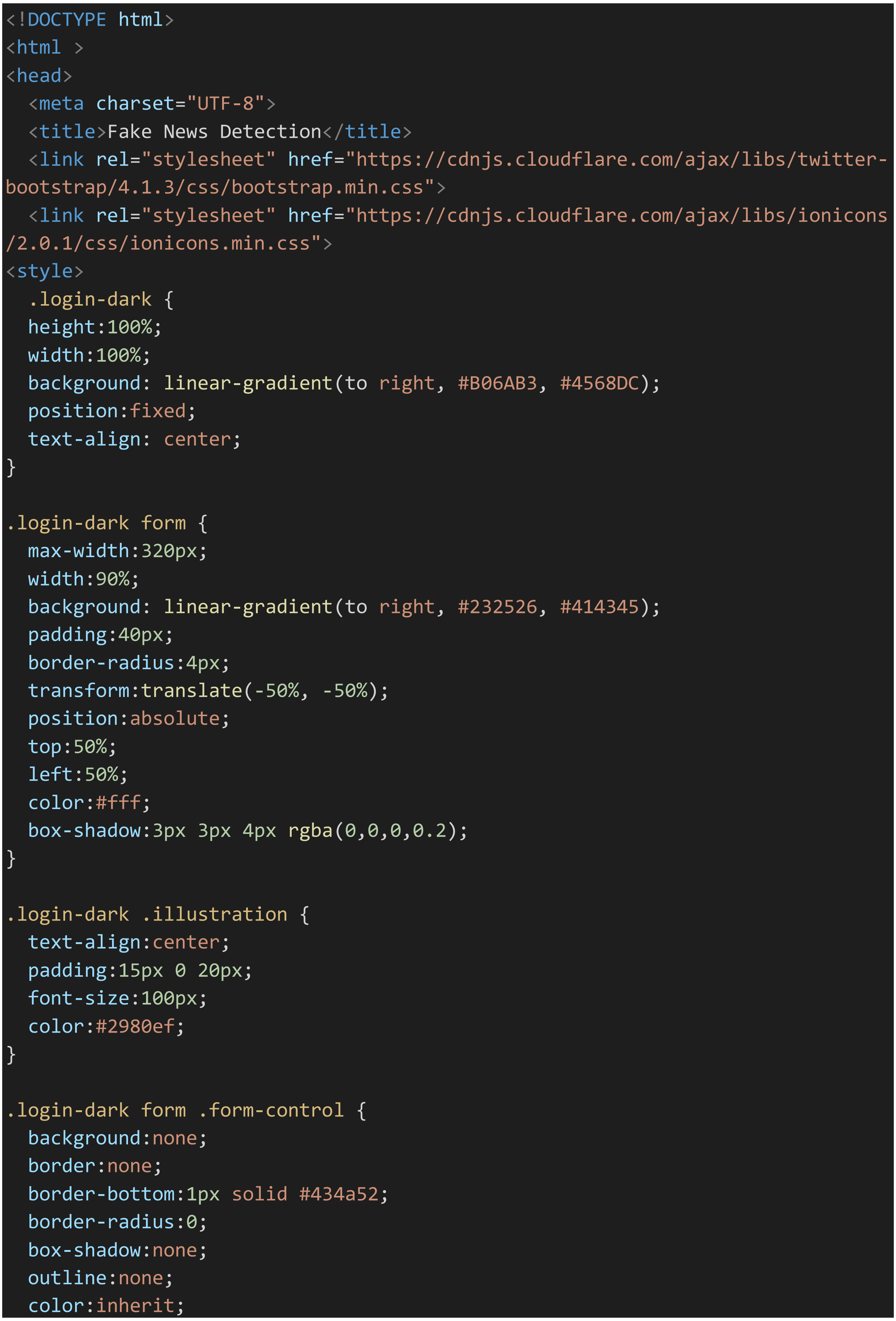


Implementation

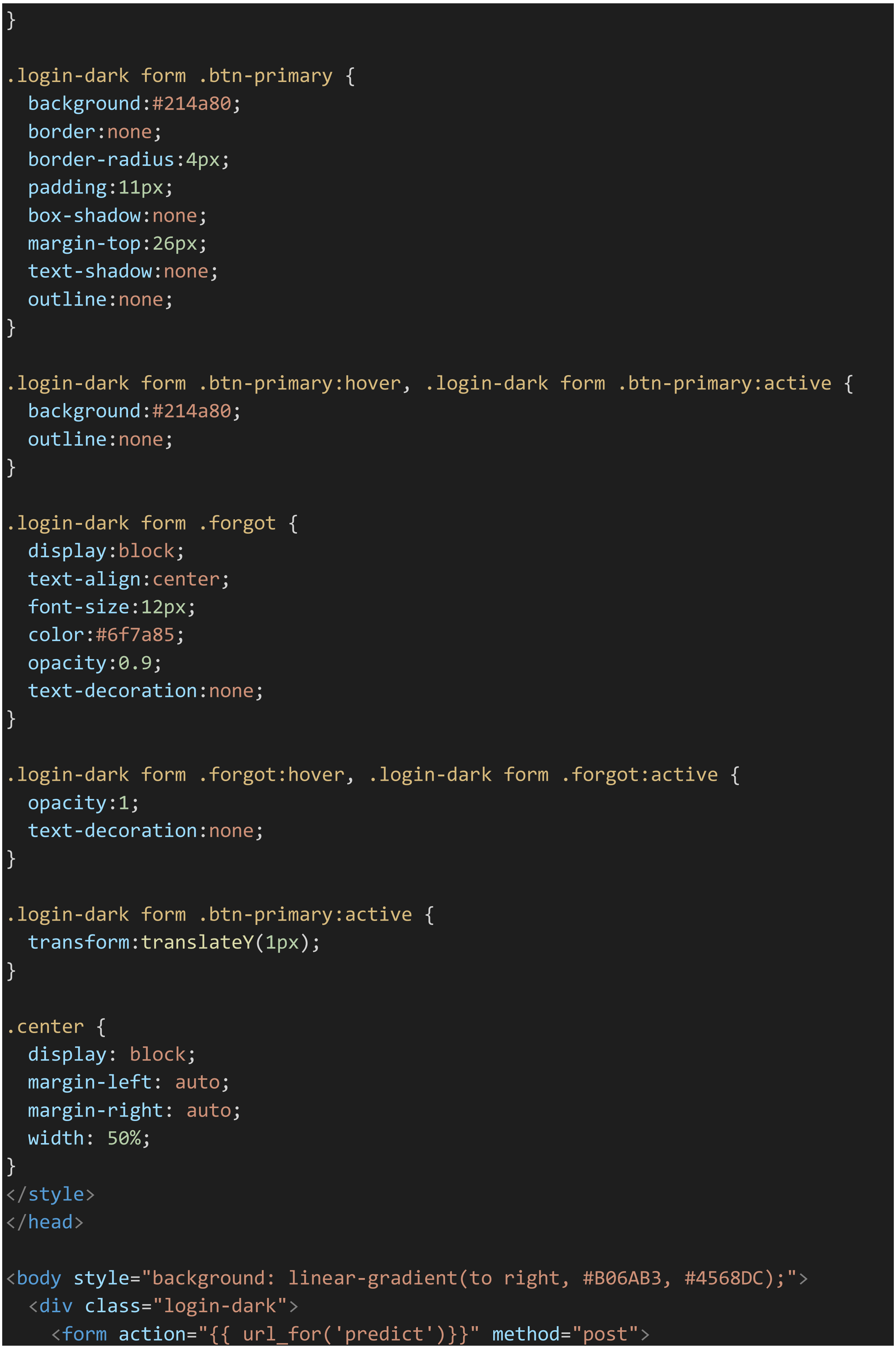
1. app.py

```
#Importing the Libraries
import numpy as np
from flask import Flask, request,render_template
from flask_cors import CORS
import os
import joblib
import pickle
import flask
import os
import newspaper
from newspaper import Article
import urllib
import pandas as pd
import re
import string

#Loading Flask and assigning the model variable
app = Flask(__name__)
CORS(app)
app=flask.Flask(__name__,template_folder='templates')

with open('C:/Users/Radley Roy/OneDrive/Desktop/College/Fake_news/LR_model.sav', 'rb') as handle:
    LR = joblib.load(handle)
with open('C:/Users/Radley Roy/OneDrive/Desktop/College/Fake_news/DT_model.sav', 'rb') as handle:
    DT = joblib.load(handle)
with open('C:/Users/Radley Roy/OneDrive/Desktop/College/Fake_news/GBC_model.sav', 'rb') as handle:
    GBC = joblib.load(handle)
with open('C:/Users/Radley Roy/OneDrive/Desktop/College/Fake_news/RFC_model.sav', 'rb') as handle:
    RFC = joblib.load(handle)
with open('C:/Users/Radley Roy/OneDrive/Desktop/College/Fake_news/tfidf_model.sav', 'rb') as handle:
    vectorization = joblib.load(handle)

def wordopt(text):
    text = text.lower()
    text = re.sub('\[.*?\]', '', text)
    text = re.sub("\\W"," ",text)
    text = re.sub('https?://\S+|www\.\S+', '', text)
    text = re.sub('<.*?>+', '', text)
    text = re.sub('[%s]' % re.escape(string.punctuation), '', text)
    text = re.sub('\n', '', text)
    text = re.sub('\w*\d\w*', '', text)
    return text

@app.route('/')
def main():
    return render_template('main.html')

@app.route('/predict',methods=['GET','POST'])
def predict():
    url =request.get_data(as_text=True)[5:]
    url = urllib.parse.unquote(url)
    article = Article(str(url))
    article.download()
    article.parse()
    article.nlp()
    news = article.summary
    testing_news = {"text":[news]}
    new_def_test = pd.DataFrame(testing_news)
    new_def_test["text"] = new_def_test["text"].apply(wordopt)
    new_x_test = new_def_test["text"]
    new_xv_test = vectorization.transform(new_x_test)
    pred_LR = LR.predict(new_xv_test)
    pred_DT = DT.predict(new_xv_test)
    pred_GBC = GBC.predict(new_xv_test)
    pred_RFC = RFC.predict(new_xv_test)
    prob_LR = round(LR.predict_proba(new_xv_test)[0][0]*100,2)
    prob_DT = round(DT.predict_proba(new_xv_test)[0][0]*100,2)
    prob_GBC = round(GBC.predict_proba(new_xv_test)[0][0]*100,2)
    prob_RFC = round(RFC.predict_proba(new_xv_test)[0][0]*100,2)
    check_LR = False
    check_RFC = False
    check = True
    if prob_LR <= 80:
        check_LR = True
    if prob_RFC <= 70:
        check_RFC = True
    return render_template('main.html', check=check, check_LR = check_LR, chec
k_RFC = check_RFC, LR = prob_LR, DT = prob_DT, GBC = prob_GBC, RFC = prob_RFC)

if __name__=="__main__":
    port=int(os.environ.get('PORT',5000))
    app.run(port=port,debug=True,use_reloader=False)
```

2. main.html



```
<!DOCTYPE html>
<html >
<head>
  <meta charset="UTF-8">
  <title>Fake News Detection</title>
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-
bootstrap/4.1.3/css/bootstrap.min.css">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/ionicons
/2.0.1/css/ionicons.min.css">
<style>
  .login-dark {
  height:100%;
  width:100%;
  background: linear-gradient(to right, #B06AB3, #4568DC);
  position:fixed;
  text-align: center;
}

.login-dark form {
  max-width:320px;
  width:90%;
  background: linear-gradient(to right, #232526, #414345);
  padding:40px;
  border-radius:4px;
  transform:translate(-50%, -50%);
  position:absolute;
  top:50%;
  left:50%;
  color:#fff;
  box-shadow:3px 3px 4px rgba(0,0,0,0.2);
}

.login-dark .illustration {
  text-align:center;
  padding:15px 0 20px;
  font-size:100px;
  color:#2980ef;
}

.login-dark form .form-control {
  background:none;
  border:none;
  border-bottom:1px solid #434a52;
  border-radius:0;
  box-shadow:none;
  outline:none;
  color:inherit;
```

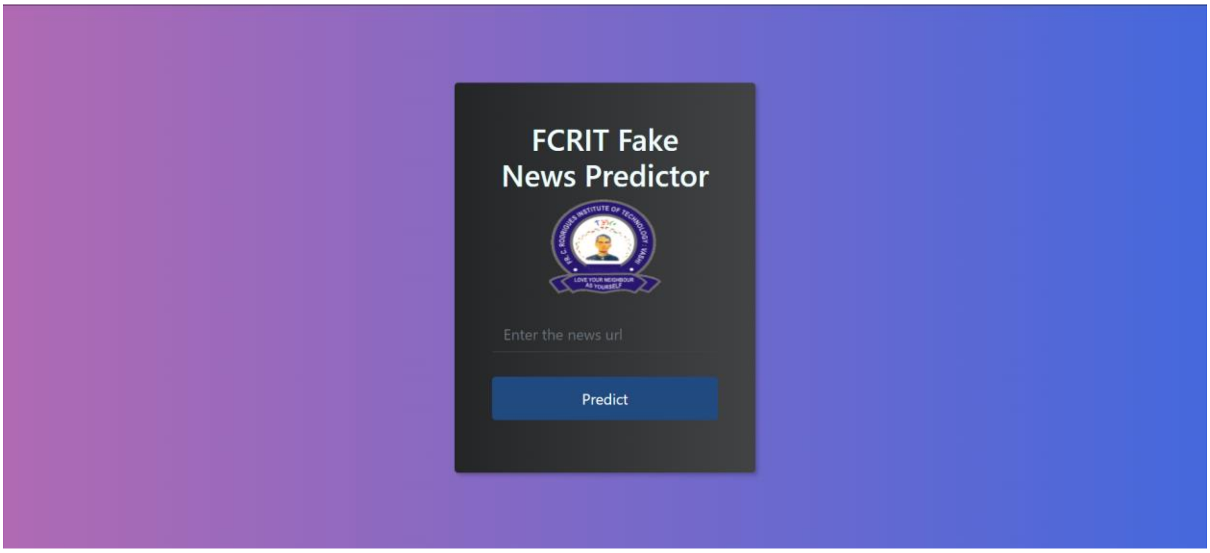


```
}

.login-dark form .btn-primary {
  background:#214a80;
  border:none;
  border-radius:4px;
  padding:11px;
  box-shadow:none;
  margin-top:26px;
  text-shadow:none;
  outline:none;
}

.login-dark form .btn-primary:hover, .login-dark form .btn-primary:active {
  background:#214a80;
  outline:none;
}

.login-dark form .forgot {
  display:block;
  text-align:center;
  font-size:12px;
  color:#6f7a85;
  opacity:0.9;
  text-decoration:none;
}

.login-dark form .forgot:hover, .login-dark form .forgot:active {
  opacity:1;
  text-decoration:none;
}

.login-dark form .btn-primary:active {
  transform:translateY(1px);
}

.center {
  display: block;
  margin-left: auto;
  margin-right: auto;
  width: 50%;
}
</style>
</head>

<body style="background: linear-gradient(to right, #B06AB3, #4568DC);">
  <div class="login-dark">
    <form action="{{ url_for('predict')}}" method="post">
```

```
      <h2 style="color:azure; text-align: center;">FCRIT Fake News Predictor</h2>
      <img src="https://fcrit.ac.in/img/fcritlogo.png" style="height:100px;" class="center">
        <h2 class="sr-only">Login Form</h2>
        <br>
        <div class="form-group"><input class="form-control" type="text" name="news" placeholder="Enter the news url" required="required"></div>
        <div class="form-group"><button class="btn btn-primary btn-block" type="submit">Predict</button></div>

        {% if check %}
        {% if check_LR or check_RFC  %}
         <div style="font-size:30px; color:green; font-weight: 500;">Not a Fake News</div>
        {% else %}
         <div style="font-size:30px; color:red; font-weight: 500;">Fake News</div>
        {% endif %}

        <div>LR Prediction : {{ LR }}%</div>
        <div>DT Prediction : {{ DT }}%</div>
        <div>GBC Prediction : {{ GBC }}%</div>
        <div>RFC Prediction : {{ RFC }}%</div>
        {% endif %}
      </form>

</div>
<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.3/js/bootstrap.bundle.min.js"></script>



</body>
</html>
```

Results

1. Home Page

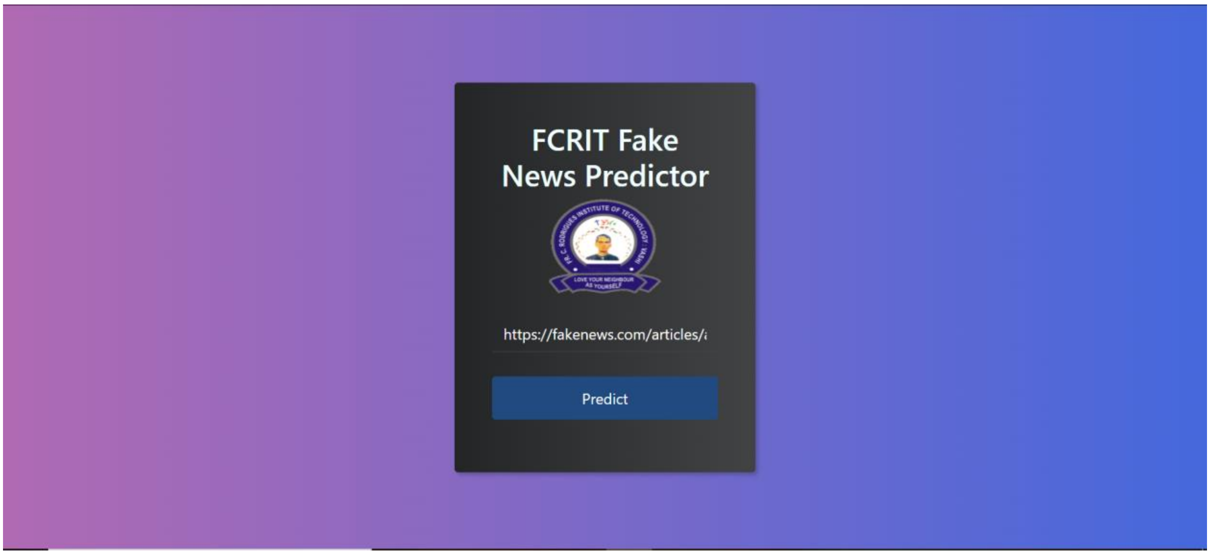


FCRIT Fake News Predictor

Enter the news url

Predict

2. Enter the URL in the space and press "Predict".

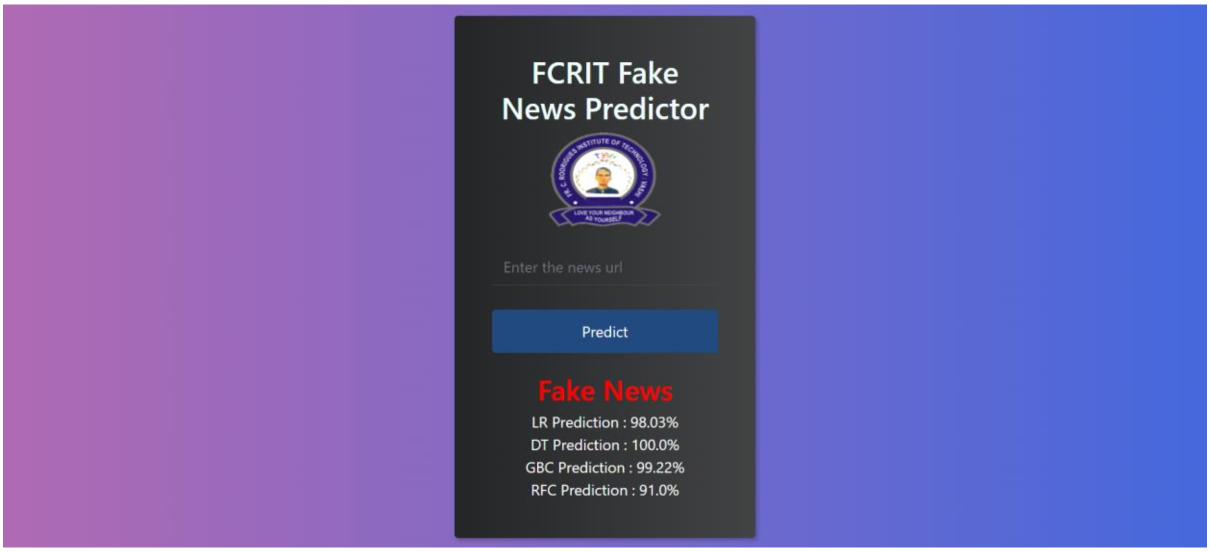


FCRIT Fake News Predictor

https://fakenews.com/articles/

Predict

3. Fake News

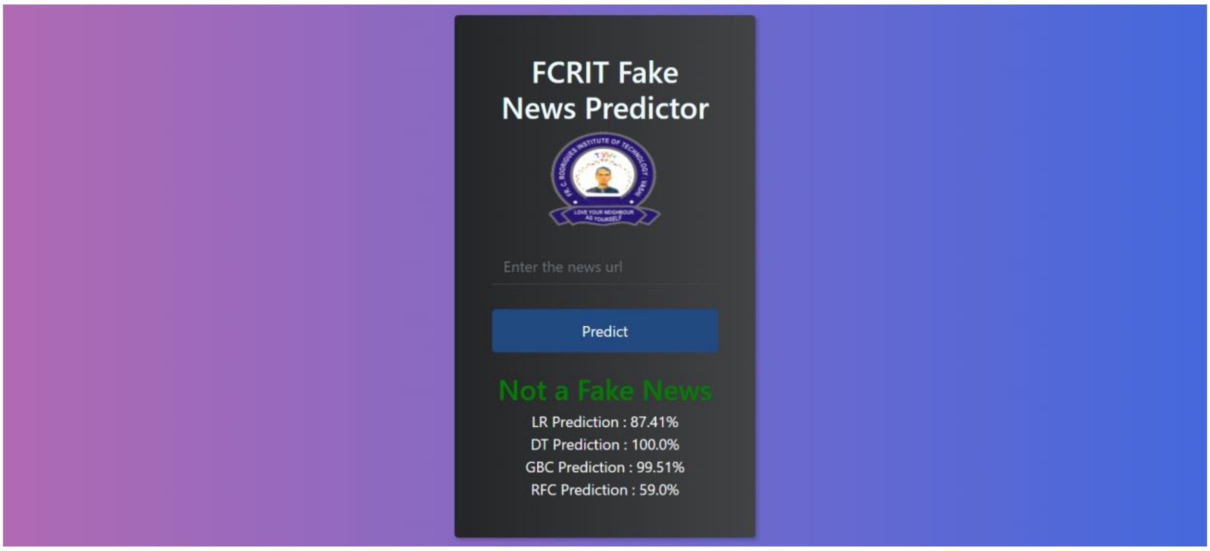


FCRIT Fake News Predictor

Fake News

LR Prediction : 98.03%
DT Prediction : 100.0%
GBC Prediction : 99.22%
RFC Prediction : 91.0%

4. Not a Fake News



FCRIT Fake News Predictor

Not a Fake News

LR Prediction : 87.41%
DT Prediction : 100.0%
GBC Prediction : 99.51%
RFC Prediction : 59.0%

Conclusion

The task of classifying news manually requires in-depth knowledge of the domain and expertise to identify anomalies in the text. In this research, we discussed the problem of classifying fake news articles using machine learning models and ensemble techniques. The primary aim of the research is to identify patterns in text that differentiate fake articles from true news. The learning models were trained and parameter-tuned to obtain optimal accuracy. Some models have achieved comparatively higher accuracy than others. We used multiple performance metrics to compare the results for each algorithm. Fake news detection has many open issues that require attention of researchers. For instance, in order to reduce the spread of fake news, identifying key elements involved in the spread of news is an important step.

References

[1] C. Silverman, "BuzzFeed News," 16 November 2016. [Online]. Available: https://www.buzzfeednews.com/article/craigsilverman/viral-fake-election-news-outperformed-real-news-on-facebook. [Accessed 19 April 2021].

[2] M. CALABRESI, "Time," 9 August 2017. [Online]. Available: https://time.com/4783932/inside-russia-social-media-war-america/. [Accessed 19 April 2021].

[3] M. P. Lynch, "The New York Times," 28 November 2016. [Online]. Available: https://www.nytimes.com/2016/11/28/opinion/fake-news-and-the-internet-shell-game.html. [Accessed 19 April 2021].

[4] J. Albright, "Welcome to the Era of Fake News," *Media and Communication,* vol. 5, no. 2, pp. 87-89, 2017.

[5] Y. C. N. R. V. Chen, "Misleading Online Content | Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection," 2015. [Online]. Available: https://dl.acm.org/doi/10.1145/2823465.2823467.

[6] C.-S. Atodiresei, A. Tănăselea and A. Iftene, "Identifying Fake News and Fake Users on Twitter," *Procedia Computer Science,* vol. 126, pp. 451-461, 2018.

[7] Reis, J. C. S., Correia, A. a. Murai, F. a. Veloso, A. a. Benevenuto and Fabrício, "Supervised Learning for Fake News Detection," *IEEE Intelligent Systems,* vol. 34, no. 2, pp. 76-81, 2019.

[8] Ruchansky, Natali, Seo, Sungyong, Liu and Yan, "CSI: A Hybrid Deep Model for Fake News Detection," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, New York, NY: Association for Computing Machinery, 2017, pp. 797-806.

[9] Karimi, Hamid, Roy, Proteek, Saba-Sadiya, Sari, Tang and Jiliang, "Multi-Source Multi-Class Fake News Detection," in *Proceedings of the 27th International Conference on Computational Linguistics*, Santa Fe, ssociation for Computational Linguistics, 2018, pp. 1546-1557.

[10] [Online]. Available: https://play.google.com/store/apps/details?id=io.scal.oigetit&hl=en&gl=US.

[11] [Online]. Available: https://apps.apple.com/us/app/civikowl/id1327496013.

[12] [Online]. Available: https://apps.apple.com/us/app/civikowl/id1327496013.

[13] [Online]. Available: https://play.google.com/store/apps/details?id=com.surajgiri.newsage&hl=en_AU.