Phase-3

Exposingthetruthwithadvancedfakenews detection powered by natural language processing

Student Name: S. Sembaruthi

Register Number: 620123106104

Institution:AVSEngineeringCollege

Department: ElectronicsandCommunication Engineering

Date of Submission: 13.05.2025

Github Repository Link:https://github.com/Sembaruthi-S/Nan- Muthalvan-Project-

---

1. ProblemStatement

Thedigitalerahasmadethespreadofmisinformationeasyandrapid. Fake news, especially on socialmedia, can manipulate public opinion, incite unrest, and cause harm. A reliable and automated solution isneeded to detect such news articles in real-time using modern AI techniques

2.Abstract

This project aims to detect fake news using Natural Language Processing(NLP)andmachinelearning.Byanalyzingtextualpatterns in news content, the system classifies whether a news article is real or fake.The project includes preprocessing, feature extraction, model training,evaluation,anddeploymentofaweb-basedinterfaceforuser interaction.

3. System Requirements

Hardware:

- *RAM:8 GB or more*
- *CPU:Inteli5/i7 orAMDequivalent*
- *GPU:Recommendedfordeeplearning*

Software:

- *OS: Windows/Linux/Mac*
- *Python3.8+*
- *Libraries:Pandas,NumPy,Matplotlib,NLTK,Scikit-learn, TensorFlow, Flask*
- *Tools:JupyterNotebook,VSCode,Postman(fortestingAPI*

4. Objectives

Collectandcleanfakeandrealnewsdatasets.

- *ApplyNLPtechniquestopreprocessandanalyzethedata.*
- *Buildclassificationmodelstodistinguishfakefromrealnews.*
- *Evaluatemodelperformance withrobust metrics.*
- *Deploythemodelasaweb-based prediction tool.*

5.FlowchartofProjectWorkflow

START

->DataCollection

->DataPreprocessing

->EDA

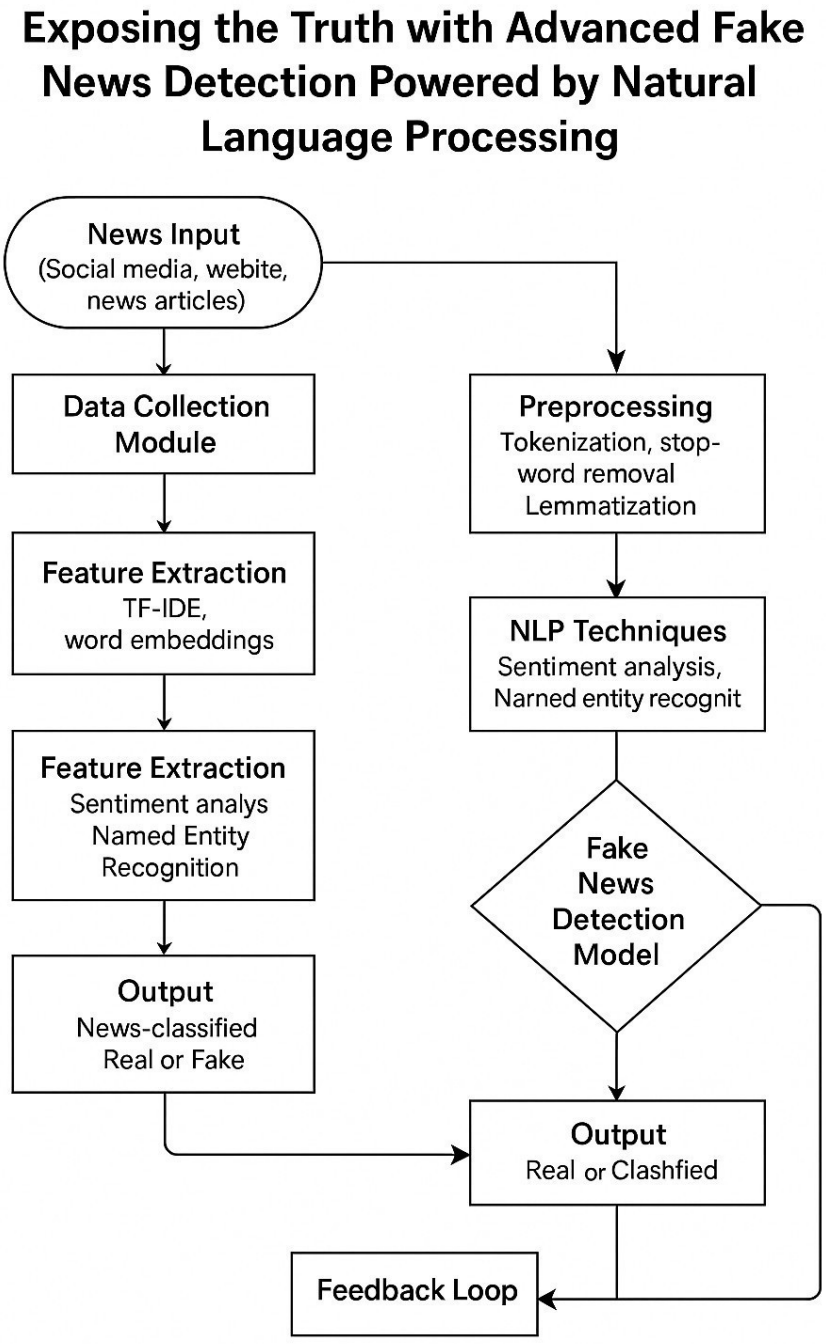
->FeatureEngineering

->ModelBuilding

->Evaluation

->Deployment

->END



6. Dataset Description

Source:Kaggle -Fake andReal NewsDataset

- ***Features:title, text,subject, date, label***
- ***Size: ~40,000 articles***
- ***Labels: FAKE, REAL***

7.Data Preprocessing

Converttext to lowercase

- *Removepunctuation,numbers,stopwords*
- *TokenizationandLemmatization(usingNLTKorspaCy)*
- *Handlemissing/nullvalues*
- *Combinetitle +text asinput features*

8.ExploratoryDataAnalysis(EDA)

Classbalancecheck(Fakevs.Real)

- *Wordcloudvisualizationforbothclasses*
- *Top word frequency count*
- *Articlelengthdistribution*
- *Sentimentanalysis(optional)*

9.FeatureEngineering

TF-IDF Vectorization

- *CountVectorizer*
- *N-gramanalysis(bi-gram,tri-gram)*
- *Wordembeddings(optional:GloVe,Word2Vec)*
- *Dimensionalityreduction(PCA/TruncatedSVDifneeded)*

10. Model Building

Modelsused:

- *LogisticRegression*
- *Naive Bayes*
- *Random Forest*
- *XGBoost*
- *LSTM/BERT(optionalforadvancedversion)*

Hyperparameter tuning via:

- *GridSearchCV*
- *Cross-validation(k-fold)*

11. Model Evaluation

Metrics:

- *Accuracy*
- *Precision,Recall, F1-Score*
- *ConfusionMatrix*
- *ROC-AUC curve*

Useavalidationsplit(e.g.,80-20or70-30)and cross-validation.

12. Deployment

Flask-based web app

- *Userinputsnewstextviaaform*
- *Backendprocessestheinputandreturnsaprediction*
- *Hostedonplatformslike:*
- *Heroku*
- *Render*
- *Streamlit(alternative GUI)*

13. Sourcecode

Install Required Libraries

```
pipinstallpandasscikit-learnnltk
```

Import Libraries

```
importpandasasp

fromsklearn.model_selectionimporttrain_test_split
from sklearn.feature_extraction.text import TfidfVectorizer from

sklearn.linear_model import PassiveAggressiveClassifier
```

```
fromsklearn.metricsimportaccuracy_score,confusion_matrix
import nltk
fromnltk.corpusimportstopwords
fromnltk.stemimportWordNetLemmatizer
import
re
```

Load and Preprocess

```
Datanltk.download('stopwords')
nltk.download('wordnet')
    df=pd.read_csv('fake_or_real_news.csv')#Assumes'text'and
'label' columns
lemmatizer = WordNetLemmatizer()
stop_words=set(stopwords.words('english'))
def preprocess(text): text=re.sub(r'\
W','',text.lower())
words=text.split()
    words=[lemmatizer.lemmatize(w)forwinwordsifwnot stop_words]
return''.join(words)
    df['text']=df['text'].apply(preprocess)
```

Feature Extraction

```
X = df['text']
y = df['label']
tfidf=TfidfVectorizer(max_df=0.7)
  X_tfidf = tfidf.fit_transform(X)
```

TrainModel

```
X_train,X_test,y_train,y_test=train_test_split(X_tfidf,y,
test_size=0.2,
random_state=42)
model=PassiveAggressiveClassifier(max_iter=50)
model.fit(X_train,
y_train)
```

14. Futurescope

Real-timefakenewsdetectionusingTwitter/FacebookAPIs
- *Transformer-basedmodels(e.g.,BERT,RoBERTa)*
- *Multilingualfakenews detection*
- *Image/videomisinformationanalysis*
- *Browserextensionintegration*

15. TeamMembersandRoles

Here is a well-written paragraph detailing the roles of S. Sembaruthi,J.ShabanaMirza,G.Shakthi,andr.NagaIshwarya withS.Sembaruthigiventhemostcriticalandhigh-impact responsibilities:

IntheprojecttitledExposingtheTruthwithAdvancedFake News Detection Powered by Natural Language Processing"

S. Sembaruthi

S. Sembaruthi served as the Project Lead and Core Developer, playing a pivotal role in the successful execution of the entire system. She was responsible for designing the system architecture, implementing advanced NLP pipelines, and developing the machine learning model that powers the fake news detection engine. Additionally, she led the integration of the backend with the frontend using Flask and ensured the

deployment was robust and user-ready. Hertechnicalleadership,decision-making,andhands-ondevelopment made her the cornerstone of the project.

J.Shabana Mirza

J. Shabana Mirza contributed as the NLP and Data Processing Engineer,focusingoncollecting,cleaning,andpreparingthedataset, and applying natural language techniques such as tokenization, stop- word removal, and lemmatization to enhance data quality.

G. Shakthi

G. Shakthi took charge as the Frontend and Deployment Specialist,designinganintuitiveuserinterfaceandensuringseamless integration between the user input and the model's prediction output through a web application.

R. Naga Ishwarya

Finally,R.NagaIshwaryahandledtheroleofQualityAssurance and Documentation Lead, conducting rigorous model testing, debugging, and preparing comprehensive documentation and user manuals to ensure clarity and maintainability of the system.