Development Part 2 - Fake News Detection Using NLP

Introduction

Phase 4 represents a critical stage in the project's development journey. In this phase, we continue to enhance the Fake News Detection model by applying advanced **Natural Language Processing (NLP)** techniques, selecting a machine learning algorithm, training the model, and conducting a comprehensive evaluation. The primary objective remains the same to build a robust system capable of effectively distinguishing between True and fake news articles.

Data Source

For this phase 4, we rely on a valuable dataset available on Kaggle, which contains a mixture of real and fake news articles. This dataset serves as the foundation for training and evaluating our Fake News Detection model.

The dataset's link is provided: https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset

Text Preprocessing and Feature Extraction

Text Preprocessing

Text preprocessing is an essential step to ensure that the data is in the right format for model training. The following are the key text preprocessing steps we undertake:

1. **Text Lowercasing**: NLP techniques are employed to clean and preprocess the text data.The text is converted to lowercase. This standardization helps ensure that the model doesn't treat words with different casings as distinct entities, improving consistency in the data.

2. **Punctuation and Number Removal**: We eliminate punctuation and numbers from the text. These characters are typically not informative in the context of the classification task.

3. **Stopword Removal**: Common stopwords, such as "the," "and," and "is," are removed. These words, although frequently occurring, do not carry substantial meaning and can be safely omitted from the text.

4. **Tokenization**: Tokenization is the process of splitting the text into individual words or tokens. This step prepares the text for feature extraction.

Feature Extraction

Feature extraction is the transformation of text data into numerical features that machine learning algorithms can work with. In this phase, we employ the TF-IDF (Term Frequency-Inverse Document Frequency) technique, which offers the following benefits:

- Measures the importance of each word in a document relative to a collection of documents (corpus).
- Assigns a weight to each word based on its frequency in the document and its rarity in the corpus.

Model Training and Evaluation

Model Selection

Selecting an appropriate machine learning algorithm is a critical decision. For Phase 4, we have chosen the **Logistic Regression** algorithm. Logistic Regression is a well-established choice for binary classification tasks and serves as a strong starting point for our Fake News Detection model.

Model Training

The selected model, Logistic Regression, is trained using the pre-processed and feature-extracted data. The dataset is thoughtfully split into training and testing sets to ensure that the model's performance is evaluated without bias.

Model Evaluation

The evaluation of the model's performance is a central aspect of this phase. We use a set of evaluation metrics tailored to the problem of Fake News Detection:

- **Accuracy**: This metric measures the overall correctness of the model's predictions. It provides a general sense of how well the model is performing.

- **Precision**: Precision indicates the model's ability to correctly classify fake news articles. It measures the ratio of true positives to the total predicted positives.

- **Recall**: Recall measures the model's ability to correctly classify fake news articles. It quantifies the ratio of true positives to the total actual positives.

- **F1 Score**: The F1 score is the harmonic mean of precision and recall. It offers a balanced assessment of the model's performance, particularly useful when there is an imbalance between classes.

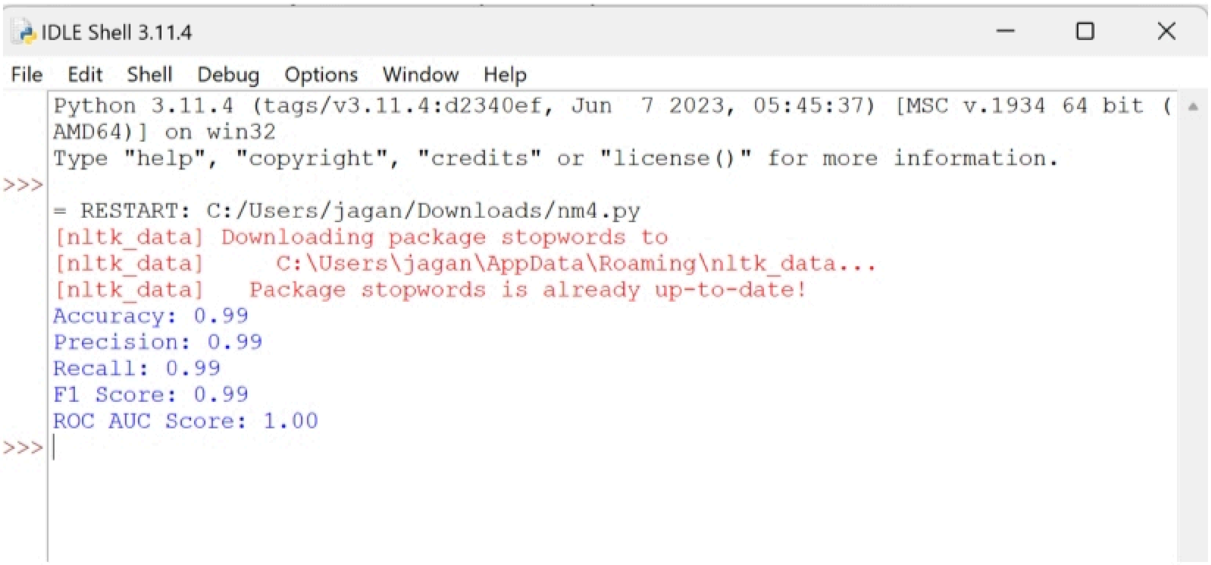
- **ROC AUC Score**: The Receiver Operating Characteristic Area Under the Curve (ROC AUC) score evaluates the model's ability to discriminate between genuine and fake news across different thresholds.

Code:

```
import pandas as pd
import re
import nltk
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, roc_auc_score
nltk.download('stopwords')
true_data = pd.read_csv(r'C:\Users\jagan\Downloads\phase4\True.csv')
fake_data = pd.read_csv(r'C:\Users\jagan\Downloads\phase4\Fake.csv')
true_data['label'] = 'true'
fake_data['label'] = 'fake'
data = pd.concat([true_data, fake_data])
data['text'] = data['text'].str.lower()
data['text'] = data['text'].apply(lambda x: re.sub(r'[^\w\s]', '', x))
from nltk.corpus import stopwords
stop_words = set(stopwords.words('english'))
data['text'] = data['text'].apply(lambda x: ' '.join([word for word in x.split() if word not in stop_words]))
X = data['text']
y = data['label']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
tfidf_vectorizer = TfidfVectorizer(max_features=5000)
X_train_tfidf = tfidf_vectorizer.fit_transform(X_train)
X_test_tfidf = tfidf_vectorizer.transform(X_test)
model = LogisticRegression()
model.fit(X_train_tfidf, y_train)
y_pred = model.predict(X_test_tfidf)
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, pos_label='fake')
```

```
recall = recall_score(y_test, y_pred, pos_label='fake')

f1 = f1_score(y_test, y_pred, pos_label='fake')

roc_auc = roc_auc_score(y_test, model.predict_proba(X_test_tfidf)[:, 1])

print(f"Accuracy: {accuracy:.2f}")

print(f"Precision: {precision:.2f}")

print(f"Recall: {recall:.2f}")

print(f"F1 Score: {f1:.2f}")

print(f"ROC AUC Score: {roc_auc:.2f}")
```

Output:



```
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/jagan/Downloads/nm4.py
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\jagan\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
Accuracy: 0.99
Precision: 0.99
Recall: 0.99
F1 Score: 0.99
ROC AUC Score: 1.00
>>>
```

Conclusion

Phase 4 represents a substantial leap forward in the Fake News Detection project. We have meticulously prepared the textual data through text preprocessing and harnessed the power of TF-IDF for feature extraction. The selection of Logistic Regression as our machine learning algorithm is a well-considered decision that forms a strong foundation for our model.

The model training and evaluation steps are crucial in assessing its performance. The evaluation metrics provide a comprehensive view of how well the model can distinguish between real and fake news articles. The insights gained in this phase will guide us in further refinements and enhancements in the upcoming phases.

This phase sets the stage for the subsequent work, bringing us closer to the goal of creating an effective and reliable Fake News Detection system.