Fake News Detection Using NLP

Phase 4 submission

Text Preprocessing and Feature Extraction ->

Dataset Preparation

In this section, we begin by acquiring the dataset from the provided Kaggle link. We'll describe the data source and perform initial data integrity checks to ensure data quality. Data cleansing is applied to eliminate any inconsistencies or outliers that may affect the reliability of our model.

```
true_data = pd.read_csv("True.csv")
fake_data = pd.read_csv("Fake.csv")


true_data['label'] = 1
fake_data['label'] = 0
```

Data Cleaning

Data cleaning is an essential step in preparing our dataset. We validate the data, select relevant attributes, handle missing data, and address outliers. This ensures that our dataset is focused on the most informative features for our fake news detection model.

Text Preprocessing

To transform the raw text data into a format suitable for machine learning, we perform text preprocessing tasks. These include converting text to lowercase, tokenizing sentences into words, removing stopwords, special characters, and applying lemmatization for word normalization.

```
def preprocess_text(text):
    stop_words = set(stopwords.words('english'))
    tokens = word_tokenize(text.lower())
    tokens = [word for word in tokens if word.isalpha()]
    tokens = [word for word in tokens if word not in stop_words]
    return ' '.join(tokens)
```

Model Training and Evaluation ->

Data Splitting

The dataset is divided into training and testing sets, or cross-validation is applied to evaluate the model's generalization performance. Data partitioning ensures that we have independent data subsets for training and evaluation.

```
data['text'] = data['text'].apply(preprocess_text)
```

Model Selection

Choosing an appropriate model is crucial. In the code implementation, we have opted for the Multinomial Naive Bayes (NB) algorithm. Naive Bayes is a classification algorithm based on Bayes' theorem, and the Multinomial Naive Bayes variant is specifically designed for text classification tasks. While we considered various models, including Logistic Regression, Random Forest, and deep learning models like LSTM or BERT, the final choice of using Multinomial Naive Bayes was influenced by its simplicity and effectiveness in text classification tasks.

Model Training

Model training involves configuring the chosen model, including hyperparameter tuning, learning rate settings, and the number of training epochs. The model learns to distinguish between real and fake news using the preprocessed text data and features extracted in earlier steps.

Model Evaluation

The effectiveness of our model is assessed using various performance metrics such as accuracy, precision, recall, F1-score, ROC-AUC, and the construction of a confusion matrix. Cross-validation results provide additional insights into our model's performance on unseen data.

```
tfidf_vectorizer = TfidfVectorizer(max_features=5000)
X = data['text']
y = data['label']
X_tfidf = tfidf_vectorizer.fit_transform(X)


classifier = MultinomialNB()
classifier.fit(X_tfidf, y)
```

Hyperparameter Tuning

To optimize our model's performance, we fine-tune hyperparameters through methods like grid search and hyperparameter optimization. This step allows us to identify the best parameter settings for our selected model.

Model Saving and Deployment

Once our model meets our expectations, we save it for future use. We also consider the deployment process, making our fake news detection model available for practical applications.

Main Function code

```
while True:

    news_text = input("Enter a news article (or 'exit' to quit): ")

    if news_text.lower() == 'exit':
        break


    preprocessed_text = preprocess_text(news_text)


    news_tfidf = tfidf_vectorizer.transform([preprocessed_text])

    label = classifier.predict(news_tfidf)

    if label == 1:
        print("This news is likely TRUE.")
    else:
        print("This news is likely FALSE.")
```

OUTPUT

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
Enter a news article (or 'exit' to quit): NASA's Perseverance rover touched down on the surface of Mars, marking a
This news is likely FALSE.
Enter a news article (or 'exit' to quit): Aliens Land on Earth, White House in Panic
This news is likely FALSE.
Enter a news article (or 'exit' to quit): NASA's Perseverance Rover Successfully Lands on Mars
This news is likely TRUE.
Enter a news article (or 'exit' to quit): Stock Market Hits All-Time High, Investors Rejoice
This news is likely TRUE.
Enter a news article (or 'exit' to quit): World Health Organization Declares Global Pandemic Over"
This news is likely TRUE.
Enter a news article (or 'exit' to quit): Giant Lizard Attacks City, Chaos Ensues
This news is likely FALSE.
Enter a news article (or 'exit' to quit): World Leaders Gather for Climate Change Summit
This news is likely TRUE.
Enter a news article (or 'exit' to quit): Ancient Egyptian Pyramid Discovered in Antarctica
This news is likely TRUE.
Enter a news article (or 'exit' to quit): World Leaders Gather for Climate Change Summit
This news is likely TRUE.
Enter a news article (or 'exit' to quit): exit
```