Report

Notebook Flow Description:

The notebook begins with importing necessary libraries and defining preprocessing functions. Then, it loads the dataset and performs basic exploratory data analysis. The preprocessing steps include converting text to lowercase, removing special characters and digits, tokenization, removing stop words, and stemming. The order of preprocessing was chosen to ensure that the text is cleaned before being tokenized, and stemming is applied after tokenization to maintain word integrity.

Following preprocessing, the notebook proceeds with feature extraction using TF-IDF vectorization and Bag of Words representation. Then, it applies classification models such as Logistic Regression and K-Nearest Neighbors (KNN) to both TF-IDF and Bag of Words features. After that, Word2Vec and Doc2Vec embedding techniques are implemented, followed by training Logistic Regression and KNN models using these embeddings.

Data Preprocessing & Features Extraction:

The chosen data preprocessing techniques include converting text to lowercase, removing special characters and digits, tokenization, removing stopwords, and stemming. These techniques were selected to clean the text data and reduce noise, thereby improving the effectiveness of the classification models. TF-IDF vectorization and Bag of Words representation were chosen for feature extraction to capture the importance of words in the text documents.

Also, text embedding techniques were used like word2vec and doc2vec which captures semantic information from text data and are commonly used in document classification, sentiment analysis, and text generation.

Data Splitting:

The data is split into training and testing sets with a ratio of 80:20. This ratio was chosen to ensure an adequate amount of data for training while still reserving a sufficient portion for testing. Additionally, a random state of 42 was used for reproducibility.

Model Training:

Logistic Regression and KNN classifiers were chosen for training the data. Logistic Regression is suitable for binary classification tasks and is efficient for text data. KNN was selected as a baseline classifier for its simplicity and ability to capture similarities between data points. Word2Vec and Doc2Vec embeddings were utilized to capture semantic information in the text, which can be beneficial for classification tasks.

Model Evaluation:

The models are evaluated using accuracy and F1 score metrics. Accuracy measures the overall correctness of predictions, while F1 score considers both precision and recall, making it suitable for imbalanced datasets like the provided text data. These metrics provide a comprehensive assessment of model performance.

Dominant Models:

Model Accuracy

Text Embedding	ML Model	Accuracy	F1 Score
Tfidf	Logistic Regression	0.9870	0.9799
Tfidf	KNN	0.9879	0.9815
Bag of Words	Logistic Regression	**0.9965**	**0.9947**
Bag of Words	KNN	0.9870	0.9802
Word2Vec	Logistic Regression	0.9905	0.9853
Word2Vec	KNN	0.9887	0.9828
Doc2Vec	Logistic Regression	**0.9948**	**0.9920**
Doc2Vec	KNN	0.9879	0.9812

The two dominant models, highlighted in bold, are Bag of Words Logistic Regression and Doc2Vec Logistic Regression. These models achieved the highest accuracy scores across different feature extraction techniques and classifiers. They were chosen based on their superior performance in accurately classifying the text data.