

Natural Language Processing “Assignment-2”

Ahmed Mohamed Sallam ID: 20210614

Roaa Gamal Youssef ID: 20210571

*Flow Description*

1. Data Cleansing: Initial cleansing to remove unnecessary or distracting elements from the data, such as HTML tags, email addresses, and non-alphabetic characters.

2. Tokenization: Breaking down text into individual words or tokens, which is essential for further processing.

3. Stemming: These were applied after tokenization because both processes require the text to be broken into tokens first. Stemming reduces words to their root form, which helps in generalizing different forms of the same word.

4. Stop-word Removal: Removal of common words that add little value in distinguishing between different classes of text.

5. Data Splitting: The data is split into training and testing sets to evaluate the performance of the models on unseen data

6. Text Embedding: Converting text into numerical form through two techniques to prepare for machine learning modeling:

- First Techniques: Not using Neural Network (Count Vectorization and TF-IDF).

- Second Techniques: using Neural Network (Word2Vec, Doc2Vec), and we use pre-trained language model BERT

7. Model Training - Using different algorithms to train on the prepared features.

8. Model Evaluation: Assessing the performance of each model on the test data using metrics like accuracy, precision, recall, and F1-score.

*Data Preprocessing & Features Extraction*

Each preprocessing and feature extraction technique was chosen based on its effectiveness in preparing textual data for spam classification:

- Regular expressions were used to clean the text thoroughly.

- Tokenization allows the breakdown of texts into tokens which is necessary for most NLP tasks.

- Stemming helps in reducing the complexity of the model by cutting down the variations of words.

- Text Embedding Techniques were employed to transform text data into a format suitable for model training by highlighting the importance of each term.

*Data Splitting*

The dataset was split into a training set (60%) and a test set (40%). This ratio provides enough data for training the models while still retaining a substantial portion for an unbiased evaluation of the model's performance on unseen data.

*Model Training*

Classifiers like Logistic Regression and Random Forest were chosen due to their robustness and effectiveness in binary classification tasks.

*Model Evaluation*

Metrics such as accuracy, precision, recall, and F1-score were chosen because they provide a comprehensive assessment of the model's performance, especially in the context of a binary classification problem like spam detection.

*Dominant Models*

The models that performed best were highlighted based on their F1-scores, as these metric balances both precision and recall, making it suitable for scenarios where both false positives and false negatives carry significant costs.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| Logistic Regression (CountVectorizer) | 0.985339 | 0.984658 | 0.968450 | 0.976487 |
| Random Forest (CountVectorizer) | 0.851660 | 0.994859 | 0.530864 | 0.692308 |
| Logistic Regression (TfidfVectorizer) | 0.968521 | 0.992492 | 0.906722 | 0.947670 |
| Random Forest (TfidfVectorizer) | 0.948254 | 0.988764 | 0.844993 | 0.911243 |
| Logistic Regression (Word2Vec) | 0.977145 | 0.992711 | 0.934156 | 0.962544 |
| **Random Forest (Word2Vec)** | 0.981026 | 0.987198 | 0.951989 | **0.969274** |
| Logistic Regression (Doc2Vec) | 0.942216 | 0.977528 | 0.835391 | 0.900888 |
| Random Forest (Doc2Vec) | 0.912893 | 0.987061 | 0.732510 | 0.840945 |
| **Logistic Regression (BERT)** | 0.982751 | 0.980474 | 0.964335 | **0.972337** |
| Random Forest (BERT) | 0.960759 | 0.978979 | 0.894376 | 0.934767 |

*Discussion:*

The BERT-based Logistic Regression model achieved the highest F1-score, indicating superior performance in balancing precision and recall, essential for effective spam detection. The Random Forest with Word2Vec also showed excellent results, demonstrating the efficacy of Word2Vec in capturing semantic information in texts.