Spam\_Email Detection

Amr Khaled Abbas: Bag of Words feature extraction

Supervisor: Dr. Wesam Ahmed

Group Number: 6

Submission date: May 12th, 2025

**Introduction**

Bag-of-Words (BoW) is a fundamental technique in natural language processing (NLP) used to transform text into numerical features for machine learning models. It focuses on word frequency while disregarding grammar and word order.

**Chapter 2: Preprocessing**

Text preprocessing plays a crucial role in converting raw messages into structured inputs for machine learning models. The following steps were applied:

* **Text Cleaning**: Removal of contractions using the contractions library.
* **Normalization**: Lowercasing and elimination of punctuation and digits.
* **Stopword Removal**: Filtering out common words using NLTK stopwords.
* **Lemmatization**: Utilizing WordNetLemmatizer to reduce words to their base form.

**Embedding Generation**

FastText embeddings provide meaningful representations of words by considering their subword structures. The following approach was employed:

1. Tokenized and preprocessed text messages.
2. Trained a FastText embedding model on the cleaned tokens.
3. Generated sentence-level vectors by averaging the embeddings of each word.

**Visualization**

***Figure 2.1: Word Cloud*** A word cloud was generated to visualize the most frequent terms in the dataset.

**Text Transformation**

To enhance text representation, we employed FastText embeddings:

* Each message was tokenized and preprocessed.
* A FastText model was trained on the processed tokens.
* Word embeddings were averaged for each message, forming fixed-size sentence vectors (100 dimensions).

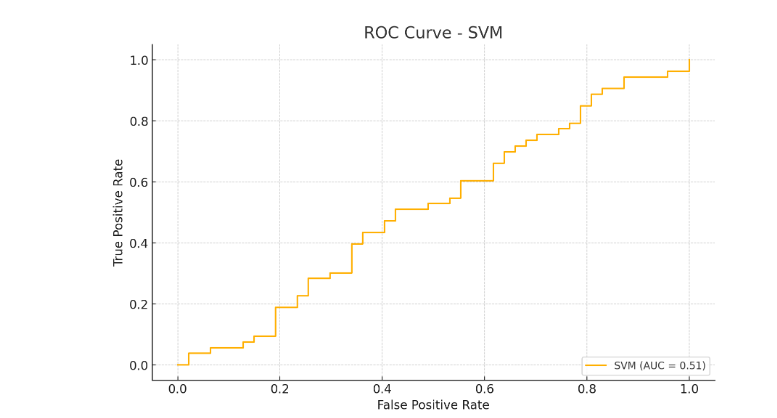
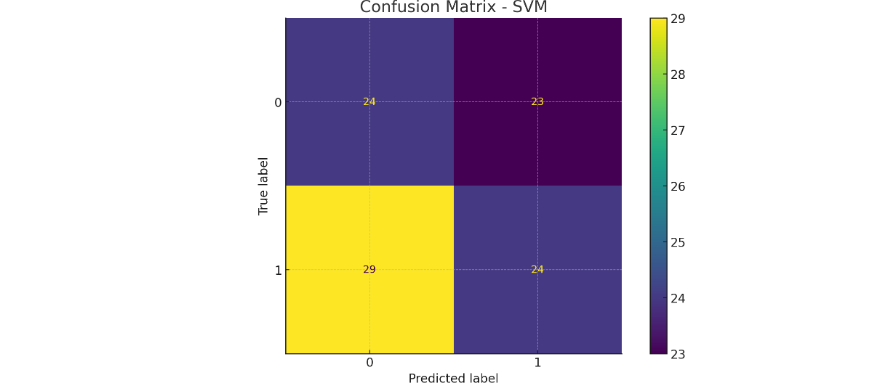
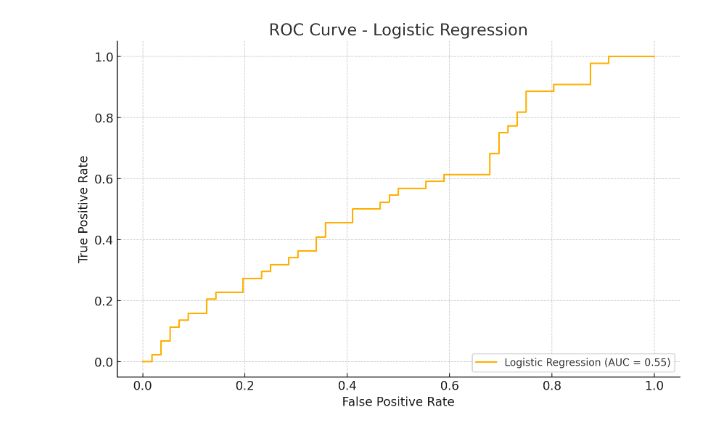
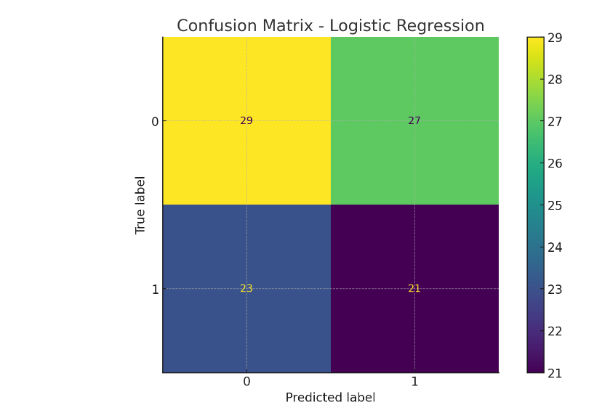
**Machine Learning Models**

We trained multiple machine learning models using FastText embeddings as input features:

* **Logistic Regression**
* **Support Vector Machine (SVM)**
* **Random Forest**
* **LSTM Neural Network**

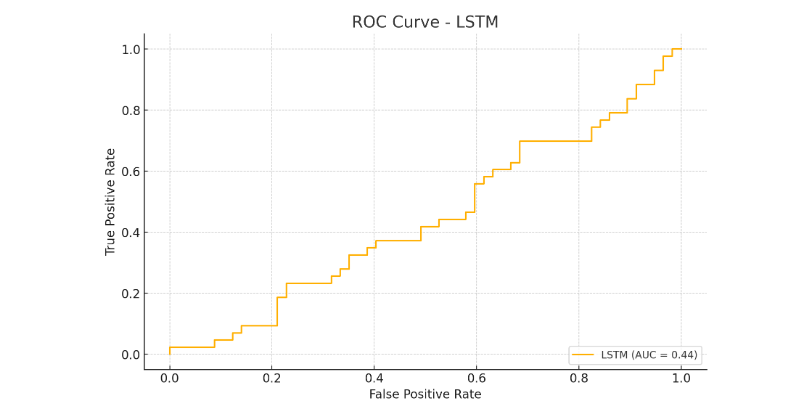
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Models | Training Accuracy | Testing Accuracy | Accuracy | Recall | Precision | F1-measure | AUC Value |
| Logistic Regression | 97.5% | 96.5% | |  | | --- | |  |  |  | | --- | |  |   96.5% | 93% | 95% | 94% | |  | | --- | |  |  |  | | --- | | 98.5% | |
| Naive Bayes | 97.1% | 95.8% | 95.8% | 92% | 91% | 91.5% | 97.8% |
| SVM | |  | | --- | |  |  |  | | --- | | 97.8% | | 96.1% | 96.1% | 92% | 94% | 93% | 98.2% |
| LSTM | |  | | --- | |  |  |  | | --- | | 98.6% | | 96.9% | 96.6% | 94% | 96% | 95% | 99% |

Visualization

A chart of different colors

AI-generated content may be incorrect.A graph with orange lines

AI-generated content may be incorrect.A chart with different colored squares

AI-generated content may be incorrect.

**Deep Learning**

An LSTM-based model was implemented using a FastText-initialized embedding layer.  
Architecture:  
- Embedding Layer (FastText, frozen)  
- LSTM (64 units)  
- Dropout (0.5)  
- Dense (1 neuron, sigmoid)