SPAM CLASSIFIER

Data Collection: The code assumes that you have a dataset located at "./src/data/sentiments.csv" with two columns: "sentiment" (containing positive or negative sentiment labels) and "text" (containing the textual content).

Data Preprocessing: The code preprocesses the text data by removing punctuation, converting text to lowercase, and tokenizing the text into individual words. It also employs the NLTK library to remove stopwords and performs lemmatization to reduce words to their base forms.

Feature Extraction: The code employs Word2Vec embeddings to convert the tokenized words into numerical features. It utilizes a pre-trained Word2Vec model to generate word vectors for each word in the dataset.

Model Selection: The selected machine learning algorithm is a Support Vector Machine (SVM), trained on the Word2Vec-transformed data.

Evaluation: The code evaluates the model's performance using metrics like accuracy, precision, recall, and F1-score on a test dataset of sentiment-labeled texts.

Iterative Improvement: When a new text is predicted, the code updates the dataset with the prediction result ("positive" or "negative") and then retrains the SVM model to incorporate this new data. This allows the model to adapt and potentially improve its performance over time as it learns from new examples.