

PRML ASSIGNMENT 3

Senkula Surya Gopala Chandra cs21b073

May 2024

Spam Classification Using Support Vector Machine

We are asked to classify the emails into spam or non spam class using any of the classification technique taught in class. I choose SVM as my classification technique.

First I got my dataset from this [repository](#).

I then split my data into 80% train dataset and 20% test dataset to train and test my SVM model.

In my "create_datafile.py" file I made few folders namely train, train_label, test1, test1_label to separate each train data, test data and their corresponding labels. I used TF-IDF vectorization technique to convert my data into numerical features. TF-IDF assigns weights to terms based on their frequency in the document and their inverse frequency across all documents in the corpus. This process transformed the text data into a matrix of TF-IDF features.

I trained an SVM classifier using a linear kernel for spam detection. The linear kernel is well-suited for high-dimensional sparse data, making it a suitable choice for text classification tasks. We fit the SVM model to the training data, learning the optimal decision boundary to distinguish between spam and non-spam emails.

Now I tested the data from test1 folder to check the accuracy of the SVM classifier. Accuracy measures the proportion of correctly classified emails out of the total number of emails. Additionally, I calculated the error rate, which represents the proportion of misclassified emails.

The SVM classifier achieved a high accuracy of 99.82% on the test set, indicating its effectiveness in detecting spam emails. The low error rate of 0.18% further underscores the robustness of the model. These results demonstrate the efficacy of machine learning techniques, specifically SVMs, in spam detection tasks.

The successful implementation of the SVM-based spam detection model showcases the potential of machine learning in combating spam emails. The high accuracy and low error rate validate the effectiveness of the TF-IDF feature extraction and linear kernel SVM classifier. However, it's essential to acknowledge the limitations of the model, such as the reliance on textual features and the

need for continuous updates to adapt to evolving spamming techniques.