Machine Learning Approaches for Text Classification: A Study on Spam Detection

Mingbao Feng

Instructor: Lili Mou

December 12, 2023

Introduction:

Within the realm of machine learning, this project undertakes the formidable challenge of categorizing text into 'spam' or 'ham (not spam).' The venture revolves around establishing a robust training-validation-test framework, accompanied by a methodical approach to hyperparameter tuning.

In our daily interactions with textings, the significance of spam detectors becomes apparent. The prevalence of spam represents an undesirable facet of user experience, necessitating the deployment of effective classifiers. Harnessing the capabilities of machine learning, this project delves into the training and fine-tuning of diverse models tailored specifically for spam detection.

In the context of this mini project within CMPUT 466, our exploration spans a spectrum of varied machine learning algorithms, satisfying the stipulated minimum of three. Since this is a Classification problem, some of the models like Linear regression and Gaussian mixture models are not the best choice for this problem. The selected algorithms, encompassing Logistic Regression, Single-layer Neural Network and Multilayer Neural Network, not only demonstrate suitability for the task at hand but also showcase distinct methodologies. Alongside the comprehensive comparison of these advanced models, we will establish trivial baselines, recognizing the importance of simplicity in our evaluation metrics.

Problem formulation:

The input data for our models consists of text messages, and the corresponding output is binary—'1' if the message is classified as spam and '0' if it is not spam. We sourced our dataset from: https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset/

The dataset aggregates messages from diverse and openly available sources on the internet, totaling approximately 5574 messages. The data compilation draws from various repositories, including Grumbletext, NUS SMS Corpus, Caroline Tag’s PhD thesis, and SMS Spam Corpus v.0.1.

Approaches and baselines:

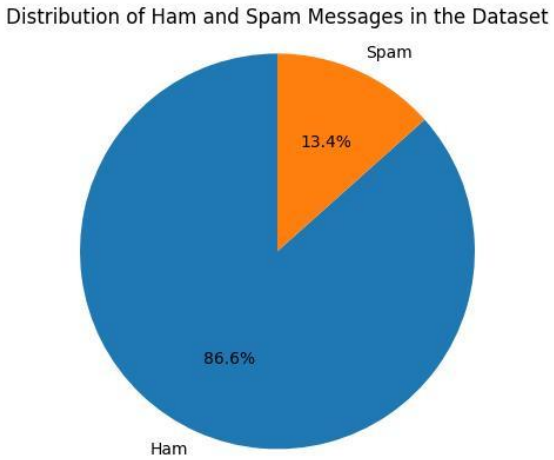
The primary approach for text classification in this project involves feature extraction using the Term Frequency-Inverse Document Frequency (TF-IDF) measure. TF-IDF is a statistical metric that gauges the significance of a word within a document collection. It dynamically increases with the frequency of a word in a specific document while being normalized by the inverse document frequency. This normalization helps mitigate the impact of ubiquitous words like "it" and "the" that appear frequently across documents.

For all 3 machine learning models, the scikit-learn library's default configurations were utilized as the baseline. A systematic exploration of hyperparameter space was conducted through a grid search, aiming to identify the optimal combination of parameters for the models. This grid search was complemented with cross-validation, ensuring a robust evaluation of the models' performance. Further details of the hyperparameters for each approach will be mentioned shortly in the results section.

Furthermore, to implement the training-validation-test infrastructure, we divided the dataset into training and testing sets, with 80% allocated for training and 20% for testing. This division ensures a comprehensive evaluation of the model's generalization performance on unseen data. The execution time for the entire process, including hyperparameter tuning, training, and evaluation, was recorded, providing insights into the efficiency of the model training with the selected configurations. The reported accuracy and best hyperparameters serve as key metrics to assess the model's performance and configuration optimization.

Evaluation metric:

In light of the dataset's inclination towards more ham messages than spam messages(shown below), a trivial classifier that assigns all messages as ham would yield an accuracy of roughly 86.59%. This baseline accuracy acts as a foundational metric, embodying a direct approach to the task. The goal is to surpass this baseline, striving for an accuracy exceeding 87%, aligning with the majority class in the dataset. This reference accuracy of 86.59% becomes a pivotal point of comparison during the evaluation. Should the average accuracy from cross-validation exceed this threshold, the classifier proves valuable for the spam detection task. This approach accommodates the dataset's class distribution imbalance, offering a robust gauge of success for the selected classification model.



Results:

Logistic Regression:

We established a parameter grid with the following parameters:

- max_iter: [100, 500, 1000]
- C: [0.01, 0.1, 1, 10].

The logistic regression model, with best Hyperparameters 'C' (the inverse of regularization strength) being 10 and 'max_iter' being 100, achieved an impressive accuracy of **97.49%** on the test set, demonstrating its efficacy in classifying spam and non-spam messages. The time taken for the entire process, including hyperparameter tuning with 5-fold cross-validation by default and evaluation, was **67.85 seconds.** This noteworthy accuracy surpasses the baseline accuracy of approximately 86.6%, showcasing the model's robustness in overcoming the majority class in the

dataset. The concise execution time further emphasizes the efficiency of the logistic regression approach in this spam detection task.

Single-layer Neural Network:

We established a parameter grid with the following parameters:

- alpha: [0.0001, 0.001, 0.01, 0.1, 1, 10, 100]
- max_iter: [100, 500, 1000]
- penalty: ['l2', 'l1']

The Perceptron, a single-layer neural network, features only one hidden layer between the input and output layers. The model exhibited an impressive accuracy of **96.23%** on the test set, achieved within a time-efficient span of **27.40 seconds.** The best-performing hyperparameters identified in the search were {'alpha': 0.0001, 'max_iter': 100, 'penalty': 'l2'}. This result outperforms the trivial classifier, underscoring the model's proficiency in discerning spam from non-spam messages. The time taken to run the model, around 27.4 seconds with 5-fold validation, further establishes the perceptron as a swift and accurate solution for spam detection compared to Logistic Regression.

Multilayer Neural Network:

We established a parameter grid with the following parameters:

- max_iter: [100, 500, 1000]
- hidden_layer_sizes: [(100,), (50, 50), (30, 30, 30)]
- activation: ['logistic', 'relu', 'tanh']
- learning_rate: ['constant', 'invscaling', 'adaptive']

In a comprehensive exploration of hyperparameter tuning for the Multi-Layer Perceptron (MLP) classifier, the process took approximately **59 minutes (3518.80 seconds)** to complete. The model's performance on the test set yielded an impressive accuracy of **98.03%.** The identified best hyperparameters through the grid search included an activation function of ReLu, a hidden layer architecture comprising 30 neurons in each of the 3 layers, a learning rate strategy set to 'invscaling,' and a maximum number of iterations capped at 100. This configuration showcases the robustness of the MLP model, particularly in capturing intricate patterns within the data. The prolonged execution time, while notable, emphasizes the exhaustive search conducted to fine-tune the model's hyperparameters, ultimately resulting in a highly accurate and optimized classifier for the given dataset.

Conclusion:

In our investigation of spam detection models, each approach exhibited its unique strengths and limitations. Logistic Regression, with optimal hyperparameters 'C' and 'max_iter' set to 10 and 100, demonstrated a commendable accuracy of 97.49%. Its efficient execution time of 67.85 seconds showcased its practicality, making it a reliable choice for real-time applications. However, the model might face challenges with highly complex patterns due to its linear nature.

The Single-layer Neural Network, or Perceptron, achieved an accuracy of 96.23%, showcasing its proficiency in discerning spam from non-spam messages. With hyperparameters {'alpha': 0.0001, 'max_iter': 100, 'penalty': 'l2'}, the Perceptron outperformed the trivial classifier and displayed remarkable efficiency with a runtime of 27.4 seconds. Its simplicity and quick training make it an attractive choice, especially for scenarios where interpretability and speed are crucial. Nevertheless, its single-layer structure may limit its ability to capture intricate patterns present in complex datasets.

The Multilayer Neural Network, specifically the MLP classifier, demonstrated the highest accuracy of 98.03% with optimal hyperparameters {'activation': 'relu', 'hidden_layer_sizes': (30, 30, 30), 'learning_rate': 'invscaling', 'max_iter': 100}. This model showcased remarkable capability in capturing intricate patterns within the data, leading to superior accuracy. However, the exhaustive hyperparameter search resulted in a significantly longer runtime of 3518.80 seconds. The MLP classifier's complexity and longer training time make it suitable for scenarios where accuracy is paramount, and computational resources are not a limiting factor.

In terms of overall performance, the Multilayer Neural Network (MLP classifier) stands out with the highest accuracy. However, the choice of the best model depends on the specific requirements of the application. Logistic Regression and the Single-layer Neural Network offer appealing alternatives based on their efficiency, interpretability, and suitability for real-time applications. The decision should consider the trade-offs between accuracy, interpretability, and ensuring alignment with the specific needs of the spam detection task at hand.