Experimental Comparison of Learning Algorithms for Spam Detection

1st Kidus Mikael Birhanu

2nd Bokhtiar Mehedy

I. Introduction

The objective of this study is to compare the performance of three supervised classification algorithms—Logistic Regression, Random Forest, and XGBoost—on a given dataset. The evaluation focuses on three key metrics: training time, accuracy, and F1 score. To ensure reliable and unbiased results, stratified 10-fold cross-validation was used, preserving the class distribution across folds during the training and testing process.

To analyze the results, the Friedman test was conducted to identify whether significant differences exist among the models. Based on these results, the Nemenyi test was applied to determine which models performed significantly differently. This report also incorporates tables formatted according to established guidelines from [2] and [3] to support the statistical analysis.

The goal of this study is to determine the most suitable model for this dataset by balancing predictive performance and computational efficiency. This comparison provides valuable insights into the selection of models for similar classification tasks.

II. Experimental Setup

A. Dataset

The dataset used in this study was sourced from the UCI repository [1] and contains 4,601 instances with 58 features and two classes: spam and non-spam. The class distribution is 60.6% spam and 39.4% non-spam.

B. Algorithms

The algorithms evaluated in this study include:

- **Random Forest**: A robust ensemble learning method combining decision trees through majority voting or averaging.
- **XGBoost**: An efficient gradient boosting algorithm known for high performance on structured data.
- **Logistic Regression**: A simple yet effective linear model for binary classification problems.

C. Evaluation Metrics

The evaluation metrics used are:

- **Accuracy**: Measures the proportion of correctly classified samples.
- **F1 Score**: A harmonic mean of precision and recall, suitable for imbalanced datasets.
- **Training Time**: The time taken to train the model.

D. Cross-Validation Strategy

We used stratified 10-fold cross-validation to ensure each fold maintains the same proportion of classes as the original dataset. This reduces variance and provides reliable performance estimates.

E. Statistical Tests

To evaluate the significance of differences among the models, we performed:

- **Friedman Test**: To identify whether significant differences exist.
- **Nemenyi Post-Hoc Test**: To determine pairwise significance differences when the Friedman test detects significance.

F. Tools and Environment

The experiments were implemented in Python using libraries such as:

- `scikit-learn` for model training and evaluation.
- `XGBoost` for gradient boosting models.
- `numpy` and `pandas` for data manipulation.

III. Results and Analysis

The following tables shows the comparison results for the XGBoost, Random Forest (RF), and Logistic Regression (LR) models after performing stratified 10-fold cross-validation. The results are based on the following metrics: time, accuracy, and F1 score for each fold. Additionally, the table includes the average and standard deviation for each metrics. following the format of Table 12.4 in the book [2]

Fold	Training Time (s)	Accuracy	F1 Score
1	1.476165	0.915401	0.892562
2	1.459002	0.923913	0.901961
3	1.156858	0.932609	0.913165
4	1.335482	0.936957	0.918768
5	1.211622	0.913043	0.890110
6	1.087201	0.936957	0.920110
7	1.698693	0.934783	0.915730
8	1.284989	0.932609	0.912676
9	1.353487	0.936957	0.915452
10	0.930187	0.915217	0.890141
Mean	1.299368	0.927844	0.907067
Std	0.218571	0.009952	0.012163

TABLE I
Results for Logistic Regression

Fold	Training Time (s)	Accuracy	F1 Score
1	0.811631	0.952278	0.938889
2	0.752213	0.956522	0.944134
3	0.879899	0.960870	0.950000
4	0.858138	0.967391	0.957983
5	0.844638	0.945652	0.931129
6	0.838471	0.956522	0.944134
7	0.871480	0.958696	0.947368
8	0.820342	0.956522	0.943820
9	0.765990	0.941304	0.923513
10	0.847109	0.952174	0.937500
Mean	0.828991	0.954793	0.941847
Std	0.042360	0.007443	0.009720

TABLE II
RESULTS FOR RANDOM FOREST

Fold	Training Time (s)	Accuracy	F1 Score
1	0.367074	0.945770	0.931507
2	0.374614	0.958696	0.947075
3	0.314988	0.958696	0.948229
4	0.325059	0.960870	0.950000
5	0.340983	0.954348	0.942779
6	0.309312	0.956522	0.944444
7	0.345881	0.950000	0.936288
8	0.323987	0.958696	0.947945
9	0.349312	0.958696	0.947075
10	0.318582	0.945652	0.929178
Mean	0.336979	0.954794	0.942452
Std	0.022281	0.005666	0.007463

TABLE III
RESULTS FOR XGBOOST

The following tables are created for the purpose of performing the Friedman test, following the format of Table 12.8 in the book [2]

Fold	Logistic Regression	Random Forest	XGBoost
1	0.8926 (3)	0.9389 (1)	0.9315 (2)
2	0.9020 (3)	0.9441 (2)	0.9471 (1)
3	0.9132 (3)	0.9500 (1)	0.9482 (2)
4	0.9188 (3)	0.9580 (1)	0.9500 (2)
5	0.8901 (3)	0.9311 (2)	0.9428 (1)
6	0.9201 (3)	0.9441 (2)	0.9444 (1)
7	0.9157 (3)	0.9474 (1)	0.9363 (2)
8	0.9127 (3)	0.9438 (2)	0.9479 (1)
9	0.9155 (3)	0.9235 (2)	0.9471 (1)
10	0.8901 (3)	0.9375 (1)	0.9292 (2)
Avg Rank	3.00	1.50	1.50

TABLE IV
F1 SCORE COMPARISON ACROSS MODELS

Fold	Logistic Regression	Random Forest	XGBoost
1	0.9176 (3)	0.9523 (1)	0.9458 (2)
2	0.9239 (3)	0.9565 (2)	0.9587 (1)
3	0.9326 (3)	0.9609 (1)	0.9587 (2)
4	0.9370 (3)	0.9674 (1)	0.9609 (2)
5	0.9130 (3)	0.9457 (2)	0.9543 (1)
6	0.9370 (3)	0.9565 (1)	0.9565 (2)
7	0.9348 (3)	0.9587 (1)	0.9500 (2)
8	0.9304 (3)	0.9565 (2)	0.9587 (1)
9	0.9370 (3)	0.9413 (2)	0.9587 (1)
10	0.9152 (3)	0.9522 (1)	0.9457 (2)
Avg Rank	3.00	1.40	1.60

TABLE V
ACCURACY COMPARISON ACROSS MODELS

Fold	Logistic Regression	Random Forest	XGBoost
1	1.0106 (3)	0.4851 (2)	0.1149 (1)
2	1.0373 (3)	0.4832 (2)	0.1173 (1)
3	1.1719 (3)	0.4888 (2)	0.1220 (1)
4	0.9989 (3)	0.4821 (2)	0.1161 (1)
5	0.9174 (3)	0.5379 (2)	0.1287 (1)
6	0.8388 (3)	0.4634 (2)	0.1164 (1)
7	1.1391 (3)	0.4666 (2)	0.1136 (1)
8	0.9315 (3)	0.4656 (2)	0.1136 (1)
9	0.7834 (3)	0.4780 (2)	0.1228 (1)
10	1.1074 (3)	0.4757 (2)	0.1142 (1)
Avg Rank	3.00	2.00	1.00

TABLE VI
TRAINING TIME COMPARISON ACROSS MODELS

From the above 3 tables, we conducted the Friedman test by calculating **the average ranks**, **the sum of squared differences spread between the ranks** and **the sum of squared differences spread over all ranks**.Using these values, we calculated the Friedman statistic, which is the ratio of the sum of squared differences between the ranks to the total sum of squared differences across all ranks.

- For the F1 score, we obtained a Friedman statistic of 15.
- For accuracy, we obtained a Friedman statistic of 15.2.
- For training time, we obtained a Friedman statistic of 20.

The critical value for k=3 and n=10 at the $\alpha = 0.05$ level is 6.20 (obtained from a table in [3]). We compared the critical value with the Friedman statistics for each metric and rejected the null hypothesis. This indicates that at least one model is performing significantly differently for the given dataset but does not specify which model. Therefore, we conducted the Nemenyi test to identify which model is performing differently from the others.

To conduct the Nemenyi test, we compared the absolute differences in the average ranks between each model with the critical difference value. To calculate the critical difference, we obtained a value of 2.343 for $\alpha = 0.05$, k=3, and degrees of freedom approaching infinity (obtained from a table in [4]). Using this, we calculated a critical difference of 1.05.

the following tables show pair wise comparison on each model with the critical value

Comparison	Rank Difference	Significant($CD > 1.05$)
LR Vs RF	1.50	True
RF Vs XGBoost	0.00	False
LR Vs XGBoost	1.50	True

TABLE VII
SIGNIFICANCE PAIRWISE COMPARISONS FOR F1 SCORE

Comparison	Rank Difference	Significant($CD > 1.05$)
LR Vs RF	1.60	True
RF Vs XGBoost	0.20	False
LR Vs XGBoost	1.40	True

TABLE VIII
SIGNIFICANCE PAIRWISE COMPARISONS FOR ACCURACY

Comparison	Rank Difference	Significant($CD > 1.05$)
LR Vs RF	1.00	True
RF Vs XGBoost	1.00	False
LR Vs XGBoost	2.00	True

TABLE IX
SIGNIFICANCE PAIRWISE COMPARISONS FOR TRAINING TIME

From the above tables, we concluded that for both F1 score and accuracy, Logistic Regression is the model performing significantly differently from the others. For training time, we concluded that Logistic Regression and XGBoost are performing significantly differently from each other.

IV. DISCUSSION

The analysis highlights key differences in the performance of Logistic Regression, Random Forest, and XGBoost. The Friedman and Nemenyi tests revealed that Logistic Regression performs significantly differently from Random Forest and XGBoost in terms of F1 score and accuracy, suggesting it is less suited for the dataset. However, Logistic Regression's training time is significantly faster compared to XGBoost, making it preferable when computational efficiency is critical.

In contrast, Random Forest and XGBoost demonstrated superior predictive performance, making them more suitable for tasks where accuracy and F1 score are priorities, despite their higher computational costs. These findings emphasize the importance of selecting models based on task-specific requirements, balancing predictive performance and computational efficiency.

V. CONCLUSION

In this study, we evaluated Logistic Regression, Random Forest, and XGBoost on the given dataset using stratified 10-fold cross-validation and analyzed their performance based on F1 score, accuracy, and training time. The results showed that Logistic Regression performed significantly worse in F1 score and accuracy compared to Random Forest and XGBoost, making it less suitable for this dataset despite its faster training time.

Between Random Forest and XGBoost, both demonstrated strong predictive performance; however, XGBoost consistently achieved higher accuracy and F1 scores, making it the best model for this dataset. While XGBoost requires more computational resources, its superior performance makes it the optimal choice for tasks where accuracy and F1 score are critical for decision-making.

REFERENCES

[1] UCI Machine Learning Repository, "Spambase Dataset," [Online]. Available: https://archive.ics.uci.edu/dataset/94/spambase. [Accessed: Dec. 14, 2024].

[2] P. Flach, *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*. Cambridge University Press, 2012.

[3] JSTOR, "Article Metadata," [Online]. Available: https://www.jstor.org/stable/3315656#metadata_info_tab_contents. [Accessed: Dec. 14, 2024].

[4] Real Statistics, "Studentized Range Q Table," [Online]. Available: https://real-statistics.com/statistics-tables/studentized-range-q-table/. [Accessed: Dec. 14, 2024].