Spams vs Hams Emails Classifications

By Burhan Sultan Basit Student ID 650387980

1. Introduction

The goal of this project was to build a simple machine learning model that can detect whether an email is spam or not. This type of project is common because spam messages are a big problem in email systems. The work was done step by step, from cleaning the data, training the models, testing them, and finally creating a small app to make predictions.

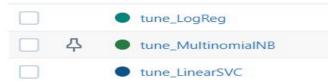
2. Data Preparation

In this step, I worked with two different CSV files that contained email data. I checked for missing or repeated lines, and then joined them into one complete dataset. After combining them, I made sure the columns were clear, one called *email* and the other *label* (where 0 means ham and 1 means spam). I saved this final clean version as CLEAN_EMAILS.csv, which was later used for training the models.

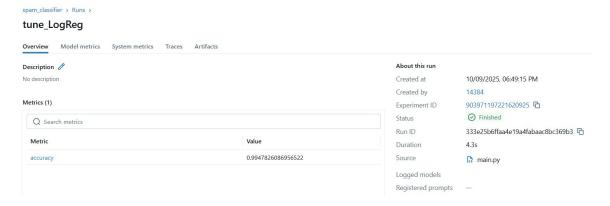
and the MANDERS aug NAMBER NAM

3. Modeling and MLflow

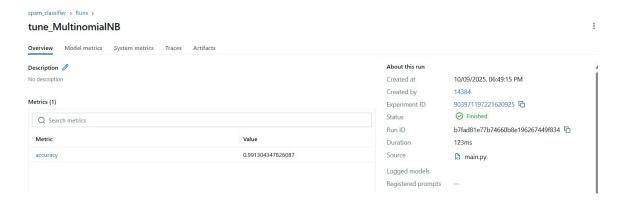
After preparing the clean data, I changed all the email text into numbers using a TF-IDF vectorizer so that the models could understand it. Then I trained three different models: Naive Bayes, Logistic Regression, and Linear SVM. Each model was tested and compared based on accuracy and F1 score. To keep everything organized, I used MLflow to track the results. It allowed me to see which model performed best and how much time each one took to train. Overall, the training went smoothly, and MLflow helped me visualize and compare all three runs in one place.



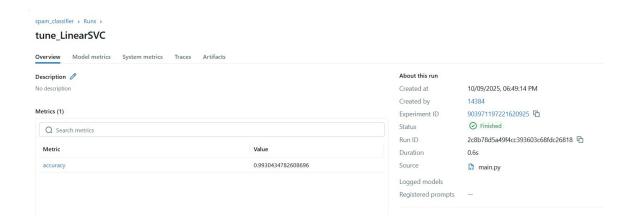
LogReg:



MultinimialNB:



Linearsvc:



4. Results

After training all three models, I compared their results. All of them gave good accuracy, but the Linear SVM and Logistic Regression models performed a little better than Naive Bayes. I checked the results using different metrics like Accuracy, Precision, Recall, and F1 Score. The confusion matrix also helped me see how many spam and ham emails were predicted correctly. Overall, the models were reliable and showed that the data cleaning and preprocessing were done properly.

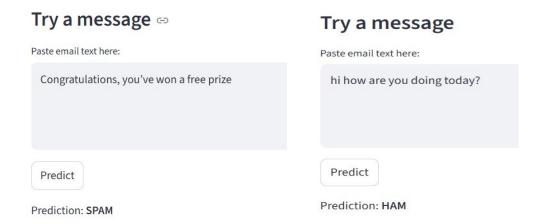
```
RESULTS (accuracy)
LinearSVC
             0.9896
MultinomialNB 0.9930
             0.9913
LogReg
BEST MODEL: MultinomialNB (acc=0.9930)
CLASSIFICATION REPORT (Ham=0, Spam=1):
              precision
                           recall f1-score
                                               support
        Ham
                 0.9939
                           0.9980
                                     0.9959
                                                   489
        Spam
                 0.9881
                           0.9651
                                     0.9765
                                                    86
                                     0.9930
                                                   575
    accuracy
                                                   575
   macro avg
                 0.9910
                           0.9815
                                     0.9862
weighted avg
                 0.9930
                           0.9930
                                     0.9930
                                                   575
```

5. Streamlit App

To make my project easy to use, I built a small Streamlit web app. In the app, users can type or paste an email and see right away if it's spam or ham. It also shows a confidence percentage, so you can know how sure the model is about its



prediction. The app also lets me choose between the three models (Naive Bayes, Logistic Regression, or Linear SVM) and compare how each one behaves.



6. Conclusion

This project helped me understand how machine learning can be used to detect spam emails. I did every step cleaning the data, building and testing models, comparing the results, and building an app. Using MLflow made it easier to track the experiments, and Streamlit made it possible to test the models interactively.

Overall, this project shows how data, code, and interface come together to solve a real problem.

Email Spam Classifier (Pick a Model)

Choose model: ○ LinearSVC MultinomialNB LogisticRegression Test accuracy: 0.9930

Classification Report

precision recall f1-score support Ham 0.9939 0.9980 0.9959 489 Spam 0.9881 0.9651 0.9765 86 accuracy 0.9930 575 macro avg 0.9910 0.9815 0.9862 575 weighted avg 0.9930 0.9930 0.9930 575

End of Report