Addis Ababa Science and Technology University

College of Electrical and Mechanical Engineering

Fraud Detection Using Machine Learning

Advanced Topics in AI Project

Name	IDNo
Anteneh Getachew	FTP 0138/09

Submitted To: Dr. Mehari K.

Submission Date: Sept 30, 2021

Introduction

Electronic mail is still a very cost-effective communication method but is also targeted by hackers to use it as a method to spread the virus, phishing, malicious code, and unnecessary advertising. Email is very handy and convenient to use but also misused by many.

HAM is genuine mail that the recipient is intended to receive and SPAM is spurious mail that is sent from unreliable sources in bulk to thousands of users, which is sent intentionally to users where the receiver is not supposed to receive it.

A machine learning model is a file that has been trained to recognize certain types of patterns in the data. There are so many machine learning models which can be applied in a lot of classification problems. In this assignment, we used two models to compare their performance on the spam-ham dataset. Random forest is a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest.

Bernoulli Naive Bayes is a variant of Naive Bayes which is a classification algorithm based on Bayes theorem which gives the likelihood of occurrence of the event. It accepts features only as binary values like true or false, success or failure,0 or 1, and so on.

Dataset

The dataset we used in this assignment is called spam_ham_dataset.csv(NSL-KDD) from the Kaggle website, which contains 5171 rows with three columns. These three columns are label, label_num, and text. Label and label_num both indicate the same thing whether the text is spam or ham. the text column contains the actual row mail text. There are no invalid or null entries in the dataset except the class imbalance can be shown in Figure 1.

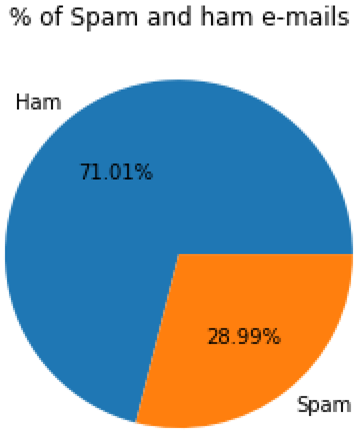

Figure 1. Percentages of Ham and spam emails

Data preprocessing

In this dataset, there are (4993, 3) unique data items, so we drop the duplicate items since this might result in some kind of bias.

The mail contains special characters other than the alphabet, and this might create a problem while we encode the text into numbers that can be understood by the algorithm, so we remove them. Then we convert the text into numbers by mapping each word as a feature and its value as a frequency of that word, to reduce the number of features we use 25 as a threshold to remove the word, which means if the word is repeated less than 25 times it will be removed.

As shown in the Figure the dataset is having class imbalance, this results in a bias prediction since machine learning models work based on the assumption that there is balanced data. In order to tackle this, we use SMOTE(synthetic minority oversampling technique) to oversample the minority class(i.e Spam).

After the balance, we got (6352, 2035) with equal negative and positive samples. Still, after using the SMOTE, the oversampled data contains 934 duplicate entries, so we remove them in order to reduce the class bias problem. Finally, as we can see in Figure 2 the data is about to be balanced.

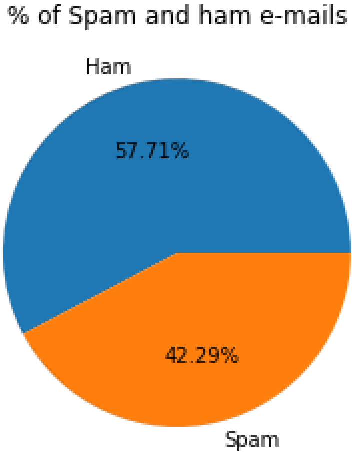

Figure 2. Class distribution after SMOTE and duplicate removal.

Methodology

After the data preprocessing phase, we then split the data into training and testing with 70% and 30% ratios respectively, and standardize the data to boost the learning speed during training.

Then we evaluate the performance of Random forest and NaiveBayes(Bernoulli) with the original dataset after duplicate removal and the oversampled data using SMOTE. In order to fix the threshold value which is used to reduce the number of features, we tried sample values 15, 20, 25, and 30, and the result is shown in Table 1.

Table 1: Performance comparison between Random forest and BernoulliNB models for samples of threshold values.

	Random Forest Model / BernoulliNB Model(%)			
words frequency	15	20	25	30
Accuracy(train)	99.94 / 94.1	99.94 / 94.54	**99.94 / 95.13**	99.93 / 95.25
Accuracy(test)	97.05 / 92.67	96.98 / 93.6	**97.12 / 93.74**	97.05 / 94.75

When the threshold value is 25 which works the best of other threshold values so we use this value to evaluate the SMOTE oversampled data.

Result and Discussion

The result we got from the experiments indicates that the random forest model outperforms both the Bernoulli and Gaussian Bayes Model as shown in Table 2.

Table 2: the performance of Random forest, and GaussianNB using SMOTE oversampled data.

Model	Train Accuracy	Test Accuracy
Random forest	**99.97**	**97.02**
BernoulliNB	96.76	96.4
GaussianNB(optional)	97.5	95.4

The detailed report of the performance using F1 score recall and precision scores is performed using training and testing datasets and the Random forest outperforms the BernoulliNB using SMOTE oversampled dataset.

RandomForestClassifier

(Training data)

Classification report:

```
          precision    recall  f1-score   support

       0       1.00      1.00      1.00      2210
       1       1.00      1.00      1.00      1615
```

(Testing data)

Classification report:

```
           precision    recall  f1-score   support

       0       0.98      0.98      0.98       944
       1       0.97      0.97      0.97       696
```

BernoulliNB

(Training data)

Classification report:

```
           precision    recall  f1-score   support

       0       1.00      0.95      0.97      2210
       1       0.93      1.00      0.96      1615
```

(Testing data)

Classification report:

```
           precision    recall  f1-score   support

       0       1.00      0.95      0.97       944
       1       0.93      0.99      0.96       696
```

The comparison graph that shows the performance evaluation of the two models using AUC(i.e using ROC curve) is shown in Figure 3.

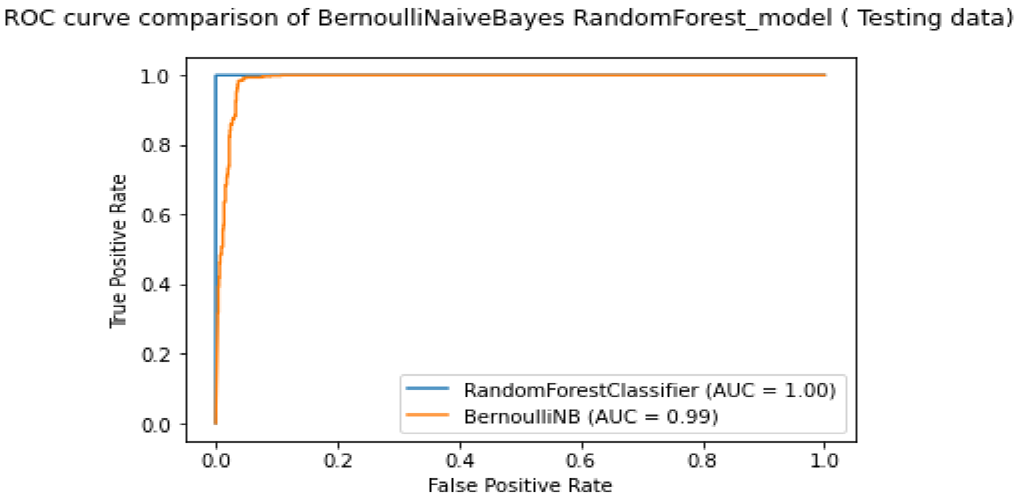

Figure 3. The performance comparison between random forest and BernoulliNB using AUC.

Since the random forest model outperforms the Bayes-based model we selected important features using Random forest as a base model, and some of the most important features are shown in Figure 4.

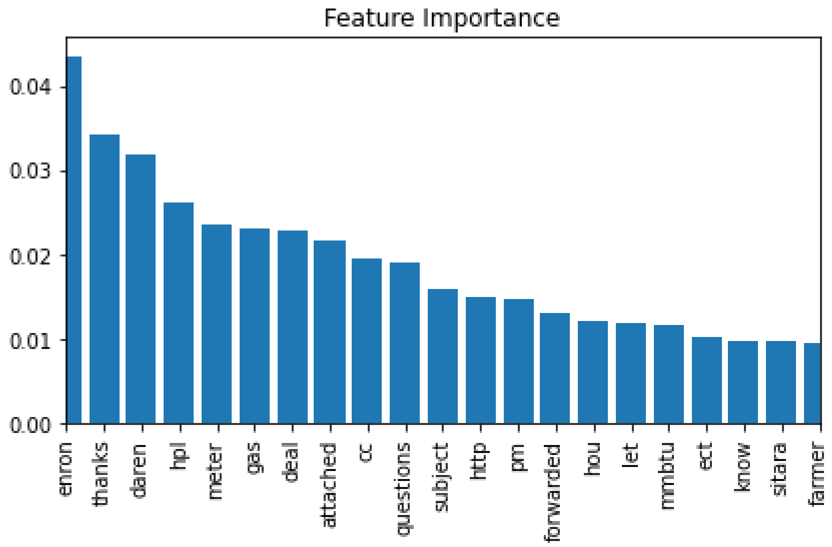

Figure 4. Feature importance analysis based on random forest model

Conclusion

As a conclusion to this assignment, the Random forest model outperforms that Bayes classifier for both the original dataset and SMOTE oversampled dataset. And the use of SMOTE oversampling technique improves the performance of the models in general, which helps the model to reduce the overfitting problem due to class imbalance. And the feature importance is also helpful to visualize and analyze which of the words have the most predictive power.