UNIVERSITY of GUELPH

Spam filtering using a Neural Network (NN)

Name: Omar Irfan Khan

ID: 1035394

Course: CIS*6420

Professor: Dr. Andrew Hamilton-Wright

Date: 29/11/2018

Table of Contents

LIST OF FIGURES

1 Introduction

Spam/Junk emails have become so common ever since the internet came into existence. Spam emails usually are in urgent nature and ask the user to immediately click here to help them or win a million dollars. These days spam is getting more personalized for each user to make the spammers seem more legitimate. This increases the chance of a user clicking on the link or downloading the email attachment substantially. Moreover, spam is also another attack vector used by hackers either to get your passwords via social engineering or to propagate viruses from one email to another. The former can lead to hijacking of the users account and impersonating the user while, the latter can infect the users system in the form of malware or adware.

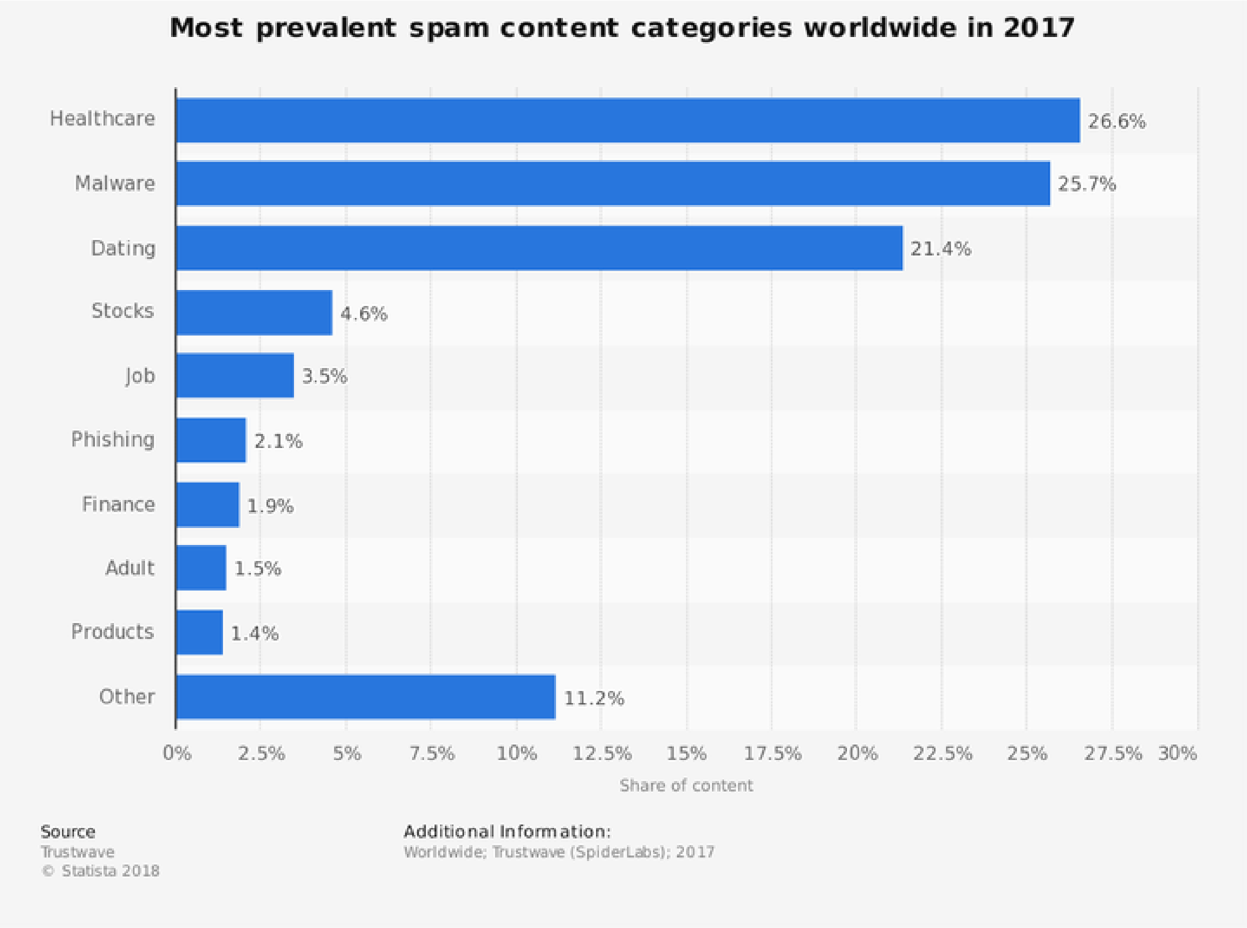


Fig.1: Spam statistic for 2017

The bar graph shows which category of spam was most prevalent in 2017. The same can be said for SMS spam we receive. It is in the same category and is done the same way as spam emails except we receive them now as text messages. Carriers such as Rogers, T-Mobile have a scanning software which filters out unwanted messages before they get sent. However, if some do arrive at the users mobile, they ask users to report the suspicious message [1] [2]. Furthermore, Canada has an Anti-Spam legislation that requires users to receive consent from recipients if they are sending commercial electronic messages (CEM) [3].

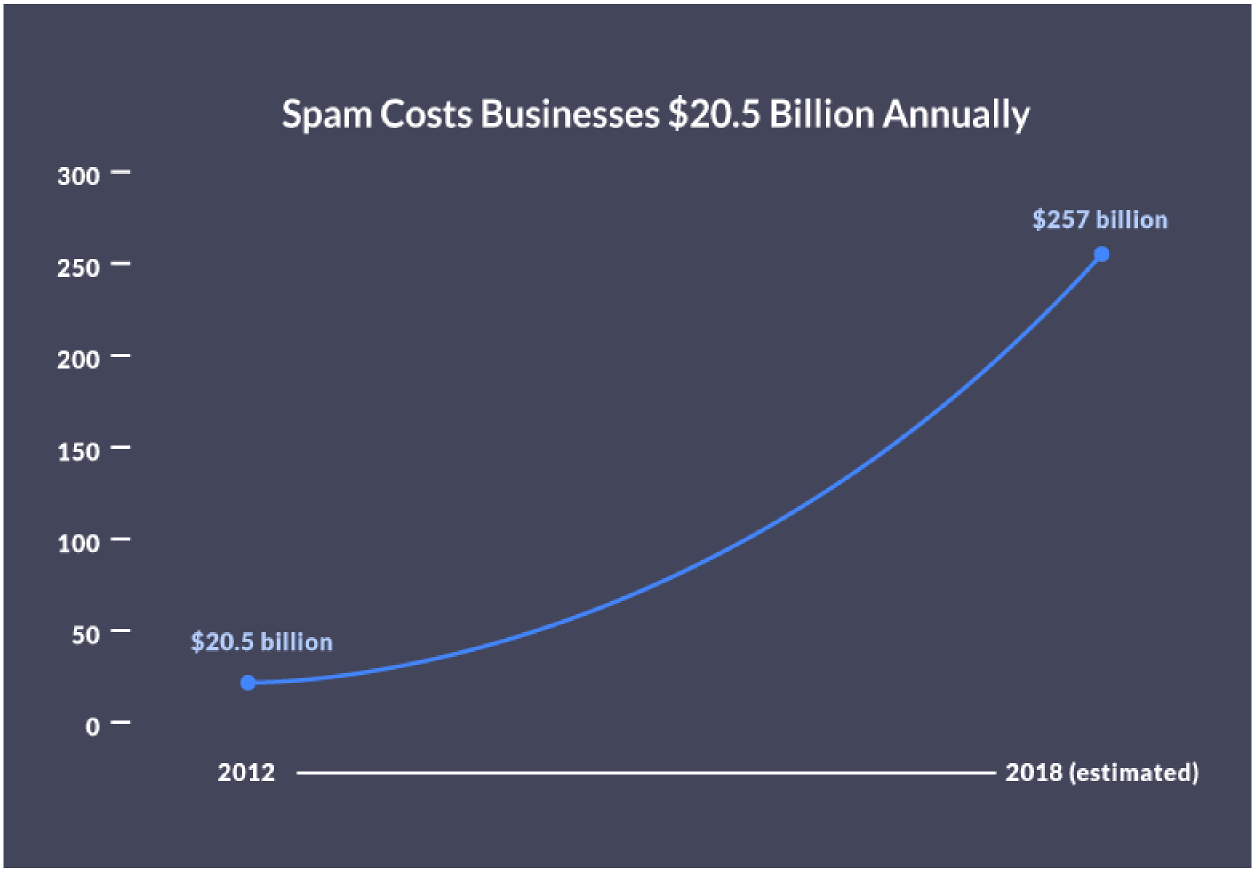


Fig.2: Spam cost

Spam results in lost productivity and technical expenses which results in businesses losing billions of dollars [4]. There is a need for an intelligent system that can accurately identify which messages are spam.

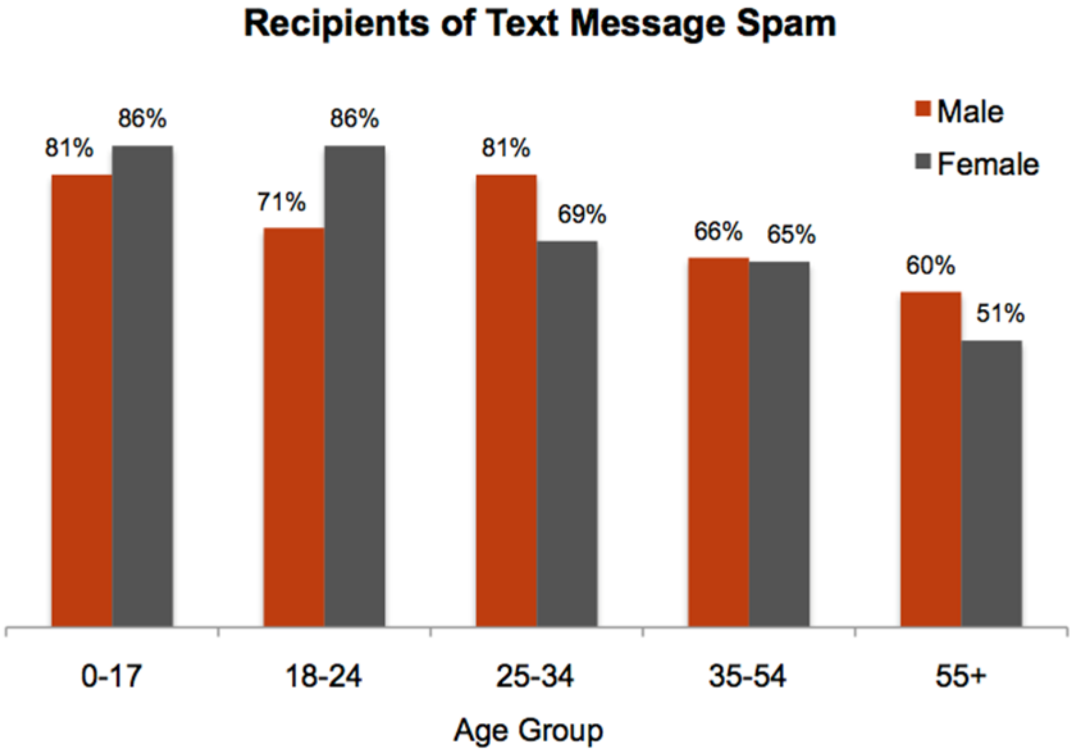


Fig.3 SMS Spam

This spam survey was carried out in US in 2011 and it shows how SMS spam varies between genders of various age groups. This project proposes using a feed forward neural network to classify SMS spam. Moreover, a comparison is made against other classifiers such as Multinomial Naïve Bayes, K nearest neighbor (KNN), Random Forest and Linear Regression to see how a feed forward neural network ranks against them.

2 Methodology

A simple feed forward neural network was used which consists of an input layer, hidden layer and output layer. The only difference between a feed forward neural network and a neural network is that a feedforward neural network only goes one way as compared to a general neural network which also back propagates to adjust the weights accordingly. A feed forward neural network is also called a multi-layer perceptron (MLP). The figure below shows how a feed forward neural network works.

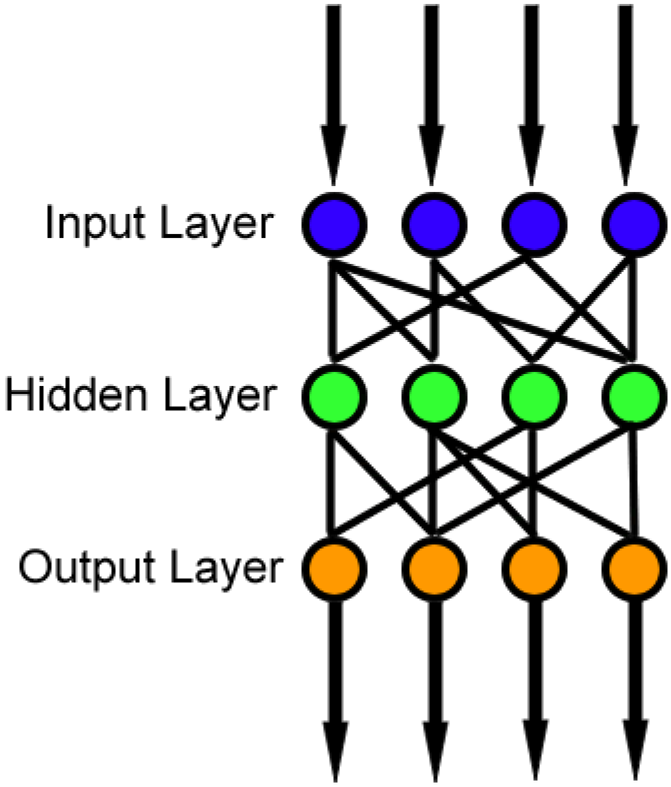
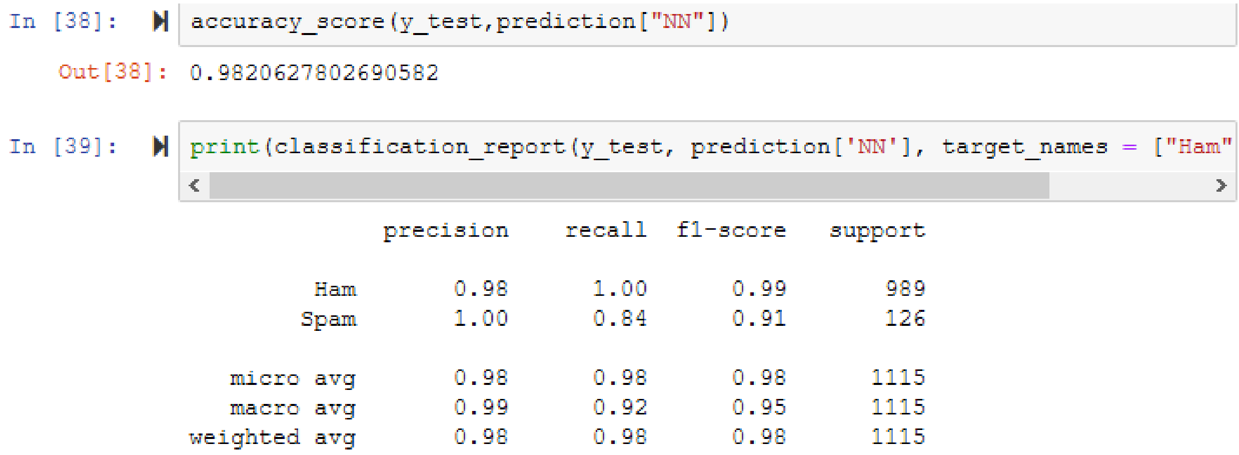


Fig.4: Feed Forward Neural Network Representation

The dataset was acquired from Kaggle [5] and consisted of 5574 messages. Moreover, out of these 5574 messages, 4827 were ham (legitimate messages) and 747 were spam messages. The data was split using the train/test split in 80:20. Training data consisted of 4459 while, testing data consisted of 1115 messages. There was minimal preprocessing needed since most of the dataset was already sorted out. A bit of labelling of the dataset was needed so that it can properly be stored in vectors.

3 Results

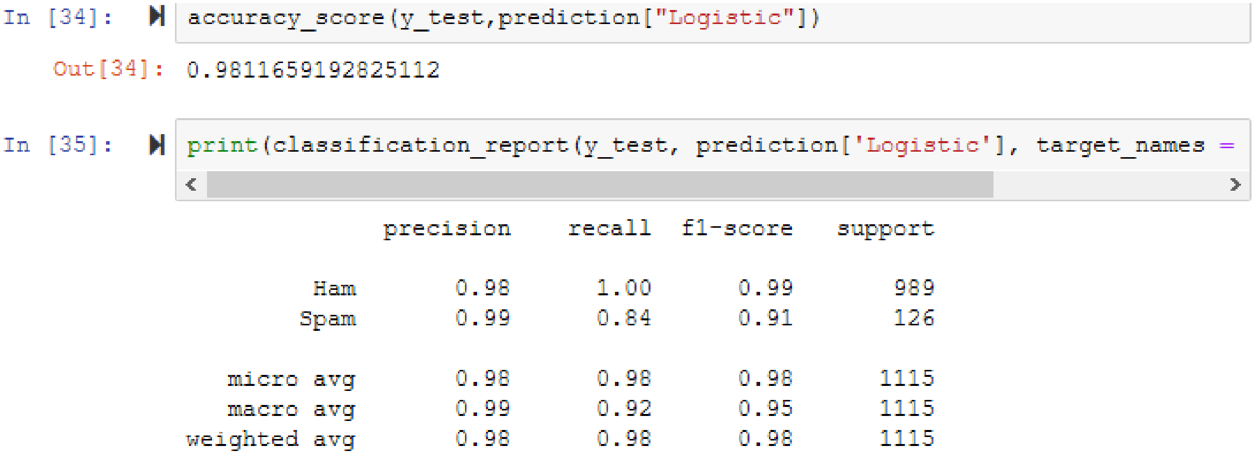
The accuracy, precision, recall and F1-score were measured for each classifier. Precision checks how many false positives are present for that specific data. While, Recall checks for how many false negatives are present. A recall value of 1 is produced if no false negatives are present. While, F1-score relies on both precision and recall. All the classifiers were able to achieve a recall score of 1.00 for ham messages, which indicates that no false negatives were produced and the models were able to correctly identify ham messages 100% of time.



```
In [38]:   accuracy_score(y_test,prediction["NN"])

Out[38]: 0.9820627802690582

In [39]:   print(classification_report(y_test, prediction['NN'], target_names = ["Ham"

              precision    recall  f1-score   support

         Ham       0.98      1.00      0.99       989
        Spam       1.00      0.84      0.91       126

   micro avg       0.98      0.98      0.98      1115
   macro avg       0.99      0.92      0.95      1115
weighted avg       0.98      0.98      0.98      1115
```

Fig.5 NN results

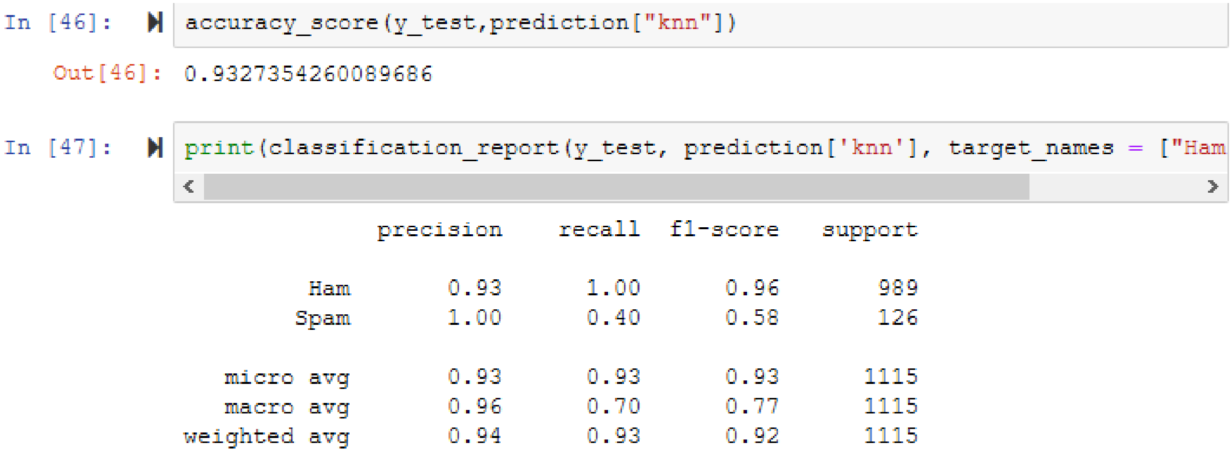
NN achieved a precision of 1.00 for spam messages meaning, that when the model predicts a message is spam, it is correct 100% of the time. This is incorrect as seen by the recall value of 84%. Further tuning may be required in the NN or k fold cross validation is needed.



```
In [34]:  accuracy_score(y_test,prediction["Logistic"])

Out[34]:  0.9811659192825112

In [35]:  print(classification_report(y_test, prediction['Logistic'], target_names =

              precision    recall  f1-score   support

         Ham       0.98      1.00      0.99       989
        Spam       0.99      0.84      0.91       126

   micro avg       0.98      0.98      0.98      1115
   macro avg       0.99      0.92      0.95      1115
weighted avg       0.98      0.98      0.98      1115
```

Fig.6 Linear Regression results

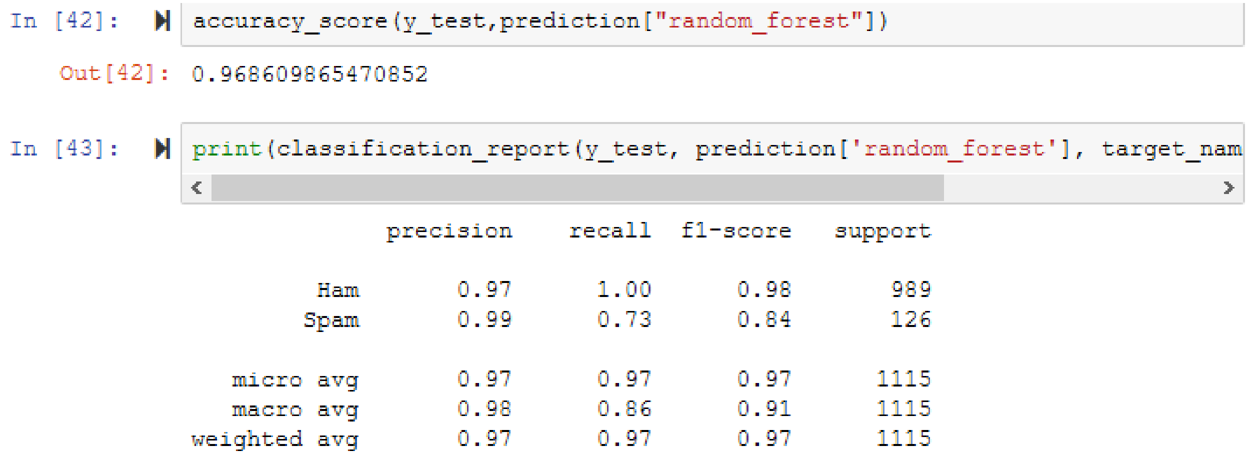
Linear regression also obtained similar results to NN and this may be due to the way the training and testing data was split. A further inspection must be done using one of the cross validation techniques.



```
In [46]:  accuracy_score(y_test,prediction["knn"])

Out[46]:  0.9327354260089686

In [47]:  print(classification_report(y_test, prediction['knn'], target_names = ["Ham

              precision    recall  f1-score   support

         Ham       0.93      1.00      0.96       989
        Spam       1.00      0.40      0.58       126

   micro avg       0.93      0.93      0.93      1115
   macro avg       0.96      0.70      0.77      1115
weighted avg       0.94      0.93      0.92      1115
```

Fig.7 KNN results

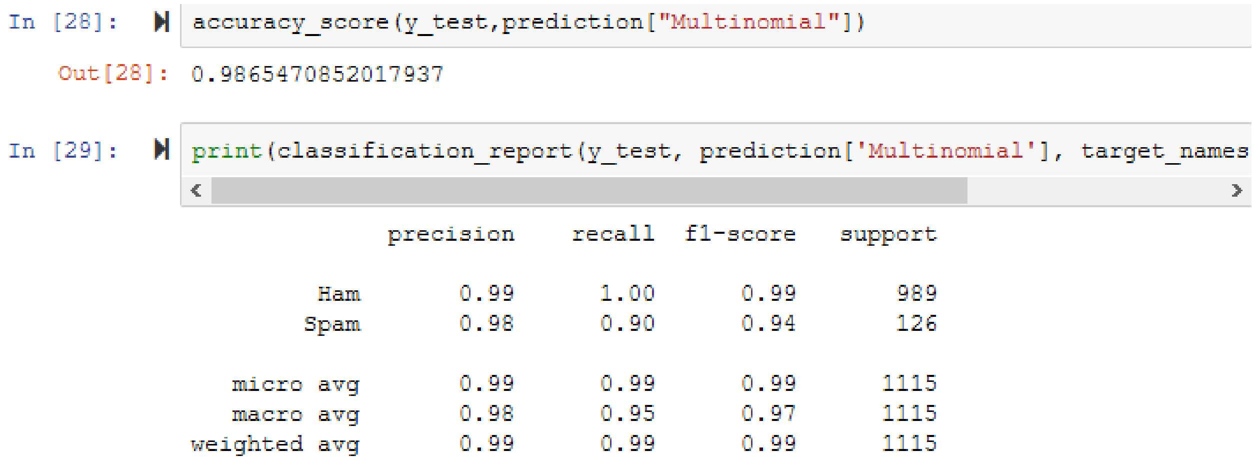
KNN also got a 1.00 in precision for spam messages which means a further tuning may be required using GridSearchCV.



```
In [42]:  accuracy_score(y_test,prediction["random_forest"])

Out[42]: 0.968609865470852

In [43]:  print(classification_report(y_test, prediction['random_forest'], target_nam

              precision    recall  f1-score   support

         Ham       0.97      1.00      0.98       989
        Spam       0.99      0.73      0.84       126

   micro avg       0.97      0.97      0.97      1115
   macro avg       0.98      0.86      0.91      1115
weighted avg       0.97      0.97      0.97      1115
```

Fig.8 Random Forest results

Random forest did semi decent on the dataset and obtained a 0.968 accuracy. While, its precision and recall for spam messages weren't nearly as bad as NN or KNN.



```
In [28]:  accuracy_score(y_test,prediction["Multinomial"])

Out[28]: 0.9865470852017937

In [29]:  print(classification_report(y_test, prediction['Multinomial'], target_names

              precision    recall  f1-score   support

         Ham       0.99      1.00      0.99       989
        Spam       0.98      0.90      0.94       126

   micro avg       0.99      0.99      0.99      1115
   macro avg       0.98      0.95      0.97      1115
weighted avg       0.99      0.99      0.99      1115
```

Fig.9 Multinomial Naïve Bayes results

Overall, Multinomial naïve bayes got the highest classification accuracy, followed by NN and linear regression. Unfortunately, due to the problems highlighted with NN and KNN another further test needs to be run with k fold cross validation to determine where the problem was and use GridSearchCV to tune the parameters of KNN.

4 Conclusion

Spam is prevalent and has been gradually increasing throughout the years. The results show that Multinomial Naïve Bayes achieved the highest accuracy as compared to the other classifiers. Moreover, the work could be improved by using k fold cross validation or leave one out cross validation (LOOCV). Furthermore, deep learning algorithms could also give a better classification accuracy with a bigger dataset.

5 References

1. *What Is Rogers Doing to Prevent Unsolicited or Suspicious Text Messages (SMS Spam)?* 2017, www.rogers.com/customer/support/article/spam-control.
2. “SMS Spam | Security Against SMS Spam.” *Cell Phones*, www.t-mobile.com/responsibility/privacy/fraud-spam/sms-spam.
3. “Canada's Anti-Spam Law (CASL) FAQ.” *Deloitte Canada*, 3 Aug. 2018, www2.deloitte.com/ca/en/pages/risk/articles/canada-anti-spam-law-casl-faq.html.
4. Bauer, Emily. *15 Outrageous Email Spam Statistics That Still Ring True in 2018*. 1 Feb. 2018, www.propellercrm.com/blog/email-spam-statistics.
5. Almeida, T.A., Gómez Hidalgo, J.M., Silva, T.P. *Towards SMS Spam Filtering: Results under a New Dataset*. International Journal of Information Security Science (IJISS), 2(1), 1-18, 2013.