Spam Detection Model Project

Submitted by:

Akanksha Padhye

ACKNOWLEDGMENT

Foremost, I would like to express my sincere gratitude to Data Trained team for the continuous support of my Data Science study and research, for the patience, motivation, enthusiasm, and immense knowledge. The guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my Data science study.

Besides Data Trained, I would like to thank Flip Robo Team, for their encouragement, insightful internship, and help to understand the study. My sincere thanks also go to SME Khushboo Garg for offering me the internship opportunities in their ally and leading me working on diverse exciting projects. I am over helmed in all humbleness and gratefulness to acknowledge my depth to all those who have helped me to put these ideas, well above the level of simplicity and into something concrete.

INTRODUCTION

Business Problem Framing:

You were recently hired in Start-up Company and you were asked to build a system to identify spam emails. We have to build a machine learning model that will predict if the email is ‘HAM’ or ‘SPAM’.

Conceptual Background of the Domain Problem:

Natural Language processing or NLP is a subset of Artificial Intelligence (AI), where it is basically responsible for the understanding of human language by a machine or a robot.

One of the important subtopics in NLP is Natural Language Understanding (NLU) and the reason is that it is used to understand the structure and meaning of human language, and then with the help of computer science transform this linguistic knowledge into algorithms of Rules-based machine learning that can solve specific problems and perform desired tasks.

Review of Literature:

In recent times, unwanted commercial bulk emails called spam has become a huge problem on the internet. The person sending the spam messages is referred to as the spammer. Such a person gathers email addresses from different websites, chatrooms, and viruses. Spam prevents the user from making full and good use of time, storage capacity and network bandwidth.
Electronical Communication is the need of the day. For shopping for bread to hollering the Emergency Services there's no second to communication.

Users who receive spam emails that they did not request find it very irritating. It is also resulted to untold financial loss to many users who have fallen victim of internet scams and other fraudulent practices of spammers who send emails pretending to be from reputable companies with the intention to persuade individuals to disclose sensitive personal information like passwords, Bank Verification Number (BVN) and credit card numbers.

Motivation for the Problem Undertaken:

This is a Natural Language Processing Problem and this will lead us to create highly communicative machines using human-native language.

Analytical Problem Framing

Mathematical/ Analytical Modeling of the Problem:

1. The Problem is of Classification.
2. The dataset contains 5572 rows and 5 columns.
3. Columns unnamed 2,3,4 are unnecessary columns. Hence we have dropped them.
4. We have renamed the column v1 as target and v2 as sms.
5. Target column has 2 values i.e ham for non-spam sms and spam for spammed sms.
6. Using appropriate metrics for scoring and evaluations .

Data Sources and their formats:

The data was provided by the client to "FlipRobo Technologies". The data is in the form of a comma separated file (CSV). The data i.e. the features and the target are in the single file.

Data Pre-processing:

The Data pre-processing done is as follows:-

1. Removing Stop words from the data.
2. Removing punctuations and other special characters from therecords
3. Some more granular cleaning for treating hyphen andunderscore joined words.
4. Removing the words which are less than 3 letters in length
5. Perform Stemming using PorterStemmer class from sklearnlibrary
6. Further, we remove all the words which do not convey anymeaning in the context of the English Language
7. Vectorize the data using tf-idf Vectoriser

Data Inputs- Logic- Output Relationships:

1. Describe the relationship behind the data input, its format, the logic in between and the output.
2. Describe how the input affects the output.
3. Data is fed in the form of a Pandas data frame to the model.
4. The data is the vectorised meaningful words of the records. For the output we get the predicted label value of the record, that is whether the document is likely to be a same email of not.
5. The output results in a binary value either 1 or 0 respectively.

State the set of assumptions (if any) related to theproblem under consideration:

There are no such formal assumptions as we are using the Naïve Bayes' Multinomial NB algorithm. The multinomial Naive Bayes classifier is suitable for classification with discrete features (e.g., word counts for text classification). The multinomialdistribution normally requires integer feature counts.

Hardware and Software Requirements and Tools Used

Hardware Required:

- A computer with a processor i3 or above.
- More than 4 GiB of Ram.
- GPU preferred.
- Around 100 Mib of Storage

Space.Software Required:

- Python 3.6 or above
- Jupyter Notebook.
- Google Collab.
- Excel

Library Used:

1. Computing Tools:
   - Numpy
   - Pandas
   - Scipy
   - Sk-learn
   - NLTK
2. Visualizing Tools:
   - Matplotlib
   - Seaborn
3. Saving Tools:
   - Joblib

Model/s Development and Evaluation

Identification of possible problem-solving approaches:

Describe the approaches you followed, both statistical and analytical, for solving of this problem.

Testing of Identified Approaches:

The Algorithms used for testing, training and Validating the modelsare as follows:

- Logistic Regression
- Decision Tree
- AdaBoost Algorithm
- Naïve Bayes
- Random Forest
- Extra Trees classifier

Run and Evaluate selected models:

Algorithms used and their Evolutions of the Selected Models:

```
MultinomialNB Classifier

Accuracy Score of MultinomialNB Classifier : 0.9825918762088974

Precision Score of MultinomialNB Classifier : 1.0

Confusion matrix of MultinomialNB Classifier :
 [[1353    0]
 [  27  171]]

classification Report of MultinomialNB Classifier
              precision    recall  f1-score   support

           0       0.98      1.00      0.99      1353
           1       1.00      0.86      0.93       198

    accuracy                           0.98      1551
   macro avg       0.99      0.93      0.96      1551
weighted avg       0.98      0.98      0.98      1551
```

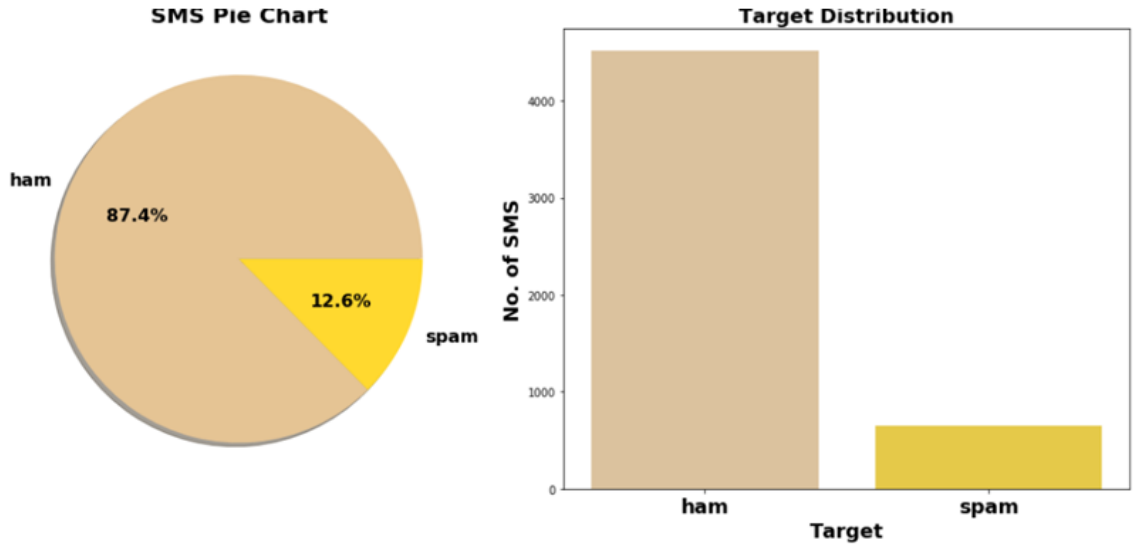
```
Cross Validation Score MultinomialNB() :

Precision CVScore : [0.99152542 1.         1.         1.         0.97391304]
Mean Precision CV Score : 0.9930876934414149
Std deviation : 0.01013358607546182
```

Key Metrics for success in solving problem underconsideration:

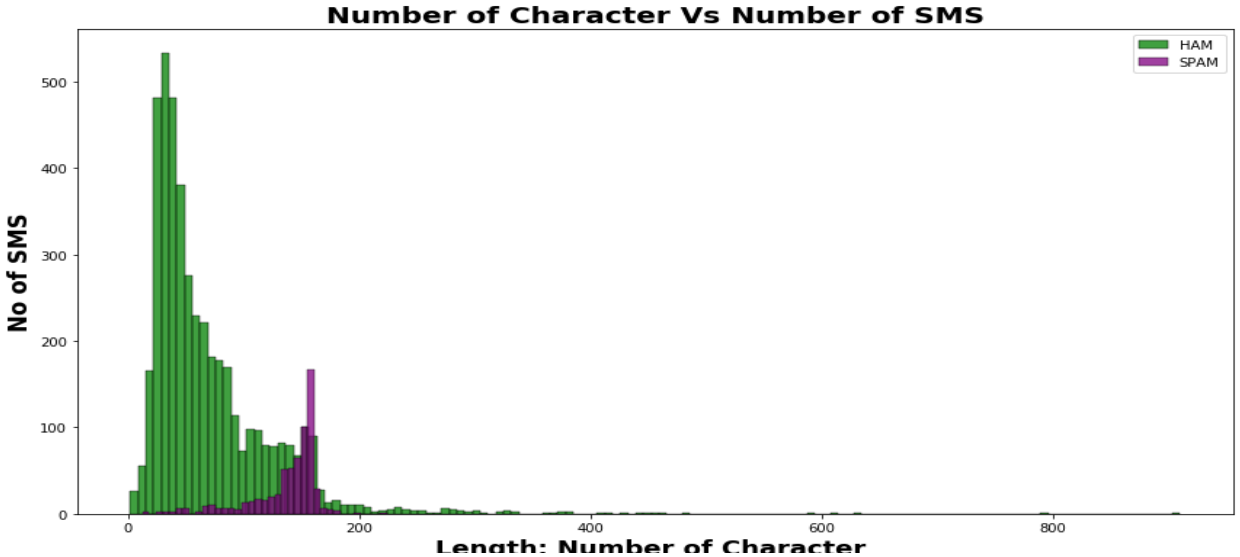
The key-metric under considerations is Recall and F1 score althoughthe model was finalized on basis of other metrics as Matthew's Correlation Coefficient (MCC) as well as F1-score.

Data Visualizations:

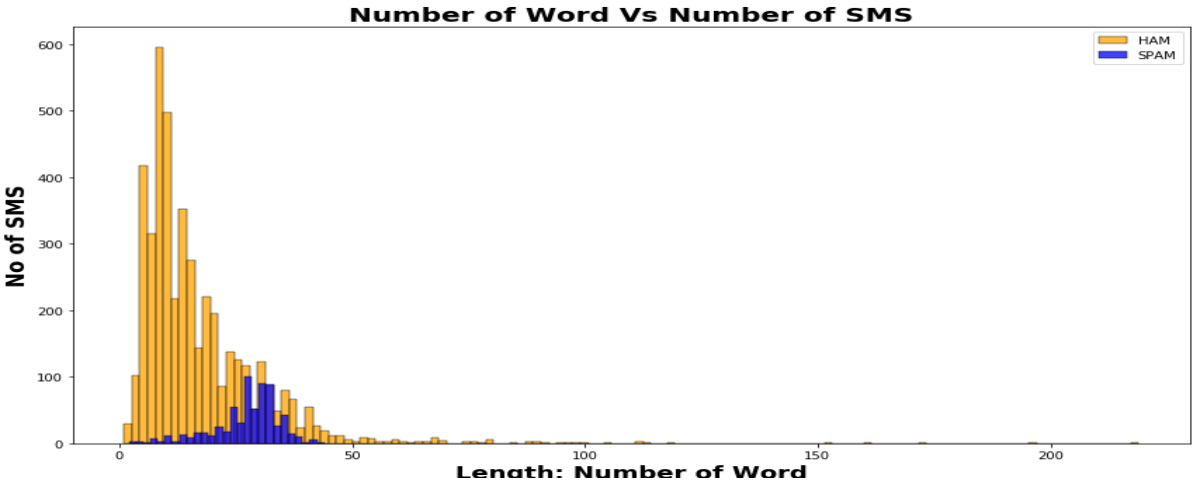
There are around 87.4% sms ham and 12.6% sms spam. i.e HAM are 4516 and SPAM are 653 in counts.



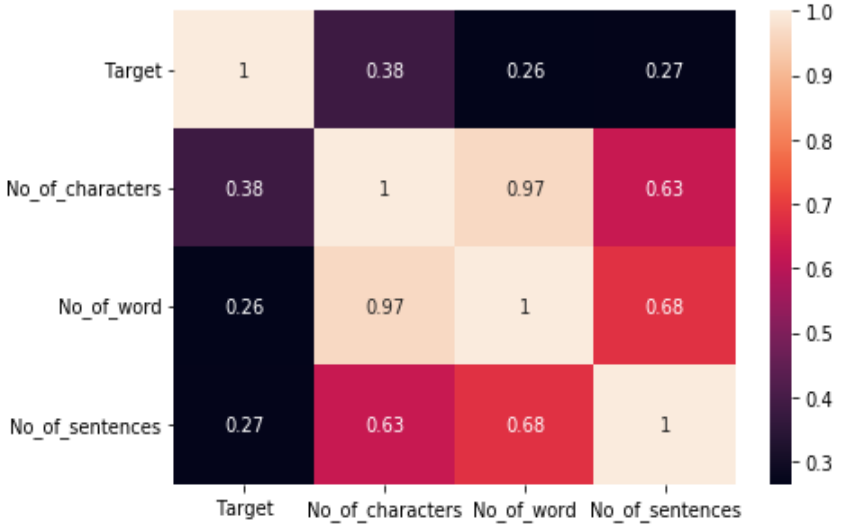
Number of character in Spam sms is comparetively much high than Non-Spam (ham) sms.
On average each ham sms contain 71 character, 17 words and 2 sentences.



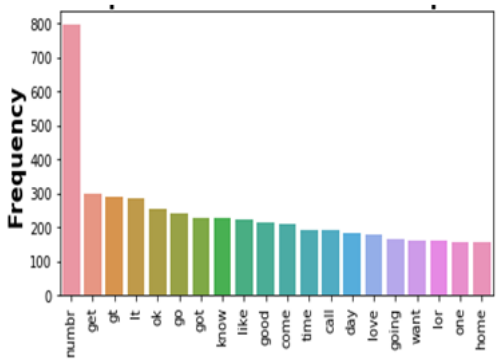
Number of Word in Spam sms is comparetively much high than Non-Spam (ham) sms.
On average each spam sms contain 138 character, 27 words and 3 sentences.



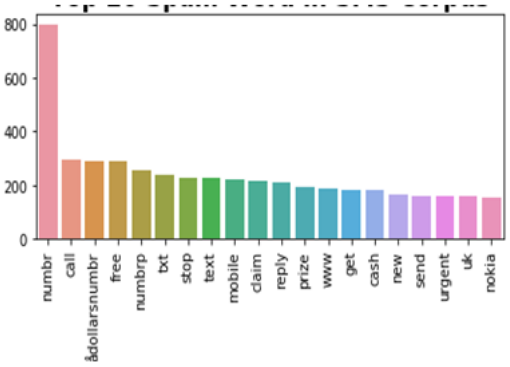
Target has 0.38 correlation with no. of characters which is maximum amongst the rest.
Target has 0.27 correlation with no. of sentences and 0.26 with no.of words.



- Top 20 Ham words in SMS corpus.



- Top 20 Spam words in SMS corpus.



Interpretation of the Results:

After performing logistic regression, multinomial NB, extra trees classifiers, random forest, adaboost classifier and bernoulli NB algorithms and doing cross validation for each model we selected multinomial NB as the best model because its giving the best accuracy as compare to other models.

```
MultinomialNB Classifier

Accuracy Score of MultinomialNB Classifier : 0.9825918762088974

Precision Score of MultinomialNB Classifier : 1.0

Confusion matrix of MultinomialNB Classifier :
 [[1353    0]
 [  27  171]]

classification Report of MultinomialNB Classifier
              precision    recall  f1-score   support

           0       0.98      1.00      0.99      1353
           1       1.00      0.86      0.93       198

    accuracy                           0.98      1551
   macro avg       0.99      0.93      0.96      1551
weighted avg       0.98      0.98      0.98      1551
```

```
Cross Validation Score MultinomialNB() :

Precision CVScore : [0.99152542 1.         1.         1.         0.97391304]
Mean Precision CV Score : 0.9930876934414149
Std deviation : 0.01013358607546182
```

CONCLUSION

Key Findings and Conclusions of the Study:

NLP gets hard are humans are not used to typing as proper grammar these years.

Sweet spot should be found between whether to pick stemming or lemmatization or both.

Naïve Bayes algorithms are quicker than rest of the algorithms.

Learning Outcomes of the Study in respect of DataScience:

List down your learnings obtained about the power of visualization, data cleaning and various algorithms used. You can describe which algorithm works best in which situation and what challenges you faced while working on this project and how did you overcome that.

Outcomes of the Study:

- Almost 90 percent of the time is spent of data cleaning anddata modelling.
- You do not get a Gaussian distribution in real-word problem.
- NLP becomes difficult due to sloppy use of language byhumans
- This also created issue while teaching to machines
- Algorithms like Support Vector Machines and K nearest neighbours may take a long time to converge on a Huge dataset like this.
- Naïve Bays is very quick as of converging rate.

Limitations of this work and Scope for Future Work:

More data is always appreciated.

The model could be integrated with the any email app used by the Data Analysts and Developers for easy spam filtering decision.

The model could be placed into a Continuous Integration and Continuous Deployment for online training environment.