# Data Science Project Report

# SMS Spam Detection

May 5, 2019

| | |
|---|---|
| Professor | Dr Atif Tahir |
| Project Member 1 | Abdul Munim Khan(K15-2897) |
| Project Member 2 | Muhammad Moiz Arif(K15-2146) |
| Submission Date | May 06, 2019 |

| Task | Performed By |
|---|---|
| Setting Research Goal | Both |
| Retrieving Data | Member 1 |
| Data Preparation | Member 1 |
| Data Exploration | Member 2 |
| Data Modeling | both |
| Data Presentation | both |

# Contents

# 1 Introduction

This document contains the steps of Data Science Process for the project that is SMS Spam Detection.

# 2 Setting Research Goal

The purpose of the project is to classify the text messages based on the prediction done by the trained model that is trained by the given dataset. We are doing the message classification as Spam or Ham that is the message is from legal sender or it's a chain of same messages in order to just generate traffic on the targeted network or with some negative intention. We will use Naïve Bayes algorithm for the text classification as it is the best algorithm used in textual analysis. The Naïve Bayes algorithm use the Bayes law which is stated following:

$$P(y|x) = (P(x|y) * P(y))/P(x) - - - - - - - - - - - - - - (i)$$

The probability of y given that the event x occurs. We are using this law for spam classification as follows.

$$P(spam|W1,W2,W3) = (P1 * P2 * P3)/(P1*P2*P3)*(1-P1)*(1–P2)*(1–P3)$$

The above formula is just elaboration of how we will predict the spam message after the training from dataset.

# 3 Retrieving Data

The data has been taken from the online dataset provider repository that is UCI Machine Learning Repository. The dataset is present at the following link.

https://archive.ics.uci.edu/ml/datasets/SMS+Spam+Collection#

# 4 Data Preparation

According to the UCI repository dataset information, the data is collected from different resources and so it must need to be is particular order so that the model can be train from specified attributes. Therefore, for data cleaning only the textual analysis involves so we removed the extra spaces from the messages and those words that are not in the dictionary that is removal of slang words.

# 5 Data Exploration

From figure below $(747*100/5572) = 13.46\%$ text messages from the given dataset are spam and the rest of the messages that is 86.54% of the messages are ham. Also the messages are repeating (some messages), so the unique messages in both ham and spam are less than the total count. The frequency of the most repeated text is thirty it means that the message 'Sorry, I'll call later' arises thirty times in the messages.

The frequency of the most repeated text is thirty it means that the message 'Sorry, I'll call later' arises thirty times in the messages.

```
message.describe()
```

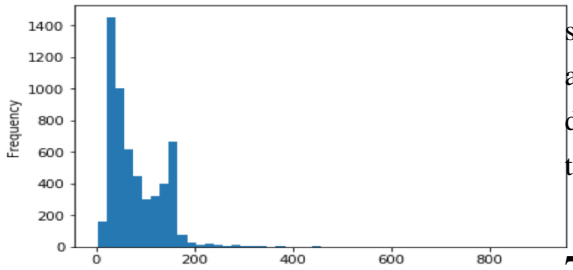
| | labels | message |
|---|---|---|
| count | 5572 | 5572 |
| unique | 2 | 5169 |
| top | ham | Sorry, I'll call later |
| freq | 4825 | 30 |

```
message.groupby('labels').describe()
```

| | message | | | |
|---|---|---|---|---|
| | count | unique | top | freq |
| labels | | | | |
| ham | 4825 | 4516 | Sorry, I'll call later | 30 |
| spam | 747 | 653 | Please call our customer service representativ... | 4 |

```
message['length'].plot(bins=50,kind='hist')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0xe6e1ac8>
```




## 6 Data Modeling

We have tokenize the data that is tokenize the messages into the list of words. Ex We have the message in the .csv file as ‘Ok lar joking wif u oni’. After tokenization this message becomes list of words as shown. [Ok, lar, joking, wif, u, oni].

Secondly, we tokenize the messages by stemming parts of speech. For Ex. In the above message given the wif is wife and u is you. Therefore replacing the correct word from its short. The results of the trained model are given as follows. The following fig showing the predicted and expected value of the message 3 in the dataset.

## 7 Presentation and Automation

The results of the trained model are given as follows. The following fig showing the predicted and expected value of the message 3 in the dataset.

```
In [174]: print('predicted:',spam_detect_model.predict(tfidf4)[0])
          print('expected:',message['labels'][3])

          ('predicted:', 'ham')
          ('expected:', 'ham')
```

The overall results for the entire dataset is shown below.

```
In [175]: all_predictions = spam_detect_model.predict(messages_tfidf)
          print(all_predictions)

          ['ham' 'ham' 'spam' ... 'ham' 'ham' 'ham']
```