Thesis: Title: Use of artificial intelligence and data analytics to compare the e-mail spam filtering (YAHOO, Gmail, Outlook) depending upon user experiences

Chapter # 4: Analysis & Results

Student Name: Shoaib

Table of Contents

[Chapter # 4: Results & Analysis 4](#_Toc95143558)

[Email Spam Classification App that build by using the streamlit & python 4](#_Toc95143559)

[Streamlit python library installation: 4](#_Toc95143560)

[Dataset information & Python file 5](#_Toc95143561)

[Data Cleaning & Pre-processing 6](#_Toc95143562)

[Machine Learning Library to build spam models 6](#_Toc95143563)

[Training & testing the dataset 7](#_Toc95143564)

[Building the Multinomial Naïve Bays Classifier Model 7](#_Toc95143565)

[Results of Streamlit python Spam Detector in Browser: 8](#_Toc95143566)

[Analysis: 9](#_Toc95143567)

[Support Vector Machine Classifier 9](#_Toc95143568)

[Python Libraries: 9](#_Toc95143569)

[Data Set information 9](#_Toc95143570)

[Split dataset in Training & Testing data 9](#_Toc95143571)

[Extract Features CountVectorizer: 10](#_Toc95143572)

[Problem with Naïve Bays: 10](#_Toc95143573)

[Results 11](#_Toc95143574)

[Analysis: 11](#_Toc95143575)

[Naïve Bays Classifier to Detect Spam Emails 12](#_Toc95143576)

[Importing Python Libraries 12](#_Toc95143577)

[Data set information 13](#_Toc95143578)

[Date Preprocessing 13](#_Toc95143579)

[Training & Testing the dataset 14](#_Toc95143580)

[CountVectorizer to convert words into matrix features 15](#_Toc95143581)

[Multinomial Naïve Bays algorithm to classify text and build the model 15](#_Toc95143582)

[Results 15](#_Toc95143583)

[Analysis 16](#_Toc95143584)

[Machine Learning Logistics Regression to Detect Spam emails 16](#_Toc95143585)

[Data Preprocessing and Importing CSV 17](#_Toc95143586)

[Data Cleaning & Text Transformation 20](#_Toc95143587)

[NLTK library to extract spam words and ham words 21](#_Toc95143588)

[TFID-IDF Vectorizer 22](#_Toc95143589)

[Transform the string text into numeric array 22](#_Toc95143590)

[Train & test the data validation 22](#_Toc95143591)

[Outcomes & Results 23](#_Toc95143592)

[Analysis 23](#_Toc95143593)

[References 24](#_Toc95143594)

# Chapter # 4: Results & Analysis

Machine learning & deep learning based modeling has been tested and verified the email spam detection, various emails contains junks and spams text and useless information. Machine modelling build to analyze the spam email method, Google emails used machine learning model in Gmail services that predict and analyze the spams emails widely in the globe. Gmail service is one of the best email service in the world that detect and filter the spam and ham emails and forward the spams mails into the spam junk folder. Python machine learning based modelling has been built to test the email spam data which is downloaded from Github repository.

## Email Spam Classification App that build by using the streamlit & python

(Freydenberg & Kevin, 2020) Project based learning to make software application using the novel python based approach, which is effectively used to test and predict the software applications within the environment. Software application tested the machine learning models and predict the spam and ham news in the internet browser. Streamlit is creative python application which used to present the application into the web browser and analyze the various working of the python application. The interactive python streamlit package instantly creates the web application which used to test any machine learning and deep learning project.

### Streamlit python library installation:

Open the python Pycharm 2021 version and installed the following libraries in the Pycharm library.

1. Pip install streamlit
2. Add the streamlit package in Pycharm community 2021 version.

The following code generate in Pycharm community edition.

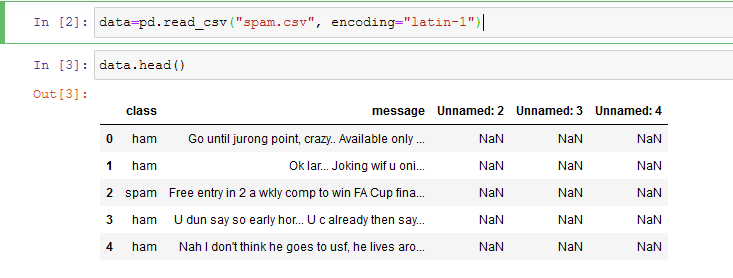


*Figure 1: Streamlit Code*

And building the spam detection model in the python anaconda jupyter notebook to verify the spam emails and ham emails.

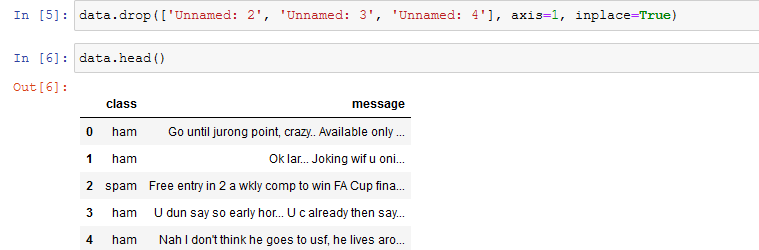
### Dataset information & Python file

The dataset and python model has been used from Github repository. (Github, 2021). The dataset contains the spam and ham email text which contain the csv file.

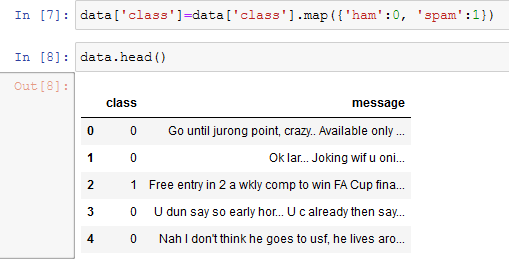


### Data Cleaning & Pre-processing

Dataset has been clean to remove the unknown values in the dataset. It also used to verify the message and class values of the dataset. Data.drop syntax remove the unnamed columns in the dataset. And remove the NaN values in the dataset.

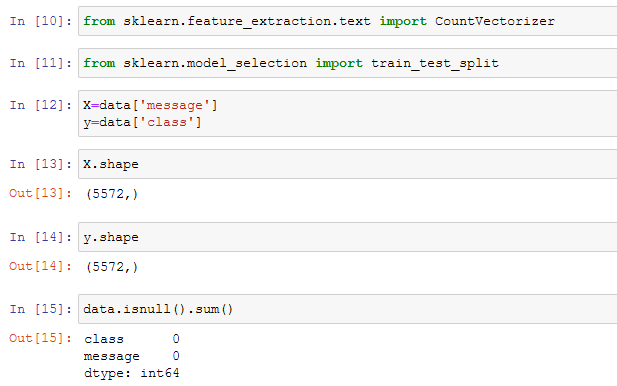


Converting the text data into the numeric values to check and verify the spam emails and ham emails from the dataset. Data class to convert the class columns to 0 and 1 values, ham = 0 and spam =1.



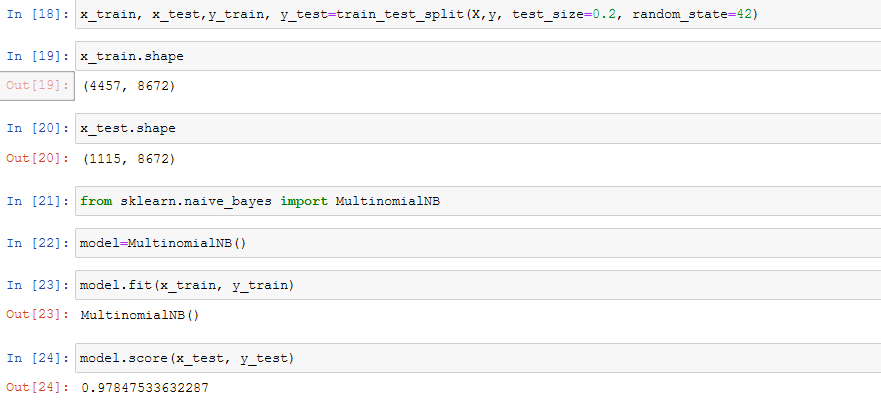
### Machine Learning Library to build spam models

Machine learning library import in python jupyter notebook, which used to analyze the dataset. The following library has been used to predict the dataset values, sklearn library and countvectorizer library contains rich information to process the text message and class values that is based on the 0 and 1 value.



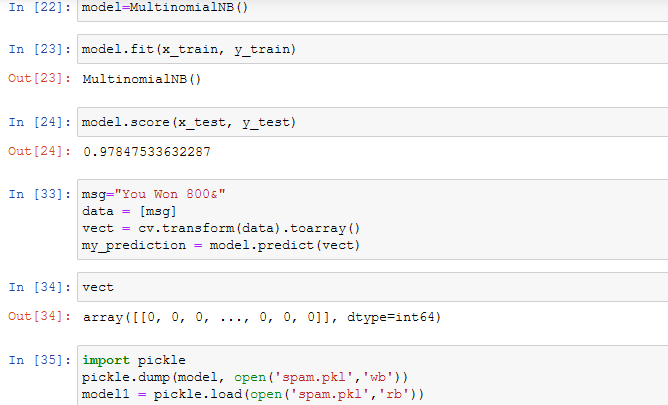
### Training & testing the dataset

To split the 20% data for testing and 80% data to training and make assumptions to verify the text data is contains spam value and ham values which is tested on the base of class values.



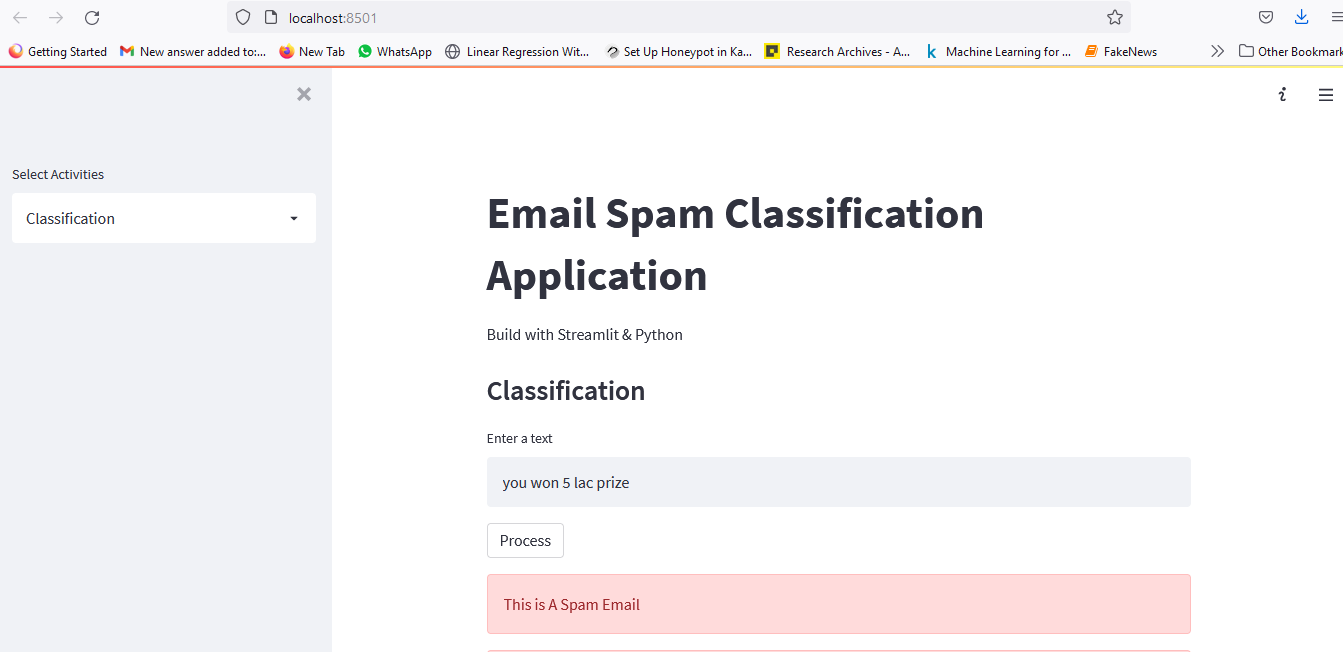
## Building the Multinomial Naïve Bays Classifier Model

The naïve bays classifier which contains multinomial features that used to predict the numeric class which identify the 0 and 1 values either it contains to proceed the spam and ham values.



After building the naïve bays classifier model with the help of pickle spam files and other spam generator to verify the naïve bays modelling. (Nance & Baumgartner, 2021) Machine learning used to analyze the working of natural language processing, various labelled machine learning approaches is based on binary classification which used to predict the text in numerical form to analyze the values is spam either ham. Training and testing the machine learning models which divided the dataset and verify each text content based on NLP prediction.

### Results of Streamlit python Spam Detector in Browser:



*Figure 2: Email Spam Detector*

The above mentioned application quickly build using the streamlit python library and make assumptions of spam and ham dataset. (Github, 2021)

### Analysis:

(N. L. Octaviani, 2020,) The comparison of multinomial naïve bays classifier which used to predict the email spams, it also used to compare the outcomes of support vector machine as well as recurrent neural network, the abused email considered spam emails and it identify the sender identity, mostly it used for marketing purposes but it also used for making fraud, hacking data and steal information to send the malware information and dummy link to the user. The naïve bays model score is 97%.

## Support Vector Machine Classifier

Support vector machine classifier (Shahi & Yadav, 2014) represent various feature to detect the spam and ham emails, the classification method used to predict the different domains such as text messages and email message which used from dataset.

### Python Libraries:

1. Pandas
2. SKLEARN
3. SKLEARN Multinomial Naïve bays and Gaussian Naïve Bays
4. SKLEARN SVM

### Data Set information

Dataset contains csv file downloaded from YouTube repository including Google Drives links

Python SVM model (Classification, 2021).

### Split dataset in Training & Testing data

Split data in x and y variable dataframe. X = EmailText Y = Label the dataset.

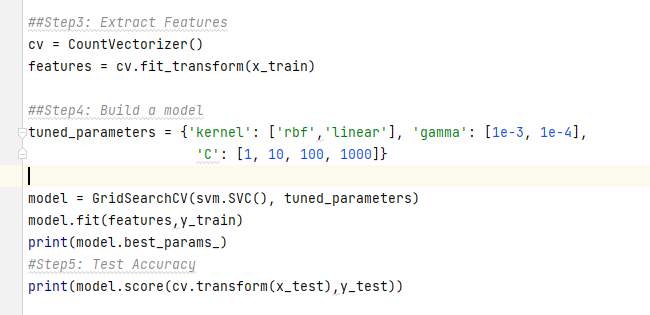


### Extract Features CountVectorizer:

(sklearn.feature\_extraction.text.CountVectorizer, 2021) The method of countvectorizer is used to convert the text into the matrix of token counts. The dictionary of dataset is provided from the csv file dataset and it analyze the vocabulary size which based on analyzing the data, build analyzer () return the callable to process the input data. Build preprocessor return the function to preprocess the text before tokenization. Build tokenize return the functions that splits string into sequence of tokens.

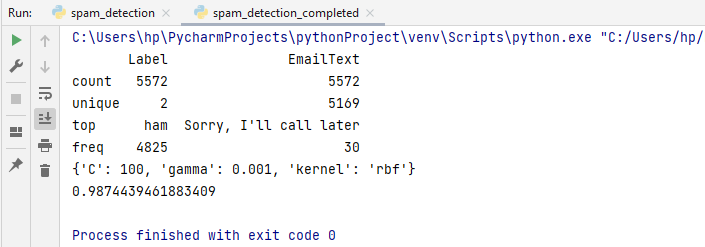
### Problem with Naïve Bays:

It assumes the attributes are independent and the week point can be solved performing some statistical analysis before using the naïve bays to measure the correlation degree among features and then selection the most uncorrelated ones. Zero probability problem it can be solved by adding values one to the frequency each attribute used the Gaussian distributed method. It treats all attributes equally so some weights can be added to the important attributes to increase the contribution in the final decision, continues values attributes problem conversion from continues to discrete value is the solution of such problem.



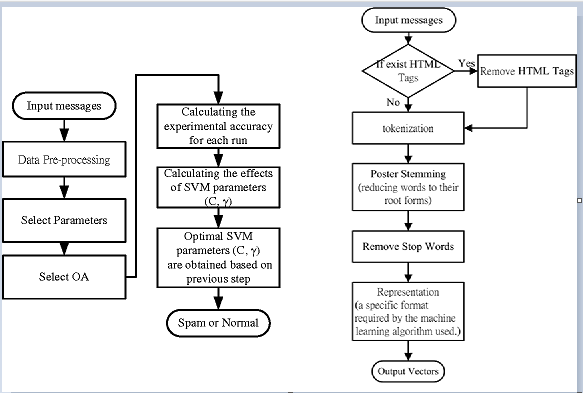
### Results

The above result shows the SVM classifier accuracy about to 98% accuracy of spam and ham email detection which is achieved the best accuracy of the model. Label count the 5572 email text and unique values is 2 and top value is ham message which contains I will call you later. Frequency is 4825 and 98% accuracy of spam detection from the given dataset.



### Analysis:

(Roy, et al., 2017) The support vector machine classifier model used the artificial neural network to compare the performance of the machine learning models, the deep SVM model used to predict the ham and spam emails. The available statistical models which analyzed the text on countvectorizer strategies and convert the string text into the numeric digital which is analyze in 0 and 1 form. The SVM classifier executes with multinomial naïve bays model and built the model with effective python sklearn library and it build the 98% accuracy which considered good accuracy in the model building framework. Spam filtering method essential features in email detection which train and test the dataset and make the model predictions which is used the labels text dataset.



*Figure 3: SVM Spam Detector flow chart*

## Naïve Bays Classifier to Detect Spam Emails

Naïve bays classifier is much effective machine learning algorithm, it uses multinomial method to extract the text countvectorizer and make assumptions on it, dataset split into two phases one is test and other training set. Machine learning python libraries used to import the spam and ham text algorithm which classify on binary method.

### Importing Python Libraries

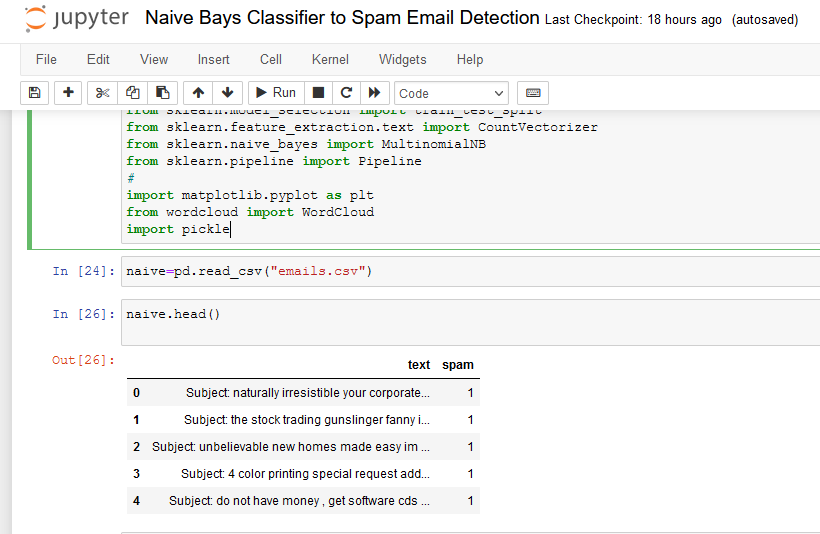
The following libraries import in anaconda jupyter notebook:

1. Sklearn Naïve bays
2. Sklearn multinomial
3. Import countvectorizer
4. Import wordcloud
5. Import pickle
6. Import pandas and numpy

Read the csv file email csv file

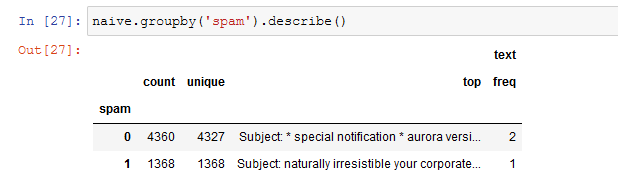
### Data set information

Data set downloaded from Github repository, email csv file



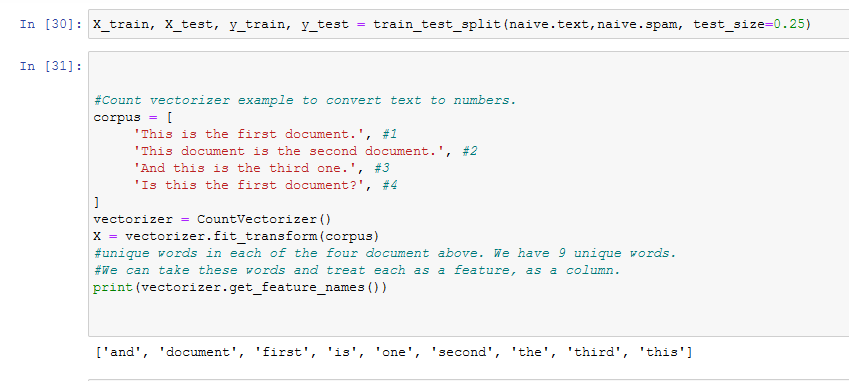
### Date Preprocessing

Data preprocessing has been performed to analyze the dataset frequency, count the word count and count the description of the dataset levels.

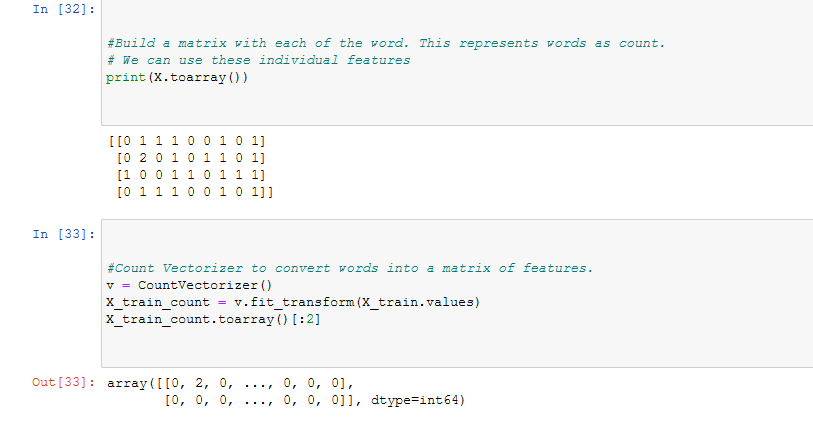


### Training & Testing the dataset

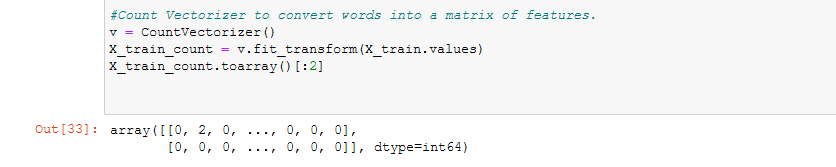
1. Split the dataset, split into x and y variables



Building the text matrix with the each word by using the countvectorizer to represent the words as counts to see the individual features of the text.

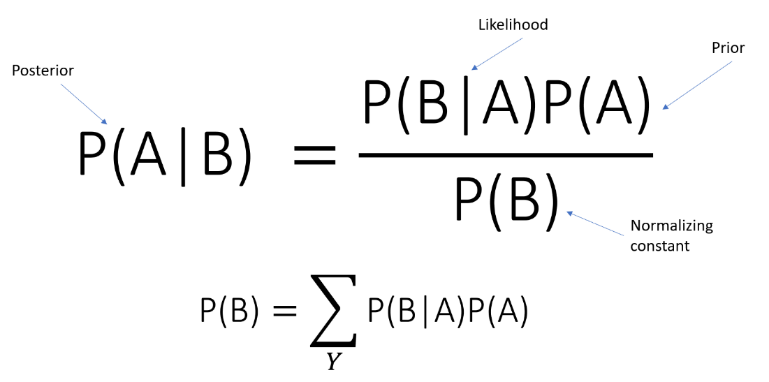


### CountVectorizer to convert words into matrix features

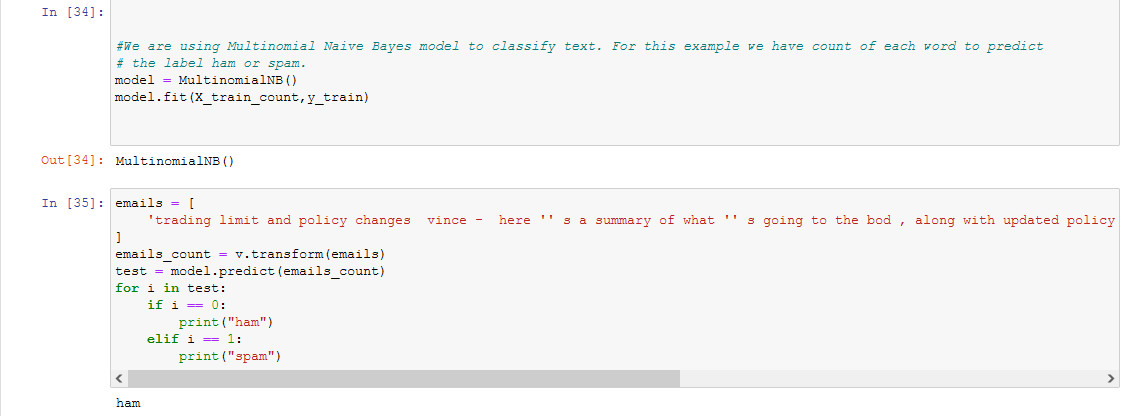


## Multinomial Naïve Bays algorithm to classify text and build the model

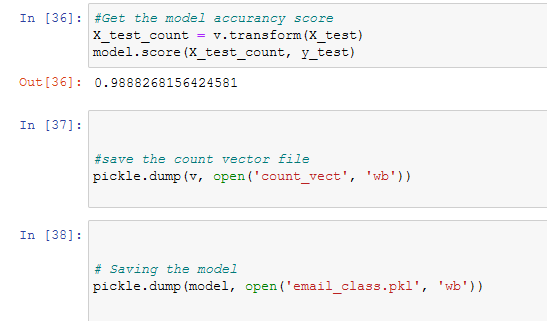
Multinomial naïve bays algorithm used to predict and classify the text either the text is spam and ham. It convert it into the string text into the numeric digit and perform calculations. 1 for spam and 0 for ham



### Results



Naïve bays model accuracy score represent the accuracy level about 98% of spam email detections. Get the model accuracy score and save the countvectorizer file.



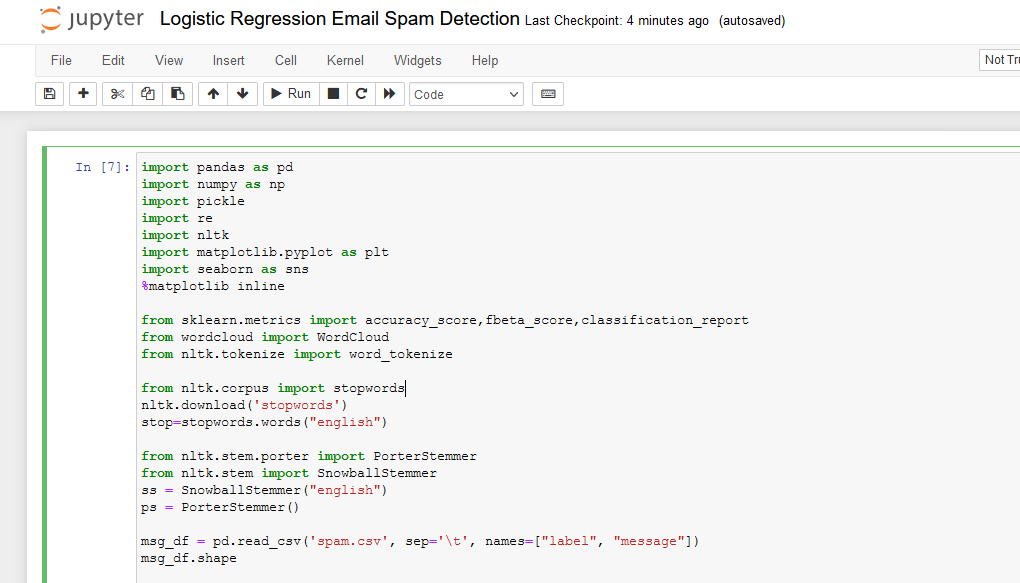
### Analysis

(Kumar, 2018,) Integrated naïve bays spam detection approach has been used in this project the classify the spam email detection the particle swarm email detection approach has been carried out to optimize the good accuracy, dataset clean and stabilize with preprocessing approach, after training and testing the email dataset, multinomial classifier has been used filter the ham and spam emails.

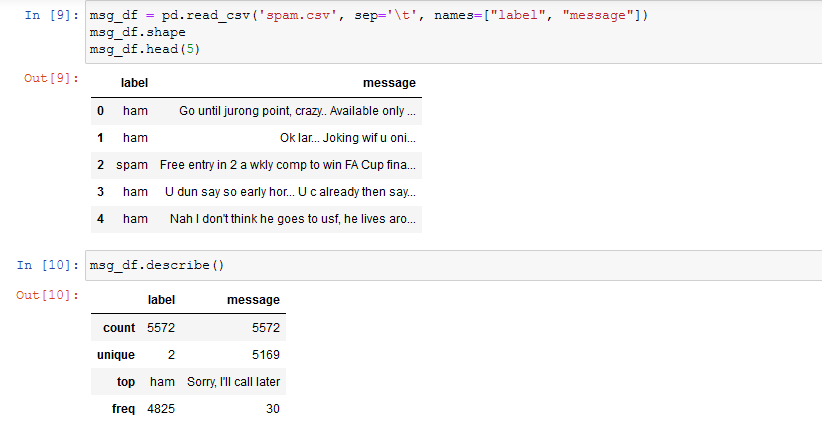
## Machine Learning Logistics Regression to Detect Spam emails

Best model to handle text classification problems, Based on binary output representation Involves probabilities of both outcomes. Process spam training data, and process non spam training data, process spam testing data and build the probabilities model to predict spam and ham emails.

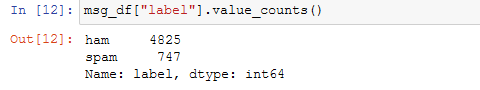
### Data Preprocessing and Importing CSV



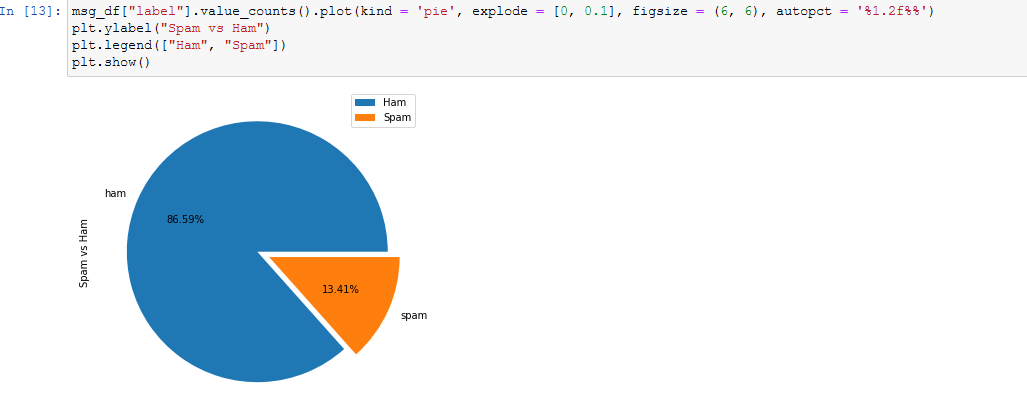
Verify the spam email dataset by visualize the columns values in the dataset. Detect the statistical values of spam email messages.



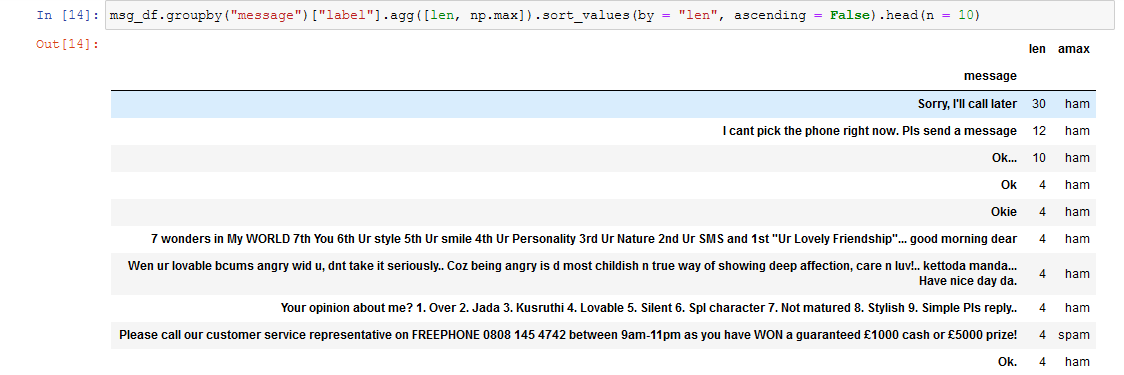
Text values count of spam and ham emails present the actual text wordcount within the dataset.



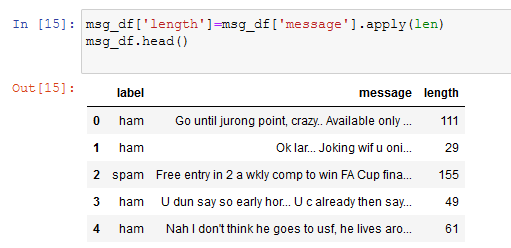
Plot the spam and ham data to evaluate the spam email messages strengths as follows:



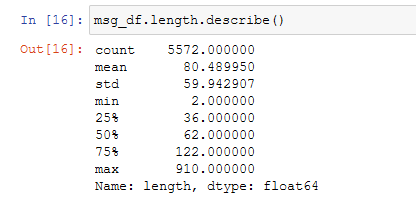
To detect and verify the text length and shapes of the text



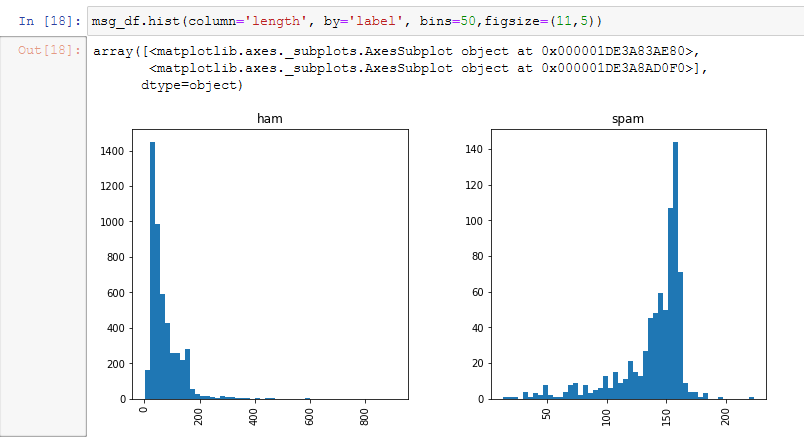
Describing the length of the messages either its spam or ham message



Statistical measurements of the text

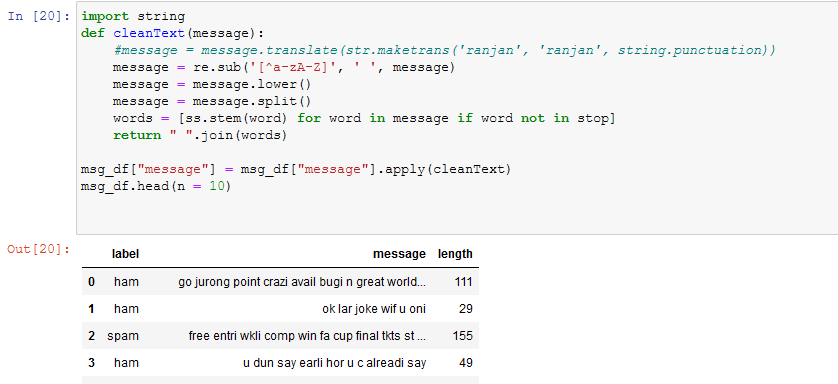


Plot the spam and ham text as follows:



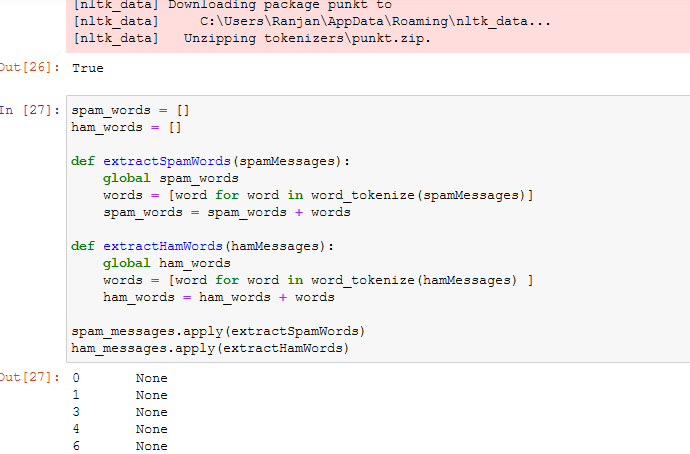
### Data Cleaning & Text Transformation

In data cleaning process using the python jupyter notebook the process remove the unnecessary values from the dataset. It also remove unimportant words, stopwords and stemming

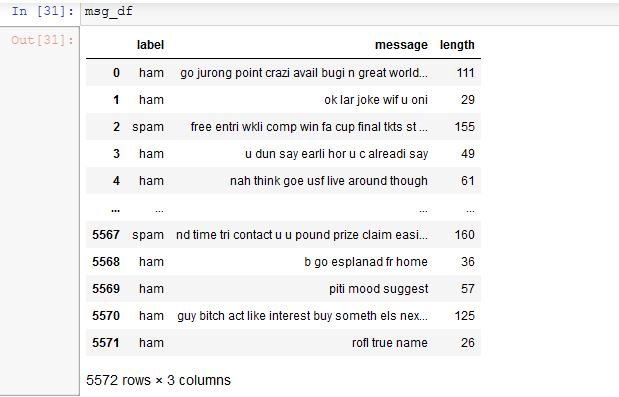


### NLTK library to extract spam words and ham words

The natural language processing toolkit effectively works in python due to extensive and large applications of computer vision and text transformation detection.

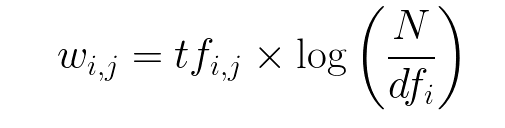


Finally counting the spam and ham messages to cover the various text transformation length and objects of the dataset that present the spam and ham messages detail over the period of stop words and count words after cleaning the dataset.

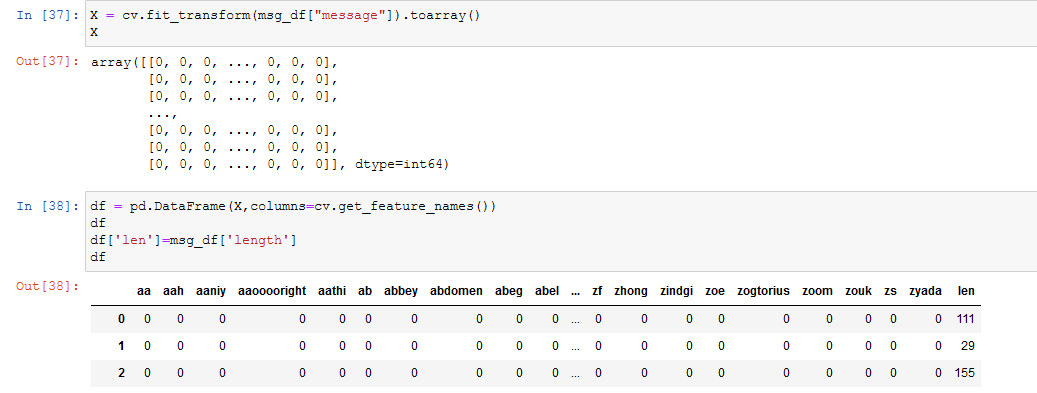


### TFID-IDF Vectorizer

The term frequency appears in document to specify the text on based on word counter and text presentation, this algorithm transform the text & evaluate effectively tf(t), and inverse document frequency is consider as, the idf values normalize the text either it evaluate to 0 and 1 to consider the numeric text and perform various calculations.

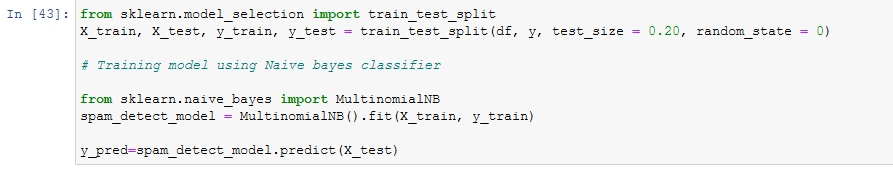


### Transform the string text into numeric array

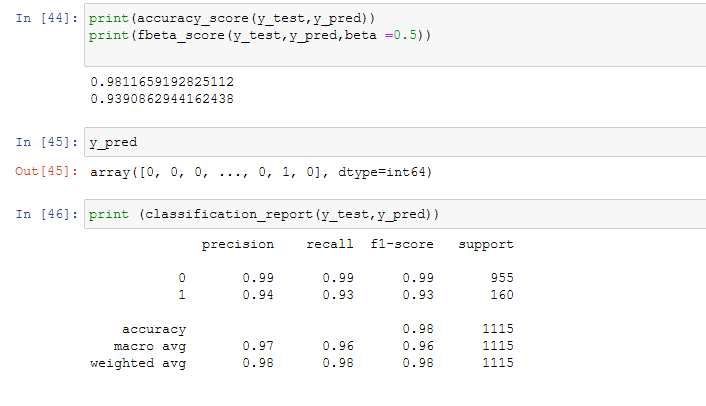


### Train & test the data validation

Training and testing the dataset to divides the each segments values on particular columns values and submit them, training the model for using the naïve bays classifier models and bring the outcomes of spam email detections and ham emails detections after classify the models.



### Outcomes & Results



The above mention spam detection models present the 98% accuracy of the spam emails detections models that used to classify the each segments of text with f1 score and precision recall.

### Analysis

(Bjagrav & Teja, 2020 ) Using natural language processing and Bayesian model to predict the text after transformation of text conversion using the method of TFIDF Vectorizer algorithm, developed three method to test & predict the method of synonym replacement in between the text modellings, technique of ham word injection and spam word spacing effectively called the machine learning models.

# References

Bjagrav & Teja, R., 2020 . Adversarial machine learning for spam filters 38. *ARES '20: Proceedings of the 15th International Conference on Availability, Reliability and Security,* p. 1–6 .

Classification, E. S., 2021. *Email Spam Classification.* [Online]   
Available at: https://www.youtube.com  
[Accessed 22 November 2021].

Freydenberg, M. & Kevin, 2020. From Engagement to Empowermint Project based learning in python coding course. *Computer Information Systems Journal Articles,* 1(1), p. 1.

Github, 2021. *Email Spam Classifier.* [Online]   
Available at: https://github.com/Chando0185/Emial\_Spam\_Classification  
[Accessed 22 November 2021].

Kumar, K. A. a. T., 2018,. Email Spam Detection Using Integrated Approach of Naïve Bayes and Particle Swarm Optimization,. *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), ,* pp. . 685-690.

N. L. Octaviani, E. H. R. C. A. S. a. I. M. S. D. R., 2020,. Comparison of Multinomial Naïve Bayes Classifier, Support Vector Machine, and Recurrent Neural Network to Classify Email Spams,. *2020 International Seminar on Application for Technology of Information and Communication (iSemantic), ,* pp. 17-21.

Nance, J. & Baumgartner, P., 2021. gobbli: A uniform interface to deep learning for text in Python. *The Journal of Open source software,* 6(62), pp. 1-10.

Roy, S. S., Sinha, A. & Roy, R., 2017. Spam Email Detection Using Deep Support Vector Machine, Support Vector Machine and Artificial Neural Network. *International Workshop Soft Computing Applications,* pp. 162-174.

Shahi, T. B. & Yadav, A., 2014. Mobile SMS Spam Filtering for Nepali Text Using Naïve Bayesian and Support Vector Machine. *International Journal of Intelligence Science, ,* Volume 4, pp. , 24-28 .

sklearn.feature\_extraction.text.CountVectorizer, 2021. *sklearn.feature\_extraction.text.CountVectorizer.* [Online]   
Available at: https://scikit-learn.org/stable/modules/generated/sklearn.feature\_extraction.text.CountVectorizer.html  
[Accessed 22 November 2021].