PROJECT

ON

MACHINE LEARNING

Submitted by

Anany Dev Garg

102083011

3CO17

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
THAPAR INSTIUTE OF ENGINEERING AND TECHNOLOGY
PATIALA, INDIA

ACKNOWLEDGEMENT

I would like to place on record my deep sense of gratitude to DR. JOOHI CHAUHAN, FACALTY, Thapar University for his generous guidance, help and useful suggestions.

I also wish to extend my thanks to other workers for guiding and providing the knowledge related to machinery and processes.

Signature of Student

Anany Dev Garg

(102083011)

DECLARATION

I, ANANY DEV GARG, student of Computer Science Engineering (Batch 2019-23), Thapar University, hereby declare that this industrial training report entitled "SPAM/HAM Detection" is a genuine report carried out by me & it is a confide record of work done at Thapar University, located at Patiala, Punjab, under guidance of Dr Joohi Chauhan, Faculty, in Thapar University, towards the partial fulfilment of the degree of Batchers of Engineering, Computer Science Engineering. This project report has not been submitted by anyone to any university for the reward of graduate program in Engineering.

Date: 18/July/2022

(Signature of student)

ANANY DEV GARG

102083011

LIST OF FIGURES

TABLE OF CONTENTS

ABSTRACT

Machine learning is a branch of artificial intelligence concerned with the creation and study of systems that can learn from data. A machine learning system could be trained to distinguish between spam and non-spam (ham) emails. We aim to analyze current methods in machine learning to identify the best techniques to use in content-based spam filtering.

Identifying and fixing the affected machines is the key step to resolve any security threats in a network. Because, it becomes a route to launch several attacks such as Denial of service attacks, spamming, stealing user identities and spreading malware etc .

A definite algorithm in this report is used to differentiate between spam and non-spam. The performance of this tool is based on the parameters like number of spam messages, percentage of spam detected and its efficiency to overcome the limitations of the existing systems.

1. INTRODUCTION

Due to the wide popularity of the internet and its communication with no cost, it

was recognized as the premium tool for advertising and marketing. With respect to economic constraints, most number of people started sending emails to thousands of people across the world. This made internet, a commercial network with the association of electronic mail as one of the quick resources of communication. The major problem in today's internet world is sending bulk or unsolicited emails to numerous users. This adds an additional advantage of launching other attacks and wasting of resources . E-mail spam comes under the electronic spam which sends bulk of unnecessary or junk mail of duplicate emails to the recipients.

1.1 Types of spam:

1.1.1 Email Spam:

Email spam is the most familiar spam that most of the users come across every day. Email spam follows three properties i.e., anonymity, mass mailing and un solicited emails. Anonymity is the property of hiding the uniqueness and whereabouts of the email sender. Mass mailing is defined as the sending of bulk identical emails to the large number of groups and unsolicited emails are the emails transferring to the recipients who do not request. Typically, an email sent to large number of groups without any request by hiding their identity is referred as email spam.

1.1.2 Comment Spam:

This is the most common spam that many users come across in various blogs.

Spammers use the posts in the blog to redirect to spam websites. The ranking of such blogs gets increased gradually in the search engines. It is basically used to promote the searching services like Wikipedia, blogs, guest books etc. There are number of tools in the market to get rid of comment spam.

1.1.3 Instant messenger spam:

It is not as widely spread as other types of spam. Yahoo messenger, My space,

Windows live messenger etc. are the end spots for the spammers. The spammers gather the data of different users and send unsolicited messages within a link that triggers viruses, spam etc.

The best way to get rid of this type of spam is to ignore the messages from the strangers. There is also a possibility to get the links from the existing friends list. A critical measure like verifying the size of the URL will be able to trim the chances of being the victim to this spam.

1.1.4 Junk fax:

Junk faxes are not as prevalent as before. It reduced periodically with the existence of internet technology. However, there are also some risk factors occurring in few corners because of this telemarketing technology. This is similar to junk email where the advertisements and messages are passed to numerous users via fax machines. The adversaries use broadcast fax as a medium to pass on the junk fax to various users. Fortunately, there are surplus tools to overcome junk fax.

1.1.5 Unsolicited text messages:

This is kind of similar to instant messenger spam but here the messages are passed via mobiles. SMS is the service through which the messages are transferred from one user to other user. The easiest way is to maintain the contact with the known friends instead of strangers. It is relatively easy to find the source where the message is coming from with the instant messenger spam. It is critically important no to click on the links that are passed via mobile by the spammers.

1.1.6 Social networking spam:

Social networking sites play an important role in today's world. With the advent of such sites, spammers also started flooding using new techniques to make the social networking sites such as face book, twitter, linked in etc. as part of the spamming activities. As of now it is targeting only the wall posts, messages but these techniques evolve certainly over a period of time.

Spammers use notes or messages through various groups or pass the messages with embedded links, which may lead to pornographic or other sites and target spam . Even though these sites have an option to report spam or abuse activities, the spammers frequently change their address or account to hide their identities.

1.2 Problems with spam:

1.2.1 Viruses:

Viruses are the most dangerous threats across the network. There are many techniques and methodologies developed to decrease the nefarious activities caused by different types of viruses. With the increase in the internet technology, wide variety of viruses produced to attack the machines.

Spam is one of the sources to launch such types of viruses. The widely spread viruses are the ones which disconnect the hosts and get

diffused into the network. Spam viruses in modern technology are more dangerous as it controls the machine itself and then annihilates them. Viruses are not visible and get launched when a particular command is triggered. There are so many techniques used by spammers in order to allow users to click or use the links to launch thousands of spam viruses across the network.

Due to increase in the intensity of spam, it captures the user's email address and passes numerous messages to the customer list, through which it disturbs the customer trust and destroys the system.

1.2.2 Server problems:

Most of the time servers are being targeted by the spammers. Due to increase in the intensity and volume of the spam, the company or any system has to use huge resources to maintain the server. In order to distill and disseminate the data that is transferring in the network more energy costs and resources are to be divided among the departments.

Due to this frequency of spam, the performance also gets affected. So, the servers must maintain a low and necessary data. Otherwise, it can create major problems on the server to maintain and causes heavy load disrupting the entire network.

1.2.3 Hacking and Phishing:

As the computers in the modern technology are becoming more and more secure, the spammers face more difficulty to capture the confidential details. So, they tend to use various methods to break through the security of different IT departments. Spammers make use of hacking methods like entering into the trusted employee system without the user's awareness. Then, spammers perform different activities and keep a record of the confidential data or hold vital information either for the cost or for self-happiness. Another way is to trap the employees of the companies to enter the passwords or any valuable information into the spammer's website, so that it keeps track of the password to reveal important credentials . Though there are many firewalls and spam filters, spammers are also improving their technical skills to intrude on organizations.

1.2.4 Productivity threats:

It is known fact that most number of employees in any organization spends approximately an hour of time to sort out and delete the spam from a cluster of good emails. This leads to heavy wastage of resources like labor cost, time and space in any system. An important email among the cluster of non-spam emails seems to be an unimportant one. This causes problems like loss of e-mail, deleting email and can also disturb the valued customer trust and internal correspondence.

1.2.5 Blank spam emails and forwarding spam emails:

Spammers also use the technique of sending a blank email to the recipients. The purpose of sending this type of email is to recognize whether the recipient possesses a valid email ID or not. If it is an invalid email ID, then it bounces back stating with a non deliverable notice. This helps the spammer to identify whether it is a valid email ID or not. Sometimes blank emails also attach few files that initiate Trojan virus if it is opened in the system. Forwarded emails are again one more cause to initiate spam emails. This makes users forcibly to forward the users in their friends list. As a result it forms a chain and delivers spam emails and becomes uncontrollable. This eats away lots of time and space and costs a lot to filter spam in the system.

1.3 Types of email spam filters:

Spam filter is a piece of software that is used to filter the spam emails based on the content and rules adhered by its corresponding software. Every single spam filter has its own set of rules through which the spam is filtered from spreading across the network. It involves the content of the spam, address of the users and where it is redirecting to etc. Based on these parameters it judges, whether an email is a spam or not. There are multiple spam filters divided based on their rules .

1.3.1 Rule based scan filtering system:

These are referred as the original spam filters. It works on the method of detecting pre-determined words or phrases that most of the spammers use. It identifies those key words and block's emails from passing on from one user to other users. The rules to detect words or phrases are to be improved daily. Because the spammers are so intelligent that it keeps track of words that are blocks and uses its synonyms for a successful transmission. Nevertheless, strict based rules also become another problem, as it blocks even a legitimate email. There is every possibility that both spam and non-spam email get blocked due to the demanding rules passed on by the rule based to scan filtering system.

Suppose there is a word spam involved in the message, it gets blocked by the rules based on the filtering system.

If the rules are weak, then spammer' tries to modify the message from "spam" to "sp@m" and passes on the message successfully and spreads the spam across the network. So, there needs to maintain a different strategy for the trule based scans. Even so, one has to accept the fact that the effects of spam can be reduced, but it's impossible to block hundred percent of spam and let the good email pass through the system.

1.3.2 Bayesian Analysis:

All the methods that were discussed above are based on some predictive methods and content in the message during the exchange of information. However, this filtering system is based on the mathematical formulae through which the email can be determined if it is a spam email or non-spam email. As the black lists take a lot of time to get itself updated, the spammers in the mean while pass on the spam across the network

1.3.3 Challenge-Response spam filter :

This spam filter is a basic filter mechanism that is used to control the spam in the emails. This does not allow any strangers or any pre-approved persons to send an email to the user. In return to the email sent by these pre-approved persons, it asks to validate them in order to pass on the email. The logic behind this strategy is that the pre-approved users do not have time to validate their own email ID from thousands of emails that it might have sent. However, there are numerous problems with respect to this system.

• There are high chances that the spammer uses its fake address in order to validate email address. So therefore, even challenge responses are sent, it can be able to validate itself based on the modified address. So, there will be exchange of messages and can cause spreading of spam due to a spammer validating email address.

• Another major problem is during the online selling or buying products. Whenever a product is purchased, a receipt or the details of that particular product may be sent to personal email through an automated email box. Here the problem arises, causing an automated mail box to verify its own email ID. As this mailbox cannot reply and validate itself, this email is discarded causing disturbance in the online purchase.

• The spam filter itself can create a problem to the other user. This is a case when an email is sent back to an actual person mentioning to authenticate the valid user. The spam filter on the other side may also stop the email that is sent by spam filter from entering into the inbox as it is not sent by the intended recipient.

• This filter requires the user to crack the code whenever an image is to be transferred via email. This perhaps becomes a problem for the users with disabilities. Even though entering the code is a good cause to determine whether it's the actual person or any part of the software, it may be troublesome for the users to break the code every single time just to send an email. So there are possibilities of not sending an email. In the business point of view, this becomes amajor problem as the email is sent to any other competitor avoiding the person containing spam filter as it is difficult to send an email. This filter filters the spam to certain extent, but not completely. Even though it could help in filtering few emails, it has its own drawbacks, which can cause disturbance in the network.

1.3.4 Global blacklists spam filter:

Global black lists contain the list of the notable spammers. Whenever an email is sent, the internet keeps track of the email sender details, i.e., from where the email has come from. Its IP address also gets recorded. So this spam filter compares against the black list containing the notable spammers. If there is a match, it discards the message before reaching it to the recipient. This makes users to escape from the notable spammers. The black list also gets updated so that it can reduce the well-known spammers to perform the spam activity again. This does not waste its space and time by verifying repeatedly with the spammers but perform only one search with just one database thus saving lot of time and space.

There are certain problems with this filtering system. The black list is decided by the users familiar with detecting spam. So before a spammer getting into this list, there happens to transfer thousands of emails to the users. Sometimes, the legitimate person may get into this black list thus getting boycotted completely without any illegitimate activity. And also, complete internet service providers are blocked because of few users involve in spamming activities. This results in disrupting the entire network.

1.3.5 Permission based spam filter:

In this type of filter, user will be given all the permissions whether to send emails to inbox or trash the emails. All the permissions on the emails can be customized by the user. The privileges are created by the user based on the content of the data, header information etc., to allow the transmission of data from one user to another user.

SYSTEM ANALYSIS & DESIGN

2.1 Requirement Specification Hardware Specifications

2.1.1 PC Specifications

1. Operating System – Windows / Linux / macOS

2. RAM – 8 GB – 4 GB RAM might be able to run the application but the system will work very slowly and there might be a need to suppress all other applications on the system.

3. Storage – minimum 1 TB

4. Processor – intel i3 minimum is recommended – If AMD processor is there in the system, then Geny Motion can be installed for creating an emulator / virtual android device on the system.

2.1.2 IDL Specifications

1. Operating System : Software required Python 3.0 or Python 2.0

2. RAM – Minimum 4 GB

2.2 Software Specifications

PYTHON 3.0 OR 2.0

Notepad editor and Microsoft word.

Use of Internet to retrieve the information of Spam and Ham.

Using the concept of Machine learning and Artificial intelligence.

SPAM FILTER TOOL DESIGN:

Proposed system:

This tool is useful in any organization in which servers and clients are connected in a network. In reference to Figure 3, the clients (PC 1.1, PC 1.2 etc.) can communicate either in a network or even outside the network. All outgoing messages and internal messages sent by any client to other machines are routed through their individual servers (server1).

The server scans the email and detects the spam that is being sent by the client. The server receives the email and delivers to its corresponding destination nodes if the email is free of spam. The server can handle any number of clients in a network. It includes two sets of software, one for the server and the other one for the client. A client can send email to any email ID (Gmail, Hotmail, Yahoo etc.). The client uses its mailbox with the client software for sending any emails. The server software detects the spam based on the content spam detection and differentiates between the spam email and the normal email.

3.Architecture

3.1 DFD

Level 0

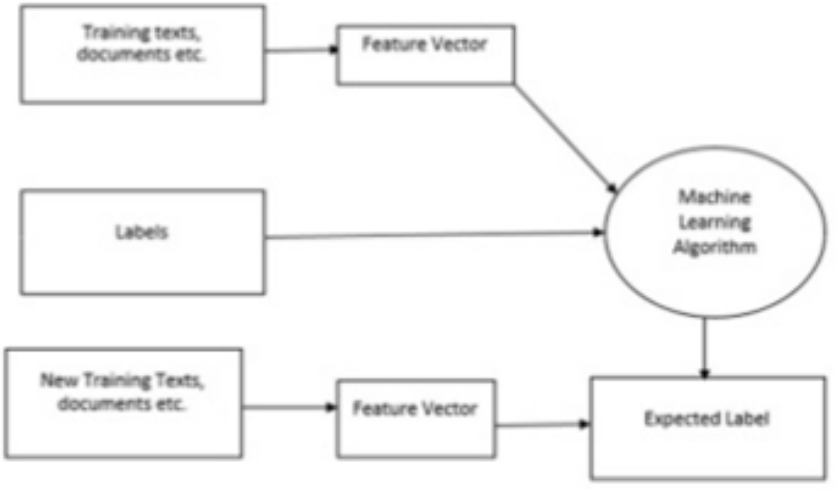


Figure 1 DFD level 0

Level 1

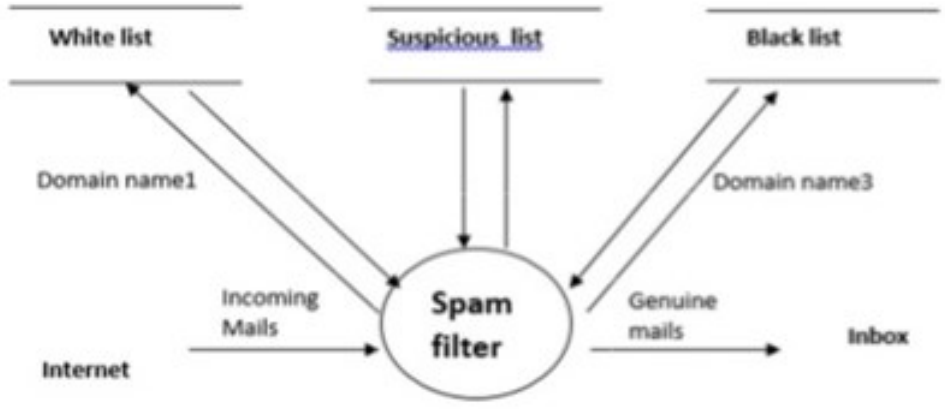


Figure 2 DFD level 1

Level-2

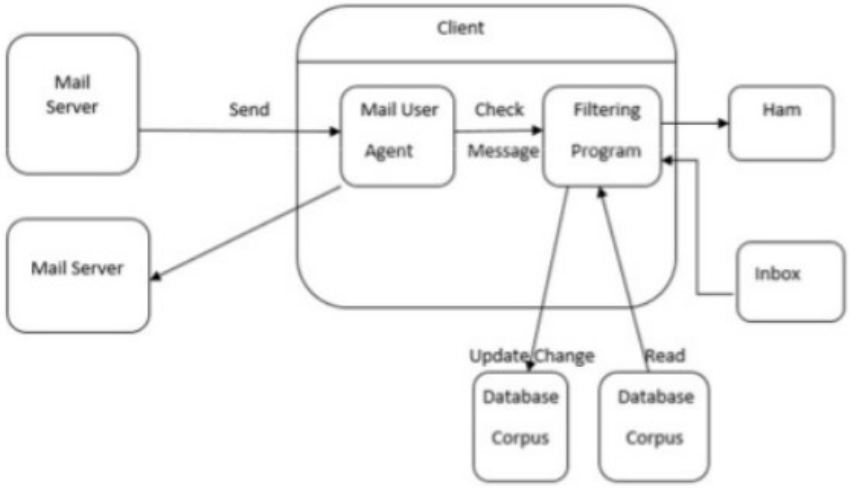


Figure 3 DFD level 2

3.2 UML Digram

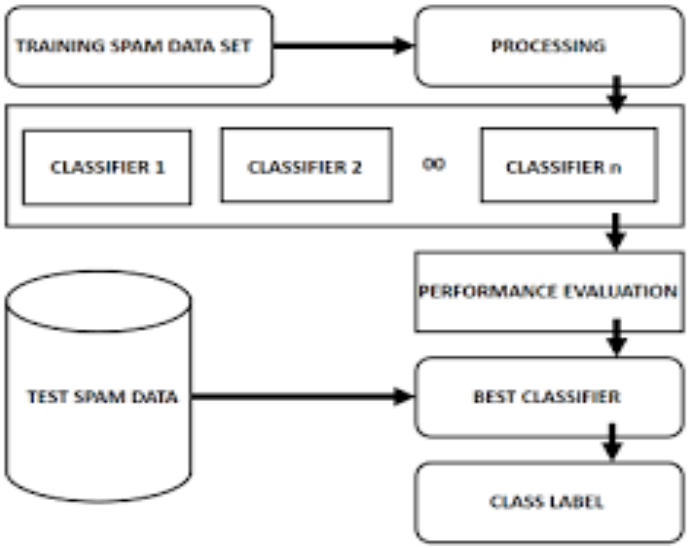


Figure 4 UML Digram

3.3Use-case diagram

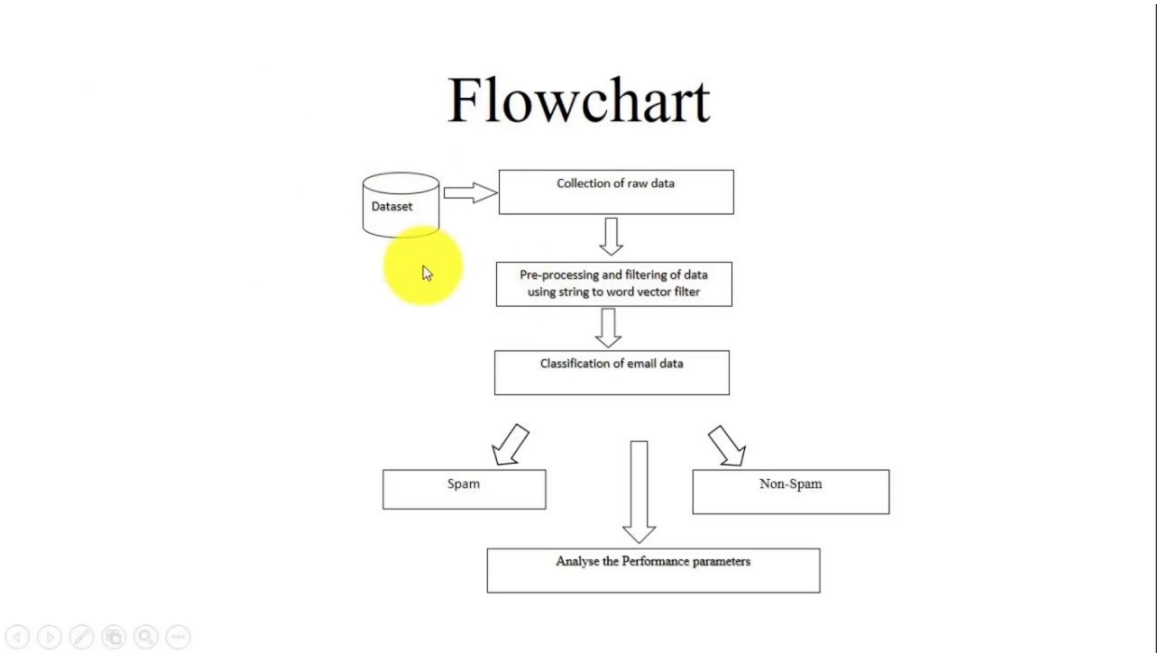


Figure 5 Use-case diagram

4. Project methodology

4.1 Preprocessing

Our classifiers require a feature matrix that consists of the count of each word in each email. As will we discuss, this feature matrix can grow very large, very quickly. Thus, preprocessing the data involves two main steps: creating the feature matrix and reducing the dimensionality of the feature matrix.

4.1.1 Creating the Feature Matrix

1. Remove meaningless words – Meaningless words, known as stop-words, do not provide
meaningful information to the classifier, but they increase dimensionality of feature matrix. In Figure 3, the red boxes outline the stop-words, which should be removed. In addition to many stop-words, we removed words over 12 characters and words less than three characters.

2. Stem - Similar words are converted to their "stem" in order to form a better feature matrix. This allows words with similar meanings to be treated the same. For example, history, histories, historic will be considered same word in the feature matrix. Each stem is placed into our "bag of words", which is just a list of every stem used in the dataset. In Figure 3, the tokens in blue circle are converted to their stems.

3. Create feature matrix - After creating the "bag of words" from all of the stems, we create a feature matrix. The feature matrix is created such that the entry in row i and column j is the number of times that token j occurs in email i.
These steps are completed using a Python NLTK (Natural Language Toolkit).

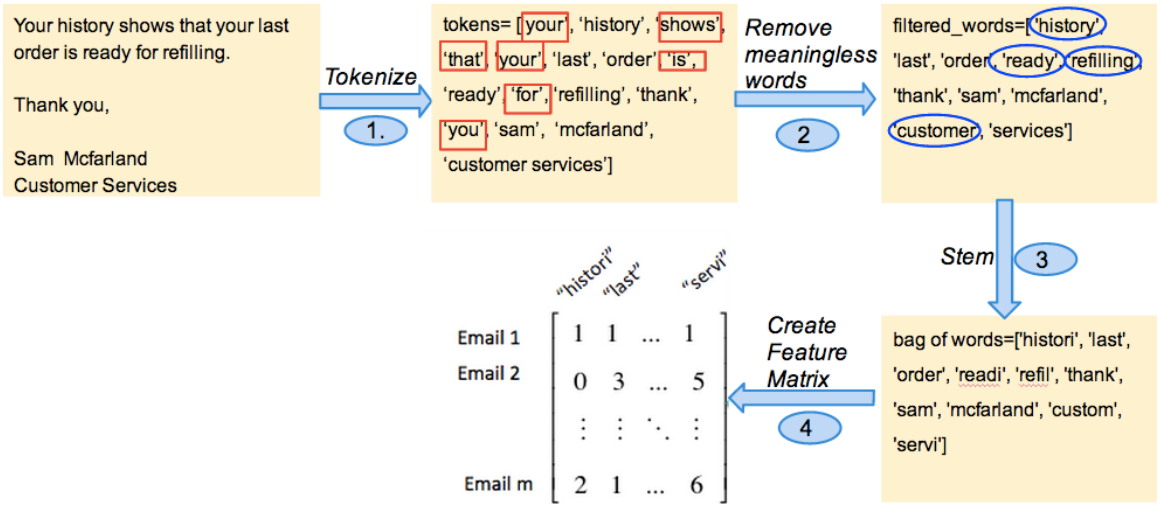


Figure 6. A summary of the preprocessing involved in creating a feature matrix.

4.1.2 Reducing the Dimensionality

The bag of word method creates a highly dimensional feature matrix and this matrix grows quickly with respect to the number of emails considered. A highly dimensional feature matrixgreatly slows the runtime of our algorithms.
When using 50 emails, there are roughly 8,700 features, but this number quickly grows to over 45,000 features when considering 300 emails (as shown in Figure 4). Thus, it is necessary to reduce the dimensionality of the feature matrix.

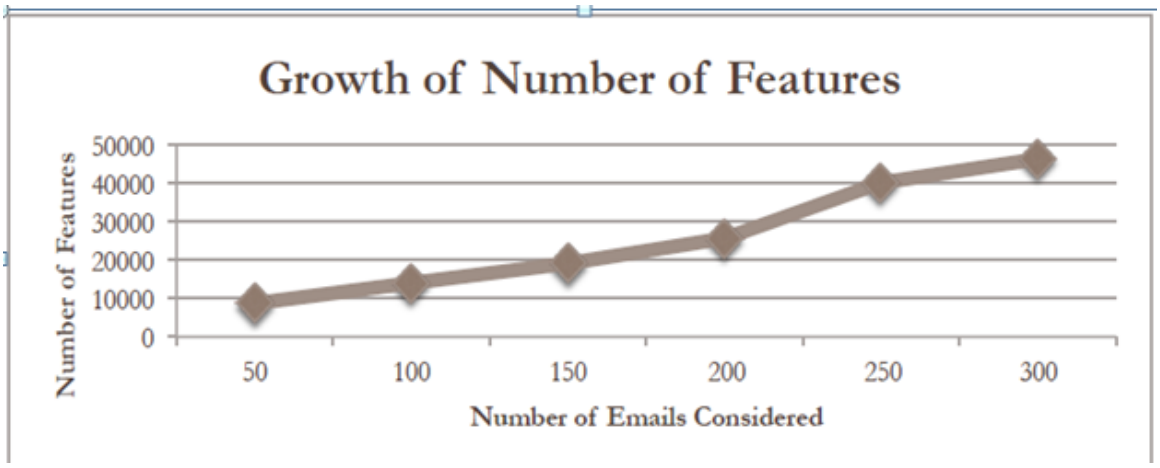


Figure 7.The growth of number of features with respect to the number of emails considered.

To reduce the dimensionality, we implemented a hash table to group features together. Each stem in the bag of words comes with a built-in hash index in Python. We can then decide how many hash buckets (or features) we would like to have. We take the built-in hash index mod the bucket size to find the new hashed index. This process is shown in Figure 5.

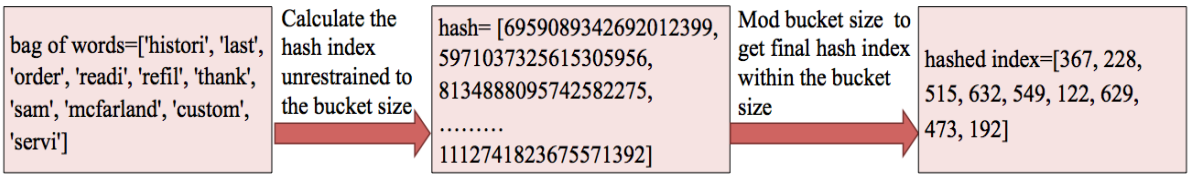


Figure 8. The growth of number of features with respect to the number of emails considered.

4.2 Classifiers

As mentioned before, a machine learning system can be trained to classify emails as spam or ham. To classify emails, the machine learning system must use some criteria to make its decision. The different algorithms that we describe below are different ways of deciding how to make the spam or ham classification.

4.2.1 Naïve Bayesian

The Naive Bayesian classifier takes its roots in the famous Bayes Theorem:

$$P(H|e) = \frac{P(e|H)P(H)}{P(e)}$$

Bayes Theorem essentially describes how much we should adjust the probability that our hypothesis (H) will occur, given some new evidence (e). For our project, we want to determine the probability that an email is spam, given the evidence of the email's features (F1,F2,...Fn). These features F1, F2,...Fn are just a boolean value (0 or 1) describing whether or not the stem corresponding to F1 through Fn appears in the email. Then, we compare P(Spam| F1,F2,...Fn) to P(Ham| F1,F2,...Fn) and determine which is more likely. Spam and ham are considered the classes, which are represented in the equations below as "C". We calculate these probabilities using the following equation:

$$p(C \mid F_1,\ldots,F_n) = \frac{p(C)p(F_1,\ldots,F_n \mid C)}{p(F_1,\ldots,F_n)}$$

Equation 2. Equation rooted in Bayes Theorem for determining whether an email is spam or ham given its features.

5. Project Working:

5.1. Preparing the text data.

The dataset used here, is split into a training set and a test set containing 702 mails and 260 mails respectively, divided equally between spam and ham mails. You will easily recognize spam mails as it contains *spmsg* in its filename.
In any text mining problem, text cleaning is the first step where we remove those words from the document which may not contribute to the information we want to extract. Emails may contain a lot of undesirable characters like punctuation marks, stop words, digits, etc which may not be helpful in detecting the spam email. The emails in Ling-spam corpus have been already preprocessed in the following ways:

a) Removal of stop words – Stop words like “and”, “the”, “of”, etc are very common in all English sentences and are not very meaningful in deciding spam or legitimate status, so these words have been removed from the emails.

b) Lemmatization – It is the process of grouping together the different inflected forms of a word so they can be analyzed as a single item. For example, “include”, “includes,” and “included” would all be represented as “include”. The context of the sentence is also preserved in lemmatization as opposed to stemming (another buzz word in text mining which does not consider meaning of the sentence).

We still need to remove the non-words like punctuation marks or special characters from the mail documents. There are several ways to do it. Here, we will remove such words after creating a dictionary, which is a very convenient method to do so since when you have a dictionary, you need to remove every such word only once. So, cheers!! As of now you don’t need to do anything.

5.2. Creating word dictionary

It can be seen that the first line of the mail is subject, and the 3rd line contains the body of the email. We will only perform text analytics on the content to detect the spam mails. As a first step, we need to create a dictionary of words and their frequency. For this task, training set of 700 mails is utilized. This python function creates the dictionary for you.

Dictionary can be seen by the command print dictionary. You may find some absurd word counts to be high but don’t worry, it’s just a dictionary and you always have the scope of improving it later. If you are following this blog with provided data-set, make sure your dictionary has some of the entries given below as most frequent words. Here I have chosen 3000 most frequently used words in the dictionary

5. Feature extraction process.

Once the dictionary is ready, we can extract word count vector (our feature here) of 3000 dimensions for each email of training set. Each **word count vector** contains the frequency of 3000 words in the training file. Of course, you might have guessed by now that most of them will be zero. Let us take an example. Suppose we have 500 words in our dictionary. Each word count vector contains the frequency of 500 dictionary words in the training file. Suppose text in training file was "Get the work done, work done" then it willbe encoded as [0,0,0,0,0,…….0,0,2,0,0,0,……,0,0,1,0,0,…0,0,1,0,0,……2,0,0,0,0,0].
Here, all the word counts are placed at 296th, 359th, 415th, 495th index of 500 length word count vector and the rest are zero.
The below python code will generate a feature vector matrix whose rows denote 700 files of training set and columns denote 3000 words of dictionary. The value at index '*ij*' will be the number of occurrences of j^{th} word of dictionary in i^{th} file.

5.4. Training the classifiers.

Here, I will be using scikit-learn ML library for training classifiers. It is an open source python ML library which comes bundled in 3rd party distribution anaconda or can be used by separate installation following this. Once installed, we only need to import it in our program.
I have trained two models here namely Naive Bayes classifier and Support Vector Machines (SVM). Naive Bayes classifier is a conventional and very popular method for document classification problem. It is a supervised probabilistic classifier based on Bayes theorem assuming independence between every pair of features. SVMs are supervised binary classifiers which are very effective when you have higher number of features. The goal of SVM is to separate some subset of training data from rest called the support vectors (boundary of separating hyper-plane). The decision function of SVM model that predicts the class of the test data is based on support vectors and makes use of a kernel trick.

Once the classifiers are trained, we can check the performance of the models on test-set. We extract word count vector for each mail in test-set and predict its class(ham or spam) with the trained NB classifier and SVM model. Below is the full code for spam filtering application. You have to include the two functions we have defined before in step 2 and step3

Checking Performance and its screenshots

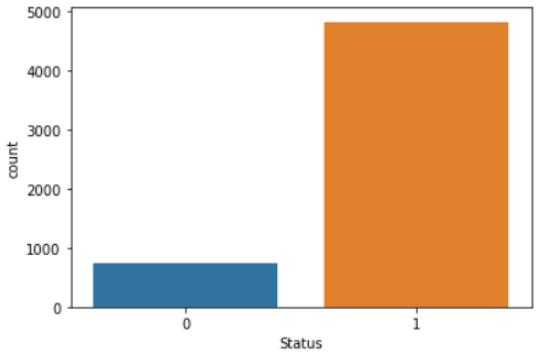
Test-set contains 130 spam and 130 non-spam emails. If you have come so far, you will find below results.

```
In [3]: df.head() # Changes done by the command in status column
```

Out[3]:

	Status	Message
0	1	Go until jurong point, crazy.. Available only ...
1	1	Ok lar... Joking wif u oni...
2	0	Free entry in 2 a wkly comp to win FA Cup fina...
3	1	U dun say so early hor... U c already then say...
4	1	Nah I don't think he goes to usf, he lives aro...

```
In [4]: sns.countplot(x="Status",data=df) # Displays the number of spam and non spam messages in the dataset
Out[4]: <AxesSubplot:xlabel='Status', ylabel='count'>
```




```
In [18]: accuracy_score(y_test,pred) # TP+TN/Total  fusiion matrix
Out[18]: 0.979372197309417
```

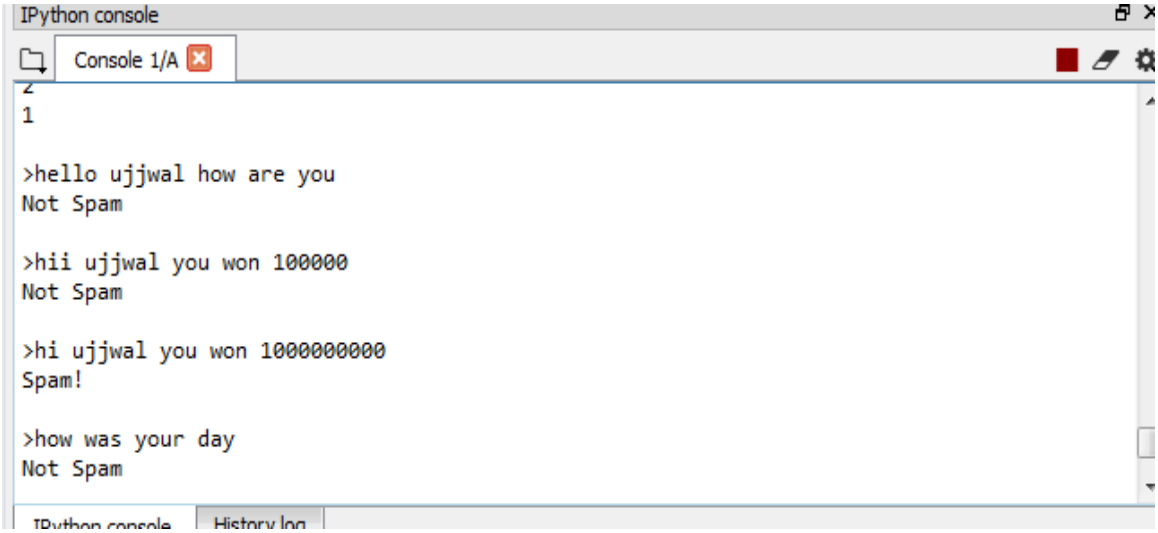


Figure 9. Final output of spam or not spam

6.Code

```
import seaborn as sns
import matplotlib.pyplot as plt
df=pd.read_csv("emails",sep="\t",names=["Status","Message"])
df.loc[df["Status"]=="ham","Status"]=1
df.loc[df["Status"]=="spam","Status"]=0
df.head()
sns.countplot(x="Status",data=df)
from sklearn.feature_extraction.text import CountVectorizer
cv=CountVectorizer()
msg=cv.fit_transform(df["Message"])
msg.shape
x=msg
y=df.Status
y=y.astype('int')
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2)
from sklearn.naive_bayes import MultinomialNB
model = MultinomialNB()
model.fit(x_train,y_train)
pred=model.predict(x_test)
from sklearn.metrics import accuracy_score
accuracy_score(y_test,pred)
m = cv.transform(["Hello how are free you"])
model.predict(m)
```

7.Conclusion and Future Scope

7.1 Conclusion

We found that the One-Nearest Neighbor algorithm outperformed the other classifiers. This simple algorithm achieved great performance and was easy to implement. However, we believe that the performance of this algorithm could still be improved. We encourage those who wish to further this research to investigate the effects of a weighted majority vote, enhanced feature selection, and different distance measures.

Another key finding was that we discovered that the recall for all of the algorithms was very high, while the precision was lower. This suggests that our algorithms are very liberal in labeling an email as spam. To combat this, perhaps mapping the features to a higher dimension, as is done in Support Vector Machine algorithms, would be a solution to this problem.

After analysis, we believe that a machine learning approach to spam filtering is a viable and effective method to supplement current spam detection techniques.