Spam Email Filtering with Machine Learning Algorithms

RQ: Which of the two algorithms is more efficient, Naïve Bayes or Multilayer Perceptron Classifier for spam email detection based on the contents?

Abstract

Spam emails have been and continue to be a significant problem in email communication. Detecting and filtering spam mails is of highest priority for a better productivity for individuals as well as corporates. The traditional filters looks for specific words or combination of words to filter the spam which is not effective anymore as the spammers continuously upgrade themselves to defeat the purpose of these static filters. In this essay, we are looking at two dynamic filters which continuously upgrade themselves so that better spam detection takes place. Two algorithms are compared, Naïve Bayes and Multilayer Perceptron Classifier (MLP) to find out which is more efficient for content based spam filtering.

The study discusses the methodologies adopted by the two algorithms and outlines their efficacy and efficiency in classifying the mails as spam and legitimate, The two algorithms use data mining approach to distinguish spam from legitimate emails. An experiment is conducted with training data sets with appropriate feature categories. The experiments are conducted with data available in public forum and WEKA data mining software is used to run the experiment..

The two algorithms are compared for spam filtering using two methodologies on the criteria of Accuracy, Spam Precision and Spam Recall. The results indicate that the Naïve Bayes algorithm scores higher than the MLP.

Table of Contents

1. Introduction

With the advent of internet and networking technologies, the communication across the Globe has become much easier and effective, Much of this communication happens through email. Emails facilitate the connection with people and sharing of knowledge. The bad effect of this development is that some entities try to push unwanted and dangerous information to many email IDs which breeds inefficiency and sometimes result in frauds. These undesirable emails are generally called Spams and any effort or tools to differentiate Spams from legitimate emails go a long way to make the communication more effective and secure. Apart from the unsolicited and potentially dangerous information spread through the Spam emails, the receiver spends valuable time and energy in identifying these mails and discarding them. These mails use vast amount of network bandwidth and the limited inbox space. Some studies have shown that more than 50% of the business emails are Spams and the internet users are spending billions of dollars as connection charges only to receive these mails.

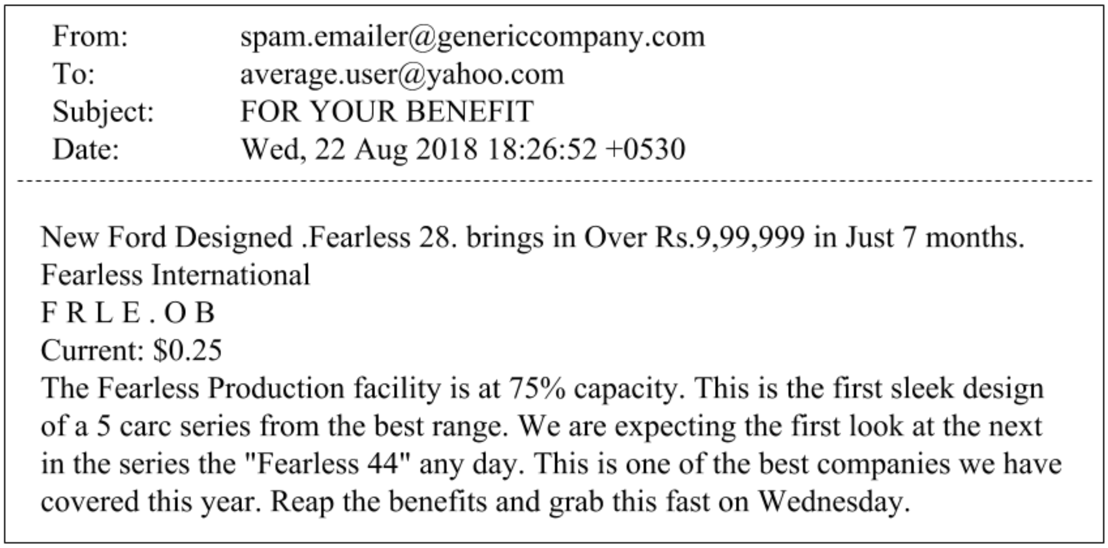
Blacklisting and Whitelisting are two commonly used methods for spam filtering. In the first case, the tool checks for known spam sources and words which are normally found in spam mails. These mails are rejected as spam mails. In the second case the mails from trusted correspondents are accepted as genuine mails. The problem is that the spammers keep themselves updated and adapt their design and content to the legitimate ones.

Dynamic methods are much more effective in spam filtering as compared to the static methods described above. These methods use a database of words or phrases which are normally found in spam mails and the detection is done based on these. These methods use machine learning techniques to understand and update their words and phrases normally found in spam mails and this database is used on fresh set of mails to classify them as spam or legitimate. This is dynamic because the database is updated with words and phrases from those mails which are already classified as spam and this process is a continuous one.

Two different techniques are to be mentioned here in this context. The first one, Naïve Bayes Algorithm consider the words independently and combinations are not considered. For example, it does not consider that the combination of the word discount and offer happens more in a spam mail than in a legitimate one. The second method, Multilayer Perceptron Classifier Neural Network (MLP-NN) , performs better in some cases but is disadvantaged in terms of time taken for parameter selection and training. This leads to the research question "Which **of the two algorithms is more efficient, Naïve Bayes or Multilayer Perceptron Classifier for content-based spam email detection?"**

2. Spam Filtering based on Contents

The filtering based on contents is of prime importance in differentiating a spam from a legitimate email. As you can see from fig 1, the standard email has two parts, the header and the body. The filtering is carried out after receiving the mail using the words used in the subject line and those used in the body. The header stores all information about the particular mail such as the sender and all those who have received a copy of the mail. The header also contains the return path which is the address to which a reply to the mail will go and it could be different from the address of the sender.



From: spam.emailer@genericcompany.com
To: average.user@yahoo.com
Subject: FOR YOUR BENEFIT
Date: Wed, 22 Aug 2018 18:26:52 +0530

New Ford Designed .Fearless 28. brings in Over Rs.9,99,999 in Just 7 months.
Fearless International
FRLE.OB
Current: $0.25
The Fearless Production facility is at 75% capacity. This is the first sleek design of a 5 carc series from the best range. We are expecting the first look at the next in the series the "Fearless 44" any day. This is one of the best companies we have covered this year. Reap the benefits and grab this fast on Wednesday.

Fig. 1. Sample Spam Email

The email undergoes a pre – treatment process where in all the unnecessary structured data is removed from the contents. What remains should only be the sender address, subject and the body of the email. Text extraction happens next in which activities such as converting words to lower case, using root words, separation of empty blanks etc. takes place. These are done by specific feature extraction programs. Fig 2 shows the flow chart of the process[1].

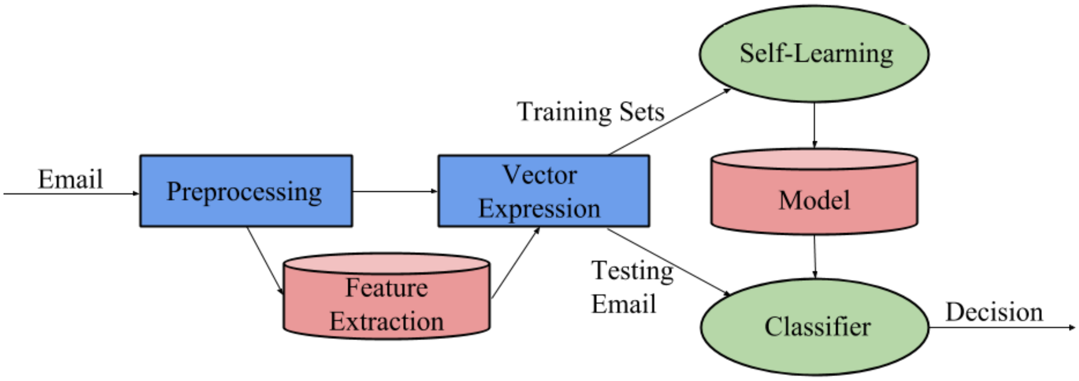


Fig.2. Spam Filtering Process

[1] S. Catarina, R. Bernardete, RVM ensemble for text classification, *International Journal of Computational Intelligence Research*

The flowchart shows the entire process of classification to Spam and legitimate emails. The conversion process starts after the initial processing of the entire collection. There are five stages – first level transformation, interface for the user, selection of features –its extraction, classification of data and the final analysis. Usage of Machine learning algorithms are used to separate spam emails from legitimate ones.

3. Scope of the Essay

This study is done to compare two machine learning techniques in its objective to identify and separate spam emails from a dataset of emails. The two methods used are the Naïve Bayes algorithm and Multilayer Perceptron Classifier Neural Network (MLP-NN). The investigation focusses on pattern classification of email content for the filtering process.

The tools used train itself with the patterns of a spam email from the samples provided and the trained tools are used as a filter to separate the legitimate emails from the collection.

The set of emails used for training contains mails from multiple categories such as business communication, advertisements, discounted promotion offers, mails with malicious intent etc.

In brief the two objectives of the study are

1. To implement the ideas of the two specified methods for spam filtering
2. Two methods are evaluated based on its contents, that is, keywords and its statistical analysis.

[2]ANN model's architecture and learning algorithms will be investigated and the trained network will be used in the testing phase.

[2] https://www.researchgate.net

4. Techniques for Filtering Spam using Machine Learning

It is worthwhile to note that the spam detection can also be done without the use of machine learning algorithms in which case the patterns are to be hard coded by the programmer. In the case of machine learning the network is trained with test data and this trained network is used for separating spam ones from legitimate. At the base level the network gain patterns from test data and finds the relevance and patterns for spam mails and this information is used to identify spam mails in the collection. This is much more efficient and effective since the patterns recognized by the network is dynamic and keeps in line with the latest trends which the spammers might use. The two algorithms used in this essay fall under the category of supervised algorith

ms.

4.1 Naïve Bayesian Algorithm

As explained before, the Naïve Bayesian algorithm filters emails by scrutinizing the contents in a message. The database is updated continuously with the test data to understand which are the words used in spam emails.

When a new email enters the mail box, the probability of it being a spam is arrived at using the benchmarking database.

Consider a Feature Vector (A feature vector a vector that carries knowledge about an entity's main characteristics.)

$M = \{m_1, m_2, m_3 ... m_s\}$, the attribute values are $M_1, M_2, M_3 ... M_s$ where s indicates the number of attributes in M.

Assuming M to represent the type of email, for example $M \{spam, \ legitimate\}$. The conditional

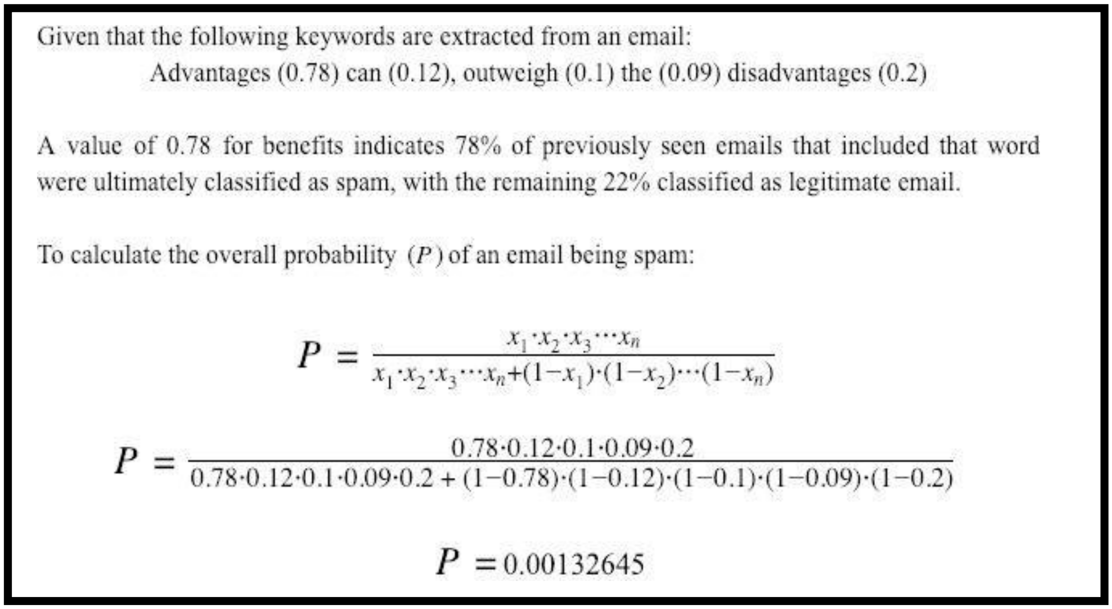
probability in the form $P\ (M_i\ \ | N)$ is calculated using a discriminant function.

Thus $P\ (M_i\ | N)$ represents the probability that N belongs to class M_i.

$$\mathbf{P(M_i|N) = \ P(M_i)xP(N|M_i)/P(N)}$$

$P\ (M_i)$ is the probability of i occurring.

P(N|Mi) indicate the probability of M_i . $P\ (N)$ is the probability of the inputted value, independent of the values. A typical illustration of this form of the Naïve Bayesian Algorithm at work is detailed below.



Given that the following keywords are extracted from an email:

Advantages (0.78) can (0.12), outweigh (0.1) the (0.09) disadvantages (0.2)

A value of 0.78 for benefits indicates 78% of previously seen emails that included that word were ultimately classified as spam, with the remaining 22% classified as legitimate email.

To calculate the overall probability (P) of an email being spam:

$$P = \frac{x_1 \cdot x_2 \cdot x_3 \cdots x_n}{x_1 \cdot x_2 \cdot x_3 \cdots x_n + (1-x_1)\cdot(1-x_2)\cdots(1-x_n)}$$

$$P = \frac{0.78 \cdot 0.12 \cdot 0.1 \cdot 0.09 \cdot 0.2}{0.78 \cdot 0.12 \cdot 0.1 \cdot 0.09 \cdot 0.2 + (1-0.78)\cdot(1-0.12)\cdot(1-0.1)\cdot(1-0.09)\cdot(1-0.2)}$$

$$P = 0.00132645$$

The value obtained indicates that the given email is legitimate and not a spam. But the final decision is on the parameter boundaries of the decision filter.

4.2 Multilayer Perceptron Classifier

The MLP (Multilayer Perceptron Classifier Neural Network) is a normal network which is not linear and works on feed-forward principle and has a sigmoid activation function

$$g(x) = \frac{1}{1+e^{-x}}$$

Fig 3 below represents the two layers, hidden layer and the output layer. The output produced by the above function is in the range [0,1]. The input represents every word contained in the spam email , and an instance of both the layers and one final output neuron is produced.

A bias neuron with a standard continued basis of 1 is contained in the input and hidden layer. When MLP-NN process an email, all the inputs related to the contents of the trained set are set to 1 while the others are set to 0. The Spam emails are those whose value is above 0.5. During the training process the required final output is set to 0.1 and 0,9 respectively for legitimate and spam emails.

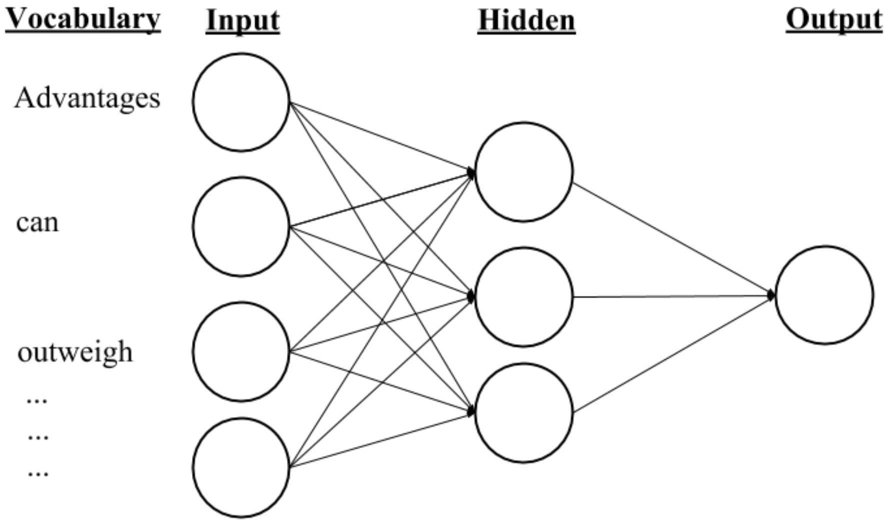


Fig. 3. Structure of a Multilayer Perceptron Classifier Neural Network

During the preparation of training set the values for filtering spam and legitimate should not be very much different. This can give erroneous confusing results. An optimization technique called gradient descent algorithm is used to optimize the weights in the network.

5. Investigation

5.1 Setting up the Test

Experiments were conducted to do investigation for the research question. TREC 07 data and UCI repository data sets were used in the experiments. Both these are publicly available and found online. The machine learning and datamining tools used were WEKA 3.6.0. This software is developed in Java at the University of Waikato. The system was configured to run this software and I ensured that the same system[3] is used for all the experiments.

5.2 Source of Data

While there are several well known libraries[4] of legitimate and spam emails, I chose the two libraries mentioned above because of its widespread acceptance and reputation.

TREC 07 dataset have 5000 mails with the spam rate of 38.03% while UCI Repository data sets have a spam rate of 37.04%. These data sets were well known for its utility in Machine learning applications and are from genuine and reliable sources. They are popular for supplying data sets for machine learning in several domains..

[3] The System had a configuration of Core2 duo 2 GHz CPU, 4GB memory and Windows 7.

[4] D.A. Karrasb, V. Zorkadisa, Panayotou M, Efficient information theoretic strategies for classifier combination, feature extraction and performance evaluation in improving false positives and false negatives for spam e-mail filtering, Pages 799-807

5.3 Initial treatment of Data

Preprocessing is the next step where the conversion of email messages to a template suitable for filtering algorithms is done. This was done by using WEKA.[5]

WEKA was used due to its capability in text categorization. The software selects unique terms representing numerals, alphabets , other symbols in the trial collection which are individually treated. To facilitate feature selection, most important words are selected. A document is generated representing each unit which contains a normalized value for each according to its relevance.

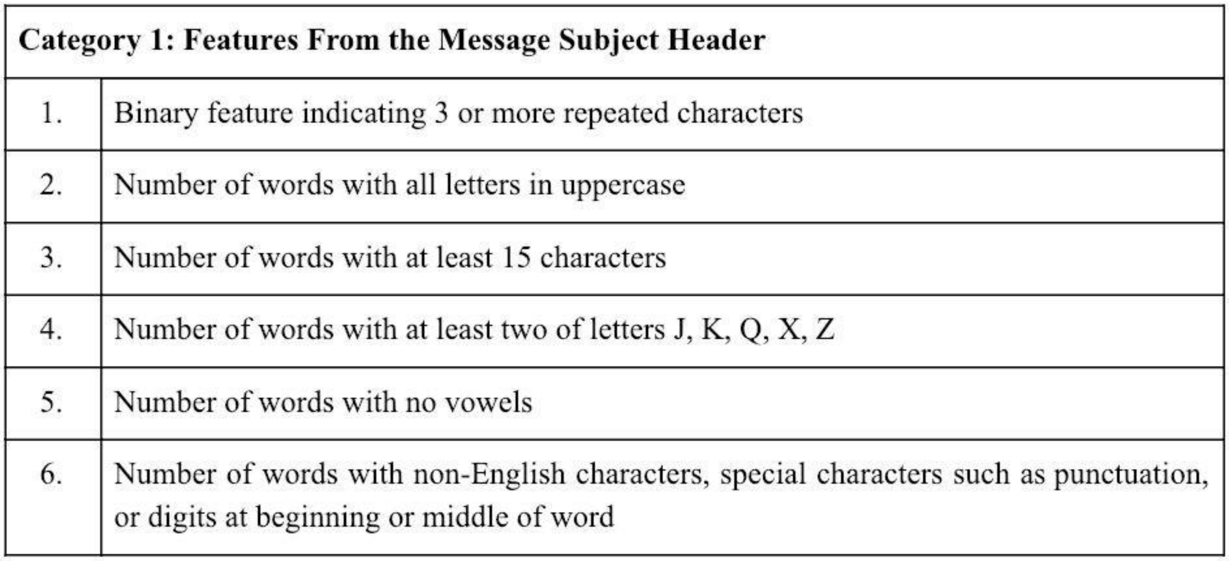
5.4 Selection Of Features

Since we are comparing two methods here, the selected features must be common for both. These features are to be predetermined to get accurate results. Emails are converted to text files before it enters the mail box and the body elements and the header elements are separated.

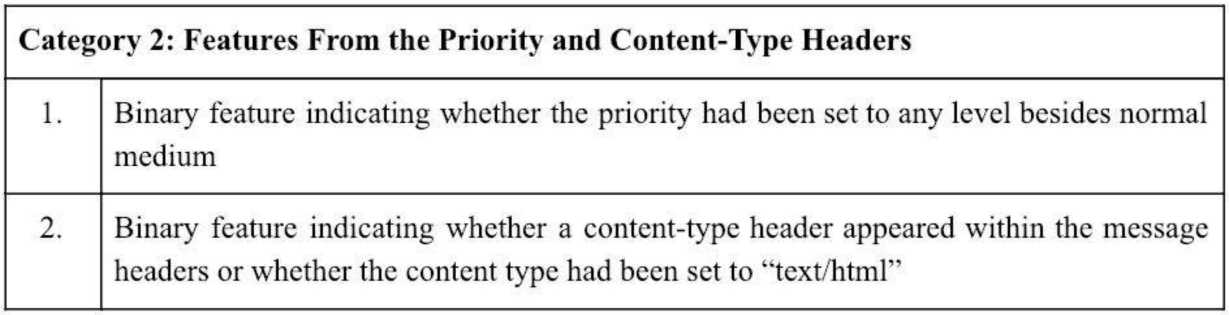
Every individual component of the email separated by a white space is considered as an individual component which could be any alpha numeric combination or symbols. These are further grouped into sets as shown below.[6]

[5] Clark I. Koprinska, J. Poon, A neural network based approach to automated e-mail classification, in:

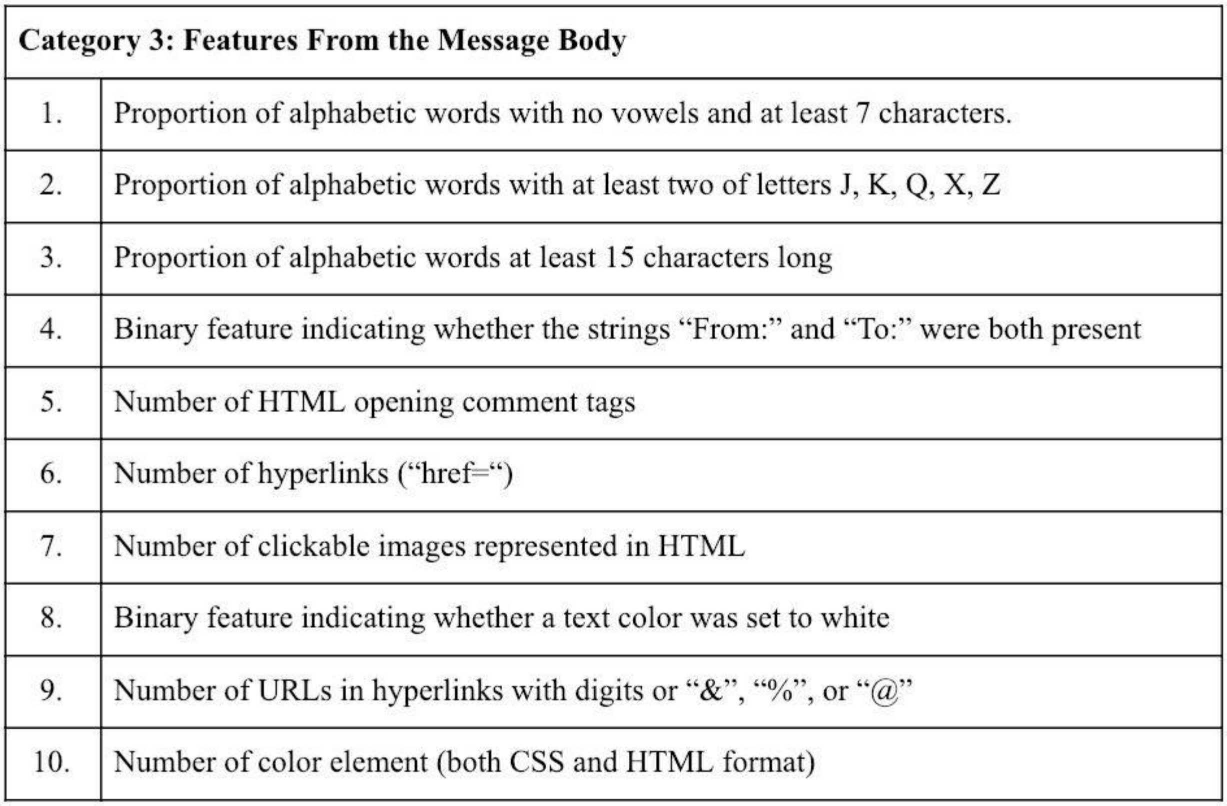
[6] Islam, Saiful, et al. "Modeling Spammer Behavior: Artificial Neural Network vs. Naïve Bayesian Classifier." *Artificial Neural Networks - Application*, 2011.



Category 1: Features From the Message Subject Header	
1.	Binary feature indicating 3 or more repeated characters
2.	Number of words with all letters in uppercase
3.	Number of words with at least 15 characters
4.	Number of words with at least two of letters J, K, Q, X, Z
5.	Number of words with no vowels
6.	Number of words with non-English characters, special characters such as punctuation, or digits at beginning or middle of word

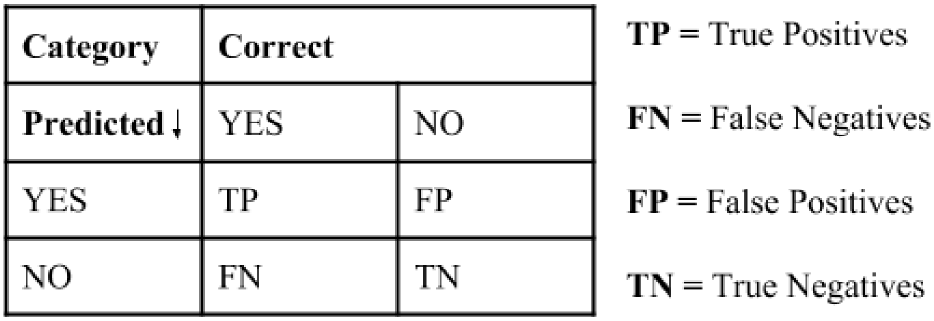


Category 2: Features From the Priority and Content-Type Headers	
1.	Binary feature indicating whether the priority had been set to any level besides normal medium
2.	Binary feature indicating whether a content-type header appeared within the message headers or whether the content type had been set to “text/html”



Category 3: Features From the Message Body	
1.	Proportion of alphabetic words with no vowels and at least 7 characters.
2.	Proportion of alphabetic words with at least two of letters J, K, Q, X, Z
3.	Proportion of alphabetic words at least 15 characters long
4.	Binary feature indicating whether the strings “From:” and “To:” were both present
5.	Number of HTML opening comment tags
6.	Number of hyperlinks (“href=“)
7.	Number of clickable images represented in HTML
8.	Binary feature indicating whether a text color was set to white
9.	Number of URLs in hyperlinks with digits or “&”, “%”, or “@”
10.	Number of color element (both CSS and HTML format)

We check through our investigation whether the features listed above are suitable for classifying the mails as legitimate and spam. The testing is done with both the methods and the results are compared in terms of the parameters defined above, namely, Accuracy, Precision and Recall . The usage is defined as per the matrix in Fig 4.



Category	**Correct**	
Predicted ↓	YES	NO
YES	TP	FP
NO	FN	TN

TP = True Positives

FN = False Negatives

FP = False Positives

TN = True Negatives

Fig.4. Confusion Matrix

5.5 Measuring the Performance

Comparing the two algorithms involve determination of standard measures as outlined in the confusion matrix shown above. The two algorithms are tested and compared using multiple data sizes and feature groups. As stated before, Accuracy , Spam Precision and Spam Recall are the parameters used for evaluation.

Accuracy is measured by dividing the quantum of samples identified correctly with sum total of all samples. It indicates the ratio of emails which are identified correctly .

$$Accuracy = \frac{Quantum\ of\ properly\ identified\ samples}{Total\ quantum\ of\ samples\ tested}$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

Spam precision indicates to what extent the algorithm is correct in identifying and classifying a spam. It is the ratio of number of true positives to total quantum of samples classified as positives. Alternately it is the percentage of mails which are identified as spam and are actually spam. A high value for Precision is a must since wrong classification of a legitimate email as spam causes much more damage than identifying a spam email as legitimate one. The mathematical representation is given below.

$$Precision \;=\; \frac{Number\ of\ true\ positives}{Sum\ total\ of\ samples\ identified\ as\ positives}$$

$$Precision \;=\; \frac{TP}{TP + FP}$$

As you can see, Precision indicates the algorithm's efficiency in differentiating legitimate emails. Spam recall gives algorithm's efficiency in classifying all correct mails. It is the ratio of true positives to the total number of samples which are actually positives. Spam Recall represents the percentage of emails in the dataset which are actually spam and are classified as spam email. These emails are actually blocked by the filter. The mathematical representation is

$$Recall \;=\; \frac{Number\ of\ true\ positives}{Total\ number\ of\ positive\ samples}$$

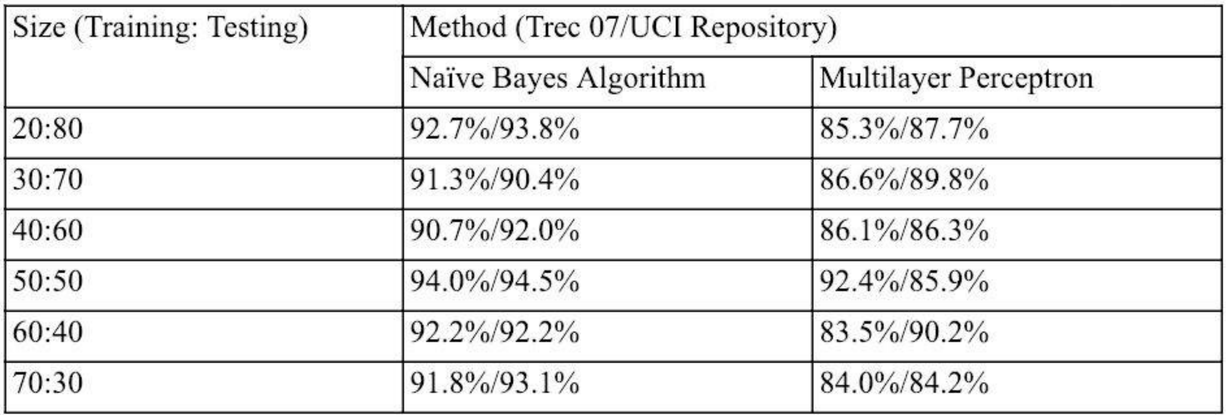
$$Recall \;=\; \frac{TP}{TP + TF}$$

All the three parameters defined above are used for comparing the two machine learning algorithms.

6. Conducting the Experiment

Independent running of Training and Testing sets have given the accuracy listed in Table 1. The benchmarking sets are in the ratio of 20:80 to 70:30. The experiment is done using the feature lists discussed above. The experiments are conducted using WEKA software.

TABLE I:ACCURACY OF BENCHMARKING SPAMBASES – Trec 07 & UCI Repository



Size (Training: Testing)	Method (Trec 07/UCI Repository)	
	Naïve Bayes Algorithm	Multilayer Perceptron
20:80	92.7%/93.8%	85.3%/87.7%
30:70	91.3%/90.4%	86.6%/89.8%
40:60	90.7%/92.0%	86.1%/86.3%
50:50	94.0%/94.5%	92.4%/85.9%
60:40	92.2%/92.2%	83.5%/90.2%
70:30	91.8%/93.1%	84.0%/84.2%

7. Data from the experiment

Table 2 lists the result of the experiment conducted to calculate all the parameters (Accuracy, Precision and Recall) on different feature sets by running the algorithms on the WEKA software. We can see that Naïve Bayesian has scored a 92.2% consistently on all three parameters as compared to Multilayer Perceptron Classifier.

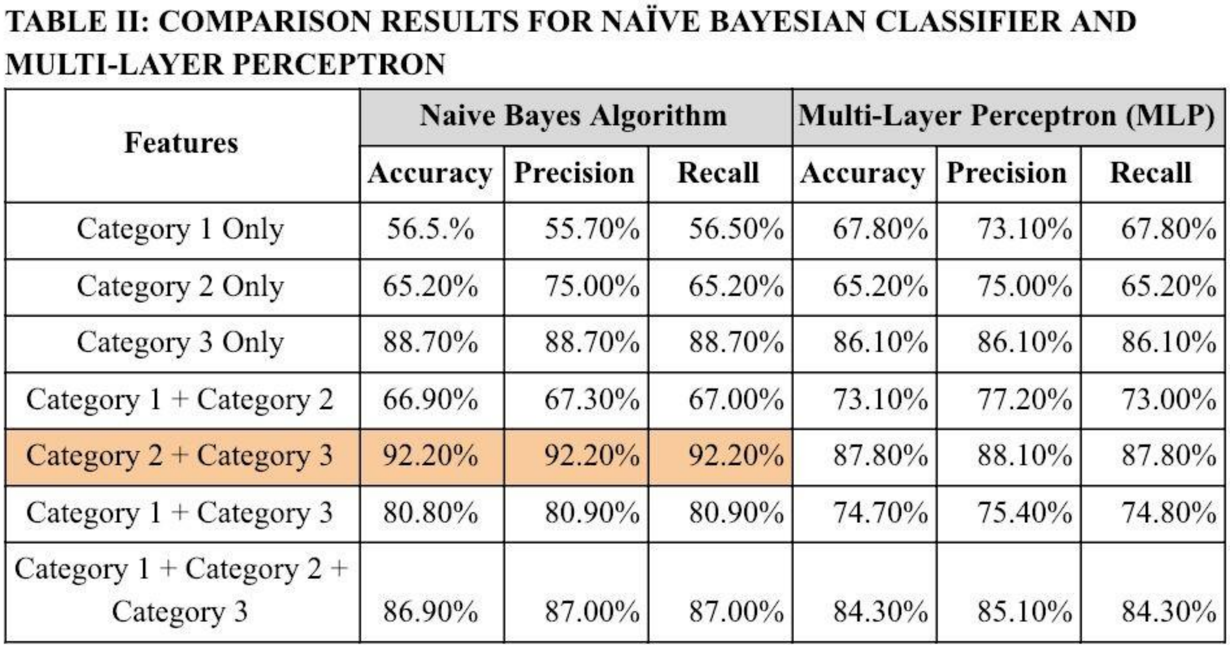


TABLE II: COMPARISON RESULTS FOR NAÏVE BAYESIAN CLASSIFIER AND MULTI-LAYER PERCEPTRON

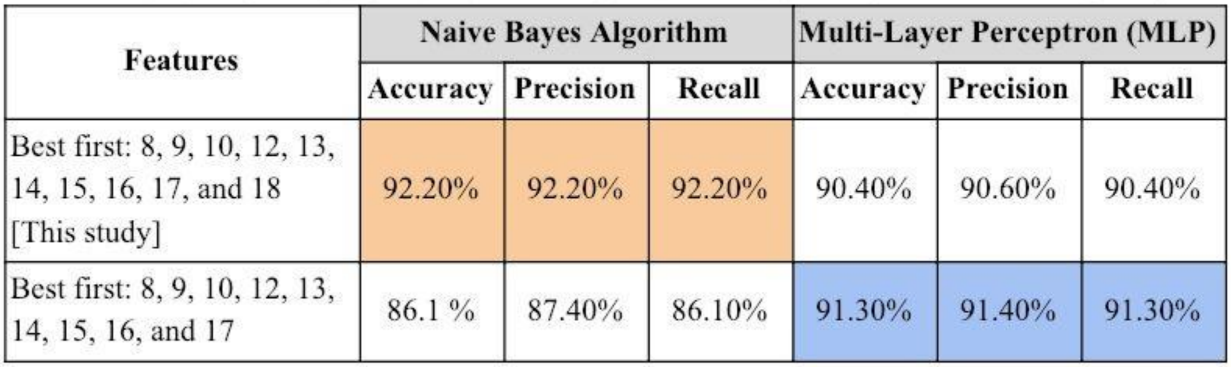
Features	Naive Bayes Algorithm			Multi-Layer Perceptron (MLP)		
	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Category 1 Only	56.5.%	55.70%	56.50%	67.80%	73.10%	67.80%
Category 2 Only	65.20%	75.00%	65.20%	65.20%	75.00%	65.20%
Category 3 Only	88.70%	88.70%	88.70%	86.10%	86.10%	86.10%
Category 1 + Category 2	66.90%	67.30%	67.00%	73.10%	77.20%	73.00%
Category 2 + Category 3	92.20%	92.20%	92.20%	87.80%	88.10%	87.80%
Category 1 + Category 3	80.80%	80.90%	80.90%	74.70%	75.40%	74.80%
Category 1 + Category 2 + Category 3	86.90%	87.00%	87.00%	84.30%	85.10%	84.30%

It is noteworthy to see that the performance was achieved in Category 2 and 3 which contribute maximum in deciding whether a particular email is legitimate or spam. Category 1 has negligible contribution while categories 2 and 3 contain the most important features for a machine learning algorithm.

We could have increased the features to the maximum while running the experiment, it was found that large number of unimportant features become a hindrance for performance and thus an optimum value had to be found.

Table 3 gives the result of the experiment for optimum values.

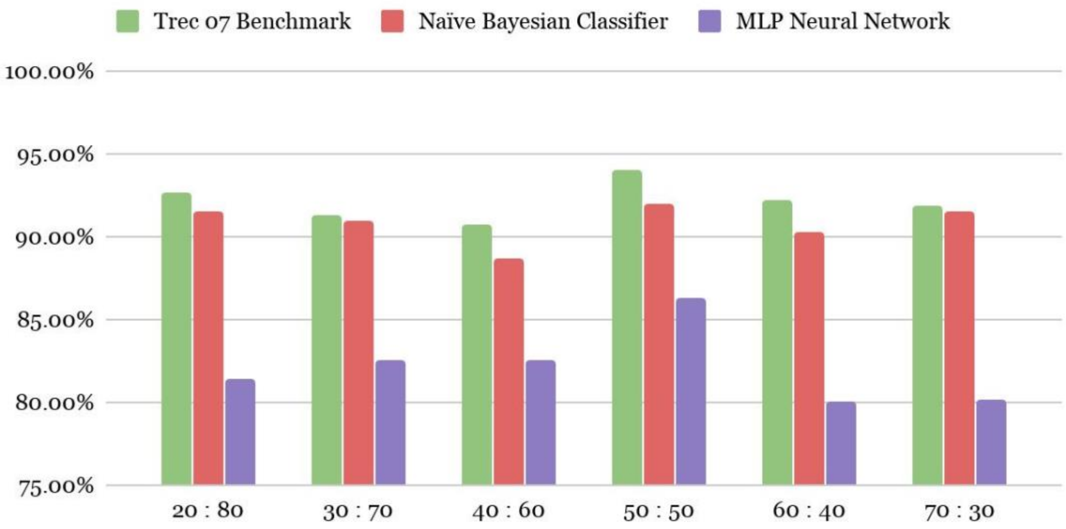
TABLE III: EVALUATION OF NAÏVE BAYES AND ANN WITH OPTIMAL FEATURE SET



Features	Naive Bayes Algorithm			Multi-Layer Perceptron (MLP)		
	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Best first: 8, 9, 10, 12, 13, 14, 15, 16, 17, and 18 [This study]	92.20%	92.20%	92.20%	90.40%	90.60%	90.40%
Best first: 8, 9, 10, 12, 13, 14, 15, 16, and 17	86.1 %	87.40%	86.10%	91.30%	91.40%	91.30%

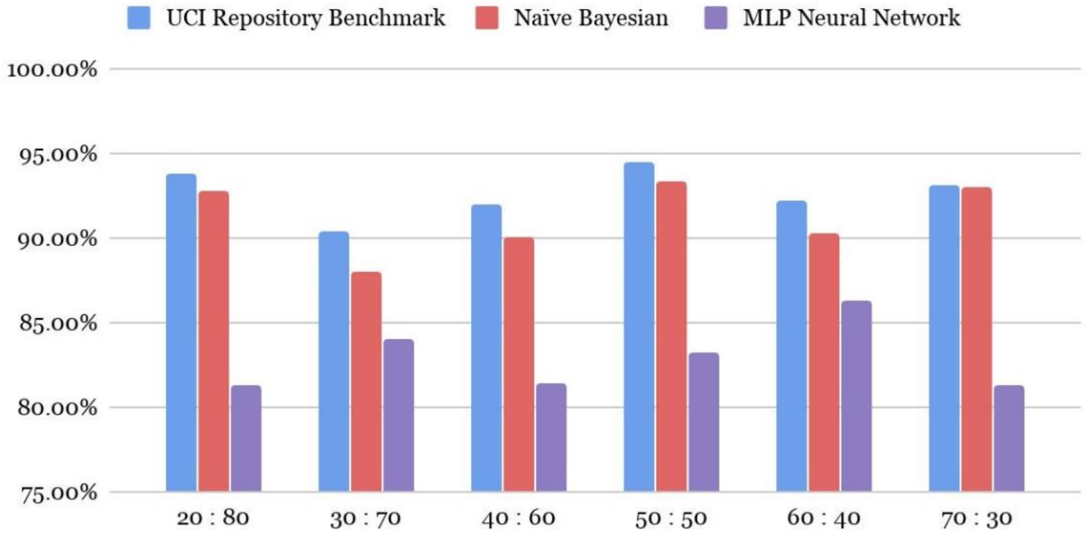
The table above shows the influence of features in the performance of the two algorithms. The Naïve Bayesian algorithm performs better than MLP when the important features are included. Removing one feature tilts the balance towards Multilayer Perceptron classifier as the data in blue highlight shows. The results show the importance of choosing the right set of features for an experiment.

The two graphs below compares the experimental results for Accuracy with the benchmark values.



Graph 1: Comparing Naïve Bayesian Classifier and MLP Neural Network with the benchmark values for Trec 07 Dataset

Graph 2: Comparing Naïve Bayesian Classifier and MLP Neural Network with the benchmark values for UCI Repository



The graphs above shows the alignment of the experimental results with the benchmark figures for Accuracy. The results indicate that Naïve Bayesian Algorithm is more effective than the Multilayer Perceptron Classifier for detection and classifying spam.

8. Limitations and Possible Improvements

The Essay had limited scope due to the paucity of time and further expanded to cover the following.

1. Different network architectures such as back propagation network, RBF network etc. can be included to expand the scope.
2. Get deeper into MLP-NN methodology by implementing additional layers and studying its influence on the results.
3. The study is limited to two methodologies. One can increase the scope by adding looking at additional methodologies like Fuzzy logic for example.

4. Also the study can be further expanded by using a combination of filters instead of looking at independently and also utilize header filters instead of only using body filter.

9. Conclusions

While it is ideal to use machine learning algorithms for spam detection and classification, obtaining optimal performance depends on the kind of test data available and its analysis. One also needs to ensure that the data is appropriate for different types of spam mails.

The above concept assumes further importance considering that the main difference between the spam and legitimate emails is in its contents and the algorithms should be able to differentiate this. Considering that there are different types of Spam mails, one may find it almost impossible to use a single machine learning algorithm to achieve 100% results. A combination could be the right kind of solution. But one thing is clear that the machine learning algorithms are the most suitable for detecting and classifying spam mails.

Concluding the question “Which **of the two algorithms is more efficient, Naïve Bayes or Multilayer Perceptron Classifier for content-based spam email detection?”,** the result clearly shows that the Naiive Bayes algorithm is the winner considering the limited scope of this study. It is also concluded that

1. The spam emails are best represented by the specific feature present in different parts of the mail, mainly the contents.
2. The subject line has no significant influence in classifying a mail as spam or not.

10.Appendix

10.1Relevant System configuration

These were my computer specifications:

Processor: Intel Core2duo i7-4330NQ CPU @ 2.00 GHz

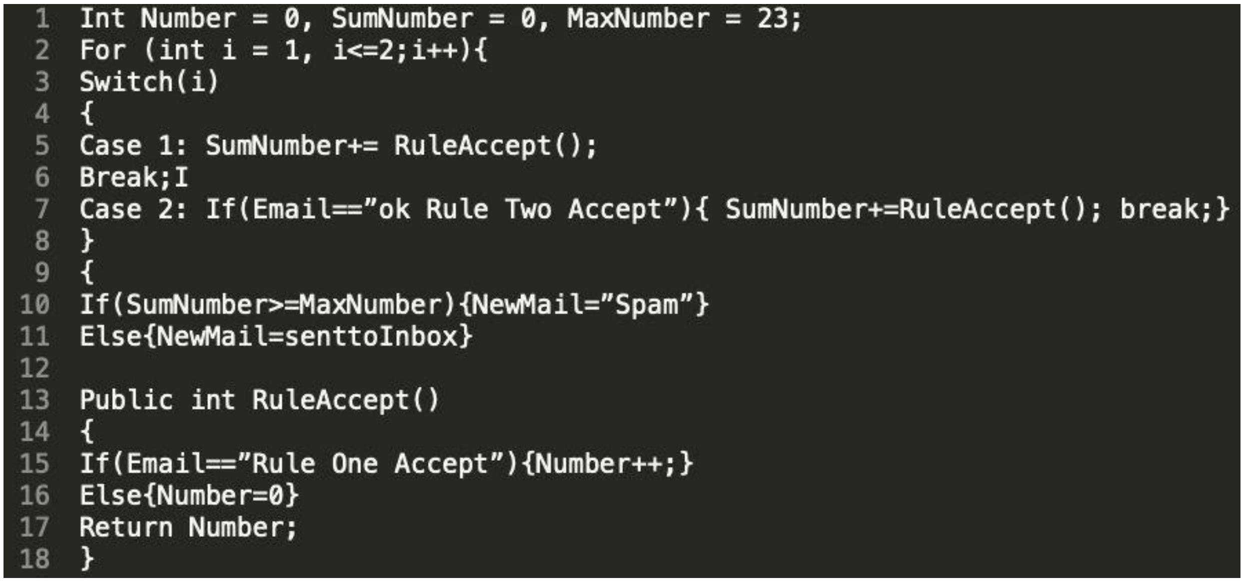
Memory: 4GB RAM DDR3

Graphics Processor: NVidia GeForce GTX 766M Mobile

Operating System: Windows 10 Home

10.2 Pseudocode for testing the Features

The pseudocode for the algorithm is as follows and can be compiled in C# language.



```
Int Number = 0, SumNumber = 0, MaxNumber = 23;
For (int i = 1, i<=2;i++){
Switch(i)
{
Case 1: SumNumber+= RuleAccept();
Break;I
Case 2: If(Email==”ok Rule Two Accept”){ SumNumber+=RuleAccept(); break;}
}
{
If(SumNumber>=MaxNumber){NewMail=”Spam”}
Else{NewMail=senttoInbox}

Public int RuleAccept()
{
If(Email==”Rule One Accept”){Number++;}
Else{Number=0}
Return Number;
}
```

11.Bibliography

1. Islam, Saiful, et al. "Modeling Spammer Behavior: Artificial Neural Network vs. Naïve Bayesian Classifier." *Artificial Neural Networks - Application*, 2011, [Accessed on 18th Nov 2019]
2. Aladdin Knowledge Systems, Anti-spam white paper. [Accessed on 1st Nov 2019]
3. Clark I. Koprinska, J. Poon, A neural network based approach to automated e-mail classification, in: Proc. *IEEE/WIC International Conference on Web Intelligence* .[Accessed on 14th Nov 2019]
4. Vapnik V, Drucker H, D. Wu, Support vector machines for spam categorization, *IEEE Transactions on Neural Networks* [Accessed on 10th Oct 2019]
5. R. Hunt, J. Carpinter, Tightening the net: a review of current and next generation spam filtering tools, *Computers and Security* [Accessed on 25th Nov 2019]
6. McCallum, K. Nigam, A comparison of event models for Naive Bayes text classification, [Accessed on 25th Nov 2019]
7. M.E. Tipping, Sparse Bayesian learning and the relevance vector machine, *Journal of Machine Learning Research* [Accessed on 25th Nov 2019]
8. Sahami M, Dumais S, Heckerman D, Horvitz, A Bayesian approach to filtering junk e-mail, in: Learning for Text Categorization: Papers from the 1998 Workshop, Madison, Wisconsin. [Accessed on 10th Nov 2019]
9. R. Bernardete, S. Caterina - RVM ensemble for text classification, *International Journal of Computational Intelligence Research* [Accessed on 23rd Nov 2019]

10. V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, Berlin. [Accessed on 10th Nov 2019]

11. D.A. Karrasb V. Zorkadisa, , Panayotou M, Efficient information theoretic strategies for classifier combination, feature extraction and performance [Accessed on 10th Nov 2019]

12. https://medium.com/analytics-vidhya/building-a-spam-filter-from-scratch-using-machine-learning-fc58b178ea56

13. https://machinelearningmastery.com/naive-bayes-for-machine-learning/

14. www.researchgate.net