International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 10 Issue 08, August-2021

Email Spam Detection and Data Optimization using NLP Techniques

Thirumagal Dhivya S[1], Nithya S[1], Sangavi Priya G[1], Pugazhendi E[2]
[1]B.Tech Student, [2] Teaching Fellow,
[1,2]Department of Information Technology, Anna University, MIT campus,
Chrompet, Chennai – 600044

Abstract— **Today, spamming mails is one of the biggest issues faced by everyone in the world of the Internet. In such a world, email is mostly shared by everyone to share the information and files because of their easy way of communication and for their low cost. But such emails are mostly affecting the professionals as well as individuals by the way of sending spam emails. Every day, the rate of spam emails and spam messages is increasing. Such spam emails are mostly sent by people to earn income or for any advertisement for their benefit. This increasing amount of spam mail causes traffic congestion and waste of time for those who are receiving that spam mail. The real cost of spam emails is very much higher than one can imagine. Sometimes, the spam emails also have some links which have malware. And also, some people will get irritated once they see their inbox which is having more spam mails. Sometimes, the users easily get trapped into financial fraud actions, by seeing the spam mails such as job alert mails and commercial mails and offer emails. It may also cause the person to have some mental stress. To reduce all these risks, the system has proposed a machine learning model which will detect spam mail and non-spam emails, and also this system will optimize the data by removing the unwanted mails which contain the advertisement mails and also some useless emails and also some fraud mails. This proposed system will detect the spam mails and ham emails with the dataset consisting of spam mails and after identifying spam mails this system will remove that spam emails and this proposed system will calculate the amount of storage before and after the removal of spam mails.**

I. INTRODUCTION

In this thesis, efficient spam detection and data optimization methodology for emails is proposed. This chapter provides the overview of the problem area, describes the specific problem that is dealt in this work, sketches the approach for solving this problem.

A. Overview

The major issues faced by all the email users are spam mails which contain unwanted information and data and some fake data to spoil the life of the people and also some mails which cause harmful effects. Today, the job issues are faced by fifty percent of the people by both educated and uneducated people. In such a case, these people will get emails about advertisement mails which are completely fake. But by seeing that mail, this people will get interested or have a thought to communicate through the mail for what they are looking into it. More people are affected by this spam mails in similar cases. To reduce this risk and to save the people from this danger of spam mails, we are proposing this system to remove the spam mails. For filtering the spam mails, in this system we are using two filtering model. Namely, Opinion Rank and NLP based n-grams model. By using these two models we will filter the spam mails and non-spam mails. And this system will optimize the data by removing the spam mails and also it calculates the storage of the mails.

B. Research Challenges

The finding of trust rank of the mail and classifying those mails as spam and ham mails based on their content. And detection of advertisement mails in those mails. After detecting the advertisement mails, optimizing the storage by deleting those advertisement mails. By deleting mails, the proposed system will optimize the data. And also, the system will identify the fake mails which look similar to real mails that people can believe.

C. Objective

The main objective of the project is to detect the spam mails and to optimize the data storage. This detection of spam mails in this proposed system is done through the two filtering models. One is Opinion Rank which is based on the trustworthiness of the mail id and this rank uses the two algorithms namely, high page rank and inverse page rank. By combining these results, and by calculating the mean of this results, the Opinion Rank will perform. And the data optimization is done by removing the advertisement mails with the help of Latent Dirichlet Allocation which is a probabilistic topic modelling to classify the contents or documents based on the topics.

D. Scope

The proposed system of the project will effectively detect the spam mails and the system will extract the spam mails by using some machine learning algorithms and it gives the result with greater accuracy and with good performance. Also, this proposed system will optimize the data storage by blocking and deleting the spam mails. And with the help of the Opinion Rank model, this proposed system will find trustworthiness of the mail and it will carry the filtering of spam messages. This proposed system will save the user's time and it destroys the risk of spam mails.

E. Contribution

This proposed system has been implemented with opinion rank to find trustworthiness of the mail id using three valued subjective logic. In this opinion rank this system will use the average mean of page rank and trust rank. By using this opinion rank this system will evaluate the mail id’s and also this system will block the mail id’s which sends the spam mails to the users. And also, this proposed system will optimize the data storage by deleting the spam mails.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

II. LITERATURE SURVEY

This chapter provides a brief insight on the related works of email spam detection and mail data optimization. Summary of various methodologies have also been discussed.

A. Opinion Rank

In this paper, XiaofeiNiu et al. (2020) proposed the Opinion Rank algorithm for computing the trustworthiness of every available website and to identify the trustworthy ones with high trust values. This algorithm is based on a breadth-first search algorithm which starts from an existing set of trustworthy websites. Because, this websites play an important role in Opinion Rank. They also used other algorithms such as High PageRank and Inverse PageRank to rank the websites based on their trustworthiness. By using the public dataset, they have validated the Opinion Rank and HarMean PageRank which analyse the impact of website selection. The Opinion Rank algorithm computes the trustworthiness of all websites, trust propagation and trust combination operations which is defined in three valued subjective logic trust model. And the HarMean PageRank combines the results of PageRank and Inverse PageRank. The convergence and the performance of this algorithm is better than Trust Rank and Good Rank algorithms. This Opinion Rank algorithm consists of two components. Namely, the website selection and trust assessment. The website selection identifies the subset of trustworthy websites from a dataset. This algorithm will update the trust values of all websites from the chosen websites. Based on this trust values, websites are ranked by this Opinion Rank algorithm. Opinion Rank detects more trustworthy websites and lesser spam websites with a shorter time. One of its disadvantages is, it is very challenging to identify the subset of websites which are needed to update its trust value.

B. Spam Detection for Secure Mobile Communication

In this paper, Luo GuangJun et al. (2020) proposed the applications of machine learning based-spam detection for accurate detection of spams. For classification of spam and ham messages in mobile device communications they have used the Logistic Regression, K-nearest neighbor and Detection Tree. The collection of SMS dataset is used for testing the methods. And the dataset is splitted into two sets as one is for testing and another one is for training. And 70 percent of data is used for training purposes and 30 percent is used for testing purposes. The Logistic Regression is a classifier which computes the predictive y in the problem of binary classification as 0 or 1 such that it belongs to class negative or class positive. It predicts values for the variable in multi classification. The Decision Tree is a supervised machine learning algorithm which is like the shape of a tree at which each node is a decision node or leaf node. In this tree the nodes are interlinked with each other. The K-nearest neighbour classification is also a supervised learning algorithm but this performance is not good enough.

C. Machine Learning Based Spam Email Detection

In this paper, Nandhini et al. (2018) proposed a machine learning model based on a hybrid bagging approach by implementing with the help of two machine algorithms for detecting the spam emails. Namely, Naive Bayes algorithm and J48 (Decision tree) algorithm. In this process of detecting the spam mails, the dataset is divided into different sets and given as input to each of the algorithm. Totally, they performed three experiments in this paper. The first experiment is performed with the Naive Bayes algorithm. It is a classifier based on the probability and it computes the probabilities of the class of the given instances. And the second experiment is performed with the J48 Decision tree algorithm. It is based on the concept of entropy and it forms the decision trees of the training data. The third experiment is the proposed Spam Mail Detection (SMD) system by using the hybrid bagged approach which is the combination of J48 algorithm and Naive Bayes Multinomial classifier. It classifies the email into spam mails and ham mails. It consists of four modules which are preparation of email dataset, pre-processing of data, feature selection and hybrid bagged approach. Only the J48 algorithm gives the experimental results better. Other two experiment gives low performance. To enhance the system's performance by using the concept of boosting approach. It will replace the features of weak classifier learning features with a strong classifier's approach.

D. Machine Learning Methods For Spam Email Classification

In this paper, Luo GuangJun et al. (2011) checked and reviewed the very popular machine learning methods for their capability of classifying the spam mails. Here the methods used are Bayesian classification, K-nearest neighbour classifier method, artificial neural network classifier method, Support vector machine classifier method, Artificial immune system classifier method, and rough sets classifier method. The Naive Bayes classifier method is based on the probability of an event occurring in the future which can be detected by the previous occurring of the same event. And based on that probability it will classify the mail as spam or ham mail. Here the probability of the word plays the major role in classification. The k-nearest neighbour classifier method is based on the example. It will check the previous documents for classification. And finding the nearest neighbour is done by using the traditional indexing methods. The Artificial Neural Network is also known as Neural Network. It is based on a biological neural network and consists of a collection of artificial neurons. At the time of the learning phase, it changes its structure based on the information that flows through the artificial network. It has the stages of training and filtering stage. The Support Vector Machine classifier method is based on the concept of decision planes which define the boundaries of the decision. This algorithm finds the optimal hyperplane with maximum margin for separating the two classes which is mainly required for solving the optimization problems. In the Artificial Immune System classifier method the overall response involves three evolutionary methods namely gene library, negative selection and global selection. This will organize the fittest antibodies by interacting with current antigens. The rough set classifier method has an ability to reduce the information systems. While they

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

summarize these six methods, the Naive Bayes is the most accurate and also in terms of spam precision this method gave the highest precision among the six methods. The neural network has the simplest and fastest algorithms, while the rough set method is most complicated and it has to be hybrid with genetic algorithms to get the deserved results. The Artificial Immune System method gave a satisfying result which is to be expected for better performance but it gave the poor performance. It will provide the good performance when it is hybridized with rough set method.

E. Email Spam Detection Using Integrated Approach Of Naive Bayes And Particle Swarm Optimization

In this paper, Kaur et al. (2018) have proposed a machine learning model by integrating the Naive Bayes algorithm and intelligence-based Particle Swarm Optimization which is used for detecting spam mail. The Naive Bayes algorithm is based on the Bayes theorem which has a strong probability distribution property. And the Particle Swarm Optimization is inspired from the behaviour of the fishes and the birds. The Naive Bayes algorithm determines the mail as spam class and non-spam class based on the keywords present on the email data. And the Particle Swarm Optimization method is further used to optimize the parameters of Naive Bayes algorithm to improve the accuracy and classification process. To perform the feature extraction, pre-processing is done for the email. The Pre-processing have some methods such as tokenization, stemming and stop word removal. After that we will apply the particle optimization method. Based on this feature of optimization method, the tokens of the mail is classified as spam or non-spam. They evaluated the performance of the system in terms of precision, recall and accuracy of the classification. Their parameters are calculated with help of true positive, true negative, false positive and false negative. It has been found that the integrated approach of Naive Bayes and Particle Swarm Optimization overcomes the failure of the Naive Bayes approach. We can also use swarm optimization concepts like ant colony optimization, artificial bee colony optimization and firefly algorithm. Further, to improve the performance instead of Naive Bayes, we can use any other machine learning algorithm.

F. Summary of Literature Survey

In spam detection approach for secure mobile communications using machine learning algorithms, the detection of spam messages is not very accurate and also the logistic model present in this approach affects the detection of ham messages. In machine learning based spam email detection, three experiments have been proposed based on naive bayes, J48 algorithm and combination of these two algorithms. But it gave low performance results in detection of spam and ham mails. In machine learning methods for spam email classification, the methods used are Bayesian classification, K-nearest neighbour classifier method, artificial neural network classifier method, Support vector machine classifier method, Artificial immune system classifier method, and Rough sets classifier method. In these six methods, the naive bayes gave good performance in detection of spam mails but the other methods gave very poor performance. In email spam detection using integrated approach of naive bayes and particle swarm optimization, it gave poor performance in detecting spam or ham mails due to the presence of naive bayes approach. But the performance can be improved by using any machine learning algorithms. In content based spam email filtering, it is only focused on the email content, but there is also some useful information such as sender email address and IP address, email subject, number of recipients or even time for detection of spam mails. Due to this approach, it doesn't give the exact accuracy in detection of spam mails. $\alpha+\beta=\chi.$ (1) (1) In spam filtering email classification using gain and graph mining algorithms, it fails to detect spam mails with good performance and with good accuracy. In identifying spam email based on statistical header feature and sender behavior, it get poor classification results if the spammers keep changing his or her email address. In opinion rank, it is used for only to find the trustworthiness of website. This proposed system is mainly designed to find the trustworthiness of a mail id to find whether the email is spam or not. And also this system will optimize the data by removing unwanted mails such as advertisement mails, commercial mails, fraud mails and so on. This proposed system will give good results in detection of spam mails by using n gram model.

III. PROPOSED SYSTEM ARCHITECTURE AND DESIGN

This chapter provides the system architecture of mail data optimization using Latent Dirichlet Allocation(LDA) and email spam detection using NLP N-grams model and Opinion Ranking.

A. Mail Data Optimization

Advertisement mails consume more space since it has attachments in .jpg, .png format. Here, mail data optimization is achieved by deleting the advertisement mails with attachments. The following steps occur during optimization. Dynamic input mails are taken from various platforms like Gmail, Yahoo, Live Mail using the Java mail API and then data preprocessing is done and various steps are carried out such as Tokenization, Lemmatization, Stemming. Figure 3.1 shows the architecture of mail data optimization using NLP architecture and Figure 3.2 shows the architecture of Spam detection using NLP N-gram model.

1) Mail Data Optimization using NLP architecture

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

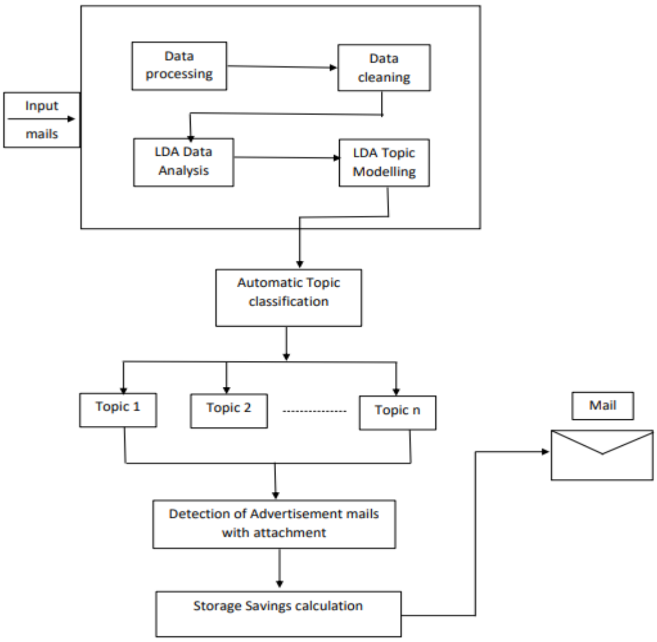

Input
mails
Data processing
Data cleaning
LDA Data Analysis
LDA Topic Modelling
Automatic Topic classification
Topic 1
Topic 2
Topic n
Mail
Detection of Advertisement mails with attachment
Storage Savings calculation

Fig 3.1 Mail Data Optimization using NLP architecture

2) *Spam Detection Using Nlp N-Gram Model Architecture*

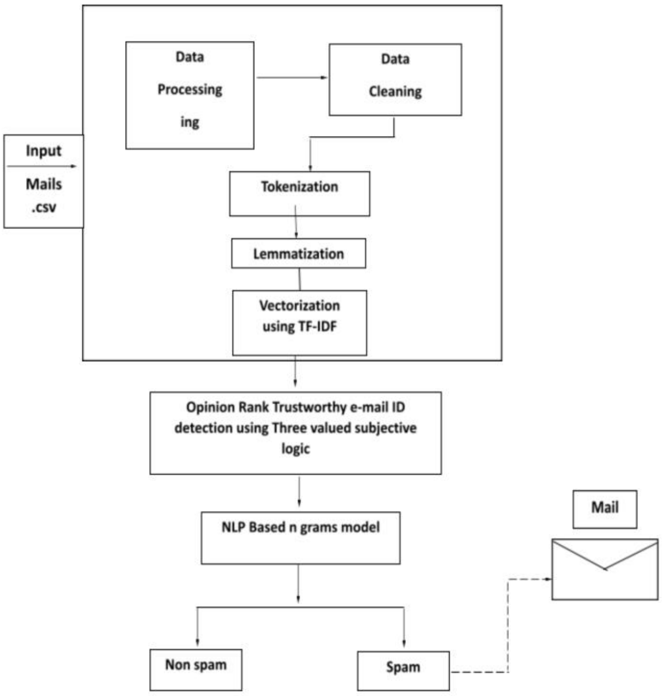


Fig 3.2 Spam Detection using NLP N-Grams Model Architecture

For designing this proposed system, first this system will take an input file in the form of a csv file. This input file has a collection of dataset consisting of more than 5000 emails consisting of both ham and spam mails. The data will be split into 80% training and 20% testing. This data will go into the process of data preprocessing and data cleaning. Then this data will be given for the tokenization and lemmatization process. This will be the initial data preparation to be done. This will then be given to the Opinion Rank model. In the Opinion Rank model, the website links which are extracted will be determined for its trustworthiness based on the Opinion Rank method. This will then be given to the NLP based n-gram model where bigram is performed to determine whether it is ham or spam. Then this will be sent to mail.

3) *Latent Dirichlet Allocation*

Using the famous topic modelling technique, Latent Dirichlet Allocation(LDA) topic classification is done by giving the dynamic topics. LDA is the generative statistical model which explains the similarity of data. LDA classifies the document to a specific topic. Then using the java mail API, advertisement mails are detected and deletion operation is performed.

B. *Opinion Rank Trustworthy Email Id Detection*

This method consists of two components, seed selection and trust assessment. From the dataset that is used the seed selection identifies the subset of Trustworthiness. In the next process the Opinion Rank algorithm will search the whole hyperlink graph from the selected seeds and will update the trust values of the websites. Then, the Opinion Rank algorithm will then be assigned based on the trust values. The HarMean PageRank algorithm is used for the next process, for this the good seed set will be selected first. The seed selection process will be carried out with the algorithms High PageRank and Inverse PageRank. The High PageRank algorithm will select trustworthy seeds without the consideration of the outdegrees of the seeds. The Inverse PageRank algorithm will choose the seed that has more outlinks and lower importance. This process is not conducive to identifying the trustworthiness of the website. These are combined by harmonic means to compute a new value of trustworthiness for the websites. The set of seeds will be set as opinions and the Opinion Rank algorithm will search level by level in the network and update the trustworthiness of every node.

C. *NLP Based N Gram Model*

The next process in this is the NLP N-gram technique. This model is based on word prediction, it will predict the word that will come next using the previous n-1 words. This Bigram will be used. A Bigram is a 2-word in N-gram model, this will predict the next word of a sentence using the previous one word. The identification of the mail as ‘Ham’ or ‘Spam’ is a classification task. Bigram is the combination of adjacent words of length 2 and this will be performed. Next TF-IDF will be applied on the mail and the relative count of the words in the sentence will be stored in the document matrix. Next, the punctuation percentage will be calculated with respect to the message length and these will be checked as good or not good. The details collected will be used to detect if it is spam or ham. This will then be updated.

IV. RESULTS AND DISCUSSION

In this chapter, various result screenshots of spam detection and data optimization such as dataset used, preprocessing steps that have been carried out, visualization of spam and ham messages, detection of spam-ham messages

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

using TF-IDF, testing with dynamic inputs, LDA topic classification have been attached and discussed.

A. System Specification

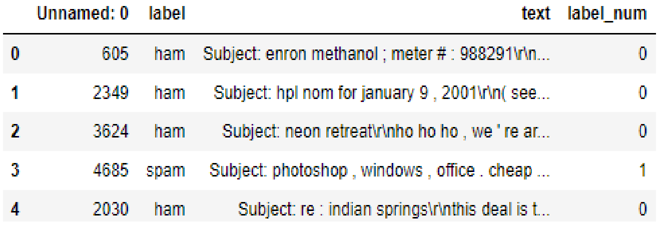
The specification of the CPU used for evaluation are as follows. The processor specification is "2.3 GHz Quad-Core Intel Core i5", RAM installed is 8GB and system specification is "64-bit operating system, x64 based processor". The specifications of IDE used are NetBeans and Google Colab. The languages used for development are Java and Python.

B. Spam Detection Using N-Grams Model

Spam emails occupy a lot of storage and hence it is detected using the N-Grams Model in which it detects the spam by analysing the n set of words. When the n set of words occur it is detected as spam or ham using the frequency and probability.

1) Dataset

5000 input mails are taken from kaggle and tested for spam using the NLP N-Grams Model. Also, it is tested with the personal mail. The figure 4.1 shows the dataset for spam
detection.

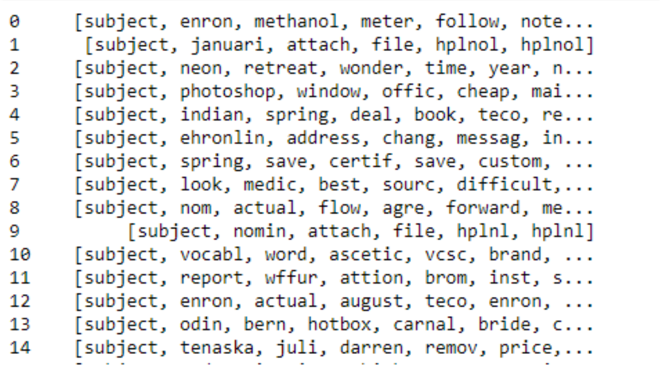


	Unnamed: 0	label	text	label_num
0	605	ham	Subject: enron methanol ; meter # : 988291\r\n...	0
1	2349	ham	Subject: hpl nom for january 9 , 2001\r\n(see...	0
2	3624	ham	Subject: neon retreat\r\nho ho ho , we ' re ar...	0
3	4685	spam	Subject: photoshop , windows , office . cheap ...	1
4	2030	ham	Subject: re : indian springs\r\nthis deal is t...	0

Fig 4.1 Dataset for spam detection

2) Data preprocessing

Data preprocessing is an important step in this. The data that is given to the model will affect the performance of the model. Therefore the data is processed before proceeding. In this model the punctuations will be removed for better prediction. The data which will be given to this model will be of string data type. This will be fed to the process which will identify each of the characters and the punctuations will be removed. This refers to the processes of identifying and correcting the errors that are present in the dataset that may negatively impact the model. In this process the stop words will be removed. Generally the stopwords will be removed. A stopword is a word which is usually the most used words in the natural language. The stopwords do not add any value to the model. So, removing these words will have a chance of good prediction. The figure 4.2 shows the removal of stopword.

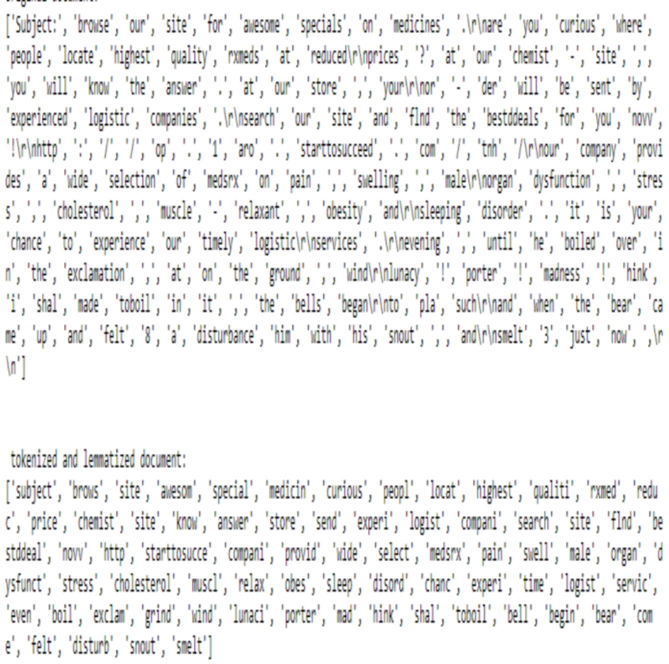


```
0      [subject, enron, methanol, meter, follow, note...
1       [subject, januari, attach, file, hplnol, hplnol]
2      [subject, neon, retreat, wonder, time, year, n...
3      [subject, photoshop, window, offic, cheap, mai...
4      [subject, indian, spring, deal, book, teco, re...
5      [subject, ehronlin, address, chang, messag, in...
6      [subject, spring, save, certif, save, custom, ...
7      [subject, look, medic, best, sourc, difficult,...
8      [subject, nom, actual, flow, agre, forward, me...
9           [subject, nomin, attach, file, hplnl, hplnl]
10     [subject, vocabl, word, ascetic, vcsc, brand, ...
11     [subject, report, wffur, attion, brom, inst, s...
12     [subject, enron, actual, august, teco, enron, ...
13     [subject, odin, bern, hotbox, carnal, bride, c...
14     [subject, tenaska, juli, darren, remov, price,...
```

Fig 4.2 Stopword Removal

3) Tokenization And Lemmatization

The next process in this is the tokenization process. In tokenization larger text is into smaller words. That is the will be split individually and will be put into appropriate data type. These tokenized words will be then used for the next purpose. Lemmatization is the process of converting a word into its natural base form. This will aim to remove the inflectional ending and return the base word. These are the starting processes that need to be done for an effective database. Figure 4.3 shows the tokens.



```
['Subject:', 'browse', 'our', 'site', 'for', 'awesome', 'specials', 'on', 'medicines', '.\r\nare', 'you', 'curious', 'where',
'people', 'locate', 'highest', 'quality', 'rxmeds', 'at', 'reduced\r\nprices', '?', 'at', 'our', 'chemist', '-', 'site', ',',
'you', 'will', 'know', 'the', 'answer', '.', 'at', 'our', 'store', ',', 'your\r\nor', '-', 'der', 'will', 'be', 'sent', 'by',
'experienced', 'logistic', 'companies', '.\r\nsearch', 'our', 'site', 'and', 'flnd', 'the', 'bestddeals', 'for', 'you', 'novv',
'!\r\nhttp', ':', '/', '/', 'op', '.', '1', 'aro', '.', 'starttosucceed', '.', 'com', '/', 'tnh', '/\r\nour', 'company', 'provi
des', 'a', 'wide', 'selection', 'of', 'medsrx', 'on', 'pain', ',', 'swelling', ',', 'male\r\norgan', 'dysfunction', ',', 'stres
s', ',', 'cholesterol', ',', 'muscle', '-', 'relaxant', ',', 'obesity', 'and\r\nsleeping', 'disorder', '.', 'it', 'is', 'your',
'chance', 'to', 'experience', 'our', 'timely', 'logistic\r\nservices', '.\r\nevening', ',', 'until', 'he', 'boiled', 'over', 'i
n', 'the', 'exclamation', ',', 'at', 'on', 'the', 'ground', ',', 'wind\r\nlunacy', '!', 'porter', '!', 'madness', '!', 'hink',
'i', 'shal', 'made', 'toboil', 'in', 'it', ',', 'the', 'bells', 'began\r\nto', 'pla', 'such\r\nand', 'when', 'the', 'bear', 'ca
me', 'up', 'and', 'felt', '8', 'a', 'disturbance', 'him', 'with', 'his', 'snout', ',', 'and\r\nsmelt', '3', 'just', 'now', ',\r
\n']


tokenized and lemmatized document:
['subject', 'brows', 'site', 'awesom', 'special', 'medicin', 'curious', 'peopl', 'locat', 'highest', 'qualiti', 'rxmed', 'redu
c', 'price', 'chemist', 'site', 'know', 'answer', 'store', 'send', 'experi', 'logist', 'compani', 'search', 'site', 'flnd', 'be
stddeal', 'novv', 'http', 'starttosucce', 'compani', 'provid', 'wide', 'select', 'medsrx', 'pain', 'swell', 'male', 'organ', 'd
ysfunct', 'stress', 'cholesterol', 'muscl', 'relax', 'obes', 'sleep', 'disord', 'chanc', 'experi', 'time', 'logist', 'servic',
'even', 'boil', 'exclam', 'grind', 'wind', 'lunaci', 'porter', 'mad', 'hink', 'shal', 'toboil', 'bell', 'begin', 'bear', 'com
e', 'felt', 'disturb', 'snout', 'smelt']
```

Fig 4.3 Tokenization and Lemmatization

.4) Visualization of spam messages

Email messages are separated into spam and ham from the dataset and it is visualised using word cloud. Using this user can easily visualise the words that commonly occurred in spam messages and in ham messages. The figure 4.4 shows the visualization of spam messages.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

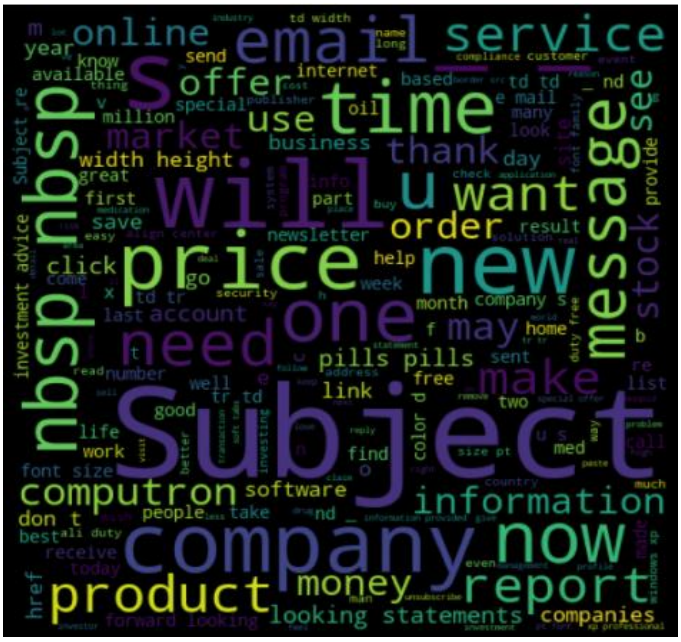


Fig 4.4 Visualization of spam messages using word cloud

5) Visualization of ham messages

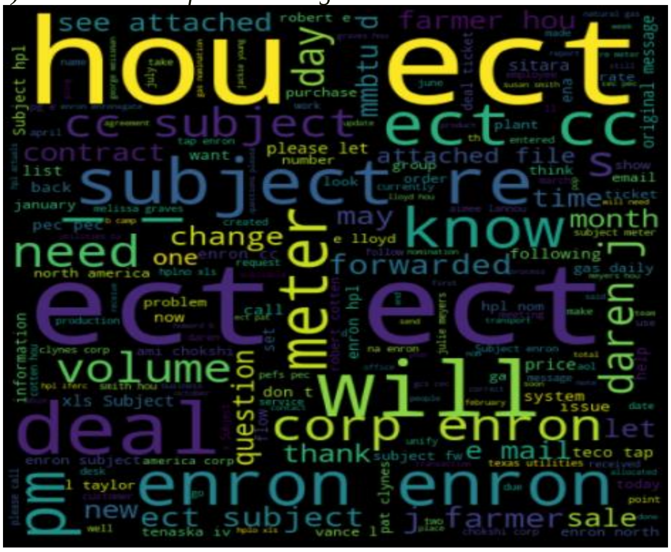


Fig 4.5 Visualization of ham messages using word cloud

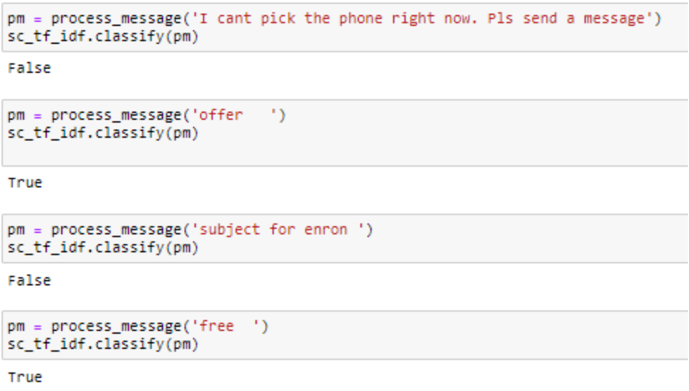
6) Vectorization using TF-IDF

TF-IDF refers to Term Frequency and Inverse Document Frequency. Term Frequency refers to how frequent a word appears in a document divided by total number of words in the document. This Vectorization using TF-IDF is shown in the figure 4.6.

TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document)

Inverse Document Frequency refers to the importance of a term in a document which is calculated by taking the logarithm of the number of documents in a corpus which is divided by how many times the specific pattern appears.

IDF(t) = log_e(Total number of documents / Number of documents with term t in it)

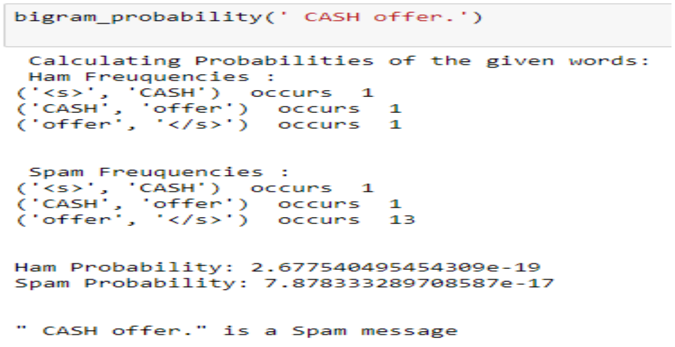


```
pm = process_message('I cant pick the phone right now. Pls send a message')
sc_tf_idf.classify(pm)

False

pm = process_message('offer   ')
sc_tf_idf.classify(pm)

True

pm = process_message('subject for enron ')
sc_tf_idf.classify(pm)

False

pm = process_message('free  ')
sc_tf_idf.classify(pm)

True
```

Fig 4.6 Vectorization using TF-IDF

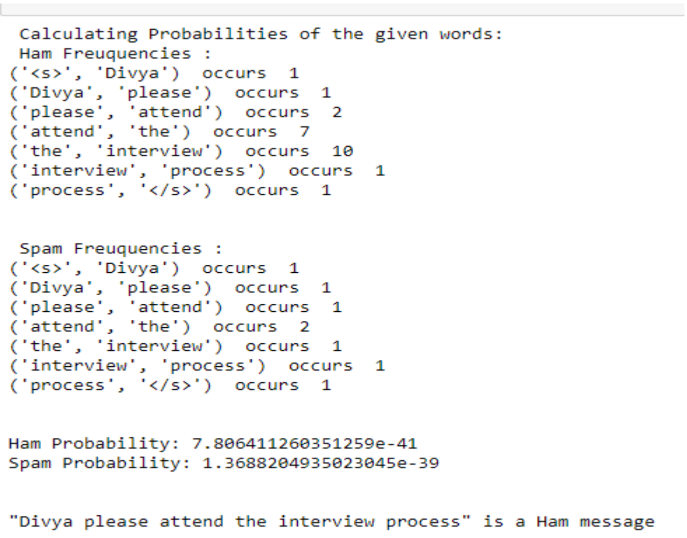
7) Detection of spam using Bigram

Bigram is the two-word sequence of n-grams where n refers to 2 here. Bigram is the model which predicts the following word using the occurrence of the previous word in a document. Here, using the occurrence of words in the mail dataset bigram predicts whether the mail is spam or ham. The detection of spam using bigram is shown in the figure 4.7 and figure 4.8.



```
bigram_probability(' CASH offer.')

 Calculating Probabilities of the given words:
 Ham Freuquencies :
('<s>', 'CASH')  occurs  1
('CASH', 'offer')  occurs  1
('offer', '</s>')  occurs  1


 Spam Freuquencies :
('<s>', 'CASH')  occurs  1
('CASH', 'offer')  occurs  1
('offer', '</s>')  occurs  13


Ham Probability: 2.677540495454309e-19
Spam Probability: 7.878333289708587e-17


" CASH offer." is a Spam message
```

Fig 4.7 Spam Detection using Bigram



```
 Calculating Probabilities of the given words:
 Ham Freuquencies :
('<s>', 'Divya')  occurs  1
('Divya', 'please')  occurs  1
('please', 'attend')  occurs  2
('attend', 'the')  occurs  7
('the', 'interview')  occurs  10
('interview', 'process')  occurs  1
('process', '</s>')  occurs  1


 Spam Freuquencies :
('<s>', 'Divya')  occurs  1
('Divya', 'please')  occurs  1
('please', 'attend')  occurs  1
('attend', 'the')  occurs  2
('the', 'interview')  occurs  1
('interview', 'process')  occurs  1
('process', '</s>')  occurs  1


Ham Probability: 7.806411260351259e-41
Spam Probability: 1.3688204935023045e-39


"Divya please attend the interview process" is a Ham message
```

Fig 4.8 Spam Detection using Bigram(continued)

8) Frequently occurring ham

Frequently occurring bigrams in the ham messages is visualized. It is seen that the bigram ((ill, call), (call, later))

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

has more occurrences in ham messages therefore considered as ham. The frequently occurring ham messages is shown in the figure 4.9.

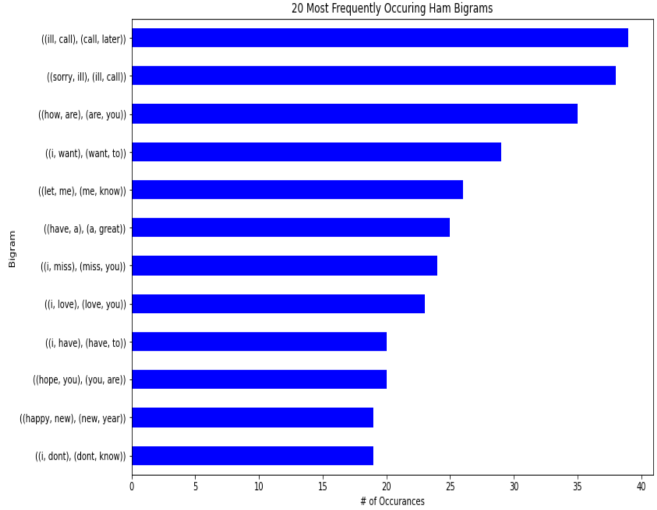


Fig 4.9 Frequently occurring ham messages

9) Frequently occurring spam

Frequently occurring bigrams in the spam messages are visualized. It is seen that the bigram ((you, have), (have, won)) has more occurrence in a spam message therefore considered as spam. The frequently occurring spam messages is shown in the figure 4.10.

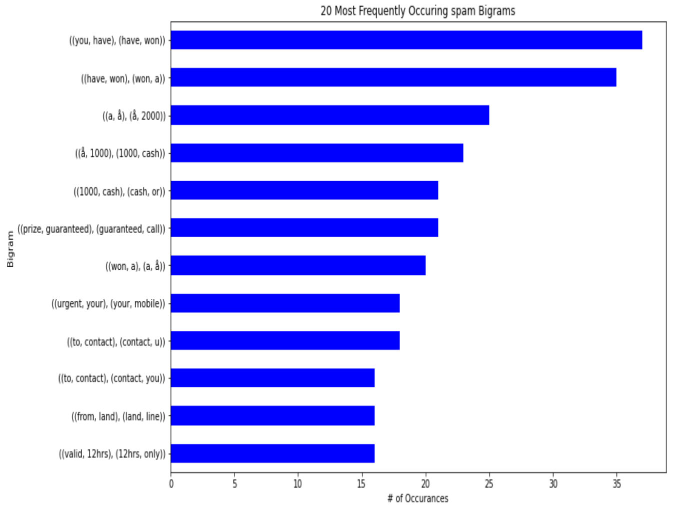


Fig 4.10 Frequently occurring spam messages

10) Detection of spam or ham

The system displays whether the email is spam or ham by taking the set of n-grams and analysing whether the set of words occurs in the spam message or in a ham message. Like any other Machine Learning or Artificial Intelligence models, the model should be trained with a huge corpus of data. After training this N-grams model, the system will have an idea on the probability of the spam or ham message by analysing the sequence of words in that message.

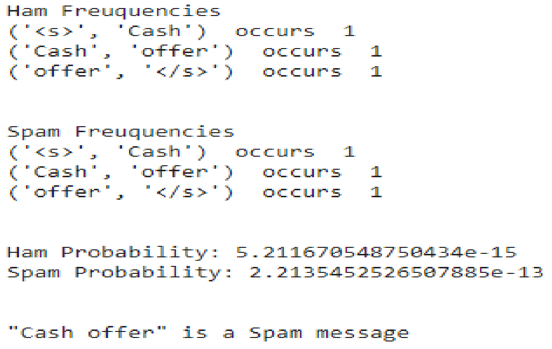
To measure the probability of spam or ham in a bigram model, the occurrence of every two words has been analysed by the model and then determines whether that is a spam or ham. The predictions will get improved when we give a bigger corpus. In a bigram the probability of spam and ham is classified by using the following formula.

- Probability of spam = count(word2 spam) / count(word 2)
- Probability of ham = count(word2 ham) / count(word 2)

Probability calculation is done by calculating the probability of the word spam or ham occurring after the word ‘w2’ which is how many times the word occurs in the required sequence and it is divided by number of times the word before the expected word occurs in the corpus. For example, if the given sentence is ‘cash offer’, then the following probabilities are calculated, P(‘null’, cash), P(cash, offer), P(offer, spam). P(offer, spam) is calculated by,

- Probability of spam=count(number of times the word ‘offer’ occurs in the given sentence * number of times the word ‘spam’ occurs i.e 1 if mentioned as spam or 0 if mentioned as ham) / number of times the word ‘offer’ occurs in the entire corpus which is mentioned as spam.

The detection of spam or ham using probability is shown in the figure 4.11.



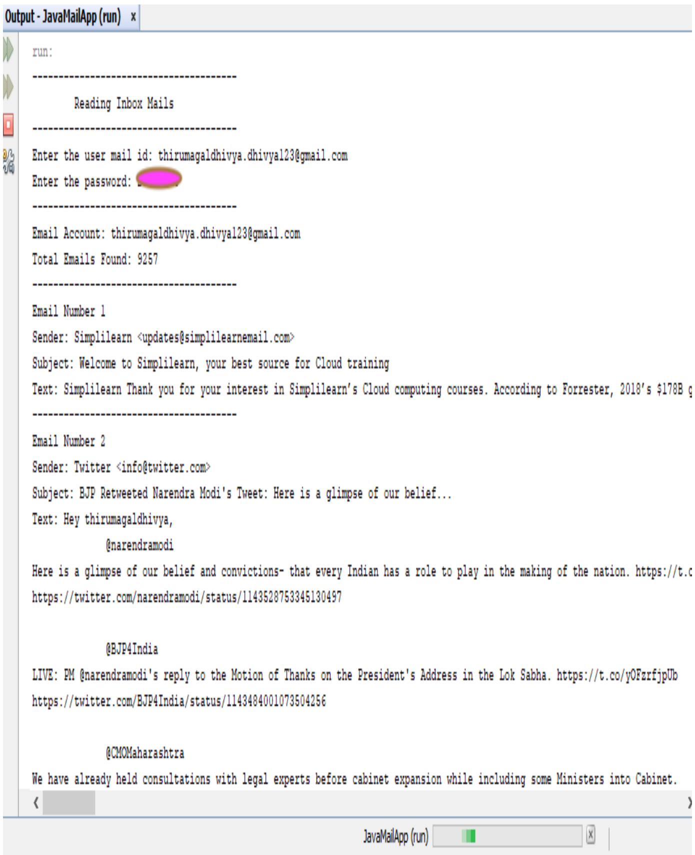
```
Ham Freuquencies
('<s>', 'Cash')  occurs  1
('Cash', 'offer')  occurs  1
('offer', '</s>')  occurs  1


Spam Freuquencies
('<s>', 'Cash')  occurs  1
('Cash', 'offer')  occurs  1
('offer', '</s>')  occurs  1


Ham Probability: 5.211670548750434e-15
Spam Probability: 2.2135452526507885e-13


"Cash offer" is a Spam message
```

Fig 4.11 Probability of spam, ham

C. Testing With Dynamic Inputs

NLP N-Grams model is tested with dynamic inputs. Java Mail API is used to extract the mail messages of the user. This works on cross platforms like Gmail, Yahoo, Live Mail. This is done by enabling the less secure apps in Google accounts of the user and by enabling the POP / IMAP protocol. Number of mails taken is 9257. These mails are read and written into the csv file which is used for testing. This testing with dynamic input is shown in the figure 4.12.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)



```
run:
---------------------------------------
      Reading Inbox Mails
---------------------------------------
Enter the user mail id: thirumagaldhivya.dhivya123@gmail.com
Enter the password: [redacted]
---------------------------------------
Email Account: thirumagaldhivya.dhivya123@gmail.com
Total Emails Found: 9257
---------------------------------------
Email Number 1
Sender: Simplilearn <updates@simplilearnemail.com>
Subject: Welcome to Simplilearn, your best source for Cloud training
Text: Simplilearn Thank you for your interest in Simplilearn's Cloud computing courses. According to Forrester, 2018's $178B g
---------------------------------------
Email Number 2
Sender: Twitter <info@twitter.com>
Subject: BJP Retweeted Narendra Modi's Tweet: Here is a glimpse of our belief...
Text: Hey thirumagaldhivya,
          @narendramodi
Here is a glimpse of our belief and convictions- that every Indian has a role to play in the making of the nation. https://t.c
https://twitter.com/narendramodi/status/1143528753345130497


          @BJP4India
LIVE: PM @narendramodi's reply to the Motion of Thanks on the President's Address in the Lok Sabha. https://t.co/yOFzrfjpUb
https://twitter.com/BJP4India/status/1143484001073504256


          @CMOMaharashtra
We have already held consultations with legal experts before cabinet expansion while including some Ministers into Cabinet.
```

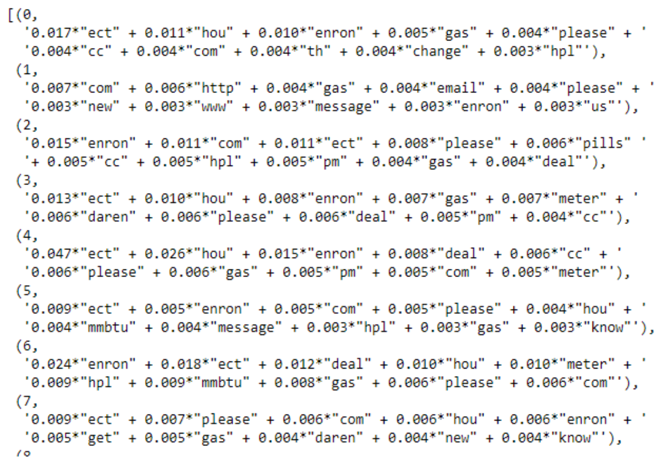
Fig 4.12 Testing with Dynamic inputs

D. *Topic Modelling*

Topic modelling is the type of statistical modelling and this is used to discover the topics which occur in a set of documents. Topic modelling is used for document clustering, feature selection, and information retrieval from the unstructured text.

1) *LDA based Topic Modelling*

Latent Dirichlet Allocation is a type of topic modelling technique used to classify the topics for the documents. Here, the frequently occurring words and their probabilities are displayed. This LDA based topic modelling is shown in the figure 4.13.



```
[(0,
  '0.017*"ect" + 0.011*"hou" + 0.010*"enron" + 0.005*"gas" + 0.004*"please" + '
  '0.004*"cc" + 0.004*"com" + 0.004*"th" + 0.004*"change" + 0.003*"hpl"'),
 (1,
  '0.007*"com" + 0.006*"http" + 0.004*"gas" + 0.004*"email" + 0.004*"please" + '
  '0.003*"new" + 0.003*"www" + 0.003*"message" + 0.003*"enron" + 0.003*"us"'),
 (2,
  '0.015*"enron" + 0.011*"com" + 0.011*"ect" + 0.008*"please" + 0.006*"pills" '
  '+ 0.005*"cc" + 0.005*"hpl" + 0.005*"pm" + 0.004*"gas" + 0.004*"deal"'),
 (3,
  '0.013*"ect" + 0.010*"hou" + 0.008*"enron" + 0.007*"gas" + 0.007*"meter" + '
  '0.006*"daren" + 0.006*"please" + 0.006*"deal" + 0.005*"pm" + 0.004*"cc"'),
 (4,
  '0.047*"ect" + 0.026*"hou" + 0.015*"enron" + 0.008*"deal" + 0.006*"cc" + '
  '0.006*"please" + 0.006*"gas" + 0.005*"pm" + 0.005*"com" + 0.005*"meter"'),
 (5,
  '0.009*"ect" + 0.005*"enron" + 0.005*"com" + 0.005*"please" + 0.004*"hou" + '
  '0.004*"mmbtu" + 0.004*"message" + 0.003*"hpl" + 0.003*"gas" + 0.003*"know"'),
 (6,
  '0.024*"enron" + 0.018*"ect" + 0.012*"deal" + 0.010*"hou" + 0.010*"meter" + '
  '0.009*"hpl" + 0.009*"mmbtu" + 0.008*"gas" + 0.006*"please" + 0.006*"com"'),
 (7,
  '0.009*"ect" + 0.007*"please" + 0.006*"com" + 0.006*"hou" + 0.006*"enron" + '
  '0.005*"get" + 0.005*"gas" + 0.004*"daren" + 0.004*"new" + 0.004*"know"'),
 (8
```

Fig 4.13 Top 10 Frequently occurring words

4.2) *Visualization of LDA model*

pyLDAvis is the interactive LDA visualization python package. The circle area shows how important a topic is over the entire corpus. The similarity between the given topics is identified by how close a circle is with another circle. Top 30 most relevant terms are listed on the right side in the histogram for each of the topics. This visualization of LDA is shown in the figure 4.14.

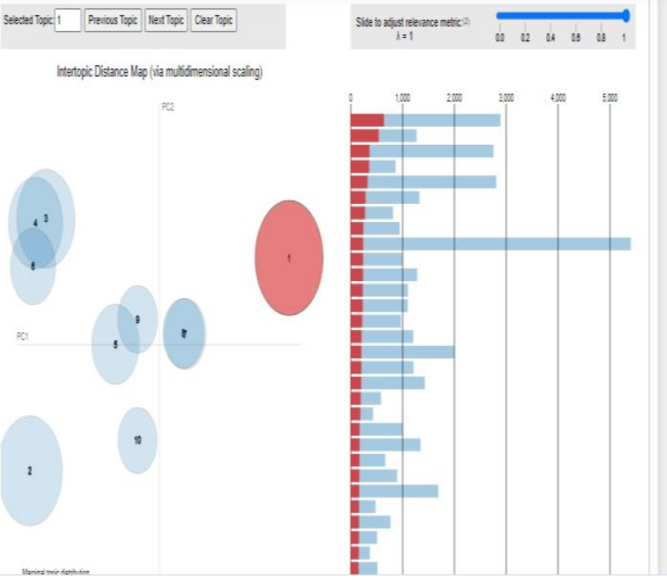
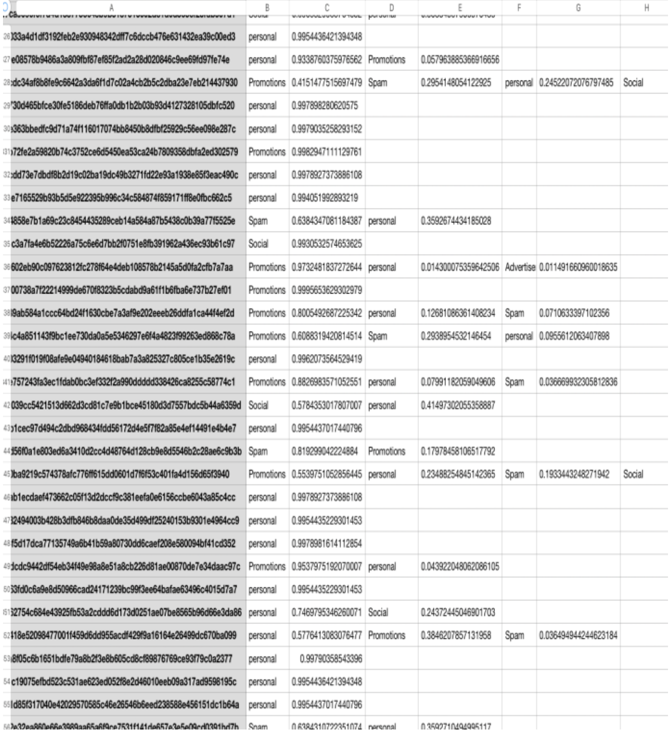


Fig 4.14 LDA visualization using pyLDAvis

4.3) *Dynamic Topic Distribution*

Dynamic topics are given to the LDA model and tested with the real time dataset and the topics are classified and written into the csv file along with the probability of the topics. Each mail message is given with an id and it maps to the mail message. This dynamic topic distribution is shown in the figure 4.15.



	A	B	C	D	E	F	G	H
25	[illegible]	Social	[illegible]	personal	[illegible]			
26	33a4d1df3192feb2e930948342dff7c6dccb476e631432ea39c00ed3	personal	0.9954436421394348					
27	e08578b9486a3a809fbf87ef85f2ad2a28d020846c9ee69fd97fe74e	personal	0.9338760375976562	Promotions	0.057963885366916656			
28	dc34af8b8fe9c6642a3da6f1d7c02a4cb2b5c2dba23e7eb214437930	Promotions	0.4151477515697479	Spam	0.2954148054122925	personal	0.24522072076797485	Social
29	'30d465bfce30fe5186deb76ffa0db1b2b03b93d4127328105dbfc520	personal	0.997898280620575					
30	363bbedfc9d71a74f116017074bb8450b8dfbf25929c56ee098e287c	personal	0.9979035258293152					
31	72fe2a59820b74c3752ce6d5450ea53ca24b7809358dbfa2ed302579	Promotions	0.9982947111129761					
32	dd73e7dbdf8b2d19c02ba19dc49b3271fd22e93a1938e85f3eac490c	personal	0.9978927373886108					
33	e7165529b93b5d5e922395b996c34c584874f859171ff8e0fbc662c5	personal	0.994051992893219					
34	4858e7b1a69c23c8454435289ceb14a584a87b5438c0b39a77f5525e	Spam	0.6384347081184387	personal	0.3592674434185028			
35	c3a7fa4e6b52226a75c6e6d7bb2f0751e8fb391962a436ec93b61c97	Social	0.9930532574653625					
36	602eb90c097623812fc278f64e4deb108578b2145a5d0fa2cfb7a7aa	Promotions	0.9732481837272644	personal	0.014300075359642506	Advertise	0.011491660960018635	
37	00738a7f22214999de670f8323b5cdabd9a61f1b6fba6e737b27ef01	Promotions	0.9995653629302979					
38	9ab584a1ccc64bd24f1630cbe7a3af9e202eeeb26ddfa1ca44f4ef2d	Promotions	0.8005492687225342	personal	0.12681086361408234	Spam	0.0710633397102356	
39	c4a851143f9bc1ee730da0a5e5346297e6f4a4823f99263ed868c78a	Promotions	0.6088319420814514	Spam	0.2938954532146454	personal	0.0955612063407898	
40	3291f019f08afe9e04940184618bab7a3a825327c805ce1b35e2619c	personal	0.9962073564529419					
41	757243fa3ec1fdab0bc3ef332f2a990ddddd338426ca8255c58774c1	Promotions	0.8826983571052551	personal	0.07991182059049606	Spam	0.036669932305812836	
42	039cc5421513d662d3cd81c7e9b1bce45180d3d7557bdc5b44a6359d	Social	0.5784353017807007	personal	0.41497302055358887			
43	1cec97d494c2dbd968434fdd56172d4e5f7f82a85e4ef14491e4b4e7	personal	0.9954437017440796					
44	56f0a1e803ed6a3410d2cc4d48764d128cb9e8d5546b2c28ae6c9b3b	Spam	0.819299042224884	Promotions	0.17978458106517792			
45	ba9219c574378afc776ff615dd0601d7f6f53c401fa4d156d65f3940	Promotions	0.5539751052856445	personal	0.23488254845142365	Spam	0.1933443248271942	Social
46	ab1ecdaef473662c05f13d2dccf9c381eefa0e6156ccbe6043a85c4cc	personal	0.9978927373886108					
47	32494003b428b3dfb846b8daa0de35d499df25240153b9301e4964cc9	personal	0.9954435229301453					
48	f5d17dca77135749a6b41b59a80730dd6caef208e580094bf41cd352	personal	0.9978981614112854					
49	dcdc9442df54eb34f49e98a8e51a8cb226d81ae00870de7e34daac97c	Promotions	0.9537975192070007	personal	0.043922048062086105			
50	3fd0c6a9e8d50966cad24171239bc99f3ee64bafae63496c4015d7a7	personal	0.9954435229301453					
51	2754c684e43925fb53a2cddd6d173d0251ae07be8565b96d66e3da86	personal	0.7469795346260071	Social	0.24372445046901703			
52	418e52098477001f459d6dd955acdf429f9a16164e26499dc670ba099	personal	0.5776413083076477	Promotions	0.3846207857131958	Spam	0.036494944244623184	
53	8f05c6b1651bdfe79a8b2f3e8b605cd8cf89876769ce93f79c0a2377	personal	0.99790358543396					
54	c19075efbd523c531ae623ed052f8e2d46010eeb09a317ad9598195c	personal	0.9954436421394348					
55	d85f317040e42029570585c46e26546b6eed238588e456151dc1b64a	personal	0.9954437017440796					
56	2e32ea860e66e3989aa65a6f9ce7531f141de657e3e5e09cd0391bd7b	Spam	0.6384310722351074	personal	0.3592710494995117			

Fig 4.15 Dynamic Topic Distribution using LDA

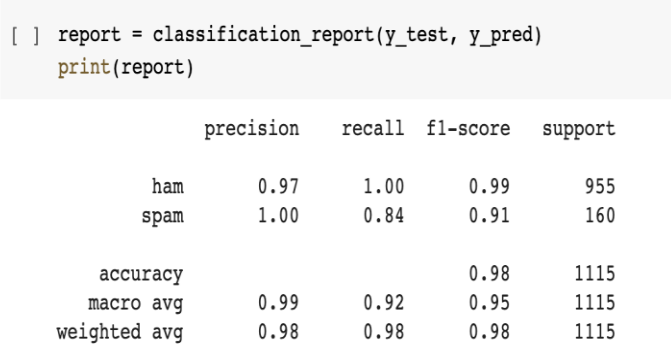
(This work is licensed under a Creative Commons Attribution 4.0 International License.)

V. PERFORMANCE EVALUATION

In this chapter, a classification report for spam detection and accuracy is discussed and also HeatMap visualization is done. Email storage optimization has also been carried out and storage efficiency is understood.

A. Classification Report For Spam Detection

955 hams and 138 spams were predicted correctly 0 hams were incorrectly identified as spams and 22 spams were incorrectly predicted as hams is shown in figure 5.1. Accuracy of 98 percent is obtained in the n-grams model.



```
[ ]  report = classification_report(y_test, y_pred)
     print(report)
```

```
              precision    recall  f1-score   support

         ham       0.97      1.00      0.99       955
        spam       1.00      0.84      0.91       160

    accuracy                           0.98      1115
   macro avg       0.99      0.92      0.95      1115
weighted avg       0.98      0.98      0.98      1115
```

Fig 5.1 Classification Report for Spam Detection

1) Heatmap Visualization

Heatmap is the type of data visualization technique which shows the magnitude of the phenomenon as color in two dimensions. Accuracy of ham and spam visualised using this technique is shown in figure 5.2.

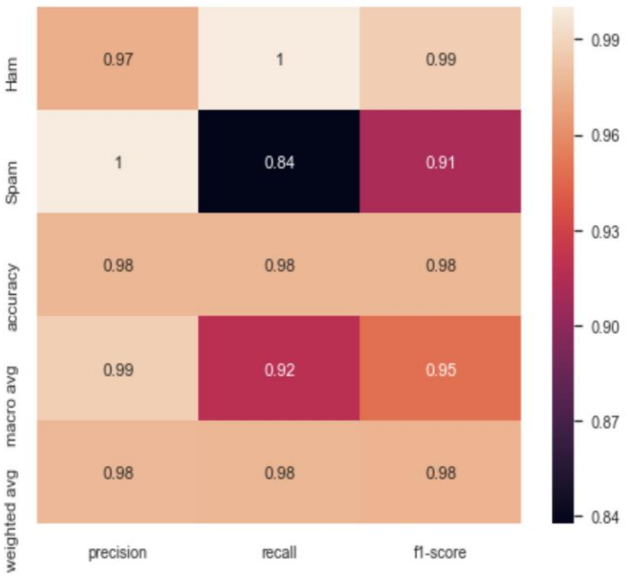


Fig 5.2 HeatMap Visualization for Spam Detection

B. Email Storage Optimization (Eso)

Table 5.1 Email Storage Optimization(ESO)

Test Run 1

Before Email Storage Optimization BES (Mail Size): 2216.9 MB

After Email Storage Optimization AES (Mail Size): 2010.4 MB

Total Savings: ((BES-AES)/BES)*100 → **9.58%**

Test Run 2

Before Email Storage Optimization BES (Mail Size): 967 MB

After Email Storage Optimization AES (Mail Size): 902 MB

Total Savings: ((BES-AES)/BES)*100 → **6.72%**

Test Run 3

Before Email Storage Optimization BES (Mail Size): 501 MB

After Email Storage Optimization BES (Mail Size): 472 MB

Total Savings: ((BES-AES)/BES)*100 → **5.6%**

Average of Email Optimization

→ (Test Run 1 + Test Run 2 + Test Run 3)/3
→ **7.3%**

Run Levels	Total number of mails	Tested Mail Account	Detected Advertisement and Spam mails	Total Mails After Cleaning Spam mails	Storage Space **Before Email Optimization** (in MB)	Storage Space **After Email Optimization** (in MB)	Savings (in %)
R1	2704	thirumagaldhivya.dhivya123@gmail.com	227	2477	2216.9 MB	2010.4 MB	9.58%
R2	1213	ansihtwhiyna@outlook.com	154	1059	967 MB	902 MB	6.72%
R3	533	srisriramasrikrishna@yahoo.com	34	499	501 MB	472 MB	5.6 %

The topics such as advertisement and spam that are classified using LDA are detected. These topics have been given dynamically. The emails that are classified as advertisement and spam are then deleted using the Java API and the storage optimization is performed. Storage Optimization is calculated by three test runs that have been performed in various platforms such as Gmail, Yahoo, Live Mail. Email Storage Optimization is shown in table 5.1.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

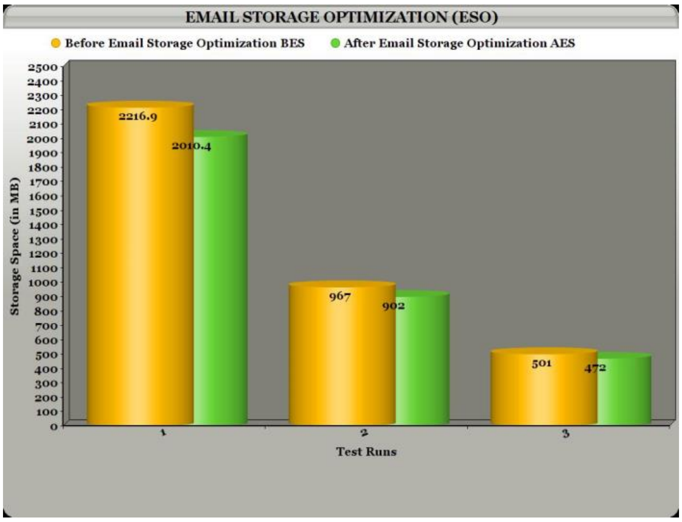


Fig 5.2 Storage Optimization graph

In the first test run, 227 mails were detected as advertisements and spam among the 2704 mails. 206.5MB storage has been saved and the percentage is 9.58%. In the second test run, 154 mails were detected as advertisements and spam among the 1213 mails. 65MB storage has been saved and the percentage is 6.72%. In the third test run, 34 mails were detected as advertisements and spam among the 533 mails. 29 MB storage have been saved and the percentage is 5.6%. The average of storage optimization achieved in these three test runs is 7.3%. This is shown in the figure 5.2.

C. *Comparison Of Proposed With Existing Models*

The results obtained from the end-to-end framework showed good improvement when compared to the existing system. Table 5.2 shows the result comparison between the existing and the proposed system. The accuracy obtained in spam detection is 96.4% in the existing system and in the proposed system it is improved as 98%. Dynamic Inputs have been given and tested in the proposed system by retrieving the personal mail inputs whereas in the existing system, it is not performed. The proposed system works on cross platforms such as Gmail, Yahoo, Live Mail. In respect to storage savings, the existing system only detects the spam mails. But the proposed system detects and also deletes the spam and advertisement mails and therefore storage is saved. The average storage saved is 7.3%.

Table 5.2 Result Comparison

Task	Existing System	Proposed System
Accuracy	96.4%	98%
Dynamic Inputs	No	Yes
Cross Platform	No	Yes
Storage Optimization	Detected, not deleted	Detected and Deleted Average storage saved- 7.3%

Fig 5.3 shows the accuracy improvement of the proposed system with the existing two systems: Email Spam Detection using integrated approach of Naïve Bayes and Particle Swarm Optimization which shows 96.4% accuracy, Content Based Spam Email Filtering which shows 92.8% accuracy. Our proposed system shows 98 percent accuracy.

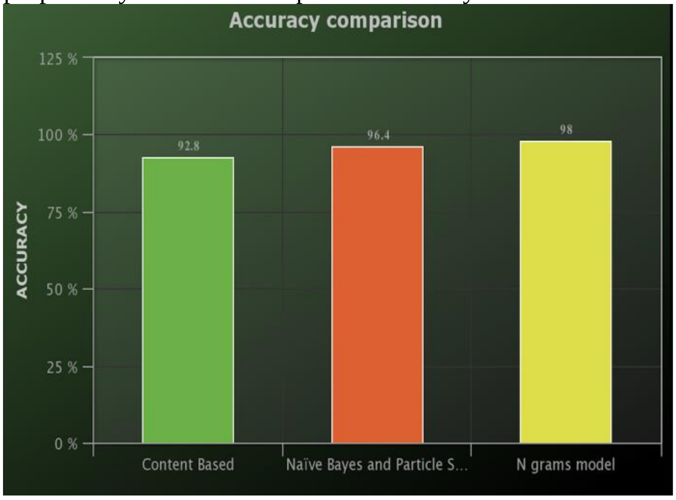


Fig 5.3 Accuracy Comparison Graph

VI. CONCLUSION AND FUTURE WORK

In this thesis, NLP based approaches for spam detection and data optimization have been devised. This chapter also mentions the future works that can be carried out.

A. *Conclusion*

This proposed system has taken emails as input and gives output as in the form of classification of the ‘Spam’ or ‘Ham’ and also data optimization. The input that is given to the model is obtained dynamically using a java mail application that takes the mail id and password for access purposes and gives the output a csv file of the mails in the inbox. The mail which is taken as input is given to the Latent Dirichlet Allocation(LDA). This is a generative probabilistic model, this is also the most popular Topic Modelling Approach. This is used to classify the emails taken as input into different appropriate topics. Using this algorithm the topics are classified as advertisements which are deleted from the extracted mail inbox resulting in data optimization. The next part in this model is the OpinionRank Trustworthy and NLP N-grams model. The OpinionRank algorithm obtains the websites from the emails and this is fed to the algorithm to find the trustworthiness of the website. In the OpinionRank algorithm the seed selection method is called the HarMean PageRank algorithm. For this HarMean PageRank algorithm the seed selection is done using the High PageRank and Inverse PageRank combined. The harmonic mean is found for the High PageRank and Inverse PageRank to determine the trustworthiness of this website. The NLP N-grams model is the next part in the model. Here the bigram model is used. The input file containing the emails are given to the model which performs a preliminary step of data preparation where the Data Cleaning, Data Preprocessing, Lemmatization, Tokenization is done. TF-IDF Vectorization is found for the words and stored in a document matrix. Then the percentage

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

of punctuation to that of the sentence is found. These above procedures are performed to detect whether the given mail is ‘Spam’ or ‘Ham’. Thereby mail data optimization is achieved by Topic classification using LDA and deleting the classified advertisement and spam emails. The percentage of data saved is 7.3%.

B. Future Work

This model could be modified to work on the sender side instead of the receiver side, this way the network traffic could be reduced and the data storage can be reduced. Also the email IDs could have a ranking system, using this way also the above mentioned problems could be overcome. The other methods can be that instead of the whole message being stored for analysis only the header, the attachments and the links could be analyzed. Using the before mentioned point the privacy of an individual could be maintained or a way to encrypt the confidential texts that are chosen by the sender could be employed. For more accuracy the dataset of the model could be updated for the latest trends i.e., the spam and advertisements can vary on the current trends that the society is boosting at the time which will be used more to attract people by scammers.

REFERENCES

[1] A Sharaff and Srinivasarao U (2020), "Towards classification of email through selection of informative features," First International Conference on Power, Control and Computing Technologies (ICPC2T), Raipur, India, pp. 316-320, DOI: 10.1109/ICPC2T48082.2020.9071488.

[2] Adebayo Abayomi-Alli, Modupe Odusami, Olusola Abayomi-Alli and Sanjay Misra (2019), “A review of soft techniques for SMS classification: methods, approaches and applications”, Engineering Applications of Artificial Intelligence, vol. 86, pp. 197-212, DOI: 10.1016/j.engappai.2019.08.024.

[3] Ajay Sharma and Harpeet Kaur (2016), “Improved email spam classification method using integrated particle swarm optimization and decision tree.” In Next Generation Computing Technologies (NGCT), 2nd International Conference on pp. 516-521, DOI: 10.1109/NGCT.2016.7877470.

[4] Akanksha Sharaff, Abhishek Dhadse and Naresh Kumar Nagwani (2016), “Comparative study of classification algorithms for spam email detection,” in Emerging Research in Computing, Information, communication and applications, pp. 237-244, Springer, Berlin, Germany, DOI: 10.1007/978-81-322-2553-9_23.

[5] Alazab M, Azam S, Kannoorpatti K, Karim A and Shanmugam B (2019), "A Comprehensive Survey for Intelligent Spam Email Detection," in IEEE Access, vol. 7, pp. 168261-168295, DOI: 10.1109/ACCESS.2019.2954791.

[6] Alfandi O, Dahmani N and Kaddoura S (2020), "A Spam Email Detection Mechanism for English Language Text Emails Using Deep Learning Approach", IEEE 29th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Bayonne, France, pp. 193-198, DOI: 10.1109/WETICE49692.2020.00045.

[7] Amin Ul Haq, Luo GuangJun, Shah Nazir,Habib and Ullah Khan (2020), “Spam Detection Approach for Secure Mobile Message Communication Using Machine Learning Algorithms”, Security and Communication Networks Volume 2020, pp. 1-6, Article ID 8873639, DOI 10.1155/2020/8873639.

[8] Amin, Hossain N and Rahman M.M (2019), "A Bangla Spam Email Detection and Datasets Creation Approach based on Machine Learning Algorithms," 2019 3rd International Conference on Electrical, Computer & Telecommunication Engineering (ICECTE), Rajshahi, Bangladesh, 2019, pp. 169-172, DOI: 10.1109/ICECTE48615.2019.9303525.

[9] F. Ahmadi-Abkenari, P. Bayat and S. JamshidiNejad (2020), "Opinion Spam Detection based on Supervised Sentiment Analysis Approach," 10th International Conference on Computer and Knowledge Engineering (ICCKE), Mashhad, Iran, pp. 209-214, DOI: 10.1109/ICCKE50421.2020.9303677.

[10] Fang Y, Gao W, Zhang F and Zhang B (2020), "Enhancing Short Text Topic Modeling with FastText Embeddings," International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), Fuzhou, China, pp. 255-259, DOI: 10.1109/ICBAIE49996.2020.00060.

[11] Guangchi Liu, Qing Yang and XiaofeiNiu (2020), “OpinionRank: Trustworthy Website Detection using Three Valued Subjective Logic”,IEEETransactions on Big Data, pp. 1-1 DOI 10.1109/TBDATA.2020.2994309.

[12] H. and Cho , H.G., Kim, S.H., Tak (2019) , "Polarized Topic Modeling for User Characteristics in Online Discussion Community", IEEE International Conference on Big Data and Smart Computing (BigComp), Kyoto, Japan, pp. 1-4, DOI: 10.1109/BIGCOMP.2019.8679489.

[13] J. Marseline K.S and Nandhini S (2020), "Performance Evaluation of Machine Learning Algorithms for Email Spam Detection," International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), Vellore, India, pp. 1-4, DOI: 10.1109/ic-ETITE47903.2020.312.

[14] Jayasinghe N, Muslam, M.M., and Raza, M (2021), "A Comprehensive Review on Email Spam Classification using Machine Learning Algorithms," 2021 International Conference on Information Networking (ICOIN), Jeju Island, Korea (South), pp. 327-332, DOI: 10.1109/ICOIN50884.2021.9334020.

[15] Jun Feng, Jiamin Lu, Xiaodong Li, Xi Zou and Yuelong Zhu (2019) , "A Novel Hierarchical Topic Model for Horizontal Topic Expansion With Observed Label Information," in IEEE Access, vol. 7, pp. 184242-184253, DOI: 10.1109/ACCESS.2019.2960468.

[16] Jyoti Prakash Singh, Pradeep Kumar Roy and Snehasish Banerjee (2019), “Deep learning to filter SMS spam,” Future Generation computer Systems, vol .102, pp. 524-533, DOI: 10.1016/j.future.2019.09.001.

[17] Lekha J, Maheshwaran J, Manikandan A, Murthy K Surya, Prathap K Ram and Tharani K (2019), "Efficient Detection of Spam Messages Using OBF and CBF Blocking Techniques," 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2019, pp. 1175-1179, DOI: 10.1109/ICOEI.2019.8862542.

[18] Mahantesh N Birje and Praveen S Challagidad (2019), "Determination of Trustworthiness of Cloud Service Provider and Cloud Customer," 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, pp. 839-843, DOI: 10.1109/ICACCS.2019.8728408.

[19] Oguz Emre Kural and Sercan Demirci (2020), "Comparison of Term Weighting Techniques in Spam SMS Detection," 28th Signal Processing and Communications Applications Conference (SIU), Gaziantep, Turkey, 2020, pp. 1-4, DOI: 10.1109/SIU49456.2020.9302315.

[20] Qindong Sun, Yaling Zhang and Zhihai Yang (2020), "Probabilistic Inference and Trustworthiness Evaluation of Associative Links toward Malicious Attack Detection for Online Recommendations," in IEEE Transactions on Dependable and Secure Computing, pp. 1-1 DOI: 10.1109/TDSC.2020.3023114.

[21] R. Abinaya, P. Naveen and B. Niveda E (2020), "Spam Detection On Social Media Platforms", 7th International Conference on Smart Structures and Systems (ICSSS), Chennai, India, pp. 1-3, DOI: 10.1109/ICSSS49621.2020.9201948.

[22] Sana Ajaz, Md. Tabrez Nafis and Vishal Sharma (2017), “Spam Mail Detection Using Hybrid Secure Hash Based Naive Classifier”, International Journal of Advanced Research in Computer Science, Vol. 8, No. 5, pp.1195-1199.

[23] Santoshi Kumari and Syed Mohammed Anas (2021) , "Opinion Mining based Fake Product review Monitoring and Removal System", 6th International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, pp. 985-988, DOI: 10.1109/ICICT50816.2021.9358716.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

[24] Satish Khumbar and Shounaak Ughade (2019), "Survey on Mathematical Word Problem Solving Using Natural Language Processing," 1st International Conference on Innovations in Information and Communication Technology (ICIICT), Chennai, India, pp. 1-5, DOI: 10.1109/ICIICT1.2019.8741437.

[25] X. Chang, X. Wang and Y. Wang (2020), "Sentiment Analysis of Consumer-Generated Online Reviews of Physical Bookstores Using Hybrid LSTM-CNN and LDA Topic Model," International Conference on Culture-oriented Science & Technology (ICCST), Beijing, China, 2020, pp. 457-462, DOI: 10.1109/ICCST50977.2020.00094.

[26] Xuejun Yu and Yukun Tian, (2021), "Trustworthiness study of HDFS data storage based on trustworthiness metrics and KMS encryption," IEEE International Conference on Power Electronics, Computer Applications (ICPECA), Shenyang, China, pp. 962-966, DOI: 10.1109/ICPECA51329.2021.9362537.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)