Spam Detection using Machine Learning and Natural Language Processing

A Project Report

submitted in partial fulfillment of the requirements

of

AICTE Internship on AI: Transformative Learning
with
TechSaksham – A joint CSR initiative of Microsoft & SAP

by

Name of Student : Gaddam Raghu Varma

Email id: raghuvarma17633@gmail.cm

Under the Guidance of

Name of Guide: Abdul Aziz Md

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to the for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We would like to take this opportunity to express our deep sense of gratitude to all individuals who helped us directly or indirectly during this thesis work.

Firstly, we would like to thank my supervisor Abdul Aziz Md, for being a great mentor and the best adviser I could ever have. His advice, encouragement and the critics are a source of innovative ideas, inspiration and causes behind the successful completion of this project. The confidence shown in me by him was the biggest source of inspiration for me. It has been a privilege working with him for the last one year. He always helped me during my project and many other aspects related to the program. His talks and lessons not only help in project work and other activities of the program but also make me a good and responsible professional.

we wish to express our thanks to all Teaching and Non-teaching staff members of the , who were helpful in many ways for the completion of the project.

ABSTRACT

Nowadays, all the people are communicating official information through emails. Spam mail is the major issue on the internet. It is easy to send an email which contains spam messages by the spammers. Spam fills our inbox with several irrelevant emails. Spammers can steal sensitive information from our device like files, contact.

Even though we have the latest technology, it is challenging to detect spam emails. This paper aims to propose a Term Frequency Inverse Document Frequency (TFIDF) approach by implementing the Support Vector Machine algorithm. The results are compared in terms of the confusion matrix and accuracy. and precision. This approach gives an accuracy of 99.9% on training data and 98.2% on testing data achieved by using the Term Frequency Inverse Document Frequency (TFIDF) based Support Vector Machine (SVM) system.

KEYWORDS : Ham/spam, Natural Language Processing, Machine Learning, Online Platform, Email.

TABLE OF CONTENT

CHAPTER 1

Introduction

Today, Spam has become a major problem in communication over internet. It has been accounted that around 55% of all emails are reported as spam and the number has been growing steadily. Spam which is also known as unsolicited bulk email has led to the increasing use of email as email provides the perfect ways to send the unwanted advertisement or junk newsgroup posting at no cost for the sender. This chances has been extensively exploited by irresponsible organizations and resulting to clutter the mail boxes of millions of people all around the world.

Spam has been a major concern given the offensive content of messages, spam is a waste of time. End user is at risk of deleting legitimate mail by mistake. Moreover, spam also impacted the economical which led some countries to adopt legislation.

Text classification is used to determine the path of incoming mail/message either into inbox or straight to spam folder. It is the process of assigning categories to text according to its content. It is used to organized, structures and categorize text. It can be done either manually or automatically. Machine learning automatically classifies the text in a much faster way than manual technique. Machine learning uses pre-labelled text to learn the different associations between pieces of text and it output. It used feature extraction to transform each text to numerical representation in form of vector which represents the frequency of word in predefined dictionary.

Text classification is important to structure the unstructured and messy nature of text such as documents and spam messages in a cost-effective way. Machine learning can make more accurate precisions in real-time and help to improve the manual slow process to much better and faster analysing big data. It is important especially to a company to analyse text data, help inform business decisions and even automate business processes.

In this project, machine learning techniques are used to detect the spam message of a mail. Machine learning is where computers can learn to do something without the need to explicitly program them for the task.

It uses data and produce a program to perform a task such as classification. Compared to knowledge engineering, machine learning techniques require messages that have been successfully pre-classified. The pre-classified messages make the training dataset which will be used to fit the learning algorithm to the model in machine learning studio.

A combination of algorithms are used to learn the classification rules from messages. These algorithms are used for classification of objects of different classes. These algorithms are provided with pre labelled data and an unknown text. After learning from the prelabelled data each of these algorithms predict which class the unknown text may belong to and the category predicted by majority is considered as final.

1.1 Problem Statement:

Spammers are in continuous war with Email service providers. Email service providers implement various spam filtering methods to retain their users, and spammers are continuously changing patterns, using various embedding tricks to get through filtering. These filters can never be too aggressive because a slight misclassification may lead to important information loss for consumer. A rigid filtering method with additional reinforcements is needed to tackle this problem.

1.2 Motivation:

To address the challenges posed by spam, this project aims to develop a more robust and intelligent spam filtering system. This system should:

Accurately Identify Spam: Employ advanced machine learning techniques to classify emails as spam or legitimate with high precision and recall.

Adapt to Evolving Spam Tactics: Continuously learn and update its models to stay ahead of evolving spam techniques.

Minimize False Positives: Reduce the number of legitimate emails mistakenly classified as spam.

1.3 **Objective:**

The objectives of this project are

i. To create a ensemble algorithm for classification of spam with highest possible accuracy.

ii. To study on how to use machine learning for spam detection.

iii. To study how natural language processing techniques can be implemented in spam detection.

iv. To provide user with insights of the given text leveraging the created algorithm and NLP.

1.4 **Scope of the Project:**

This project needs a coordinated scope of work.

i. Combine existing machine learning algorithms to form a better ensemble algorithm.

ii. Clean, processing and make use of the dataset for training and testing the model created.

iii. Analyse the texts and extract entities for presentation.

1.5 Limitations :

This Project has certain limitations.

i. This can only predict and classify spam but not block it.

ii. Analysis can be tricky for some alphanumeric messages and it may struggle with entity detection.

iii. Since the data is reasonably large it may take a few seconds to classify and anlayse the message

CHAPTER 2

Literature Survey

2.1 Review relevant literature or previous work in this domain.

2.2 Mention any existing models, techniques, or methodologies related to the problem.

2.3 Highlight the gaps or limitations in existing solutions and how your project will address them.

This chapter discusses the machine learning literature review classifier that has been used in previous research and projects. The purpose of that is to summarise prior research relevant to this topic rather than to gather information. It entails finding, reading, analysing, summarising, and assessing project-based reading materials. The majority of spam filtering and detection systems require periodic training and updating, according to assessments of the literature on machine learning. Setting up rules is also necessary for spam filtering to begin functioning

Problem Analysis: Email is a kind of communication that uses telecommunication to exchange computer-stored information. Several groups of people as well as individuals receive the emails. Even though email facilitates the sharing of information, spam and junk mail pose a severe threat. Spam messages are unwanted communications that people get and that annoy them and are inundated in their mailboxes. By wasting their time and causing bandwidth problems for ISPs, it irritates email users. Therefore, it is more crucial to identify and categorise incoming email as spam or junk. Thus, a review of earlier studies presenting email detection and classification algorithms is provided in this section.

Spam Detection using Feature Selection Approach:

The research work done by Jieming Yang et al (2011) uses binomial hypothesis testing to do out content- or text-based spam filtering. In this study, the author focuses on feature selection as a means of removing incoming spam emails. The bi-test method verifies whether the emails' contents fit a specific probability of spam email. Six distinct benchmark datasets are used to estimate the performance of the proposed system, and comparisons are done using several feature selection algorithms, including Poisson distribution, Improved Gina Index, information gain, and @g2-statistic.

Using the Support Vector Machine method, the spam emails are then categorised. In 2014, Seyed Mustafa Pourhashemi et al. suggested using a hybrid feature selection approach to detect spam emails. The Chi Square-2 filter, which In this study, the message body's basic contents are filtered using a method that eliminates spam-related characteristics (contents)[3].

The best features from a pool of characteristics are selected to create these filtered contents utilising the wrapper selection approach. Using the SupportVector Machine, Multinomial Naive Bayesian Classifier, Discriminative Naive Bayesian Classifier, and Random Forest classifier, the classification process is finally completed. The proposed filter and wrapper-based spam classification increases the efficiency of detection and lowers error rates. The main problem with the operation is how long it takes to process everything.

Spam Detection using Collaborative Filtering Technique:

The research work is done by Guangxia Li and others. The collaborative filtering-based spam detection is discussed in this section, and the pertinent discussions are explained as follows. They suggested a strategy for filtering spam emails that involved collaborative online multitask learning. The model is created using the entire data set in the proposed method, which aids in connecting the various tasks. The attributes used to distinguish between spam and non-spam email are then learned via the collaborative online technique. Therefore,

The suggested collaborative online approach efficiently categorises the various assignments, but demonstrates the high rejection rate. The self-learning based collaborative filtering that is used to identify spam emails is the topic of Xiao Zhou et al. (2007). With the aid of an improved hash-based technique, this method learns how similar emails are measured before reducing the traffic that spam emails were responsible for creating. As a result, the effectiveness of the system is assessed using the current spam categorization method, however the filtering procedure is time consuming.

Spam Detection using Email Abstraction based Scheme:

The research work is done by Dakhare et al (2013). He proposed using email abstraction to detect spam. Emails are divided into content-based and non-content-based email categories. The email abstraction is created from the HTML data during the spam detection procedure. These data are kept in a tree-structured database, and the matching algorithm is utilised to identify spam emails. The system's performance is then evaluated in comparison to a spam detection method based on the content of web pages using the sensitivity, specificity, precision, accuracy, and recall numbers.

Spam Detection using Neural Networks

This section explains various discussions about neural networks to classify the spam emails. Kumar et al (2015)[11] removes the spam mails from the group of mails using the preprocessing, redundancy removal, and feature selection and classification steps Three main steps—stop word removal, stemming, and tokenization—are used to preprocess the data set in this study. The redundant information is then deleted by employing the vector

quantization method on the preprocessed data. Particle swarm optimization was used to select the best features from the non-redundant data, which were then used for classification. Finally, Probabilistic Neural Networks handle the classification.

METHODOLOGY:

The era of creation of this product includes models i.e., object-oriented model, Prototype model, waterfall model etc. for making the correct system. water model, the oldest model of creation of correct system. The product model used by our framework is the cascade model. Cascade model could be a precise and successive

way to contend with the merchandise improvement. This incorporates framework coming up with and displaying that sets up requirements for all the framework parts and distributes some set of those conditions to programming. Framework building and examination incorporate requirement gathering at the framework level with modest amount of top-ranking arrange. Examination info building consolidate would like assortment at the key business level and at the business space level

EXISTING SYSTEM:

Email Spam Classifier based on Machine Leaning Techniques had done by using SVM, KNN, Naive Bayes and Decision tree algorithms etc.

SVM had an average accuracy of 99.6%.

It had good accuracy when compared to the other algorithms in proposed system.

PROPOSED SYSTEM:

1. E-mail Spam Classifier is used to classify email data into spam and ham emails.

2. This method is performed by using Support Vector Machine (SVM) algorithm.

3. In this method, dataset is divided into two sets based on labels and given as input to algorithm.

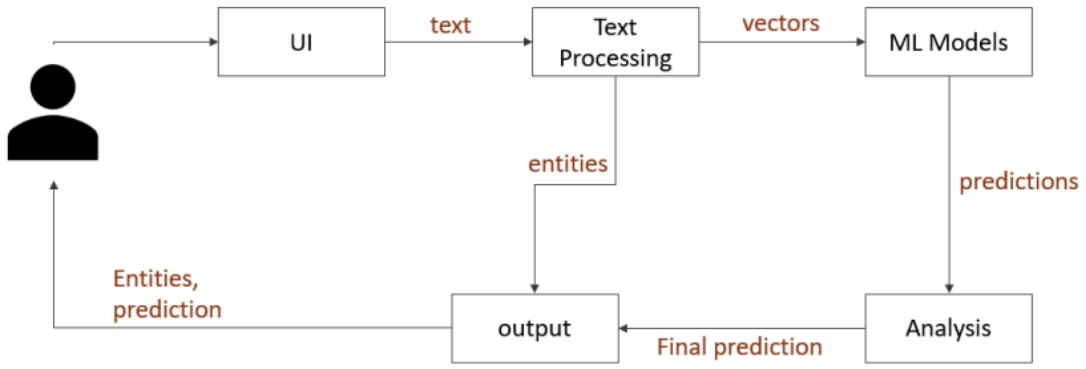
4. The accuracy of 99% on training data and 98.2% on test data is obtained through the proposed system.

CHAPTER 3

Proposed Methodology

3.1 System Design

The application overview has been presented below and it gives a basic structure of the application.



The UI, Text processing and ML Models are the three important modules of this project. Each Module's explanation has been given in the later sections of this chapter.

A more complicated and detailed view of architecture is presented in the workflow section

3.2 Requirement Specification

This incorporates framework coming up with and displaying that sets up requirements for all the framework parts and distributes some set of those conditions to programming. Framework building and examination incorporate requirement gathering at the framework level with modest amount of top-ranking arrange

3.2.1 Hardware Requirements:

PC/Laptop

Ram – 8 Gig

Storage – 100-200 Mb

3.2.2 Software Requirements:

OS – Windows 7 and above

Code Editor – Pycharm, VS Code, Built in IDE

Anaconda environment with packages nltk, numpy, pandas, sklearn, tkinter, nltk data. Supported browser such as chrome, firefox, opera etc..

CHAPTER 4

Implementation and Result

4.1 Snap Shots of Result:

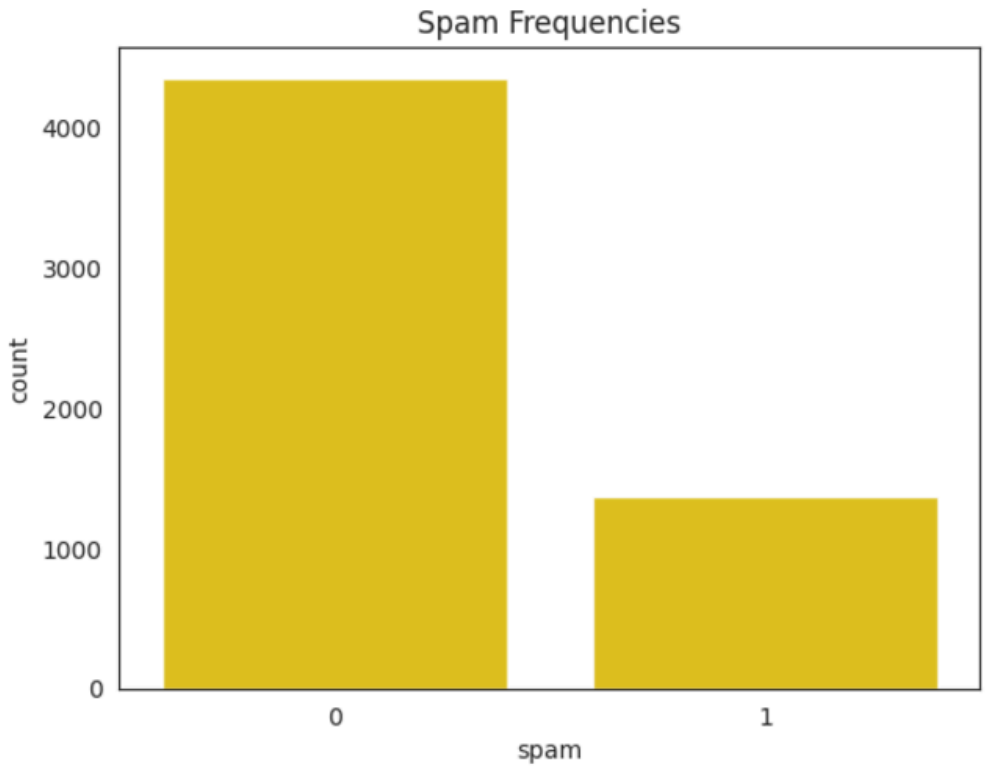
```
[4]  # Load the dataset
     dataset = pd.read_csv('emails.csv')
     dataset.shape
```

```
(5728, 2)
```

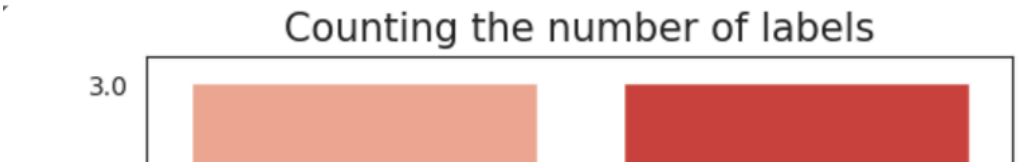
```
     # Show dataset head (first 5 records)
     dataset.head()
```

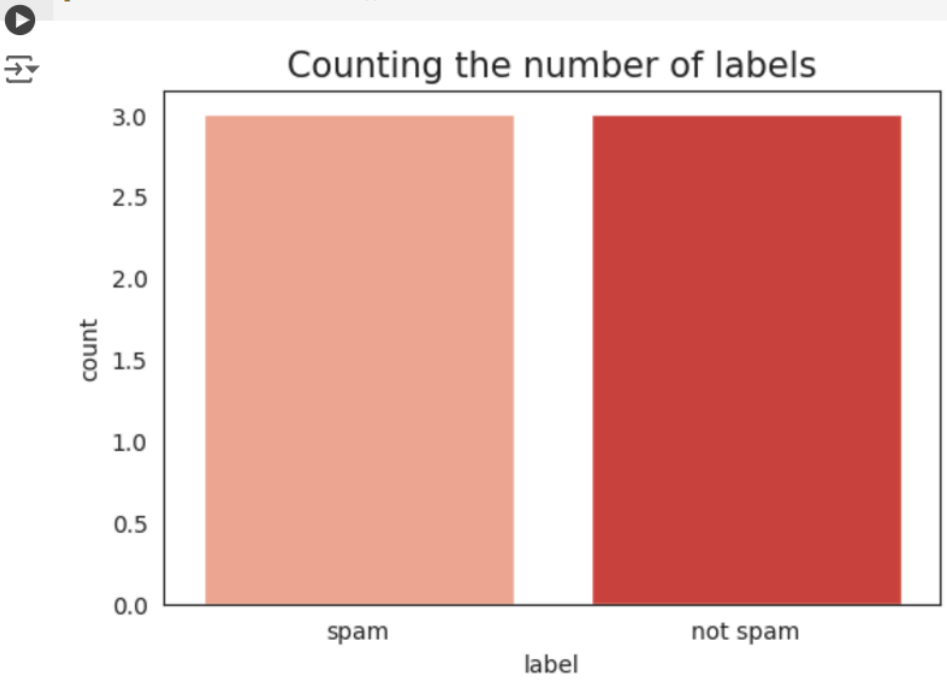
	text	spam
0	Subject: naturally irresistible your corporate...	1
1	Subject: the stock trading gunslinger fanny i...	1
2	Subject: unbelievable new homes made easy im ...	1
3	Subject: 4 color printing special request add...	1
4	Subject: do not have money , get software cds ...	1

```
import seaborn as sns

# Assuming your data is loaded into a DataFrame named 'dataset'
sns.countplot(x="spam", data=dataset, color="gold")  # Use the named color "gold"
plt.title("Spam Frequencies")
plt.show()
```



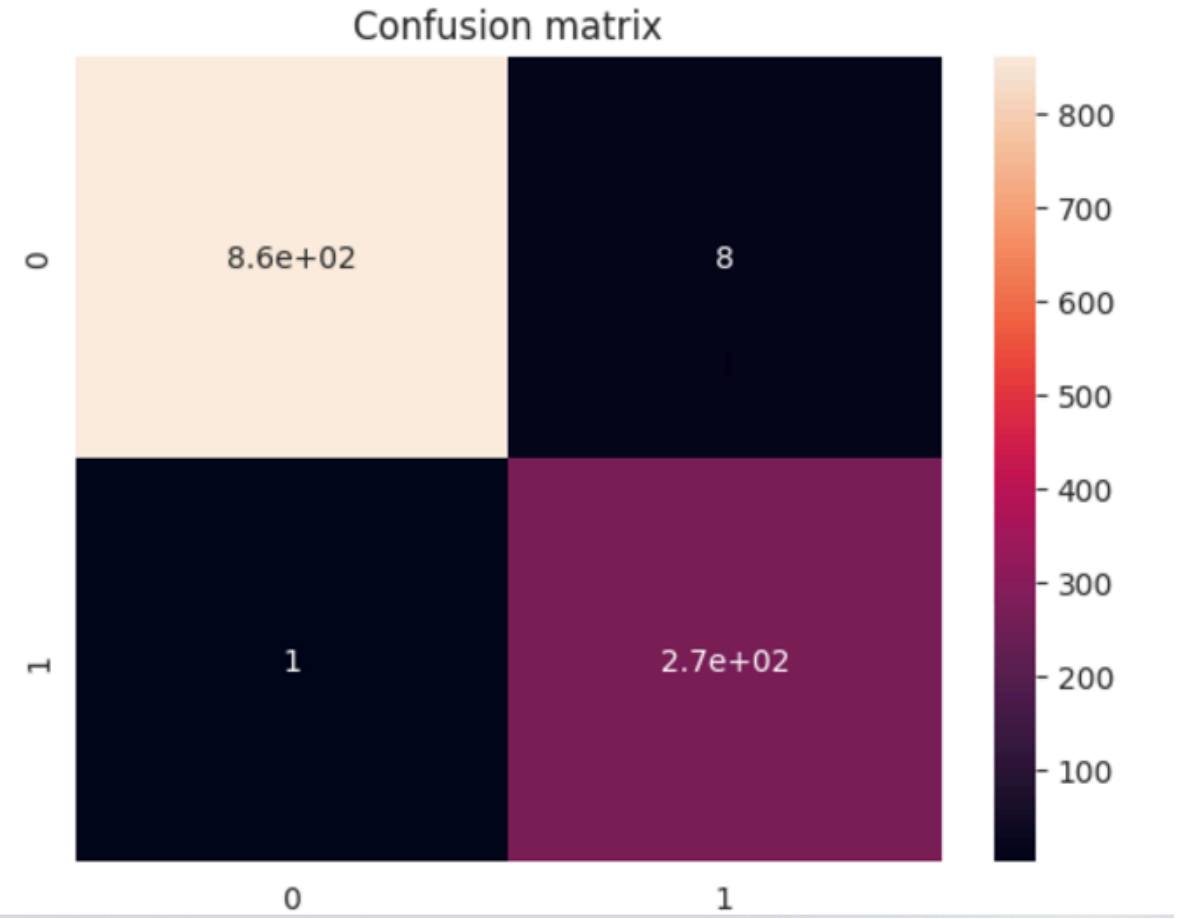
```
import seaborn as sns
import pandas as pd
import matplotlib.pyplot as plt

# Create some sample data
data = {'label': ['spam', 'not spam', 'spam', 'not spam', 'not spam', 'spam']}
df = pd.DataFrame(data)

# Create a countplot of the labels
plt.figure(figsize=(6, 4))
sns.countplot(x="label", data=df, palette="Reds")

# Set the title and rotate x-axis labels for readability
plt.title("Counting the number of labels", fontsize=15)
plt.xticks(rotation='horizontal')
plt.show()

# Print the value counts of the labels
print(df['label'].value_counts())
```






```
label
spam       3
not spam   3
Name: count, dtype: int64
```

```
# Model Evaluation | Confusion matrix
cm = confusion_matrix(y_test, y_pred)
plt.figure(dpi=100)
sns.heatmap(cm, annot=True)
plt.title("Confusion matrix")
plt.show()
```



Kindly provide 2-3 Snapshots which showcase the results and output of your project and after keeping each snap explain the snapshot that what it is representing.

4.2 GitHub Link for Code:

https://github.com/raghuvarm17633/Spam-Email-Classification-using-NLP-and-Machine-Learning/tree/main

CHAPTER 5
Discussion and Conclusion

5.1 Future Work:

The research from this investigation can be expanded upon further recipient-related characteristics that can be added from organisation databases, as well as file level Metadata elements like document path location, author names, and so forth. Additionally, it can broaden multi-class results that connect to a particular recipient. This method is quite helpful for corporate email messaging processes (for instance, a medical email web portal, where a message may belong to more than two folders, and where the strategy of folding processes sends the incoming message to the multiple folder with a specified weighing scheme which will help in classification with more accuracy.

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 abovementioned 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 above-mentioned point, the privacy of an individual could be maintained or encrypting 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.

5.2 Conclusion:

From the results obtained we can conclude that an ensemble machine learning model is more effective in detection and classification of spam than any individual algorithms. We can also conclude that TF-IDF (term frequency inverse document frequency) language model is more effective than Bag of words model in classification of spam when combined with several algorithms. And finally we can say that spam detection can get better if machine learning algorithms are combined and tuned to needs.

REFERENCES

[1]. Ming-Hsuan Yang, David J. Kriegman, Narendra Ahuja, "Detecting Faces in Images: A Survey", IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume. 24, No. 1, 2002.

[2]. Alistair McDonald, "Spam Assassin: A Practical Guide to Integration and Configuration", 1st Edition, Packet publishers, 2004.

[3]. Ian H. Witten, Eibe Frank, "Data Mining – Practical Machine Learning Tools and Techniques," 2nd Edition, Elsevier, 2005.

[4]. Deepika Mallampati, Nagaratna P. Hegde "A MachineLearning Based Email Spam Classification Framework Model" in IJITEE, ISSN: 2278-3075, Vol.9 Issue.4, Feb 2020.

[5]. Javatpoint, "Machine Learning Tutorial" 2017

[6]. Spam Assassin, "Email.cvs Dataset", Kaggle, 2018.https://www.kaggle.com/veleon/ham-andspam-dataset.

[7]. https://spamassassin.apache.org/old/publiccorpus