

**SPAM MAIL DETECTION USING SUPERVISED  
MACHINE LEARNING METHODS**

**SUBMITTED BY**

**S. THAMEENA KAWSAR (20PCS010)**



Project Report Submitted

*In partial fulfillment of the requirements for the Award of*

**MASTER OF SCIENCE IN COMPUTER SCIENCE**

**Department of Computer Science**

**Avinashilingam Institute for Home Science and Higher Education for**

**Women Coimbatore – 641043**

**May – 2022**

## ABSTRACT

Spam emails are usually called junk mail and bulk unsolicited emails are delivered to the inbox. Emails used in advertising are regularly sent to the user via a subscription email that they may not have requested. Users of the spam email problem regularly experience it. Recently, Russia produced the largest share of 23.52 percent of the total spam emails in the world. One event claims that Google has registered 2,145,013 sensitive patent sites since Jan. 17, 2021. This has increased from 1,690,000 in Jan. 19, 2020 (increased by 27% in 12 months). As it is a well-known fact that 91% of all online attacks start with spam emails and about security issues, 97% of users fail to identify spam emails so finding a solution by filtering spam email can help reduce the risk of being a network. threats to business or employees. The "Spam Mail Recovery" project is a model built using a machine-readable learning method. It starts with data collection and continues with the previous processing method using NLP (Indigenous Language Processing Strategy). To select features on the website, the TF-IDF (Term Frequency and Inverse Document Frequency) method and vectorizer feature are used. The main stage is to build a model based on those features using machine learning algorithms such as Decision Tree, Naïve Bayes, Support Vector Class, Logistic Regression, KNearest Neighbor, Random Forest, Ada Boost, Bagging, Extra Tree Classifier, Gradient Boost Classifier and XG Boost all of these algorithms we have used on the website and based on comparative analysis of different machine learning models, the Extra Tree Classifier model provides 98% high accuracy of spam email detection.

***Keywords:* Machine learning, Natural Language Processing, Spam detection, Spam emails, Supervised learning.**

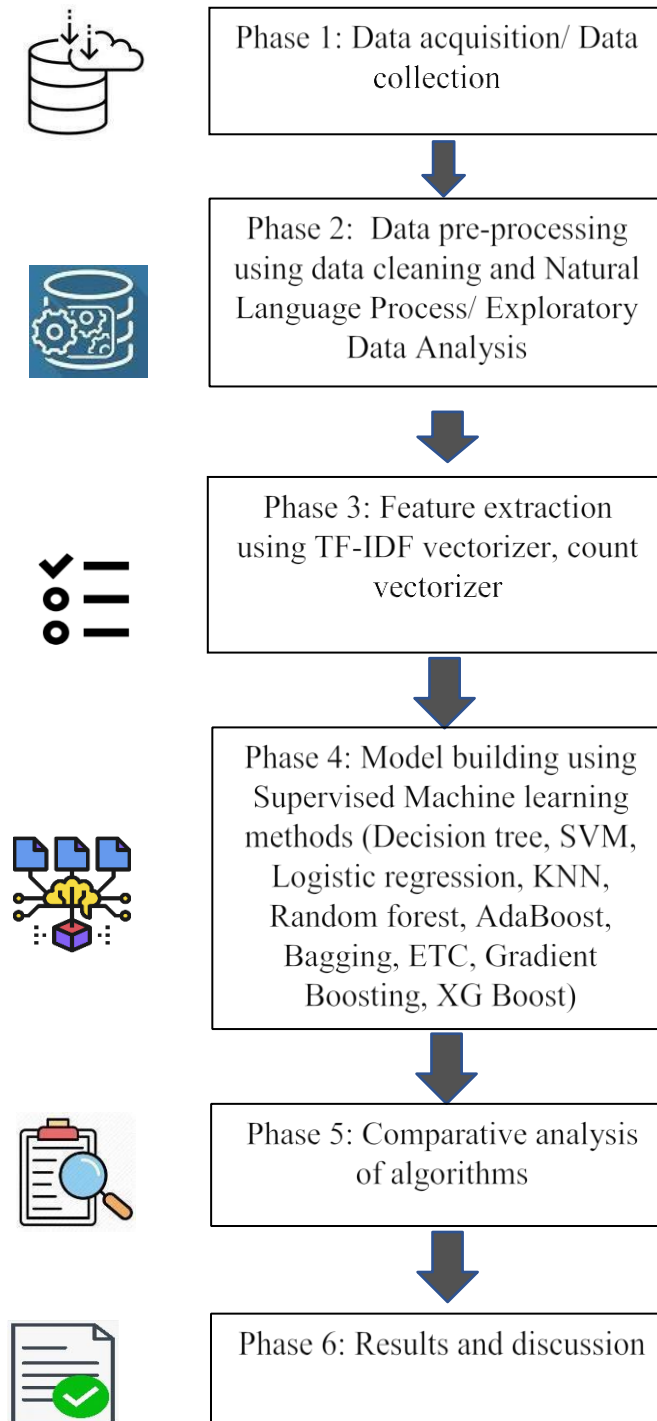
# 1. INTRODUCTION

## 1.1 About the Project

Spam mail detection is the way to protect the personal information from the spammers who attacks by sending malicious links through email which leads them to seek the user's system and are used in illegal and unethical conduct like phishing and fraud. Spam email is a ploy to trick people with little knowledge of these types of attacks. The entire organization has security issues that have deeply troubled users, site developers, and professionals, in order to protect confidential data from this type of social media attack. Email spam can affect every individual in a different way by stealing personal information or even by retrieving the bank details and stealing money.

## 2. METHODOLOGY

The following Figure 4.1 represents the workflow of the methodology implemented in the project.



## Figure 4. Proposed Workflow of the Methodology

The methodology used in the project is divided into six different phases based on machine learning architecture. As per the methodology shown in Figure 4.1, phase 1 explains the data acquisition. The phase 2 describes the data pre-processing technique using two methods data cleaning and Natural Language Processing (NLP) and Exploratory Data Analysis (EDA) for a detailed analysis. Phase 3 focus on the importance of feature extraction method where TF-IDF vectorizer and count vectorizer methods are used, Phase 4 is the model building phase using supervised machine learning methods which contains 11 algorithms. Phase 5, a comparative analysis is performed among the supervised machine learning models. Phase 6 is carried out with the results and discussion.

### 2.1 Data Acquisition/ Data collection

The process of acquiring data from relevant sources before it is saved, cleaned, pre-processed and used in other processes is referred to as "data acquisition." It is the process of acquiring critical business information, converting it into the proper business form, and loading it into the relevant system.

The below Figure 4.2 shows the first phase of the methodology which is dataset acquired for the project.

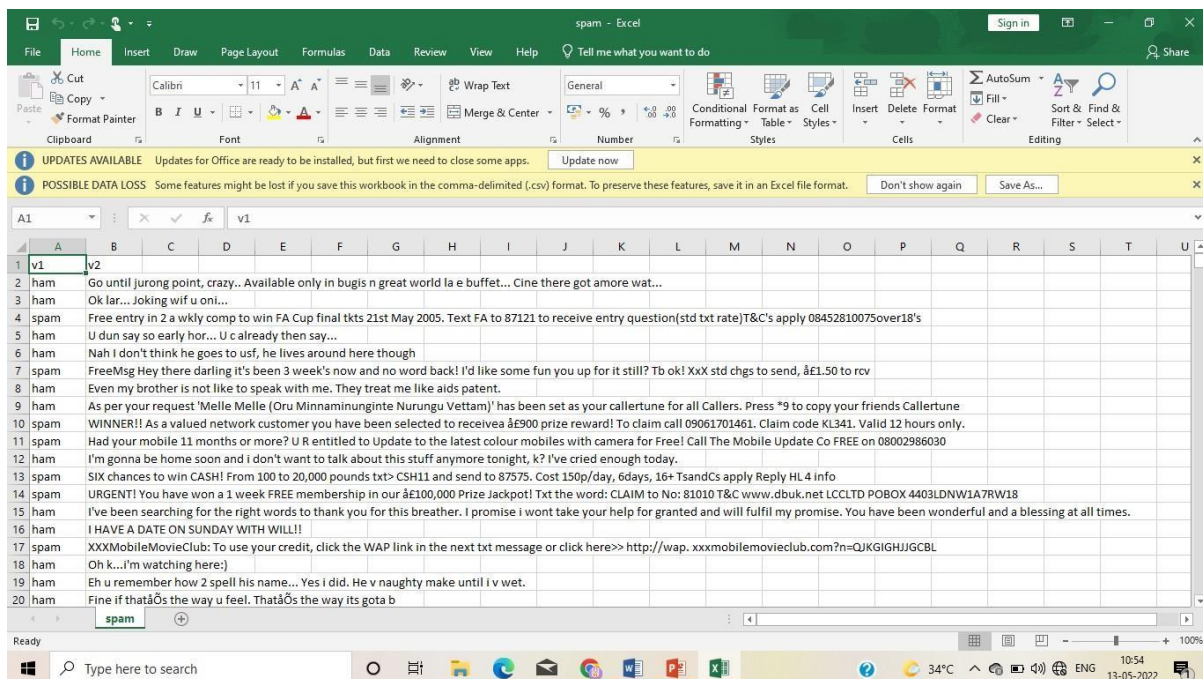


Figure 2.1 A preview of dataset.

### **2.1.1 Dataset attribute description**

The dataset attributes consist of two columns one is the v1 which contains the target information to determine if the email is spam or ham. Second attribute is named as v2 which contains the body of the message with different collection of words in text format.

### **2.1.2 Types of spam email**

The following list of five different spam email categories that can occur in a dataset. Common types of spam mail are:

- Commercial advertisements
- Antivirus warnings
- Email spoofing
- Sweepstakes winners
- Money scams

## **2.2 Data Pre-processing**

Data Pre-processing is an important Data Mining stage that deals with data preparation and modification of the data collection while also attempting to improve the efficiency of knowledge discovery. To put it another way, Data Pre-processing is a step in Data Mining that gives strategies that can help us analyze and uncover knowledge from data at the same time. The data in the real world is frequently incomplete, noisy, and inconsistent. This might result in low-quality data collection and, as a result, low-quality models based on that data. Data Pre-processing provides activities that can organize data into a correct shape for better understanding in the data mining process to address these challenges.

### **2.2.1 Techniques used in Data Pre-processing**

Data Pre-processing techniques that are commonly used in this machine learning project are given below:

- i) Data Pre-processing using Data cleaning/cleansing
- ii) Data Pre-processing using Natural Language Process

## **Data cleansing techniques**

The selection of data cleaning strategies by users is influenced by a variety of factors. First and foremost, what kind of data are you working with? Are they strings or numeric values? Unless the user has a small number of values to deal with, it is unrealistic to expect the user's data to be cleaned using only one technique. For a better result, the user may need to employ various strategies. The more data types you have to deal with, the more purification strategies you'll need to employ. Knowing all of these strategies will aid in the correction of errors and the removal of unnecessary data.

Data cleansing can be done using the following methods

- Remove Irrelevant Values
- Get Rid of Duplicate Values
- Avoid Typos (and similar errors)
- Convert Data Types
- Take Care of Missing Values
  - a. Imputing Missing Values
  - b. Highlighting Missing Values

### **Remove Irrelevant Values**

The first and most important step is to delete any unnecessary data from the user's PC. Any data that is useless or unnecessary is not required. It might not be appropriate in the context of the user's problem. Make sure that a piece of data is irrelevant before removing it because you may need it later to check its linked values (for checking the consistency). The user does not want to erase some values and then come to regret it later. However, once the data has been determined to be irrelevant, get rid of it.

## **Get Rid of Duplicate Values**

Duplicates are similar to useless values in that they are unnecessary. They merely serve to expand the amount of data available to customers while also wasting their time. Simple searches can be used to get rid of them. For a variety of reasons, duplicate values may exist in a user's system. It's possible that data from various sources was blended. Perhaps the data submitter made a mistake and repeated a value. When filling out an online form, some users pressed the 'enter' button twice. Duplicates should be removed as soon as they are discovered.

## **Avoid Typos (and similar errors)**

Typographical errors are caused by human error and can occur anywhere. Multiple algorithms and strategies can be used to correct mistakes. The values can be mapped and converted to the correct spelling. Because models treat various values differently, typos must be corrected. Strings rely heavily on their case and spelling.

## **Convert Data Types**

The data types in each user's dataset should be consistent. A numeric cannot be a string, and a numeric cannot be a Boolean. When it comes to converting data types, there are a few things to keep in mind:

- Keep numeric values that way.
- Determine whether or not a number is a string. It would be incorrect if it was entered as a string.

If a certain data value cannot be converted, enter 'NA value' or something like. Make sure the user also includes a warning to indicate that this value is incorrect.

## **Take Care of Missing Values**

There would always be some lacking information. It is unavoidable. There may be too many missing values in a single column in the user's dataset. In that situation, getting rid of the entire column is a good idea because there isn't enough data to work with. Ignoring missing numbers is a big mistake since it contaminates the user's input and leads to inaccurate results. Missing values can be dealt with in a variety of ways.

### **a. Imputing Missing Values**

Missing values can be imputed, which involves assuming an approximate value. The missing value can be calculated using linear regression or the median. However, this method has drawbacks because it is impossible to know if it is the true value.



## **b. Highlighting Missing Values**

If users' missing values aren't due to chance, it may be advantageous to highlight or report them. For example, because the user's client didn't want to answer the question in the first place, the user's records may not include many replies to a certain question of the user's survey.

## **Natural Language Processing**

Natural Language Processing (NLP) is an area of Artificial Intelligence that analyses, processes, and retrieves text data efficiently. Summarizing texts, title generators, caption generators, fraud detection, speech recognition, recommendation systems, machine translation, and other realworld challenges can all be solved with the help of NLP.

### **Libraries used to deal with NLP**

To deal with NLP-based difficulties, a variety of libraries and algorithms are employed. For text cleaning, a regular expression(re) is the most commonly used library. The next-level libraries NLTK and spacy are used to do natural language tasks such as stopword removal, named entity recognition, part of speech tagging, phrase matching, and so on.

Data Pre-processing techniques that are used in natural language processing are given below

- Lower case
- Tokenization
- Removing special characters
- Removing stop words and punctuations
- Stemming

#### **A. Lower case**

Since the machine treats lower case and upper case differently, it is easy for a computer to read the words if the text is in the same case. Words like Ball and ball, for example, are processed differently by machines. To avoid such issues, we must make the text in the same case, with lower case being the most preferable instance.

#### **B. Tokenization**

Tokenization is the process of breaking down large pieces of text into smaller ones. Tokenization divides the raw text into words and sentences, which are referred to as tokens. These tokens aid in the comprehension of the context or the development of the NLP model. By evaluating the sequence of words, tokenization aids in interpreting the meaning of the text.

For example, the text “It is raining” can be tokenized into ‘It’, ‘is’, ‘raining’

Tokenization can be done using a variety of methods and libraries. Some of the libraries that can be utilised to do the work are NLTK, Gensim, and Keras. Tokenization can be used to single words or entire sentences. Word tokenization is when a text is broken into words using a separation technique, while sentence tokenization is when the same separation is done for sentences. Stop words are words in a sentence that add no meaning to the sentence and whose removal will have no effect on the text's processing for the specified goal. To decrease noise and the size of the feature set, they are deleted from the lexicon. There are a variety of tokenization strategies that can be used depending on the language and modelling aim. A few tokenization approaches used in NLP are listed below.

### **C. Removing special characters**

Non-alphanumeric characters are known as special characters. These characters can be found in a variety of places, including comments, references, and currency numbers. These characters provide little benefit to text comprehension and cause algorithmic noise. Regular expressions (regex) can thankfully be used to remove these characters and integers.

### **D. Removing stop words and punctuation**

#### **What are stop words?**

Stop words are terms that are typically filtered out of natural language processing. These are the most common words in any language (articles, prepositions, pronouns, conjunctions, and so on) and provide little information to the text. "The," "a," "an," "so," and "what" are some examples of stop words in English.

#### **Importance of removing stop words**

In any human language, there are plenty of stop words. By deleting these words, we may focus more on the key information while removing the low-level information from our text. In other words, removing such phrases has no negative impact on the model that the user trains for the specific task. Because there are fewer tokens involved in the training, removing stop words minimises the dataset size and hence reduces training time. It is not necessary to remove the stop words all of the time. The elimination of stop words is greatly dependant on the task at hand and the aim we are attempting to achieve. For instance, if we're developing a model to perform sentiment analysis,

**Example:** Movie review: "The movie was not good at all." Text after removal of stop words:  
"movie good"

### **E. Stemming**

The process of reducing a word to its word stem, which affixes to suffixes and prefixes or to the roots of words known as a lemma, is known as stemming. Natural language understanding (NLU)

and natural language processing (NLP) both benefit from stemming (NLP). Stemming is a language study of morphology and information retrieval and extraction using artificial intelligence (AI).

### **2.2.2 Exploratory Data Analysis**

John Tukey championed exploratory data analysis to encourage statisticians to investigate data and maybe generate hypotheses that would lead to future data gathering and trials. EDA is more specifically focused on the assumptions that must be checked for model fitting and hypothesis testing. It also performs checks while dealing with missing values and transforming variables as needed.

### **2.2.3 Types of Exploratory Data Analysis**

Different types of Exploratory Data Analysis are listed below:

- Univariate non -graphical
- Multivariate non-graphical
- Univariate graphical
- Multivariate graphical

#### **A. Univariate Non-graphical**

This is the simplest type of data analysis because we only utilize one variable to gather information. The standard purpose of univariate non-graphical EDA is to understand the sample distribution/data and make population observations. The analysis also includes the detection of outliers. The goal of univariate non-graphical EDA approaches is to understand the underlying sample distribution and make population observations. Outlier detection is also a part of this process. We're interested in the range and frequency of univariate categorical data.

The characteristics of population distribution are given below:

- Central tendency
- Spread
- Skewness and kurtosis

## **Central tendency**

The usual or middle values are related to the central tendency or distribution location. The statistics known as mean, median, and sometimes mode are widely used to measure central tendency, with mean being the most prevalent. The median may be selected when the distribution is skewed or when outliers are a concern.

## **Spread**

The Spread is a measurement of how far away from the center we should look for information values. Two relevant measurements of spread are the quality deviation and variance. The variance is the root of the variance since it is the mean of the squares of the individual deviations.

## **Skewness and kurtosis**

The skewness and kurtosis of the distribution are two more useful univariate characteristics. When compared to a normal distribution, skewness is a measure of asymmetry, while kurtosis may be a more nuanced measure of peakness.

## **B. Multivariate Non-graphical**

The use of a multivariate non-graphical EDA technique to show the relationship between two or more variables in the form of cross-tabulation or statistics is common. Cross-tabulation, a tabulation extension for categorical data, is particularly useful. Making a two-way table with column headings that match the amount of one variable and row headings that match the amount of the opposite two variables, then filling the counts with all subjects that share an analogous pair of levels, is chosen for two variables. We construct statistics for quantitative variables separately for each level of the specific variable for each categorical variable and then compare the statistics across the quantity of categorical variables. ANOVA is an off-the-cuff form of comparing the means, and comparing the means is an off-the-cuff version of ANOVA.

## **C. Univariate graphical**

Non-graphical approaches are quantitative and objective, but they do not provide a whole picture of the data; thus, graphical methods require a certain amount of subjective analysis. The following are examples of common univariate graphs.

Common types of univariate graphics are listed below:

- Histogram
- Stem-and-leaf plots
- Boxplots
- Quantile-normal plots

### **Histogram**

A histogram, which is a barplot in which each bar reflects the frequency (count) or proportion (count/total count) of occurrences for various values, is the most basic graph. Histograms are one of the most basic ways to quickly learn about a user's data, such as central tendency, spread, modality, shape, and outliers.

### **Stem-and-leaf plots**

Stem-and-leaf plots are a simple replacement for a histogram. It displays all data values and, as a result, the distribution's form.

### **Boxplots**

The boxplot is another helpful univariate graphical method. Boxplots are great for presenting information regarding central tendency and displaying reliable measurements of location and spread, as well as symmetry and outliers, yet they can be misleading when it comes to multimodality. One of the most basic applications of boxplots is in the form of side-by-side boxplots.

### **Quantile-normal plots**

The most complicated is the ultimate univariate graphical EDA approach. It's known as the quantile-normal plot (QN plot) or the quantile-quantile plot (QQ plot). It is customary to examine how closely a given sample follows a given theoretical distribution. It enables for the discovery of abnormalities as well as the diagnosis of skewness and kurtosis.

## **D. Multivariate graphical**

Graphics are used in multivariate graphical data to show links between two or more sets of knowledge. A grouped barplot, with each group representing one level of one of the variables and each bar within a gaggle reflecting the amount of the opposing variable, is the most popular. Other types of multivariate graphics that are commonly used are:

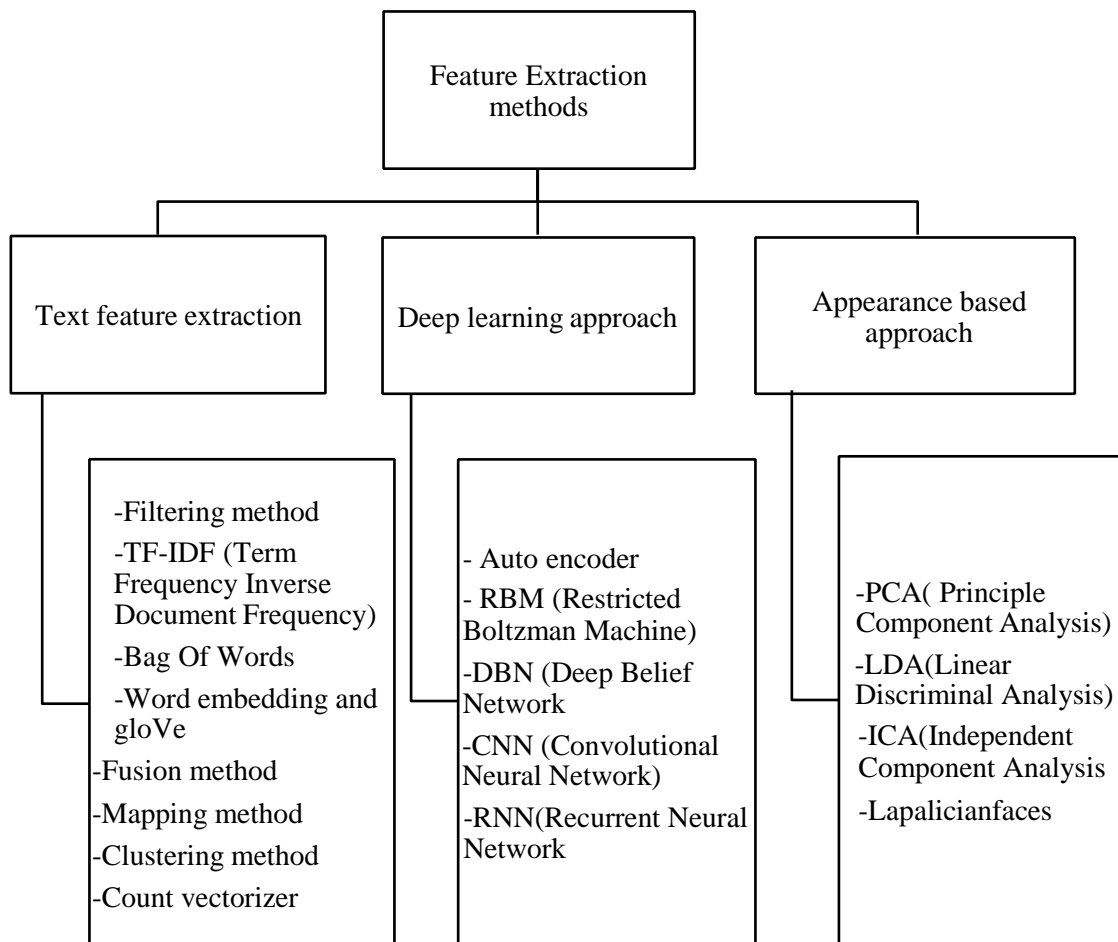
- Scatterplot
- Run chart
- Heatmap
- Multivariate chart
- Bubble chart
- In a nutshell

## **2.3 Feature extraction**

Feature extraction is a broad phrase that refers to strategies for creating combinations of variables to get around these issues while still accurately representing the data. Many practitioners of machine learning feel that well optimized feature extraction is the key to building good models. When you have a large data set and need to reduce the number of resources without sacrificing any critical or relevant information, the feature extraction technique comes in handy. Feature extraction aids in the reduction of unnecessary data in a data set.

### **2.3.1 Types of Feature Extraction**

To provide output for the test data, Machine Learning algorithms learn from a pre-defined collection of features from the training data. However, the primary issue with language processing is that machine learning algorithms cannot work directly on raw text. To transform text into a matrix (or vector) of features, we'll need some feature extraction techniques. When you have a large data set and need to reduce the number of resources without sacrificing any critical or relevant information, the feature extraction technique comes in handy. Feature extraction aids in the reduction of unnecessary data in a data set. Finally, the reduction of data allows the model to be built with less machine effort and at a faster rate.



**Figure 2.3.1 Types of Feature extraction**

Some of the most popular methods of feature extraction are

- Bag-of-Words
- TF-IDF vectorizer

### **Bag of Words**

The Bag-of-Words approach is one of the most basic methods for converting tokens into a set of features. Each word is used as a feature for training the classifier in the BoW model, which is employed in document classification. The text content is converted to numerical feature vectors using this vectorization technique. Bag of Words takes a document from a corpus and converts it into a numeric vector for the machine learning model by mapping each document word to a feature vector. In a task of review-based sentiment analysis, for example, the presence of terms like "fantastic" and "great" suggests a favorable review, whereas phrases like "annoying" and "bad" indicate a negative review.

There are 2 steps while creating a BoW model

- i) The first step is text pre-processing which involves: converting the entire text into lower case characters. Removing all punctuations and unnecessary symbols.
- ii) The second step is to create a vocabulary of all unique words from the corpus.

### 2.3.3 TF-IDF Vectorizer

The term term frequency-inverse document frequency (TF-IDF) stands for term frequency-inverse document frequency. It draws attention to a specific issue that, while not common in our corpus, is extremely important. The TF-IDF value rises in proportion to the number of times a word appears in the document and falls in proportion to the number of documents in the corpus containing the word. It is composed of 2 sub-parts, which are

- Term Frequency (TF)
- Inverse Document Frequency (IDF)

#### Term Frequency (TF)

The term frequency specifies how often a term appears throughout the document. It might be compared to the likelihood of discovering a word within a document. It determines how many times a word  $w_i$  appears in a review  $r_j$  in relation to the total number of words in the review  $r_j$ . It's written like this:

$$tf(w_i, r_j) = \frac{\text{No. of times } w_i \text{ occurs in } r_j}{\text{Total no. of words in } r_j}$$

A different scheme for calculating tf is *log normalization*. And it is formulated as:

$$tf(t, d) = 1 + \log f_{t, d}$$

where,  $f_{t, D}$  is the frequency of the term  $t$  in document  $D$ .

#### Inverse Document Frequency (IDF)

The inverse document frequency is a metric that determines whether a term is rare or common across all documents in a corpus. It highlights terms that appear in a small number of texts across the corpus, or in plain English, words with a high IDF score. The logarithm of the overall term is derived by dividing the total number of documents  $D$  in the corpus by the number of documents containing the term  $t$ .

$$idf(d, D) = \log \frac{|D|}{\dots}$$



$d \in D$ :  $t \in D$  where,

- $f_{\{t, D\}}$  stands for frequency of the term  $t$  in document  $D$ .
- $|D|$  is the total number of documents in the corpus.
- $\{d \in D : t \in d\}$  is the count of documents in the corpus, which contains the term  $t$ .

The value of IDF (and consequently TF-IDF) is greater than or equal to 0 since the ratio inside the IDF's log function must always be greater than or equal to 1. The ratio inside the logarithm approaches 1 when a phrase appears in a high number of documents, and the IDF approaches 0.

Term Frequency-Inverse Document Frequency (TF-IDF)

TF-IDF is the product of TF and IDF. It is formulated as:

$$tf\ idf(t, d, D) = tf(t, d) * idf(d, D)$$

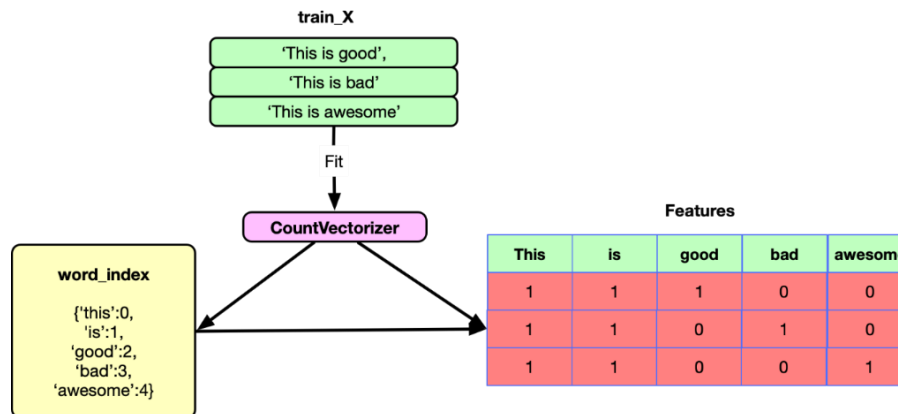
A term with a high frequency in a document and a low document frequency in the corpus gets a high TF-IDF score. The IDF value approaches 0 for a word that appears in practically all documents, bringing the tf-idf value closer to 0. When both IDF and TF values are high, the TF-IDF value is high, indicating that the term is uncommon throughout the document yet common within it

#### **Important points about TF-IDF vectorizer**

- a. The TF-IDF method generates a document term matrix, with each column representing a single unique word, similar to the count vectorization method.
- b. The TF-IDF method differs in that each cell does not carry a term frequency number, but rather a weight value that indicates how essential a word is for a certain text message or document.
- c. The TF-IDF gives less often occurring events more weight and predicted events less weight. As a result, it penalizes frequently occurring terms that appear frequently in a document, such as "the" and "is," while giving less frequent or rare words more weight.
- d. A word's TF x IDF product indicates how common the token appears in the document and how unique it is over the whole corpus of documents.

#### **2.3.4 Count vectorizer**

It is one of the most straightforward methods of text vectorization. It generates a document term matrix, which is a collection of dummy variables that indicate whether or not a certain word exists in the document. Count vectorizer will try to generate a document term matrix in which the individual cells reflect the frequency of that word in a given document, also known as term frequency, and the columns are allocated to each word in the corpus, by fitting and learning the word vocabulary. Figure 2.3.4 describes the process of count vectorizer with a simple example.



**Figure 2.3.4 Process of Count vectorizer**

## 2.4 Model building

### What is machine learning?

“Machine Learning is defined as the study of computer programs that leverage algorithms and statistical models to learn through inference and patterns without being explicitly programmed. Machine Learning field has undergone significant developments in the last decade.”

AI was able to progress beyond only doing the tasks it was designed to do thanks to machine learning algorithms. AI systems were only employed to automate low-level operations in corporate and enterprise settings until ML became ubiquitous. Intelligent automation and basic rule-based classification were among the tasks included.

### 2.4.1 Supervised learning

One of the most basic types of machine learning is supervised learning. The machine learning algorithm is trained on labelled data in this case. Despite the fact that precise labelling of data is required for this method to work, supervised learning can be incredibly effective when utilized under the right situations. The ML algorithm is given a short training dataset to work with in supervised learning.

Supervised machine learning is divided into two types

- Classification
- Regression.

## **Classification**

The Classification algorithm is a Supervised Learning technique that uses training data to determine the category of new observations. Classification is the process of a software learning from a dataset or observations and then classifying fresh observations into one of several classes or groupings. Yes or No, 0 or 1, Spam or Not Spam, cat or dog, and so forth. Targets/labels or categories are all terms that can be used to describe classes. Unlike regression, Classification produces a category rather than a value, such as "Green or Blue," "fruit or animal," and so on. As a result, it accepts labelled input data, which means it has input and output.

The following are the list of supervised machine learning algorithms that are used in model building phase in the project:

- a) Decision Tree classifier
- b) Naïve Bayes classifier
- c) Support Vector Class
- d) Logistic regression
- e) K-Nearest Neighbour algorithm
- f) Random Forest
- g) Ada Boost
- h) Bagging
- i) Extra Tree Classifier
- j) Gradient boosting algorithm

## **Decision tree**

Decision Trees are a type of Supervised Machine Learning in which the data is continually separated according to a parameter (that is, one explains what the input is and what the corresponding output is in the training data). Two entities, decision nodes and leaves, can be used to explain the tree. The decisions or final outcomes are represented by the leaves. And the data is separated at the decision nodes.

## **Naïve Bayes algorithm**

The Nave Bayes method is a supervised learning technique for addressing classification issues that is based on the Bayes theorem. It is mostly utilized in text classification tasks that require a large training dataset. The Nave Bayes Classifier is a simple and effective classification method that aids in the development of fast machine learning models capable of making quick predictions. It is a probabilistic classifier, which means it makes predictions based on an object's probability. Spam

filtration, sentiment analysis, and article classification are all frequent uses of the Nave Bayes Algorithm.

### **Support Vector Machine algorithm**

The Support Vector Machine, or SVM, is a popular Supervised Learning technique that may be used to solve both classification and regression issues. However, it is mostly utilized in Machine Learning for Classification difficulties. The SVM algorithm's purpose is to find the optimum line or decision boundary for categorizing n-dimensional space into classes so that additional data points can be readily placed in the correct category in the future.

### **Logistic Regression algorithm**

Under the Supervised Learning approach, one of the most common Machine Learning algorithms is logistic regression. It's a method for predicting a categorical dependent variable from a set of independent variables. A categorical dependent variable's output is predicted using logistic regression. As a result, the result must be a discrete or categorical value. It can be Yes or No, 0 or 1, true or false, and so on, but instead of giving exact values like 0 and 1, it delivers probabilistic values that are somewhere between 0 and 1. Except for how they are employed, Logistic Regression is very similar to Linear Regression. Regression problems are solved using Linear Regression, while classification problems are solved using Logistic Regression. Instead of fitting a regression line, we fit a "S" shaped logistic function in logistic regression, which predicts two maximum values (0 or 1).

### **K-Nearest Neighbor**

The K-Nearest Neighbor algorithm is based on the Supervised Learning technique and is one of the most basic Machine Learning algorithms. The KNN algorithm assumes that the new case/data and existing cases are similar and places the new case in the category that is most similar to the existing categories. The KNN method stores all available data and classifies a new data point based on its similarity to the existing data. This means that utilizing the KNN approach, fresh data can be swiftly sorted into a well-defined category.

### **Random Forest algorithm**

Random Forest is a well-known machine learning algorithm that uses the supervised learning method. In machine learning, it can be utilized for both classification and regression issues. It is based on ensemble learning, which is a method of integrating numerous classifiers to solve a complex problem and increase the model's performance. "Random Forest is a classifier that contains a number of decision trees on various subsets of a given dataset and takes the average to enhance the predicted accuracy of that dataset," according to the name. Instead than relying on a single decision tree, the

random forest collects the forecasts from each tree and predicts the final output based on the majority votes of predictions. The more people, the better.

### **Ada boost algorithm**

Boosting is an ensemble modelling strategy that aims to create a strong classifier out of a large number of weak ones. It is accomplished by constructing a model from a sequence of weak models. To begin, a model is created using the training data. The second model is then created, which attempts to correct the faults in the previous model. This procedure is repeated until either the entire training data set is correctly predicted or the maximum number of models has been added.

### **Bagging algorithm**

Bagging classifiers are ensemble meta-estimators that fit base classifiers to random subsets of the original dataset and then aggregate their individual predictions (either by voting or average) to generate a final prediction. By integrating randomness into the construction technique of a blackbox estimator (e.g., a decision tree), such a meta-estimator may often be used to lower the variance of a black-box estimator (e.g., a decision tree).

### **Extra Tree Classifier**

Extremely Randomized Trees Classifier (Extra Trees Classifier) is an ensemble learning technique that combines the outcomes of several de-correlated decision trees collected in a "forest" to provide a classification result. It is conceptually identical to a Random Forest Classifier, with the exception of how the decision trees in the forest are constructed. The Extra Trees Forest's Decision Trees are all made from the original training sample. Then, at each test node, each tree is given a random sample of  $k$  features from the feature set, from which it must choose the best feature to split the data according to certain mathematical criteria (typically the Gini Index).

### **Gradient Boost algorithm**

A popular boosting algorithm is gradient boosting. Each predictor in gradient boosting corrects the error of its predecessor. Unlike Adaboost, the training instance weights are not adjusted; instead, each predictor is trained using the predecessor's residual errors as labels. CART is the base learner in a technique called Gradient Boosted Trees (Classification and Regression Trees).

### **XGBoost algorithm**

Gradient Boosted decision trees are implemented in XGBoost. Many Kaggle competitions are dominated by XGBoost models. Decision trees are constructed sequentially in this approach. In

XGBoost, weights are very significant. All of the independent variables are given weights, which are subsequently fed into the decision tree, which predicts outcomes. The weight of factors that the tree predicted incorrectly is increased, and these variables are fed into the second decision tree.

### **3. RESULTS AND DISCUSSION**

In this study, the dataset is trained and implemented with eleven different supervised machine learning method to find the best model to classify a mail if it is spam or ham.

#### **5.1 Data acquisition and data collection**

Dataset is collected from online website like Kaggle to implement all the phases from methodology and to train the model.

	v1	v2
1	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat...
2	ham	Ok lar... Joking wif u oni...
3	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std txt rate)T&C's apply 08452810075over18's
4	ham	U dun say so early hor... U c already then say...
5	ham	Nah I don't think he goes to usf, he lives around here though
6	spam	FreeMsg Hey there darling it's been 3 week's now and no word back! I'd like some fun you up for it still? Tb ok! XxX std chgs to send, å£1.50 to rcv
7	ham	Even my brother is not like to speak with me. They treat me like aids patent.
8	ham	As per your request 'Melle Melle (Oru Minnaminunginte Nuringu Vettam)' has been set as your callertune for all Callers. Press *9 to copy your friends Callertune
9	spam	WINNER!! As a valued network customer you have been selected to receive a å£900 prize reward! To claim call 09061701461. Claim code KL341. Valid 12 hours only.
10	spam	Had your mobile 11 months or more? U R entitled to Update to the latest colour mobiles with camera for Free! Call The Mobile Update Co FREE on 08002986030
11	ham	I'm gonna be home soon and i don't want to talk about this stuff anymore tonight, k? I've cried enough today.
12	spam	SIX chances to win CASH! From 100 to 20,000 pounds txt> CSH11 and send to 87575. Cost 150p/day, 6days, 16+ TsandCs apply Reply HL 4 info
13	spam	URGENT! You have won a 1 week FREE membership in our å£100,000 Prize Jackpot! Txt the word: CLAIM to No: 81010 T&C www.dbuk.net LCCLTD POBOX 4403LDNW1A7RW18
14	ham	I've been searching for the right words to thank you for this breather. I promise i wont take your help for granted and will fulfil my promise. You have been wonderful and a blessing at all times.
15	ham	I HAVE A DATE ON SUNDAY WITH WILL!!
16	spam	XXXMobileMovieClub: To use your credit, click the WAP link in the next txt message or click here>>> http://wap.xxxmobilemovieclub.com?n=QJGIGHJGCBL
17	ham	Oh k...i'm watching here:)
18	ham	Eh u remember how 2 spell his name... Yes i did. He v naughty make until i v wet.
19	ham	Fine if thatå the way u feel. Thatå the way its gota b
20	ham	

**Figure 5.1 Collection of Dataset**

The above figure 5.1 shows the collection of dataset

## 5.2 Data Pre-processing

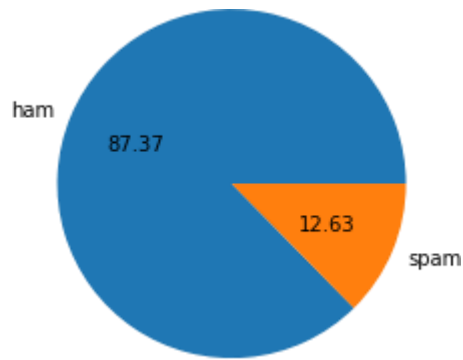
Data Pre-processing is the second phase from the methodology it begins by data cleaning method and datas are pre-processed by using Natural Language Process and by performing EDA

```
Out[13]:
```

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

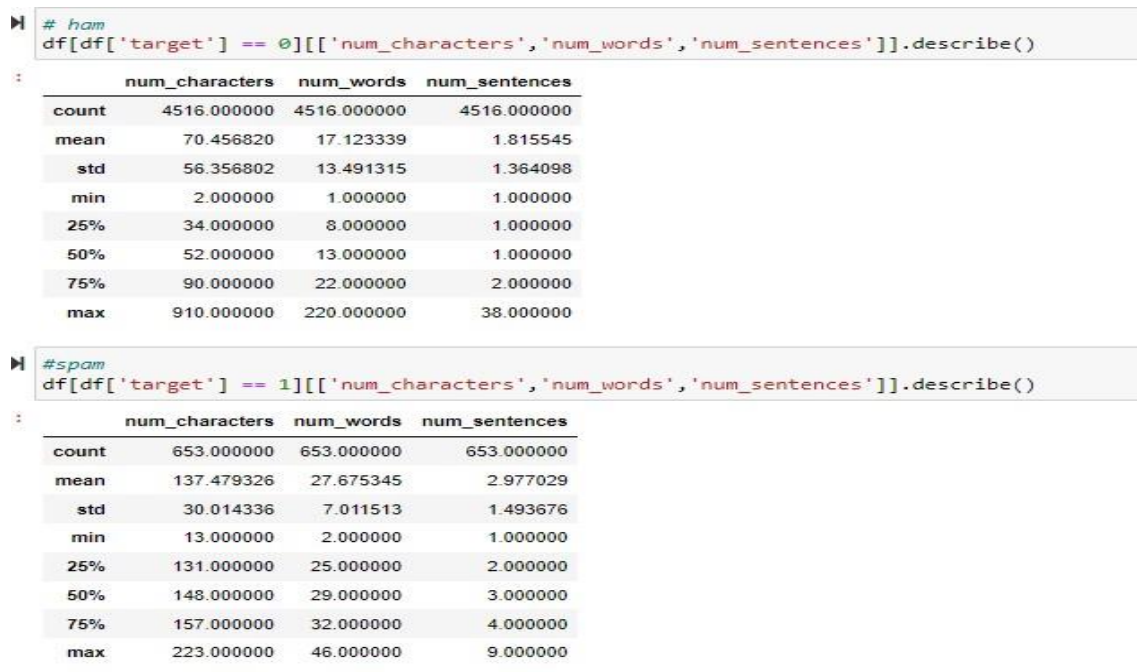
**Figure 5.2 After Data cleaning**

The above figure 5.2 shows the outcome after data cleaning one of the data pre-processing technique which went through removing null values and duplicate values and label encoding for transforming the data



**Figure 5.2.1 percentage of ham and spam email**

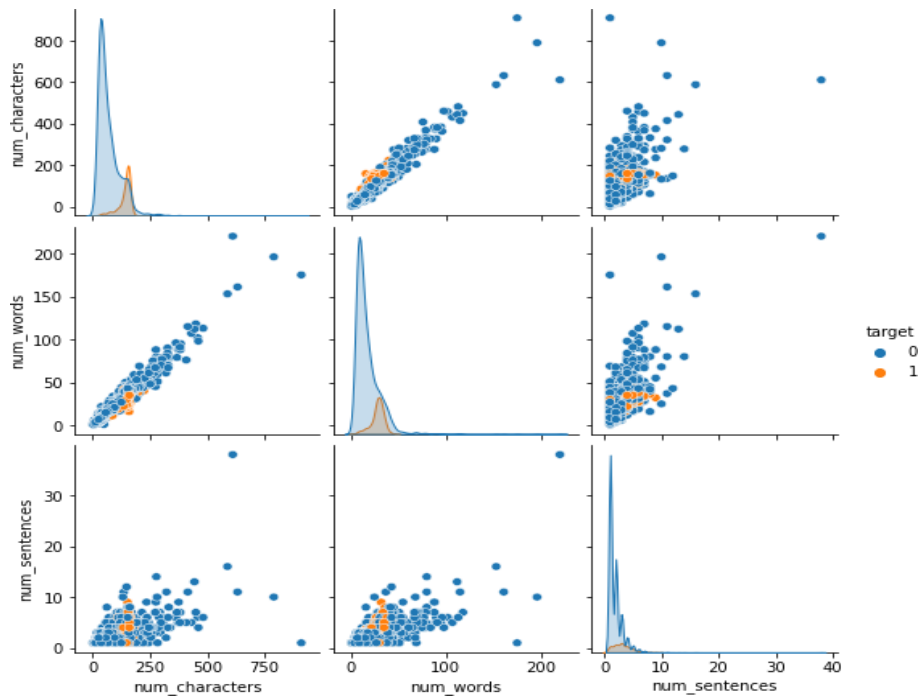
The above figure 5.2.1 shows EDA analyses the percentage level of ham mails and spam mail from dataset



**Figure 5.2.2 after tokenization**

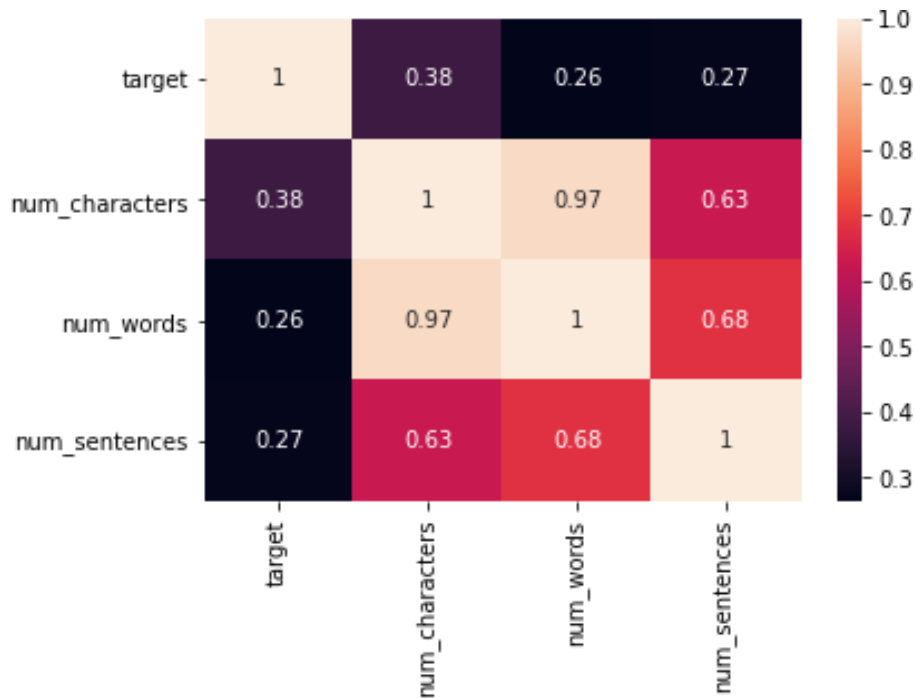
The above figure 5.2.2 number of characters, number of words, number of sentences calculated for both ham and spam emails after tokenization.





**Figure 5.2.3 representation of ham and spam**

The above figure 5.2.3 all the values that are calculated are plotted in scatter plot by color coded for ham and spam emails



**Figure 5.2.4 correlation analysis**

The above figure 5.2.4 plotted in heat map for all the transformed text.



Figure 5.2.5 word cloud for mostly appeared words

Figure 5.2.5 shows the most commonly appeared words of ham in dataset

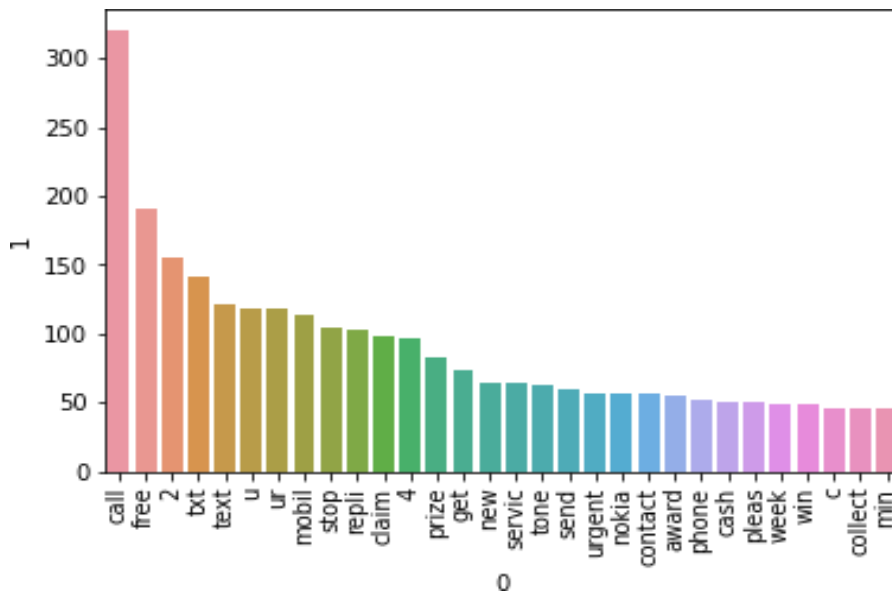


Figure 5.2.6 words are rated accordingly

Above figure 5.2.6 shows the word that has appeared in maximum time from the outcome received it is observed that word 'call' has appeared mostly from spam data and so in continues with words like 'free' etc.

### 5.3 Feature Extraction

Feature extraction is the third phase from methodology designed for this project and it includes two vectorization method, TF-IDF vectorizer and count vectorizer

```
from sklearn.feature_extraction.text import CountVectorizer,TfidfVectorizer
cv = CountVectorizer()
tfidf = TfidfVectorizer(max_features=3000)

X = tfidf.fit_transform(df['transformed_text']).toarray()

#from sklearn.preprocessing import MinMaxScaler
#scaler = MinMaxScaler()
#X = scaler.fit_transform(X)

# appending the num_character col to X
#X = np.hstack((X,df['num_characters'].values.reshape(-1,1)))

X.shape
(5169, 3000)
```

**Figure 5.3 Feature extraction**

The above figure 5.3 shows the method used for feature extraction is implemented

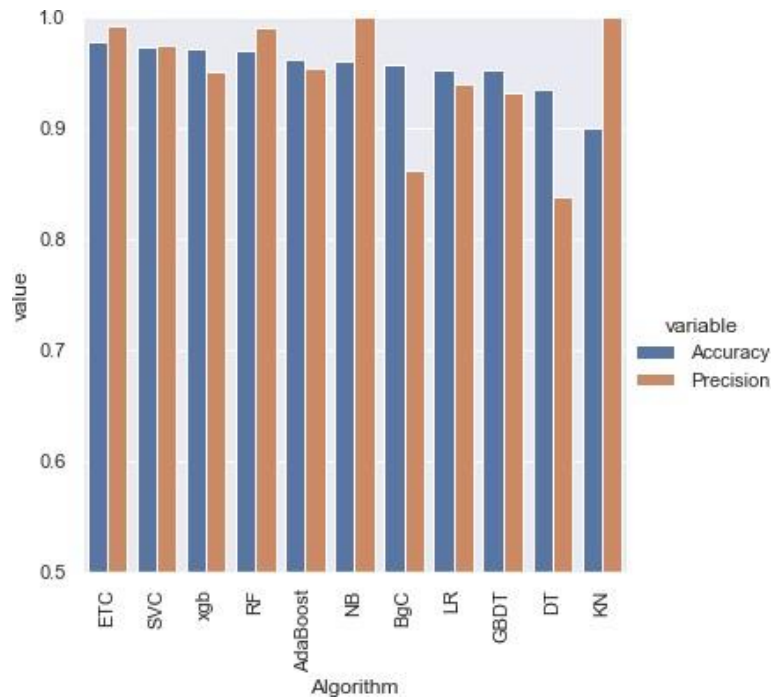
### 5.4 Model building

The dataset is trained on eleven machine learning models in this project to discover the best spam mail classification approach. Accuracy and precision was calculated based on the performance of each machine learning model. The results of the best-performing algorithm are considered for the final spam classification based on those rank matrix.

	Algorithm	Accuracy	Precision
1	KN	0.900387	1.000000
2	NB	0.959381	1.000000
8	ETC	0.977756	0.991453
5	RF	0.970019	0.990826
0	SVC	0.972921	0.974138
6	AdaBoost	0.962282	0.954128
10	xgb	0.971954	0.950413
4	LR	0.951644	0.940000
9	GBDT	0.951644	0.931373
7	BgC	0.957447	0.861538
3	DT	0.935203	0.838095

**Figure 5.4 Model Building**

The above given figure 5.4 shows all the eleven algorithms with accuracy and precision score by which it is analyzed that Extra Tree Classifier is considered to be the best working model with 98% of accuracy and 99% of precision.



**Figure 5.4.1 Comparative analysis of model building**

The above figure 5.4.1 shows the comparative analysis for accuracy and precision value that is calculated for all eleven algorithms.

#### **4. CONCLUSION AND FUTURE SCOPE**

The system in this project focuses on detecting the spam mail by reviewing it in two stages: feature extraction and classification. In the first stage, the basic concepts and principles of spammail are highlighted in social media. The data selected for the training dataset all the feature extraction and classification is performed by using two methods, TF-IDF vectorizer and countvectorizer. Feature extraction is performed for the text with email content. During the discoverystage, the current methods are reviewed for detection of spam mail using different supervised learning algorithms including high range methods like naïve Bayes and decision tree classification methods. The outcome of all the algorithms in this project gives the best score form Extra Tree Classifier algorithm with 98% of the accuracy and 99% level of precision from the dataset.

Spam mail detection is always improving, but hackers are still able to break security, thus it is falling behind. Networking is the source of many computer security threats, but it also amplifies others. Secure computing is dependent on secure networks, and vice versa. It's no coincidencethat as network security grows more fragile, people are becoming increasingly concerned aboutsecurity and privacy. The use of these technologies in real time with numerous future scenariosto analyze the depth and classify spam based on its level of susceptibility. The current proposedsolution is for English language mails, however we can expand the scope to include more languages in the future.

## 5. REFERENCES

1. Nikhil Kumar, Sanket Sonowal and Nishant,” Email Spam Detection Using Machine Learning Algorithms” in IEEE Xplore Part Number: CFP20N67-ART; ISBN: 978-1-7281- 5374-2.
2. M. Bassiouni, M. Ali & E. A. El-Dahshan (2018) Ham and Spam E-Mails Classification Using Machine Learning Techniques, Journal of Applied Security Research, 13:3, 315-331, DOI: 10.1080/19361610.2018.1463136
3. Nandhini.S and Dr.Jeen Marseline.K.S ,“ Performance Evaluation of Machine Learning Algorithms for Email Spam Detection” in 2020 IEEE 10.1109/ic-ETITE47903.2020.312
4. Ms.D.Karthika Renuka<sup>1</sup> ,Dr.T.Hamsapriya<sup>2</sup> , Mr.M.Raja Chakkaravarthi<sup>3</sup> ,Ms.P. Lakshmi Surya<sup>4</sup>,” Spam Classification based on Supervised Learning using Machine Learning Techniques” in 78-1-61284-9764-1/11/\$26.00 ©2011 IEEE
5. Kriti Agarwal and Tarun Kumar,” Email Spam Detection using integrated approach of NaïveBayes and Particle Swarm Optimization” in IEEE Xplore Compliant Part Number: CFP18K74- ART; ISBN:978-1-5386-2842-3
6. Simran gibson <sup>1</sup> , biju issac <sup>1</sup> , (senior member, ieee), li zhang <sup>1</sup> , (senior member, ieee), and seibu mary jacob<sup>2</sup> , (member, ieee)” detecting spam email with machine learning optimized with bio-inspired metaheuristic algorithms”
7. Nikhil Govil<sup>1</sup> and Kunal Agarwal<sup>2</sup> , Ashi Bansal<sup>3</sup> , Astha Varshney<sup>4</sup> “A Machine Learning based Spam Detection Mechanism” in IEEE Xplore Part Number:CFP20K25-ART;ISBN:978-1-7281-4889-2
8. Priti Sharma<sup>1</sup> and Uma Bhardwaj<sup>1</sup> “ Machine Learning based Spam E-Mail Detection” International Journal of Intelligent Engineering and Systems, Vol.11, No.3, 2018 DOI: 10.22266/ijies2018.0630.01
9. Ruhul Amin, Md. Moshir Rahman, and Nahid Hossain “A Bangla Spam Email Detection and Datasets Creation Approach based on Machine Learning Algorithms”, 978-1-7281-6410-6/19/\$31.00 ©2019 IEEE
10. Said Sallouma\*, Tarek Gabera,b, Sunil Vaderaa , and Khaled Shaalanc “Phishing Email Detection Using Natural Language Processing Techniques: A Literature Survey” ,Said Salloum et al. / Procedia Computer Science 189 (2021) 19–28