|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HUMBER INSTITUTE OF TECHNOLOGY AND ADVANCED LEARNING  (HUMBER COLLEGE)  **SMS SPAM CLASSIFICATION USING MACHINE LEARNING ALGORITHMS**  GROUP NUMBER 14  Submitted by:   |  |  | | --- | --- | | First Name | Student Number | | Vrushali Uchade | N01531537 | | Joel Lalen | N01550628 | | Joyanta Mridha | N01548103 | | Darshan Soni | N01511682 | | Evan Paul | N01514719 |   Professor - Salam Ismaeel  Submission Date: 7 December 2023 |

**INTRODUCTION**

SMS represents one of the most potent means of communication. Numerous organizations utilize SMS to interact with their clients/customers, while banks and government bodies also leverage it for communication purposes. Additionally, many businesses employ SMS for advertising. This underscores the crucial role of SMS, as it operates independently of an active internet connection. However, the widespread use of SMS has made it a prime target for hackers and spammers. Exploiting its vulnerabilities, hackers can easily compromise a person's cell phone by sending malicious links. Clicking on such links or messages can automatically compromise the mobile device, allowing the hacker/spammer to exploit the system at will. Therefore, there's a pressing need to control the content that end-users receive. A system capable of distinguishing between SPAM and non-SPAM messages (referred to as HAM) is essential. To address these concerns, the authors have developed a system employing Machine Learning techniques to identify whether a message is SPAM or HAM based on its content.

In this report, the authors have applied different machine learning algorithms to SMS spam classification problem and compared their performance to gain insight. Machine learning refers to an application's ability to enhance its predictive outcomes through iterative improvements or learning from experience. This enhancement through experience is termed "training." Achieving improved results can necessitate numerous iterations to steadily enhance performance. Throughout the training process, data is fed into a machine-learning algorithm, allowing it to refine its internal representation and numerical parameters as it encounters discrepancies or errors in training.

**OBJECTIVE & INDUSTRY IMPORTANCE**

The objective of this research paper is to help one classify emails using artificial intelligence as spam emails or non-spam emails. The aim of this research work is to explore different methods of text classification into spam or ham and comparing results based on accuracy and precision (false positive rate). The spam detection is crucial across industries as it not only protects against security threats but also preserves brand reputation, optimizes resources, ensures compliance, fosters customer trust, and encourages innovation in cybersecurity practices. Therefore, investing in robust spam detection mechanisms is pivotal for the sustained growth and success of businesses in the digital age. By using the process of spam detection, we can protect the brand images and reputation of big companies, Spam emails and messages often contain malicious content, including phishing attempts, malware, or fraudulent links. Effective spam detection helps in safeguarding users' personal information, financial data, and overall cybersecurity. Various industries, especially those handling sensitive information like healthcare and finance, are subject to strict regulations concerning data protection and privacy. Effective spam detection helps in complying with these regulations by ensuring secure communication channels. Spam-free environments foster better user experiences. By filtering out unwanted messages, companies can ensure that their customers receive relevant and genuine communications, leading to improved engagement and trust. Spam can result in financial losses due to various reasons like phishing scams leading to fraud, loss of productivity, or reputational damage impacting sales. Efficient spam detection mitigates these risks, thereby safeguarding the financial health of an organization.

**DATA EXPLORATION**

We have used a database from Kaggle website having 5574 text messages from UCI Machine Learning repository gathered in 2012 [1] [2]. It contains a collection of 425 SMS spam messages manually extracted from the Grumbletext Web site (a UK forum in which cell phone users make public claims about SMS spam), a subset of 3,375 SMS randomly chosen non-spam (ham) messages of the NUS SMS Corpus (NSC), a list of 450 SMS non-spam messages collected from Caroline Tag’s PhD Thesis, and the SMS Spam Corpus v.0.1 Big (1,002 SMS non-spam and 322 spam messages publicly available). The dataset is a large text file, in which each line starts with the label of the message, followed by the text message string. After preprocessing of the data and extraction of features, machine learning techniques such as naive Bayes, SVM, and other methods are applied to the samples, and their performances are compared. The main reason for choosing this dataset is combined together by randomly sampling from different resources hence it gives exposure to different scenarios of ham and spam SMSs. In statistics, it is proven that random sampling is the best.

|  |  |
| --- | --- |
| **Source** | **Total Messages** |
| The Grumbletext Web site | 425 |
| NUS SMS Corpus (NSC) | 3375 |
| Caroline Tag's Ph.D. Thesis | 450 |
| Spam Corpus v.0.1 Big | 1324 |
| **Total** | **5574** |

*Table 1. Detail Of the Dataset*

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| **V1** | Label having values either as spam or ham |
| **V2** | Actual text messages |
| **Unnamed:2** | NaN |
| **Unnamed:3** | NaN |
| **Unnamed:4** | NaN |

*Table 2: SMS dataset variables*

**DATA CLEANING & PREPARATION**

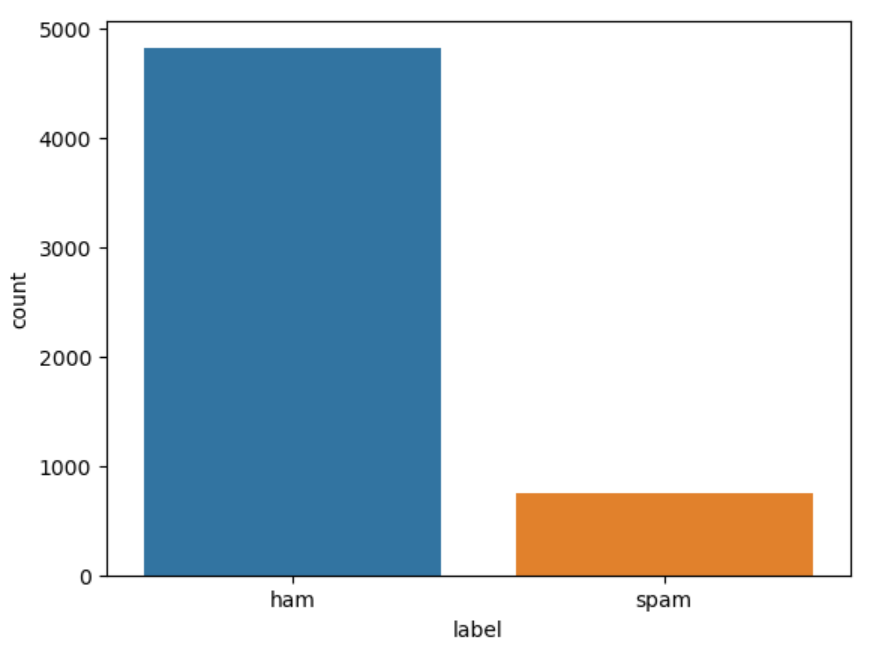
The dataset has 5 columns v1, v2, Unnamed:2, Unnamed:3 and Unnamed:4, we have performed the below data cleaning operations to make the data good for analysis:

1. Columns “Unnamed: 2”, “Unnamed: 3”, and “Unnamed: 4” contain "NaN" (not a number) signifying missing values. They are not needed, so they can be dropped as they are not going to be useful in building the model. We have used the **drop** **function** to remove these columns from the dataset. Remaining two relevant columns are not named properly to get a clear meaning hence we have used **rename** **function** to rename the v1 column to label and v2 to Text.
2. After renaming the column names, added a new column as **label\_enc** which provides binary mapping of two labels (ham and spam). Ham label is termed as 0 and spam label is termed 1.
3. Replacing all special characters which are part of email, address, currency symbols, numbers, punctuations, whitespaces etc from the text messages column with words like ‘email’, ‘money-symbol’, ‘phone-number’ etc.
4. Replacing **stop words** (words that are commonly used in English like is, in, the…) with blank spaces because they do not add any value to performance of machine learning models.
5. Text messages are converted to lower case to avoid mismatching.

**DATA VISUALIZATION**

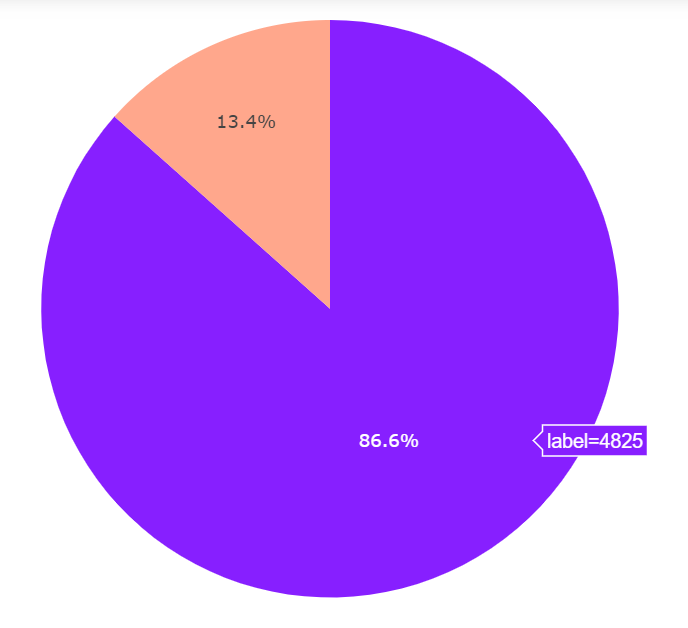
Data visualization is the process where we presenting the available data in the forms of graphs, charts and other visual forms so that linking or correlation between the values of dataset is identified easily. We have used below listed libraries of python to present the visual representation the spam and ham messages:

**Matplotlib and seaborn:** We have used these two libraries to create below bar graph visualizing the total count of ham and spam messages in the data set. As observed from dataset the spam and ham messages of the dataset are unevenly distributed.



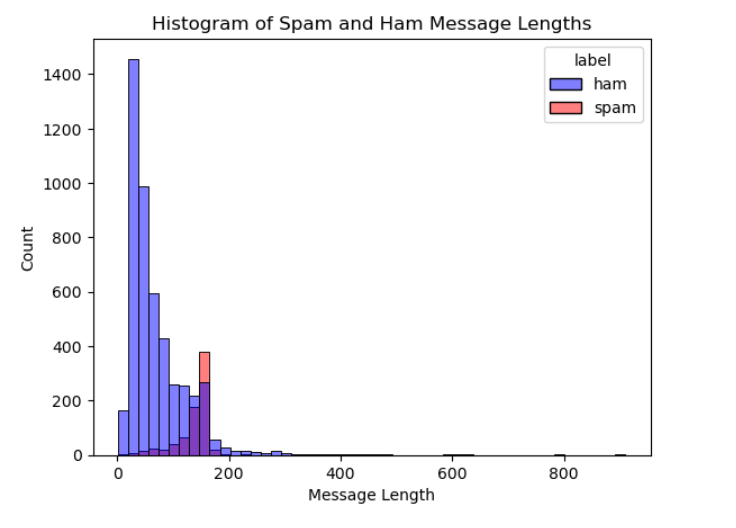
*Fig 1: Bar graph for ham and spam count*

**Plotly express:** We have used this library to show the pie chart distribution of spam and ham messages. We could see that 13.4% of total messages are spam and remaining 86.6% percentage are ham messages form the entire dataset.



*Fig 2: Pie chart showing ham and spam weightage*

We have used plotly express library to show the histogram for length of spam and ham messages vs their count from the dataset. We can observe that the length of spam messages varies between 0-200 words whereas length of ham messages is from 0-800 words having most messages in the range less than 100. The average count of word in spam messages are around 400 when compared to 1400 for ham.



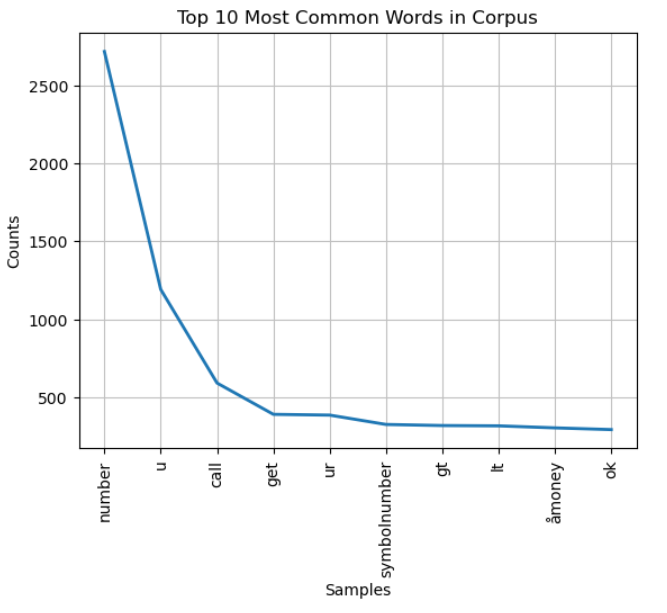
*Fig 3: Histograms for length vs count distribution of messages*

**Wordcloud:** Word Cloud is a data visualization technique used for representing text data in which the size of each word indicates its frequency or importance. Significant textual data points can be highlighted using a word cloud. Below figures shows the word cloud for ham and spam messages respectively. By looking at the word cloud of ham messages we can state that words like ok, got, will, know, come, sorry, etc. are most prominently used by people in regular testing. If we look at the spam messages words like free, call, mobile, text, urgent, claims are used by spammer to send messages.



*Fig 4: Ham and Spam messages word cloud*

The below line chart shows the total count of most used in the entire data set.



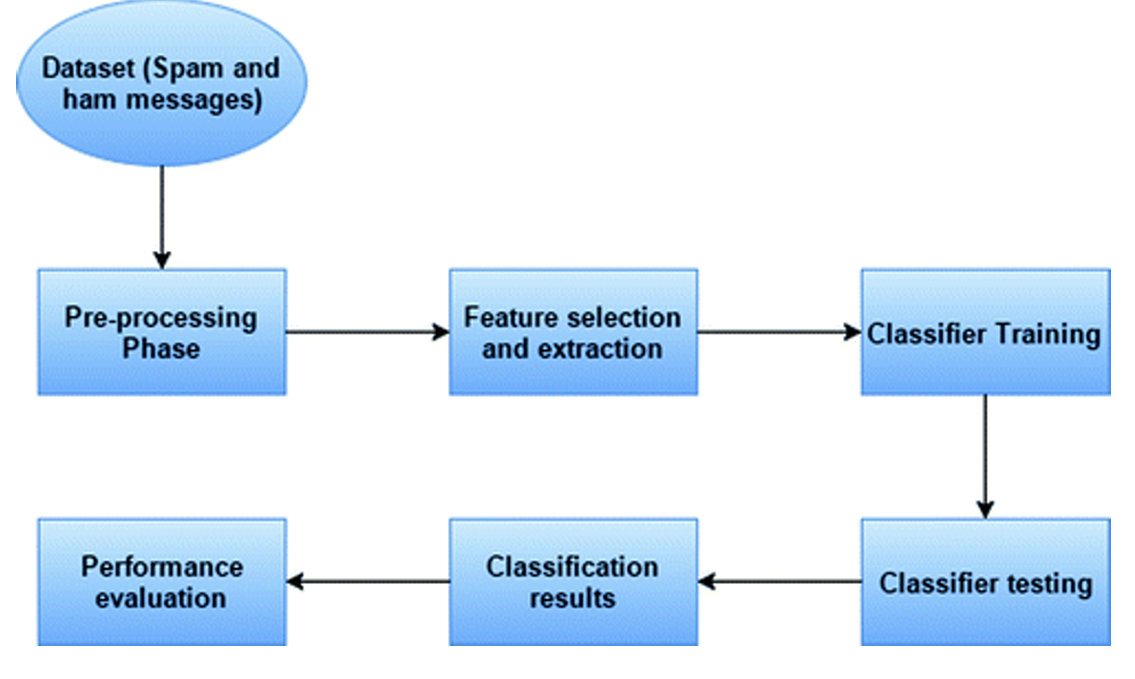
*Fig 5: Top 10 most commonly used words in the messages*

**MODEL BUILDING & ANALYSIS**

We will be using supervised learning algorithms for predicting the classification of SMS messages. The supervised predictive [approach](https://machinelearningmastery.com/gentle-introduction-to-predictive-modeling/) is the problem of developing a model using historical data to make a prediction on new data where we do not have the answer, in our case we will using the test data set evaluate the models created on the training data set using below techniques. Predictive modeling can be described as the mathematical problem of approximating a mapping function (f) from input variables (X) to output variables (y). This is called the problem of function approximation.

We will be using one modelling technique based on regression and three others based on classification for this SMS problem. Regression predictive modeling is the task of approximating a mapping function (f) from input variables (X) to a continuous output variable. Classification predictive modelling is the task of approximating a mapping function (f) from input variables (X) to discrete output variables (y).

We have used four main machine learning models Linear Regression, Naïve Baye’s algorithm, K Nearest Neighbours and Support Vector Machines (SVC). Data is first segregated into test and train shares and then vectorized before being provided to these models.



*Fig 6: Workflow for spam detection*

Initially the dataset is pre-processed using multiple techniques mentioned below then the data is converted in training and testing data which is then fed to the classifier algorithms. Going ahead with the process the test data is used to find the performance parameters of the model based on which the model is judged.

**Pre-Processing Data**

**Test\_Train Data:** This is a model validation process wherein dataset into divided into two part test and train on the basis of test\_size parameter. Training data set is the actual dataset which is fed to the model to discover and learn patterns. Test data is new or unknown dataset which is fed to the model to evaluate its performance and response for unknown data. We have divided our dataset in the ratio of 40-60 for this process.

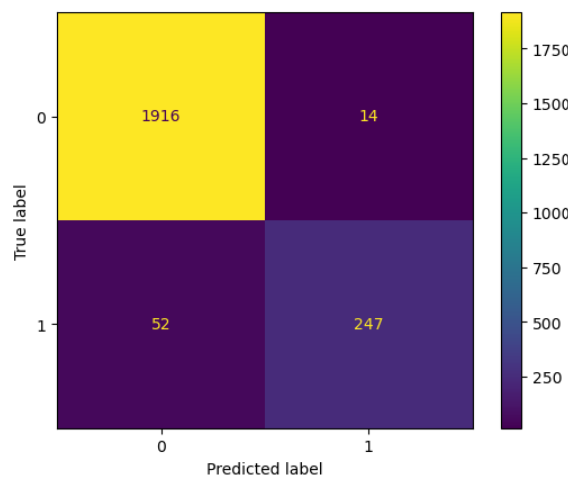
Output: ((3343,), (3343,), (2229,), (2229,))

**Text Vectorization:** We use text vectorization to transform textual data into a numerical format that computers can understand and process the input. After text vectorization is performed, the resulting numerical data can be used for more advanced linguistic applications. Here we have used sklearn feature\_extraction library to convert the given text messages into its vector forms. The ‘TfidfVectorizer’ object is used for feature extraction, setting specific parameters for minimum document frequency, stop word removal (using English stop words), and converting all words to lowercase.

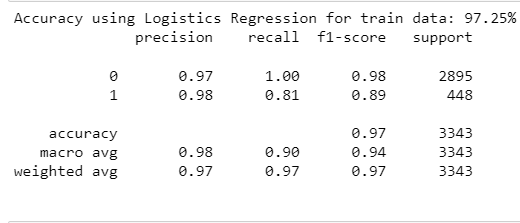
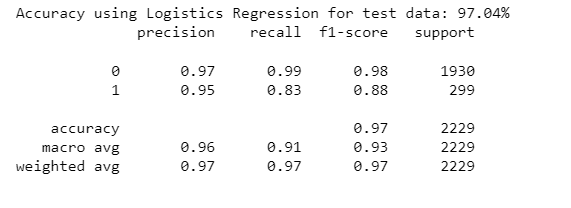
**Machine Learning Models**

**Model 1: Logistic Regression:** Logistic regression is a Machine Learning classification algorithm that is used to predict the probability of certain classes based on some dependent variables. Logistic regression is part of the regression family as it involves predicting outcomes based on quantitative relationships between variables. the [logistic regression algorithm](https://www.interviewbit.com/data-science-interview-questions/) analyses relationships between variables. It assigns probabilities to discrete outcomes using the Sigmoid function, which converts numerical results into an expression of probability between 0 and 1.0. Probability is either 0 or 1, depending on whether the event happens or not.

Results of Logistic Regression Model:



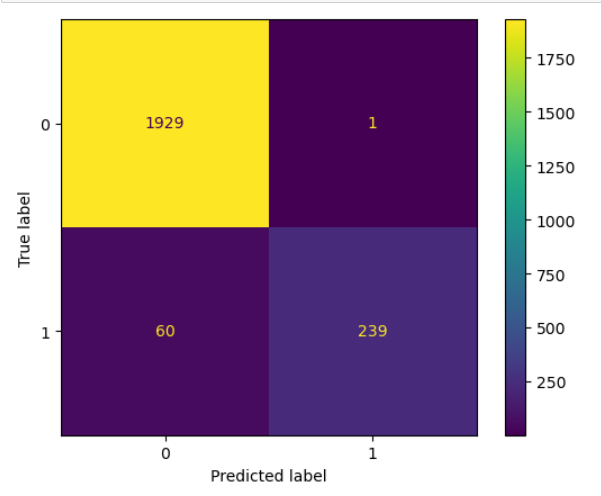
*Fig 7: Confusion matrix for test data set using Logistic Regression*



*Fig 8: Comparison between performance parameters of training and test dataset using Logistic Reg*

**Model 2: Naïve Baye’s Algorithm:** It is a probabilistic approach for constructing classification, it basically deals with likelihood that a data belongs to a specific class. We have used **multinomial naïve baye’s model** for our spam classification because it provided strong classification of textual data and needs small training dataset. It aims to assign fragments of text (i.e. documents) to classes by determining the probability that a document belongs to the class of other documents, having the same subject. NB algorithm is applied to the final extracted features. The speed and simplicity along with high accuracy of this algorithm makes it a desirable classifier for spam detection problems.

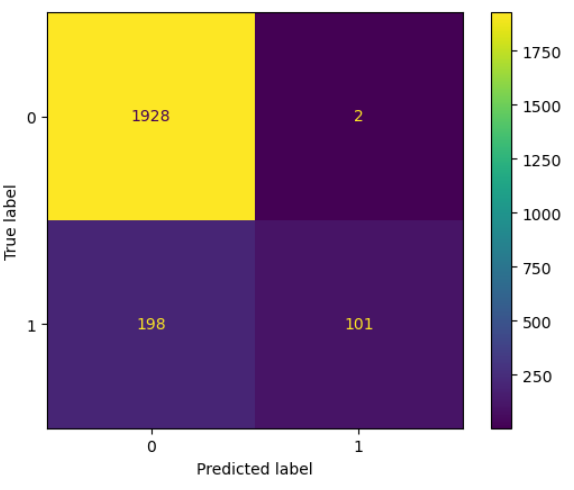
Results of Naïve Baye’s Algorithm:



*Fig 9: Confusion matrix for test data set using Naïve Baye’s*

**Model 3: K Nearest Neighbours Model:** KNN is one of the simplest supervised algorithms which is used to solve the classification problem. We have used KNN with n=5 neighbours in spam classification problem, it determines the 5 nearest datapoints of the test data and predict the classification of spam or ham messages of train data set.

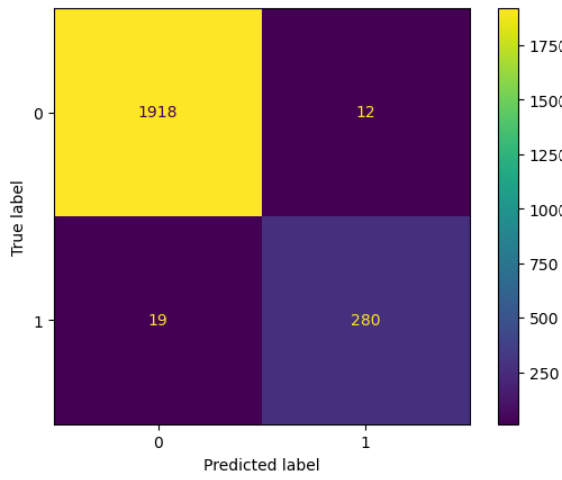
Result of K Nearest Neighbours Model:



*Fig 10: Confusion matrix for test data set using KNN*

**Model 4: Support Vector Machines:** SVM is yet another algorithm used for text classification problem. We plot each data item as a point in n-dimensional space (where n is the number of features you have), with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the optimal hyper-plane that differentiates the two classes very well. Support Vectors are simply the coordinates of individual observation, and a hyper-plane is a form of SVM visualization.

Results of Support Vector Machines:



*Fig 11: Confusion matrix for test data set using SVM*

**MODEL COMPARISON**

We will be comparing the above four models based on the 4 main features – Accuracy, F1 score, Recall and Precision.

**Accuracy**: Accuracy represents the number of correctly classified data instances over the total number of data instances.

Accuracy = TN+TP/(TN+FP+TP+FN)

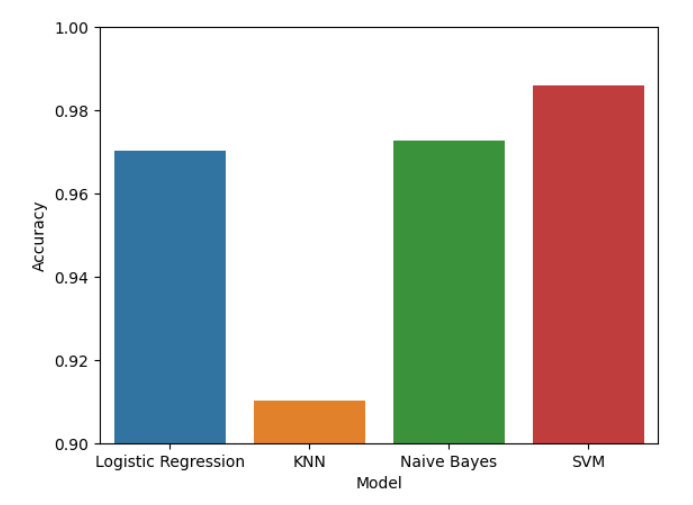
**Precision**: Precision is a metric that gives you the proportion of true positives to the amount of total positives that the model predicts. It answers the question **“Out of all the positive predictions we made, how many were true?”**

**Recall**: Recall focuses on how good the model is at finding all the positives. Recall is also called true positive rate and answers the question “**Out of all the data points that should be predicted as true, how many did we correctly predict as true?**”

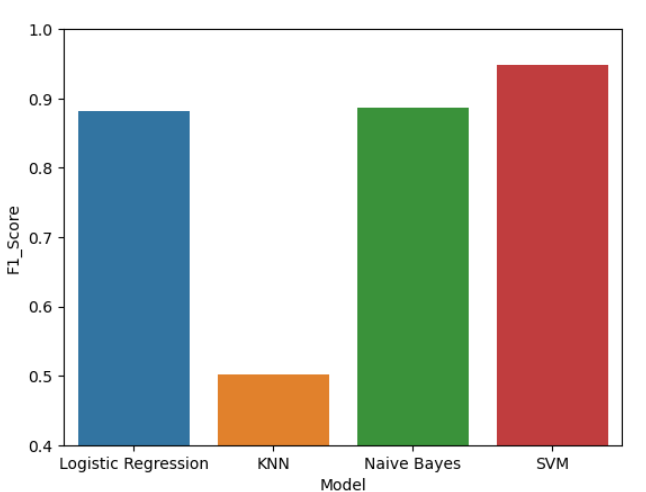
Recall = TP/TP+TN

**F1-Score**: F1 Score is a measure that combines recall and precision. As we have seen there is a trade-off between precision and recall, F1 can therefore be used to measure how effectively our models make that trade-off. F1-Score is harmonic mean of Precision and Recall value.

F1 Score = 2\*(Precision\*Recall/Precision Recall)



*Fig 12: Accuracy comparision of all models*

**

*Fig 13: F1 score comparision of all models*

**CONCLUSIONS:**

After looking the test accuracy score of the above four model we can clearly say that Support Vector Machine algorithm has the highest accuracy score of 98.61 for test data set and 99.94% for train data set. We can also see that Naïve Baye’s and Logistic Regression based approach has almost same accuracy score of 97%.

But as our dataset is unevenly distributed we can have to look for the precision values, f1-score and recalls values. Naïve Baye’s and Linear Regression model have f1-score of 0.89 respectively which is lesser than SVM model which has f1-score of 0.96, thus it is the best suited algorithm amongst all other implemented models.

**REFERENCES:**

[1] SMS Spam Collection Data Set from UCI Machine Learning Repository, <http://archive.ics.uci.edu/ml/datasets/SMS+Spam+Collection>

[2] SMS Spam Collection v.1, [http://www.dt.fee.unicamp.br/\_tiago/ smsspamcollection](http://www.dt.fee.unicamp.br/_tiago/%20smsspamcollection)

[3] https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset