Report - PRML Minor Project

Identify and Eliminate Spam Comments

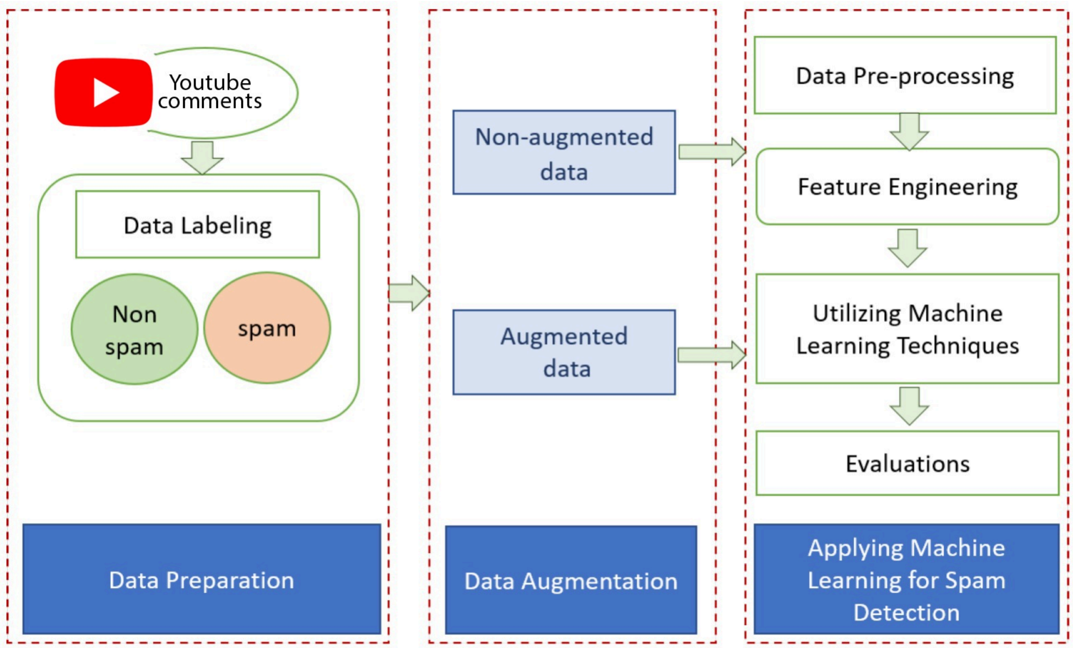
- Marri Bharadwaj (B21CS045)
Mohit Kumawat (B21CS046)
Shrashti Saraswat (B21CS081)

Abstract

This paper reports our experience with building a model for identifying and removing spam comments from the given dataset containing videos of four YouTubers. The dataset contains more than 350,000 comments which helped us understand the contents of a spam comment, the intention of their authors and how to identify and eliminate one. We used different supervised and unsupervised learning algorithms and compared their results in this report.

I. Introduction

Spam comments are irrelevant or unwanted messages posted in online forums, blogs, or social media platforms. They are usually posted by spammers, who try to promote their products, services or websites using automated bots or manual methods; or even others who post meaningless comments which render the author-viewers' interaction disturbed. Spam comments can harm online communities, reduce user engagement and damage the reputation of the affected websites or platforms. To solve this problem, we have been provided with the comments posted in the comment section, on the YouTube platform, of four different YouTubers. Using this, our job is to cluster and identify the spam comments from the non-spam comments and eliminate them.



Dataset

YouTube Comments Data: The dataset provided was the YT_Videos_Comments.csv dataset. It contains the comments of the videos of four YouTube channels- Cleo Abram, Physics Girl, Jet Lag: The Game and Neo. The dataset consists of 379528 rows, where each row represents an instance of a posted comment with 9 columns containing:

- User: Name of the YouTube channel
- Video Title: Title of the respective video
- Video Description: Description of the respective video
- Video ID: YouTube ID of the respective video
- Comment (Displayed): comment modified to HTML characters
- Comment (Actual): the actual comment posted by the author
- Comment Author: person who posted the comment
- Comment Author Channel ID: YouTube ID of the comment's author
- Comment Time: time at which the comment was posted

II. Methodology

Overview

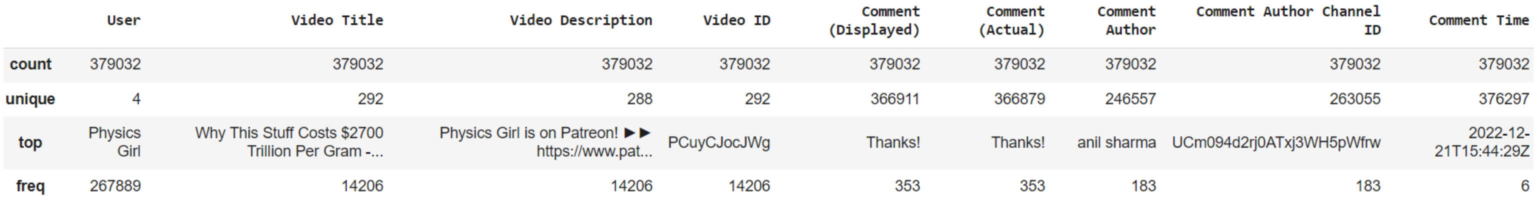
There are various supervised and unsupervised machine learning algorithms present out of which we implemented the following:

- K-Means Clustering
- Logistic Regression
- Gaussian Naïve Bayes Classifier
- Decision Tree Classifier
- K-Nearest Neighbours

We also make use of the Word Embedding technique initially.

Exploratory Data Analysis and Pre-Processing

- We installed the necessary library packages for cleaning the comments and pre-processing; and then mounted our Google Drive to provide access to the dataset uploaded on it.
- We checked for null values and dropped them. We then received a holistic summary about the dataset as follows:



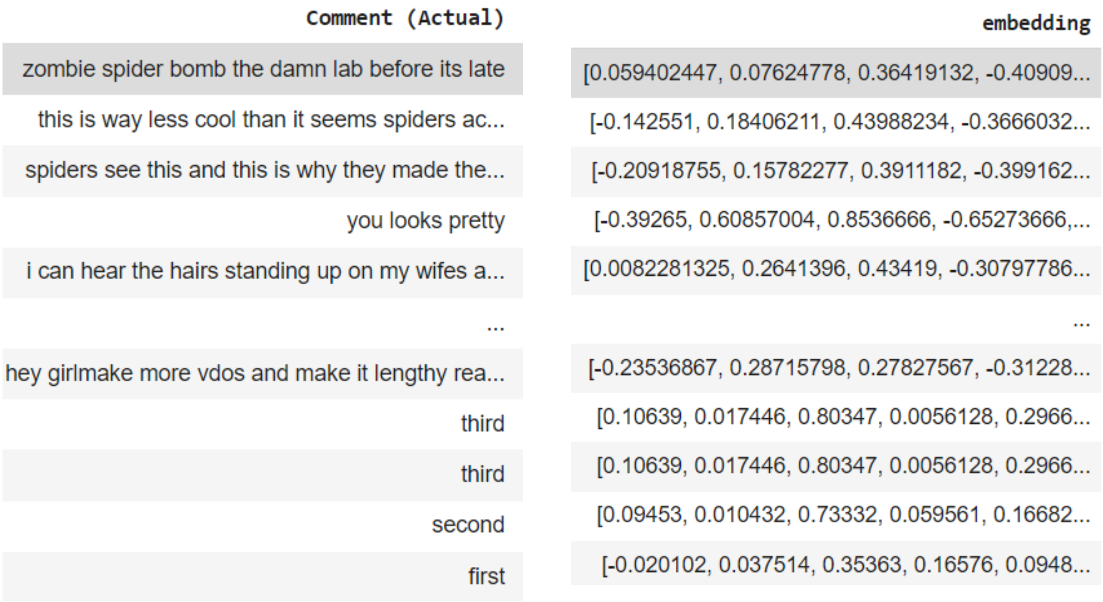
	User	Video Title	Video Description	Video ID	Comment (Displayed)	Comment (Actual)	Comment Author	Comment Author Channel ID	Comment Time
count	379032	379032	379032	379032	379032	379032	379032	379032	379032
unique	4	292	288	292	366911	366879	246557	263055	376297
top	Physics Girl	Why This Stuff Costs $2700 Trillion Per Gram -...	Physics Girl is on Patreon! ►► https://www.pat...	PCuyCJocJWg	Thanks!	Thanks!	anil sharma	UCm094d2rj0ATxj3WH5pWfrw	2022-12-21T15:44:29Z
freq	267889	14206	14206	14206	353	353	183	183	6

- We dropped unnecessary columns such as User, Video Title, Video Description, Comment (Displayed) and Comment Author. We also dropped emojis, unnecessary punctuation marks and lowered the text.

Implementation of Different Algorithms and Techniques

- *Word Embedding:* Word embedding is a technique to represent words or phrases as numerical vectors in a high-dimensional space. This is done by training a model on a large corpus of text data to learn the relationships between words and their contextual meanings. The resulting word embeddings capture semantic and syntactic relationships between words, and are useful for a variety of natural language processing tasks.

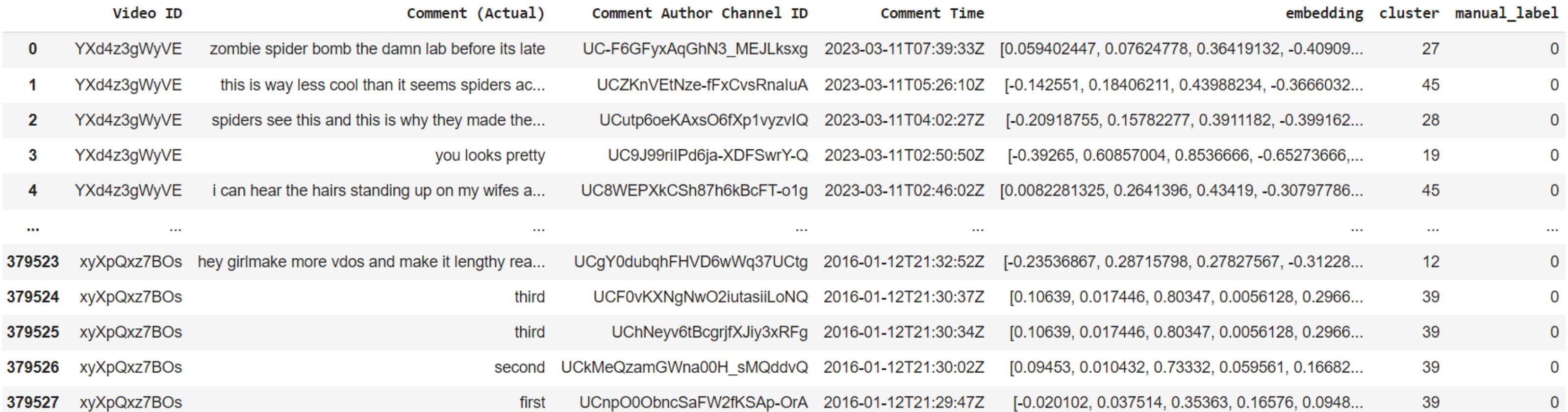
Thus, after applying word embedding to the dataset, each instance, i.e., comment, was converted into a n-dimensional array.



Comment (Actual)	embedding
zombie spider bomb the damn lab before its late	[0.059402447, 0.07624778, 0.36419132, -0.40909...
this is way less cool than it seems spiders ac...	[-0.142551, 0.18406211, 0.43988234, -0.3666032...
spiders see this and this is why they made the...	[-0.20918755, 0.15782277, 0.3911182, -0.399162...
you looks pretty	[-0.39265, 0.60857004, 0.8536666, -0.65273666,...
i can hear the hairs standing up on my wifes a...	[0.0082281325, 0.2641396, 0.43419, -0.30797786...
...	...
hey girlmake more vdos and make it lengthy rea...	[-0.23536867, 0.28715798, 0.27827567, -0.31228...
third	[0.10639, 0.017446, 0.80347, 0.0056128, 0.2966...
third	[0.10639, 0.017446, 0.80347, 0.0056128, 0.2966...
second	[0.09453, 0.010432, 0.73332, 0.059561, 0.16682...
first	[-0.020102, 0.037514, 0.35363, 0.16576, 0.0948...

- *K-Means Clustering:* K-means is a popular unsupervised machine learning algorithm used for clustering data points into groups or clusters based on their similarity. The algorithm works by iteratively grouping data points together until the specified number of clusters, represented by the parameter k, is reached. The K-means algorithm assigns each data point to the nearest cluster centroid, which is the mean of all the points in that cluster. It then recalculates the centroids of each cluster based on the new groupings, and repeats the process until convergence is achieved.
Hence, we have applied this algorithm to the embedded dataset by setting the numbers of clusters as 50; and generated the clusters.

- *Manual Labelling:* Manual labelling is a process in which we assign labels or categories to data points based on their characteristics or features. This process is commonly used in supervised machine learning, where a labelled dataset is required to train a machine learning model. The use of manual labelling can improve the performance of machine learning models by providing them with high-quality labelled data that accurately reflects the characteristics of the real-world data.
We analysed 20 sample instances from each of the 50 clusters and assigned them a class of 0 or 1 in the newly-created 'manual_label' column; with 0 indicating spam and 1 indicating non-spam.



	Video ID	Comment (Actual)	Comment Author Channel ID	Comment Time	embedding	cluster	manual_label
0	YXd4z3gWyVE	zombie spider bomb the damn lab before its late	UC-F6GFyxAqGhN3_MEJLksxg	2023-03-11T07:39:33Z	[0.059402447, 0.07624778, 0.36419132, -0.40909...	27	0
1	YXd4z3gWyVE	this is way less cool than it seems spiders ac...	UCZKnVEtNze-fFxCvsRnaluA	2023-03-11T05:26:10Z	[-0.142551, 0.18406211, 0.43988234, -0.3666032...	45	0
2	YXd4z3gWyVE	spiders see this and this is why they made the...	UCutp6oeKAxsO6fXp1vyzvIQ	2023-03-11T04:02:27Z	[-0.20918755, 0.15782277, 0.3911182, -0.399162...	28	0
3	YXd4z3gWyVE	you looks pretty	UC9J99riIPd6ja-XDFSwrY-Q	2023-03-11T02:50:50Z	[-0.39265, 0.60857004, 0.8536666, -0.65273666,...	19	0
4	YXd4z3gWyVE	i can hear the hairs standing up on my wifes a...	UC8WEPXkCSh87h6kBcFT-o1g	2023-03-11T02:46:02Z	[0.0082281325, 0.2641396, 0.43419, -0.30797786...	45	0
...	...	...	...	...	...	...	...
379523	xyXpQxz7BOs	hey girlmake more vdos and make it lengthy rea...	UCgY0dubqhFHVD6wWq37UCtg	2016-01-12T21:32:52Z	[-0.23536867, 0.28715798, 0.27827567, -0.31228...	12	0
379524	xyXpQxz7BOs	third	UCF0vKXNgNwO2iutasiiLoNQ	2016-01-12T21:30:37Z	[0.10639, 0.017446, 0.80347, 0.0056128, 0.2966...	39	0
379525	xyXpQxz7BOs	third	UChNeyv6tBcgrjfXJiy3xRFg	2016-01-12T21:30:34Z	[0.10639, 0.017446, 0.80347, 0.0056128, 0.2966...	39	0
379526	xyXpQxz7BOs	second	UCkMeQzamGWna00H_sMQddvQ	2016-01-12T21:30:02Z	[0.09453, 0.010432, 0.73332, 0.059561, 0.16682...	39	0
379527	xyXpQxz7BOs	first	UCnpO0ObncSaFW2fKSAp-OrA	2016-01-12T21:29:47Z	[-0.020102, 0.037514, 0.35363, 0.16576, 0.0948...	39	0

- *Database for Supervised Learning:* We have created a new database, for implementing supervised learning algorithms, consisting of 2000 non-spam instances and 500 spam instances and named it ‘data_for_supervised’. Next, we flattened and normalised it; and this created the training and testing data for the upcoming supervised learning algorithms.

Next, we have used three Supervised Learning methods on this database-

- *Logistic Regression:* Logistic regression is an algorithm used for binary classification problems, where the goal is to predict whether a data point belongs to one of two classes based on its features. It works by modelling the probability of a data point belonging to a particular class as a function of its features. It uses a logistic function, also known as the sigmoid function, to transform the output into a probability value between 0 and 1.

- *Gaussian Naïve Bayes Classifier:* Gaussian Naive Bayes Classifier is a popular probabilistic algorithm used for classification tasks. It is based on Bayes' theorem and assumes that the features of a data point are independent of each other, given the class label. The algorithm works by calculating the conditional probability of a class given the feature values, using the Gaussian probability density function to estimate the likelihood of each feature value. It then combines the likelihood values with the prior probability of each class to obtain the posterior probability of each class, and predicts the class with the highest probability.

- *Decision Tree Classifier:* Decision Tree Classifier is a popular machine learning algorithm used for classification and regression tasks. It works by recursively partitioning the feature space into smaller subsets, based on the most informative features, until a stopping criterion is met. The algorithm builds a tree-like structure where each node represents a feature and each branch represents a possible value of that feature. The decision tree algorithm selects the most informative feature at each node based on a criterion such as information gain or Gini impurity, and partitions the data based on the selected feature.

- *K-Nearest Neighbours:* KNN regression is a non-parametric machine learning algorithm that, in an intuitive manner, approximates the association between independent variables and the continuous outcome by averaging the observations in the same neighbourhood. Here ‘k’ is the number of nearest neighbours to be considered in the majority voting process and can be chosen based on the size of the dataset and the complexity of the problem.

Hence, we have applied this algorithm to our clustering on the labelled dataset by setting the numbers of nearest neighbours to 5.

III. Evaluation of Algorithms

We have two evaluation results here- first, evaluating the supervised learning algorithms on our ‘data_for_supervised’ database; and the evaluation of the K-Means algorithm using K-Nearest Neighbours.

- ❖ Evaluation of Logistic Regression, Gaussian Naïve Bayes Classifier and Decision Tree Classifier using-

- *Accuracy:* Accuracy is a common evaluation metric used in machine learning to measure how well a model predicts the correct class labels for a given dataset. It is calculated by dividing the number of correct predictions by the total number of predictions made by the model.
- *Precision:* Precision is a common evaluation metric used in machine learning to measure how well a model predicts the true positive class labels for a given dataset. It is calculated by dividing the number of true positive predictions by the total number of positive predictions made by the model. It is particularly useful when the cost of false positive predictions is high.
- *Recall:* Recall is a common evaluation metric used in machine learning to measure how well a model captures all the positive instances in a dataset. It is calculated by dividing the number of true positive predictions by the total number of actual positive instances in the dataset. It is particularly useful when the cost of false negative predictions is high.
- *F1 Score:* F1 score is a common evaluation metric used in machine learning to balance the tradeoff between precision and recall. It is the harmonic mean of precision and recall, and ranges between 0 and 1, with higher values indicating better performance. It is useful when both precision and recall are important, and can be used to compare models with different precision-recall tradeoffs.

Thus, when applying these evaluation metrics, we have got the following report-

Evaluation metrics for *Logistic Regression*:

- Accuracy: 0.962
- Precision: 0.865
- Recall: 0.375

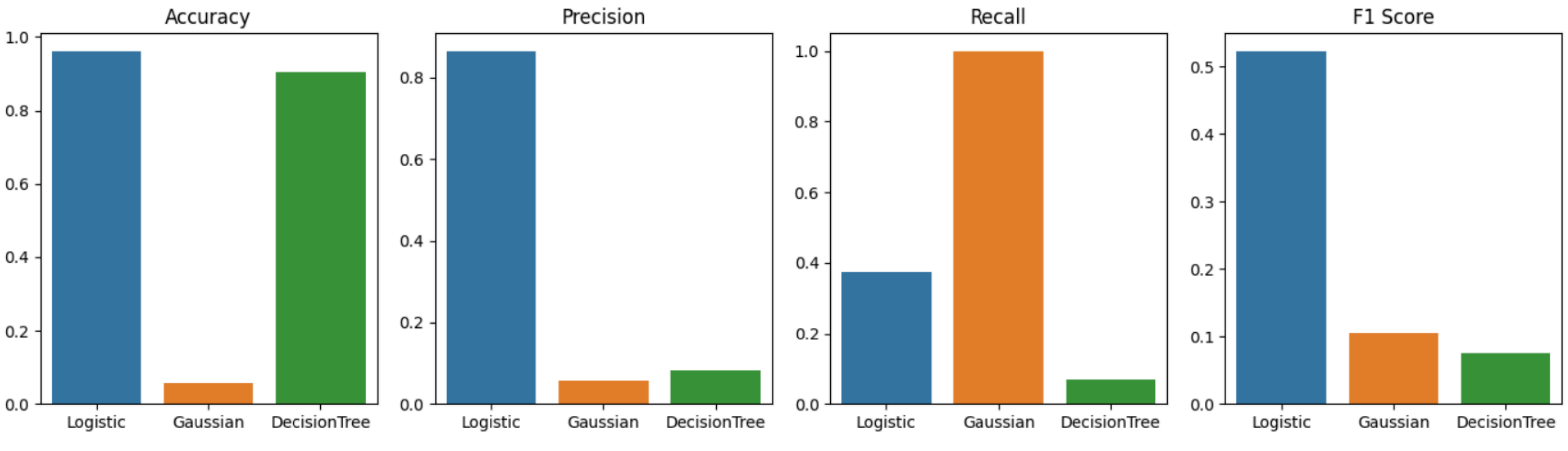
F1 Score: 0.523

Evaluation metrics for *Gaussian Naïve Bayes Classifier*:

- Accuracy: 0.056
- Precision: 0.056
- Recall: 1.000
- F1 Score: 0.106

Evaluation metrics for *Decision Tree Classifier*:

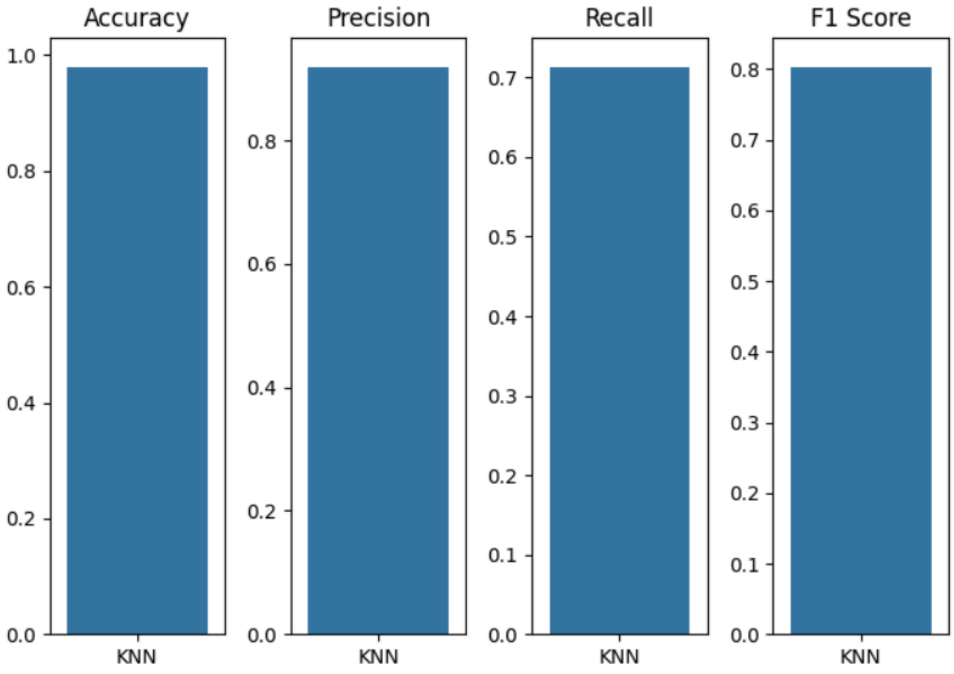
- Accuracy: 0.903
- Precision: 0.082
- Recall: 0.070
- F1 Score: 0.076



- Evaluation of K-Means Clustering Algorithm by applying K-Nearest Neighbours technique-

  We have the same above four evaluation metrics- Accuracy, Precision, Recall and F1 Score and got the report-

  - Accuracy: 0.9801424676809427
  - Precision: 0.9198966408268734
  - Recall: 0.7138670368656487
  - F1 Score: 0.8038909154073302



IV. Analysis of Results generated

From the above scores we can say that LR seems to be biassing the data by not correctly identifying the spam comments. Its precision and recall scores are lower than its accuracy, indicating that it may have struggled with correctly identifying positive instances in the data.

Whereas The Naive Bayes gave biassed results only in favour of the spam comments, i.e. it classified all the comments as spam giving it a perfect recall score but very less accuracy and precision. The poor performance of Naive could be due to it assuming that the input variables are independent.

While the Decision tree is opposite of Naive Bayes, Its precision score is very low, which means it made many false positive predictions. Its recall score is also low, indicating that it failed to identify many positive instances in the data.This could be due to the fact that Decision Tree is prone to overfitting if not properly regularised or if the depth of the tree is too high.

Also, for the KNN technique, we can see that the final accuracy score is 98%, while maintaining an F1-score of 0.8. This tells us that our pipeline is able to detect spam vs non-spam without biassing between both.