

# SMS SPAM DETECTION USING MACHINE LEARNING

- [1] Mr. E.Sankar, Assistant Professor, CSE Department,  
Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya,  
Enathur, Kancheepuram.
- [2] Y Y S Shekhar Babu, B.E, 4<sup>th</sup> Year, CSE Branch,  
Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya,  
Enathur, Kancheepuram.
- [3] M.Trivedi, B.E, 4<sup>th</sup> Year, CSE Branch,  
Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya,  
Enathur, Kancheepuram.

**ABSTRACT:** In recent times, there has been considerable interest among people to use short communication service (SMS) as one of the essential and straightforward dispatches services on mobile bias. The increased popularity of this service also increased the number of mobile bias attacks similar as SMS spam dispatches. SMS spam dispatches constitute a real problem to mobile subscribers. this worries telecommunication service providers as it disturbs their guests and causes them to lose business. to enhanced SMS spam filtering performance by combining two of data mining task association and bracket. FP- growth in association is employed for mining frequent pattern on SMS and Naive Bayes Classifier is used to classify whether SMS is spam or ham. Training data was using SMS spam collection from former exploration. The result of using collaboration of Naive Bayes and FP- Growth performs the highest average accuracy of 98,506 and 0.025% better than without using FP- Growth for dataset SMS Spam Collection v. 1, and improves the perfection score; therefore, the bracket result is more accurate.

**Keywords:** Spam, Machine Learning, Naïve Bayes Algorithm,  
SMS (Short Message Service), Vectorizer.

**Introduction:** SMS is a text-based communication media that allows mobile phone users to partake in a short text message (generally it's well beyond 160 characters in 7-bit). Along with the wide use and popularity as the most important dispatches media, there are plenty of those who use it for marketable purposes similar as advertising media and indeed fraud. The reduced SMS rate is one of the causes of adding SMS spam as well, as in China, the SMS tariff is well under than \$0.001. Also, grounded on the Korea Information Security (KISA), this quantum exceeds the dispatch spam. For case, mobile users in the US earnings 1.1 billion SMS spams, and Chinese users receives 29 billion SMS spams daily.

There is a result that could be performed to break the below problems. It is by filtering SMS grounded on the text-based bracket. There are some popular text-based groups ways, including decision trees, Naive Bayes, rule induction, neural network, nearest neighbors, and Support Vector Machine. Nonetheless, this SMS bracket is different from the bracket on a regular document text-based e-mail due to the veritably short text-based (160 7-bit characters maximum), plenty of abbreviated text-based, and tend to be informal text-based in SMS.

### **Literature Survey:**

[1] SMS (Short Message Service) is still the primary choice as a communication medium even though nowadays mobile phone is growing with a variety of communication media messenger applications.

[2] In the modern world where digitization is everywhere, SMS has become one of the most vital forms of communications, unlike other chatting-based messaging systems like Facebook, WhatsApp etc., SMS does not require active internet connection at all.

[3] Despite the fact that in many parts of the world, versatile informing channel is right now viewed as "spotless" and trusted, on the complexity ongoing reports obviously show that the volume of cell phone spam is drastically expanding step by step.

[4] Due to its convenient, economical, fast, and easy to use nature Electronic mail is a vital revolution taking place over traditional communication systems. A main obstruction in electronic communications is the vast

publicizing of unwanted, harmful emails known as spam emails. Lots of time of client is being wasted for sorting approaching mail and erasing undesirable correspondence, so there is a need for spam detection so that its outcomes can be reduced. The main aim is to development of suitable filters that can appropriately detect those emails and results in a high-performance rate.

[5] Over recent years, as the popularity of mobile phone devices has increased, Short Message Service (SMS) has grown into a multi-billion dollars industry. At the same time, reduction in the cost of messaging services has resulted in growth in unsolicited commercial advertisements (spams) being sent to mobile phones. In parts of Asia, up to 30% of text messages were spam in 2012. Lack of real databases for SMS spams, short length of messages and limited features, and their informal language are the factors that may cause the established email filtering algorithms to underperform in their classification. In this project, a database of real SMS Spams from UCI Machine Learning repository is used, and after pre-processing and feature extraction, different machine learning techniques are applied to the database. Finally, the results are compared and the best algorithm for spam filtering for text messaging is introduced. Final simulation results using 10-fold cross validation shows the best classifier in this work reduces the overall error rate of best model in original paper citing this dataset by more than half.

[6] In the recent advanced society the online social networking sites like Twitter, Facebook, LinkedIn are very popular. Twitter, an online Social Networking site, is one of the most visited sites. Lot of users communicates with each other using Twitter. The rapidly growing social network Twitter has been infiltrated by large amount of spam. As Twitter spam is not similar to traditional spam, such as email and blog spam, conventional spam filtering methods are not appropriate and effective to detect it. Thus, many researchers have proposed schemes to detect spammers.

[7] Spam involves sending someone unwanted messages. Currently the internet is the biggest platform to get some information, also social media is going to be very popular nowadays. Because of that, many spammers will try to mislead users by sending lots of spam messages. And because of spam messages, there are lots of problems, fraud occurs. So we want to filter messages into spam or ham. To classify the messages as spam or not spam we are using machine learning (the multinomial naïve Bayes classifier algorithm) and Count Vectorizer provided by Scikit-learn library in python programming. First, we collect the datasets and convert

them into numerical data (matrix) by Count Vectorizer and then we apply the naïve Bayes algorithm on datasets for classification purposes

[8] presents detection of Spam and ham messages using various supervised machine learning algorithms like naïve Bayes Algorithm, support vector machines algorithm, and the maximum entropy algorithm and compares their performance in filtering the Ham and Spam messages. As people indulge more in Web-based activities, and with rising sharing of private – data by companies, SMS spam is very common. SMS spam filter inherits much functionality from E-mail Spam Filtering. Comparing the performance of various supervised learning algorithms, we find the support vector machine algorithm gives us the most accurate result.

**Problem Statement:** Short Message Service (SMS) is one of the well- known communication services in which a communication sends electronically. The lessening in the cost of SMS benefits by telecom associations has urged the expanded application of SMS. This ascent pulled in assaulters, which have brought about SMS Spam problem. Spam dispatches include announcements, free services, elevations, awards, etc. People are using the ubiquity of mobile phone bias is expanding day by day as they give a vast variety of services by reducing the cost of services. Short Communication Service (SMS) is one of the astronomically employed communication service. In any case, this has urged an expansion in mobile phones attacks like SMS Spam. In this problem, primary results are mentioned or explained herein grounded on Singapore grounded intimately available datasets. This problem is farther expanded using multiple background datasets.

**Proposed System:** Several techniques have been suggested to combat spam. To automatically detect spam, researchers have implemented data mining algorithms to make spam detection a classification issue. Data mining has many types but, in this research, spamming extraction from tweets by using deep learning and machine learning. Machine learning uses two main techniques: Supervised learning allows you to collect data or produce a data output from a previous ML deployment. Unsupervised learning finds the hidden pattern or data grouping without the requirement for human intercession.

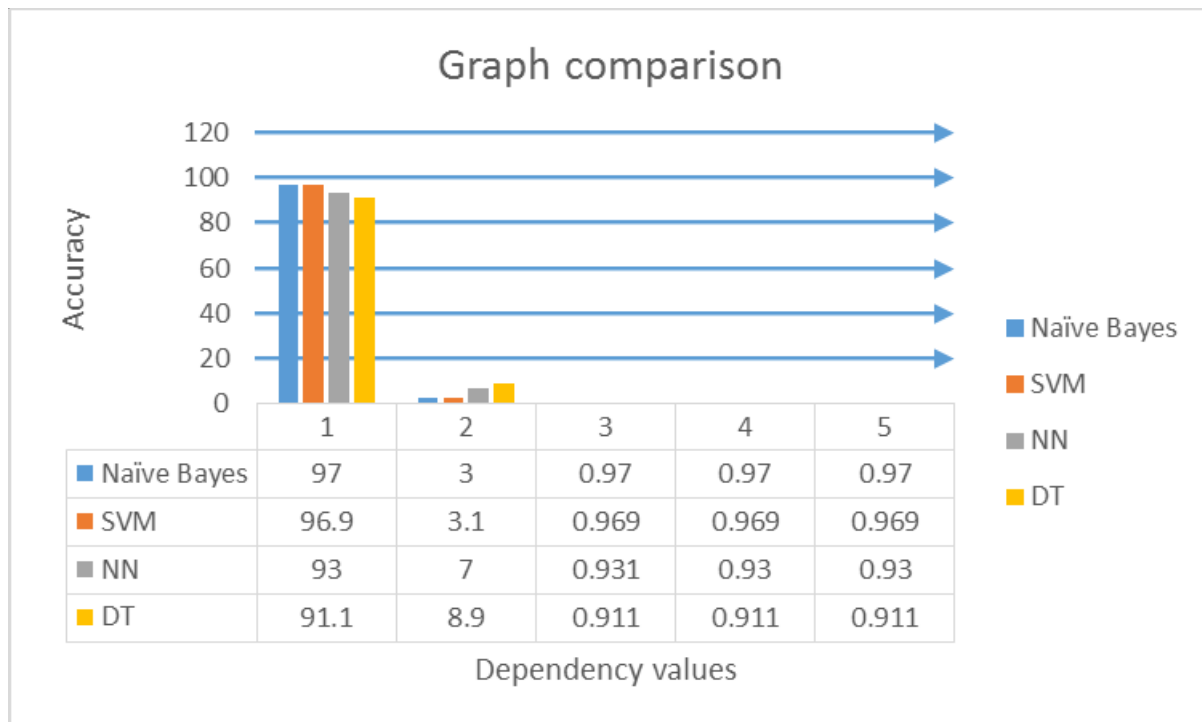
**Feasibility Algorithm:**

Comparison of algorithm aims to get the algorithm that's considered the fastest and accurate to make a vaticination of a problem. Result of comparison of Naive Bayes and K-Nearest Neighbor algorithm can be concluded that Naive Bayes algorithm yield 98.03% accuracy and K-Nearest Neighbor algorithm yield 88.52% accuracy.

Naive Bayes is better classifier than decision tree and k-NN with the accuracy percentage 98.03% and the accuracy rate for decision tree was 96.86% and for K- NN, it was 88.52% which is much lower than Naive Bayes accuracy value.

S.L.NO	Algorithm	Accuracy
1	Naïve Bayes	98.03%
2	K-NN	88.52%
3	Decision Tree	96.86%

Method/ Parmeter	Decision Tree	Naïve Bayesian	KNN
Understandibility	Simple to understand and generate	Easy to Understand and build	Easy to understand
Data Type	Numerical and categorical	Numerical and categorical	Numerical and categorical
Deterministic/ Non deterministic	Deterministic	Non Deterministic	Non Deterministic
Effectiveness on	Large data	Huge data	Large data
Applicable	Pattern Recognition, Sequence Recognition, Financial Applications	Text Classification, Spam Filtering	Text Classification, Decision Making



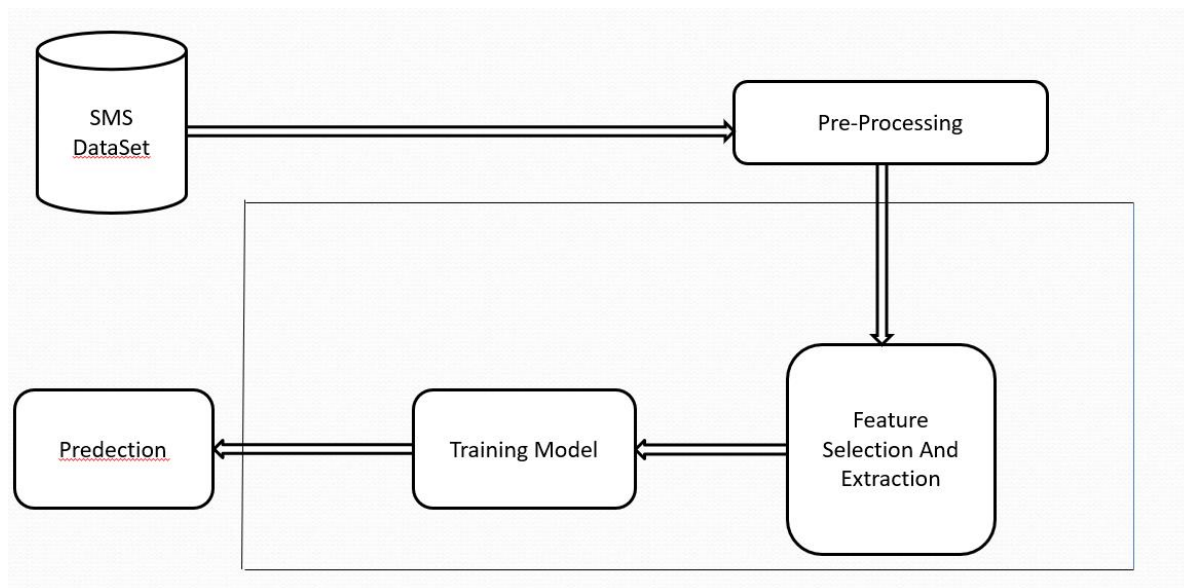
**Algorithm:** The Naive Bayes bracket algorithm is a probabilistic classifier. It's grounded on probability models that incorporate strong independence hypotheticals. The independence hypotheticals frequently do not have an impact on reality. thus, they're considered as naive. Naive Bayes styles are a set of supervised literacy algorithms grounded on applying Bayes' theorem with the “naive” supposition of tentative independence between every brace of features given the value of the class variable.

**Naive Bayes Classifier:** Naive Bayes classifiers belong to the probabilistic classifiers and are largely scalable, taking several parameters direct in the number of variables (features predictors) in a literacy problem. Maximum- liability training can be done by assessing a unrestricted- form expression, which takes direct time, rather than by precious iterative approximation as used for numerous other types of classifiers. This system uses a mongrel approach using both rule- grounded systems and automatic systems in order to achieve advanced delicacy of sentiment computation. ‘Text Blob’ library has been used for recycling textbook data. It provides an API for natural language processing (NLP) tasks.

**Algorithm:**

1. Consider a training data set D consists of documents which belong to different classes say class A and B.
2. Prior probability of both classes A and B is calculated as shown Class A=number of objects of class A / total number of objects. Class B=number of objects of class B / total number of objects.
3. Now calculate the total number of word frequencies of both classes A and B i.e.,  $n_i$  na = the total number of word frequencies of class A.  $n_b$  =the total number of word frequency of class B.
4. Calculate the conditional probability of keyword occurrence for given class  $P(\text{word1} / \text{class A}) = \text{wordcount} / n_i(A)$   $P(\text{word1} / \text{class B}) = \text{wordcount} / n_i(B)$   $P(\text{word2} / \text{class A}) = \text{wordcount} / n_i(A)$   $P(\text{word2} / \text{class B}) = \text{wordcount} / n_i(B)$  . . . . .  $P(\text{word n} / \text{class B}) = \text{wordcount} / n_i(B)$
5. Uniform distributions are to be performed in order to avoid zero frequency problems.
6. Now a new document M is classified based on calculating the probability for both classes A and B  $P(M/W)$ . a) Find  $P(A / W) = P(A) * P(\text{word1}/\text{class A}) * P(\text{word2}/ \text{class A}) \dots * P(\text{word n} / \text{class A})$ . b) Find  $P(B / W) = P(B) * P(\text{word1}/\text{class B}) * P(\text{word2}/ \text{class B}) \dots * P(\text{word n} / \text{class B})$ .
7. After calculating probability for both classes A and B the class with higher probability is the one the new document M assigned.



**System Architecture:**

**Result:** From the results we can clearly see that Naïve Bayes performs clearly in classifying the reviews as OR or CB followed by Random Forest classifier and KNN is averagely performing by giving only 88.52% accuracy. Thus, we can conclude that Naïve Bayes is a clear winner in detecting fake reviews.

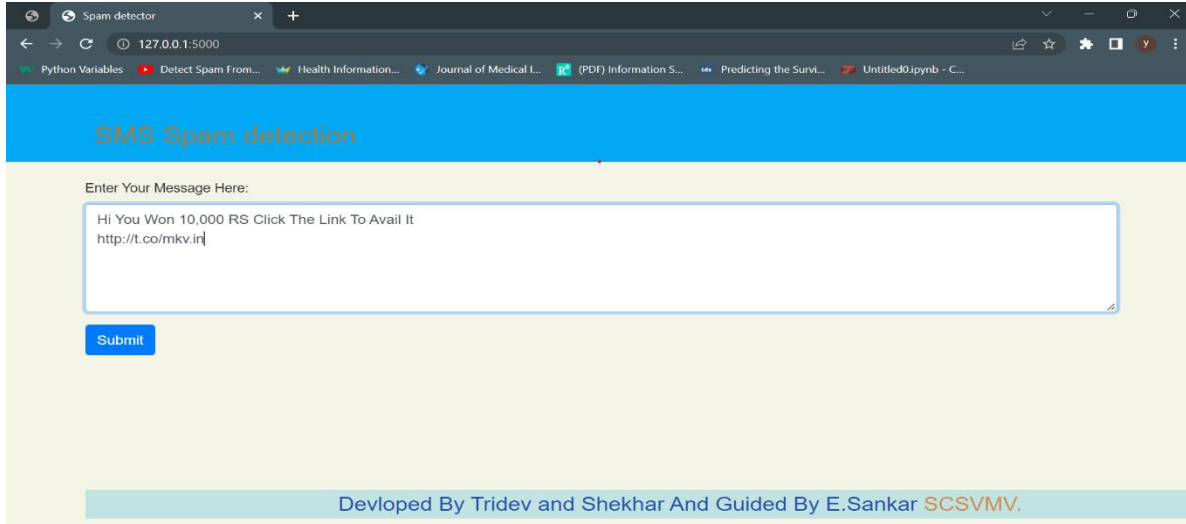
```
from sklearn.metrics import accuracy_score
print("Naive_bayes:",str(np.round(accuracy_score(y_test,classifier_y_pred)*100,2)) + '%')
print("knn:",str(np.round(accuracy_score(y_test,classifier1_y_pred)*100,2)) + '%')
print("Decision tree:",str(np.round(accuracy_score(y_test,classifier2_y_pred)*100,2)) + '%')
print("LogisticRegression:",str(np.round(accuracy_score(y_test,classifier3_y_pred)*100,2)) + '%')
```

```
Naive_bayes: 98.03%
knn: 88.52%
Decision tree: 96.86%
LogisticRegression: 96.86%
```



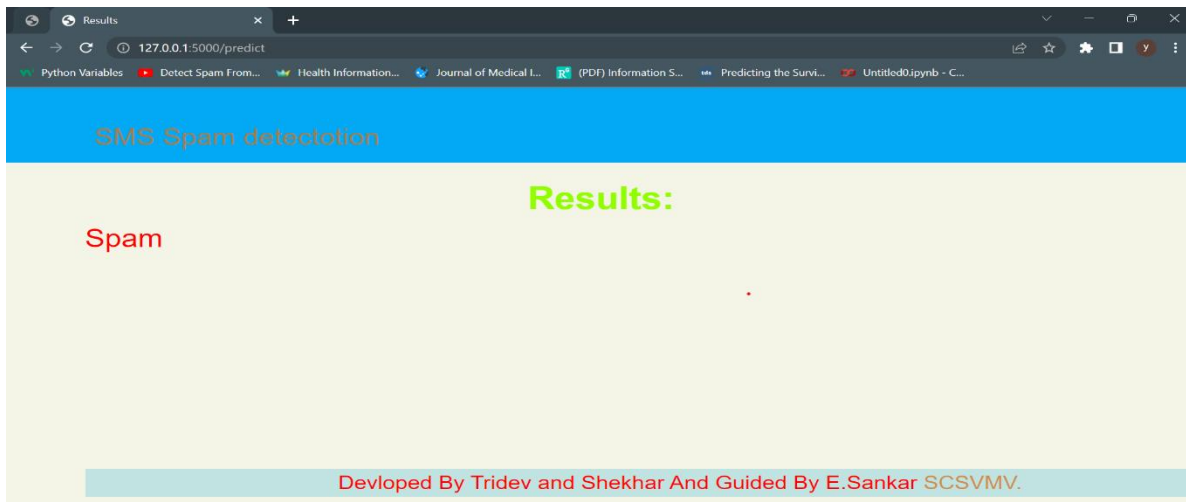
## Output:

Giving Input To Detect The Message



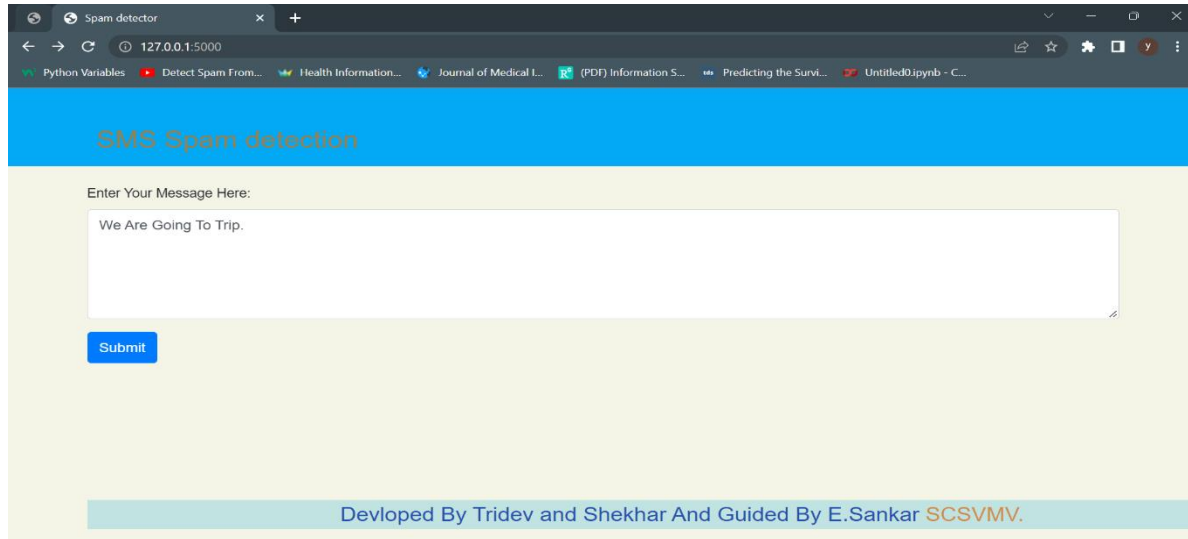
The screenshot shows a web browser window with the title "Spam detector". The address bar shows "127.0.0.1:5000". The page has a blue header with the text "SMS Spam detection". Below the header, there is a text input field with the placeholder "Enter Your Message Here:". The input field contains the text "Hi You Won 10,000 RS Click The Link To Avail It http://t.co/mkv.ir". Below the input field is a blue "Submit" button. At the bottom of the page, there is a light blue footer with the text "Developed By Tridev and Shekhar And Guided By E.Sankar SCSVMV."

Given Message Is Detected As A Spam



The screenshot shows the same web browser window, but the page now displays the result. The header "SMS Spam detection" is still present. Below the header, the word "Spam" is displayed in red text. To the right of "Spam", the word "Results:" is displayed in green text. At the bottom of the page, the same light blue footer with the text "Developed By Tridev and Shekhar And Guided By E.Sankar SCSVMV." is visible.

## Giving Input To Detect The Message



The screenshot shows a web browser window with the title "Spam detector". The address bar shows "127.0.0.1:5000". The page has a blue header with the text "SMS Spam detection". Below the header, there is a text input field with the placeholder "Enter Your Message Here:". The input field contains the text "We Are Going To Trip.". Below the input field is a blue "Submit" button. At the bottom of the page, there is a light blue footer with the text "Developed By Tridev and Shekhar And Guided By E.Sankar SCSVMV."

## Given Input Is Detected As A Ham



The screenshot shows the same web browser window as before, but the address bar now shows "127.0.0.1:5000/predict". The page has a blue header with the text "SMS Spam detection". Below the header, there is a green "Results:" label. Below the label, there is a blue text box with the text "Not a Spam (It is a Ham)". At the bottom of the page, there is a light blue footer with the text "Developed By Tridev and Shekhar And Guided By E.Sankar SCSVMV."

**Conclusion:** From the above discussion and experimentation authors have concluded that Machine Learning algorithms can play a vital role in identifying SPAM SMS. The accuracy obtained in this work is more than 95% in both the cases. The aims and objectives of the project, which achieved throughout the course, defined at the very first stage of the process. To collect all the information, the research work involved a careful study on the different filtering algorithms and existing anti-spam tools. These large-scale research

papers and existing software programs are one of the sources of inspiration behind this project work. The whole project was divided into several iterations. Each iteration was completed by completing four phases: inception, where the idea of work was identified; elaboration, where architecture of the part of the system is designed; construction, where existing code is implemented; transition, where the developed part of the project is validated. However, there are still some parts that can be improved: for example, adding additional filtering techniques or changing aspects of the existing ones. The changes such as incrementing or decrementing the number of interesting words of the message and reorganizing the formula for calculating interesting rate can be done later.

**References:**

- [1] Enhancing Spam Detection on Mobile Phone Short Message Service (SMS) Performance using FP-Growth and Naive Bayes Classifier BY Dea Delvia, Arifin, Shaufiah Moch. Arif Bijaksana IN 2016.
- [2] SMS Spam Detection Using Machine Learning BY Suparna Das gupta , Soumyabrata Saha, Suman Kumar Das IN 2020.
- [3] Spam Detection In Sms Using Machine Learning Through Text Mining BY M.Rubin Julis, S.Alagesan in 2020
- [4] MACHINE LEARNING BASED SPAM DETECTION SYSTEM BY Saurabh Masurkar, Arjunsingh Rajput, Anish Angane, Simran Madaan, Shaveta Malik in 2020
- [5] SMS Spam Detection using Machine Learning Approach BY Houshmand Shirani-Mehr in 2017
- [6] Classification Methods for Spam Detection In Online Social Network BY SUPRIYA RAMHARI MANWAR, Prof. P.D. LAMBHATE, Prof. J. S. PATIL, in 2017.
- [7] Implementation of Spam Classifier using Naïve Bayes Algorithm BY  
Ajay Gangare, Jitesh Rathore, Akash Kumar Tadge, Abhinav Shrivastav, Rashmi Yadav,Pankaj Singh Sisodiya, in 2022.
- [8] SMS Spam Filtering using Supervised Machine Learning Algorithms BY Pavas Navaney, Gurav Dubey, Ajay Rana, in 2018.