Journal of Information Systems Engineering and Management
2025, 10(2)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

SMS Spam Detection with Machine Learning Model & Classification

[1*]Dr. Prolay Ghosh, [2]Dr. Debashis Sanki, [3]Mr. Ritadrik Chowdhury, [4]Kaberi Dutta

[1]Assistant Professor, Department of Information Technology, JIS College of Engineering, Kalyani, Nadia, West Bengal, India. Email: prolay.ghosh@jiscollege.ac.in. https://orcid.org/0000-0001-9267-5766

[2]Assistant Professor, Department of Information Technology, JIS College of Engineering, Kalyani, Nadia, West Bengal, India. Email: debasish.sanki@jiscollege.ac.in

[3]Assistant Professor, Department of Information Technology, JIS College of Engineering, Kalyani, Nadia, West Bengal, India. Email: ritadrik.chowdhury@jiscollege.ac.in

[4]Department of Computer Application, JIS College of Engineering, Kalyani, Nadia, West Bengal, India. Email: duttakaberi4@gmail.com

**Corresponding Author: 1*Dr. Prolay Ghosh, Email: prolay.ghosh@jiscollege.ac.in*

ARTICLE INFO

Received: 30 Sept 2024

Revised: 29 Nov 2024

Accepted: 10 Dec 2024

ABSTRACT

Tool wear monitoring and predictive maintenance are critical in manufacturing, where traditional methods often struggle to adapt to changing conditions. This research presents an Adaptive Reinforcement Learning Framework for Real-Time Tool Wear Optimization and Predictive Maintenance (ARTOM). The framework integrates reinforcement learning with real-time sensor feedback to optimize machining parameters and maintenance schedules dynamically. Proximal Policy Optimization (PPO) is used to guide decision-making by balancing tool life, product quality, and operational costs. Multi-agent reinforcement learning divides tasks among agents to handle diverse machining scenarios, while sliding window techniques and dimensionality reduction ensure efficient data processing. The study has used the benchmark dataset, which include time-series sensor data and machining parameters. Metrics potential metrics have been used to evaluate prediction accuracy, while runtime and memory usage assess computational efficiency. Results has shown that ARTOM consistently achieves lower prediction errors and faster execution times than contemporary baseline models. These findings demonstrate ARTOM's ability to adapt to different tool conditions and improve operational decision-making.

Keywords: Tool Wear, Predictive Maintenance, Reinforcement Learning, Proximal Policy Optimization, Multi-Agent Learning, Sensor Data, Time-Series Analysis, Machining Optimization

INTRODUCTION

1) Problem Statement: SMS spam, involving fictitious messages without user consent, affects 68% of mobile users. Despite its importance in communication and business, it threatens its integrity. As SMS usage grows, spam filtering becomes crucial to protect against fraudulent activities. Accurate spam detection is essential for safeguarding human lives and maintaining text-based communication integrity.

2) Purpose of The Study and Motivation: The study aims to explore the increasing reliance on mobile devices for storing sensitive data and real-time communication, particularly through SMS. However, this has attracted spammers who exploit SMS for financial gain, posing a threat to users' privacy and security. The study emphasizes the need for advanced spam filtration methods to protect users from these threats.

3) Research Methodology: Word embedding methodologies like count vectorizer, TF-IDF, Hashing Vectorizer, Word2vec, and Glo Ve are used to classify SMS as spam or non-spam. However, these methods can lead to ambiguity and biases, and require significant data and computational resources. Contextual sentence embedding captures the overall meaning of sentences, aiding sentiment analysis and text classification. Combining these methods with models like naive, random forest, and K-nearest neighbor can improve SMS spam classification accuracy.

4) Research Scope: This study aims to develop an AI-driven model for SMS classification, integrating machine learning, deep learning, and natural language processing techniques. The goal is to distinguish between legitimate SMS messages and spam based on content and context. The research includes a thorough review of literature and analyzing a large dataset of SMS messages. The study focuses on technical aspects, excluding ethical and legal considerations, to improve spam prevention strategies.

II. LITERATURE REVIEW:

1) A study utilizing the term frequency-inverse document frequency (TF-IDF) method and the Random Forest Algorithm for SMS spam detection achieved an impressive 97.50% accuracy rate. The method includes pre-processing, feature extraction, selection, and classification. The Random Forest Algorithm demonstrated robust performance across various evaluation metrics. Future research could explore additional features and refinements to enhance spam detection classifiers.

2) This paper discusses the use of machine learning algorithms for spam detection in email content. It highlights the effectiveness of KNN in spam detection, highlighting the need for future research to test the model on diverse datasets and explore new techniques to improve accuracy and efficiency. Existing methods, such as linguistic-based, behaviour-based, and graph-based methods, face challenges in handling imbalanced datasets.

3) The paper discusses the problem of SMS spam, a growing concern due to mobile device usage, using machine learning and deep learning methods, including Naive Bayes, SVM, PCA, LSTM networks, and future advancements in neural networks.

4) Email communication is vulnerable to spam, affecting security and time. Despite machine learning algorithms like Naïve Bayes, decision trees, and neural networks, spam detection remains a significant issue. The Internet of Things and social networking platforms also face spam challenges. The paper explores machine-learning algorithms for spam detection, identifies research gaps, and suggests future directions like real-life data training and deep learning techniques.

5) The text introduces a novel SMS spam detection method using a discrete hidden Markov model (HMM), which improves accuracy compared to traditional bag-of-words models. The method's language-independence is validated on a Chinese SMS dataset, making it suitable for commercial applications, especially multilingual processing.

6) The study uses machine-learning algorithms to detect SMS spam. It uses a Kaggle dataset with 611 instances of spam and non-spam messages. The authors found that Random Forest and Random Tree algorithms achieved 100% accuracy. Other algorithms like J48, Rip, and LMT also showed high accuracy. The study suggests future work should incorporate feature reduction algorithms and stemming techniques to improve detection accuracy.

7) The author thanks Dr Majd, Gutta, and San Marcos for their support in completing a thesis on spam messages in SMS communication. The research methodology provides an overview of filtering techniques, and the results showcase machine learning and deep learning models.

8) SMS spam, a growing concern in electronic communication, is a significant threat due to its unique characteristics and challenges. Despite advances in detection techniques, spammers continue to adapt their tactics, necessitating ongoing research and innovation in spam filtering methods. Understanding SMS spam's characteristics and mechanisms is crucial for effective prevention and detection.

9) Dr Raga and Mrs. Chaitra B L conducted a study on SMS spam detection using machine learning and deep learning techniques. They used a dataset of 5,574 spam messages and found that long short-term memory networks outperformed traditional algorithms, achieving 98% accuracy.

10) Samadhan Nagre's study explores the use of machine learning techniques for detecting spam SMS messages. It reviews existing techniques, algorithms, and their applications in spam detection. The paper categorizes SMS spam filtering into white and black list approaches, content-based and non-content-based methods, and discusses the study selection procedure.

11) Review spam is a growing issue in online marketplaces, affecting consumer behaviour. Natural Language Processing and machine learning are promising methods to detect and mitigate spam, but lack of label data presents challenges. Feature engineering improves spam detection algorithms, but performance discrepancies can occur. Future research should explore unsupervised and semi-supervised learning methods.

12) The project addresses SMS spam issues by applying machine-learning techniques to a database of real spams. Naive Bayes outperforms the original model, reducing the overall error rate by over half. However, SVM with different kernels does not yield significant improvements.

13) The dissertation examines text classification methods to combat spam messages in SMS and email communication, comparing static embedding methods like Word2Vec and Glo Ve with dynamic embedding using BERT, and employs machine learning and deep learning techniques.

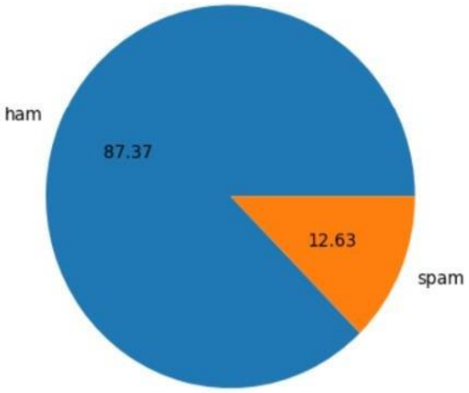
14) The study uses machine-learning techniques to tackle SMS spam, proposing a new Transformer model and a CNN-LSTM model for improved accuracy and performance, with future work focusing on additional features.

III. PROPOSED METHODOLOGY:

3.1 Data Collection:

The research examines the SMS Spam Collection Dataset, which includes 5,574 English messages classified as legitimate or spam. The dataset is sourced from various sources, including Grumble text, NUS SMS Corpus, Caroline Tag's PhD Thesis, and SMS Spam Corpus v.0.1 Big. The dataset is valuable for SMS spam research and academic studies.

3.2 Data Visulization:



Data visualization is crucial for detecting spam SMS, revealing imbalances between SPAM and HAM SMS articles, revealing textual feature patterns, and identifying indicators of spam SMS. Python and R tools offer insightful visualizations for effective detection strategies.

3.3 Data Preparation:

To prepare data for visualizations, standardize and normalize it by addressing quality issues, ensuring consistency, and correcting outliers or errors, streamlining the preparation process for further analysis and model development.

3.4 Data Augmentation:

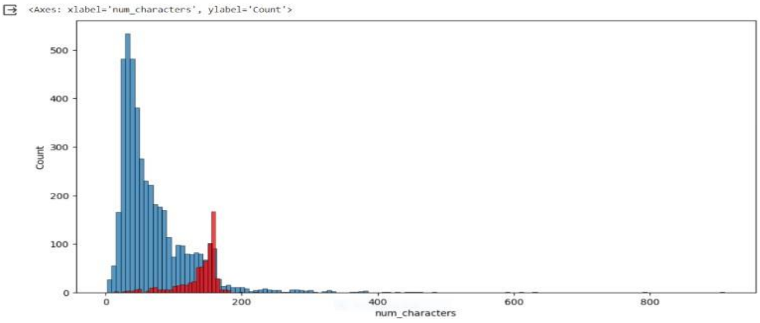
Data augmentation in SMS spam detection enhances training data by generating diverse samples through techniques like text transformation, synthetic generation, translation, and contextual augmentation.

3.5 Features Extraction:

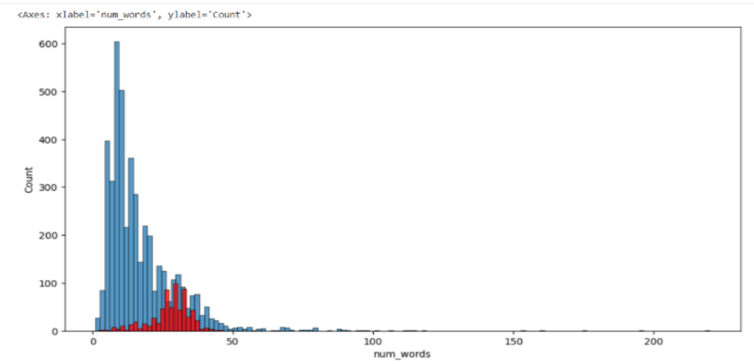
Python framework enhances spam message detection by extracting key features, improving detection rate and reducing processing time by investigating various spam message characteristics.

3.6 Exploratory Data Analysis (EDA):

Data science relies heavily on exploratory data analysis to better understand and contextualise data. It improves correlation with the business context. Open-source tools, such as Word Cloud, enable excellent visualisation and analysis of text data. Word The cloud symbolises text data, and the size of the text indicates its frequency or importance. Initially, data would be visualised using pie charts to identify spam and ham SMS and indicate imbalances. Next, word, sentence, and character counts will be compared for spam and ham SMS to identify discrepancies. Histograms will show them. Ham SMS often has higher values than spam. Correlation research reveals strong connections between spam and ham data parameters, including word count (0.68), character count (0.62), and sentence count (0.68), helping to identify dataset features.



```
<Axes: xlabel='num_characters', ylabel='Count'>
```



```
<Axes: xlabel='num_words', ylabel='Count'>
```

3.7 Data processing:

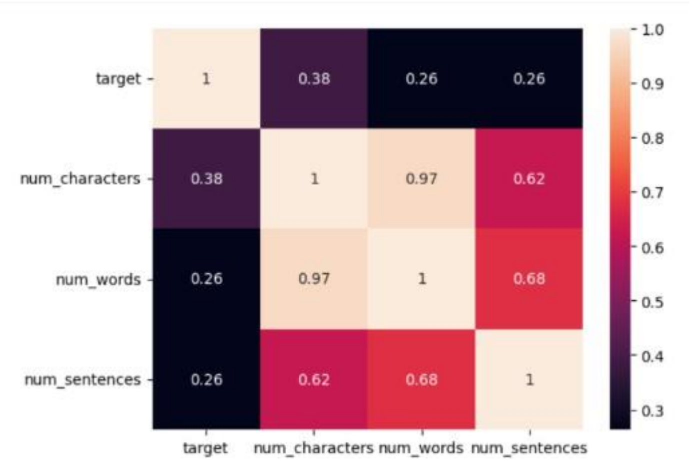
SMS spam detection requires multiple procedures to prepare text data for analysis.

1). Lowercase all text to ensure uniformity and delete duplication.

2). Tokenization is the process of breaking down text into distinct words for easier analysis.

3). Remove Special Characters: Remove non-alphanumeric characters, symbols, and punctuation marks that are irrelevant to the analysis.

4). Remove Stop Words and Punctuation: Remove frequent words and punctuation that are not useful for spam detection.



	target	num_characters	num_words	num_sentences
target	1	0.38	0.26	0.26
num_characters	0.38	1	0.97	0.62
num_words	0.26	0.97	1	0.68
num_sentences	0.26	0.62	0.68	1

5). Stemming: Reducing words to their root form enhances analytical accuracy. Text data can be transformed into numerical format for machine learning algorithms using techniques like bag-of-words or text vectorization. These techniques represent the frequency of each word in a dataset, such as in SMS messages. Combining data pre-processing steps with text vectorization techniques helps effectively detect spam messages. Pre-processing focuses on addressing symbols and stop words, which are common in SMS messages. High-frequency words dominate spam messages, while ham messages contain pronouns and verbs. Histograms highlight the disparity in word usage.

3.8 Library Used:

3.8.1 Pandas:

Pandas is a powerful data manipulation tool used in SMS spam detection, offering features like Data Frame, Series, data cleaning, transformation, and aggregation, and easy integration with other libraries.

- Ability to pivot and reshape datasets to make analysis easier.
- Sorting data into groups for transformations and aggregations to find trends.
- Managing absent information and ensuring datasets are consistent.
- Assistance with time series analysis to monitor temporal patterns and trends.

3.8.2 NumPy:

NumPy is used for SMS spam detection, offering efficient handling of large datasets, facilitating data preprocessing and analysis, and seamless integration with other libraries for comprehensive data analysis and visualization.

- Effective management of enormous volumes of data through the use of strong data structures like matrices and arrays.
- Easy support for data reshaping and matrix operations, which are necessary for text content analysis.
- Exceptional performance that allows for quick calculations on big datasets.
- Data manipulation is made easy using array-oriented computation.
- Multidimensional arrays optimised for scientific computations are implemented.
- Integrated data and Fourier Transform support.

3.8.3 NLTK:

NLTK is a natural language toolkit used for tasks like text tokenization, stopword removal, and stemming in SMS spam detection. It offers a comprehensive suite of libraries and tools for text analysis

3.8.4 Seaborn:

Seaborn is a statistical data visualization tool used for SMS spam detection, offering high-level abstractions, customizable plot types, and seamless integration with pandas Data Frames for efficient data analysis and visualization.

3.8.5 String:

The string module simplifies text manipulation tasks like removing special characters and punctuation marks in SMS spam detection. It offers built-in functions, flexibility, and efficient removal of unwanted characters, improving data quality and analysis accuracy.

3.8.6 Matplotlib:

Matplotlib is used for creating customizable plots and visualizations in SMS spam detection, offering a wide range of functions, fine-grained control over aesthetics, and support for interactive plotting in GUI tools.

3.8.7 Scikit-learn:

Scikit-learn is a crucial tool for building and evaluating machine learning models in SMS spam detection. It offers a wide range of algorithms for classification, feature selection, and model evaluation, simplifies model construction, and offers a user-friendly interface. Its extensive documentation and community support make it a preferred choice.

IV. MODELING:

4.1 Naïve Bayes (NB):

Naïve Bayes (NB) is a probabilistic classification technique based on Bayes' theorem, allowing for efficient classification of large, high-dimensional datasets. It assumes feature independence, treating all features equally and accurately scoring probabilities. NB is popular for text classification tasks, including spam detection, but its performance may degrade if features aren't independent

Gaussian NB:

- Usage in SMS Spam Detection: Gaussian NB is a Gaussian distribution technique used in SMS spam detection for continuous or real-valued features like word frequencies or character counts.

Multinomial NB:

- Multinomial NB is a widely used technique in SMS spam detection, focusing on counting data like word frequencies or document-term matrices. It models the probability of observing a word in a document, handling count data efficiently and sparsely.

Bernoulli NB:

- Bernoulli NB is a simple and efficient text classification algorithm used in SMS spam detection. It models the probability of observing a word as a binary outcome, assuming features are independent. Bernoulli NB performs well with sparse datasets and is less sensitive to irrelevant features.

4.2 Logistic Regression:

Logistic Regression is a binary classification algorithm that models the probability of spam in SMS messages using a sigmoid function, aiding tasks like credit risk assessment and fraud detection.

4.3 Support Vector Machine:

The reliable classifier Support Vector Machine (SVM) is frequently used in SMS spam detection. It ensures maximum margin between classes by employing a hyperplane to divide data points in n-dimensional space. SVM manages categorization by locating a distinct. Demarcation line between reputable SMS and spam. It builds a model that correctly classifies fresh messages to the right class through supervised learning. Text categorization and anomaly detection are two areas in which SVM is useful due to its adaptability and efficiency.

4.4 Decision Tree:

Decision Trees (DT) are simple, easy to understand models that work well for SMS spam identification. Based on characteristics like word frequency or the presence of particular keywords, they divide the data. A decision based on a feature is represented by each node, which helps identify if a message is spam or ham. Decision trees can handle both numerical and categorical data, but they are sensitive to even tiny changes in the data and are prone to overfitting. They have uses in a variety of industries, such as fraud detection and medical diagnosis, despite their drawbacks.

4.5 Random Forest:

Because Random Forest is an accurate and resilient algorithm, it may be used to detect SMS spam. It builds a set of decision trees, training them on arbitrary subsets of the feature set and data. Spam is effectively identified by Random Forest by using majority voting to aggregate the predictions of these trees. It is an effective tool for this endeavour due to its resistance to overfitting and capacity to handle high-dimensional data. Its interpretable outputs further increase its usefulness in spam detection applications by offering insights into the classification process.

4.6 AdaBoost:

AdaBoost's capacity to merge several weak classifiers into one strong one makes it useful in SMS spam identification. Here, weak classifiers could be only basic models that outperform chance by a little margin. AdaBoost focuses on hard-to- classify cases by giving misclassified samples a higher weight. AdaBoost efficiently learns to differentiate between messages that are spam and those that are not by iteratively training weak classifiers and modifying weights. Its capacity to improve the performance of weak models and adapt to unbalanced datasets make it a useful tool for raising the accuracy of SMS spam identification.

4.7 Bagging Classifier:

A machine learning ensemble technique called a "bagging classifier" combines several classifiers that were bootstrapped and trained on various subsets of the training data. Bagging Classifier can be used in SMS spam detection to increase the reliability and reduction of variance and overfitting in order to make the model more broad. The overall accuracy and dependability of spam detection are improved by Bagging Classifier, which aggregates predictions from several base classifiers. This method makes Bagging Classifier an effective tool for enhancing SMS spam detection systems' effectiveness, especially when working with noisy or imbalanced datasets.

4.8 Extra Trees Classifier:

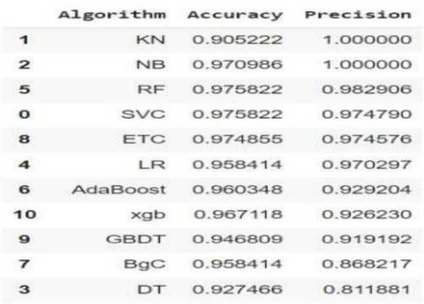
Extra Trees Classifier is an ensemble learning technique used for classification tasks that is comparable to Random Forest. Using random feature subsets and dataset subsets, it builds several decision trees. But in contrast to Random Forest, Extra Trees. An even more randomised decision tree is produced when the classifier chooses random thresholds for every feature. By decreasing overfitting and boosting resilience against noisy data, Extra Trees Classifier can improve model performance in SMS spam detection, improving the accuracy of identifying spam from non-spam texts.

4.9 Gradient Boosting:

A reliable machine learning approach used for regression and classification applications is gradient boosting. It creates a sequence of decision trees and uses gradient descent to minimise a number of weak learners into strong ones. Loss function. This approach has applications in picture identification, stock price forecasting, and customer churn prediction. It works well for handling complex and noisy information. It is a popular choice in Kaggle competitions because to its excellent performance and adaptability.

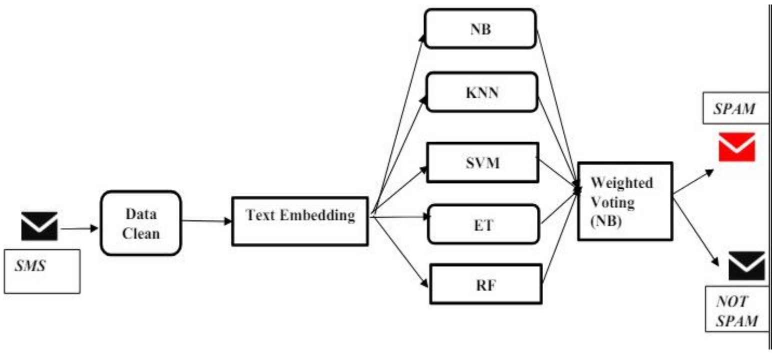
4.10 XGBoost:

X G Boost is a powerful and widely used ensemble learning method that can be applied to both regression and classification tasks. It excels at managing large and complex datasets by gradually building a network of decision trees using gradient boosting to curtail a loss function. Well-known for its proficiency in a wide range of machine learning tasks, XGBoost is a popular choice in many applications because of its ability to handle large datasets, provide accurate results, and prevent overfitting.



	Algorithm	Accuracy	Precision
1	KN	0.905222	1.000000
2	NB	0.970986	1.000000
5	RF	0.975822	0.982906
0	SVC	0.975822	0.974790
8	ETC	0.974855	0.974576
4	LR	0.958414	0.970297
6	AdaBoost	0.960348	0.929204
10	xgb	0.967118	0.926230
9	GBDT	0.946809	0.919192
7	BgC	0.958414	0.868217
3	DT	0.927466	0.811881

V. SYSTEM DESIGN FLOWCHART:



VI. RESULT ANALYSIS:

➢ Evaluation Metrics:

An essential tool for evaluating the effectiveness of classification models is the confusion matrix. It lists true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN), together with the actual and anticipated class labels. This matrix's metrics include .The results include F1 score, recall, accuracy, and precision. Recall analyses the capacity to identify positives, accuracy counts overall correctness, precision measures the genuine positive rate, and the F1 score strikes a balance between recall and precision.

The percentage of SMS messages that are accurately categorised as spam as well as legitimate is measured by accuracy. However, because of the imbalance between spam and real texts, accuracy in SMS spam detection can be deceiving. The following formula can be used to determine accuracy:

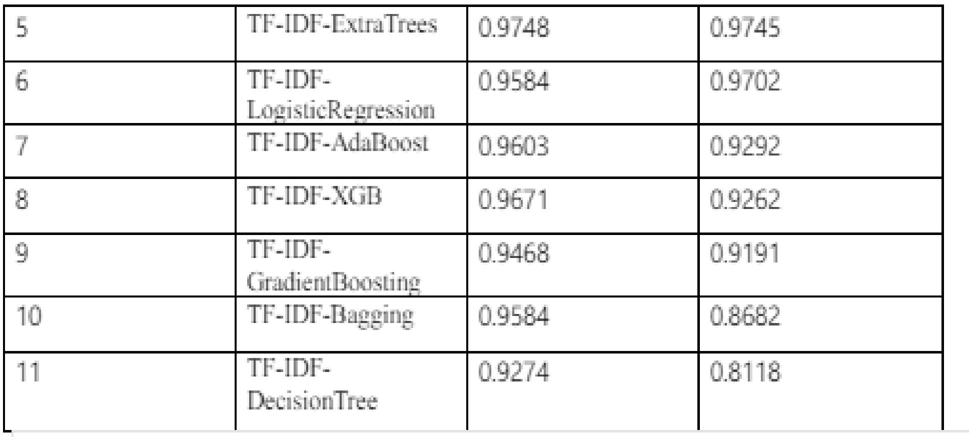
$$□□□□□□□□ = \frac{□□ + □□}{□□ + □□ + □□ + □□}$$

The percentage of SMS messages labelled as spam that are genuinely spam is measured by precision. The ratio of true positives, or spam messages that are accurately categorised as spam, to the total of true positives and false positives, or valid communications, is how it is computed. communications that were mistakenly categorised as spam. Precision is important in SMS spam detection since it shows how well spam is identified without mistakenly reporting valid messages.

$$□□□□□□□□□ = \frac{□□}{□□ + □□}$$

- **Machine Learning model:**

SERIAL NUMBER	ALGORITHM	ACCURACY	PRECISION
1	TF-IDF-NB	0.9709	1.0
2	TF-IDF-KNeighbors	0.9052	1.0
3	TF-IDF-RandomForest	0.9758	0.9829
4	TF-IDF-SVC	0.9758	0.9748



5	TF-IDF-ExtraTrees	0.9748	0.9745
6	TF-IDF-LogisticRegression	0.9584	0.9702
7	TF-IDF-AdaBoost	0.9603	0.9292
8	TF-IDF-XGB	0.9671	0.9262
9	TF-IDF-GradientBoosting	0.9468	0.9191
10	TF-IDF-Bagging	0.9584	0.8682
11	TF-IDF-DecisionTree	0.9274	0.8118

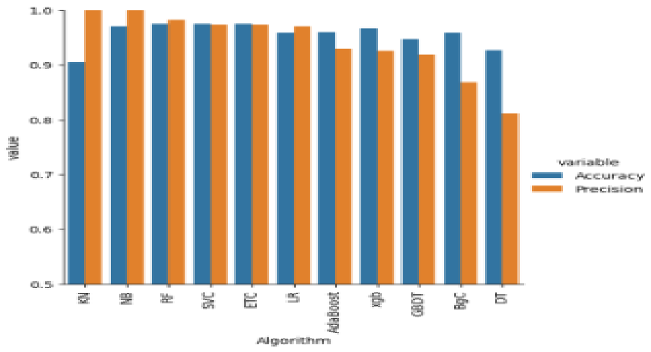


Fig 10:- Comparison of accuracy & precision of different model

VII. CONCLUSION:

Several supervised machine-learning methods were examined in the study for efficiently identifying spam SMS messages, a problem that is common in mobile communication around the world. After extensive testing, it was discovered that TF-IDF in combination with Compared to other methods, the MultinomialNB classifier produced results with higher accuracy. However, accuracy and precision were also important criteria to consider while evaluating the dataset because of its imbalance. The NB algorithm exhibited strong performance, achieving a precision score of 97%. This emphasises how crucial it is for spam detection programmes to take accuracy and precision into account.

Subsequent research directions may investigate the integration of extra features, including message durations, to improve classifier performance even more. The study also demonstrated the efficacy of content-linked features using NB and KNN classifiers. reaching absolute precision. This highlights how important feature engineering and selection are to the identification of SMS spam. All things considered, the results provide insightful information on how to fight SMS spam and highlight the necessity of complex assessment criteria to guarantee successful model operation. The paper offers a thorough examination of several machine learning techniques and suggests useful solutions for dealing with the enduring problem of spam in mobile communication networks.

REFERENCES:

[1] SMS Spam Message Detection using Term Frequency-Inverse Document Frequency and Random Forest Algorithm, Was represented by Nilam Nur Amir Sjarif, Nurulhuda Firdaus Mohd Azmi, Suriayati Chuprat, Haslina Md Sarkan, Yazriwati Yahya, Suriani Mohd Samvol. 16, no. 3, pp. 456–471, Mar. 2018.

[2] Email Spam Detection using Machine Learning Techniques, Was represented by Dr. Nilesh Jain, Dr. B. K. Sharma vol. 22, no. 4, pp. 789–802, Dec. 2018.

[3] "SMS Spam Detection Using Machine Learning and Deep Learning Techniques", Was represented by Pradeep K . Jan. 2019, pp. 115–123.

[4] Machine Learning Techniques for Spam Detection in Email and IoT Platforms: Analysis and Research Challenges. Was represented by Naeem Ahmed, Tariq Shah, Rashid Amin ,HamzaAldabbas,DeepikaKoundal,BaderAlouffi, vol. 30, no. 2, pp. 187–201, Feb. 2019.

[5] A Discrete Hidden Markov Model for SMS Spam Detectiontian was represented by Xia and XueminChen, May 2019, pp. 220–235.

[6] A Study of SMS Spam using Machine Learning was represented by Tayba Asgher. vol. 15, no. 3, pp. 310–325, Oct. 2019.

[7] SMS SPAM CLASSIFICATION USING MACHINE LEARNING wasrepresented by MANDAR SHIVAJI HANCHATE., Aug. 2020, pp. 310–325.

[8] SMS Spam Detection using Machine Learning Approach was represented by Abhishek Patel, Priya Jhariya, Sudalagunta Bharath, Ankita Wadhawan vol. 12, no. 4, pp. 765–778, Nov. 2020.

[9] Machine Learning and Deep Learning Techniques for SMS Spam Detection, Accuracy Check and Comparative Study was represented by Dr. Sarika Raga, Mrs Chaitra B , vol. 25, no. 4, pp. 789–802, Jan. 2021.

[10] Mobile SMS Spam Detection using Machine Learning Techniques Was represented by Samadhan Nagre vol. 22, no. 3, pp. 187–201, July 2022.

[11] Survey of review spam detection using machine learning techniques Was represented by Michael Crawford, Taghi M. Khoshgoftaar, Joseph D. Prusa, Aaron N. Richter and Hamzah Al Najada vol. 22, no. 4, pp. 789–802, Dec. 2022.

[12] SMS Spam Detection using Machine Learning Approach Was represented by Houshmand Shirani-Mehr vol. 15, no. 3, pp. 310–325, Oct. 2023.

[13] SPAM DETECTION USING MACHINE LEARNING AND DEEPLEARNING was represented by Unbounded Stephen Agboola May 2023, pp. 220–235.

[14] Spam Detection Using Machine Learning was represented by Supriya Yerakaraju, P Gopala Krishna, N V Ganapathi Raju vol. 25, no. 4, pp. 789–802, Jan. 2023.