**NORTH SOUTH UNIVERSITY**

# Department of Electrical and Computer Engineering



**Messaging web app with end-to-end encryption and spam detection using ML**

Al Kuresh Muna 2221987042

Mohammad Ishzaz Asif Rafid 2221370642

Sumaiyah Gazi 2131853042

## Instructor: Md. Ishan Arefin Hossain (IAH)

CSE 299: Junior Design

Section 19

Team Tensor

**Table of content**

Abstract 1

Introduction 2

Literature review 4

Motivation 6

Aim 7

Application of the project in real life 7

Tools 9

Projected plan 10

Projected timeline 10

Projected Cost 11

Conclusion 11

Contribution 12

References 13

**List of Figures :**

1. A Graphical figure of our Project 6
2. Block diagram of the functionality of our Web App 8

**List of Tables**

Aim 7

Tools 9

Timeline 10

**Abstract:**

In this modern world, for communication, we heavily rely on various international messaging apps ( Messenger, WhatsApp) which makes our data accessible to foreign companies. There are a lot of scammers and frauds outside who are always trying to steal our data and identity, intervene in our privacy, and eventually scam us. In this era of Artificial Intelligence, Our “**Messaging web app with end-to-end encryption and spam detection using ML**” will be beneficial in our country resulting in all our data being only accessible to our people and will help users to avoid spam messages automatically. Without even any issue, the AI will filter out all the spam messages from the inbox and give the users the best messaging experience without having any risk of being scammed, being blackmailed, or the manipulation of information. People can be free of the fear of getting cyber-harassment or attack and malware viruses from those spam messages.

***Keywords***—Machine Learning, Deep Learning

**Introduction:**

In the quickly evolving digital environment of today, messaging apps have become increasingly important for communication. As these platforms are used more often, worries about data security, privacy, and unsolicited spam have increased dramatically. Secure communication channels are desperately needed in communities all over the world, including Bangladesh, to shield consumers from online dangers including identity theft, hacking, and data breaches.

Like many other nations, Bangladesh has witnessed a rise in digital communication for both private and business needs. But because messaging apps are so widely used, people are more prone to privacy invasions because many well-known apps have weak encryption or are easily tricked by spammers or phishers. Users' safety and trust in digital platforms are at risk due to the prevalence of unsolicited messages, malicious information, and fraudulent schemes.

A **Messaging Web App** with **end-to-end encryption** and **spam detection using machine learning** addresses these challenges. End-to-end encryption guarantees the privacy and security of communication, while machine learning (ML)--powered spam detection screens out undesirable messages to protect users from potential threats. In Bangladesh, where digital literacy is still growing and consumers might not always be able to recognize and prevent cyber risks, such a solution is especially important. This initiative would promote a safer digital ecosystem for both individuals and organizations, while also improving security and confidence in online communication.

We will use the below machine learning models to have maximum accuracy in our project and detecting spams

**The Random Forest:** An ensemble technique called the Random Forest algorithm constructs several decision trees and then adds up their output to arrive at a final forecast. Random Forest can categorize communications as real or spam by analyzing a variety of factors, including the sender's behavior, the frequency of specific terms, and the message structure. It is perfect for spam filtering because of its durability and capacity to manage big datasets.

**K-Nearest Neighbor (KNN):** KNN is a straightforward and efficient model that uses similarities to prior labeled messages to classify messages. KNN will calculate the "distance" between a new message and a collection of known spam and non-spam messages in order to detect spam. The new message is marked if most of its nearest neighbors are spammers. Due to its high computational cost, KNN may perform poorly with large-scale data, but it does well with smaller datasets.

**Logistic Regression:** A common statistical model for binary classification problems, such as spam vs. non-spam, is logistic regression. It determines the likelihood that a message is spam based on particular characteristics (e.g., quantity of links, presence of specific keywords). For real-time spam detection, logistic regression is quick, easy to understand, and efficient; nevertheless, it can need feature engineering to catch subtleties in the message.

**Support Vector Machine (SVM):** SVM is a potent classification technique that determines the best hyperplane to divide messages into spam and non-spam categories. When the data is high-dimensional and non-linearly separable, SVM is especially helpful. SVM can reliably identify complex messages in a messaging app by examining their textual patterns, even if the messages change slightly from recognized spam.

**Deep Learning (Neural Networks):** For complicated text analysis, deep learning models like Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) work incredibly well. Through the comprehension of language patterns, context, and semantics, deep learning in the messaging app is able to automatically learn features from text data and identify spam messages. Deep learning needs a lot of processing power and big datasets, even though it is quite accurate.

**Decision Tree:** Decision Tree models divide the data according to decision rules, like the sender's reputation or the existence of specific phrases, in order to classify messages. A decision route that results in the classification of a message as spam or non-spam is represented by each branch of the tree. Decision trees are simple to understand and can be improved for higher accuracy by combining them with other models, such as Random Forest.

Each of these models can be trained on a labeled dataset comprising spam and non-spam communications in the spam detection module. A combination of these models can be used to continuously detect and prevent spam messages while enabling real communication to continue unhindered, depending on the apps unique requirements for instance speed, accuracy, dataset size.

**Literature review:**

In order to successfully filter out undesired or malicious content, spam detection in messaging platforms is a big task that involves a variety of machine learning and deep learning techniques. This overview of the literature covers current work on models for message spam detection and performance measures (F1-score, in particular).[1]

Improving Spam Message Classification and Detection with Transformer-Based Embedding and Ensemble Learning: In order to detect spam, this work combines ensemble learning with pre-trained Transformer models, such GPT-3. Several classifiers, including Decision Trees, SVM, and Logistic Regression, are part of the ensemble model. By combining these models, a reliable detection system with an F1-score of 0.9991 is produced. The remarkable outcome can be ascribed to the superior text embeddings furnished by the GPT-3 Transformer and the amalgamated potencies of several classifiers, which facilitate the intricate identification of spam messages in diverse circumstances (MDPI).[2]

Detection of Spam Worldwide Through Transfer Learning of BERT Model: The authors of this study employ BERT (Bidirectional Encoder Representations from Transformers) to detect spam in a variety of datasets, including messaging apps, email, and SMS. These varied datasets are used to refine the model and show how generalizable it is. With an accuracy of 97%, the universal spam detection model obtains an F1-score of 0.96. Because BERT can catch intricate linguistic nuances and contextual linkages in text, it is extremely effective in handling varied spam detection circumstances, as seen by this high F1-score (ar5iv).[3]

An Analysis of Machine Learning Algorithms in Comparison for SMS Spam Identification In order to detect SMS spam, this study analyzes several classical machine learning models, such as Random Forest, Naïve Bayes, and Support Vector Machines (SVM). SVM performed the best, obtaining an F1-score of 0.94. Using TF-IDF for feature extraction improved the model's capacity to distinguish between spam and valid communications, which contributed to its excellent score. To increase detection accuracy, the study emphasizes how crucial it is to choose the right feature extraction techniques and machine learning algorithms (Source: ResearchGate).[4]

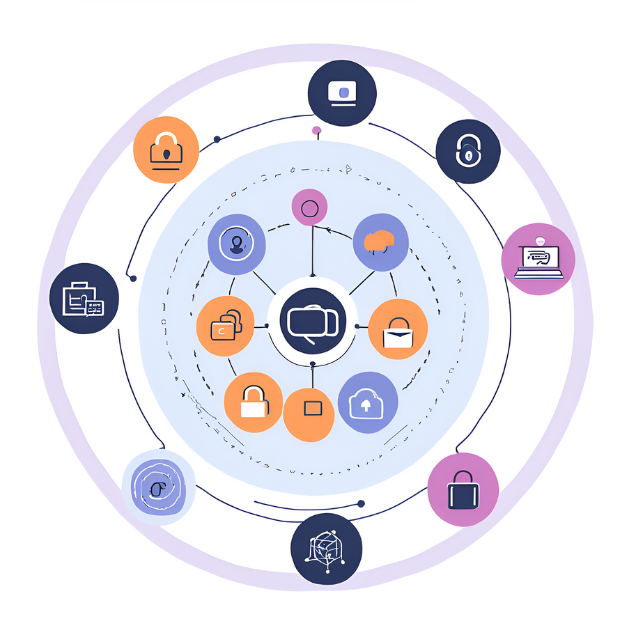
A comparative analysis of deep learning techniques for SMS spam detection The paper assesses various deep learning architectures for SMS spam detection, such as CNNs and LSTM networks. Having the ability to identify sequential dependencies in text data, the LSTM model obtained the highest F1-score of 0.97. This excellent performance was made possible by the dense and insightful feature representation of text data provided by word embeddings like Word2Vec and GloVe. According to IEEE Xplore, this illustrates how well deep learning models—in particular, RNNs—understand the sequential and contextual information contained in spam messages (Source: IEEE Xplore).[5]

Hybrid Deep Learning and Conventional Machine Learning Techniques for the Identification of Spam SMS: This study suggests a hybrid method that combines standard classifiers like SVM for classification with RNNs for feature extraction. With an F1-score of 0.95, the hybrid model outperforms the solo models. The model may take advantage of the advantages of both deep learning and conventional machine learning techniques, offering a comprehensive solution for spam identification. While the SVM classifier has strong classification capabilities, the RNN efficiently collects the text's sequential properties (Source: JETIR).[6]

**Motivation:**

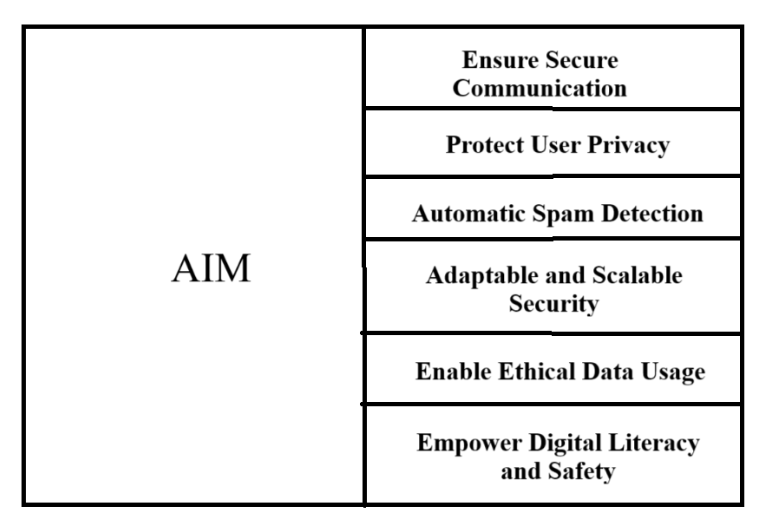
The digital revolution is speeding up in nations like Bangladesh, but digital literacy is still growing. Cybercriminals can easily target users in these locations since they may not always know the risks of spam or unencrypted messages. By offering a straightforward and user-friendly messaging platform that guarantees secure communication and shields users from fraud and unwelcome content, this initiative seeks to close the security gap.

Advances in machine learning and encryption technologies make it feasible to implement complex features like real-time spam detection and end-to-end encryption on a large scale. Leveraging these technologies allows the project to deliver a sophisticated, future-proof solution that addresses the evolving challenges of modern communication

****

**Fig 1: A Graphical figure of our Project**

**Aim:**



**Table 1: Aims of our Project**

**Application of the Project in Real life :**

Personal usage, through our web app, users can safely share text, image, or audio messages. Only the intended receiver will be able to access these confidential communications thanks to end-to-end encryption. Users are shielded from phishing scams and unsolicited promotions by the integrated spam detection feature, which also filters out any unwanted or hazardous messages

In Business, the web app can provide employees and clients with a secure platform where business discussions, contracts, or private documents can be shared safely.

In the medical department, the encrypted system guarantees that sensitive health data is protected, while spam detection prevents unwanted or fraudulent communication from reaching healthcare professionals or patients.

**Messaging WebApp**

**With Spam detection**

Message Receiving

(User Y)

Message Sending

(User X)

**Decryption**

**(Message Decrypted**)

**End-to-End Encryption**

**(Message Encrypted)**

**Encrypted Message is securely passing through Network**

**Machine Learning Model with Spam Detection**

**Message Received Successfully**

**(User B)**

Blocked

**Fig 2: Block diagram of the functionality of our Web App**

**Tools needed and short description:**

As we are building a web application implementing machine learning, we can divide our project into three-part, Frontend, Backend and Machine learning models.

|  |  |  |
| --- | --- | --- |
| Frontend | Backend | Machine Learning Model |
| Tools:  Language: HTML, CSS,  JavaScript  Framework: Bootstrap | Tools:  Language: Python  Framework: Django | Tools:  Language: Python  Framework/library: NumPy,  Pandas, Seaborn, Matplotlib,  PyTorch, Skit-learn |
| We will use HTML for the skeleton structure and CSS for the basic design. And bootstrap for responsive design. | Our web application will contain Django as the backend.    After building the machine learning model, we will connect those models with Django. | We will use Python as a programming language to build a machine-learning model.    We will use PyTorch to develop the Deep Learning model. |

Table 2: Tools

For version control, we will use GitHub And we will use the Django’s SQLite database.

**Project Plan:**

1. At beginning, we will collect data from Kaggle for spam detection emails and messages
2. Then. we will build our machine-learning system using different models(LR, SVM, KNN, RF etc)
3. After training our model with the labelled dataset, we will test the system.
4. We will compare the F1 scores (accuracy percentage) with each model. Whoever gives maximum accuracy we will use that model
5. Afterwards, we will work in our UI design and frontend part
6. Once the frontend will be done, we will start working on our backend part
7. Finally, connect it with the database

**Projected timeline:**

|  |  |  |
| --- | --- | --- |
| **Work** |  | **Approximate time** |
| Spam detection Model | 2 weeks |  |
| UI design | 1 weeks |  |
| Registration and log in and frontend | 3 weeks |  |
| Connect model with backend | 1 week |  |
| Report writing | 1 week |  |

Table 3: Timeline

**Projected Cost:**

Since we are not going to use any hardware tool, our project will be free of cost.

**Conclusion:**

A complete response to the growing worries about privacy and security in digital communication is provided by the creation of a messaging web application that uses machine learning for spam identification and end-to-end encryption. In a world where digital interactions are growing commonplace, especially in Bangladesh and other developing nations, this kind of technology is crucial to creating a more secure and reliable online environment. This project not only solves important user problems but also lays the groundwork for scalable, safe, and effective communication platforms by combining security with intelligent spam identification. In the end, this messaging software can provide users with a flawless and safe communication experience while drastically lowering the danger of cyber threats.

**Contribution:**

1. Al Kuresh Muna
   * UI design
   * Register and Log in
   * Frontend (HTML, CSS, BOOTSTRAP, JAVASCRIPT
   * Machine learning models

1. Mohammad Ishzaz Asif Rafid
   * Machine learning models
   * Connect model in the backend
   * Database management

1. Sumaiyah Gazi
   * Responsive design
   * UI Design
   * Frontend (HTML, CSS, BOOTSTRAP, JAVASCRIPT)
   * Prediction Model build

**References:**

V. S. Tida and S. Hsu, "Universal Spam Detection using Transfer Learning of BERT Model," arXiv preprint, vol. 2202.03480, Feb. 2022. [Online]. Available: arxiv.org.

J. Pérez and A. Zambrano, "Chat Analysis and Spam Detection of WhatsApp Using Machine Learning Algorithms," ResearchGate, May 2021. [Online]. Available: researchgate.net.

R. K. Sharma, "Spam SMS Classification Using Machine Learning," IEEE Xplore, vol. 8, pp. 134-141, Mar. 2020. [Online]. Available: ieeexplore.ieee.org.

S. Gupta and M. A. Khan, "Deep Learning Techniques for SMS Spam Detection: A Comparative Study," Journal of Emerging Trends in Computing and Information Sciences, vol. 11, no. 4, pp. 178-183, Apr. 2020. [Online]. Available: jetir.org.

A. K. Singh, "Spam SMS Detection Using Hybrid Techniques of Deep Learning and Traditional Machine Learning," International Journal of Engineering Research and Applications (IJERA), vol. 10, no. 2, pp. 54-60, Feb. 2021. [Online]. Available: ijera.com.[.](https://doi.org/10.3390/app13031346)