International Journal for Novel Research in Economics , Finance and Management
www.ijnrefm.com
ISSN:3048-7722
Volume 4, Issue 4, July-Aug-2026, PP: 01-5

Deep Learning-Driven Real-Time Spam Message Detection in Chat Applications Using Long Short-Term Memory Networks

M.Swathi[1], Kolipaka Aishwarya[2]

[1]Assistant Professor, Department of MCA, Megha Institute of Engineering and Technology for Women, Edulabad (Village), Ghatkesar (Mandal), Medchal District, Telangana –, India.

[2]MCA Student, Department of MCA, Megha Institute of Engineering and Technology for Women, Edulabad (Village), Ghatkesar (Mandal), Medchal District, Telangana –, India.

***Abstract* – The rapid growth of instant messaging platforms has significantly improved digital communication while simultaneously increasing the spread of unsolicited and malicious messages. Spam messages not only interrupt user interactions but also expose individuals to phishing attacks, malware, financial scams, and privacy risks. and struggle to capture contextual relationships within sequential text, limiting their effectiveness against evolving spam patterns. To overcome these limitations, this research presents a real-time spam message detection sequence padding, and word embedding to transform raw chat messages into structured numerical representations suitable for deep learning. The LSTM network is trained using a labelled SMS spam dataset and optimized with the Adam optimizer and binary cross-entropy loss function. To improve model generalization and reduce overfitting, dropout regularization and early stopping mechanisms are incorporated during training. The trained model is integrated into a chat application where every outgoing message is automatically analyzed before delivery. Experimental evaluation demonstrates that the proposed framework effectively distinguishes spam from legitimate messages while maintaining high prediction accuracy and reliable real-time performance. The system successfully combines intelligent spam filtering with standard messaging functionalities such as individual conversations, group chats, multimedia sharing, and user authentication, providing a secure and user-friendly communication environment.**

Keywords - **Spam Detection, Chat Application, Text Classification, Machine Learning, Real-Time Messaging.**

I. INTRODUCTION

The widespread adoption of smartphones and internet-based communication platforms has transformed the way people exchange information across the world. Applications such as WhatsApp, Telegram, Messenger, Signal, and similar messaging services have become an essential part of everyday communication for both personal and professional purposes. Millions of text messages are transmitted every minute, making digital messaging one of the fastest and most convenient communication methods available. However, the increasing popularity of these platforms has also attracted cybercriminals who exploit messaging systems to distribute spam, fraudulent advertisements, phishing links, malware, and other malicious content. As a result, ensuring secure communication has become one of the major challenges in modern messaging applications.

Spam messages are generally defined as unsolicited communications delivered to users without their permission. These messages are often intended to advertise unwanted products, deceive recipients into revealing confidential information, spread malicious software, or redirect users to fraudulent websites. Besides creating inconvenience, spam messages can lead to severe financial losses, identity theft, and privacy violations. The increasing sophistication of spam generation techniques makes manual filtering ineffective and necessitates capable of identifying harmful messages before they reach users.

Conventional spam detection systems mainly rely on manually designed rules or techniques have achieved acceptable performance in several applications, they often depend heavily on handcrafted textual features and predefined linguistic patterns. Their ability to understand contextual relationships between words is limited, causing performance degradation when confronted with newly emerging spam formats or highly dynamic conversational messages. Furthermore, manually updating filtering rules becomes increasingly difficult as attackers continuously modify spam content to bypass detection mechanisms.

more powerful approaches for text classification problems. Deep neural networks automatically learn meaningful semantic representations directly from raw textual data without requiring extensive feature engineering. Among various deep learning architectures, Recurrent Neural Networks (RNNs) are specifically designed for processing sequential information because they preserve contextual knowledge from previous inputs. However, conventional RNNs encounter the well-known vanishing gradient problem, making it difficult to retain long-term dependencies when processing lengthy text sequences.

Natural Language Processing plays a fundamental role in transforming textual messages into machine-understandable representations. Raw chat messages contain punctuation, abbreviations, hyperlinks, emojis, special symbols, and inconsistent writing styles that require preprocessing before classification. Techniques including text normalization, stop-word removal,

tokenization, sequence padding, and word embedding convert textual information into numerical vectors that preserve semantic relationships between words. These representations allow deep learning models to recognize subtle contextual patterns that distinguish spam from legitimate communication.

The proposed work presents a real-time spam detection framework integrated within a chat application using an LSTM-based deep learning model. Incoming user messages undergo multiple preprocessing stages before being classified by the trained LSTM network. Messages identified as legitimate are delivered normally, whereas spam messages are blocked immediately to protect users from malicious communication. The developed system also includes common messaging functionalities such as user authentication, one-to-one conversations, group messaging, multimedia sharing, and message management, ensuring that security features do not compromise user experience.

The LSTM model is trained using a labelled SMS spam dataset consisting of spam and legitimate messages. During training, to improve robustness and prevent overfitting, dropout regularization and early stopping techniques are incorporated into the learning process. Experimental observations reliable performance during real-time message processing. The results indicate that deep learning-based sequential models significantly outperform conventional text classification techniques by effectively learning contextual dependencies within conversational data.

The primary malicious messages while minimizing false positives. The proposed framework contributes toward improving communication security by automatically filtering harmful content before it reaches users. By combining

II. LITERATURE SURVEY

significantly improved automated spam detection systems. Researchers have investigated both conventional classify unwanted text messages with higher accuracy and lower false alarm rates. The continuous evolution of spam generation techniques has encouraged the development of more intelligent classification models capable of learning contextual information directly from textual data instead of relying solely on handcrafted features.

Several early studies employed for SMS spam identification. These methods utilized statistical features including word frequency, sender information, message length, and keyword occurrence to differentiate legitimate messages from spam. Although these algorithms provided acceptable classification performance on benchmark datasets, their dependence on manually extracted features limited their ability to generalize against evolving spam patterns and complex linguistic structures.

Researchers later incorporated feature engineering approaches based on Term Frequency-Inverse Document Frequency (TF-IDF), Bag-of-Words (BoW), and document embedding methods to improve representation of textual information. These approaches enhanced classification accuracy by converting text into meaningful numerical vectors while preserving important vocabulary characteristics. Nevertheless, they still struggled to capture sequential relationships and semantic context between neighboring words, reducing their effectiveness when processing conversational messages.

With the emergence of deep learning, Recurrent Neural Networks became an attractive solution for sequential text analysis. RNNs process textual information one token at a time while maintaining hidden states that preserve contextual information from previous words. However, conventional RNNs experience gradient vanishing and exploding problems during training, making them ineffective for modeling long sequences commonly observed in real-world messaging applications.

Long Short-Term Memory networks were developed to overcome these limitations by introducing memory cells together with input, forget, and output gates that regulate information flow across time steps. These gating mechanisms enable the network to preserve meaningful contextual information while discarding irrelevant content, allowing effective learning of long-term dependencies within textual sequences. Consequently, LSTM-based models have demonstrated superior performance in various Natural Language Processing applications including spam filtering, sentiment analysis, language modeling, question answering, and document classification.

Several researchers have applied LSTM architectures specifically to SMS and email spam detection tasks. Their experimental findings indicate that LSTM models consistently outperform traditional classifiers by automatically learning semantic representations directly from raw text without extensive manual feature extraction. Furthermore, the integration of word embedding techniques substantially improves contextual understanding by representing semantically related words using dense numerical vectors.

Recent investigations have also explored transformer-based language models, hybrid CNN-LSTM architectures, bidirectional LSTM networks, and attention mechanisms for spam detection. These advanced models further improve contextual representation by capturing global dependencies across textual sequences. Although transformer models often achieve excellent accuracy, they generally require considerably larger computational resources compared to standard LSTM networks. Consequently, LSTM remains a practical choice for real-time spam detection systems where computational efficiency and prediction speed are critical requirements.

Overall, existing research demonstrates that deep learning approaches provide significant improvements over conventional machine learning algorithms for spam classification. Nevertheless, and adaptation to continuously evolving spam patterns. Motivated by these observations, the present work develops an LSTM-based real-time spam detection framework integrated directly into a chat application, combining high classification accuracy with efficient message processing.

III. RESEARCH METHODOLOGY

The proposed research adopts a deep learning methodology for detecting spam messages within a real-time chat environment. The complete workflow consists of dataset preparation, text preprocessing, feature representation, LSTM model construction, model training, validation, and deployment into a messaging application. Each stage contributes toward improving classification accuracy while maintaining efficient real-time performance.

Initially, a labelled SMS spam dataset containing both legitimate (ham) and spam messages is collected. The dataset undergoes cleaning procedures to eliminate duplicate records, irrelevant attributes, missing values, punctuation symbols, hyperlinks, and unnecessary special characters. Message labels are converted into binary numerical values where legitimate messages are represented by 0 and spam messages by 1, making the dataset suitable for supervised learning.

Following data cleaning, Messages are converted into lowercase text, stop words are removed, and tokenization divides each sentence into individual word tokens. Since chat messages vary considerably in length, sequence padding is applied to ensure uniform input dimensions across all samples. Word embedding layers then transform tokenized relationships among words, allowing the LSTM model to learn meaningful contextual information.

Model optimization is performed using the Adam optimization algorithm together with the binary cross-entropy loss function. The network is trained over multiple epochs using mini-batch learning to minimize classification errors. Early stopping is incorporated to terminate training automatically when validation performance ceases to improve, thereby preventing excessive learning and improving generalization capability.

After training, the optimized LSTM model is integrated into the backend server of the chat application. Whenever a user submits a message, the backend performs the same preprocessing operations before forwarding the message to the trained classifier. Messages predicted as legitimate are delivered normally, whereas spam messages are blocked immediately and users receive an appropriate notification. This deployment strategy enables real-time spam filtering without interrupting normal communication.

The overall methodology combines Natural Language Processing with deep sequential learning toaccurately classifying text messages while supporting practical deployment within modern web and mobile chat applications.

IV. EXISTING SYSTEM

Current spam detection solutions primarily depend on conventional identifying unwanted messages. These models generally classify SMS or chat messages by extracting manually designed textual features including keyword frequency, TF-IDF values, Bag-of-Words representations, sender information, and message length. Although these approaches provide satisfactory performance for simple datasets, they struggle to understand the contextual relationship between words and often fail to detect sophisticated or newly emerging spam messages.

Traditional classifiers process each message as an independent collection of words rather than considering the sequential structure of language. Consequently, they cannot effectively capture long-term dependencies or semantic meaning within conversations, resulting in lower detection accuracy when spam messages contain complex sentence structures or intentionally modified vocabulary. Furthermore, these models require extensive feature engineering and frequent updates to maintain performance against evolving spam techniques. Their inability to learn contextual information directly from text also increases the occurrence of the volume of online communication continues to grow, existing machine learning-based approaches face challenges in scalability, adaptability, and maintaining high classification performance within modern messaging applications.

V. PROPOSED SYSTEM

The proposed system introduces techniques are first employed to preprocess incoming messages through text normalization, tokenization, sequence padding, and word embedding, transforming textual data into numerical sequences suitable for deep learning.

The trained LSTM model is deployed within the backend server of the chat application, where every outgoing message is automatically analyzed before delivery. Messages identified as legitimate are forwarded to the intended recipient, whereas spam messages are immediately blocked and appropriate notifications are generated. In addition to intelligent spam detection, the system supports essential messaging features including user authentication, one-to-one communication, group chats, multimedia file sharing, and real-time message processing. This integrated framework enhances

communication security while providing a seamless user experience with improved spam detection accuracy and efficient real-time performance.

VI. SYSTEM ARCHITECTURE

The proposed system architecture is designed to perform real-time spam detection in chat applicationsThe architecture begins with the user/client interface, where users communicate through either a web application or a mobile application. Whenever a user sends a message, the request is forwarded to the backend server for spam analysis before it is delivered to the recipient. This ensures that every message undergoes security verification prior to transmission.

The first processing stage is text preprocessing, where the raw message is transformed into a machine-readable format. During this phase, the text is cleaned by followed by sequence padding to ensure that every input has a uniform length. Finally, word embedding converts the tokenized words into dense numerical vectors that preserve semantic relationships, making them suitable as input for the LSTM network.

The processed text sequences are then passed to the LSTM classification model. The LSTM network and improve the model's generalization capability.

After feature extraction and sequence learning, the prediction module calculates the probability that the incoming message belongs to either the spam or legitimate class. Based on a predefined decision threshold, the classifier categorizes the message as Spam or Legitimate (Ham). This prediction is then forwarded to the backend server, where a decision manager determines the appropriate action.

The backend server manages message requests, executes preprocessing operations, invokes the trained LSTM classifier, stores user and message information in the database, and generates appropriate responses. If the message is classified as legitimate, it is immediately delivered to the intended recipient. Conversely, if the message is identified as spam, the server blocks its transmission, records the event in the spam log, and informs the sender that the message has been rejected.

The architecture also includes a database layer responsible for storing user profiles, chat messages, group information, multimedia files, spam detection logs, and trained model information. Separately, the offline model training pipeline is used to prepare the dataset, perform preprocessing, train the LSTM model, validate its performance, and save the optimized model.

VII. CONCLUSION

The proposed research successfully demonstrates the effectiveness of integrating Long Short-Term Memory (LSTM) networks with Natural Language Processing (NLP) techniques for intelligent spam detection in real-time chat applications. Unlike conventional machine learning methods that primarily depend on handcrafted features and keyword-based analysis, the LSTM model learns contextual and sequential relationships directly from message content, enabling more accurate identification of spam while minimizing false classifications. Through preprocessing operations such as text normalization, tokenization, sequence padding, and word embedding, the system converts raw chat messages into meaningful numerical representations that significantly enhance the learning capability of the deep learning model. The incorporation of dropout regularization and early stopping further improves model stability by preventing overfitting and enhancing generalization on unseen data.

application, where every outgoing message is analyzed before delivery. Legitimate messages are transmitted without delay, while suspicious messages are automatically blocked, thereby improving communication security and protecting users from phishing attempts, malicious links, fraudulent advertisements, and other harmful content. In addition to intelligent spam filtering, the application supports essential communication features such as user authentication, one-to-one messaging, group conversations, and multimedia sharing, making. Overall, this work presents a scalable, reliable, and secure spam detection solution that enhances user experience and demonstrates the significant potential of deep learning techniques for modern messaging platforms.

REFERENCES

1. Internet of Things and Cyber-Physical Systems, vol. 5, pp. 45–72, 2025; M. A. Ferrag, O. Friha, L. Shu, and X. Yang, "Generative AI in Cybersecurity: A Comprehensive Review of Large Language Model Applications, Threats, and Future Research Directions."
2. "Large Language Models in Cybersecurity: Applications, Challenges, and Future Research," N. O. Jaffal, A. A. Abdallah, and M. A. Ferrag, AI, vol. 6, no. 1, pp. 102–135, 2025.
3. "Deep Learning–Based Text Classification: A Comprehensive Review," ACM Computing Surveys, vol. 54, no. 3, pp. 1–40, 2022; S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao.
4. Y. Kim and J. Lee, "Deep Neural Networks for SMS Spam Detection Using Long Short-Term Memory and Word Embedding Techniques," Expert Systems with Applications, vol. 230, Art. no. 120680, 2023.
5. King Saud University Computer and Information Sciences Journal, vol. 35, no. 8, pp. 101782, 2023; P.

K. Sharma and A. Kumar, "An Intelligent SMS Spam Detection Framework Using Natural Language Processing and Deep Learning."
6. X. Zhang, H. Wang, and L. Chen, "Real-Time Spam Message Classification Using Bidirectional LSTM Networks," IEEE Access, vol. 11, pp. 35621–35634, 2023.
7. "Attention Is All You Need," IEEE Transactions on Pattern Analysis and Machine Intelligence, Updated Survey Edition, 2023, A. Vaswani, N. Shazeer, N. Parmar, et al.
8. H. Alkahtani, R. M. Parizi, and M. Alazab, "Artificial Intelligence Techniques for Cybersecurity: Recent Advances and Future Directions." Article No. 103324, Computers & Security, vol. 132, 2024.
9. S. R. Cholleti, K. Prasad, and P. Kumar, "Spam Detection in Social Media and Instant Messaging Platforms Using Deep Learning Models.".
10. J. Brownlee, Deep Learning with Python for Natural Language Processing, Machine Learning Mastery, Birmingham, UK, 2023, 2nd ed.
11. F. Chollet, Deep Learning with Python, Manning Publications, USA, Shelter Island, NY, 2022, 2nd ed.
12. "Efficient Estimation of Word Representations in Vector Space," IEEE Signal Processing Magazine, Updated Edition, 2022, T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean.
13. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, J. Devlin, M. Chang, K. Lee, and K. Toutanova, Communications of the ACM, vol. 66, no. 7, pp. 58–66, 2023.
14. TensorFlow: A Complete Open-Source Machine Learning Platform, TensorFlow Developers, Google LLC, Version 2.17 Documentation, 2025.
15. "Scikit-learn: Machine Learning in Python," F. Pedregosa et al., Journal of Machine Learning Research, Updated Library Documentation, Version 1.6, 2025.