IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 10 October 2023, accepted 18 October 2023, date of publication 26 October 2023, date of current version 8 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3327897

RESEARCH ARTICLE

Multi-Type Feature Extraction and Early Fusion Framework for SMS Spam Detection

HUSSEIN ALAA AL-KABBI[1], MOHAMMAD-REZA FEIZI-DERAKHSHI[1], AND SAEID PASHAZADEH[2]

[1]Computerized Intelligence Systems Laboratory, Department of Computer Engineering, University of Tabriz, Tabriz 5166616471, Iran
[2]Department of Computer Engineering, University of Tabriz, Tabriz 5166616471, Iran

Corresponding author: Mohammad-Reza Feizi-Derakhshi (mfeizi@tabrizu.ac.ir)

ABSTRACT SMS spam is a pervasive issue that affects millions worldwide, leading to significant inconvenience, time wastage, and potential financial scams. Given the prevalence and potential harm, accurate and real-time detection of SMS spam is crucial. This paper proposes a novel approach to SMS spam detection involving five steps: preprocessing, feature extraction, feature fusion, feature selection, and classification. Our model is designed to simultaneously capture local, temporal, and global text message features using a hybrid deep learning model to enhance feature representation. We evaluated our model using the UCI dataset, comparing it with traditional and deep learning algorithms such as RF and BERT using cross-validation to ensure the robustness of our results. Our proposed method exhibited superior performance, achieving a good accuracy of 99.56%, surpassing other methods. The effectiveness of this method in SMS spam detection proved its potential for real-world implementation, where it could substantially mitigate the prevalence and impact of SMS spam.

INDEX TERMS CNN, data fusion, deep learning, LSTM, SMS spam detection.

I. INTRODUCTION

With the rise of mobile phones and networks, Short Message Service (SMS) has become a popular method of communication. According to Portio Research, the world sent 16 million SMS messages per minute, 23 billion per day, and 8.3 trillion SMS messages in 2017 [1]. However, SMS users are also vulnerable to SMS spam or irrelevant electronic messages sent through mobile networks. Several factors contribute to the prevalence of spam, including the large number of mobile phone users and the low cost of sending spam messages [2]. Most mobile phone classifiers are weak in recognizing spam messages as they lack computing resources.

The study identifies three primary types of SMS spam: (i) SMS spam, involving unsolicited texts for mass advertising and viral hoaxes, (ii) premium rate fraud, which tricks subscribers into calling premium rate numbers or signing up for costly subscription services, and (iii) phishing/smishing, where unsolicited texts ask subscribers to call specific numbers to extract confidential information for malicious purposes [3].

Deep learning is one of the most rapidly advancing technologies, driving modern artificial intelligence (AI) progress. Substantial improvements in technology and algorithms in recent years have ushered in a new era of AI applications [4]. Deep learning algorithms have demonstrated performance levels that match or even surpass human accuracy across various applications, including text, sound, and image classification, image classification [5], [6], and Anomaly Detection [7]. Moreover, they employ multiple hidden layers in neural networks [8], which perform computing tasks similar to the functions of biological neurons in the human brain. Recurrent Neural Networks (RNN) and their derivatives, including Long Short-Term Memory (LSTM), have demonstrated significant potential in identifying spam [9]. Convolutional Neural Networks (CNN) can also effectively detect SMS spam and other communication media, including emails, social network systems, and online reviews [10]. Also used in networks and cyber security applications [11].

The associate editor coordinating the review of this manuscript and approving it for publication was Donato Impedovo.

Numerous methods have been developed to combat spam SMS messages. However, these efforts have not yet achieved complete spam detection or eliminated the misclassification of important messages as spam. Most previous methods have attempted to improve classification results by modifying the classifier rather than focusing on the crucial aspect of feature extraction from texts.

This paper presents a novel hybrid approach to detect SMS spam, combining deep learning with traditional methods for feature extraction. The proposed method utilizes CNN, LSTM, and TF-IDF techniques simultaneously to capture local, temporal, and global features, respectively. This approach offers a comprehensive solution for effectively addressing SMS spam.

The CNN and LSTM methods are deep learning models that capture semantic information by modeling local and temporal dependencies. In contrast, the TF-IDF method, a traditional technique, captures global features by calculating the importance of each term in the text based on its frequency and inverse document frequency.

The proposed method employs early data fusion to combine the features extracted by the CNN, LSTM, and TF-IDF techniques. Early data fusion integrates the features at an initial stage of the classification process, which helps reduce overfitting by incorporating diverse and complementary features from multiple methods [12]. Furthermore, it can enhance classification performance by providing a more robust feature representation.

The proposed method uses an attention mechanism as a feature selection method to select relevant and important features for SMS spam detection [13]. The paper's main contributions can be outlined as follows:

1) The proposed method integrates multiple features, namely local, temporal, and global, by combining CNN, LSTM, and TF-IDF techniques for feature extraction. Semantic features are extracted by leveraging CNN and LSTM, while TF-IDF is utilized to capture statistical features. This holistic approach results in a comprehensive and efficient representation of features from the input text.
2) The proposed method demonstrates superior accuracy and performance compared to other modern SMS spam detection techniques, making it a promising solution for SMS spam detection.

Section II reviews previous studies on SMS spam filtering, while Section III provides the theoretical background. Section IV describes our proposed method, followed by the evaluation process in Section V. Section VI presents our method's results and performance analysis. Finally, in the last section, we provide the paper's conclusion.

II. RELATED WORKS

Several research papers have proposed different methods for detecting SMS spam using artificial intelligence and Natural Language Processing (NLP) techniques. Traditional machine learning algorithms like Support Vector Machine (SVM) [14], Logistic Regression (LR), Random Forest (RF) [15], Naïve Bayes (NB), and Decision Trees (DT) have been widely employed in SMS spam detection. However, the emergence of deep learning has led to new approaches that utilize deep learning models such as CNN, RNN, and LSTM for enhanced SMS spam detection.

A. TRADITIONAL METHODS

These methods include rule-based filters, such as SVM, naive Bayes classifiers, and DT. However, these methods often require extensive feature engineering and have limited performance on noisy or imbalanced datasets [16].

Sjarif et al. [15] introduced a new method for SMS spam filtering based on the TF-IDF and Random Forest (RF) Algorithm. The experimental tests were performed using the SMS Spam Collection v.1 dataset [17]. The results of the dataset analysis outperformed and achieved an accuracy of 97.5%.

Xia et al. [18] introduced a new approach for SMS spam filtering that leverages the hidden Markov model (HMM) to incorporate word order information and overcome the limitations of the common term frequency problem. The proposed method achieved an accuracy rate of 98.5%.

Ghatasheh et al. [19] proposed a modified genetic algorithm to simultaneously address dimensionality reduction and hyperparameter optimization in datasets. This approach initialized an extreme Gradient Boosting (XGBoost) classifier and reduced the feature space of the dataset, resulting in a spam prediction model with 99.1% accuracy. Other hybrids mothed by Ubale et al. [20] propose an approach that analyzes message content and extracts features using TF-IDF techniques to differentiate between ham and spam messages. Employing a Voting classifier further enhances the accuracy of spam detection.

B. DEEP LEARNING METHODS

Using deep learning models for SMS filtering has shown promising results in recent research. By leveraging the power of neural networks, these models can learn complex patterns in SMS messages and effectively classify them as spam or ham. They often outperform traditional machine learning algorithms and require less feature engineering [21].

Liu et al. [22] presented a modified transformer approach to filter SMS spam messages. The transformer employs a multi-head attention mechanism, unlike traditional RNN variants used as encoders and decoders. This approach reduces training costs through parallelization and enhances performance in translation tasks. However, in some instances, unknown words negatively impacted the model, leading to false predictions. Despite this, the proposed approach achieved an accuracy of 98.92% for spam detection.

Al Bataineh et al. [23] proposed an approach for enhancing text classification, including SMS spam classification and sentiment analysis, using LSTM topologies and the clonal

selection algorithm (CSA). The results indicated that this approach achieved an accuracy rate of 98.48%, highlighting its potential for effectively addressing the SMS spam problem and other text classification tasks.

Ghourabi et al. [24] presented a deep learning approach for filtering SMS spam messages in their paper. Their model integrates two deep learning methods, LSTM and CNN, to effectively handle text messages in Arabic or English. The proposed method attained an accuracy rate of 98.37%.

Rahman et al. [25] introduced a hybrid method that combines CNN and Bidirectional Long Short-Term Memory (Bi-LSTM) networks for SMS spam detection, addressing the limitations of using either method alone. The proposed model achieved 98-99% accuracy on the UCI SMS spam collection dataset.

Debnath et al. [26] proposed deep learning methods, LSTM and BERT, to detect SMS spam, achieving high accuracy rates of 98.84% and 99.28%, respectively. Results are compared with previous models using the UCI dataset.

Srinivasarao et al [27]. Proposed FRNN-HHO architecture performs post-classification to improve classification accuracy. The performance is evaluated using various metrics and three datasets, with accuracy values of 98.1% (SMS), 95.8% (Email), and 95% (spam assassin).

Many methods for detecting spam messages have been presented with good results. However, they were not accurate enough to ensure the safety of user information, preserve their privacy, and reduce annoyance and time waste. Therefore, we have presented a more accurate approach to detecting SMS spam messages. Other papers dealt with SMS Spam messages using local databases in different languages, such as [28] or databases containing spam images and texts like [29].

III. BACKGROUND

In this section, we provide a succinct overview of the theoretical principles that underpin the proposed method.

A. CONVOLUTIONAL NEURAL NETWORKS (CNN)

CNN is a deep learning algorithm commonly used in image and text classification tasks. In text classification, CNN can be used for local text feature extraction, capturing important phrases or patterns within the text data [30]. One common approach for local text feature extraction is to use n-grams, which are contiguous sequences of n words within a text document. CNN can be trained to extract features from these n-grams, providing a way to capture local context within the text. This approach has been used in various NLP tasks, such as text classification, spam detection, and fake news detection [31].

B. LONG SHORT-TERM MEMORY (LSTM)

LSTM, a recurrent neural network (RNN), is highly effective in NLP tasks because it captures temporal dependencies in sequential data [32]. It processes text data sequentially, updating its internal state at each time step. This enables LSTM to capture long-term dependencies and extract features like sentiment or topic information. When combined with techniques like convolutional layers, LSTM becomes a powerful tool for text feature extraction, enhancing accuracy in downstream NLP tasks. LSTM achieves this by utilizing specialized gates (input, forget, and output gates) to selectively retain relevant information, discard irrelevant data, and control information flow throughout sequential processing [9].

C. TERM FREQUENCY-INVERSE DOCUMENT FREQUENCY (TF-IDF)

TF-IDF is a widely used technique for text feature extraction, identifying essential words or phrases across a text corpus [33]. It computes a weight for each word based on its frequency and importance in distinguishing documents. Words common across the corpus but not in all documents receive higher weights, signifying important global features [34]. For tasks like text classification or topic modeling, TF-IDF can be combined with dimensionality reduction or other methods to enhance feature quality. This technique captures vital word distribution and unique information, improving performance in downstream NLP tasks.

D. DATA FUSION

Data fusion combines multiple data sources to enhance feature quality and accuracy [35]. Feature extraction integrates various techniques or data modalities to capture different aspects of the input data, which is helpful in complex domains like NLP [12]. Combining textual and visual features can improve tasks like image captioning [36]. Fusing features from different deep learning methods can enhance text representation for downstream NLP tasks, such as sentiment analysis and text classification [37]. According to [35], various data fusion techniques exist, including early fusion, decision fusion, score fusion, and hybrid fusion. Early fusion combines features from different sources to create a comprehensive feature set. Decision fusion combines decisions from multiple algorithms or systems to generate a unified decision. Score fusion combines scores from multiple techniques to produce an overall score. Hybrid fusion combines different fusion methods, such as feature-level and decision-level fusion, to create a comprehensive and robust solution.

Early data fusion combines extracted features for SMS spam detection, creating a comprehensive and robust data representation to enhance algorithm accuracy.

E. ATTENTION MECHANISM

The attention mechanism, widely employed in deep learning, machine learning, and NLP, enhances feature extraction by focusing on the most relevant parts of the input data [38]. In text analysis, attention identifies crucial words or phrases and assigns weights based on their relevance. This enables

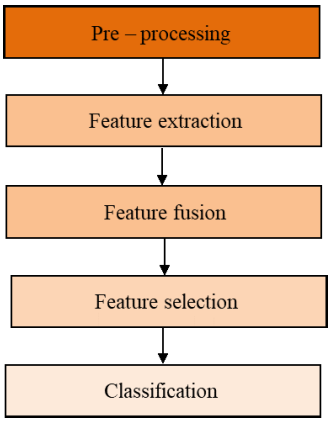

FIGURE 1. Proposed method steps.

the model to prioritize informative text elements while disregarding noise or irrelevant information, thereby improving the performance of downstream deep and machine-learning tasks.

IV. PROPOSED METHOD

The proposed method consists of five steps: preprocessing, feature extraction, feature fusion, feature selection, and classification, as depicted in Figure 1. Our main objective is to accurately classify incoming SMS messages into two categories: spam (including advertisements and fraudulent content) and not spam.

A. PREPROCESSING

Data preprocessing is a critical initial step in our SMS spam detection approach, aimed at preparing the raw text messages for effective analysis by machine learning and deep learning models. We perform the following preprocessing actions to enhance the data quality [24]:

1) Punctuation Removal: We eliminate unnecessary punctuation marks and symbols from text messages. Punctuation may not carry significant discriminatory information for distinguishing between spam and legitimate messages. By removing punctuation, we focus on the content and meaning of the messages, simplifying the data representation and reducing the vocabulary size [39].
2) Lowercasing: To ensure uniformity and consistency in our data, we convert all words in the text to lowercase. This step avoids any discrepancies due to variations in capitalization and ensures that identical words are treated similarly, contributing to a standardized vocabulary.
3) Stop-word removal: Common words like "the," "is," and "a" are known as stop words, and they occur frequently in language but often do not add substantial meaning to the text. We remove these stop words from the text messages to reduce the dimensionality of the data and prevent them from affecting the spam detection process. This allows us to focus on the words that carry more discriminatory power [23].
4) Tokenization: Following preprocessing, we tokenize each message into individual words or tokens. Tokenization breaks down the text into discrete units, enabling us to analyze the text at the word level [40]. This step is crucial for the subsequent bag-of-words representation, where the frequency of each token is used to create feature vectors for modeling.
5) Bag-of-Words Representation: We adopt the widely used bag-of-words representation technique from natural language processing (NLP) [38]. Each text message is represented as a "bag" or "multi-set" of its constituent words, discarding grammar and sequence information. This representation captures the occurrence patterns of words in the text, providing valuable features for our SMS spam detection model.

By meticulously applying these preprocessing steps, we ensure that the text data is in an appropriate format for machine learning and deep learning models. The resulting numerical representations facilitate efficient analysis and contribute to the success of our proposed SMS spam detection approach.

We believe that these preprocessing steps play a vital role in enhancing the data quality and contributing to the reproducibility of our research. Researchers can replicate our experiments by following the same preprocessing steps, which enables validation and comparison of our findings.

B. FEATURE EXTRACTION

This section presents a novel simultaneous feature extraction model that combines three approaches: CNN, LSTM, and TF-IDF. Unlike other models that rely on only one approach for feature extraction [15], our model aims to create a comprehensive integration of features, as depicted in Figure 2. This approach aims to capture local, temporal, and global features from SMS messages. By leveraging multiple approaches, we can extract various features, ultimately leading to a more precise classification of SMS messages.

1) LOCAL FEATURE EXTRACTION

The main objective of using a CNN in text classification is to capture and extract local features within the text. This is achieved by applying convolution on the word vectors generated by the preceding word embedding layer. The convolution operation discussed in this section is based on [24].

To explain the CNN process, we represent each word in a message as a d-dimensional vector. These word vectors are arranged to construct the input message. Window vectors, which consist of sequential word vectors, are used to extract features from the message. A filter is convolved with these window vectors, resulting in a feature map. Each element of the feature map is calculated using a nonlinear function. In our CNN model, we use the Rectified Linear Unit (ReLU)

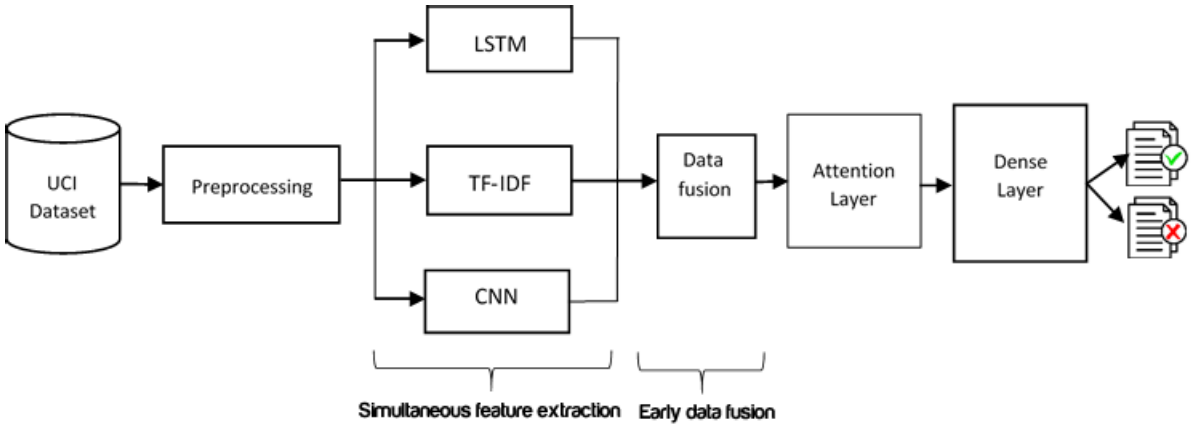

FIGURE 2. The proposed method architecture.

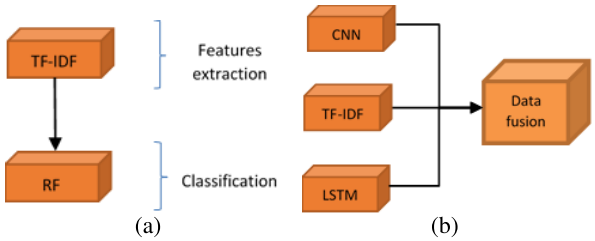

FIGURE 3. a- traditional feature extraction [15], b - simultaneous feature extraction.

as the nonlinear function. The convolution layer employs a one-dimensional convolution with a specific filter window size.

2) TEMPORAL FEATURE EXTRACTION

In our model, LSTM is employed to address the limitation of CNN in capturing temporal features. As mentioned earlier, LSTM is capable of learning long-term dependencies and is composed of repeated units for each time step. Each unit consists of a cell and three gates (input, output, and forget) that control the information flow within the LSTM unit. These gates determine the updating of the memory cell, and the transition functions between the LSTM units are defined in the literature [24]. The LSTM layer in our model comprises 64 units and incorporates a dropout rate of 0.2 to mitigate overfitting.

3) GLOBAL FEATURE EXTRACTION

We utilize TF-IDF to extract global features in our SMS spam detection approaches. The TF-IDF value of a word in a message is calculated by multiplying its Term Frequency (TF) with its Inverse Document Frequency (IDF). TF measures the word frequency within the message, while IDF evaluates its rarity across the entire SMS corpus. More specifically, IDF is computed as the logarithm of the ratio between the total number of SMS messages in the corpus and the number of messages that contain the word.

Feature extraction plays a crucial role in our SMS spam detection approach, employing three simultaneous methods: CNN, LSTM, and TF-IDF, to capture various aspects of the input data and enhance the accuracy of our model. CNNs are utilized to extract local features, identifying important patterns within the text. LSTM is employed to capture temporal relationships between words or phrases, enabling the learning of long-term dependencies. TF-IDF focuses on global features, evaluating the significance of each term across the SMS corpus. By combining these features, our approach achieves improved accuracy in SMS detection, effectively classifying spam and non-spam messages.

C. FEATURE FUSION

We employ the early data fusion approach to merge the multi-type extracted features from CNN, LSTM, and TF-IDF in the feature fusion step. The feature vectors obtained from each method are combined to create a more comprehensive and robust representation of the SMS data [41]. CNN, LSTM, and TF-IDF features can be represented as

$$F_{CNN} \in R^{m \times C} \tag{1}$$

$$F_{LSTM} \in R^{m \times l} \tag{2}$$

$$F_{TFIDF} \in R^{m \times t} \tag{3}$$

where m is the number of samples. Our proposed early data fusion strategy concatenates these feature vectors, resulting in a concatenated matrix

$$F_{concat} \in R^{m \times (C+l+t)} \tag{4}$$

In this section, data fusion aims to fuse complementary information from different feature types to create a comprehensive and robust representation of the underlying system or phenomenon [42]. When we merge different extracted features, we end up with a large size of new features. These features are not equally important for understanding our messages.

Algorithm 1 Simultaneous SMS Spam Detection

Input: UCI SMS dataset
Output: Spam detection model
Procedure:
1. Load the SMS dataset
2. Preprocess the data
3. Perform feature extraction:
    a. Extrad global features using TF-IDF
    b. Extrad temporal features using LSTM
    c. Extrad local features using CNN
4. Fuse all extracted features
5. Apply attention mechanism to derive feature weights
6. Train the model using weighted features

return Trained spam detection model
End Procedure

D. FEATURE SELECTION

In addition to merging the different types of extracted features, an important supplementary step involves feature selection, especially in high-dimensional datasets [43]. Attention mechanisms have emerged as a recent approach to feature selection and have shown high effectiveness in various applications [44]. In our model, the attention layer assigns weights to the various feature types based on their relevance to spam detection. This allows the model to prioritize and focus on the most significant features, thereby improving its effectiveness in identifying spam messages.

E. CLASSIFICATION

The last layer of our model is a dense layer that uses the output of the attention layer for classification. With our binary classification task (spam or not spam), we employ a dense layer with one neuron and a sigmoid activation function. The output of this layer represents the probability of a message being spam, ranging from 0 to 1. Values closer to 0 indicate non-spam messages, while values closer to 1 indicate spam.

The proposed method steps can be summarized in Algorithm 1.

V. EXPERIMENT

A. DATASET

The SMS spam collection v.1 is an open-access dataset comprising SMS messages categorized as either ''spam'' or ''ham'' (not spam). The dataset consists of 5,574 messages, with 4,827 classified as ham and 747 as spam. These messages were sourced from UK users and encompassed various message types, such as promotional offers, alerts, notifications, and personal messages. An overview of the dataset's statistics can be found in Table 1.

B. EVALUATION MEASURES

The effectiveness of the proposed model was evaluated using standard metrics such as accuracy, precision, recall, and f1-score. These metrics were calculated based on the confusion matrix, which encapsulated the model's true positives, false positives, true negatives, and false negatives. However, the evaluation faced challenges due to the imbalanced nature of the SMS spam collection v.1 dataset, where the number of spam messages was significantly fewer than that of ham messages. To address this class imbalance, cross-validation was employed. This technique helps ensure that the model's performance is consistently assessed across multiple subsets of the data, minimizing the potential bias introduced by the skewed distribution of classes. In light of this imbalance, the proposed model's effectiveness was assessed comprehensively, incorporating accuracy, precision, recall, and f1-score alongside the confusion matrix, as shown in Table. 2.

TABLE 1. The statistics of the dataset.

	Number of SMS	Percentage of SMS
HAM	4827	86.6%
SPAM	747	13.4%
Total	5574	100%

TABLE 2. The confusion matrix.

Actual	Predicted	
	Spam	Ham
Spam	TP	FN
Ham	FP	TN

- TP (true positive): correctly predicted spam messages.
- TN (true negative): correctly predicted non-spam messages (ham).
- FP (false positive): non-spam messages incorrectly classified as spam.
- FN (false negative): spam messages incorrectly classified as non-spam.

Accuracy: reflects how well the model correctly classifies spam and non-spam messages among all the messages predicted by the model. It considers true positives (correctly predicted spam) and true negatives (correctly predicted non-spam) while accounting for false positives and false negatives. The formula calculates the ratio of correctly classified messages (TP and TN) to the total number of predicted messages (TP, FP, TN, and FN) [45]. It is calculated as

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FP} \tag{5}$$

Precision: focuses on the accuracy of positive predictions, specifically identifying how many messages predicted as spam are spam. This metric is valuable when minimizing false positives is crucial. The formula calculates the ratio of true positives to the total number of messages predicted as positive (both true positives and false positives). It is calculated as

$$Precision = \frac{TP}{(TP + FP)} \tag{6}$$

Recall: measures the ability of the model to identify all actual positive cases (spam messages), including true positives, while excluding false negatives. It is particularly important when the aim is to ensure that as few actual positive cases as possible are missed. The formula calculates the ratio of true positives to the total actual positive cases (both true positives and false negatives). It is calculated as

$$Recall = \frac{TP}{(TP + FN)} \tag{7}$$

F1-Score: combines precision and recall, providing a balanced assessment of the model's performance. It is particularly useful when there is a need to balance precision and recall. The harmonic mean is used to account for cases where one of the metrics might be significantly higher than the other, resulting in a balanced score. The formula calculates the F1-Score by taking into account both precision and recall, considering their contribution in a harmonic way. It is calculated as.

$$F1score = \frac{2 \times (Recall \times Precision)}{(Recall + Precision)} \tag{8}$$

C. DATA SPLITTING

The UCI SMS Spam dataset is unbalanced. To address this issue, we employed sampling methods to modify the original data and achieve a more balanced distribution. Specifically, we adopted the 10-fold stratified cross-validation approach [46], ensuring that each fold maintained the class distribution while random undersampling was performed to balance the number of ''ham'' and ''spam'' instances. The results presented in this paper are based on the average values obtained from the 10-fold cross-validation. This approach allows for robust evaluation and provides a more comprehensive assessment of the proposed SMS spam detection technique.

D. DATA TRAINING

We need to feed the training data into the model, adjust its weights to minimize the loss function, and evaluate its performance using 10-fold stratified cross-validation to train a model for SMS spam detection. During training, the model receives input data in batches. By adjusting its weights or parameters, the model seeks to minimize the discrepancy between its predicted output and the true label of the SMS message. This iterative adjustment process aims to optimize the model's performance and improve its accuracy in classifying SMS messages. The training process continues until a stopping criterion is met, such as a certain number of epochs or when the model's performance on the cross-validation set stops improving.

The proposed model consists of several layers with specific functions and parameters. The input layer for tokenized and padded text sequences is the starting point. The embedding layer maps each word to a 128-dimensional vector. Next, a 1D convolutional layer (conv1D) with 128 filters and a kernel size of 3 is used to extract features. The globalmaxpooling1D operation reduces the dimensions of the CNN output [43].

An LSTM layer with 64 units is then used to capture the contextual information in the text. The input layer for TF-IDF features is added, followed by a concatenation layer that merges the outputs of the CNN, LSTM, and TF-IDF input layers. An attention layer with 128 units learns the importance of each feature [47]. The model continues with a fully connected layer (dense 1) containing 256 units and Relu activation, followed by a dropout layer with a rate of 0.5 to prevent overfitting [48]. Another fully connected layer (dense 2) with 128 units and Relu activation is added, along with a second dropout layer with a rate of 0.5.

Finally, the output layer consists of 2 units (Spam or non-spam) and a softmax activation function for classification.

The evaluation of the model's performance is carried out using 10-fold stratified cross-validation. This process involves dividing the dataset into ten subsets of roughly equal size while preserving the proportion of spam and non-spam messages in each subset. The model is trained and evaluated ten times, with each fold as the validation set once and the remaining data as the training set. The values reported are the average values for the ten folds, providing a more robust assessment of the model's performance.

By using 10-fold cross-validation, we can ensure a more comprehensive and reliable evaluation of our proposed model's effectiveness in SMS spam detection. This approach allows us to optimize hyperparameters and assess performance more confidently, leading to more meaningful and trustworthy research findings. Additionally, it is worth noting that the different layers in our model operate simultaneously (in parallel), contributing collectively to the model's enhanced performance.

VI. RESULTS AND ANALYSIS

The proposed method has demonstrated a good performance. Our model achieved an accuracy of 99.56%, showing it is great at spotting spam messages. The precision score of 0.9990 means it correctly labeled most spam messages, while the recall of 0.9960 shows it caught a large portion of actual spam. The F1-score of 0.9975 is a good balance of precision and recall. Our method surpasses competing methodologies. Noteworthy alternatives include TF-IDF with random forest achieving 97.50%, FRNN-HHO at 98.10%, CNN+LSTM reaching 98.37%, and HMM with 98.50%. Additionally, CSALSTM achieves 98.74%, CNN-BiLSTM records 98.76%, and M-Transformer demonstrates 98.92%, while GA+XGBoost and BERT showcase 99.1% and 99.28% accuracy, respectively. In our assessment, the utilization of a confusion matrix aids in measuring our model's ability to distinguish between spam and non-spam messages [49]. This matrix considers True Positives, True Negatives, False Positives, and False Negatives, allowing us to calculate accuracy, precision, recall, and the F1-score. By scrutinizing this matrix, we gain insights into performance and improvement areas.

Figure 4 provides a comparative visualization of our proposed model's exceptional SMS spam detection accuracy of

TABLE 3. Performance comparison of SMS spam detection approaches.

Methods	Accuracy	Precision	Recall	F1 score
TF-IDF + RF [15]	97.50%	0.980	0.970	0.970
FRNN-HHO [27]	98.10%	0.997	0.981	0.989
CNN+LSTM [24]	98.37%	0.953	0.878	0.914
HMM [18]	98.50%	0.985	0.986	0.986
CSA-LSTM [23]	98.74%	0.933	0.965	0.949
CNN-BiLSTM [25]	98.76%	0.988	0.993	0.972
M-Transformer [22]	98.92%	0978	0.945	0.961
GA+XGBoost [19]	99.1%	0.983	0.946	0.968
BERT [26]	99.28%	0.996	0.992	0.990
Proposed method	**99.56%**	**0.999**	**0.996**	**0.997**

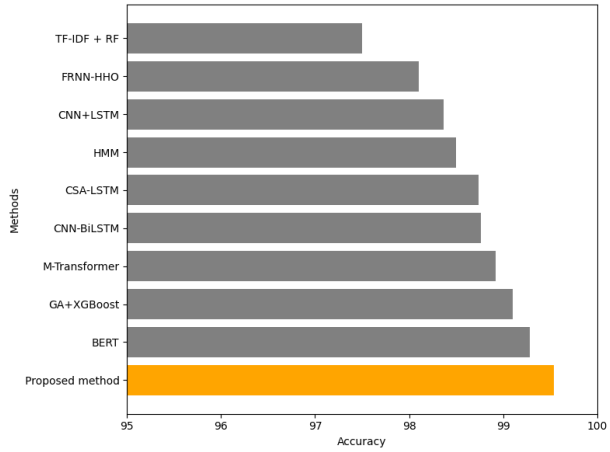

FIGURE 4. Comparison of SMS spam detection methods.

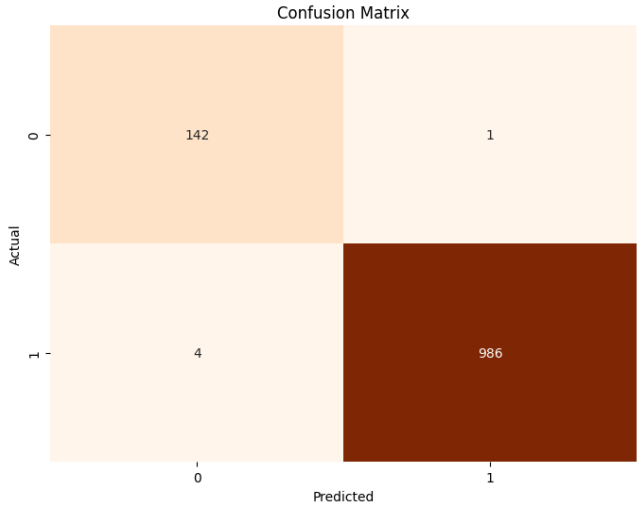

FIGURE 5. Confusion matrix for the proposed method.

99.56% against alternative methods, showcasing its superior performance. Figure 5 depicts the confusion matrix for our SMS spam detection model, incorporating CNN, TF-IDF, and LSTM techniques. It provides a visual representation of how well the model separates spam and non-spam messages, showcasing strong precision and minimal misclassification, affirming the effectiveness of our approach. Cross-validation significantly contributes to the assessment of our SMS spam

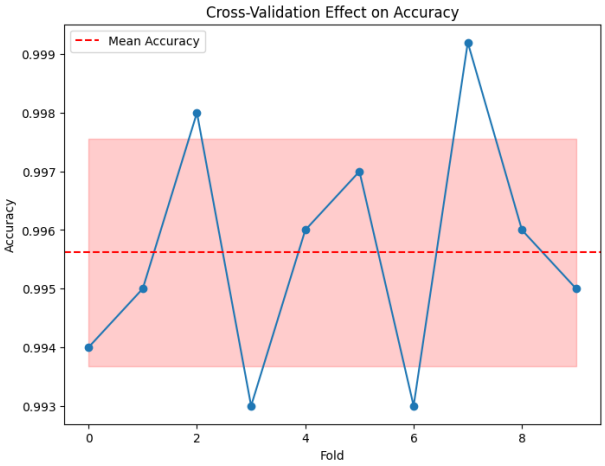

FIGURE 6. Training and validation accuracy curve of the datasets.

detection model's robustness. Figure 6 showcases the cross-validation effect, plotting accuracy across ten distinct dataset divisions. The red dashed line indicates a mean accuracy of 99.56%, with the shaded area around it representing consistency. This rigorous validation approach confirms the stability and credibility of our SMS spam detection methodology. Our proposed hybrid model, which combines simultaneous feature extraction using CNN, LSTM, and TF-IDF, along with data fusion, achieved the best results among all evaluated models. That confirms the potential of combining deep learning techniques to create effective SMS spam detection and classification systems. While our proposed method has achieved state-of-the-art performance in short text classification tasks, it is essential to recognize its limitations to understand its constraints comprehensively. Firstly, our model may not generalize effectively to diverse datasets, as it was trained and evaluated on a specific dataset. Our evaluation methodology is task-specific, and its performance across various tasks remains unexplored. Furthermore, our model is not trained for multi-language text spam detection, a complex task due to language and text style variations. To address these limitations, we plan to enhance generalization, explore broader task domains, and incorporate multi-language capabilities in future research.

VII. CONCLUSION AND FUTURE WORK

This study introduced a hybrid method for detecting SMS spam using a combination of local, temporal, and global features using CNN, LSTM, and TF-IDF techniques simultaneously with early data fusion. The effectiveness of our proposed method was evaluated against several other well-known SMS spam filtering methods, including FRNN-HHO-KEL, TF-IDF-RF, CNN-LSTM, CSA-LSTM, HMM, CNN-BiLSTM, Modified Transformer, GA and XGBoost, and BERT. Our proposed spam filtering model, tested on the SMS spam collection v.1 dataset, outperformed other classifiers in results. This affirms its proficiency in identifying SMS spam. also, the proposed method solved the

imbalance of the UCI SMS Collection V.1 dataset by applying the cross-validation technique.

In future work, we aim to enhance our model's performance by integrating diverse data sources, including linguistic features and user behaviour patterns. We plan to assess its scalability and generalizability through tests on various datasets, especially the Multilingual dataset. Additionally, we will explore advanced techniques like ensemble learning and transfer learning to boost the model's efficiency. These steps are geared towards refining our model, making it more effective and adaptable for practical applications.

REFERENCES

[1] M. Morreale, "Daily SMS mobile statistics," SME EAGLE, Poznan, Poland, Mar. 2017. [Online]. Available: https://www.smseagle.eu/2017/03/06/daily-sms-mobile-statistics/

[2] S. Rao, A. K. Verma, and T. Bhatia, "A review on social spam detection: Challenges, open issues, and future directions," *Expert Syst. Appl.*, vol. 186, Dec. 2021, Art. no. 115742.

[3] S. J. Delany, M. Buckley, and D. Greene, "SMS spam filtering: Methods and data," *Expert Syst. Appl.*, vol. 39, no. 10, pp. 9899–9908, Aug. 2012.

[4] P. K. Roy, J. P. Singh, and S. Banerjee, "Deep learning to filter SMS spam," *Future Gener. Comput. Syst.*, vol. 102, pp. 524–533, Jan. 2020.

[5] I. K. S. Al-Tameemi, M.-R. Feizi-Derakhshi, S. Pashazadeh, and M. Asadpour, "Interpretable multimodal sentiment classification using deep multi-view attentive network of image and text data," *IEEE Access*, vol. 11, pp. 91060–91081, 2023.

[6] Z. Wu, J. Cao, Y. Wang, Y. Wang, L. Zhang, and J. Wu, "HPSD: A hybrid PU-learning-based spammer detection model for product reviews," *IEEE Trans. Cybern.*, vol. 50, no. 4, pp. 1595–1606, Apr. 2020.

[7] F. Carrera, V. Dentamaro, S. Galantucci, A. Iannacone, D. Impedovo, and G. Pirlo, "Combining unsupervised approaches for near real-time network traffic anomaly detection," *Appl. Sci.*, vol. 12, no. 3, p. 1759, Feb. 2022.

[8] A. Shrestha and A. Mahmood, "Review of deep learning algorithms and architectures," *IEEE Access*, vol. 7, pp. 53040–53065, 2019.

[9] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, Jul. 2019.

[10] A. F. Gad, *Practical Computer Vision Applications Using Deep Learning With CNNs*. New York, NY, USA: Apress, 2018.

[11] V. Gattulli, D. Impedovo, G. Pirlo, and L. Sarcinella, "Cyber aggression and cyberbullying identification on social networks," in *Proc. 11th Int. Conf. Pattern Recognit. Appl. Methods*, 2022, pp. 644–651.

[12] T. Meng, X. Jing, Z. Yan, and W. Pedrycz, "A survey on machine learning for data fusion," *Inf. Fusion*, vol. 57, pp. 115–129, May 2020.

[13] S. B. Alex, L. Mary, and B. P. Babu, "Attention and feature selection for automatic speech emotion recognition using utterance and syllable-level prosodic features," *Circuits, Syst., Signal Process.*, vol. 39, no. 11, pp. 5681–5709, Nov. 2020.

[14] S. Y. Yerima and A. Bashar, "Semi-supervised novelty detection with one class SVM for SMS spam detection," in *Proc. 29th Int. Conf. Syst., Signals Image Process. (IWSSIP)*, Jun. 2022, pp. 1–4.

[15] N. N. Amir Sjarif, N. F. Mohd Azmi, S. Chuprat, H. M. Sarkan, Y. Yahya, and S. M. Sam, "SMS spam message detection using term frequency-inverse document frequency and random forest algorithm," *Proc. Comput. Sci.*, vol. 161, pp. 509–515, Jan. 2019.

[16] O. Abayomi-Alli, S. Misra, A. Abayomi-Alli, and M. Odusami, "A review of soft techniques for SMS spam classification: Methods, approaches and applications," *Eng. Appl. Artif. Intell.*, vol. 86, pp. 197–212, Nov. 2019.

[17] T. A. Almeida, J. M. G. Hidalgo, and A. Yamakami, "Contributions to the study of SMS spam filtering," in *Proc. 11th ACM Symp. Document Eng.*, Sep. 2011, pp. 259–262.

[18] T. Xia and X. Chen, "A discrete hidden Markov model for SMS spam detection," *Appl. Sci.*, vol. 10, no. 14, p. 5011, Jul. 2020.

[19] N. Ghatasheh, I. Altaharwa, and K. Aldebei, "Modified genetic algorithm for feature selection and hyper parameter optimization: Case of XGBoost in spam prediction," *IEEE Access*, vol. 10, pp. 84365–84383, 2022.

[20] G. Ubale and S. Gaikwad, "SMS spam detection using TFIDF and voting classifier," in *Proc. Int. Mobile Embedded Technol. Conf. (MECON)*, Mar. 2022, pp. 363–366.

[21] W. H. Gomaa, "The impact of deep learning techniques on SMS spam filtering," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 1, pp. 1–6, 2020.

[22] X. Liu, H. Lu, and A. Nayak, "A spam transformer model for SMS spam detection," *IEEE Access*, vol. 9, pp. 80253–80263, 2021.

[23] A. A. Bataineh and D. Kaur, "Immunocomputing-based approach for optimizing the topologies of LSTM networks," *IEEE Access*, vol. 9, pp. 78993–79004, 2021.

[24] A. Ghourabi, M. A. Mahmood, and Q. M. Alzubi, "A hybrid CNN-LSTM model for SMS spam detection in Arabic and English messages," *Future Internet*, vol. 12, no. 9, p. 156, Sep. 2020.

[25] S. E. Rahman and S. Ullah, "Email spam detection using bidirectional long short term memory with convolutional neural network," in *Proc. IEEE Region 10 Symp. (TENSYMP)*, Jun. 2020, pp. 1307–1311.

[26] K. Debnath and N. Kar, "SMS spam detection using deep learning approach," in *Human-Centric Smart Computing*. Singapore: Springer, 2023.

[27] U. Srinivasarao and A. Sharaff, "SMS sentiment classification using an evolutionary optimization based fuzzy recurrent neural network," *Multimedia Tools Appl.*, pp. 42207–42238, Nov. 2023.

[28] M.-R. Feizi-Derakhshi, Z. Mottaghinia, and M. Asgari-Chenaghlu, "Persian text classification based on deep neural networks," *Soft Comput. J.*, vol. 11, no. 1, pp 1–11, Nov.2022.

[29] H. Lee, S. Jeong, S. Cho, and E. Choi, "Visualization technology and deep-learning for multilingual spam message detection," *Electronics*, vol. 12, no. 3, p. 582, Jan. 2023.

[30] H. Liang, X. Sun, Y. Sun, and Y. Gao, "Text feature extraction based on deep learning: A review," *EURASIP J. Wireless Commun. Netw.*, vol. 2017, no. 1, p. 211, Dec. 2017.

[31] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions," *J. Big Data*, vol. 8, no. 1, p. 53, Mar. 2021.

[32] H. Jelodar, Y. Wang, R. Orji, and S. Huang, "Deep sentiment classification and topic discovery on novel coronavirus or COVID-19 online discussions: Nlp using LSTM recurrent neural network approach," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 10, pp. 2733–2742, Oct. 2020.

[33] S. Robertson, "Understanding inverse document frequency: On theoretical arguments for IDF," *J. Documentation*, vol. 60, no. 5, pp. 503–520, Oct. 2004.

[34] J.-W. Sun, J.-Q. Bao, and L.-P. Bu, "Text classification algorithm based on TF-IDF and bert," in *Proc. 11th Int. Conf. Inf. Commun. Technol. (ICTech))*, Feb. 2022, pp. 1–4.

[35] J. Bleiholder and F. Naumann, "Data fusion," *ACM Comput. Surv.*, vol. 41, pp. 1–41, Jan. 2009.

[36] S. Lu, M. Liu, L. Yin, Z. Yin, X. Liu, and W. Zheng, "The multi-modal fusion in visual question answering: A review of attention mechanisms," *PeerJ Comput. Sci.*, vol. 9, May 2023, Art. no. e1400.

[37] L. Zhang, Y. Xie, L. Xidao, and X. Zhang, "Multi-source heterogeneous data fusion," in *Proc. Int. Conf. Artif. Intell. Big Data (ICAIBD)*, May 2018, pp. 47–51.

[38] G. Liu and J. Guo, "Bidirectional LSTM with attention mechanism and convolutional layer for text classification," *Neurocomputing*, vol. 337, pp. 325–338, Apr. 2019.

[39] A.-R. Feizi-Derakhshi, M.-R. Feizi-Derakhshi, M. Ramezani, N. Nikzad-Khasmakhi, M. Asgari-Chenaghlu, T. Akan, M. Ranjbar-Khadivi, E. Zafarni-Moattar, and Z. Jahanbakhsh-Naghadeh, "Text-based automatic personality prediction: A bibliographic review," *J. Comput. Social Sci.*, vol. 5, no. 2, pp. 1555–1593, Nov. 2022.

[40] G. Kim, J. Son, J. Kim, H. Lee, and H. Lim, "Enhancing Korean named entity recognition with linguistic tokenization strategies," *IEEE Access*, vol. 9, pp. 151814–151823, 2021.

[41] S. Lu, Y. Ding, M. Liu, Z. Yin, L. Yin, and W. Zheng, "Multiscale feature extraction and fusion of image and text in VQA," *Int. J. Comput. Intell. Syst.*, vol. 16, no. 1, p. 54, Apr. 2023.

[42] F. Alam, R. Mehmood, I. Katib, N. N. Albogami, and A. Albeshri, "Data fusion and IoT for smart ubiquitous environments: A survey," *IEEE Access*, vol. 5, pp. 9533–9554, 2017.

[43] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, *Feature Selection for High-Dimensional Data*. Cham, Switzerland: Springer, 2015.

[44] F. Jia, J. Xu, Y. Ma, J. Wang, and Z. Liang, "Joint attention mechanism feature selection for single image reflection separation," in *Proc. 26th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2022, pp. 1982–1988.
[45] X. Liu, T. Shi, G. Zhou, M. Liu, Z. Yin, L. Yin, and W. Zheng, "Emotion classification for short texts: An improved multi-label method," *Humanities Social Sci. Commun.*, vol. 10, no. 1, pp. 1–9, Jun. 2023.
[46] A. Cannarile, V. Dentamaro, S. Galantucci, A. Iannacone, D. Impedovo, and G. Pirlo, "Comparing deep learning and shallow learning techniques for API calls malware prediction: A study," *Appl. Sci.*, vol. 12, no. 3, p. 1645, Feb. 2022.
[47] Q. Zhang, Y. Liu, C. Gong, Y. Chen, and H. Yu, "Applications of deep learning for dense scenes analysis in agriculture: A review," *Sensors*, vol. 20, no. 5, p. 1520, Mar. 2020.
[48] A. Mnih and Y. W. Teh, "A fast and simple algorithm for training neural probabilistic language models," in *Proc. 29th Int. Conf. Mach. Learn.*, 2012, pp. 419–426.
[49] M. Heydarian, T. E. Doyle, and R. Samavi, "MLCM: Multi-label confusion matrix," *IEEE Access*, vol. 10, pp. 19083–19095, 2022.

HUSSEIN ALAA AL-KABBI received the bachelor's degree in computer engineering from EECT, Baghdad, Iraq, and the master's degree in software engineering from IRIU, Mashhad, Iran, in 2019. He is currently pursuing the Ph.D. degree with the Department of Computer Engineering, University of Tabriz, Iran. His research interests include natural language processing, machine learning, deep learning, and optimization algorithms.

MOHAMMAD-REZA FEIZI-DERAKHSHI received the B.S. degree in software engineering from the University of Isfahan, Iran, and the M.Sc. and Ph.D. degrees in artificial intelligence from the Iran University of Science and Technology, Tehran, Iran. He is currently a Professor with the Faculty of Computer Engineering, University of Tabriz, Iran. His research interests include natural language processing, optimization algorithms, deep learning, social network analysis, and intelligent databases.

SAEID PASHAZADEH received the B.Sc. degree in computer engineering from the Sharif University of Technology, Tehran, Iran, in 1995, and the M.Sc. and Ph.D. degrees in computer engineering from the Iran University of Science and Technology, Tehran, in 1998 and 2010, respectively. He is currently a Professor with the Department of Information Technology, Faculty of Electrical and Computer Engineering, University of Tabriz, Iran. His research interests include formal verification, software engineering, modeling and verification, performance evaluation, and distributed systems.