Received: 18 March 2020 | Revised: 21 February 2022 | Accepted: 9 March 2022

DOI: 10.1002/cpe.6989

RESEARCH ARTICLE

WILEY

# A deep learning method for automatic SMS spam classification: Performance of learning algorithms on indigenous dataset

Olusola Abayomi-Alli[1] | Sanjay Misra[2] | Adebayo Abayomi-Alli[3]

[1]Department of Software Engineering, Kaunas University of Technology, Kaunas, Lithuania

[2]Department of Computer Science and Communication, Østfold University College, Halden, Norway

[3]Department of Computer Science, Federal University of Agriculture, Abeokuta, Nigeria

**Correspondence**
Sanjay Misra, Department of Computer Science and Communication, Ostfold University College, Halden, Norway.
Email: sanjay.misra@hiof.no

## Abstract

SMS, one of the most popular and fast-growing GSM value-added services worldwide, has attracted unwanted SMS, also known as SMS spam. The effects of SMS spam are significant as it affects both the users and the service providers, causing a massive gap in trust among both parties. This article presents a deep learning model based on BiLSTM. Further, it compares our results with some of the states of the art machine learning (ML) algorithm on two datasets: our newly collected dataset and the popular UCI SMS dataset. This study aims to evaluate the performance of diverse learning models and compare the result of the new dataset expanded (ExAIS_SMS) using the following metrics the true positive (TP), false positive (FP), F-measure, recall, precision, and overall accuracy. The average accuracy for the BiLSTSM model achieved moderately improved results compared to some of the ML classifiers. The experimental results achieved significant improvement from the ground truth results after effective fine-tuning of some of the parameters. The BiLSTM model using the ExAIS_SMS dataset attained an accuracy of 93.4% and 98.6% for UCI datasets. Further comparison of the two datasets on the state-of-the-art ML classifiers gave an accuracy of Naive Bayes, BayesNet, SOM, decision tree, C4.5, J48 is 89.64%, 91.11%, 88.24%, 75.76%, 80.24%, and 79.2% respectively for ExAIS_SMS datasets. In conclusion, our proposed BiLSTM model showed significant improvement over traditional ML classifiers. To further validate the robustness of our model, we applied the UCI datasets, and our results showed optimal performance while classifying SMS spam messages based on some metrics: accuracy, precision, recall, and F-measure.

KEYWORDS

algorithms, classification, deep learning, machine learning, short messages

## 1 | INTRODUCTION

Advancements in mobile device technology have increased user's dependence due to portability and ever-present use in daily living,[1] thereby making it easier for users to store exclusive and private information such as banking details, contact lists, emails, medical records and other delicate records.[2] In addition, the rapid growth of mobile networks has enhanced everyday activities in distributing information and improving communications in real-time.[3] The most popular service of the GSM is the Short Message Service (SMS) due to its network reliability, cheap use, and non-internet enabled services, with over 200,000 SMS sent per second.[4]

Modern-day technology has increased the number of messaging applications on mobile devices such as WhatsApp, Wechat, Hangout and so forth.[5] However, with all the new messaging services available on smart devices, the impact of SMS on mobile users still remains on the rise based on the cheap rate of sending text and network reliability, thereby making SMS messages a profitable alternative for GSM subscribers.[6] Notably, SMS is also accessible on a broad span of network standards such as code division multiple access (CDMA), time division multiple access (TDMA), and supported tactically on every phone.

The increasing usage of SMS by mobile users has built spammers' interest, and literature has shown that 33% of SMS in Asia is considered spam.[7] SMS spam is still an emerging problem globally due to mobile users' high response rate based on its trusted and personal services.[8,9] Spam is generally referred to as an unwanted or unsolicited message sent randomly by an individual without connection to the user, primarily for commercial, fraudulent, and malicious intent.[10,11]

The motivation of this research study is based on the continuous rise of SMS spam received by an end-user network congestion via SMS flood at the mobile network operators' end.[12] In addition, an SMS spammer uses telemarketing to congest a network; hence the SMS gateway has a potential problem since it uses a script to send a large number of messages through one gateway, thus creating a denial or delay of service to mobile users. The effect of SMS spam on users is relatively high compared to email spam because SMS subscribers feel more secure using this service for exchanging confidential information, sanctioning payments, and other daily applications. Effective classification of SMS is very challenging due to the following[13]:

1. The limited length of characters involved in SMS (160 characters) affects the overall availability of a sufficient SMS dataset for training and testing purposes.
2. The constant use of idiosyncratic languages and unstandardized abbreviations (punctuation, emoticons, etc.) influences the performance of existing classifiers.

Existing literature on SMS classification has shown research efforts to detect spam messages using different approaches and methods. However, there is still a crucial need to identify practical approaches for classifying SMS efficiently, thus securing users' information and providing a better experience for users.[14] Bin et al.[15] analyzed the existing problems of SMS spam as:

1. The inability of existing SMS spam detection techniques to filter appropriately on the network side;
2. The ability of spammers to bypass the traditional ways of spam filtering and
3. Low accuracy of existing SMS spam classifiers.

However, the success of machine learning (ML) techniques in filtering SMS spam relies extremely on the choice of a suitable feature set for classification.[13] This research study presents SMS spam detection on two datasets by applying the vast and effective performance of the BiLSTM model for efficient detection. We also compared our results with the best six ML algorithms.This research study raises the following questions:

RQ1: What is the accuracy of various ML algorithms on the ExAIS SMS dataset?
RQ2: What is the accuracy of various ML algorithms on the UCI SMS dataset?
RQ3: How can we measure the computational cost of various ML classifiers on these two datasets with respect to time and speed?

This study aims at conducting a comparative experiment on the new dataset and also comparing the result of the ExAIS_SMS dataset on MATLAB environment for the BiLSTM model, and the WEKA platform was used to evaluate the performance of six different ML algorithms. The remaining part of the article is sectioned as Section 2, discussing the literature review and related work in detail. Section 3 presents the methodology, while the result obtained and discussion was presented in Section 4. The study concludes with the future recommendations in Section 5.

## 2 | BACKGROUND AND RELATED WORKS

This section reviews closely related SMS spam filtering techniques using various ML algorithms and backgrounds of the work. Some of the state-of-the-art methods and algorithms that are often used in the SMS spam classification are explained in this section in Section 2.1 and the related work in Section 2.2.

### 2.1 | Various models and algorithms

A brief description of BiLSTM and other ML classifiers is discussed in the subsections.

### 2.1.1 | Deep learning models

Deep learning models have consistently shown a great and significant performance in natural language-based applications ranging from text classification,[16] sentiment analysis, email spam detection, fake news detection and so forth. One of the deep learning models used efficiently in diverse natural language-based applications is the recurrent neural network (RNN). The unique feature of RNN architecture is the application of a gating mechanism consisting of an input gate, forget gate, and output gate,[16] which aids in addressing the preservation of long-term information. A good example of the RMM model can be fully expressed in LSTM and BiLSTM models. Bi-LSTM neural network consists of a bi-directional LSTM that could be trained in both time directions concurrently, as shown in Figure 1,[17] with different hidden layers.[18] This method has been widely applied to different research domains due to its efficient properties. In many recent researches, the overall performance of the BiLSTM model simply outperforms other traditional RNNs. However, to the best of our knowledge, this article is the first to analyze the impact of BiLSTM methods in SMS spam classification.

### 2.1.2 | Machine learning

ML involves learning structure from data through the combination of representation, evaluation, and optimization.[19] ML, also referred to as data mining or predictive analytics, is expected to drive the next big wave of innovation.[20] Its application is useful in performing a task associated with artificial Intelligence such as classification, prediction, speech recognition, diagnosis, face recognition, robot control, named entity recognition, ranking and so forth.[21,22] However, the representation of text documents is crucial for ML approaches for text classification.[23] According to the literature, ML can be categorized into three classes which include: supervised, unsupervised, and semi-supervised learning.

Learned-Miller[24] described supervised learning as the task of learning from a deduced function (classifier) based on training data which comprises a set of examples which are input object (vector) and anticipated output (supervisory signal). In addition, supervised ML predicts the right output value for any valid input, thereby drawing conclusions from training data in a practicable way. Examples of supervised models are neural networks, decision trees, and multilayer perceptron. This learning method is very expensive since the majority of the data comes as unlabeled; therefore, the cost of labeling this data for prior knowledge is very expensive.

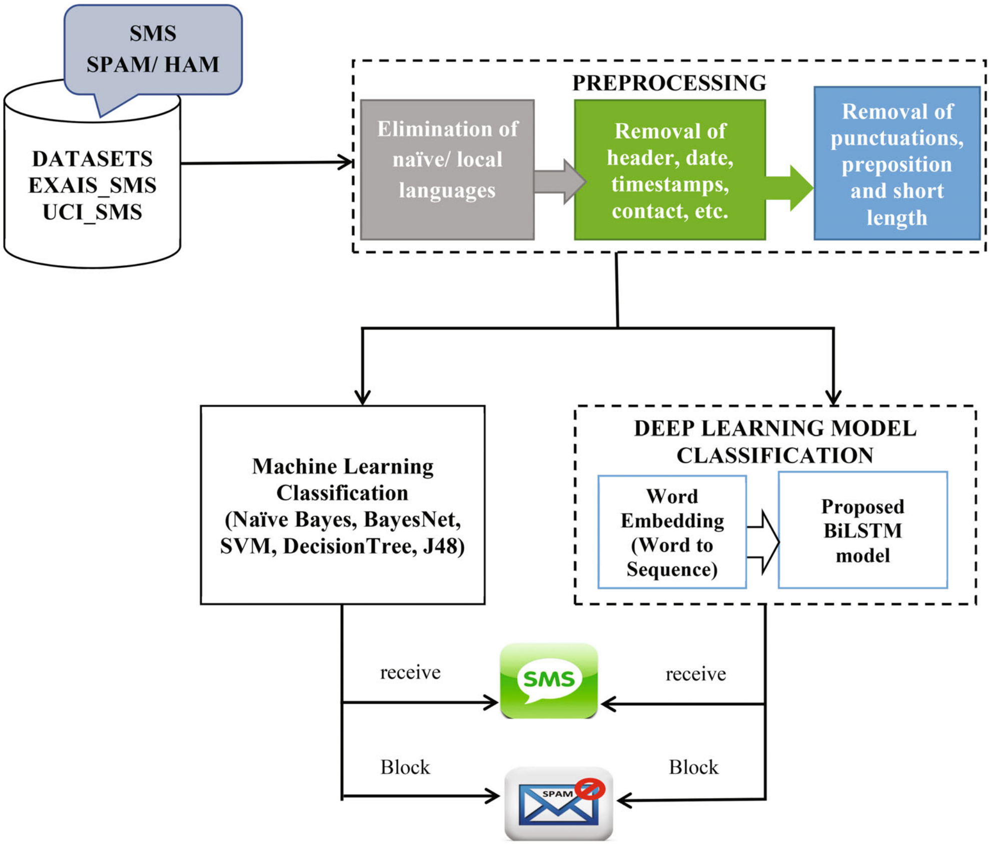

SMS
SPAM/ HAM
DATASETS
EXAIS_SMS
UCI_SMS
PREPROCESSING
Elimination of naïve/ local languages
Removal of header, date, timestamps, contact, etc.
Removal of punctuations, preposition and short length
Machine Learning Classification (Naïve Bayes, BayesNet, SVM, DecisionTree, J48)
DEEP LEARNING MODEL CLASSIFICATION
Word Embedding (Word to Sequence)
Proposed BiLSTM model
receive
SMS
receive
Block
SPAM
Block

**FIGURE 1** The flow diagram of the proposed SMS spam filter

Unsupervised learning is basically learning with unlabeled data used in the clustering of input classes on the basis of their statistical properties only.

Semi-supervised utilizes both labeled and unlabeled data to perform the learning task. It involves the use of large numbers of unlabeled data against the minimal number of labeled data. This type of ML has attracted a lot of attention in different research domains such as web mining and so forth.[25] Examples of semi-supervised learning are Gaussian distribution (GMM), hidden Markov models (HMM), multinomial distributions and so forth.[26]

This study aims at evaluating the performance of ML classifiers on two datasets on MATLAB and WEKA environments. For a detailed literature review, there is a need to discuss some of the ML classifiers used in this study which include: Naïve Bayes, BayesNet, support vector machine (SVM), K-Nearest Neighbor Decision Rule, J48 and so forth.

*Naïve Bayes*

Based on existing literature, NB has shown to be the simplest means of classification, which is based on probabilistic analysis from Bayes' theorem (Bayesian statistics) with strong (Naive) independence assumptions.[27] This classifier combines the initial understanding of observed data, thereby assigning a posterior probability to a class based on its initial likelihood given from the training data.[24] In addition, there is an assumption of conditional independence between attributes hence, setting maximum a posterior class to new instances. The mathematical expression for NB classifier is given as:

$$P\left(a_n = w_k | v_j\right) \tag{1}$$

$$v_{NB} = \arg_{v_j \epsilon \left\{ \substack{\max \\ \text{spam,ham}} \right\}} \mathrm{P}\left(v_j\right) \prod_n P\left(a_n | v_j\right) \tag{2}$$

where: Prob. of the *n*th word in the SMS is the k-word in our lexicon, given the SMS has been classified as $v_j$. $v_{NB}$ is the naïve Bayes classifier.

*Support vector machine*

A SVM is a non-linear classifier and a typical example of supervised learning that uses a discriminative classifier formally defined by a separating hyperplane.[28] SVM has been stated based on the existing study as a superior classifier when compared to other methods. However, the shortcoming of this method is the very large number of support vectors used in the training set, thus increasing the computational complexity with respect to time.[29] SVM decision function is represented as:

$$f(x) = \text{sgn}\left( \sum_{n=1}^{m} \propto_n y_n . k\left(x, x_n\right) + b \right) \tag{3}$$

where: α and y are the weights of the support vectors;

$x$ is the input vector;

$y_n$ is an element of +1 to −1;

b is the bias.

*K-nearest neighbor*

The k nearest-neighbor algorithm is considered a non-parametric method based on its non-initial assumptions of the class boundary. Instead, it adapts closely to non-linear boundaries as the values of training data increase.[30] The K-nearest-neighbor method can be thought of as estimating the values of the probabilities of the classes given A. The denser the points around X, the larger the value of k, and the more effective the evaluation. The Euclidean distance is used for calculating distance among numerical attributes for nearest-neighbor methods, as expressed in Equation (4).

The distance between two patterns ($a_{11}$, $a_{12}$, … , $a_{1n}$) and ($a_{21}$,$a_{22}$, … , $a_{2n}$) is:

$$\text{Euclidean distance between two patterns} = \sqrt{\sum_{j=1}^{n} \left(a_{1j} - a_{2j}\right)^2} \tag{4}$$

## 2.2 Literature review

Recent systematic reviews from previous researchers like Delany et al.,[13] Abayomi-Alli et al.,[17] Rao et al.[31] have demonstrated different classification methods, feature extraction, and selection approaches adapted by previous researchers in the analysis and detection of SMS spam classification. Sjarif et al.[32] presented a feature extraction method based on the term-frequency-inverse document frequency method to assess relevant words.

They applied different ML models to evaluate the classifier's overall performance. The study showed that the combination of TF-IDF and random forest algorithm outperforms other state-of-the-art algorithms, thus improving the detection of SMS spam. A similar study from Lim and Singh[33] proposed cost-sensitive techniques based on stacking of multilayer perceptron (MLP) algorithm and Bayesian network. The contribution of the study was its ability to minimize misclassification rate and very impressive performance when compared with other ML algorithms.

Sharaff et al.[34] proposed a novel SMS spam filter model based on a biologically inspired algorithm named krill herd optimization and dendritic cell algorithm. Their experimental results showed that the proposed model gave more accurate results compared with other ML classifiers like NB, LR, SVM, and XgBoost classifier. Another study from Bosaeed et al.[35] developed a multi-filter that applied multiple ML based classifiers using three classification methods, namely Naïve Bayes (NB), SVM, and Naïve Bayes Multinomial (NBM). The study shows the flexibility of multiple platforms by implementing their proposed model partly and fully on both mobile and server apps, thus ensuring computational resource optimization.

Alzahrani and Rawat[36] also presented a comparative study of different ML algorithms for SMS spam detection. Four ML algorithms were applied, and the best performance was achieved with the neural network algorithm compared with other classifiers. A similar study was carried out by authors Theodorus et al.[37] that compared the performance of eight ML classifiers in Bahasa Indonesia SMS text classification. Other applications of ML algorithms have been presented by different works of literature like the Naïve Bayes algorithm,[27,38–40] neural network classifier,[41] self organizing map,[15] KNN, H20 framework,[42] and so on. Ensemble learning was applied by Sisodia et al. The authors presented an automated framework for SMS spam classification using various classifiers, namely individual classifiers such as KNN, NB, SVM, ID3, CART, C4.5, and ensemble classifiers such as Adaboost, random forest, and voting. The experimental results showed promising results with the best accuracy obtained with the ensemble learning classifiers based on random forest.

Sharma and Sharaff[43] recently considered a different perspective, which applied genetic programming to SMS spam filters to reduce false-positive errors. The performance of the proposed method shows significant improvement in the classification of SMS spam with the increasing number of generations. Other evolutionary methods also gained effective relevance in the SMS spam classification; studies from Al-Hasan and El-Alfy[2] proposed a novel approach DCA on NB and SVM algorithms using various feature sets. Onashoga et al.[44] developed a collaborative and adaptive filtering system based on an artificial immune system similar to Mahmoud and Mahfouz,[45] which also applied artificial immune system for SMS spam.

Authors presented the application of deep learning techniques for SMS spam detection using LSTM in Gadde et al.[46] and Al-Bataineh and Kaur.[47] Authors in the former also applied three different word embedding techniques based on the count, TF-IDF, and hashing vectorizer. The experimental results for LSTM were compared with some of the state-of-the-art ML techniques. However, authors in the latter demonstrated the robustness of LSTM topologies with a clonal selection algorithm for text classification. The study was evaluated using three datasets and benchmarked with some of the state-of-the-art ML classifiers. The experimental results showed that their proposed model outperforms other models regarding accuracy, precision, recall, F1-score, and computational time. Roy et al.[48] also proposed a deep learning method based on convolutional neural networks and long short-term memory models in the classification of SMS spam. The study showed a significant performance of deep learning models using three configurations, and the addition of regularization parameters such as dropout improved classification accuracy. Another interesting work by Xia and Chen[49] introduced an enhanced Hidden Markov Model for a weighted feature and label words. Their study shows that the application of weighted features enhanced HMM outperforms the LSTM with respect to accuracy and computational speed.

A new method based on a lightweight deep neural model referred to as Lightweight Gated Recurrent Unit (LGRU) was presented by Wei and Nguyen[50] for SMS spam detection. In order to show the effectiveness of the proposed method, the authors compared their results with over 30 different machine and deep learning classifiers. Aside from that, the proposed method achieved better results compared with existing models; the authors also claimed that it also incurs less complexity in terms of training time. Authors Annareddy and Tammina[51] and Huang[52] applied deep learning models for the detection of SMS spam. The former showed a comparative study of two deep learning models based on a convolutional neural network and RNN on a large SMS corpus. The overall results show interesting findings. At the same time, the latter applied the CNN model on the Chinese Wikipedia corpus. The authors further demonstrated the impact of hyper-parameters in achieving optimal results. Another interesting finding that combined two deep learning models, namely RNNs and the LSTM model, was proposed by Chandra and Khatri.[53] The study showed that the proposed model could predict based on previous knowledge of patterns and the current vector set. The study concluded with a significant improvement in performance based on accuracy with excellent and acceptable runtime.

Lee and Kang[54] introduced word embedding methods for building a feature vector and applied deep learning methods for SMS spam classification. The experimental result shows little improvement in the performance of the deep learning method compared with the conventional ML method of SVM-light. Chen et al.[3] and Baaqeel and Zagrouba[55] introduced a hybrid system for SMS spam classification. The former applied a trust management scheme using behavioral, and SMS traffic data and the article concluded that the proposed prototype achieved effective detection with respect to efficiency, robustness, and accuracy. However, the latter combined six supervised models and unsupervised methods for SMS spam detection. However, the overall experiments showed that the combination of K-means with SVM gave an outstanding performance based on accuracy. Thus, the study concluded that the hybrid system performed way better than the single classifier.

Other interesting studies include the application of graph centrality in SMS spam detection,[56,57] factorial analysis filtering approaches on linguistic techniques,[14] the Rough-set and Naïve-Bayes algorithm,[58] FIMESS approach which uses external features like invalid characters, time inaccuracies and blacklisted keywords to filter SMS spam was presented in Androulidakis et al.[59]

After a thorough review of related studies, some of the identified shortfalls include the following: insufficient dataset for training effectively and classifying SMS, the problem of class imbalance, challenges with time complexity of some existing approaches, problem of increasing misclassification rate and so forth. Therefore, this study adopted a deep learning model based on BiLSTM for SMS spam classification using two SMS datasets. To further evaluate the robustness of our method, we compared our results with some-of-the-art ML algorithms based on performance and time complexity.

## 3 PROPOSED DESIGN METHODOLOGY

This section presents the steps involved in classification. The classification experiment is carried out using various ML algorithms. A schematic flow diagram of the proposed SMS filtering system is shown in Figure 1. This section is divided into three subsections, namely:

1. The datasets;
2. The preprocessing;
3. Classification (experiment and evaluation)

### 3.1 Datasets

This section describes the data set that was used for our evaluation. For the purpose of this work, two datasets were used: the ExAIS_SMS dataset* and the UCI dataset. The ExAIS_SMS dataset contains a total number of 5240 SMS messages,[60] while the UCI dataset contains 5572 SMS.[61]

### 3.2 Pre-processing stage

In this subsection, we preprocessed the SMS corpus using three different techniques on each SMS corpus. The following was carried out for preprocessing the corpus, which are:

i. Elimination of native languages from the SMS database: a thorough search was carried out to eliminate duplicated messages from the database. The overall total number of unique SMS evaluated in this study is 4420 SMS which consists of 2453 spam and 1967 ham messages. A summary of class distributions for the two datasets used in this study is depicted in Figures 2 and 3.

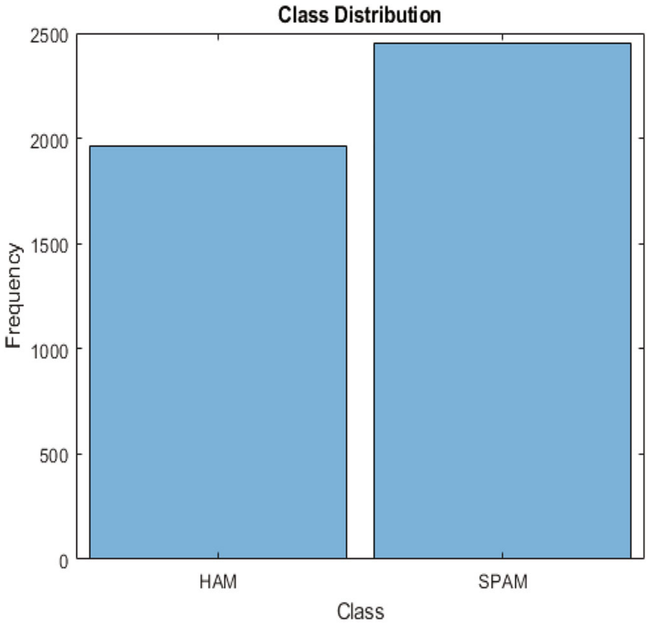


FIGURE 2 Class distribution for ExAIS_SMS dataset

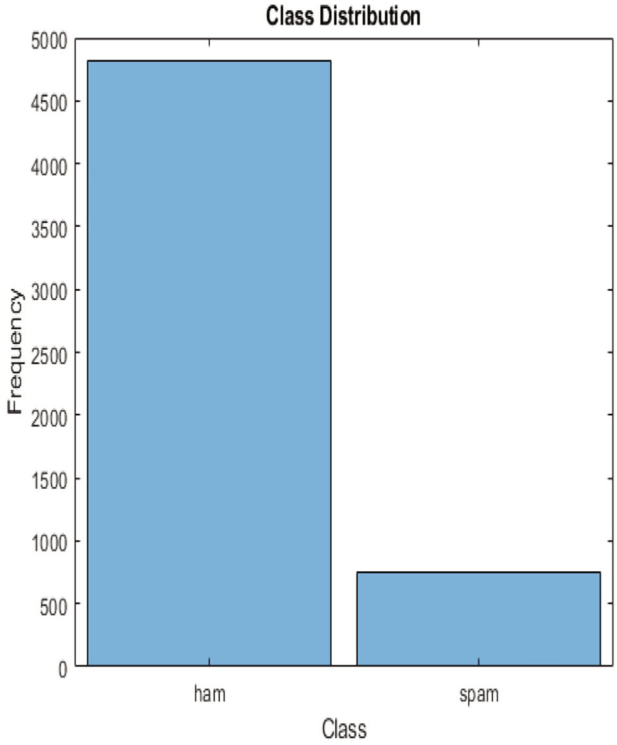
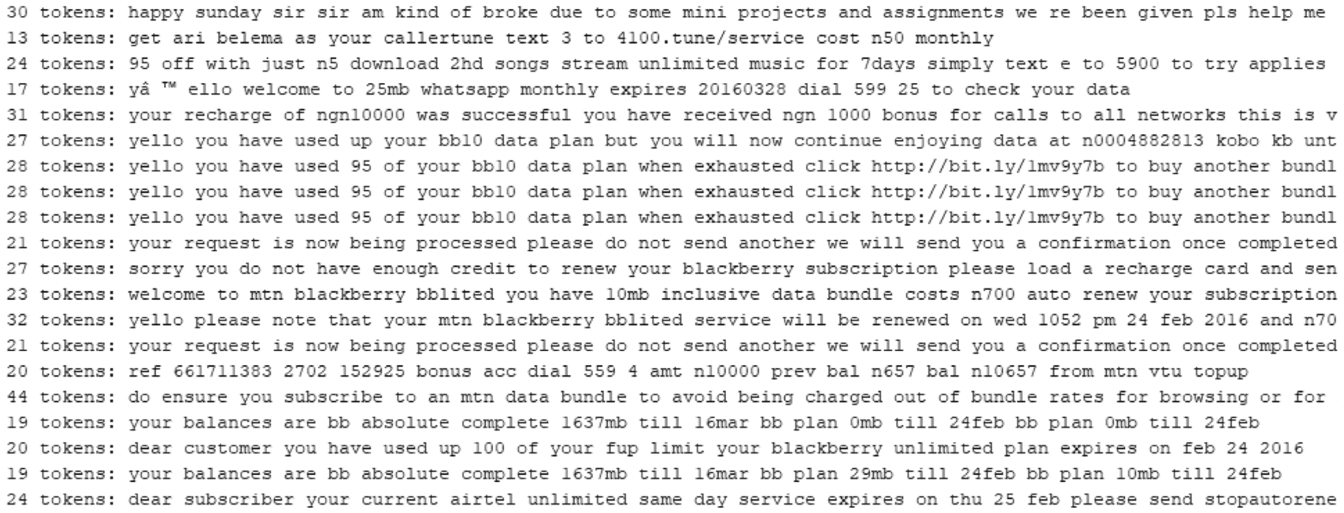


**FIGURE 3** Class distribution for UCI_SMS dataset



```
30 tokens: happy sunday sir sir am kind of broke due to some mini projects and assignments we re been given pls help me
13 tokens: get ari belema as your callertune text 3 to 4100.tune/service cost n50 monthly
24 tokens: 95 off with just n5 download 2hd songs stream unlimited music for 7days simply text e to 5900 to try applies
17 tokens: yâ ™ ello welcome to 25mb whatsapp monthly expires 20160328 dial 599 25 to check your data
31 tokens: your recharge of ngn10000 was successful you have received ngn 1000 bonus for calls to all networks this is v
27 tokens: yello you have used up your bb10 data plan but you will now continue enjoying data at n0004882813 kobo kb unt
28 tokens: yello you have used 95 of your bb10 data plan when exhausted click http://bit.ly/1mv9y7b to buy another bundl
28 tokens: yello you have used 95 of your bb10 data plan when exhausted click http://bit.ly/1mv9y7b to buy another bundl
28 tokens: yello you have used 95 of your bb10 data plan when exhausted click http://bit.ly/1mv9y7b to buy another bundl
21 tokens: your request is now being processed please do not send another we will send you a confirmation once completed
27 tokens: sorry you do not have enough credit to renew your blackberry subscription please load a recharge card and sen
23 tokens: welcome to mtn blackberry bblited you have 10mb inclusive data bundle costs n700 auto renew your subscription
32 tokens: yello please note that your mtn blackberry bblited service will be renewed on wed 1052 pm 24 feb 2016 and n70
21 tokens: your request is now being processed please do not send another we will send you a confirmation once completed
20 tokens: ref 661711383 2702 152925 bonus acc dial 559 4 amt n10000 prev bal n657 bal n10657 from mtn vtu topup
44 tokens: do ensure you subscribe to an mtn data bundle to avoid being charged out of bundle rates for browsing or for
19 tokens: your balances are bb absolute complete 1637mb till 16mar bb plan 0mb till 24feb bb plan 0mb till 24feb
20 tokens: dear customer you have used up 100 of your fup limit your blackberry unlimited plan expires on feb 24 2016
19 tokens: your balances are bb absolute complete 1637mb till 16mar bb plan 29mb till 24feb bb plan 10mb till 24feb
24 tokens: dear subscriber your current airtel unlimited same day service expires on thu 25 feb please send stopautorene
```

**FIGURE 4** Screenshot of a few preprocessed training documents

ii. Removal of the header, time, and date of each SMS message: second, we removed some features such as header, time stamp, and date from the corpus.

iii. Eliminating all punctuation marks, prepositions, short-length words less than or equals to two (<=) alphabets. To ensure the effectiveness of our model, we carefully removed all unnecessary punctuations such as stopwords, exclamation, short words and so forth. There was a need to make our preprocessing more flexible due to the unstandardized abbreviation mostly used in SMS. A display of the first few preprocessed trainings document and the token breakdown per SMS is depicted in Figure 4.

After the ExAIS_SMS dataset was preprocessed, a total number of 4420 SMS was realized and used for further classification. The 4420 ExSIS_SMS dataset comprises 2453 Spam, and 1967 Ham was finally used for evaluation. For optimal results, each document was converted to sequences using word encoding, and each sequence was applied during training, validation, and testing for effective feature selection. The number of instances used in ExAIS_SMS summed up to 4420 instances, while 5572 instances were evaluated in the UCI dataset.

## 3.3 | Proposed model architecture

Our choice of the BiLSTM model is based on the significant performance in other text classification tasks. This sequence prediction model with the ability to learn data points weights and determine decision boundaries between classes has helped previous research endeavors in the effectively classification of text, especially in sentiment analysis.

The proposed BiLSTM model, as depicted in Figure 5, is a six-layer model which involves the L1-input sequence layer consisting of one dimension input and the L2-word embedding layer consisting of 50 dimensions. The L3-BiLSTM layer is summarized in Table 1, While the L4-the fully connected layer and L5-SoftMax layer and final layer L6- the output layer is used in classifying as either spam or ham.

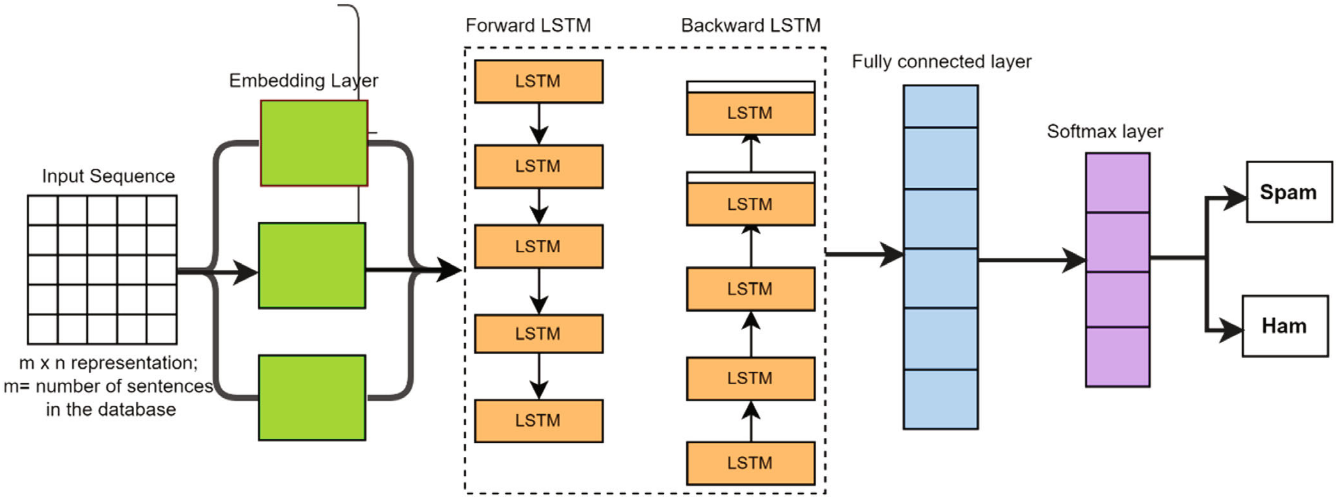


FIGURE 5 The architecture of our BiLSTM model

TABLE 1 Summary of the BiLSTM architecture

| BiLSTM architecture | |
|---|---|
| Sequence input | 1 dimension |
| Number of hidden units | 100 |
| State activation function | tanh |
| Gate activation function | sigmoid |
| Output | 2 |
| Epsilon | 1e-8 |
| **Training parameters** | |
| Optimizer | Adam |
| Gradient decay factor | 0.9 |
| Squared gradient decay factor | 0.999 |
| Max Epoch | 150 |
| Mini Batchsize | 128 |
| LearnRateDropFactor | 1e-2 |
| Initial learning rate (α) | 1e-3 |
| Execution environment | Auto |

For our embedding model, we used the standard/default setting word2vec for training the data. This embedding model is based on the continuous-Bag-of-Words (CBOW) method. Considering the sparsity of our datasets, it was hard for deep learning to learn good features; therefore, the word2vec embedding model was able to represent each word using a low dimension vector of 50 dimensions. The BiLSTM model used in this study enables a lookup method that allows a forward LSTM and progressively processes the sequence, and also a backward LSTM, that operates the sequence in reverse order. Therefore, the output is the interconnection of the corresponding states of both the forward and backward LSTM. The parameter values and training parameters of the designed architecture are depicted in Table 1.

### 3.4 | Experimental setting and evaluation

The proposed BiLSTM model was implemented on MATLAB R2020b (Mathworks, USA) running on Windows 10 64 bits Intel Core i5 CPU and 8 GB RAM. In addition to MATLAB, we also used the WEKA environment for analyzing our ML algorithms. To evaluate the performance of our model and other ML algorithms, we applied the following metrics: precision, recall, F-measure, and accuracy (ACC). The lexical (bag of words) feature was used to develop the classification model for this study.

For the BiLSTM model, our evaluation is divided into two major parts; the first is based on the BiLSTM model and applied the ratio 60:20:20 percentage split training, validation, and testing, respectively. Therefore, the total number of instances utilized for training was 2652, validation was 884, and 884 instances for testing using the ExAIS_SMS dataset. To further validate our work's performance, we applied the same validation split on the UCI SMS dataset to give a total split of 3344 training, 1114 validation, and 1114 testing instances, respectively.

## 4 | RESULT AND DISCUSSION

This section gives detailed results of the ML experiments on the ExAIS_SMS and UCI SMS datasets, respectively. Each of the datasets is partitioned using cross-validation methods of ratio 60:20:20 for training, validation, and testing.

1. Experiment based on ExAIS_SMS dataset
2. Experiment based on UCI dataset

### 4.1 | Experiment results for ExAIS_SMS

The comparison result for the carried out with the best performance result obtained for six classifiers and the results obtained are presented in Table 2. In order to effectively optimize the performance of the BiLSTM model from the initial ground-truth classification rate of 90.8, we finetuned some of the parameters and after several adjustments, our model was able to achieve a significant improvement from an initial 90.8% to 93.4% accuracy. The overall summary of results obtained using the ExAIS_SMS datasets on the seven algorithms is presented in Table 2.

Table 2 shows the result for the best five ML algorithms in comparison BiLSTM model. The performance results are very impressive, and each classifier is evaluated based on accuracy, precision, recall, and F-measure. BiLSTM achieved the best results with an accuracy of 93.4%, precision 96.9, recall 91.7%, and F-measure 94.23%, respectively. The second best is obtained from Naïve Bayes with an accuracy of 84.4%, 84.6% precision, 84.4% recall, and 84.3% F-measure. The screenshot of the training progress for BiLSTM is depicted in Figure 6, and Figure 7 shows the confusion matrix obtained during testing.

**TABLE 2** Experimental results for ExAIS_SMS (*best result in bold)

| S/N | Classifier | ACC (%) | Precision (%) | Recall (%) | F-measure (%) |
|---|---|---|---|---|---|
| 1 | BayesNet | 79.1 | 81.2 | 79.1 | 78.3 |
| 2 | NB | 84.4 | 84.6 | 84.4 | 84.3 |
| 3 | Decision table | 71.09 | 73.1 | 71.1 | 69.5 |
| 4 | C4.5 | 60.65 | 71.3 | 60.7 | 57.7 |
| 5 | J48 | 73.63 | 78.8 | 73.6 | 71.4 |
| 6 | BiLSTM | **93.4** | **96.9** | **91.7** | **94.23** |

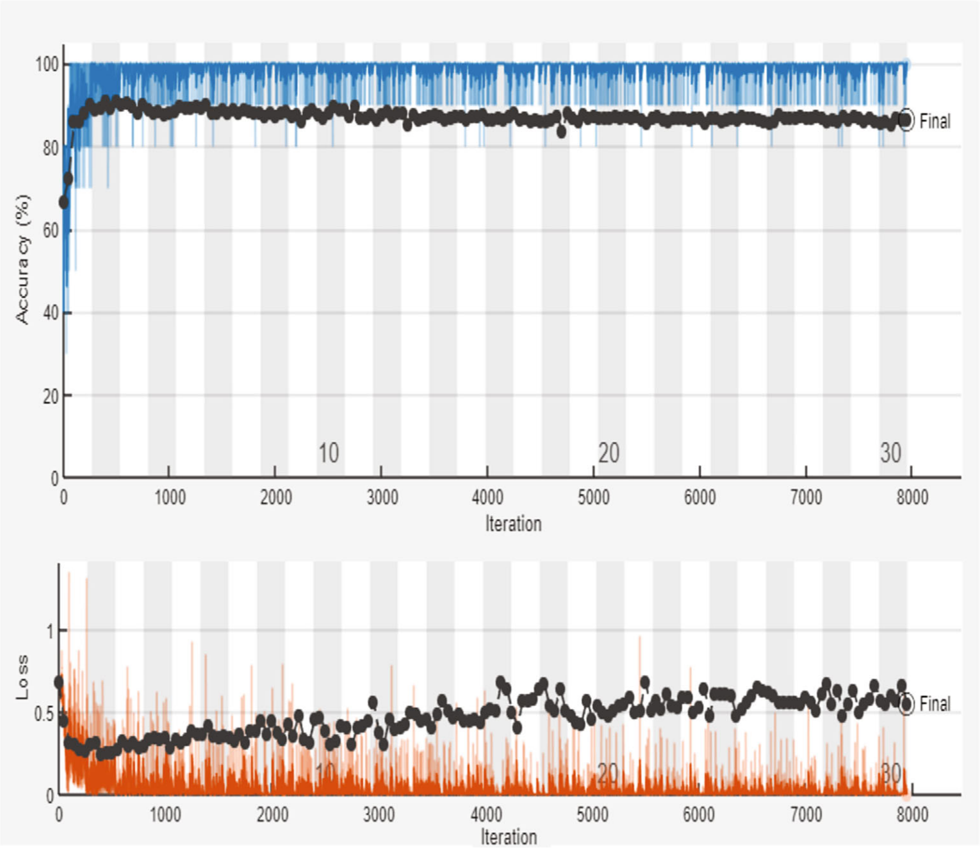
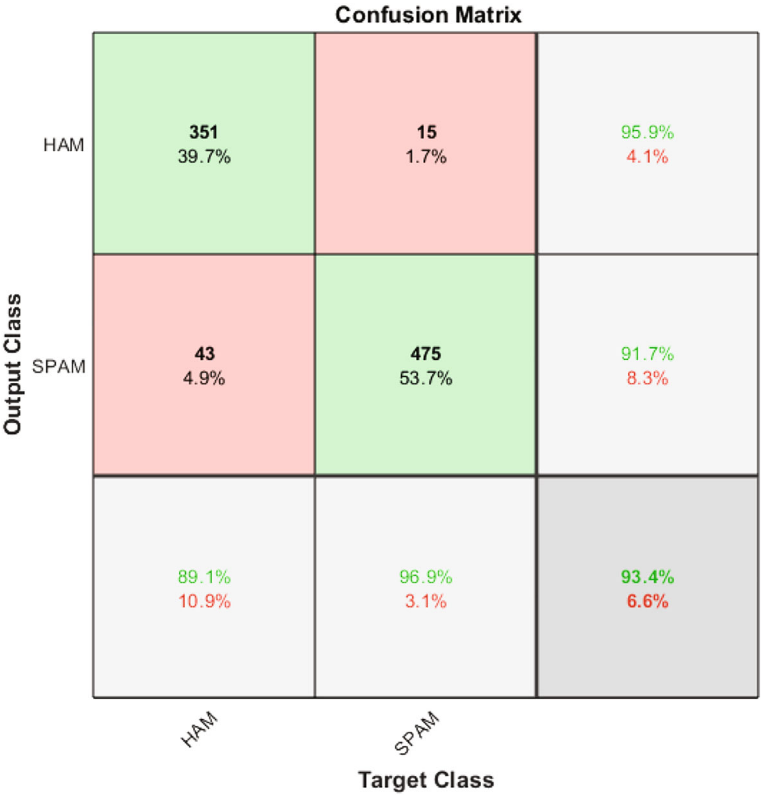


FIGURE 6 Screenshot of training progress for ExAIS_SMS dataset


Confusion Matrix

| Output Class | HAM | SPAM | |
|---|---|---|---|
| HAM | 351<br>39.7% | 15<br>1.7% | 95.9%<br>4.1% |
| SPAM | 43<br>4.9% | 475<br>53.7% | 91.7%<br>8.3% |
| | 89.1%<br>10.9% | 96.9%<br>3.1% | 93.4%<br>6.6% |

Target Class

FIGURE 7 Confusion matrix for BiLSTM using ExAIS_SMS dataset

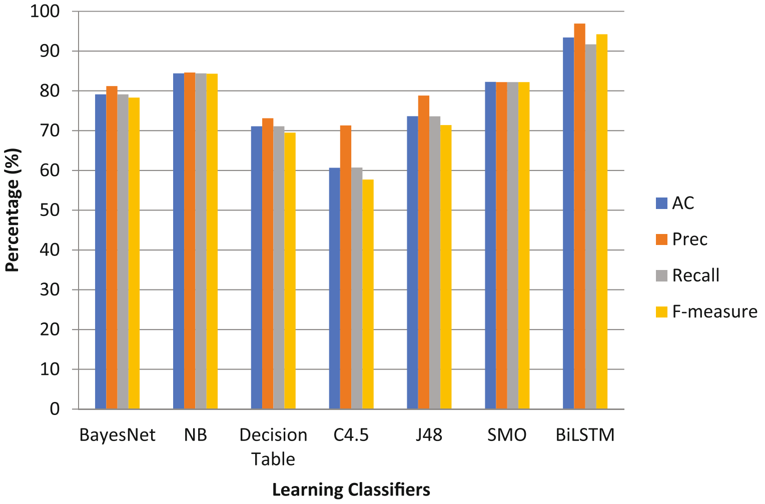


FIGURE 8 Summary of comparison with traditional machine learning (ExAIS_SMS)

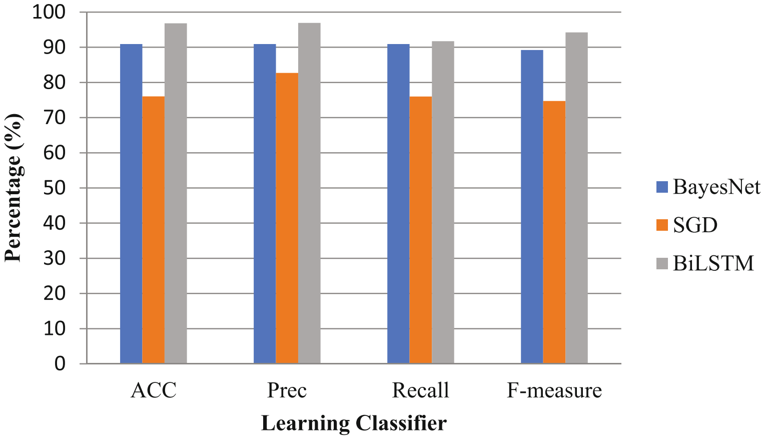


FIGURE 9 Summary of comparison with BiLSTM and traditional machine learning UCI dataset

From Figure 8, this article shows a great improvement in classification rate, but the SMO classifier is computationally intensive with respect to training time of 2714.64 s when compared with others.

## 4.2 | Experiment results for UCI_SMS

To further validate our proposed model, we applied the popular UCI SMS dataset and the summary of our experimental results. The time taken to compute each classifier is represented in Figure 9.

The graph from Figure 9 shows that our BiLSTM model still outperforms traditional ML classifiers like SGD and BayesNet with accuracy, precision, recall, and F-measure of 96.8%, 96.9%, 91.7%, and 94.23%, respectively. The screenshot for the training progress of BiLSTM on UCI data set is depicted in Figure 10, and the confusion matrix for summarized analysis is presented in Figure 11.

## 4.3 | Comparison results based on UCI dataset

We compared our proposed BiLSTM model with experimental results of some recently published literature based on the UCI dataset of 5572 or 5574 SMS spam. Interestingly, our model performed well on our indigenous dataset and showed an impressive performance compared to existing studies, as summarized in Table 3.

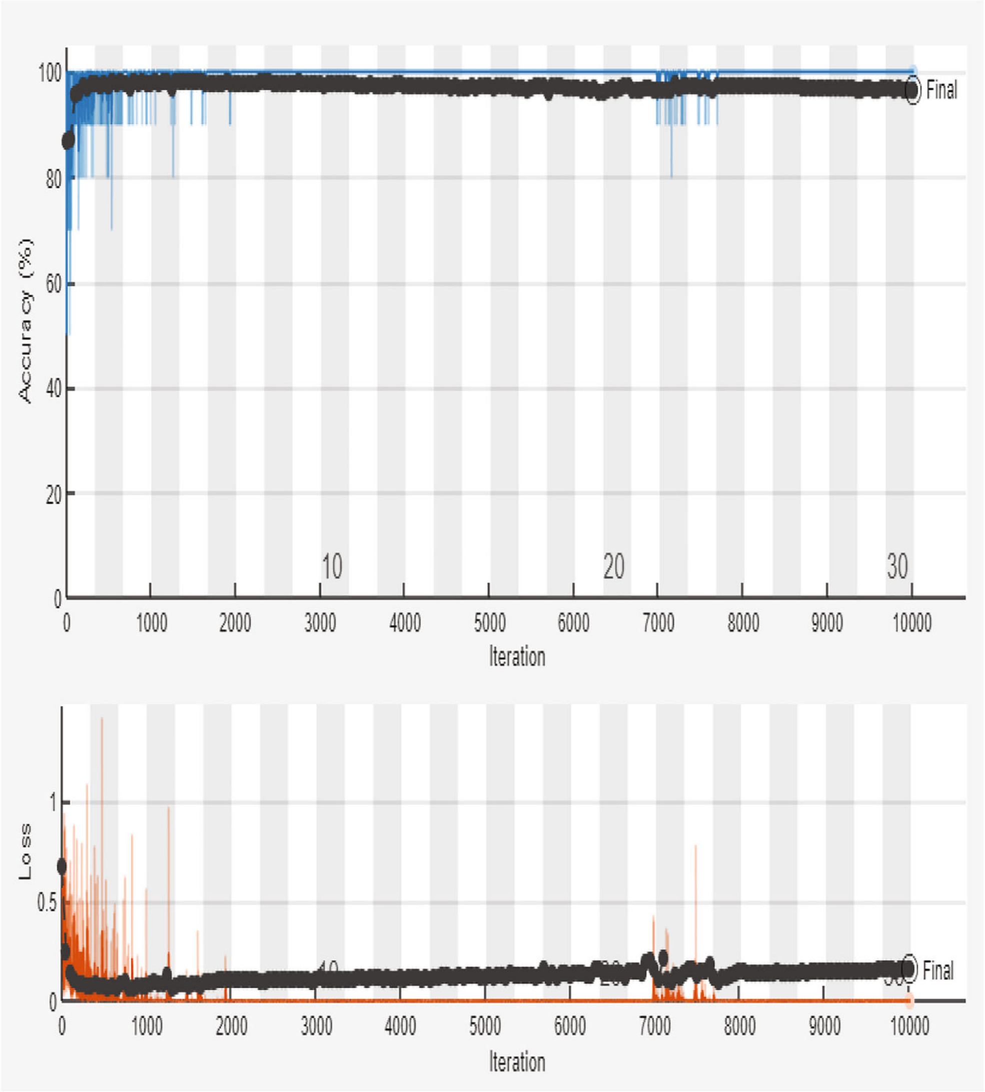


FIGURE 10 Screenshot of training progress for UCI dataset

## 5 | CONCLUSION AND RECOMMENDATION

This article proposes a deep learning method based on a RNN (BiLSTM model). The study was carried out on two datasets; an indigenous dataset referred to ExAIS_SMS and the popularly known UCI dataset. In the proposed method, according to the significant properties of our proposed bi-directional long short-term memory, the ExAIS_SMS and UCI SMS dataset during the experiments were able to obtain a significant classification rate in the detection of spam SMS.

The experiment was carried out using cross-validation methods at ratio 60: 20:20 for train, validation, and test, respectively, for each of the datasets. The overall performance result gave a consistent result for seven classifiers: BiLSTM, Naïve Bayes, BayesNet, SVM, K-nearest neighbor, J48, and decision tree. Our proposed BiLSTM model outperformed the traditional ML classifier for both ExAIS_SMS and UCI datasets. The results of BiLSTM on the two datasets have been consistently promising results with accuracy for the ExAIS_SMS database is 93.4%, and performance accuracy for UCI dataset was 96.8%. The experimental results obtained for the six ML classifiers used in this article are as follows: Naive Bayes, BayesNet, SOM, decision tree, C4.5, J48 with 89.64%, 91.11%, 88.24%, 75.76%, 80.24%, and 79.2%, respectively for ExAIS_SMS datasets respectively. While the results using UCI dataset are as follows: 90.92% BayesNet, SGD, 76.02, respectively.

By comparing our experimental results with other state-of-the-art ML classifiers, it was found that BiLSTM achieved a considerable improvement in classification accuracy and has less computational time with respect to other classifiers. Therefore, it is possible to say that the proposed BiLSTM method is suitable for deployment at either the client (mobile app) or server-side (network provider). One of the shortfalls of our model is the problem of limited datasets, and preprocessing time was very high as a result of the manual removal of unstandardized abbreviations. In our

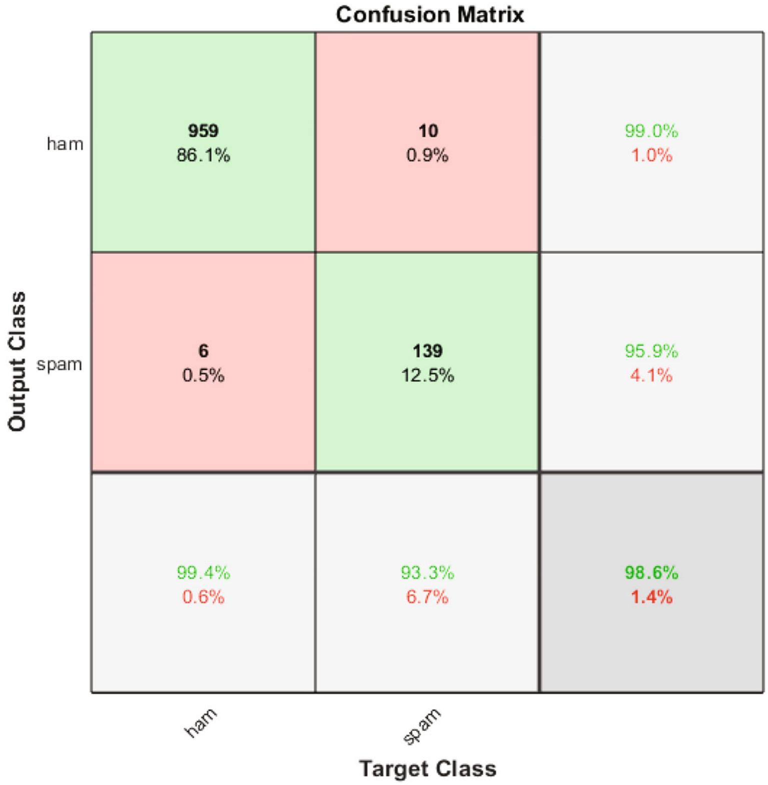


FIGURE 11 Confusion matrix for UCI dataset

TABLE 3 Comparison with existing studies on the UCI SMS dataset

| Method | Accuracy (%) | Reference |
|---|---|---|
| TF-IDF and random forest | 97.5 | Sjarif et al.[32] |
| Cost sensitive classifiers and Bayesian network | 99.8 | Lim and Singh[33] |
| Danger theory + Krill herd optimization model | 96 | Sharaff et al.[34] |
| Neural network | 97.67 | Alzahrani and Rawat[36] |
| CNN+ LSTM | 98.5 | Al Bataineh and Kaur[47] |
| LSTM | 98.5 | Gadde et al.[46] |
| BiLSTM | 98.6 | Our proposed model |

future work, we will be focusing on applying some practical and less computational data augmentation methods to solve the class imbalance problem by generating synthetic SMS dataset for improved classification performance.

## DATA AVAILABILITY STATEMENT

Data used in this work is made available at- https://github.com/AbayomiAlli/SMS-Spam-Dataset

## ORCID

*Olusola Abayomi-Alli* https://orcid.org/0000-0003-2513-5318
*Sanjay Misra* https://orcid.org/0000-0002-3556-9331
*Adebayo Abayomi-Alli* https://orcid.org/0000-0002-3875-1606

## REFERENCES

1. Wilmer HH, Sherman LE, Chein JM. Smartphones and cognition: a review of research exploring the links between mobile technology habits and cognitive functioning. *Front Psychol*. 2017;8:605.

2. Al-Hasan AA, El-Alfy ESM. Dendritic cell- algorithm for mobile phone pam filtering. Proceedings International Conferences on Ambient Systems, Networks and Technologies (ANT); 2015:244-251; Elsevier B.V.
3. Chen L, Yan Z, Zhang W, Kantola R. TruSMS: a trustworthy SMS spam control system based on trust management. *Future Gener Comput Syst*. 2014;49:77-93.
4. Cloudmark. SMS spam overview; 2013. Accessed 13 February, 2014. https://www.cloudmark.com/en/s/.../whitepapers/sms-spam-overview
5. Botha J, Vant WC, Leenen L. A comparison of chat applications in terms of security and privacy. Proceedings of the 18th European Conference on Cyber Warfare and Security; July 2019:55.
6. Yoon, J. W., Hyoungshick, K. and Huh, J. H. (2010). Hybrid spam filtering for mobile communication, J Comput Secur, 29(4), pp. 446–459.
7. Shirani-Mehr H. SMS spam detection using machine learning approach. CS229 Project 2013:1-4; Standford University.
8. Kondamudi M. Classifying and predicting spam messages using text mining in SAS® Enterprise miner™. South Central SAS Users Group (SCSUG 2017); 2017. https://www.sas.com/content/dam/SAS/support/en/sas-global.../2650-2018.pdf
9. Sheikhi S, Kheirabadi MT, Bazzazi A. An effective model for SMS spam detection using content-based features and averaged neural network. *Int J Eng*. 2020;33(2):221-228.
10. Graham P. APlan for spam; 2002. http://www.paulgraham.com/spam.html
11. Zeltsan Z. General overview of spam and technical measures to mitigate the problem.ITU-T SG 17. Proceedings of the Interim Rapporteur Meeting; November, 2004. http://www.docstoc.com/docs/3731634/business-proposal-letters
12. Chaminda DT, Amarasinghe HK, Jayakody JM. Content-based hybrid SMS spam filtering system. Proceedings of ITRU Research Symposium; 2013:31-35; Sri Lanka, University of Moratuwa.
13. Delany SJ, Buckley M, Greene D. SMS spam filtering: methods and data. *Expert Syst Appl*. 2012;39:9899-9908.
14. Aragao MV, Frigieri EP, Ynoguti CA, Paiva AP. Factorial design analysis applied to the performance of SMS anti-spam filtering systems. *Expert Syst Appl*. 2016;64:589-604.
15. Bin Z, Gang Z, Yunbo F, et al. Behavior analysis based SMS spammer detection in mobile communication networks. International Conference on Data Science in Cyberspace (DSC); 2016:538-543; IEEE.
16. Jang B, Kim M, Harerimana G, Kang SU, Kim JW. Bi-LSTM model to increase accuracy in text classification: combining Word2vec CNN and attention mechanism. *Appl Sci*. 2020;10(17):5841.
17. Abayomi-Alli O, Misra S, Abayomi-Alli A, Odusami M. A review of soft techniques for SMS spam classification: methods, approaches and applications. *Eng Appl Artif Intel*. 2019;86:197-212.
18. Yu Y, Si X, Hu C, Zhang J. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput*. 2019;31(7):1235-1270.
19. Domingos P. A few useful things to know about machine learning. *J Commun ACM*. 2015;55:78-87.
20. Manyika J, Chui M, Brown B, Bughin J, Dobbs R, Roxburgh C, Byers A. Big Data: the Next Frontier for Innovation, Competition, and Productivity. Technical report, McKinsey Global Institute; 2011.
21. Nilsson NJ. *Introduction to Machine Learning. An Early Draft of a Proposed Textbook*. Robotics Laboratory, Department of Computer Science, Standford University; 1998.
22. Smola A. and Vishwanathan S.V.N. (2008). *Introduction to Machine Learning*. Publisher Cambridge University Press, https://homes.cs.washington.edu/~pedrod/papers/cacm12.pdf
23. Bandhakavi A, Wiratunga N, Deepak P, Massie S. Lexicon based feature extraction for emotion text classification. *Pattern Recognit Lett*. 2016;93:133-142.
24. Learned-Miller EG. *Introduction to Supervised Learning. I: Department of Computer Science*. University of Massachusetts; 2014.
25. Triguero I, García S, Herrera F. Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. *Knowl Inf Syst*. 2015;42(2):245-284.
26. Zhu X. Semi-supervised learning tutorial. Proceedings of the International Conference on Machine Learning (ICML); June 2007:1-35.
27. Kaur H. Enhancing the spam detection techniques using naive Bayes classifier algorithm. *Int J Modern Trends Eng Res*. 2015;2(7):157-118.
28. Vishwanathan SVN, Narasimha Murty M. Neural networks, 2002. IJCNN '02. Proceedings of the International Joint Conference on Neural Networks;Vol 3, 2002:2393-2398; Honolulu, HI.
29. Yuriy C. Support vector machine classifier. Code project; 2018. Accessed June 26, 2018. https://www.codeproject.com/Articles/25215/Support-Vector-Machine-Classifier
30. Bzdok D, Krzywinski M, Altman N. Machine learning: supervised methods, SVM and kNN. *Nature Methods*. Nature Publishing Group; 2018, 2018:1-6.
31. Rao S, Verma AK, Bhatia T. A review on social spam detection: challenges, open issues, and future directions. *Expert Syst Appl*. 2021;186:115742.
32. Sjarif NNA, Azmi NFM, Chuprat S, Sarkan HM, Yahya Y, Sam SM. SMS spam message detection using term frequency-inverse document frequency and random forest algorithm. *Proc Comput Sci*. 2019;161:509-515.
33. Lim LP, Singh MM. Resolving the imbalance issue in short messaging service spam dataset using cost-sensitive techniques. *J Inf Secur Appl*. 2020;54:102558.
34. Sharaff A, Kamal C, Porwal S, Bhatia S, Kaur K, Hassan MM. Spam message detection using danger theory and krill herd optimization. *Comput Netw*. 2021;199:108453.
35. Bosaeed S, Katib I, Mehmood R. A fog-augmented machine learning based SMS spam detection and classification system. Proceedings of the 2020 5th International Conference on Fog and Mobile Edge Computing (FMEC); April 2020:325-330; IEEE.
36. Alzahrani A, Rawat DB. Comparative study of machine learning algorithms for SMS spam detection. Proceedings of the 2019 SoutheastCon; 2019:1-6; IEEE. Doi: 10.1109/SoutheastCon42311.2019.9020530
37. Theodorus A, Prasetyo TK, Hartono R, Suhartono D. Short message service (SMS) spam filtering using machine learning in Bahasa Indonesia. Proceedings of the 2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT); April 2021:199-203; IEEE.
38. Adel H, Bayati MA. Building bi-lingual anti-spam SMS filter. *Int J New Technol Res (IJNTR)*. 2018;4(1):94-98.
39. Bao LQ, LV LX, Li JL. Optimizing Naïve Bayes algorithm for SMS spam filtering on Mobile phone to reduce the consumption of resources. *J Comput*. 2017;28(3):174-183.
40. Sethi G, Bhootna V. SMS Spam filtering application using Android. *Int J Comput Sci Inf Technol*. 2014;5(3):1424-1426.

41. Anchal, Sharma A. SMS spam detection using neural network classifier. *Int J Adv Res Comput Sci Softw Eng (IJARCSSE)*. 2014;4(6):240-244. https://www.semanticscholar.org/paper/SMS-Spam-Detection-Using-Neural-Network-Classifier-Anchal-Sharma/0ab3a45d40a1e084ca42dff10bec71949d2915b8
42. Suleiman D, Al-Naymat G. SMS spam detection using H2O framework. Proceedings of the 8th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2017). Procedia Computer Science; Vol. 113, 2017:154-161.
43. Sharma D, Sharaff A. Identifying spam patterns in SMS using genetic programming approach. Proceedings of the 2019 International Conference on Intelligent Computing and Control Systems (ICCS); May 2019:396-400; IEEE, India. Doi: 10.1109/ICCS45141.2019.9065686
44. Onashoga A, Abayomi-Alli O, Sodiya A, Ojo D. An adaptive and collaborative server- side SMS spam filtering scheme using artificial immune system. *Inf Secur J Global Perspect*. 2015;24(4–6):133-145.
45. Mahmoud, T. M. and Mahfouz, A. M. (2012). SMS spam filtering techniques based on artificial immune system. IJCSI *Int J Comput Sci Issues*, 9 (2):1, pp. 589–597.
46. Gadde S, Lakshmanarao A, Satyanarayana S. SMS spam detection using machine learning and deep learning techniques. Proceedings of the 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS); Vol. 1, 2021:358-362; IEEE.
47. Al-Bataineh A, Kaur D. Immunocomputing-based approach for optimizing the topologies of LSTM networks. *IEEE Access*. 2021;9:78993-79004. doi:10.1109/ACCESS.2021.3084131
48. Roy PK, Singh JP, Banerjee S. Deep learning to filter SMS spam. *Future Gener Comput Syst*. 2020;102(2020):524-533. doi:10.1016/j.future.2019.09.001
49. Xia T, Chen X. A weighted feature enhanced hidden Markov model for spam SMS filtering. *Neurocomputing*. 2021;444(2021):48-58. doi:10.1016/j.neucom.2021.02.075
50. Wei F, Nguyen T. A lightweight deep neural model for SMS spam detection. Proceedikngs of the 2020 International Symposium on Networks, Computers and Communications (ISNCC); October 2020:1-6; IEEE.
51. Annareddy S, Tammina S. A comparative study of deep learning methods for spam detection. Proceedings of the 2019 3rd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC); December 2019:66-72; IEEE.
52. Huang T. A CNN model for SMS spam detection. Proceedings of the 2019 4th International Conference on Mechanical, Control and Computer Engineering (ICMCCE); October 2019:851; IEEE.
53. Chandra A, Khatri SK. Spam SMS filtering using recurrent neural network and long short term memory. Proceedings of the 2019 4th International Conference on Information Systems and Computer Networks (ISCON); November 2019:118-122; IEEE.
54. Lee HY, Kang SS. Word embedding method of sms messages for spam message filtering. Proceedings of the 2019 IEEE International Conference on Big Data and Smart Computing (BigComp); 2019:1-4; IEEE.
55. Baaqeel H, Zagrouba R. Hybrid SMS spam filtering system using machine learning. Techniques. Proceedings of the 2020 21st International Arab Conference on Information Technology (ACIT); November 2020:1-8; IEEE.
56. Dewi DAC, Asror I. Analysis and implementation of cross lingual short message service spam filtering using graph-based k-nearest neighbor. *J Phys Conf Ser*. 2018;971(1):012042.
57. Ishtiaq A, Islam MA, Iqbal MA, Aleem M, Ahmed U. Graph centralitybased spam sms detection. Proceedings of the 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST); 2019:629-633; IEEE.
58. Wadhawam A, Negi N. Novel approach for generating rules for SMS spam filtering using rough sets. *Int J Sci Technol Res*. 2014;3(7):80-86.
59. Androulidakis I, Vlachos V, Papanikolaou A. FIMESS: filtering mobile external SMS spam. Proceedings of the 6th Balkan Conference in Informatics (BCI '13); 2013:221-227; ACM, New York, NY. DOI: 10.1145/2490257.2490288
60. Abayomi-Alli O. ExAIS SMS dataset; 2021. Accessed March 09, 2021. https://github.com/AbayomiAlli/SMS-Spam-Dataset
61. Almeida TA, Silva TP, Santos I, Hidalgo JMG. Text normalization and semantic indexing to enhance instant messaging and SMS spam filtering. *Knowl Based Syst*. 2016;108:25-32.

**How to cite this article:** Abayomi-Alli O, Misra S, Abayomi-Alli A. A deep learning method for automatic SMS spam classification: Performance of learning algorithms on indigenous dataset. *Concurrency Computat Pract Exper*. 2022;34(17):e6989. doi: 10.1002/cpe.6989