World Journal of Advanced Engineering Technology and Sciences
eISSN: 2582-8266
Cross Ref DOI: 10.30574/wjaets
Journal homepage: https://wjaets.com/

(RESEARCH ARTICLE)

Check for updates

A comparative analysis of recurrent neural network and support vector machine for binary classification of spam short message service

David Odera [1,*] and Gloria Odiaga [2]

[1] *Tom Mboya University, Homabay-Kenya.*
[2] *Jaramogi Oginga Odinga University of Science and Technology, Bondo-Kenya.*

World Journal of Advanced Engineering Technology and Sciences, 2023, 09(01), 127–152

Publication history: Received on 09 April 2023; revised on 26 May 2023; accepted on 28 May 2023

Article DOI: https://doi.org/10.30574/wjaets.2023.9.1.0142

Abstract

Over the years, communication through Short Message Service (SMS) has been a primary tool for mobile subscribers. SMS has varied applications in health, industry, finances, education and social networking among others. The growth of mobile devices and SMS usage has consequently increased the attack surface for cyber-criminals culminating to the proliferation of malicious activities introduced using SMS spam, phishing, spyware, malware etc. Ham messages are normal messages people trade with one another and are usually not unwanted by the recipient, while spam messages are unsolicited junk and redundant messages that may be sent to a large number of people at once and are usually unwanted. Various spam detection models have been developed using various traditional machine and deep learning techniques. However, most studies where comparison between deep and traditional machine learning algorithms is done, have unfortunately omitted K-Nearest Neighbors and Support Vector Machine (SVM) which are empirically deemed as the most popular traditional machine learning algorithms. In this study, therefore, we develop a deep learning model based on Recurrent Neural Network (RNN) for Spam and Ham SMS classification and compare its performance against SVM model for the same University of California (UCl) SMS dataset. The results show that RNN has a slightly higher training and validation accuracy of 0.98 compared to SVM at 0.94, however, the false positive rate of SVM is marginally lower. Exploring application of deep learning with better optimization algorithms such as RNN improves accuracy, reduces computational complexities i.e. memory consumption and speed, and thus minimizing false positive rates. For future work, we suggest the use of varied performance metrics to validate the model in a distributed dataset environment.

Keywords: Spam; Ham; Recurrent Neural Networks; Support Vector Machine; Comparison; Short Message Service

1. Introduction

Critical infrastructures are essential for the economy, society and governments [1]-[5]. The communication sector is one of the critical infrastructures essential for the economic, social and financial stability of any country [6]. Communication over the Internet is significant to all users that operate on the cyberspace [7]. The Internet of Things (IoT) main aim is to connect everything within a common infrastructure to enable control and updates of devices from anywhere and at any time [8]-[13]. Critical infrastructure components are vulnerable to varied threats including, natural disasters, terrorism, or cybercrime among others [6].Organizations that require Internet for communication are vulnerable to cyber-attacks leading to the exploitation of cybercrime risks. Basically, the interconnected nature of the internet and the increasing dependence on digital systems make organizations potential targets for malicious actors seeking to exploit vulnerabilities for financial gain, data theft, or disruption of operations [14]-[17]. Table 1 presents some of the cybercrime risks faced by organizations.

*Corresponding author: David Odera

Table 1 Organizational cybercrime risks

Risk	Description
Data Breaches	Cybercriminals may attempt to breach an organization's network security to gain unauthorized access [18] to sensitive information such as customer data, employee records, or intellectual property. This stolen data can be used for identity theft, financial fraud, or sold on the dark web.
Ransomware Attacks	Ransomware is a type of malware that encrypts an organization's data and demands a ransom in exchange for the decryption key. These attacks can cripple operations and result in significant financial losses if organizations are unable or unwilling to pay the ransom [19]-[23].
Phishing and Social Engineering	Cybercriminals often employ deceptive tactics, such as phishing emails or phone calls, to trick employees into revealing sensitive information like login credentials or financial details [24] [25], [26], [27]. This information can be used to gain unauthorized access to systems or carry out fraudulent activities.
Distributed Denial of Service (DDoS) Attacks	DDoS attacks involve overwhelming a targeted organization's network or website with a flood of traffic, rendering it inaccessible to users [28]-[31]. This can disrupt operations, cause financial losses, and damage an organization's reputation.
Insider Threats	Employees or contractors with malicious intent can pose a significant cybercrime risk. They may misuse their access privileges to steal or leak sensitive information, disrupt operations, or introduce malware into the organization's systems [32]-[33].

This prompts the risks and security control solutions for those entities operating in the global market space [34]. Building secure communication infrastructure requires trusted communication components that are resilient to threats, vulnerabilities and risks in cyber-security. Some key considerations for establishing a secure communication infrastructure are presented in Table 2.

Table 2 Key considerations for secure communication infrastructure

Key considerations	Description
Encryption	Utilize strong encryption protocols to protect the confidentiality and integrity of communication channels. This ensures that data transmitted between parties remains secure and unreadable to unauthorized individuals [35], [36], [37], [38], [39].
Secure Protocols	Implement secure communication protocols such as Transport Layer Security (TLS) for websites, Secure Shell (SSH) for remote access, and Virtual Private Networks (VPNs) for secure remote connections [40], [41], [42], [43]. These protocols provide encryption and authentication mechanisms for secure data transfer.
Access Control	Employ robust access control mechanisms to limit unauthorized access to communication components. This includes implementing strong authentication methods, role-based access controls, and regularly reviewing and updating user privileges [44], [45], [46], [47].
Intrusion Detection and Prevention Systems	Deploy intrusion detection and prevention systems (IDPS) to monitor network traffic, detect potential threats, and prevent unauthorized access or malicious activities [48]-[52]. These systems can provide real-time alerts and automated responses to mitigate cyber threats.
Regular Updates and Patch Management	Keep all communication components up to date with the latest security patches and software updates. Regularly monitor and apply patches to address vulnerabilities that may be discovered in the software or hardware components [53]-[55].
Network Segmentation	Divide the communication infrastructure into separate segments or subnets to limit the potential impact of a security breach [56]-[58]. This helps contain any security incidents and prevents lateral movement of attackers within the network.

Security Audits and Penetration Testing	Conduct regular security audits and penetration testing to identify vulnerabilities and weaknesses in the communication infrastructure. This allows organizations to proactively address security gaps and strengthen their defenses [59]-[65].
Employee Training and Awareness	Educate employees about cybersecurity best practices, including safe browsing habits, recognizing phishing attempts, and maintaining strong passwords [66]-[68]. Regular training and awareness programs can help mitigate human error, which is often exploited by cybercriminals.
Incident Response Plan	Develop and implement an incident response plan to effectively respond to security incidents [69]-[71]. This includes defined procedures for reporting and addressing security breaches, isolating affected systems, and restoring operations while minimizing the impact.
Continuous Monitoring	Implement continuous monitoring and logging of network activity to detect any suspicious behavior or anomalies [72]-[76]. This enables organizations to identify and respond to potential security incidents in a timely manner.

Smart-phones are emerging as versatile devices enabling the user to perform various activities [77]. According to a report by Statista, indicated that smart-phone mobile network subscriptions worldwide was at almost 6.6 billion in 2022, and this is expected to rise beyond 7.8 billion subscriptions by 2028.

The primary communication tool widely used by mobile subscribers is Short Message Service popularly known as SMS [78], [79], [80], [81], [82], [83]. There are also existing SMS and chatting utilities such as WhatsApp, Hangout, Viber, WeChat, etc. with similar and advanced functionalities [84]. The intention and the role of SMS ranges through various areas of life, i.e.health, education, financial, security, career, social networking among others. According to [84]and [85], the growth of mobile devices and SMS communication in support of various facets of life is tremendous but on the flip side it has led to proliferation of malicious activities [86] as well. Text messages or SMS are a part of smart-phones through which attackers can target the users [87]; [85], [88].

Multimedia SMS pose a challenge of malicious content and it is therefore imperative to adopt atechnique that can process languages, images, emoji, and videos. Activities could be introduced using SMS spam, phishing, spyware, malware etc. Spam could be unsolicited messages which may contain viruses, malwares, adverts and un-demanded contents targeting individuals, companies and business organizations [88], [89]. Ham messages are the everyday messages that individuals trade with each other, these are not junk messages, while spam messages can be classified as redundant messages sent to a large number of people at once [83]. The rise of spam messages is based on factors such as accessibility to affordable bulk SMS plans [83]. For instance, a ham message may state "are you available on Thursday?", while a spam message may utilize the expression "free melodies and ringtones" [83]. Unfortunately, these unsolicited messages are increasing at an alarming rate and according to [87], attackers use various communication mechanisms such as SMS phishing to get sensitive information from mobile users [89]- [93]. These subscribers may not require internet connection in order to receive an SMS, making it convenient and efficient for cybercriminals to exploit. There is need for researchers to exploit machine learning-based algorithms for classification, clustering and association to analyze data such as with SMS in order to gain more insight on the protection of communication systems [94]-[99]. This approach will eventually support confidentiality as one of the tenets of information security by developing robust security systems.

1.1. Motivation

In order to understand data patterns and challenges, research in machine-learning techniques is paramount. We therefore desire to indulge in solution of practical problems in order to contribute in research, especially in developing text-based and language-based machine learning algorithms. This paper intends to design and develop a deep learning model based on Recurrent Neural Network (RNN) for Spam and Ham SMS classification and compare its performance against SVM model on the same dataset.

1.2. Contribution

The contributions of this study are as follows:

- Mathematically describe and propose a framework of RNN for Spam SMS classification
- Design a comparative model of RNN and SVM in order to illustrate the process flow from data preprocessing, training, fitting of SMS dataset, compilation and evaluation of results.

- Provide a comparison between RNN and SVM for SMS spam detection

The rest of the paper is organized as follows:Section 2 is the background of the study; Section 3 presents related work; Section 4 is the proposed methodology. Section 5, presents a discussion on the model experimental performance for both RNN and the SVM classifier and finally, Section 6 gives the conclusion of the study.

1.3. Background of Study

A number of supervised algorithms such as Naïve Bayes, Support Vector Machine (SVM), neural networks and regression have been used to develop most SMS spam classifiers [83], [89], [85], [100], [101]. This is perhaps due to the availability of the output column (labelled data) of the SMS dataset making it possible to train classification problems. The authors in [102] designed an artificial model [103] that applies the concept of functional biological neurons in the human brain to process input and produce output as the sum of weighted inputs as shown in the formula below:

$$Z = \sum_{I=0}^{n} w_i \, x_i$$

Like human biological neurons, the computer based artificial neuron accepts inputs x_1, x_2, x_3….x_n, and each input is multiplied by corresponding weights w_1, w_2, w_3,….w_n. The sum of each subsequent product of weighted inputs, gives the resultant sum and is considered as the logit neuron. Sometime the logit is comprised of a constant value called the *bias*. The logit is then expressed as a function, f, to make the desired output y = f (z).

The researchers in [104] describe deep learning as a subsequent derivative of machine learning that applies algorithms, processes the data, and develops abstractions. Deep learning is an emerging technology that drives artificial intelligence (AI) and the processing of big data; deep learning is ubiquitous, and the machine learning algorithms assist in modeling high-level abstract view of data by means of processing layers which encompasses complex structures [104]-[108].

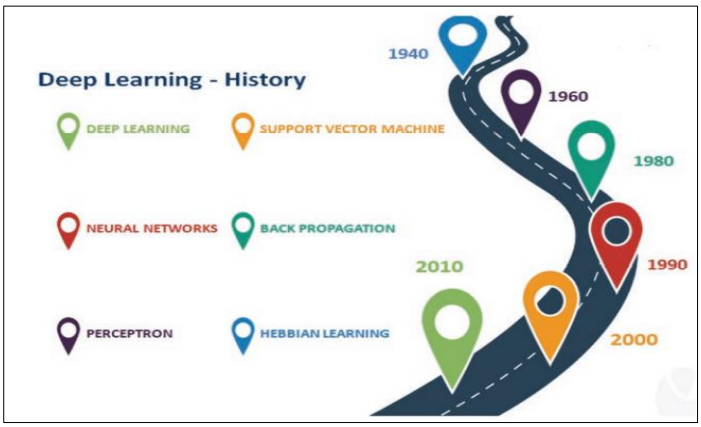


Figure 1 Roadmap of Deep Learning [104]

Researchers in [109] reported that various studies have shown that neural network have been used to understand human beings in a psychological way and therefore in certain occasions it has even outperformed human beings. Some of the researches done using neural network include a deep learning model where the California Renewable Production 2010–2018 dataset, is trained to predict the solar photovoltaic output (California ISO, 2020), deep learning for natural language processing in [110] among others. Researchers in [111] also conducted a research using CNN and extensively used it for image identification, object detection, face detection and classification of images. The technique is applied in OCR (optical character recognition) where text is identified from images.

The authors in [104] discussed in their paper some deep learning architecture applicable in healthcare system such as recurrent neural networks (RNNs), convolutional neural networks (CNNs), auto-encoders (AEs), and restricted Boltzmann machines (RBMs). The major application of deep learning in healthcare system falls on image processing, especially to predict Alzheimer's disease using magnetic resonance imaging (MRI) scans [112], [113].

1.4. Related Work

Machine learning (ML)and deep learning (DL) methods have been preferred by researchers across different disciplines for providing solutions to detection of different categories of attacks [109], [114], [115], [116], [117]. Existing methods on SMS spam classification show that Machine learning (ML), Statistical analysis and evolutionary methods are at 49%, 39%, and 12%, respectively [84]. A number of machine learning algorithms have been used to develop most SMS spam classifiers.

The authors in [115] created a dictionary using the Turn Frequency Inverse Document Frequency (TF-IDF) Vectorizer algorithm, which included all the features of words a spam SMS would possess, based on the content of the message, then referring to this dictionary, the system classifies the SMS as either spam or ham. TF-IDF is used in machine learning (ML) and text mining as a weighting factor for identifying word features [115]. The weight increases as the word frequency in a document increases, however, an offset is also used to differentiate important words from common words (stop words) like 'the' or 'a' that appear often in documents [115]. TF-IDF Vectorizer is therefore used often in relevance ranking and scoring and to remove stop words from ML models. According to [115], ML algorithms [118]-[122] can play a vital role in identifying spam SMS because the accuracy obtained in their study was more than 95% in every try.

Result based on existing approaches shows that 83% is based on a content approach, 5% based on non-content and 12% based on the hybrid [84]. Result analysis for existing anti-spam solutions status shows that Evaluated (E), Implemented (I), Proposed (P), Proposed and Evaluated (PE) and Proposed and Implemented (PI), was 29%, 6%, 20%, 35%, and 10%, respectively [84] and their study, therefore, concludes that majority of existing SMS spam filtering solutions are between the ''Proposed'' status or ''Proposed and Evaluated'' status. Methods such as Random Forest, Dendritic Cell Algorithm, SVM, Naive Bayes, and Artificial Immune system (AIS), show optimal performance result with higher accuracy [84].

According to [123]-[125], SVM is one of the most robust algorithms that solve the problems related to classification, which plots the data items in n-dimensional space, as points where the various features of the given data acts for the given coordinates. SVM is a supervised machine learning algorithm which can be used for both classification and regression challenges, however, it is mostly used in classification problems [114], [126]-[128]. SVM separates the different data groups using the boundaries based on decisions and supports both binary and multi-class classifications; a set of instances having different class values between two groups are separated by using decision boundaries [123]-[125].

Researchers in [83] presented the detection of spam and ham messages using various supervised machine learning algorithms: Naive Bayes, SVM, and maximum entropy and compared their performance in filtering of ham and spam messages. They concluded that building an SMS spam classifier using SVM gives the best results possible with an accuracy of 97.4%. Maximum entropy gave an accuracy of 91.95 % while Naïve Bayes gave an accuracy of 94.55% [83].

Multimedia SMS also pose a security challenge by increasing likelihood for SMS spam through rich media including images, videos and emoji. It is therefore imperative to adopt deep learning techniques such as Recurrent Neural Networks (RNN) which process languages, images, emoji, and videos [84], [129]-[134].

Neural networks have been applied to separate unwanted SMS (spam) messages from normal (ham) messages [129]. Chandra and Khatri (2019) proposed a method utilizing RNN and LSTM using Keras models and Tensorflow backend to detect Spam and Ham from the Spam SMS Collection dataset at University of California (UCI)ML repository and achieved 98% accuracy. The proposed method in [129], that applied RNN SMS spam filtering, achieved prediction accuracy of 98.11%, and indicated a considerable improvement compared to SVM, token-based SVM and Bayesian algorithms with accuracies of 97.81%, 97.64%, and 80.54% respectively. This paper therefore presents a comparative analysis of RNN and SVM in binary classification of spam SMS.

2. Proposed solution

2.1. Challenges with existing works

Google's Bidirectional Encoder Representations from Transformers (BERT) was used in spam detection on 4 different datasets [135], [136] which recorded very promising results of not less than 97% in each case. However, their model did not validate the result against traditional machine learning models [137]-[140]. Machine learning researchers such as the one in [141] and [142] also developed spam detection model using deep learning and LSTM respectively.

Researchers in [143]-[145] have developed deep learning models for spam detection and compared them against various traditional machine learning algorithms, however, there is no empirical evidence on comparison with popular traditional algorithms, such as K-Nearest Neighbor and SVM, and also on how fitting of data on a similar dataset was done. It is also evident that most models do not verify and validate their results against others, so that an evidence-based decision on the choice of algorithm in relation to the type of data is factual. The authors in [141] contributed towards the solution by comparing deep learning and traditional ML algorithms [146], which unfortunately omitted KNN and SVM in the result analysis. According to [83], SVM is poised as best fit for binary classification of SMS spam dataset based on empirical result of 94% accuracy.

In this study, a deep learning model is used for binary classification of SMS data and compared against SVM model in order to validate its performance. Exploring application of deep learning with better optimization algorithms such as RNN will improve accuracy, reduce computational complexities with respect to memory consumption and speed, and thus minimize false positives.

2.2. Methodology

RNN algorithm will be used to train the dataset because it is able to use information from the past and therefore, prediction of high temporal dependencies is possible in the dataset.RNN, which is used mainly in language tasks, is an improved concept after convolutional neural networks (CNN) which is mainly used in image processing [109]. A use case of RNN includes Google mail, where when you type a sentence, it auto completes it; another use case is Google translation, for named entity recognition and sentiment analysis [147], [148]. Tensor Flow comes with RNN models out of the box [149]. Figure 2 below illustrates learning through back propagation technique using the feed forward artificial neural network (ANN) [150], [151], [152].

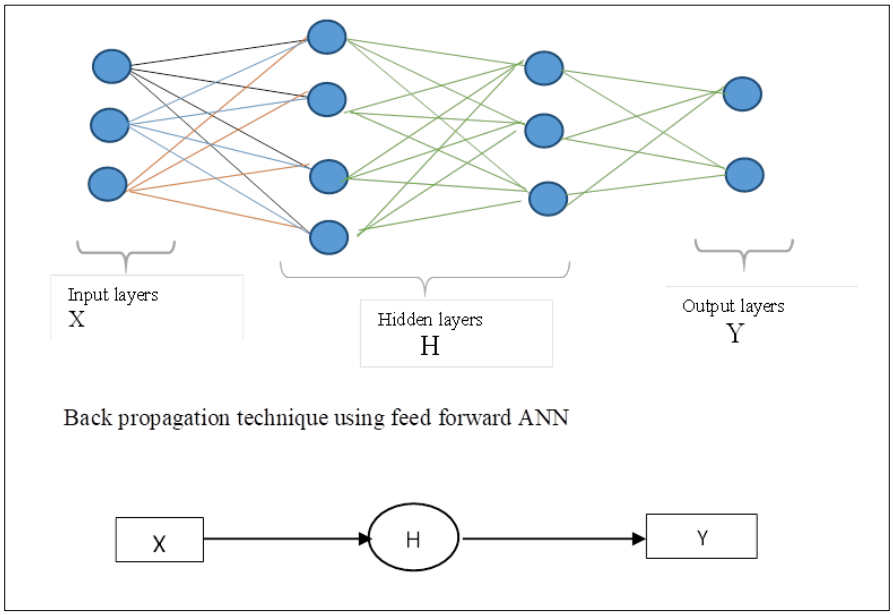


Figure 2 ANN technique

When you multiply input (X) by weight and add bias it goes through different activation functions within the hidden layers (H),then finally you get the output (Y); that is known as the feed forward propagation, which is then followed by back propagation. The reason for back propagation is to modify the weights in order to minimize the error [153]. The error is the square of the difference between the model output and the actual output (known). We differentiate those errors with individual weights [154]-[158].The combination of back-propagation with feed forward multi-layers usually generates finer results [125].

The challenge with ANN is that it does not have the memory concept i.e. there is no connection between the previous input-output and next input-output [147], [159]. Therefore, in order to have dependencies between the data we need to use a recurrent method such as RNN. Feedback loops present in RNNs store information in 'memory' for lengthy phases [160]. This allows for quick classification of a message as spam or ham just by parsing a few initial words [141].

Figure 3 above shows how multiple ANNs have been used to create aRNN, such that for every layer there is an input (X_{t-1}, X_t, X_{t+1} and X_{t+n}), a hidden layer (H_{t-1}, H_t, H_{t+1}, H_{t+n}) and output (Y_{t-1}, Y_t, Y_{t+1}, Y_{t+n}). The hidden layer H_{t-1}passes its output to H_t, which then passes toH_{t+1} in that order. We use gradient descent to find the loss (L) in each layer such that:

$$L = \sum_{t=1}^{T} l^{<t>}$$

To find the change in weight we use:

$$dW = \acute{\eta}\frac{\Delta L}{\Delta w},$$

where $\acute{\eta}$ is the learning rate,

then we update weight as follows:

$$w = w + \Delta w$$

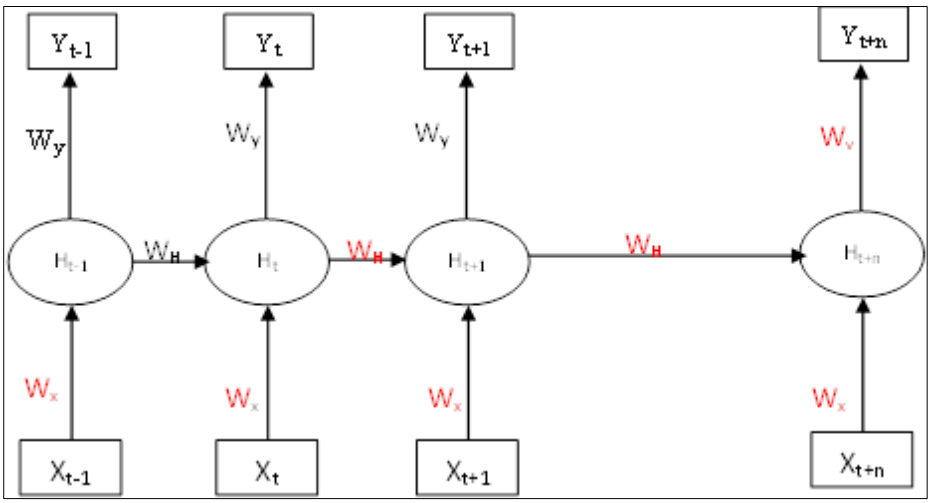


Figure 3 Unfolded Representation of RNN

The change in weight (Δw), could be very small number closer to zero, especially when the RNN structure is made up of very many layers of ANN [161-[164] and this is known as vanishing gradient challenge solved through the Long Short-Term Memory (LSTM) neural network.

2.3. Performance Comparison between RNN and SVM

TensorFlow provides basic building blocks such as fully connected layer, convolutional layer, recurrent neural network module [166-[170], and non-linear activation functions (Developers, 2022). The computation of the gradients of loss will be done on the output in order to accomplish forward pass of the model. Some of the loss functions that TensorFlow uses for computations include, mean squared error and cross-entropy. Minimization of errors is performed using auto-differentiation, which automatically calculates the gradients. To import TensorFlow we use this line of code in google Colab environment: *import tensorflow as tf*.

TensorFlow is an open-source machine learning framework developed by the Google Brain team. It has gained significant popularity and has become one of the most widely used frameworks for building and deploying machine learning models. Table 3 describes some key points about TensorFlow.

TensorFlow's versatility, scalability, and extensive ecosystem make it a powerful framework for various machine learning tasks. Its wide adoption and active community support ensure that it continues to evolve and stay at the forefront of the machine learning and deep learning landscape.

Various software libraries are available, that accelerate research and application of neural network models in solution of various problems. TensorFlow, originally created by researchers at Google, is the most popular among different deep

learning libraries [165]. Also, neural networks are flexible and scalable and thus, have the potential to promote data analysis and modeling applications; however, implementation and optimization algorithms in neural networks are time consuming and prone to errors [165]. TensorFlow is an end-to-end open source platform for machine learning with a wide-ranging set of tools, and libraries for easy building and deployment of machine learning applications. TensorFlow greatly eases and accelerates the research and application of neural network models [165].

Table 3 Key points about TensorFlow

Key concept	Description
Architecture	TensorFlow follows a flexible architecture that allows users to define and execute computational graphs. The core of TensorFlow is based on a data flow graph, where nodes represent mathematical operations and edges represent the flow of data between operations. This graph-based approach enables efficient parallel computation and distributed training across multiple devices or machines.
Deep Learning and Neural Networks	TensorFlow provides extensive support for deep learning tasks, especially for building and training neural networks. It offers a rich set of pre-built operations and modules for constructing various types of neural network architectures, including convolutional neural networks (CNNs), recurrent neural networks (RNNs), and transformer models.
Ecosystem and High-Level APIs	TensorFlow provides a comprehensive ecosystem of tools, libraries, and high-level APIs that simplify the process of developing and deploying machine learning models. The most commonly used high-level API is TensorFlow Keras, which offers a user-friendly interface for defining and training models with minimal boilerplate code.
Model Deployment	TensorFlow offers various options for deploying trained models in production environments. It provides tools for model serialization and serving, allowing models to be deployed as web services or integrated into existing applications. TensorFlow Serving, TensorFlow Lite, and TensorFlow.js are examples of deployment frameworks that cater to different deployment scenarios.
Distributed Computing	TensorFlow supports distributed computing, enabling training and inference across multiple devices or machines. It includes features for distributed training, data parallelism, and model parallelism, allowing users to scale their machine learning workloads to large clusters or specialized hardware such as GPUs or TPUs (Tensor Processing Units).
TensorFlow Extended (TFX)	TFX is a platform built on top of TensorFlow that provides end-to-end machine learning pipeline orchestration. It includes components for data ingestion, data validation, preprocessing, training, evaluation, and deployment. TFX helps streamline the process of developing and maintaining machine learning workflows at scale.
TensorFlow Hub and Model Zoo	TensorFlow Hub is a repository that hosts a wide range of pre-trained models, including both TensorFlow-native models and models from other frameworks. It allows users to easily discover, reuse, and transfer learned representations or entire models for their specific tasks. The TensorFlow Model Zoo offers a curated collection of state-of-the-art models for computer vision, natural language processing, and other domains.
Community and Documentation	TensorFlow has a large and active community of developers and researchers. This vibrant community contributes to the development of new features, provides support through forums and mailing lists, and shares resources, tutorials, and research papers. The official TensorFlow website provides extensive documentation, tutorials, and examples to help users get started and explore different functionalities.

2.4. Materials

The stages of implementation applied in this research include, data collection and pre-processing, rules formulation, libraries importation in the Google Colab platform, model development and training using deep learning (Tensorflow), and evaluation of the developed deep neural network spam detection model.

2.5. Data Collection and Pre-processing

The deep learning spam detection model was encoded, trained and tested by use of UCl SMS spam dataset v.1, that consists of 5574 text messages in English language, categorized as either spam and ham at 747 and 4827 messages respectively [171].

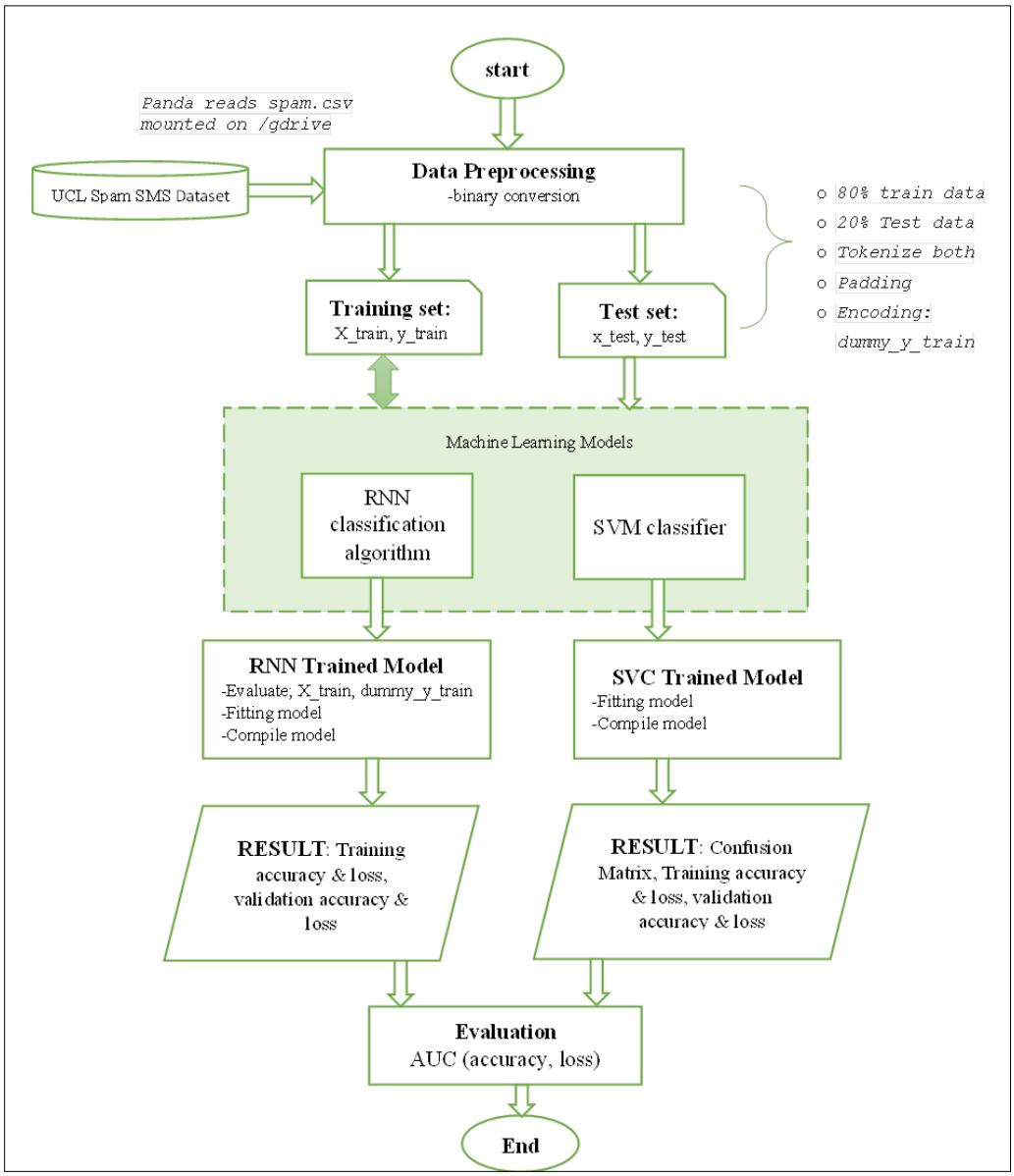


Figure 4 RNN and SVM evaluation model design

The dataset was loaded from Google drive using the code below to enable colab specific Google drive integration:

```
from google.colab import drive

drive.mount('/gdrive')

%cd /gdrive

df=pd.read_csv('/gdrive/MyDrive/Colab Notebooks/spam.csv', encoding=('ISO-8859-1'))

#This enables panda(pd) to read the file spam.csv mounted on /gdrive and assigns it to variable called df
```

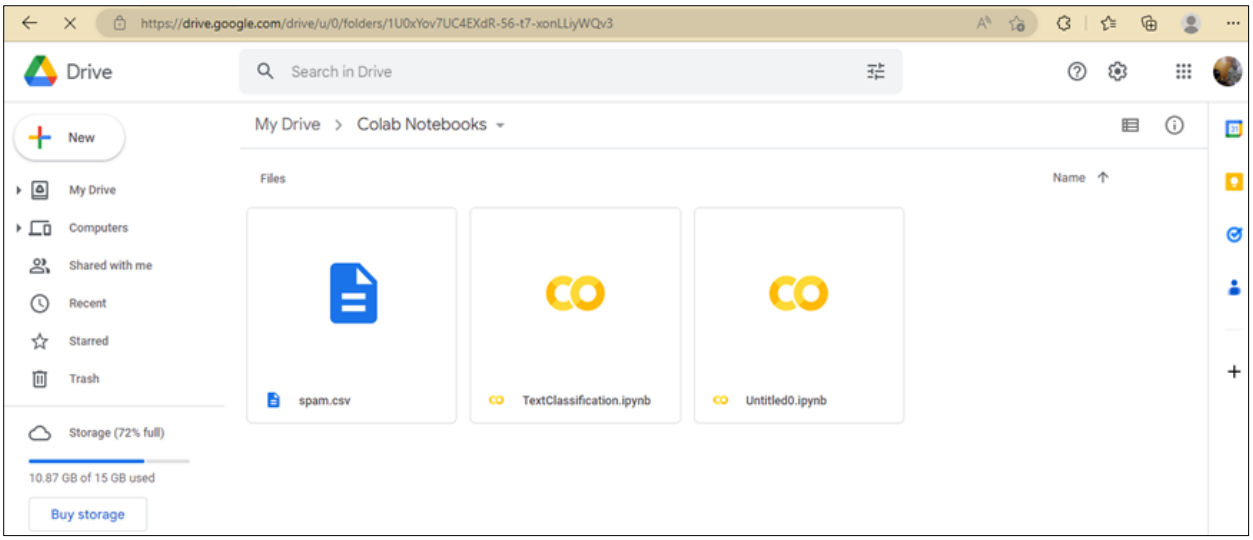


Figure 5 SMS dataset on Google drive

The preprocessing stages involves case conversion, punctuation mark removal, abbreviation expansion, tokenization, stemming and stop words removal.

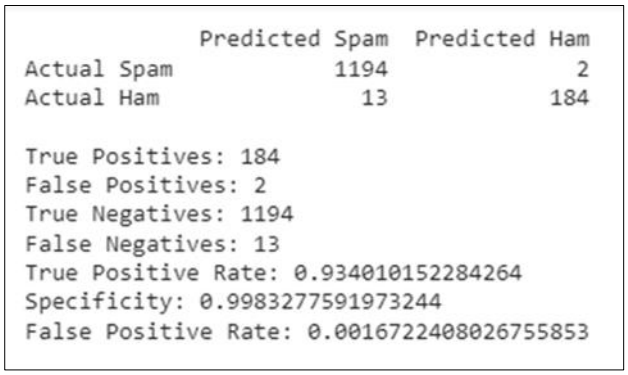
2.6. SVM Algorithm

In the proposed system, we have one dataset with a single feature which includes the source *text* and *label*. The only feature which is selected here is "*text*" that would be used to perform binary classification using machine learning recurrent deep neural networks in order to find out the most dominant and accurate model for classification with respect to SVM. Refer to the SVM algorithm in Table 4 below.

Table 4 SVM algorithm

Pseudo-code : Support Vector Classification, SVC (Outline)
Step 1: Define feature function: add_New_feature(X, feature):
Import csr_matrix, hstack from scipy.sparse#hstack adds sparse feature into training data
Step 2: Fit and transform training and testing dataset using Tfidf Vectorizer:
Find transformed_X_train by fitting X_train in vectorizer
Find and add length of transformed_X_train.str.len() as feature i.e. save in variable transformed_X_train_length
Repeat Step 2 to find and add transformed_X_test and transformed_X_test_length by transforming and fitting X_test datagram
Step 3: Use regularization of C=10000 for SVC i.e cl=SVC(C=10000)
Fit cl to (transformed_X_train_length,y_train), cl is variable that holds fitted data
Predict,y, i.epred_y=cl.predict(transformed_X_test_length)
Step 4: Return area under curve (AUC) score
roc_auc_score(y_test, pred_y)
Step 5: Compute performance metrics using confusion matrix
tn, fp, fn, tp = confusion_matrix(y_test, y_pred).ravel()
True Positive rate=tp / (tp + fn)
Specificity= tn / (tn + fp)
False Positive Rate=fp / (fp + tn)

In order to consider worst case possible, we have used the following characteristics: sigmoid kernel for SVC, sigmoid kernel for neural networks. First, the respective data is split into train data (80%) and test data (20%). Like the case of SVM model we have used Spam dataset in its entirety without equating the outcome values.



```
              Predicted Spam  Predicted Ham
Actual Spam             1194              2
Actual Ham                13            184

True Positives: 184
False Positives: 2
True Negatives: 1194
False Negatives: 13
True Positive Rate: 0.934010152284264
Specificity: 0.9983277591973244
False Positive Rate: 0.0016722408026755853
```

Figure 6 SVM model result

The train data is again split into Training set (80% of Train data) and Validation set (20% of Train data) as shown in Table 5 below. Next, we have tokenized the data frames, followed by padding of the text and finally encoding of the labels after splitting them.

Table 5 Data partitioning

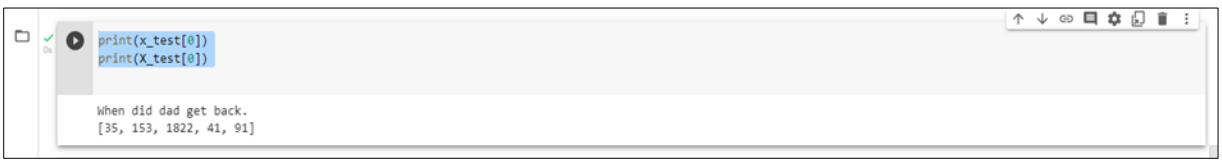
#train test split for SVM model
X_train, X_test, y_train, y_test = train_test_split(df['v2'], df['v1'], test_size=0.20, random_state=0)
#train test split for RNN model
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.20, random_state=0)

2.7. Tokenization

The text essentially was turned into tokens so that they can be processed easily by the system. The text, which is in alphanumeric format, must be converted into numerical format. This tokenization will tend to keep the order of words intact unlike vectorization. Each word will be indexed and mapped to the corresponding ones in the training dataset. This can be easily checked by printing the first SMS text both before and after tokenization:

print(x_test[0]) #small x, prints actual text begging from 0+1

print(X_test[0]) #big X, prints the indexed tokens in numbers

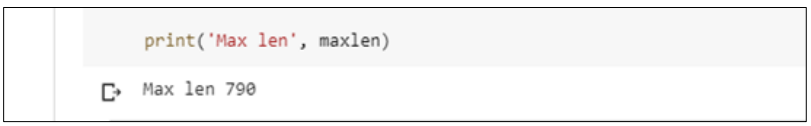


```
print(x_test[0])
print(X_test[0])

When did dad get back.
[35, 153, 1822, 41, 91]
```

Figure 7 Tokenization

2.8. Padding



```
print('Max len', maxlen)

Max len 790
```

Figure 8 Padding

After words have been tokenized, they are made to be of same length because the model structure is the same for training, testing and deployment. Then we pad our inputs to fit that size.

2.9. Encoding

Transforms categorical variables into vectors of zeros (0s) and ones (1s) where each of the vector has a link equal to the number of output categories. Therefore, each time the model receives input label it takes it as a 1 while the other input labels are represented by 0. For instance, if you have two inputs (x_1, x_2) and you receive x_1 the model will receive vector as (1, 0), where x_1 is 1 and x_2 is 0. This is known as One-hot/dummy encoding [172].

2.10. RNN Model Algorithm

We use sequential neural network with 9layers where the first layer is *embedding input layer*, which is determined by maximum length calculated before (790).The embedding of the word (output), essentially acts as the input of another layer. The next layer is *dropout layer*, which uses 20% of neurons to train the dataset. The dropouts are added to help with overtraining. There is a *pool layer*, which reduces neurons and additional hidden layers with 50neurons each and activation function. Finally, a *prediction layer* with 2neurons, which represents the number of labels we have (SPAM, HAM). After building the model, we compile by specifying the *optimizer* as "adam", *loss* as "categorical_cross entropy" and *accuracy* as "metrics".

Table 6 RNN Model Algorithm

Pseudo-code : Recurrent Neural Network, RNN (Outline)
Step 1: Set embedding dimension
embedding_dim=100 #representation of words in vector
Step 2: Initiates the model
model=Sequential()
Step 3: Add first layer (to be tokenized), embedding and size of layer by maxlen
model.add(layers.Embedding(input_dim=vocab_size, output_dim=embedding_dim,input_length=maxlen*)*
Step 4: Add dropout layer to help with overtraining
model.add(layers.Dropout(0.2))
model.add(layers.GlobalMaxPool1D())
Step 5: Add dropout layer and additional hidden layer with activation function
model.add(layers.Dropout(0.2))
model.add(layers.Dense(no_of_neurons, activation=[function]))
Step 6: Add Prediction layer (activation function may be 'softmax')
model.add(layers.Dropout(0.2))
model.add(layers.Dense(no_of_output_classes, activation=[function]
Step 7: Compile model using parameters(optimizer, loss, metrics)
Step 8: Evaluate model with parameters (X_train, dummy_y_train)to find its performance before fitting data
Step 9: Fit model by specifying (X_train,
dummy_y_train,
no of epochs,
verbose=True,
validation_data=(X_test, dummy_y_test),
batch_size)
Step 10: Use keras API to tensorflow, adds additional layers to the model with activation="relu" of weight matrix and bias
model=tf.keras.Sequential([3 layers of relu and sigmoid function])
Step 11: Compile model by invoking Optimizer class with Adam's learning rate and compute loss function (BinaryCrossentropy())and accuracy

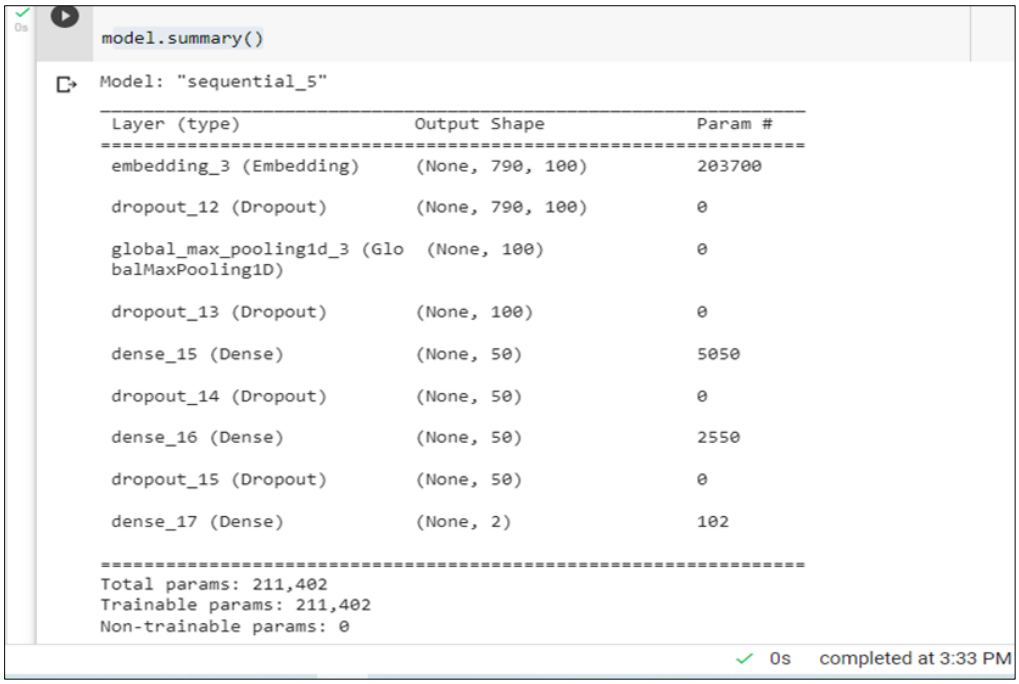


Figure 9 Model summary

The evaluation of the model before fitting it shows an accuracy of 0.696179211139679, and a loss of 0.22576302289962769 which is significant. At this point our concern is more on the reduction of the loss.

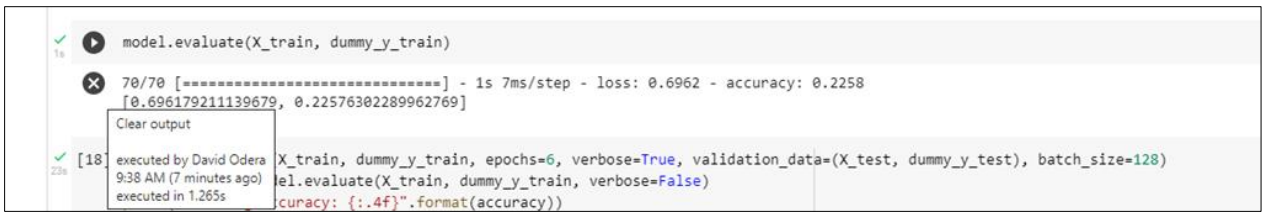


Figure 10 Accuracy and loss

The model is then trained by fitting it in the dataset, specifying the epochs, which determine the number of times model will run through the data:

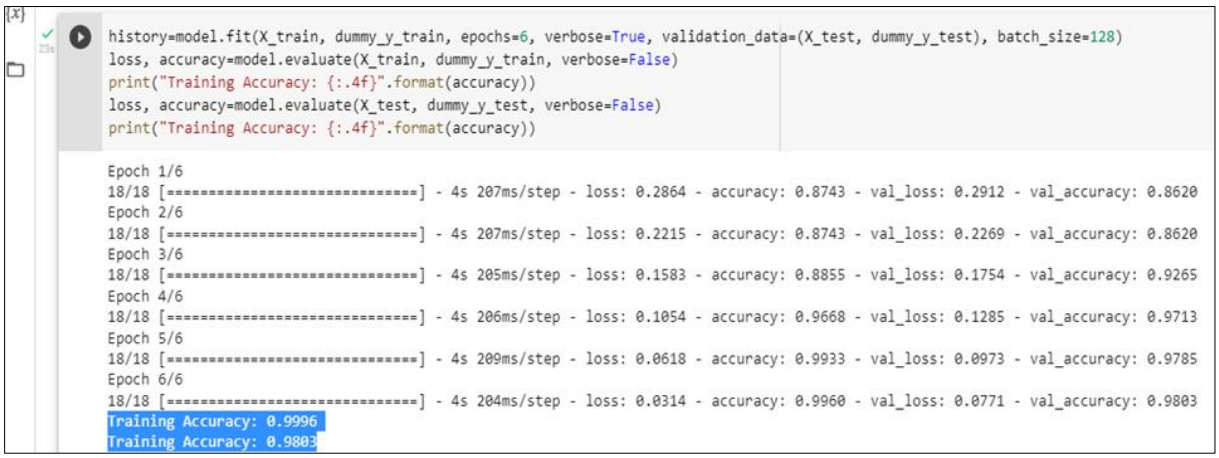


Figure 11 RNN model result

3. Discussion

3.1. Performance of the Model

It is simple to follow how the model performs during each epoch by looking at the graph. The blue and red lines represent the training accuracy and validation accuracy respectively. We then plot the accuracy and loss of the model. Usually, you would see training and validation accuracy both increase over time. The evaluation of the model before fitting it shows an accuracy of 0.696179211139679, at this point it has not been subjected to training. Essentially, at the point where testing crosses validation, you would want to stop training as it means that the model is beginning to remember the exact things instead of learning patterns. In this case, validation accuracy keeps growing even on the 6th epoch. Using RNN algorithm, the minimization of loss is performed through back propagation using auto differentiation (the mathematical notation is explained in the methodology section). The performance of the model shows the loss is decreasing in every epoch from 0.22to 0.0314 at 1st epoch and 6th epoch respectively.

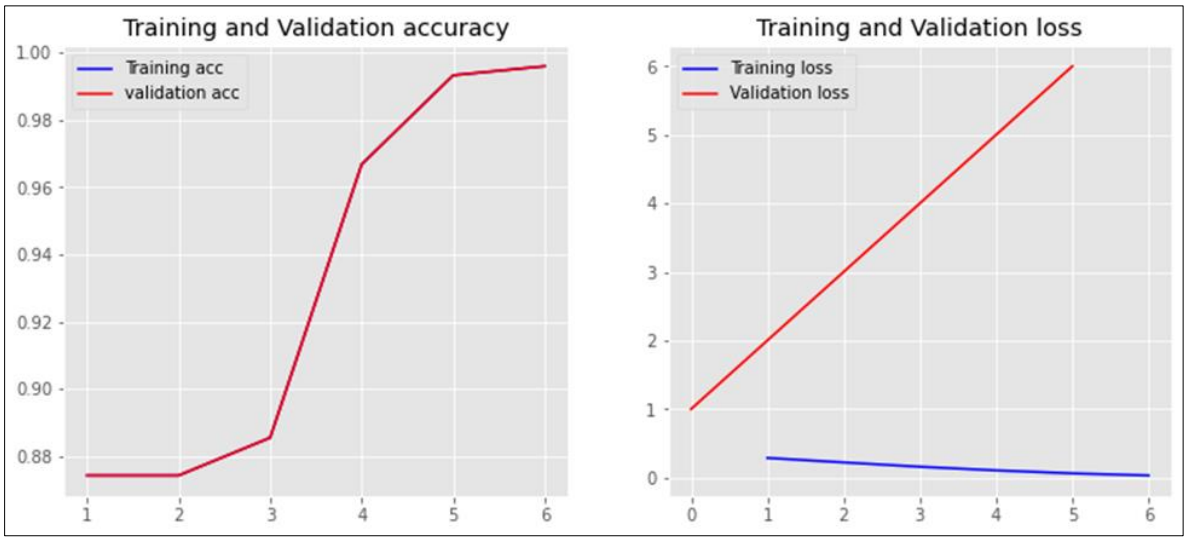


Figure 12 Training and validation accuracy and loss graphs

3.2. Comparison of the Models

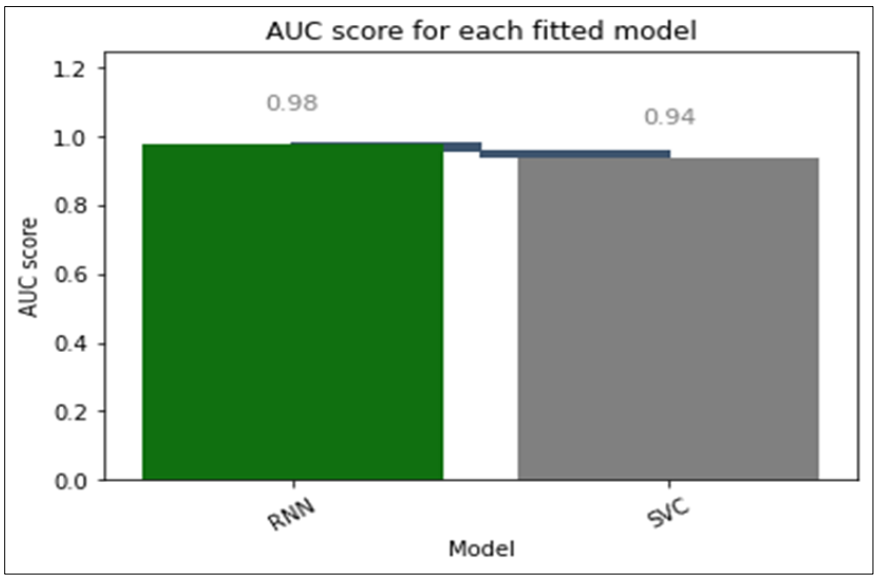


Figure 63 AUC score for each fitted model

The deep learning model was then compared with SVM model in order to find out the model best suited for binary classification.Previous evaluations from other researchers have qualified SVM as the best fit for SMS classification [83].

We performed training, testing and evaluation of SVM and RNN model with an intention to compare them. We have used a similar dataset, sample size, random state and kept all parameters constant. The validation accuracy and loss of RNN outperforms that of SVM model classifier using parameters of same dataset. Based on the graph, RNN has a slightly higher training and validation accuracy of 0.98 compared to SVM at 0.94, however the false positive rate of SVM is marginally lower.

3.3. Key findings

It has been noted that RNN and SVM are both popular machine learning algorithms used for binary classification tasks, including spam detection in Short Message Service (SMS) messages. While they approach the problem from different perspectives, both methods can be effective in identifying spam messages. Table 7 gives a summary of each these machine learning algorithms.

Table 7 Summary of RNN and SVM for SMS spam detection

Algorithm	Description
RNN	A type of neural network specifically designed to handle sequential data by processing information in a sequential manner. They are well-suited for text-based classification tasks, including spam detection. RNNs have the ability to capture the temporal dependencies in SMS messages, considering the ordering and context of words within the message. Process input sequences step by step, maintaining a hidden state that captures the previous context. They learn to predict the next word or classify the text based on the accumulated information from the previous steps. This hidden state allows RNNs to capture long-term dependencies, making them effective for spam detection where the presence of certain words or patterns throughout the message is important. Can be trained using various architectures, such as vanilla RNNs, Long Short-Term Memory (LSTM), or Gated Recurrent Units (GRU). LSTM and GRU are more commonly used due to their ability to mitigate the vanishing gradient problem, allowing them to capture long-term dependencies more effectively.
	Challenges: While Recurrent Neural Networks (RNNs) have proven to be effective for spam detection in Short Message Service (SMS), they also face certain challenges. Here are some of the key challenges associated with using RNNs for spam detection in SMS: *Limited context modeling*: RNNs are designed to capture sequential dependencies in data by maintaining a hidden state [173]. However, in the case of SMS messages, the available context is often limited due to the short length of messages. This can make it challenging for RNNs to effectively capture and model the context necessary for accurate spam detection. *Data sparsity*: SMS spam detection often deals with imbalanced datasets, where the number of spam messages is significantly lower than non-spam messages. This data sparsity can lead to difficulties in training RNNs effectively, as the network may not have sufficient examples of spam messages to learn from. Addressing data sparsity requires careful pre-processing, sampling techniques, or using specialized loss functions to account for class imbalance. *Out-of-vocabulary words*: SMS messages can contain informal language, slang, abbreviations, or misspelled words, which may not be present in the vocabulary used during training [174]. RNNs can struggle to handle out-of-vocabulary words, as they rely on pre-existing word representations. Handling these out-of-vocabulary words often requires preprocessing techniques like word normalization, stemming, or incorporating external sources to enrich the vocabulary. *Over-fitting*: RNNs can be prone to overfitting, especially when the training data is limited. Overfitting occurs when the model learns to memorize the training data instead of generalizing well to new, unseen SMS messages. Regularization techniques, such as dropout and L2 regularization, can be applied to mitigate overfitting. Additionally, using techniques like data augmentation or incorporating external data sources can help improve generalization [176]. *Computational requirements*: RNNs, especially when using complex architectures like LSTM or GRU, can be computationally expensive and require significant resources for training. Training large-scale

	RNN models on large SMS datasets may be time-consuming and require high-performance hardware or distributed computing infrastructure. *Interpretability*: RNNs, particularly with deep architectures, are often considered as black boxes, meaning it can be challenging to interpret the internal workings and understand the decision-making process. Interpreting RNN-based spam detection models and providing explanations for their predictions can be difficult, which may be a requirement in certain applications or industries.
SVM	Is a widely used machine learning algorithm for binary classification tasks. It aims to find an optimal hyperplane that separates the data points of different classes with a maximum margin. In the case of spam classification, SVM tries to find a decision boundary that distinguishes between spam and non-spam messages. To use SVM for text classification, SMS messages need to be represented as numerical feature vectors. One common approach is to convert the messages into a numerical representation using techniques like TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings (such as Word2Vec or GloVe). These representations capture the important characteristics of the text, allowing SVM to learn the decision boundary. Can handle high-dimensional feature spaces and are known for their ability to generalize well even with limited training data. They work effectively when the data is linearly separable or when it can be mapped into a higher-dimensional space where linear separation is possible through the use of kernel functions.
	Challenges: While Support Vector Machines (SVMs) have been widely used for spam detection in various domains, including Short Message Service (SMS), they also face certain challenges when applied to this specific task. Here are some challenges associated with using SVMs for spam detection in SMS: *Text representation*: SVMs require numerical representations of text data as input. Converting SMS messages into suitable feature vectors can be challenging, especially considering the unique characteristics of SMS language. Dealing with slang, abbreviations, misspellings, and the use of informal language in SMS requires careful preprocessing and feature engineering to ensure relevant information is captured effectively. *Curse of dimensionality*: SMS messages can contain a large number of features or words, leading to high-dimensional feature spaces. The curse of dimensionality refers to the fact that as the number of features increases, the sparsity of the data increases as well [176]. This sparsity can adversely affect the performance of SVMs, as the available training instances become sparser, and it becomes harder to find an optimal decision boundary. *Imbalanced datasets*: Like many classification tasks, SMS spam detection often involves imbalanced datasets, where the number of spam messages is much lower than non-spam messages. SVMs can be sensitive to class imbalance, leading to biased models favoring the majority class. Techniques like resampling (e.g., oversampling or under-sampling), cost-sensitive learning, or using specialized loss functions can help address the class imbalance issue. *Selection of kernel function*: SVMs rely on kernel functions to map the input data into a higher-dimensional space where linear separation is possible. Selecting an appropriate kernel function can be challenging, as different kernels may work better for different SMS spam detection scenarios. Choosing the wrong kernel can result in poor performance or difficulties in achieving an effective separation of spam and non-spam classes [177]. *Scalability and computational requirements*: SVMs can become computationally expensive, particularly when dealing with large-scale SMS datasets or high-dimensional feature spaces. Training SVMs with large datasets or with complex kernels can require significant computational resources and time. This can be a challenge when dealing with real-time or high-throughput SMS spam detection systems. *Interpretability*: While SVMs can provide clear decision boundaries and support vectors, they may lack interpretability when applied to text classification tasks. Understanding the importance of individual features or words in the classification decision can be challenging with SVMs, as they primarily focus on optimizing the separation between classes rather than providing feature-level interpretability.

Despite these RNNs challenges, they have demonstrated good performance in spam detection for SMS messages. Researchers continue to explore techniques to address these challenges, such as incorporating attention mechanisms, transfer learning, or combining RNNs with other models to improve overall performance and overcome limitations. On the other hand, addressing SVM challenges often requires careful preprocessing, feature engineering, and hyper-parameter tuning to achieve optimal performance. Additionally, incorporating techniques such as dimensionality reduction, ensemble methods, or combining SVMs with other algorithms may help mitigate some of the challenges associated with SVM-based SMS spam detection. Table 8 compares and contrasts RNN and SVM algorithms using key performance concepts.

Table 8 Comparison of RNN and SVM algorithms

Key performance concept	**Descriptions**
Training	RNNs require more computational resources and training time compared to SVMs, especially when dealing with large datasets. SVMs, on the other hand, can be trained relatively quickly.
Feature Engineering	RNNs can automatically learn representations from the input data, eliminating the need for extensive feature engineering. SVMs, however, typically require manual feature engineering to convert text data into numerical representations.
Interpretability	SVMs provide better interpretability as they offer clear decision boundaries and support vectors that can be inspected. RNNs, being more complex, are often considered as black boxes, making it challenging to understand their internal workings.
Handling Sequential Data	In practice, both RNNs and SVMs have been used successfully for spam detection in SMS messages. The choice between the two depends on the specific requirements of the problem, available computational resources, and the trade-offs between interpretability and performance.

In summary, both RNNs and SVMs have been successfully applied to spam SMS detection. RNNs excel at capturing sequential information and context but require more computational resources and lack interpretability. SVMs are efficient, interpretable, and generalize well but rely on effective text representation and manual feature engineering. The choice between the two depends on the specific requirements, available resources, and the trade-offs between interpretability and performance.

4. Conclusion

A mathematical concept of RNN and back propagation was used to operationalize the model under the development environment of Google Colab using Tensorflow libraries. The study also provides a block diagram, which illustrates process flow from reading of UCl Spam SMS dataset mounted on \gdrive to preprocessing, training, fitting, compilation and evaluation of results. It is evident that both SVM and RNN provides best classification results with a marginal performance in favor of RNN. Back propagation technique in RNN enables minimization of gradient loss, which reduces false positives in every given epoch. The higher the epoch the lesser the loss. Therefore, our result shows that RNN has higher probability when it comes to identification and classification of spam SMS as compared to SVM as shown in area under curve score (AUC) score above. In general, the experimental setting proves that validation accuracy and loss of RNN outperforms that of SVM model classifier using parameters of the same UCl Spam SMS dataset. We suggest use of more performance metrics to validate the model in a distributed dataset environment as a future work.

Compliance with ethical standards

Acknowledgments

We would like the help extended to us by our colleagues at the school level during the writing of this research work.

Disclosure of conflict of interest

The authors declare that they do not have any conflict of interest.

References

[1] Buzdugan A, Capatana G. Cyber security maturity model for critical infrastructures. InEducation, Research and Business Technologies: Proceedings of 20th International Conference on Informatics in Economy (IE 2021) 2022 Apr 16 (pp. 225-236). Singapore: Springer Singapore.

[2] Appiah M, Onifade ST, Gyamfi BA. Building critical infrastructures: Evaluating the roles of governance and institutions in infrastructural developments in Sub-Sahara African countries. Evaluation Review. 2022 Aug;46(4):391-415.

[3] Huddleston P, Smith T, White I, Elrick-Barr C. Adapting critical infrastructure to climate change: A scoping review. Environmental Science & Policy. 2022 Sep 1;135:67-76.

[4] Rathnayaka B, Siriwardana C, Robert D, Amaratunga D, Setunge S. Improving the resilience of critical infrastructure: Evidence-based insights from a systematic literature review. International Journal of Disaster Risk Reduction. 2022 Jun 23:103123.

[5] De Felice F, Baffo I, Petrillo A. Critical Infrastructures Overview: Past, Present and Future. Sustainability. 2022 Feb 16;14(4):2233.

[6] Wisniewski M, Gladysz B, Ejsmont K, Wodecki A, Van Erp T. Industry 4.0 solutions impacts on critical infrastructure safety and protection–a systematic literature review. IEEE Access. 2022 Aug 1.

[7] Nyangaresi VO. Privacy Preserving Three-factor Authentication Protocol for Secure Message Forwarding in Wireless Body Area Networks. Ad Hoc Networks. 2023 Apr 1;142:103117.

[8] Kumar A, Sharma S, Singh A, Alwadain A, Choi BJ, Manual-Brenosa J, Ortega-Mansilla A, Goyal N. Revolutionary strategies analysis and proposed system for future infrastructure in Internet of Things. Sustainability. 2022 Jan;14(1):71.

[9] Vermesan O, Friess P, editors. Digitising the Industry Internet of Things Connecting the Physical, Digital and VirtualWorlds. CRC Press; 2022 Sep 1.

[10] Mishra S, Tyagi AK. The role of machine learning techniques in internet of things-based cloud applications. Artificial intelligence-based internet of things systems. 2022:105-35.

[11] Pourrahmani H, Yavarinasab A, Zahedi R, Gharehghani A, Mohammadi MH, Bastani P. The applications of Internet of Things in the automotive industry: A review of the batteries, fuel cells, and engines. Internet of Things. 2022 Jul 8:100579.

[12] Elgazzar K, Khalil H, Alghamdi T, Badr A, Abdelkader G, Elewah A, Buyya R. Revisiting the internet of things: New trends, opportunities and grand challenges. Frontiers in the Internet of Things. 2022 Nov 21;1:7.

[13] Nyangaresi VO, Ogundoyin SO. Certificate based authentication scheme for smart homes. In2021 3rd Global Power, Energy and Communication Conference (GPECOM) 2021 Oct 5 (pp. 202-207). IEEE.

[14] Mijwil M, Unogwu OJ, Filali Y, Bala I, Al-Shahwani H. Exploring the Top Five Evolving Threats in Cybersecurity: An In-Depth Overview. Mesopotamian journal of cybersecurity. 2023 Mar 6;2023:57-63.

[15] Telo J. Smart City Security Threats and Countermeasures in the Context of Emerging Technologies. International Journal of Intelligent Automation and Computing. 2023 Feb 27;6(1):31-45.

[16] Smith KT, Smith LM, Burger M, Boyle ES. Cyber terrorism cases and stock market valuation effects. Information & Computer Security. 2023 Feb 2.

[17] Arogundade OR. Network Security Concepts, Dangers, and Defense Best Practical. Computer Engineering and Intelligent Systems. 2023;14(2).

[18] Nyangaresi VO. Lightweight anonymous authentication protocol for resource-constrained smart home devices based on elliptic curve cryptography. Journal of Systems Architecture. 2022 Dec 1;133:102763.

[19] Chesti IA, Humayun M, Sama NU, Jhanjhi NZ. Evolution, mitigation, and prevention of ransomware. In2020 2nd International Conference on Computer and Information Sciences (ICCIS) 2020 Oct 13 (pp. 1-6). IEEE.

[20] Patyal M, Sampalli S, Ye Q, Rahman M. Multi-layered defense architecture against ransomware. International Journal of Business and Cyber Security. 2017;1(2).

[21] Mohurle S, Patil M. A brief study of wannacry threat: Ransomware attack 2017. International Journal of Advanced Research in Computer Science. 2017 May 15;8(5):1938-40.

[22] Mos MA, Chowdhury MM. The growing influence of ransomware. In2020 IEEE International Conference on Electro Information Technology (EIT) 2020 Jul 31 (pp. 643-647). IEEE.

[23] Gonzalez D, Hayajneh T. Detection and prevention of crypto-ransomware. In2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON) 2017 Oct 19 (pp. 472-478). IEEE.

[24] Pethers B, Bello A. Role of Attention and Design Cues for Influencing Cyber-Sextortion Using Social Engineering and Phishing Attacks. Future Internet. 2023 Jan;15(1):29.

[25] Nyangaresi VO. Provably Secure Pseudonyms based Authentication Protocol for Wearable Ubiquitous Computing Environment. In2022 International Conference on Inventive Computation Technologies (ICICT) 2022 Jul 20 (pp. 1-6). IEEE.

[26] Al-Khateeb M, Al-Mousa M, Al-Sherideh A, Almajali D, Asassfeha M, Khafajeh H. Awareness model for minimizing the effects of social engineering attacks in web applications. International Journal of Data and Network Science. 2023;7(2):791-800.

[27] Fadhil HS. Social Engineering Attacks Techniques. Int. J. Progress. Res. Eng. Manag. Sci. IJPREMS. 2023;3:18-20.

[28] Said D. Quantum Computing and Machine Learning for Cybersecurity: Distributed Denial of Service (DDoS) Attack Detection on Smart Micro-Grid. Energies. 2023 Apr 20;16(8):3572.

[29] Singh A, Gupta BB. Distributed denial-of-service (DDoS) attacks and defense mechanisms in various web-enabled computing platforms: issues, challenges, and future research directions. International Journal on Semantic Web and Information Systems (IJSWIS). 2022 Jan 1;18(1):1-43.

[30] de Neira AB, Kantarci B, Nogueira M. Distributed denial of service attack prediction: Challenges, open issues and opportunities. Computer Networks. 2023 Jan 3:109553.

[31] Nyangaresi VO. Masked Symmetric Key Encrypted Verification Codes for Secure Authentication in Smart Grid Networks. In2022 4th Global Power, Energy and Communication Conference (GPECOM) 2022 Jun 14 (pp. 427-432). IEEE.

[32] Pal P, Chattopadhyay P, Swarnkar M. Temporal feature aggregation with attention for insider threat detection from activity logs. Expert Systems with Applications. 2023 Aug 15;224:119925.

[33] Marbut AR, Harms PD. Fiends and Fools: A Narrative Review and Neo-socioanalytic Perspective on Personality and Insider Threats. Journal of Business and Psychology. 2023 May 9:1-8.

[34] Luo Y. A general framework of digitization risks in international business. Journal of international business studies. 2022 Mar;53(2):344-61.

[35] Behera PK, Gangopadhyay S. Evolving bijective S-Boxes using hybrid adaptive genetic algorithm with optimal cryptographic properties. Journal of Ambient Intelligence and Humanized Computing. 2023 Mar;14(3):1713-30.

[36] Alsamhi SH, Shvetsov AV, Kumar S, Shvetsova SV, Alhartomi MA, Hawbani A, Rajput NS, Srivastava S, Saif A, Nyangaresi VO. UAV computing-assisted search and rescue mission framework for disaster and harsh environment mitigation. Drones. 2022 Jun 22;6(7):154.

[37] Shoka AA, Dessouky MM, El-Sayed A, Hemdan EE. An efficient CNN based epileptic seizures detection framework using encrypted EEG signals for secure telemedicine applications. Alexandria Engineering Journal. 2023 Feb 15;65:399-412.

[38] Selvaraj D, Sankar SM, Dhinakaran D, Anish TP. Outsourced Analysis of Encrypted Graphs in the Cloud with Privacy Protection. arXiv preprint arXiv:2304.10833. 2023 Apr 21.

[39] Abd-El-Atty B, ElAffendi M, El-Latif AA. A novel image cryptosystem using Gray code, quantum walks, and Henon map for cloud applications. Complex & Intelligent Systems. 2023 Feb;9(1):609-24.

[40] Devi SR, Kalyampudi PL, Charitha NS. Cyber attacks, security data detection, and critical loads in the power systems. InSmart Energy and Electric Power Systems 2023 Jan 1 (pp. 169-184). Elsevier.

[41] Nyangaresi VO, Ma J. A Formally Verified Message Validation Protocol for Intelligent IoT E-Health Systems. In2022 IEEE World Conference on Applied Intelligence and Computing (AIC) 2022 Jun 17 (pp. 416-422). IEEE.

[42] Miller M, Kisiel A, Cembrowska-Lech D, Durlik I, Miller T. IoT in Water Quality Monitoring—Are We Really Here?. Sensors. 2023 Jan;23(2):960.

[43] Easttom C. Virtual Private Networks, Authentication, and Wireless Security. InModern Cryptography: Applied Mathematics for Encryption and Information Security 2022 Oct 30 (pp. 309-327). Cham: Springer International Publishing.

[44] Mughal AA. Well-Architected Wireless Network Security. Journal of Humanities and Applied Science Research. 2022 Dec 6;5(1):32-42.

[45] Rahaman MS, Tisha SN, Song E, Cerny T. Access Control Design Practice and Solutions in Cloud-Native Architecture: A Systematic Mapping Study. Sensors. 2023 Mar 24;23(7):3413.

[46] Nyangaresi VO. A Formally Validated Authentication Algorithm for Secure Message Forwarding in Smart Home Networks. SN Computer Science. 2022 Jul 9;3(5):364.

[47] Golightly L, Modesti P, Garcia R, Chang V. Securing Distributed Systems: A Survey on Access Control Techniques for Cloud, Blockchain, IoT and SDN. Cyber Security and Applications. 2023 Mar 15:100015.

[48] Möller DP. Intrusion detection and prevention. InGuide to Cybersecurity in Digital Transformation: Trends, Methods, Technologies, Applications and Best Practices 2023 Apr 19 (pp. 131-179). Cham: Springer Nature Switzerland.

[49] Cheng L, Liu F, Yao D. Enterprise data breach: causes, challenges, prevention, and future directions. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 2017 Sep;7(5):e1211.

[50] Mughal AA. Cyber Attacks on OSI Layers: Understanding the Threat Landscape. Journal of Humanities and Applied Science Research. 2020 Jan 15;3(1):1-8.

[51] Balakrishnan N, Rajendran A, Pelusi D, Ponnusamy V. Deep Belief Network enhanced intrusion detection system to prevent security breach in the Internet of Things. Internet of things. 2021 Jun 1;14:100112.

[52] Nyangaresi VO, Ahmad M, Alkhayyat A, Feng W. Artificial neural network and symmetric key cryptography based verification protocol for 5G enabled Internet of Things. Expert Systems. 2022 Dec;39(10):e13126.

[53] Mugarza I, Parra J, Jacob E. Cetratus: A framework for zero downtime secure software updates in safety-critical systems. Software: Practice and Experience. 2020 Aug;50(8):1399-424.

[54] Nikitin K, Kokoris-Kogias E, Jovanovic P, Gailly N, Gasser L, Khoffi I, Cappos J, Ford B. CHAINIAC: Proactive Software-Update Transparency via Collectively Signed Skipchains and Verified Builds. InUSENIX Security Symposium 2017 Aug 16 (pp. 1271-1287).

[55] Yim KS, Malchev I, Hsieh A, Burke D. Treble: Fast software updates by creating an equilibrium in an active software ecosystem of globally distributed stakeholders. ACM Transactions on Embedded Computing Systems (TECS). 2019 Oct 8;18(5s):1-23.

[56] Fan X, Fan K, Wang Y, Zhou R. Overview of cyber-security of industrial control system. In2015 international conference on cyber security of smart cities, industrial control system and communications (SSIC) 2015 Aug 5 (pp. 1-7). IEEE.

[57] Jarmakiewicz J, Parobczak K, Maślanka K. Cybersecurity protection for power grid control infrastructures. International Journal of Critical Infrastructure Protection. 2017 Sep 1;18:20-33.

[58] Nyangaresi VO. ECC based authentication scheme for smart homes. In2021 International Symposium ELMAR 2021 Sep 13 (pp. 5-10). IEEE.

[59] Tudosi AD, Graur A, Balan DG, Potorac AD. Research on Security Weakness Using Penetration Testing in a Distributed Firewall. Sensors. 2023 Mar 1;23(5):2683.

[60] Altulaihan EA, Alismail A, Frikha M. A Survey on Web Application Penetration Testing. Electronics. 2023 Mar 4;12(5):1229.

[61] George AS, Sagayarajan S. Securing Cloud Application Infrastructure: Understanding the Penetration Testing Challenges of IaaS, PaaS, and SaaS Environments. Partners Universal International Research Journal. 2023 Mar 31;2(1):24-34.

[62] Steinmetz KF. Executing effective social engineering penetration tests: A qualitative analysis. Journal of Applied Security Research. 2023 Apr 3;18(2):246-66.

[63] Benyahya M, Bergerat P, Collen A, Nijdam NA. Symbiotic Analysis of Security Assessment and Penetration Tests Guiding Real L4 Automated City Shuttles. InTelecom 2023 Mar 20 (Vol. 4, No. 1, pp. 198-218). MDPI.

[64] Shah S, Mehtre BM. An overview of vulnerability assessment and penetration testing techniques. Journal of Computer Virology and Hacking Techniques. 2015 Feb;11:27-49.

[65] Nyangaresi VO. Provably secure protocol for 5G HetNets. In2021 IEEE International Conference on Microwaves, Antennas, Communications and Electronic Systems (COMCAS) 2021 Nov 1 (pp. 17-22). IEEE.

[66] Saleem J, Hammoudeh M. Defense methods against social engineering attacks. Computer and network security essentials. 2018:603-18.

[67] Kioskli K, Fotis T, Nifakos S, Mouratidis H. The Importance of Conceptualising the Human-Centric Approach in Maintaining and Promoting Cybersecurity-Hygiene in Healthcare 4.0. Applied Sciences. 2023 Mar 7;13(6):3410.

[68] Campbell CC. Solutions for counteracting human deception in social engineering attacks. Information Technology & People. 2019 Sep 23;32(5):1130-52.

[69] Ahmad A, Desouza KC, Maynard SB, Naseer H, Baskerville RL. How integration of cyber security management and incident response enables organizational learning. Journal of the Association for Information Science and Technology. 2020 Aug;71(8):939-53.

[70] Naseer H, Maynard SB, Desouza KC. Demystifying analytical information processing capability: The case of cybersecurity incident response. Decision Support Systems. 2021 Apr 1;143:113476.

[71] Stevens R, Votipka D, Dykstra J, Tomlinson F, Quartararo E, Ahern C, Mazurek ML. How ready is your ready? assessing the usability of incident response playbook frameworks. InProceedings of the 2022 CHI Conference on Human Factors in Computing Systems 2022 Apr 29 (pp. 1-18).

[72] Nyangaresi VO. Terminal independent security token derivation scheme for ultra-dense IoT networks. Array. 2022 Sep 1;15:100210.

[73] Landauer M, Wurzenberger M, Skopik F, Hotwagner W, Höld G. Aminer: A modular log data analysis pipeline for anomaly-based intrusion detection. Digital Threats: Research and Practice. 2023 Mar 31;4(1):1-6.

[74] Wawrowski Ł, Białas A, Kajzer A, Kozłowski A, Kurianowicz R, Sikora M, Szymańska-Kwiecień A, Uchroński M, Białczak M, Olejnik M, Michalak M. Anomaly detection module for network traffic monitoring in public institutions. Sensors. 2023 Mar 9;23(6):2974.

[75] Kim B, Alawami MA, Kim E, Oh S, Park J, Kim H. A comparative study of time series anomaly detection models for industrial control systems. Sensors. 2023 Jan 23;23(3):1310.

[76] Nyangaresi VO. Hardware assisted protocol for attacks prevention in ad hoc networks. InEmerging Technologies in Computing: 4th EAI/IAER International Conference, iCETiC 2021, Virtual Event, August 18–19, 2021, Proceedings 4 2021 (pp. 3-20). Springer International Publishing.

[77] Kalia P, Dwivedi YK, Acevedo-Duque Á. Cellulographics©: A novel smartphone user classification metrics. Journal of Innovation & Knowledge. 2022 Apr 1;7(2):100179.

[78] Bojjagani S, Sastry VN. A secure end-to-end SMS-based mobile banking protocol. International journal of communication systems. 2017 Oct;30(15):e3302.

[79] Yerima SY, Bashar A. Semi-supervised novelty detection with one class SVM for SMS spam detection. In2022 29th International Conference on Systems, Signals and Image Processing (IWSSIP) 2022 Jun 1 (pp. 1-4). IEEE.

[80] Ezpeleta E, Zurutuza U, Hidalgo JM. Short messages spam filtering using personality recognition. InProceedings of the 4th Spanish Conference on Information Retrieval 2016 Jun 14 (pp. 1-7).

[81] Fernandes D, Da Costa KA, Almeida TA, Papa JP. SMS spam filteringthrough optimum-pathforest-basedclassifiers. In2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA) 2015 Dec 9 (pp. 133-137). IEEE.

[82] Omollo VN, Musyoki S. Global Positioning System Based Routing Algorithm for Adaptive Delay Tolerant Mobile Adhoc Networks. International Journal of Computer and Communication System Engineering. 2015 May 11, 2(3): 399-406.

[83] Navaney P, Dubey G, Rana A. SMS spam filtering using supervised machine learning algorithms. In2018 8th international conference on cloud computing, data science & engineering (confluence) 2018 Jan 11 (pp. 43-48). IEEE.

[84] Abayomi-Alli O, Misra S, Abayomi-Alli A, Odusami M. A review of soft techniques for SMS spam classification: Methods, approaches and applications. Engineering Applications of Artificial Intelligence. 2019 Nov 1;86:197-212.

[85] Liu X, Lu H, Nayak A. A spam transformer model for SMS spam detection. IEEE Access. 2021 May 17;9:80253-63.

[86] Nyangaresi VO. Lightweight key agreement and authentication protocol for smart homes. In2021 IEEE AFRICON 2021 Sep 13 (pp. 1-6). IEEE.

[87] Mishra S, Soni D. Dsmishsms-a system to detect smishing sms. Neural Computing and Applications. 2021 Jul 28:1-8.

[88] Shafi'I MA, Abd Latiff MS, Chiroma H, Osho O, Abdul-Salaam G, Abubakar AI, Herawan T. A review on mobile SMS spam filtering techniques. IEEE Access. 2017 Feb 10;5:15650-66.

[89] Wei F, Nguyen T. A lightweight deep neural model for sms spam detection. In2020 International Symposium on Networks, Computers and Communications (ISNCC) 2020 Oct 20 (pp. 1-6). IEEE.

[90] Sidhpura J, Shah P, Veerkhare R, Godbole A. FedSpam: Privacy Preserving SMS Spam Prediction. InNeural Information Processing: 29th International Conference, ICONIP 2022, Virtual Event, November 22–26, 2022, Proceedings, Part VI 2023 Apr 14 (pp. 52-63). Singapore: Springer Nature Singapore.

[91] Cinar AC, Kara TB. The current state and future of mobile security in the light of the recent mobile security threat reports. Multimedia Tools and Applications. 2023 Jan 30:1-3.

[92] Tinubu CO, Falana OJ, Oluwumi EO, Sodiya AS, Rufai SA. PHISHGEM: a mobile game-based learning for phishing awareness. Journal of Cyber Security Technology. 2023 Jan 26:1-20.

[93] Nyangaresi VO, Alsamhi SH. Towards secure traffic signaling in smart grids. In2021 3rd Global Power, Energy and Communication Conference (GPECOM) 2021 Oct 5 (pp. 196-201). IEEE.

[94] Sarker IH. Machine learning: Algorithms, real-world applications and research directions. SN computer science. 2021 May;2(3):160.

[95] Liu K, Xu S, Xu G, Zhang M, Sun D, Liu H. A review of android malware detection approaches based on machine learning. IEEE Access. 2020 Jul 1;8:124579-607.

[96] Yu K, Tan L, Mumtaz S, Al-Rubaye S, Al-Dulaimi A, Bashir AK, Khan FA. Securing critical infrastructures: deep-learning-based threat detection in IIoT. IEEE Communications Magazine. 2021 Oct;59(10):76-82.

[97] Sarker IH. Context-aware rule learning from smartphone data: survey, challenges and future directions. Journal of Big Data. 2019 Dec;6(1):1-25.

[98] Hussain B, Du Q, Ren P. Semi-supervised learning based big data-driven anomaly detection in mobile wireless networks. China Communications. 2018 May 16;15(4):41-57.

[99] Honi DG, Ali AH, Abduljabbar ZA, Ma J, Nyangaresi VO, Mutlaq KA, Umran SM. Towards Fast Edge Detection Approach for Industrial Products. In2022 IEEE 21st International Conference on Ubiquitous Computing and Communications (IUCC/CIT/DSCI/SmartCNS) 2022 Dec 19 (pp.39-244). IEEE.

[100] Sjarif NN, Yahya Y, Chuprat S, Azmi NH. Support vector machine algorithm for SMS spam classification in the telecommunication industry. Int. J. Adv. Sci. Eng. Inf. Technol. 2020;10(2):635-9.

[101] Xia T, Chen X. A discrete hidden Markov model for SMS spam detection. Applied Sciences. 2020 Jul 21;10(14):5011.

[102] McCelloch WS, Pitts W. A logical calculus of the idea immanent in neural nets. Bulletin ofMathematical Biophysics. 1943;5:115-33.

[103] Nyangaresi VO, Rodrigues AJ. Efficient handover protocol for 5G and beyond networks. Computers & Security. 2022 Feb 1;113:102546.

[104] Indrakumari R, Poongodi T, Singh K. Introduction to Deep Learning. Advanced Deep Learning for Engineers and Scientists: A Practical Approach. 2021:1-22.

[105] Blake R, Michalikova KF. Deep learning-based sensing technologies, artificial intelligence-based decision-making algorithms, and big geospatial data analytics in cognitive internet of things. Analysis and Metaphysics. 2021;20:159-73.

[106] Huang-Fu CY, Liao CH, Wu JY. Comparing the performance of machine learning and deep learning algorithms classifying messages in Facebook learning group. In2021 International Conference on Advanced Learning Technologies (ICALT) 2021 Jul 12 (pp. 347-349). IEEE.

[107] Goh GB, Hodas NO, Vishnu A. Deep learning for computational chemistry. Journal of computational chemistry. 2017 Jun 15;38(16):1291-307.

[108] Ghrabat MJ, Hussien ZA, Khalefa MS, Abduljabba ZA, Nyangaresi VO, Al Sibahee MA, Abood EW. Fully automated model on breast cancer classification using deep learning classifiers. Indonesian Journal of Electrical Engineering and Computer Science. 2022 Oct;28(1):183-91.

[109] Ma D, Song X, Li P. Daily traffic flow forecastingthrough a contextualconvolutionalrecurrent neural network modeling inter-and intra-daytraffic patterns. IEEE Transactions on Intelligent Transportation Systems. 2020 Feb 24;22(5):2627-36.

[110] Lauriola I, Lavelli A, Aiolli F. An introduction to deep learning in natural language processing: Models, techniques, and tools. Neurocomputing. 2022 Jan 22;470:443-56.

[111] Prakash N, Manconi A, Loew S. A new strategy to map landslides with a generalized convolutional neural network. Scientific reports. 2021 May 6;11(1):9722.

[112] Liu S, Liu S, Cai W, Pujol S, Kikinis R, Feng D. Early diagnosis of Alzheimer's disease with deep learning. In2014 IEEE 11th international symposium on biomedical imaging (ISBI) 2014 Apr 29 (pp. 1015-1018). IEEE.

[113] Brosch T, Tam R, Alzheimer's Disease Neuroimaging Initiative. Manifold learning of brain MRIs by deep learning. InMedical Image Computing and Computer-Assisted Intervention–MICCAI 2013: 16th International Conference, Nagoya, Japan, September 22-26, 2013, Proceedings, Part II 16 2013 (pp. 633-640). Springer Berlin Heidelberg.

[114] Geetha R, Thilagam T. A review on the effectiveness of machine learning and deep learning algorithms for cyber security. Archives of Computational Methods in Engineering. 2021 Jun;28:2861-79.

[115] Gupta SD, Saha S, Das SK. SMS spam detection using machine learning. InJournal of Physics: Conference Series 2021 Feb 1 (Vol. 1797, No. 1, p. 012017). IOP Publishing.

[116] Nyangaresi VO, Abduljabbar ZA, Al Sibahee MA, Ibrahim A, Yahya AN, Abduljaleel IQ, Abood EW. Optimized Hysteresis Region Authenticated Handover for 5G HetNets. InArtificial Intelligence and Sustainable Computing: Proceedings of ICSISCET 2021 2022 Nov 16 (pp. 91-111). Singapore: Springer Nature Singapore.

[117] Aslam M, Lee JM, Kim HS, Lee SJ, Hong S. Deep learning models for long-term solar radiation forecasting considering microgrid installation: A comparative study. Energies. 2019 Dec 27;13(1):147.

[118] Herm LV, Heinrich K, Wanner J, Janiesch C. Stop ordering machine learning algorithms by their explainability! A user-centered investigation of performance and explainability. International Journal of Information Management. 2023 Apr 1;69:102538.

[119] Pabuçcu H, Ongan S, Ongan A. Forecasting the movements of Bitcoin prices: an application of machine learning algorithms. arXiv preprint arXiv:2303.04642. 2023 Mar 8.

[120] Wagner M, Müller-Stich BP, Kisilenko A, Tran D, Heger P, Mündermann L, Lubotsky DM, Müller B, Davitashvili T, Capek M, Reinke A. Comparative validation of machine learning algorithms for surgical workflow and skill analysis with the heichole benchmark. Medical Image Analysis. 2023 May 1;86:102770.

[121] Atalan A. Forecasting drinking milk price based on economic, social, and environmental factors using machine learning algorithms. Agribusiness. 2023 Jan;39(1):214-41.

[122] Nyangaresi VO, El-Omari NK, Nyakina JN. Efficient Feature Selection and ML Algorithm for Accurate Diagnostics. Journal of Computer Science Research. 2022 Jan 25;4(1):10-9.

[123] Xin Y, Kong L, Liu Z, Chen Y, Li Y, Zhu H, Gao M, Hou H, Wang C. Machine learning and deep learning methods for cybersecurity. Ieee access. 2018 May 15;6:35365-81.

[124] Al-Hawawreh M, Moustafa N, Garg S, Hossain MS. Deep learning-enabled threat intelligence scheme in the internet of things networks. IEEE Transactions on Network Science and Engineering. 2020 Oct 20;8(4):2968-81.

[125] Kurani A, Doshi P, Vakharia A, Shah M. A comprehensive comparative study of artificial neural network (ANN) and support vector machines (SVM) on stock forecasting. Annals of Data Science. 2023 Feb;10(1):183-208.

[126] Rawson A, Brito M. A survey of the opportunities and challenges of supervised machine learning in maritime risk analysis. Transport Reviews. 2023 Jan 2;43(1):108-30.

[127] Diaz-Escobar J, Díaz-Montiel P, Venkataraman S, Díaz-Ramírez A. Classification and characterization of damage in composite laminates using electrical resistance tomography and supervised machine learning. Structural Control and Health Monitoring. 2023 Feb 8;2023.

[128] Nyangaresi VO, Rodrigues AJ, Abeka SO. ANN-FL secure handover protocol for 5G and beyond networks. InTowards new e-Infrastructure and e-Services for Developing Countries: 12th EAI International Conference, AFRICOMM 2020, Ebène City, Mauritius, December 2-4, 2020, Proceedings 12 2021 (pp. 99-118). Springer International Publishing.

[129] Taheri R, Javidan R. Spam filtering in SMS using recurrent neural networks. In2017 Artificial Intelligence and Signal Processing Conference (AISP) 2017 Oct 25 (pp. 331-336). IEEE.

[130] Batani J, Mbunge E, Muchemwa B, Gaobotse G, Gurajena C, Fashoto S, Kavu T, Dandajena K. A Review of Deep Learning Models for Detecting Cyberbullying on Social Media Networks. InCybernetics Perspectives in Systems: Proceedings of 11th Computer Science On-line Conference 2022, Vol. 3 2022 Jul 5 (pp. 528-550). Cham: Springer International Publishing.

[131] Abid F, Li C, Alam M. Multi-source social media data sentiment analysis using bidirectional recurrent convolutional neural networks. Computer Communications. 2020 May 1;157:102-15.

[132] Kang Y, Cai Z, Tan CW, Huang Q, Liu H. Natural language processing (NLP) in management research: A literature review. Journal of Management Analytics. 2020 Apr 2;7(2):139-72.

[133] Kursuncu U, Gaur M, Lokala U, Thirunarayan K, Sheth A, Arpinar IB. Predictive analysis on Twitter: Techniques and applications. Emerging research challenges and opportunities in computational social network analysis and mining. 2019:67-104.

[134] Nyangaresi VO, Abeka SO, Rodrigues AJ. Delay sensitive protocol for high availability LTE handovers. American Journal of Networks and Communications. 2020 Feb; 9(1): 1-10.

[135] Sahmoud T, Mikki D. Spam detection using BERT. arXiv preprint arXiv:2206.02443. 2022 Jun 6.

[136] Tida VS, Hsu S. Universal spam detection using transfer learning of BERT model. arXiv preprint arXiv:2202.03480. 2022 Feb 7.

[137] Khan S, Sajjad M, Hussain T, Ullah A, Imran AS. A review on traditional machine learning and deep learning models for WBCs classification in blood smear images. Ieee Access. 2020 Dec 30;9:10657-73.

[138] Wang P, Fan E, Wang P. Comparative analysis of image classification algorithms based on traditional machine learning and deep learning. Pattern Recognition Letters. 2021 Jan 1;141:61-7.

[139] Caroppo A, Leone A, Siciliano P. Comparison between deep learning models and traditional machine learning approaches for facial expression recognition in ageing adults. Journal of Computer Science and Technology. 2020 Oct;35:1127-46.

[140] Al Sibahee MA, Ma J, Nyangaresi VO, Abduljabbar ZA. Efficient Extreme Gradient Boosting Based Algorithm for QoS Optimization in Inter-Radio Access Technology Handoffs. In2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA) 2022 Jun 9 (pp. 1-6). IEEE.

[141] Roy PK, Singh JP, Banerjee S. Deep learning to filter SMS Spam. Future Generation Computer Systems. 2020 Jan 1;102:524-33.

[142] Liu Y, Wang L, Shi T, Li J. Detection of spam reviews through a hierarchical attention architecture with N-gram CNN and Bi-LSTM. Information Systems. 2022 Jan 1;103:101865.

[143] Rahman SE, Ullah S. Email spam detection using bidirectional long short term memory with convolutional neural network. In2020 IEEE Region 10 Symposium (TENSYMP) 2020 Jun 5 (pp. 1307-1311). IEEE.

[144] Raj H, Weihong Y, Banbhrani SK, Dino SP. LSTM based short message service (SMS) modeling for spam classification. InProceedings of the 2018 International Conference on Machine Learning Technologies 2018 May 19 (pp. 76-80).

[145] Ghourabi A, Mahmood MA, Alzubi QM. A hybrid CNN-LSTM model for SMS spam detection in arabic and english messages. Future Internet. 2020 Sep 18;12(9):156.

[146] Nyangaresi VO, Rodrigues AJ, Abeka SO. Machine learning protocol for secure 5G handovers. International Journal of Wireless Information Networks. 2022 Mar;29(1):14-35.

[147] Datta D, David PE, Mittal D, Jain A. Neural machine translation using recurrent neural network. International Journal of Engineering and Advanced Technology. 2020 Apr 10;9(4):1395-400.

[148] Ali S, Masood K, Riaz A, Saud A. Named Entity Recognition using Deep Learning: A Review. In2022 International Conference on Business Analytics for Technology and Security (ICBATS) 2022 Feb 16 (pp. 1-7). IEEE.

[149] Dinghofer K, Hartung F. Analysis of criteria for the selection of machine learning frameworks. In2020 International Conference on Computing, Networking and Communications (ICNC) 2020 Feb 17 (pp. 373-377). IEEE.

[150] Irmak B, Karakoyun M, Gülcü Ş. An improved butterfly optimization algorithm for training the feed-forward artificial neural networks. Soft Computing. 2023 Apr;27(7):3887-905.

[151] Xia JS, Khabaz MK, Patra I, Khalid I, Alvarez JR, Rahmanian A, Eftekhari SA, Toghraie D. Using feed-forward perceptron Artificial Neural Network (ANN) model to determine the rolling force, power and slip of the tandem cold rolling. ISA transactions. 2023 Jan 1;132:353-63.

[152] Nyangaresi VO. Target Tracking Area Selection and Handover Security in Cellular Networks: A Machine Learning Approach. InProceedings of Third International Conference on Sustainable Expert Systems: ICSES 2022 2023 Feb 23 (pp. 797-816). Singapore: Springer Nature Singapore.

[153] Lillicrap TP, Santoro A, Marris L, Akerman CJ, Hinton G. Backpropagation and the brain. Nature Reviews Neuroscience. 2020 Jun;21(6):335-46.

[154] Calisir T, Çolak AB, Aydin D, Dalkilic AS, Baskaya S. Artificial neural network approach for investigating the impact of convector design parameters on the heat transfer and total weight of panel radiators. International Journal of Thermal Sciences. 2023 Jan 1;183:107845.

[155] Liu B, Li F, Wang X, Zhang B, Yan J. Ternary weight networks. InICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2023 Jun 4 (pp. 1-5). IEEE.

[156] Yi SI, Kendall JD, Williams RS, Kumar S. Activity-difference training of deep neural networks using memristor crossbars. Nature Electronics. 2023 Jan;6(1):45-51.

[157] Linka K, Pierre SR, Kuhl E. Automated model discovery for human brain using Constitutive Artificial Neural Networks. Acta Biomaterialia. 2023 Apr 1;160:134-51.

[158] Nyangaresi VO, Rodrigues AJ, Abeka SO. Neuro-fuzzy based handover authentication protocol for ultra dense 5G networks. In2020 2nd Global Power, Energy and Communication Conference (GPECOM) 2020 Oct 20 (pp. 339-344). IEEE.

[159] Kaur M, Mohta A. A review of deep learning with recurrent neural network. In2019 International Conference on Smart Systems and Inventive Technology (ICSSIT) 2019 Nov 27 (pp. 460-465). IEEE.

[160] Abiodun OI, Jantan A, Omolara AE, Dada KV, Umar AM, Linus OU, Arshad H, Kazaure AA, Gana U, Kiru MU. Comprehensive review of artificial neural network applications to pattern recognition. IEEE Access. 2019 Oct 4;7:158820-46.

[161] Almutairi MS, Almutairi K, Chiroma H. Hybrid of deep recurrent network and long short term memory for rear-end collision detection in fog based internet of vehicles. Expert Systems with Applications. 2023 Mar 1;213:119033.

[162] Ruma JF, Adnan MS, Dewan A, Rahman RM. Particle swarm optimization based LSTM networks for water level forecasting: a case study on Bangladesh river network. Results in Engineering. 2023 Mar 1;17:100951.

[163] Naeem M, Mashwani WK, Abiad M, Shah H, Khan Z, Aamir M. Soft computing techniques for forecasting of COVID-19 in Pakistan. Alexandria Engineering Journal. 2023 Feb 1;63:45-56.

[164] Nyangaresi VO, Rodrigues AJ, Abeka SO. Secure Handover Protocol for High Speed 5GNetworks. International Journal of Advanced Networking and Applications. 2020Mar; 11(06): 4429-4442.

[165] Pang B, Nijkamp E, Wu YN. Deep learning with tensorflow: A review. Journal of Educational and Behavioral Statistics. 2020 Apr;45(2):227-48.

[166] Zhang Z, Li J. A Review of Artificial Intelligence in Embedded Systems. Micromachines. 2023 Apr 22;14(5):897.

[167] Inizan TJ, Plé T, Adjoua O, Ren P, Gökcan H, Isayev O, Lagardère L, Piquemal JP. Scalable hybrid deep neural networks/polarizable potentials biomolecular simulations including long-range effects. Chemical Science. 2023.

[168] Erak O, Abou-Zeid H. Accelerating and Compressing Deep Neural Networks for Massive MIMO CSI Feedback. arXiv preprint arXiv:2304.01914. 2023 Jan 20.

[169] Rościszewski P, Krzywaniak A, Iserte S, Rojek K, Gepner P. Adaptation of AI-accelerated CFD Simulations to the IPU Platform. InParallel Processing and Applied Mathematics: 14th International Conference, PPAM 2022, Gdansk, Poland, September 11–14, 2022, Revised Selected Papers, Part II 2023 Apr 27 (pp. 223-235). Cham: Springer International Publishing.

[170] Nyangaresi VO, Abeka SO, Rodgrigues A. Secure timing advance based context-aware handover protocol for vehicular ad-hoc heterogeneous networks. International Journal of Cyber-Security and Digital Forensics. 2018 Sep 1;7(3):256-75.

[171] Almeida TA, Hidalgo JM, Yamakami A. Contributions to the study of SMS spam filtering: new collection and results. InProceedings of the 11th ACM symposium on Document engineering 2011 Sep 19 (pp. 259-262).

[172] Carrizosa E, Restrepo MG, Morales DR. On clustering categories of categoricalpredictors in generalizedlinearmodels. Expert Systemswith Applications. 2021 Nov 15;182:115245.

[173] Diao E, Ding J, Tarokh V. Restricted recurrent neural networks. In2019 IEEE International Conference on Big Data (Big Data) 2019 Dec 9 (pp. 56-63). IEEE.

[174] Wang J, Hu Y, Joseph K. NeuroTPR: A neuro-net toponym recognition model for extracting locations from social media messages. Transactions in GIS. 2020 Jun;24(3):719-35.

[175] Badillo S, Banfai B, Birzele F, Davydov II, Hutchinson L, Kam-Thong T, Siebourg-Polster J, Steiert B, Zhang JD. An introduction to machine learning. Clinical pharmacology & therapeutics. 2020 Apr;107(4):871-85.

[176] Debie E, Shafi K. Implications of the curse of dimensionality for supervised learning classifier systems: theoretical and empirical analyses. Pattern Analysis and Applications. 2019 May 1;22:519-36.

[177] Barushka A, Hajek P. Spam filtering using integrated distribution-based balancing approach and regularized deep neural networks. Applied Intelligence. 2018 Oct;48:3538-56