International Journal of Emerging Science and Engineering (IJESE)
ISSN: 2319–6378 (Online), Volume-10 Issue-7, June 2022

Network Intrusion Detection Systems: A Systematic Literature Review of Hybrid Deep Learning Approaches

Stephen Kahara Wanjau, Geoffrey Mariga Wambugu, Aaron Mogeni Oirere

Abstract*: *Network Intrusion Detection Systems (NIDSs) have become standard security solutions that endeavours to discover unauthorized access to an organizational computer network by scrutinizing incoming and outgoing network traffic for signs of malicious activity. In recent years, deep learning based NIDSs have emerged as an active area of research in cybersecurity and several surveys have been done on these systems. Although a plethora of surveys exists covering this burgeoning body of research, there lacks in the literature an empirical analysis of the different hybrid deep learning models. This paper presents a review of hybrid deep learning models for network intrusion detection and pinpoints their characteristics which researchers and practitioners are exploiting to develop modern NIDSs. The paper first elucidates the concept of network intrusion detection systems. Secondly, the taxonomy of hybrid deep learning techniques employed in designing NIDSs is presented. Lastly, a survey of the hybrid deep learning based NIDS is presented. The study adopted the systematic literature review methodology, a formal and systematic procedure by conducting bibliographic review, while defining explicit protocols for obtaining information. The survey results suggest that hybrid deep learning-based models yield desirable performance compared to other deep learning algorithms. The results also indicate that optimization, empirical risk minimization and model complexity control are the most important characteristics in the design of hybrid deep learning-based models. Lastly, key issues in the literature exposed in the research survey are discussed and then propose several potential future directions for researchers and practitioners in the design of deep learning methods for network intrusion detection.

Keywords*: *Complexity Control, Empirical Risk Minimization, Hybrid Deep Learning, Network Intrusion Detection, Optimization.

I. INTRODUCTION

The increasing growth and popularity of the Internet coupled with the advent of the information age has revolutionized all aspects of our lives [1].

Manuscript received on 11 April 2022.
Revised Manuscript received on 20 May 2022.
Manuscript published on 30 June 2022.
* Correspondence Author
Stephen Kahara Wanjau*, School of Computing and Information Technology, Murang'a University of Technology, Murang'a, Kenya. Email: steve.kahar@gmail.com
Geoffrey Mariga Wambugu, School of Computing and Information Technology, Murang'a University of Technology, Murang'a, Kenya. Email: gmariga@mut.ac.ke
Aaron Mogeni Oirere, School of Computing and Information Technology, Murang'a University of Technology, Murang'a, Kenya. Email: amogeni@mut.ac.ke

Today, the Internet is an essential part of modern technology pertaining to transfer of data and information, thus necessitating a secure global network. Even though the Internet has given us considerable convenience, it has also brought about a multiplicity of network security threats, such as brute force attacks, Denial-of-Service, phishing, malware, Man-in-the-Middle, backdoors, and rootkits. These attacks can cause economic losses, loss of user privacy, loss of sensitive data such as business plans, loss of corporate reputation and even threatening national security [2]. Such threats have motivated researchers to design novel Network Intrusion Detection System (NIDS) that monitor, analyse and classify real-time network attacks and provide countermeasures to various network intrusions [3].

The enormous, high-dimensional network traffic and how to precisely detect anomalous traffic is the principal task of intrusion detection systems. Two main categories of NIDSs based on the detection approach exists in the literature including the signature-based (misuse) and anomaly-based (user-behaviour) intrusion detection system [4]. Anomaly-based NIDS are more striking for the research and practitioners as they are capable of detecting any deviation from the normal traffic pattern. As a result of recognizing differences in the features obtained from the network traffic, they are capable of efficiently detecting and intercepting network attacks in advance thus, effectively minimising the losses normally caused by network attacks [5]. In the recent years, machine learning approaches such as support vector machine, Naïve Bayes and decision trees have been used to develop effective NIDSs [6], [7]. However, application of these approaches is based on manually extracted features that may cause loss of the original flow information. In addition, these approaches have shown inefficiencies in detecting zero-day attacks, output high false positive alarms and are known to incur high computational costs thus reducing their implementation in real-time in actual situations [8].

Researchers have explored the possibility of deep learning techniques in the development of NIDS [9], [10]. The architecture comprising of deep learning and network intrusion detection has turn into a trending research topic in today's network security domain. This has been occasioned by the significant ability of deep learning methods to automatically draw out discriminatory feature representations from voluminous high-dimensional amounts of data to generate models with better generalization capabilities [1].

In addition, over the last decade we have witnessed the discovery of powerful graphics processor units (GPUs) that supports processing voluminous and data [11].

Several studies have been reported that apply deep learning for intrusion detection systems achieving better performance by learning valuable information from big data [12]. Further, in the literature, researchers have proposed several hybrid deep learning models in the design of effective network intrusion detection systems [8], [13]. Convolutional neural network (CNN) and recurrent neural network (RNN) are gaining prominence in recent years in the network intrusion detection domain. For instance, CNN methods have demonstrated success in converting one-dimensional network traffic into two-dimensional grayscale images and then utilise the convolutional kernel to draw out definite features from the network traffic with the objective of increasing the detection rate and reducing false alarms.

In this paper, we survey and analyse the hybrid deep learning based network intrusion detection models in the current literature covering the period between 2017 and 2021. The study also highlights their characteristics which researchers and practitioners are exploiting to develop modern day NIDSs. The rest of this paper is organized as follows: Section 2 provides a brief background to Network intrusion detection and the deep learning approach in the design of network intrusion detection systems. Section 3 presents the methodology used to give an in-depth analysis of the existing hybrid deep learning models in literature and presents their characteristics for network intrusions detection. The results and discussion of the findings are reported in section including a discussion and details the open research issues identified from the analysis. The paper conclusion and future research work is discussed in the last section.

II. NETWORK INTRUSION DETECTION SYSTEMS

Intrusion Detection Systems is defined as the "automation of intrusion detection process of finding events of violation of security policies or standard security practices in computer networks" [14]. Network Intrusion Detection Systems are designed to recognize different types of malicious network traffic and computer misuse, unable to be detected by a traditional firewall [15]. They are usually located in-side the network to monitor all incoming traffic, document existing threats and deter adversaries. Denning [16] pioneered the proposition of developing intrusion detection systems by utilizing Artificial Intelligence methods on security events to detect abnormal usage patterns and intrusion. Since then machine learning techniques are being employed as a conventional approach to developing network intrusion detection systems. Since 2010s, with the popularity of big data, high performance computing and cloud computing, NIDSs have received enormous attractions as a border checkpoint for the security of a network besides the development of machine learning paradigm to obtain better performance [17].

A. Taxonomy of Network Intrusion Detection Systems

Liao, et al. [15] generally classified Intrusion Detection Systems (IDS) into two categories, namely, Anomaly-based Detection, and Signature-based Detection while [18] added Stateful Protocol Analysis. Signature-based detection, similarly well-known as misuse-detection, techniques identify an intrusion by matching patterns or signatures already in existence within the database with the event (either attack or intrusion) that was going on, the idea in which intrusion using the same pattern will be discovered. In other words, when there is a match between an intrusion signatures with that of a previous intrusion that already exists in the signature database, an alarm signal is triggered [19]. This technique is widely used in commercial products due to its predictability and precision. Nonetheless, for this method to be effective, it is essential to keep the database of signatures up-to-date. The weakness of this method is that it doesn't recognize new non-existent attacks in the database [20]. Further, the increasing rate of zero-day attacks [21] has rendered signature-based detection techniques increasingly less effective since no previous existence of any such attacks signature. Khraisat, et al. [4] in their survey describes anomaly-based intrusion detection in networks as the action of identifying exceptional patterns in network traffic that do not conform to the predictable normal behaviour. These non-conforming patterns are frequently denoted as anomalies, exceptions, outliers, surprises, peculiarities, aberrations or discordant observations in various application domains [22]. A normal model of the behaviour of an information system can be generated using knowledge-based, machine learning, or statistical-based methods. Any deviation noted between the model and the observed behavior is regarded as an anomaly, which is normally interpreted as an intrusion. Anomaly-based detection methods are also referred as behavior-based detection methods. Stateful Protocol Analysis, also known as specification-based detection, depend on ordinary profiles for specific protocols defined by the vendors. Normally, these network protocol models are usually grounded on standards of protocols from international organizations. Stateful Protocol Analysis acts on the network layer, transport layer, and the application layer making it more powerful than the above two methods [23].

B. Deep learning Approach to Designing Network Intrusion Detection Systems

Today, deep learning, a class of machine learning algorithms that uses artificial neural networks with multiple layers of nonlinear processing units to learn data representations is becoming a major contributor of the contemporary rise of Artificial Intelligence (AI) in nearly all walks of life [24], [25]. In recent years, deep learning models with multilayer processing architecture are presenting better performance compared to shallow learning or traditional machine learning algorithms [26]. Deep learning enables a neural network to learn hierarchies of information akin to the function of the human brain. The main benefit of deep learning is the colossal flexibility in designing each part of the architecture, resulting in several ways of discovering the most efficient activation functions or learning algorithms [27], [28].

Retrieval Number: 100.1/ijese.F25300510622
DOI: 10.35940/ijese.F2530.0610722
Journal Website: www.ijese.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

A significant principle of deep learning is work out higher-level features from lower-level ones from observational data [29]. Deep learning is mainly based on neural networks whose essential part is a neuron with a set of weights (w), an activation function (σ) and a set of biases (b). Based on these parameters, transformation can be expressed as follows:

$$a = \sigma\,(w^T x + b) \quad (1)$$

Where, x is the inputs of neurons, and T is matrix transpose. Deep learning approaches may be classified into three categories depending on architecture and techniques namely; discriminative (supervised), generative (unsupervised) and hybrid combining two (or more) methods [30].

i) Discriminative/Supervised Deep Learning Models

Consider a classification task where we want to determine whether an email is either a spam or not given a set of words present in a particular email, where, labels: $Y = y$ and the features: $X = \{x_1, x_2, x_3, \ldots, x_n\}$. A joint distribution of the model can be denoted as:

$$\mathrm{p(Y,X)} = \mathrm{P(y, x_1, x_2, x_3, \ldots, x_n)} \quad (2)$$

A discriminative architecture models the decision boundary between the classes by learning the conditional probability distribution $p(y|x)$ [31] as shown in figure 1. The discriminative models makes predictions on the unseen data based on conditional probability which may be used either for classification or regression tasks. They are described as 'deep' since they use layers of latent or hidden variables. Discriminative models tend to outperform their generative counterparts in supervised tasks. However, they cannot learn from unlabelled data [32]. In addition, these models incapable of generating new data points. As a result, the definitive objective of discriminative models is to separate one class from another. Deep discriminative models include three approaches, namely, deep neural networks, recurrent neural networks, and convolutional neural networks. Discriminative models are computationally cheap while compared to generative models as well as have the advantage of being more robust to outliers, unlike the generative models.

Figure 1: Discriminative/supervised models decision boundary (Source**:** *[33]*)

ii) Generative/Unsupervised Deep Learning Models

A generative architecture conspicuously models the actual distribution of respective classes and learns the joint probability distribution $p(x, y)$ in addition to determining $p(y|x)$ indirectly via the Bayes Theorem [31]. In other words, generative models focus on the distribution of a dataset to return a probability for a given example. They are regarded as a class of statistical models that can generate new data instances and use probability estimates and likelihood to model the data points to differentiate between different classes labels present in a dataset [33] as shown in figure 2.

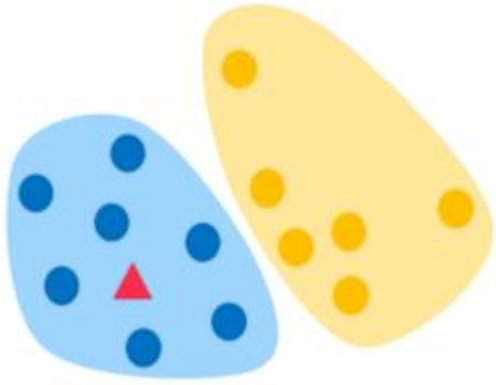

Figure 2: Generative/unsupervised models decision boundary (Source**:** *[33]*)

Generative models can be used to learn representations, to handle exploration/exploitation trade-offs, and to make use of the large amounts of unlabelled data. Deep generative models can generate input examples from the feature learned by the model, which gives a way to understand the model behaviour [34]. However, these models are significantly affected by the presence of outliers. The aim is to combine the interpretable representations and quantified uncertainty offered by probabilistic models, with the flexibility and scalable learning of deep neural networks. As such, generative models are utilized in unsupervised learning in a scheme to perform tasks such as Likelihood and Probability estimation, modelling data points, define the phenomenon in data, and discriminate between classes based on these probabilities [33]. The generative models consists mainly of four approaches namely, restricted Boltzmann machine, deep auto encoders, deep belief networks, and generative adversarial networks.

iii) Hybrid/Ensemble Deep Learning Models

The hybrid deep learning, also called ensemble learning approaches are a progressive method that combine multiple learning algorithms to take excellent properties of each algorithm to obtain better generalization performance result [35]. Hybrid deep learning techniques provide more accurate and less computationally expensive solutions. In the literature, most of the deep hybrid deep learning models refers to an architecture that makes use of both generative and discriminative components. For instance, a hybrid model combining CNN that explore and learn spatial features from images and LSTM to learn and depict temporal patterns. Figure 3 shows a visual representation of a hybrid deep learning model combining CNN and LSTM.

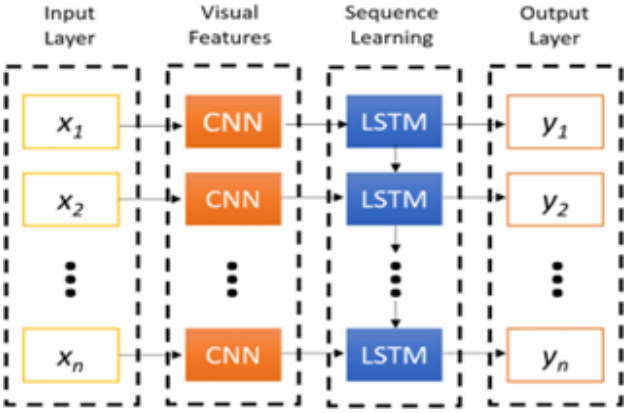

Figure 3: A Visual representation of an architecture of the CNN-LSTM network (source: *[36]*)

In ensemble modelling, the combination may be implemented by aggregating the output from each algorithm with two main objectives: reducing the model error and maintaining its generalization. The technique to implement such an aggregation may be realized using some techniques such as voting, averaging, boosting, bagging, stacking and Negative correlation learning [26].

III. METHODOLOGY

In this paper, we conducted a systematic literature review of the hybrid deep learning-based NIDS and investigated published journal articles 2017 and 2021. Systematic literature review is a methodology normally followed by researchers to identify, examine, and extract essential information from the literature related to particular research topics [37]. The systematic literature review was conducted in two phases: First, we identified the information resource (search engine) and keywords and executed a query for obtaining an initial list of published articles. Secondly, the study applied a specific criteria on the initial list to choose the best articles, store them into a final list for the review. Figure 4 shows the methodology adopted.

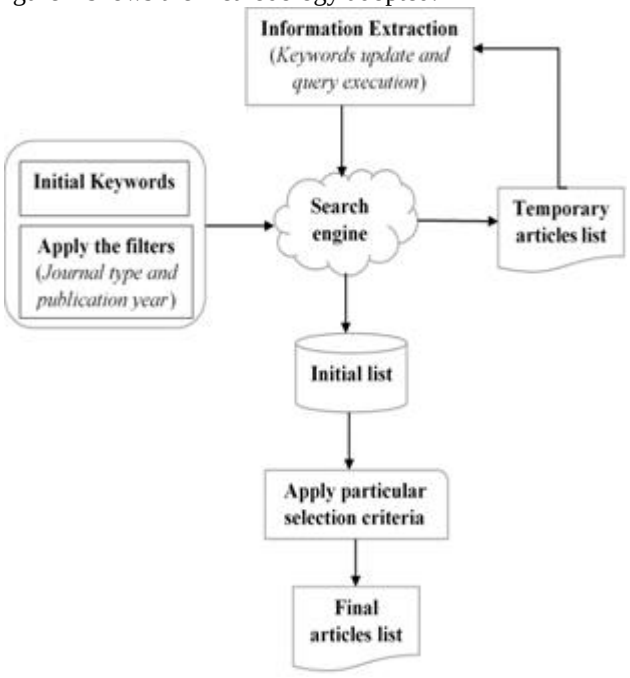

Figure 4: Methodology used for the study

A. Objectives of the review

The main objective of the review was to synthesize available research on hybrid deep learning approaches for NIDS design. Through the systematic literature review, the study attempted to answer the following research questions:

i. What are the recent hybrid deep learning approaches adopted for the design NIDS?
ii. What are the salient characteristics of hybrid deep learning-based NIDS? and
iii. What is the future scope of research in hybrid deep learning-based NIDS?

B. Review Protocol

The preparation of a review protocol is a vital constituent a systematic review process. It demonstrates that a systematic review is thoughtfully organized and that which is planned is evidently documented before the review starts. Essentially, this promotes consistency in conducting the review, research integrity, accountability and transparency of the eventual completed review. The study adopted the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) developed by [38] as the review protocol. PRISMA entails a 17-item checklist that is envisioned to aid the preparation and reporting of a robust protocol for the systematic review.

C. Information Sources

Since the existing data in the internet is enormous, we considered three search engines namely, Google Scholar, Worldwide Science and Core owing to their capacity to search from almost all the well-known databases. The documents generated were journal articles and conference papers. Selection of the work considered was based on the relevance of the title, abstract and the full article.

D. Search Strategy

This section summaries the process used to generate the search terms, the searching strategy, and the search documentation. As shown in figure 1, first we identified the search engine and keywords for the article search so that we answer the research question in a meaningful way. The study executed a search query using the initial keyword "network intrusion detection" and adjusted the filter to restrict and obtain the journal articles published between the years 2017 and 2021.

The search query was further refined to include other keywords as signature-based network intrusion detection and network anomaly detection with the combination of hybrid deep learning to get additional relevant articles. The gathered articles were stored as an initial list. The gathered papers were then scanned for inclusion and exclusion based a search criteria to determine relevant articles for further analysis. Based on the criteria, final list was largely completed afterwards, by using a backward and forward snowballing strategy that consists of using the reference list of the selected papers and the citations to these papers to identify additional papers.

Retrieval Number: 100.1/ijese.F25300510622
DOI: 10.35940/ijese.F2530.0610722
Journal Website: www.ijese.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

E. Selection Criteria

An inclusion and exclusion criteria ensure that only relevant studies are incorporated in data analysis. Selection of articles for inclusion in the SLR charted a three-stage process that entails: (a) initial selection of articles by considering the title; (b) selection of articles based upon reading the abstract; and (c) further selection of articles after reading through the papers. The PRISMA flow chart that describes the selection of the articles was used and is shown in figure 5. In summary, the following criteria was used for inclusion and exclusion:

a) Include those articles which were written in the English language
b) Include articles based on hybrid deep learning techniques.
c) Include journal articles that are cited and fully referenced.
d) Include articles published between the period January 2017 and December 2021.
e) Exclude conference papers, case studies, reviews and survey articles.
f) Exclude articles not categorized as peer-reviewed.
g) Exclude document's digital object identifier (DOI) equivalent to the DOI of another document.

F. Quality Assurance

Quality systematic reviews demand good quality literature searches as well as accuracy in reporting. A checklist was prepared for use to evaluate the quality of each paper with only papers that were deemed to meet the evaluation criteria included in the systematic review. An analysis of the quality of the papers selected after applying the inclusion/exclusion criteria was carried out.

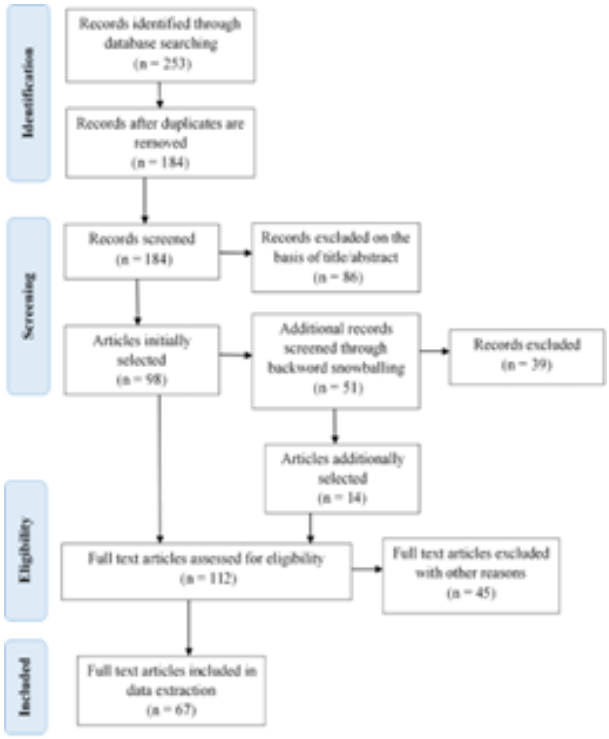

Figure 5: PRISMA flow chart for the selection of the articles

G. Data Extraction and Synthesis

The authors created a spreadsheet such that each row of the spreadsheet represented an article published in the respecting academic databases for the years 2017–2021 inclusive. For specific articles, relevant bibliographic information was captured (e.g., title, author, database, page numbers) together with a hyperlink to the article. Primarily, as a key part of the first phase of this analysis, the abstract for every article was captured. The inclusion of an article for the final review followed a two-phase approach. The first phase entailed reading the abstract and coding the article as either relevant or not to the review. An article's relevance was determined by rating the information contained in the abstract against the search terms identified above. The columns considered in the review were as follows: the authors, publication date, article type (e.g. journal), and technique-based taxonomy. Retrieval of this information was related to the research questions.

IV. RESULTS AND DISCUSSION

In this study, we focused on analyzing hybrid deep learning based approaches for designing network intrusion detection through a systematic literature review. This review considered journal articles and conference papers published between 2017 and 2021. From the search, journal articles and conference papers containing the terms hybrid deep learning, and network intrusion detection were retrieved from the identified databases. Further, though snowballing search other relevant journal articles and conference papers were identified and added even if they were not from the identified databases. Table 1 shows the articles retrieved for analysis.

Table 1: Relevant journal articles retrieved

Search Engine	Initial List of articles	Final List of articles
Google Scholar	101	32
World Wide Science	98	14
Core	54	9
Snowballing	-	12
	253	**67**

Out of the 67 papers synthesized, 49 were journal articles and 18 were conference papers. Figure 6 shows the distribution of the articles on hybrid deep learning-based NIDS over the years. We observed that most of the research on hybrid deep learning-based NIDS were published in journal articles between the years 2020 and 2021.

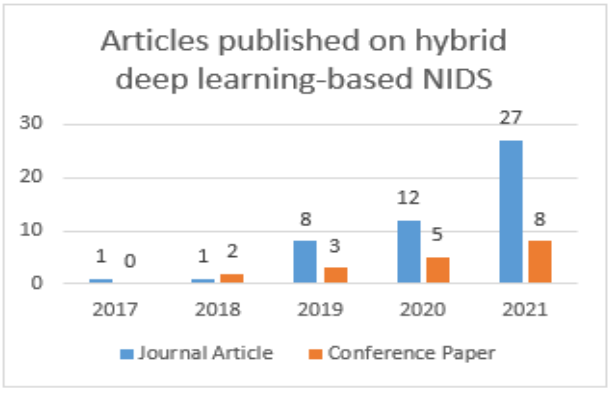

Figure 6: Distribution of articles on hybrid deep learning-based NIDS over the years

Retrieval Number: 100.1/ijese.F25300510622
DOI: 10.35940/ijese.F2530.0610722
Journal Website: www.ijese.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

A. Hybrid Deep Learning Based NIDS

The authors sort to establish the recent hybrid deep learning approaches adopted for the design NIDS. Several hybrid deep learning methods have been proposed for deployment in network intrusion detections systems. Hybrid architectures incorporate both generative and discriminative models. Also referred as ensemble architectures, they combining different algorithms so as to build capacity to fill the gaps of network intrusion detection models and get the best results. Table 2 presents the summary of the techniques and area of intrusion detection focused from the papers reviewed. Hybrid deep learning-based NIDSs can be built in different ways and for quite different purposes. The following describes the various approaches available.

i). Algorithmic Hybrid Models

In algorithmic deep learning models, techniques or algorithms are hybridized to perform a single task together. For example, Alqahtani [39] developed a novel hybrid optimized long short-term memory (LSTM) model whereby firefly swarm optimization was integrated with LSTM to reduce the computational overhead, which in turn increases the prediction accuracy. D dataset. An architectural model presented by Zeng et al. [40] proposed a hybrid neural network with a stack auto encoder to evaluate the traffic features and selected the best feature vectors from the network traffic as labels. Ma et al. [41] developed a hybrid intrusion detection system for the evaluation of multiple types of flow features using a hybrid one-dimensional convolutional network and evaluated in real-time the efficacy of the proposed system using the ISCX-IDS-2012 and CIC-IDS-2017 datasets. In a study by ElSayed, et al. [42] a hybrid deep learning approach based on the CNN was used to classify the flow traffic into either normal or malicious attack classes in Software-defined network environment.

ii). Cooperative Hybrid Models

Diverse techniques are utilized to conduct a variety of independent tasks that are then combined in some way to yielding a holistic system. For instance, Megantara and Ahmad [43] proposed one technique for both signature-based detection and anomaly-based detection. In their work, the signature based detection employs a known set of rules (indicators) from the system attack database to postulate whether an activity is malicious or not, whereas anomaly-based detection recognises the attack based on uncommon user behaviour patterns. In case users perform unusual actions or activities, then it can be flagged as an attack. The work of Yao, *et al* [44] proposed an intrusion detection model founded on the cross layer feature fusion of a LSTM and CNN networks for Advanced Metering Infrastructure. In this cooperative arrangement, the CNN component identifies regional features to capture global features, whereas the LSTM component capture periodic features using the memory function.

iii). Hierarchical Hybrid Models

In a hierarchical architecture, the IDS includes different methods that accomplish different tasks at each level. For instance, Khan [45] proposed a convolutional recurrent neural network (CRNN) that was employed to build a deep learning-based hybrid intrusion detection framework that can predict and classify malicious cyberattacks in the network. The study of [46] and the model presented by [47] demonstrate the hierarchical hybrid deep learning models for network intrusion detection. The study by Wang, et al. [46] presented a hierarchical hybrid deep learning architecture that incorporated feature representation learning and dimensionality reduction of network traffic features to improve the efficiency and effectiveness of the resultant model.

In [48] the authors proposed a hybrid model dubbed STDeepGraph that was designed based on identifying the flow-to-flow resemblance of network communication graphs. In this case, the model was a hierarchical combination of a CNN and LSTM with graph similarity measures capable of learning high-dimensional representations from the network traffic. The approach exploited graph structures as supplementary prior information, feature extraction via graph Laplacian matrix, and the hybrid deep learning algorithms to learn long-term information on communication graphs.

Table 2: Summary of Hybrid Deep Learning Based NIDS

Ref.	Year	Deep Learning Algorithms	Intrusion Detection Approach	Dataset Used
[49]	2020	CNN and a weight-dropped, LSTM (WDLSTM)	Intrusion detection in big data environments	UNSW-NB15
[50]	2020	CNN and LSTM	Analyze network traffic information of network raw dataset from both spatial and temporal dimensions	CICIDS2017
[51]	2021	Feed Forward, LSTM, and Gated Recurrent Unit	Analyze network traffic information network raw dataset	Kyoto Honeypot Dataset
[35]	2021	CNN and LSTM	Intrusion detection system for imbalanced dataset on big data environment	CIDDS-001 (for multiclass),UNS-NB15 (for binary classification)
[52]	2020	CNN and LSTM	Improving detection accuracy in NIDSs	NSL KDD.
[53]	2020	CNN and LSTM	Cuda-enabled DL-driven architecture that makes use of the predictive power of LSTM and CNN for timely and efficient detection of multi-vector threats and attacks in Software defined network (SDN)	CICIDS2017
[54]	2021	weighted deep belief network (HW-DBN) with improved Gaussian–Bernoulli restricted Boltzmann machine and weighted deep neural networks (WDNN)	Enhancing IoT network intrusion detection	CICIDS2017

[55]	2021	2-D CNN, (RNN) and (MLP)	For detection of 9 Cyber Attacks versus normal flow.	Kitsune Network attack dataset
[56]	2021	CNN and GRU	Detection of anomalous activities in Internet of Things (IoTs) networks	BoT-IoT, IoT Network Intrusion, MQTT-IoT-IDS2020, and IoT-23
[57]	2021	DNN and LSTM	Improve detection rate and reduce false positive rate in NIDSs	KDD CUP 99
[58]	2021	Hybrid-DBN	Secure network by controlling network traffic in Industrial control systems (ICS).	actual data of the industrial control system in the water tank
[59]	2020	CNN and LSTM	An architecture normalizing UTF-8 character encoding for Spatial Feature Learning so as to sufficiently find the characteristics of real-time HTTP traffic without calculating entropy, encryption, and compression	CSIC-2010, CICIDS2017, fixed real-time data
[60]	2021	CNN and LSTM	A hybrid CNN-LSTM model to detect brute-force attacks in encrypted traffic in protocols such as SMTPS, IMAPS, HTTPS, FTPS, and SSH.	Real-world traffic from a Tor exit node on the Internet.
[61]	2021	Long short-term memory autoencoder (LAE) and deep bidirectional long short-term memory (BLSTM)	A scheme that reduce the feature dimensionality of large-scale IoT network traffic data	BoT-IoT
[62]	2019	Conv-LSTM	Two-stage learning platform for both Anomaly-based and Signature-base classification approaches network intrusion detection	ISCX-UNB
[63]	2021	LSTM an Autoencoder (AE)	cyber-attacks Intrusion Detection in connected and autonomous vehicles (CAVs)	NSL-KDD
[64]	2019	LeNet-5 (CNN) and LSTM	A deep hierarchical network that incorporates improved to learn both the spatial and temporal features from original network flow and used to detect abnormal flow.	CICIDS2017, CTU dataset
[65]	2020	A Stacked Autoencoder and Feed forward neural network	Detection of normal and abnormal behaviour in networks	CICIDS2017
[66]	2018	DBN and Probabilistic Neural Network (PNN).	Improve the detection rate and classification accuracy of NIDSs	NSL-KDD
[67]	2021	unsupervised Sparse autoencoder (SAE) with smoothed l1 regularization and DNN	A two-stage hybrid intrusion detection scheme that improves overall performance in detection rate and low false positive rate.	KDDCup99, NSL-KDD and UNSW-NB15.
[68]	2020	Spider monkey optimization (SMO) algorithm and DNN	improve the detection rate and classification accuracy of NIDSs	NSL-KDD and KDD Cup 99
[69]	2021	Cuda Deep Neural Network Long Short-Term Memory (CuDNNLSTM) and LSTM	A hybrid deep learning approach to identify attacks in networks.	Kitsune Network attack dataset
[70]	2021	CNN and DNN	A Forward Feature Selection (FFS) method for detecting distributed denial of service attacks in Software Defined Networks.	CICIDS2017
[71]	2021	hybrid rule-based feature selection and deep feedforward neural network model	A deep learning-based intrusion detection paradigm for Industrial Internet of Things Network with hybrid rule-based feature selection to train and verify information captured from TCP/IP packets.	NSL-KDD and UNSW-NB15.
[72]	2021	Autoencoder with improved genetic algorithm (IGA-BP)	Address problems of slow detection rate and easy to get into local optimality for NIDSs	KDD CUP99
[73]	2019	improved restricted Boltzmann machine and gradient descent-based SVM	Anomaly detection architecture for suspicious flow detection in the context of social multimedia in Software defined networks (SDN)	Carnegie Mellon University (CMU)-based insider threat dataset
[74]	2021	Cuda-deep neural network, gated recurrent unit (Cu- DNNGRU), and Cuda-bidirectional long short-term memory (Cu-BLSTM	An SDN-enabled deep-learning-driven framework for threats detection in an IoT environment	CICIDS2018
[75]	2020	Convolutional-auto encoder (Conv-AE)	Misuse attack detection in NIDSs	CSE-CIC-IDS2018
[76]	2019	Grey wolf optimization (GWO) and CNN	Improved network anomaly detection accuracy in NIDSs	DARPA'98 and KDD'99
[77]	2021	Multi-head attention mechanism and a skip-LSTM	A better generalizability model named HALNet for detection of Command and Control (C&C) malwares	CCE2021 and CICIDS2017
[78]	2021	LSTM, CNN), and SVM	A hybrid semantic deep learning (HSDL) model for secure cloud storage and intrusion detection	NSL-KDD and UNSW-NB15
[79]	2021	Hybrid LSTM	SDN-enabled architecture used to detect sophisticated cyber-attacks in fog-to-IoT environment as well as identify new attacks targeting IoT devices and other threats	Coburg Intrusion Detection Data Set (CIDDS-001) flow-based dataset
[80]	2021	CNN and GRU	Model for detection of anomalies in encrypted network traffic	NSL-KDD, UNSW-NB15, and CIC-IDS-2017

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

[81]	2019	Ensemble CNN	An Ensemble architecture for efficient detection of DDoS in SDNs.	ISCX IDS 2012
[82]	2021	CNN and LSTM	Achieve high accuracy, thus improving the detection rate in NIDSs	DARPA and CIC-IDS2018
[83]	2020	CNN and LSTM with Jumping Gene adapted NSGA-II multi-objective optimization	Improving DDoS Attacks detection in IoT Networks	CISIDS2017
[84]	2020	Deep Sparse AutoEncoder (DSAE), DNN and LSTM	Ensemble model for network anomaly and cyber-attack detection in IoT environment	IoT-23, LITNET-2020, and NetML-2020.
[85]	2021	CNN and LSTM for binary classification; CNN and gate recurrent unit (GRU) for multiclass classification)	Achieve high accuracy, thus improving the detection rate in NIDSs	ISCX IDS 2012
[86]	2019	Bloomfilter and k-NN	Anomaly detection in gas pipeline supervisory control and data acquisition (SCADA) systems among ground devices in industry control systems	Mississippi State University's in-house SCADA dataset
[87]	2021	CNN and LSTM	Combines base CNN and LSTM as an ensemble approach and used the Stacking algorithm in combination with a neural network as a meta learner to detect network intrusions	CCD-IDSv1
[88]	2020	Dendritic Cell Algorithm (DCA) and Self Normalizing Neural Network (SNN	A DeepDCA model for detection of IoT Attacks Using Artificial Immune System, classify IoT intrusion and reduce the false alarm generation	IoT-Bot
[48]	2019	CNN and LSTM	Learn high-dimensional representations from the network traffic. The technique utilize graph structures as the supplementary prior information, feature extraction using the graph Laplacian matrix for long-term information learning on communication graphs	UNSW-NB15 and CICIDS-2017
[89]	2021	DBN-LSSVM	Achieve high accuracy, thus improving the detection rate detection rates for known attack types in NIDSs	KDD CUP 99
[90]	2021	CNN and Bi-LSTM	A new deep learning-based hyperparameter search (HPS) model known as HPS-CBL developed for intrusion detection in big data environments.	UNSW-NB15
[91]	2021	CNN and LSTM	A SDN-enabled framework for timely and efficient detection of sophisticated Internet of Medical Things (IoMT) malwares	IoT-23
[92]	2021	CNN and LSTM	Masquerade attack detection using Schonlau data set and the Green-berg datasets	Greenberg and Schonlau
[93]	2021	DBN and deep auto encoders	Proposed a SecureDeepNet-IoT model for detecting IoT-based intrusions.	UNSW-NB15
[94]	2021	CNN and LSTM	detect botnet attacks, namely, Mirai and BASHLITE, on nine commercial IoT devices	Real N-BaIoT dataset extracted from a real system
[95]	2018	CNN, LSTM and DNN	A model for extracting spatial-temporal information from raw web traffic data.	Yahoo! Webscope S5
[96]	2020	CNN and weight-dropped LSTM (WDLSTM)	Achieve high accuracy, thus improving the detection rate detection rates for known attack types in NIDSs	IDS big data
[97]	2020	CNN and conventional learning classifier system (LCS)	detecting database intrusion via insider attack based on the RBAC mechanism with Feature selection using Genetic Algorithms	Synthetic query dataset based on the RBAC mechanism
[98]	2021	Model fusion of two DNNs	Anomaly detection and classification of various network attacks in large-scale and highly imbalanced traffic dataset	ZYELL's real-world dataset
[99]	2021	CNN and RNN	A hybrid framework that captures local and temporal features to predict and classify malicious cyberattacks in the network	CSE-CIC-DS2018
[100]	2019	CNN and RNN	Network-based payload classification approach without feature engineering to support and improve accuracy and detection rate end-to-end.	DARPA1998
[101]	2019	improved conditional variational Auto Encoder (ICVAE) and DNN	Improving the detection rate of imbalanced attacks by automatically reducing data dimension, learning and exploring potential sparse representations between network data features and classes	NSL-KDD and UNSW-NB15
[102]	2020	Adversarial Auto-encoder (AAE) and Generative Adversarial Nets (GAN	Employs semi-supervised learning to build a NIDS for network intrusion detection	NSL-KDD
[103]	2020	Model voting ensemble, ensemble adversarial training, and query detection using MLP, CNN, and Conv-LSTM	A general framework dubbed Tiki-Taka, for evaluating the robustness of deep learning-based NIDS against adversarial manipulations with the aim of increasing the NIDS' resistance to attacks while engaging such evasion techniques	CSE-CIC-IDS2018
[104]	2021	CNN and LSTM	Model based on the association and combination of individual deep learning models to achieve better results in network intrusion detection	UNS-NB15

[105]	2021	CNN, LSTM and SVM	Hybrid semantic deep learning (HSDL) architecture that classifies the intrusion present in the text alongside its corresponding attack class	NSL-KDD and UNSW-NB15.
[106]	2019	CNN, LSTM, and Stacked Auto-Encoder (SAE)	Deep-Full-Range (DFR) model that learns spatial, temporal and extracting features from coding characteristics capable of classifying encrypted traffic and malware traffic	ISCX VPN-nonVPN traffic dataset and ISCX 2012
[46]	2017	CNN and LSTM	A hierarchical spatial-temporal features-based intrusion detection system (HAST-IDS) architecture for designing NIDSs	DARPA1998 and ISCX2012
[107]	2020	CNN and quasi-recurrent neural network (QRNN).	Improve threats classification performance in Cyber Threat Intelligence for Secure Smart City that comprise a large number of sensors that constantly generate a significant amounts of sensitive data such as credit card numbers, location coordinates, and medical records.	ToN_IoT
[108]	2020	DBN and GRU	Combines both dynamic analysis and static analysis technology to deal with spread use of obfuscation in android malware detection	Android PRAGuard
[109]	2021	CNN and WDLSTM	Improve speed and extract meaningful features from network traffic in big data environments for effective intrusion detection	UNSW-NB15
[110]	2019	CNN and LSTM	Intrusions detection on a large-scale network by using CNN and LSTM algorithms for spatial and temporal feature learning in the traffic data	NSL-KDD and UNSW-NB15
[111]	2018	CNN and LSTM	Significantly increase accuracy of intrusions detection in networks where CNN is used for feature extraction before LSTM layer and classify the packet traffic.	NSL-KDD
[112]	2021	Feature fusion of CNN and LSTM	Extraction of spatial-temporal information from network traffic for network intrusion discovery scenarios of video surveillance system (VSS).	KDD Cup 99

B. Characteristics of Hybrid Deep Learning-Based Models

The literature review sought to establish the salient characteristics of hybrid deep learning-based models. From the reviewed papers, the following are the characteristics of the hybrid deep learning-based models in the design of network intrusion detection.

i.) Approximation

Majority of the hybrid deep learning-based models exploit neural networks in the design of NIDS. Neural networks are considered to be universal function approximators. They have the capacity to approximate any function arbitrarily well, when presented with enough neurons. The success of deep learning techniques rests in their capacity to estimate complex unknown functional forms for the relationship between the predictors and the outcome variable [113]. The authors further posits that the accuracy of the approximation function is dependent on the neural network structure, characterized by the number, dependence and hierarchy between the nodes within and across layers. Essentially, a neural network successfully implements a mapping which approximates a function that is learned based on a given set of input-output value pairs. Substantial developments in understanding the approximation capabilities of neural networks has been made in the works of [113], [114], [115], [116]. According to these authors, deep neural networks perform better approximation compared to shallow networks due to their ability to mimic any compositional structure inherent in the target function; an ability that shallow networks cannot have. The universal approximation theorem advanced by [117] postulates that considering only continuous activation functions σ, at that point, a standard feed forward neural network consisting of one hidden layer as shown in figure 3 is capable of approximating any continuous multivariate function f, to any given approximation threshold, $\mathcal{E}$ on condition that σ is non-polynomial. The objective of a feed-forward network is to approximate a given function f. Considering a classifier, $y = f(x)$ maps an input x to a category y.

Such a feed-forward neural network determines a mapping $y = f(x; \boldsymbol{\theta})$ that learns the value of the parameters $\boldsymbol{\theta}$ that yields the most outstanding function approximation from a sample dataset$(x_i, y_i)_{i=1}^{n}$.

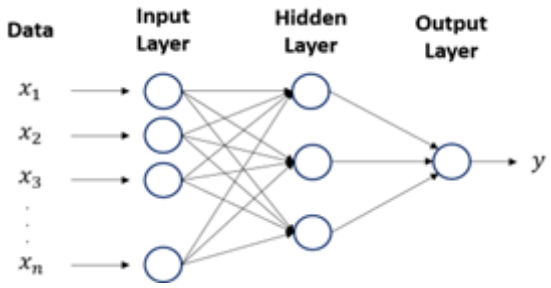

Figure 7: Feed forward neural network

In his study [115], explicitly quantified the number of hidden units essential for neural network approximation. A standard neural network is determined by its architecture and weights. Considering a feedforward neural network with n input units, m output units, and with one or more hidden layers, denotes a computational model $\mathcal{M}$ which can calculate a given class of functions $\rho: \mathbb{R}^n \rightarrow \mathbb{R}^t$, where $\rho = \rho w$ is parametrized by W (referred as weights of $\mathcal{M}$). Yarotsky [114] demonstrated that in the optimal approximations by neural networks the weights generally discontinuously depend on the approximated function, and that many nonlinear approximation schemes involve some form of discontinuity, often explicitly.

The author proved that using very deep fully connected networks of depth $L \sim W$ one can approximate function f with good propagation of errors $O\left(\omega_f\left(O(W^{-\frac{2}{v}})\right)\right)$ that are not attainable with less deep networks. In the literature, it is widely recognized that deeper models can reduce the number of computational units (hidden units) needed to approximate the target function by a factor that is exponential in the depth of the feedforward neural network [118]. In the wok of [119], it was demonstrated that deep networks are better when the complexity is computed in terms of the rank of certain tensors. Mhaskar and Poggio [116] also proved that good propagation of errors enables researchers to lift theorems on approximation power of shallow networks to those of deep networks if all the constituent functions are Lipschitz continuous [120]. Table 3 presents the results in approximation-theoretic networks that have multiple hidden layers.

Table 3: Approximation-theoretic results for networks with multiple hidden layers

S/No.	Authors			
		General Functions	Continuous Functions	Functions With Their Derivatives
1	[121]	√		
2	[122]	√		
3	[123]		√	
4	[124]			√
5	[125]		√	

There are a plethora of neural network architectures and activation functions. For instance, in their work [125] acknowledged existence of a function which, though expressible through a small three-layer network, can be represented using a very large two-layer network; where the total number of neurons in the network determines its size. Mhaskar and Poggio [116] also proved that the results for continuous activation functions in deep convolutional neural networks are similar to the results in [125].

In the surveyed papers, the Rectified Linear Units (ReLU), a linear piecewise function which directly gives an output if the input is positive and outputs zero if the input is negative, is normally adopted as a default activation function, helping the neural network better perform and train mainly because the neural networks are multilayered [118], [126], [127]. The multilayer networks cannot use hyperbolic tangent and sigmoid activation functions due to the vanishing gradient problem [128]. The vanishing gradient problem is a consequence of the depth of the network. In backpropagation algorithms, the model weights are adjusted in proportion to the gradient error where the error vector may shrink exponentially, causing the gradients in model to vanish as it approaches the early layers of the network.

ii.) Empirical Risk Minimization

Empirical risk minimization (ERM) is a principle that most neural network optimizations presently follow, that is, the error or risk of a learner trained using known empirical data (training samples), also known as "empirical error" or "training error" [129]. As a theory in statistical learning, ERM outlines a collection of learning algorithms that gives theoretical bounds on the performance of such algorithms. Consequently, ERM helps delineate a good classification and regression learning function from a bad one [130] and are based on replacing (or approximating) the average prediction error with the empirical risk incurred by a predictor when applied to a finite set of labelled data points (the training set).

If we assume that we have chosen a particular hypothesis space $\mathcal{H}$, that comprises of all computationally feasible predictor maps h, in machine learning tasks such as classification or regression, it is important to determine which predictor map h out of all the maps in the hypothesis space $\mathcal{H}$. Machine learning methods aim at finding the predictor $h \in \mathcal{H}$ with minimal average prediction error. There is need to describe a measure of the loss (or error) incurred when the predictor $h(x)$ is used whereas the true label is y. The loss function can be formally defined as:

$$\mathcal{L}: \boldsymbol{x} \times \mathcal{y} \times \mathcal{H} \mapsto \mathbb{R} \quad (3)$$

Where the loss $\mathcal{L}((\boldsymbol{x}, y)h)$ incurred is measured by predicting the label y of a data point by means of the prediction $h(\boldsymbol{x})(=: \hat{y})$. The calculation of the empirical error is greatly expensive even for a moderate sample size n, since it necessitates averaging $O(n^d)$ terms [131]. The ERM is a valuable technique whereby a good approximation of globally optimal classifier maybe attained to provide a good statistical classification result. As such, ERM is employed to define the risk (loss) function especially in supervised learning tasks where the idea is to learn a predictor map having a small training error.

$$\varepsilon := (1/\ |\mathcal{T}\ |)\ \sum_{(x,y)\ \in \mathcal{T}}\ \mathcal{L}(g, (\boldsymbol{x}, y)) \quad (4)$$

The training error is calculated on a given set of data points that are labelled;

$$\mathcal{T} \subseteq \mathcal{D} = \left\{\left(\boldsymbol{x}^{(1)}, y^{(1)}\right), \left(\boldsymbol{x}^{(2)}, y^{(2)}\right), \ldots, \left(\boldsymbol{x}^{(m)}, y^{(m)}\right)\right\} \quad (5)$$

Where the true label values $y^{(i)}$ are known and the set $\mathcal{D}$ stand for all available labelled data points. Here, the training error $\varepsilon\ (g)$ is given by the average loss incurred on some labelled (training) data $\mathcal{T} \subseteq \mathcal{D}$. The utmost application of the ERM principle is the learning of a predictor map from a model class that best fits the sample through solving:

$$\hat{g}\ \in arg \min_{g \in \mathcal{H}}\ \varepsilon(g) \quad (6)$$

Where the strict subset $\mathcal{T} \subset \mathcal{D}$ of all data points that are labelled is employed to compute the training error and the remainder of the data points $\mathcal{D} \backslash \mathcal{T}$ not used during the training, validate the predictor learnt. In the surveyed literature, ERM covers many of the popular learning methods and is extensively used in practice based on the papers reviewed.

iii.) Optimization and Model Complexity Control

Zhou [132] opined that popular of deep learning applications are logically formulated as non-convex optimization owing to the complex mechanism of the underlying model.

Retrieval Number: 100.1/ijese.F25300510622
DOI: 10.35940/ijese.F2530.0610722
Journal Website: www.ijese.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

Novak, et al. [133] also observed that neural networks are prominently non-convex models with extreme capacity that train fast and generalize well. Moreover, neural networks consists of several symmetric configurations for instance, exchanging intermediate neurons, hence considered as non-convex. Through the advent of deep learning, researchers needed to progressively manage non-convex optimization, more predominantly on account of the benefits hidden behind its complexity. A non-convex optimization is any problem where the objective or any of the constraints are non-convex [134] particularly because such algorithms operate in high-dimensional spaces. In the reviewed papers, all the NIDS models were tested on datasets with high dimensional data. The liberty to express the learning problem as a non-convex optimization problem grants immense modelling power to the designer of an algorithm [135]. In the surveyed literature, local optimization methods are widely used in applications where there is value in finding a good point, if not the very best. In their work, [136] demonstrated that deep neural networks are acquiescent to optimization by gradient descent despite being non-convex. Model complexity arises in the context of predictive learning and adaptive estimation of dependencies from finite data [137]. According to [138] the complexity of the network is contingent on suitable measures of complexity of the space of functions realized by the network such as VC-dimension, covering numbers and Rademacher numbers. Over time, researchers have acknowledged that the key to suitable predictive mode performance is in the control of the complexity of the neural network. The authors [138] further posit that model complexity may possibly be controlled during optimization by introducing a constraint, usually in the form of a regularization penalty, on the mean of the weights. Assuming that we have chosen a particular hypothesis space $\mathcal{H}$ that entails all computationally feasible predictor maps h. Generally, model overfitting occur mainly because the hypothesis space $\mathcal{H}$ is extremely huge for the sample size n. Obviously, the complexity of the hypothesis space (that is, the size of $\mathcal{H}$) attainable relies on the quantity of training data available. Figure 4 illustrates the relationship between model complexity, true risk, and the empirical risk for a particular training dataset.

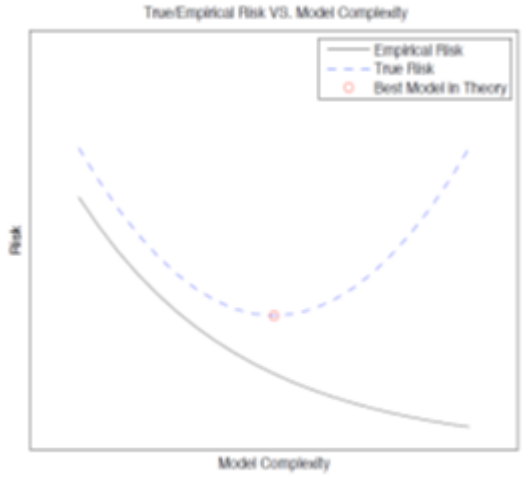

Figure 8: Relationship between Model Complexity, Empirical Risk and True Risk

One method in avoiding model overfitting is to choose $\mathcal{H}$ to guarantee suitability for the sample size. Several mechanisms exists to control the model complexity, which are in reality quite similar in operation. In the literature, two frequently used approaches are:

a) Structural risk minimization

Consider $\mathcal{H}_1, \mathcal{H}_2, \ldots, \mathcal{H}_n$ as a sequence of growing sized spaces. For instance, one normally has $\mathcal{H}_k \subset \mathcal{H}_{k+1}$ and $\cup \mathcal{H}_k = \mathcal{H}$. Assuming the training data $\mathcal{D}_n$, one can find $\hat{h}_n$ by reducing

$$\hat{h}_n = arg \min_{h \in \mathcal{H}_n} \hat{R}_n(h) \quad (7)$$

b) Penalized empirical risk minimization.

This entails defining penalty function $\Omega: \mathcal{H} \mapsto \mathbb{R}^+$ and determine $\hat{h}_n$ as shown in the ensuing optimization procedure:

$$\hat{h}_n = arg \min_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, h(x_i)) + \lambda_n \Omega(h) \quad (8)$$

Where $\lambda_n > 0$ provides a balanced trade-off between model complexity and goodness-of-fit. Actually, there is a regular need to select $\mathcal{H}_n$ or λ_n based on the training data in order to attain a good balance between model complexity and goodness-of-fit. In the literature, gradient descent algorithms are the most popular techniques for optimizing deep learning related models. These techniques utilize similar learning rate to adjust all parameters of the model. Such techniques include Stochastic Gradient Descent, Mini batching, Nesterov Accelerated Gradient Descent, Momentum and Adam which are classified as classical iterative optimization algorithms. More recently, new variants of adaptive methods have similarly been proposed. According to [139], new optimization techniques have been developed that adaptively modify the learning rate per parameter throughout the training process. These techniques are used to change the attributes of weights and learning rate in order to reduce the model losses and are considered adaptive gradient methods mainly employed in both supervised and unsupervised tasks [140]. Based on the literature review, it is observed that the Adaptive Momentum (Adam) optimization technique [141] is the prevalent technique employed for optimizing many of the deep learning-based models. Adam combines the RMSprop and momentum techniques, that are known to use the squared gradient for scaling the learning rate parameters similar to RMSprop and works the same way as the momentum by adding averages of moving gradients. Together with its variant, Adam computes adaptive learning rates for every parameter by coalescing the concepts of momentum and adaptive gradient retaining an exponentially decaying average of past gradients. Besides, these algorithms are known as first-order adaptive optimization algorithms owing to their super-fast convergence speed while solving large scale optimization tasks [142]. The adaptive gradient methods iteratively update parameters by moving them to the direction of the negative gradient of the cost function with non-fixed learning rate. Further, the problem of rapid decay of learning rate is addressed by scaling down the gradient by the square roots of exponential moving average of past squared gradients [143].

Retrieval Number: 100.1/ijese.F25300510622
DOI: 10.35940/ijese.F2530.0610722
Journal Website: www.ijese.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

The analysis results demonstrates that the effective learning rate, in theory, grows over time in a fairly quick manner during model training and evaluation.

C. Open Challenges and Future Research Issues

Finally, the literature review sought to evaluate the future scope of research in hybrid deep learning-based NIDSs. In this section, we outline open challenges and highlight possible future research directions. Hybrid deep learning-based modeling revel in the benefits of both generative and discriminative learning. In effect, hybridization may outdo the others in terms of demonstrated performance. From the review, hybridization or ensembles of these techniques have proved to yield better performance in regard to accuracy, detection rate and reduction of false alarm rates. Regardless of the many years of development and enormous efforts by the researchers, current hybrid deep learning-based NIDSs still encounters challenges while improving detection accuracy as well as minimizing false alarm rates and in detecting new intrusions [6]. To address these gaps, scholars have focused on building NIDSs that capitalize on deep learning methods that have the capacity to improve the detection accuracy of known and unknown attacks and have strong generalizability. The effect of the overfitting problem during the execution of such deep learning algorithms [8] is still an area that requires further research particularly in the design of NIDSs. Neural networks suffer from generalization errors due to their special structure or the regularization techniques used during training. As such, suitable regularization techniques are required to control the generalization error particularly in deep neural network models. Regularization prevent overfitting of models to the training data, where modification are made to the learning algorithm with the intention to minimise its generalization error rather than its training error [144]. A main constituent in creation of an efficient NIDS is the preprocessing of network traffic and identification of important features that is indispensable for building robust classifier [145]. Developing NIDS in a dynamically changing computing environment require fast and suitable feature selection [146] and dimensionality reduction methods remains a challenging matter given that most of the network IDS are dependent on the deployed environment [20]. Further, the high latitude and non-linear characteristics of computer network data make the network intrusion detection work difficult to break through [147]. Therefore, future research should focus on approaches that extracts significant and relevant features from voluminous amount of noisy, high-dimensional, and unlabelled network traffic data. Again, the choice and use of an appropriate dataset is a major issue in the design of hybrid deep learning-based NIDSs. Hybrid deep learning-based models comes at the expense of being more complex, thus harder to maintain and explain. Further, they similarly require more resources and time to analyze the network activities [148]. In their work [149] concluded that low-complexity models have a small norm of Hessian matrix with respect to model parameters. Thus, the property of the Hessian denotes that the volume of good minima dominates over that of poor ones, which ultimately yields an almost sure convergence to good solutions, as demonstrated by various empirical results obtained from the reviewed papers. Future research in this area need to consider mechanisms for reducing the computational complexity in their designs. Most of the hybrid deep learning-based NIDS has focused on discriminative/supervised learning [1]. Discriminative architectures that are applied in the design of NIDS largely comprise of DNN, CNN, and RNN, together with their variants. Future research should take into account optimization, model complexity control, and applicability, consistent with the nature of the data. This would be a novel contribution in the domain, which arguably can be a key future aspect in discriminative learning.

V. CONCLUSION

Network security issues are becoming increasingly prominent with deep learning attracting the attention of scholars in network security domain. Hybrid deep learning approaches have been recommended and distinguished to be ideal in identifying network attacks more accurately. In this study we sought to synthesize available research on hybrid deep learning-based approaches for NIDSs design. Through the systematic literature review, the study endeavoured to answer the following three research questions: what are the recent hybrid deep learning approaches adopted for the design NIDS? What are the salient characteristics of hybrid deep learning-based NIDS? and What is the future scope of research in hybrid deep learning-based NIDS? Through the detailed review, we selected 67 papers published in peer reviewed journals and conference papers between 2027 and 2021. The study presented a structured and comprehensive systematic literature review of hybrid deep learning-based NIDSs. A taxonomy of deep learning approaches was presented taking into account the deep networks for discriminative or supervised learning, generative or unsupervised learning, and finally hybrid learning that can be used to design a variety of hybrid systems. Our focus was on hybrid based systems where the study described their categorization into either algorithmic, cooperative or hierarchical architectures. We also detailed the salient characteristics exhibited by these architectures in the design of such hybrid systems. The study finally presented a summary and discussion on the open challenges still facing hybrid deep learning-based NIDSs and the potential future research directions in the area. The authors opine that our study points in a promising path to the design of NIDSs and can be utilized for future research and implementations targeting other relevant application fields by academicians and industry professionals.

REFERENCES

1. Z. Wang, Z. Li, J. Wang and D. Li, “Network Intrusion Detection Model Based on Improved BYOL Self-Supervised Learning,” Security and Communication Networks, vol. 2021, pp. 1-23, 2021. [CrossRef]
2. M. Ahmed, A. N. Mahmood and J. Hu, “A survey of network anomaly detection techniques,” Journal of Network and Computer Applications, vol. 60, pp. 19-31, 2016. [CrossRef]
3. W. Liang, S. Xie, D. Zhang, X. Li and K. Li, “A mutual security authentication method for RFID-PUF circuit based on deep learning,” ACM Transactions on Internet Technology, pp. 1-20, 2020. [CrossRef]
4. A. Khraisat, I. Gondal, P. Vamplew and J. Kamruzzaman, “Survey of intrusion detection systems: techniques, datasets and challenges,” Cybersecurity, vol. 2, no. 20, pp. 1-22, 2019. [CrossRef]

Retrieval Number: 100.1/ijese.F25300510622
DOI: 10.35940/ijese.F2530.0610722
Journal Website: www.ijese.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

5. Q. Tian, D. Han, K.-C. Li, X. Liu, L. Duan and A. Castiglione, "An intrusion detection approach based on improved deep belief network," Applied Intelligence, vol. 50, no. 10, pp. 3162-3178, 2020. [CrossRef]
6. Z. Ahmad, A. Khan, C. Shiang, J. Abdullah and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," Transactions on Emerging Telecommunications Technologies, vol. 32, pp. 1-23, 2021. [CrossRef]
7. B. Dong and X. Wang, "Comparison deep learning method to traditional methods using for network intrusion detection," in Proceedings of the 8th IEEE International Conference on Communication Software and Networks (ICCSN), Beijing, China, 2016. [CrossRef]
8. M. ElSayed, N.-A. Le-Khac, M. Albahar and A. Jurcut, "A novel hybrid model for intrusion detection systems in SDNs based on CNN and a new regularization technique," Journal of Network and Computer Applications, vol. 191, pp. 1-18, 2021. [CrossRef]
9. B. Lee, S. Amaresh, C. Green and D. Engels, "Comparative Study of Deep Learning Models for Network Intrusion Detection," SMU Data Science Review, vol. 1, no. 1, pp. 1-14, 2018.
10. M. Al Imran and S. Ripon, "Network Intrusion Detection: An Analytical Assessment Using Deep Learning and State of the Art Machine Learning Models," International Journal of Computational Intelligence Systems, vol. 14, no. 200, pp. 1-20, 2021. [CrossRef]
11. J. Lew, D. Shah, S. Pati, S. Cattell, M. Zhang, A. Sandhupatla, C. Ng, N. Goli, M. Sinclair, T. Rogers and T. Aamodt, "Analyzing machine learning workloads using a detailed GPU simulator," in Proceedings of the IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), Madison, WI, USA, 2019. [CrossRef]
12. Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou and C. Wang, "Machine learning and deep learning methods for cybersecurity," IEEE access, vol. 6, pp. 35365-35381, 2018. [CrossRef]
13. A. Megantara and T. Ahmad, "A hybrid machine learning method for increasing the performance of network intrusion detection systems," Journal of Big Data, vol. 8, no. 142, pp. 1-19, 2021. [CrossRef]
14. K. Scarfone and P. Mell, "SP 800-94. Guide to Intrusion Detection and Prevention Systems (IDPS)," National Institute of Standards & Technology, Gaithersburg, MD, USA, 2007. [CrossRef]
15. H. Liao, C. Lin, Y. Lin and K. Tung, "Intrusion detection system: a comprehensive review," Journal of Network Computing Applications, vol. 36, no. 1, pp. 16-24, 2013. [CrossRef]
16. D. E. Denning, "An intrusion-detection model," IEEE Transactions in Software Engineering, SE-13, vol. 2, pp. 222-232, 1987. [CrossRef]
17. K. Kim, "Intrusion Detection System Using Deep Learning and Its Application to Wi-Fi Network," IEICE Transactions on Information and Systems, vol. E103–D, no. 7, pp. 1433-1447, 2020. [CrossRef]
18. T. Riera, J.-R. Higuera, J. Higuera, J.-J. Herraiz and J.-A. Montalvo, "Prevention and Fighting against Web Attacks through Anomaly Detection Technology. A Systematic Review," Sustainability, vol. 12, pp. 1-45, June 2020. [CrossRef]
19. A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman and A. Alazab, "Hybrid Intrusion Detection System Based on the Stacking Ensemble of C5 Decision Tree Classifier and One Class Support Vector Machine," Electronics, vol. 9, no. 173, pp. 1-18, 2020. [CrossRef]
20. V. Jyothsna and K. Prasad, "Anomaly-Based Intrusion Detection System," in Computer and Network Security, IntechOpen, 2019, pp. 1-15. [CrossRef]
21. Symantec, "Internet security threat report 2017," 17 April 2017. [Online]. Available: https://www.symantec.com/content/dam/symantec/docs/reports/. [Accessed 23 May 2020].
22. N. Ampah, C. M. Akujuobi, M. N. O. Sadiku and S. Alam, "An intrusion detection technique based on continuous binary communication channels," International Journal of Security and Networks, vol. 6, no. 2-3, pp. 174-180, 2011. [CrossRef]
23. R. Vinayakumar, M. Alazab, K. Soman, P. Poornachandran, A. Al-Nemrat and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," IEEE Access, vol. 7, pp. 4 1525-4 1550, 2019. [CrossRef]
24. R. Vinayakumar, M. Alazab, K. Soman, P. Poornachandran, A. Al-Nemrat and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," IEEE Access, pp. 41525-41550, 2019. [CrossRef]
25. N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," IEEE Access, vol. 6, p. 14410–14430, 2018. [CrossRef]
26. M. Ganaiea, M. Hu, A. Malika, M. Tanveera and P. Suganthan, "Ensemble deep learning: A review," Arxiv pre prints, p. arXiv:2104.02395v2, 2022.
27. D.-A. Clevert, T. Unterthiner and S. Hochreiter, "Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)," arXiv preprint , p. arXiv:1511.07289, 2016.
28. K. Janocha and W. Czarnecki, "On Loss Functions for Deep Neural Networks in Classification," arXiv Preprints, p. arXiv:1702.05659v1, 18 Feb 2017. [CrossRef]
29. L. Deng and D. Yu, "Deep Learning: Methods and Applications," Foundations and Trends in Signal Processing, vol. 7, no. 3-4, pp. 197-387, 2014. [CrossRef]
30. I. Goodfellow, Y. Bengio and A. Courville, Deep Learning, Cambridge, Massachusetts: MIT Press, 2016.
31. T. Che, X. Liu, S. Li, Y. Ge, R. Zhang, C. Xiong and Y. Bengio, "Deep Verifier Networks: Verification of Deep Discriminative Models with Deep Generative Models," arXiv , p. arXiv:1911.07421v2, 8 Feb 2020.
32. J. Gordon and J. Hernández-Lobato, "Combining deep generative and discriminative models for Bayesian semi-supervised learning," Pattern Recognition, vol. 100, pp. 1-10, 2020. [CrossRef]
33. C. Goyal, "Deep Understanding of Discriminative and Generative Models in Machine Learning," analyticsvidhya, 19 July 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/07/deep-understanding-of-discriminative-and-generative-models-in-machine-learning/. [Accessed 22 March 2022].
34. W. Sun, H. Liu, R. Tang, Y. Lang, J. He and Q. Huang, "sEMG-Based Hand-Gesture Classification Using a Generative Flow Model," Sensors, vol. 19, no. 1952, pp. 1-16, 2019. [CrossRef]
35. S. Al and M. Dener, "STL-HDL: A new hybrid network intrusion detection system for imbalanced dataset on big data environment," Computers and Security, vol. 110, no. C, 2021. [CrossRef]
36. A. Tasdelen and B. Sen, "A hybrid CNN-LSTM model for pre-miRNA classification," Scientific Reports, vol. 11, no. 14125, pp. 1-9, 2021. [CrossRef]
37. S. Keele, "Guidelines for Performing Systematic Literature Reviews in Software Engineering," EBSE, 2007.
38. D. Moher, L. Shamseer, M. Clarke, D. Ghersi, A. Liberati, M. Petticrew, P. Shekelle and L. Stewart, "Preferred reporting items for systematic reviewand meta-analysis protocols (PRISMA-P) 2015 statement," Systematic Reviews, vol. 4, no. 1, pp. 1-9, 2015. [CrossRef]
39. A. Alqahtani, "FSO-LSTM IDS: hybrid optimized and ensembled deep-learning network-based intrusion detection system for smart networks.," The Journal of Supercomputing, vol. 21, 2022. [CrossRef]
40. Y. Zeng, H. Gu, W. Wei and Y. Guo, "Deep-Full-Range: A Deep Learning Based Network Encrypted Traffic Classification andIntrusion Detection Framework," IEEE Access, vol. 7, pp. 45182-45190, 2019. [CrossRef]
41. C. Ma, X. Du and L. Cao, "Analysis of Multi-Types of Flow Features Based on Hybrid Neural Network for Improving Network Anomaly Detection," IEEE Access, vol. 7, pp. 148363-148380, 2019. [CrossRef]
42. M. ElSayed, N.-A. Le-Khac, M. Albahar and A. Jurcut, "A novel hybrid model for intrusion detection systems in SDNs based on CNN and a new regularization technique," Journal of Network and Computer Applications, vol. 191, pp. 1-18, 2021. [CrossRef]
43. A. Megantara and T. Ahmad, "A hybrid machine learning method for increasing the performance of network intrusion detection systems," Journal of Big Data, vol. 8, no. 142, pp. 1-19, 2021. [CrossRef]
44. R. Yao, N. Wang, Z. Liu, P. Chen and X. Sheng, "Intrusion Detection System in the Advanced Metering Infrastructure: A Cross-Layer Feature-Fusion CNN-LSTM-Based Approach," Sensors, vol. 21, no. 626, pp. 1-17, 2021. [CrossRef]
45. M. Khan, "HCRNNIDS: Hybrid Convolutional Recurrent Neural Network-Based Network Intrusion Detection System," Processes, vol. 9, no. 834, pp. 1-14, 2021. [CrossRef]

46. W. Wang, Y. Sheng, J. Wang, X. Zeng, X. Ye, Y. Huang and M. Zhu, "HAST-IDS: Learning Hierarchical Spatial-Temporal Features Using Deep Neural Networks to Improve Intrusion Detection," IEEE Access, vol. 6, pp. 1792-1806, 2017. [CrossRef]
47. W. Wang, Y. Sheng, J. Wang, X. Zeng, X. Ye, Y. Huang and M. Zhu, "HAST-IDS: Learning Hierarchical Spatial-Temporal Features Using Deep Neural Networks to Improve Intrusion Detection," IEEE Access, vol. 6, pp. 1792-1806, 11 December 2017. [CrossRef]
48. Y. Yao, L. Su, Z. Lu and B. Liu, "STDeepGraph: Spatial-Temporal Deep Learning on Communication Graphs for Long-Term Network Attack Detection," in 2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), Rotorua, New Zealand, 2019.
49. M. Hassan, A. Gumaei, A. Alsanad, M. Alrubaian and G. Fortino, "A hybrid deep learning model for efficient intrusion detection in big data environment," Information Sciences, vol. 513, pp. 386-396, 2020. [CrossRef]
50. P. Sun, P. Liu, Q. Li, C. Liu, X. Lu, R. Hao and J. Chen, "DL-IDS: Extracting Features Using CNN-LSTM Hybrid Network for Intrusion Detection System," Security and Communication Networks, vol. 2020, pp. 1-11, 2020. [CrossRef]
51. M. Al-Imran and S. Ripon, "Network Intrusion Detection: An Analytical Assessment Using Deep Learning and State-of-the-Art Machine Learning Models," International Journal of Computational Intelligence Systems, vol. 14, no. 200, pp. 1-20, 2021. [CrossRef]
52. A. M. and N. K., "Convolutional neural networks with LSTM for intrusion detection," in Proceedings of the 35th International Conference, Seville, Spain, 2020.
53. J. Malik, A. Akhunzada, I. Bibi, M. Imran, A. Musaddiq and S. Kim, "Hybrid Deep Learning: An Efficient Reconnaissance and Surveillance Detection Mechanism in SDN," IEEE Access, vol. 8, pp. 134695-134706, 16 July 2020. [CrossRef]
54. Z. Maseer, R. Yusof, S. Mostafa, N. Bahaman, O. Musa and B. A.-S. Al-rimy, "DeepIoT. IDS: hybrid deep learning for enhancing IoT network intrusion detection," Computers, Materials & Continua, vol. 69, no. 3, pp. 3945-3966, 2021. [CrossRef]
55. A. Psathas, L. Iliadis, A. Papaleonidas and D. Bountas, "A Hybrid Deep Learning Ensemble for Cyber Intrusion Detection," in Proceedings of the 22nd Engineering Applications of Neural Networks Conference, 2021. [CrossRef]
56. I. Ullah, A. Ullah and M. Sajjad, "Towards a Hybrid Deep Learning Model for Anomalous Activities Detection in Internet of Things Networks," IoT, vol. 2, no. 3, pp. 428-448, 2021. [CrossRef]
57. X. Li and S. Zhang, "Network Intrusion Detection Methods Based on Deep Learning," Recent Patents on Engineering, vol. 15, no. 4, pp. 80-88(9), 2021.
58. A. A. Süzen, "Developing a multi-level intrusion detection system using hybrid-DBN," Journal of Ambient Intelligence and Humanized Computing, vol. 12, pp. 1913-1923, 2021. [CrossRef]
59. A. Kim, M. Park and D. H. Lee, "AI-IDS: Application of Deep Learning to Real-Time Web Intrusion Detection," IEEE Access, vol. 8, pp. 70245-70261, 2020. [CrossRef]
60. P. Wichmann, M. Marx, H. Federrath and M. Fischer, "Detection of Brute-Force Attacks in End-to-End Encrypted Network Traffic," in ARES 2021: The 16th International Conference on Availability, Reliability and Security, Vienna, Austria, 2021. [CrossRef]
61. S. Popoola, B. Adebisi, M. Hammoudeh, G. Gui and H. Gacanin, "Hybrid Deep Learning for Botnet Attack Detection in the Internet-of-Things Networks," IEEE Internet of Things Journal, vol. 8, no. 6, pp. 4944-4956, 15 March 2021. [CrossRef]
62. M. A. Khan, M. R. Karim and Y. Kim, "A Scalable and Hybrid Intrusion Detection System Based on the Convolutional-LSTM Network," Symmetry, vol. 11, no. 4, p. 14, 2019. [CrossRef]
63. M. M. Moussa and L. Alazzawi, "A Hybrid Deep Learning Cyber-Attacks Intrusion Detection System for CAV Path Planning," in 2021 IEEE International Midwest Symposium on Circuits and Systems (MWSCAS), Lansing, MI, USA, 2021. [CrossRef]
64. Y. Zhang, X. Chen, L. Jin, X. Wang and D. Guo, "Network Intrusion Detection: Based on Deep Hierarchical Network and Original Flow Data," IEEE Access, vol. 7, pp. 37004-37016, 2019. [CrossRef]
65. H. Mennour and S. Mostefai, "A hybrid Deep Learning Strategy for an Anomaly Based N-IDS," in 2020 International Conference on Intelligent Systems and Computer Vision (ISCV), Fez, Morocco, 2020. [CrossRef]
66. Y. Zhang, H. Zhang, X. Zhang and D. Qi, "Deep Learning Intrusion Detection Model Based on Optimized Imbalanced Network Data," in 2018 IEEE 18th International Conference on Communication Technology (ICCT), Chongqing, China, 2018. [CrossRef]
67. K. Rao, K. Rao and P. Reddy, "A hybrid Intrusion Detection System based on Sparse autoencoder and Deep Neural Network," Computer Communications, vol. 180, no. C, pp. 77-88, 2021. [CrossRef]
68. N. Khare, P. Devan, C. Chowdhary, S. Bhattacharya, G. Singh, S. Singh and B. Yoon, "SMO-DNN: Spider Monkey Optimization and Deep Neural Network Hybrid Classifier Model for Intrusion Detection," Electronics, vol. 9, no. 4, p. 18, 2020. [CrossRef]
69. M. Anwer, G. Ahmed and A. S. S. Akhunzada, "Intrusion Detection Using Deep Learning," in 2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), Mauritius, Mauritius, 2021. [CrossRef]
70. L. Matsa, P. G.-A. Zodi-Lusilao and P. F. Bhunu-Shava, "Forward Feature Selection for DDoS Detection on Cross-Plane of Software Defined Network Using Hybrid Deep Learning," in 3rd International Multidisciplinary Information Technology and Engineering Conference (IMITEC), Windhoek, Namibia, 2021. [CrossRef]
71. J. Awotunde, C. Chakraborty and A. Adeniyi, "Intrusion Detection in Industrial Internet of Things Network-Based on Deep Learning Model with Rule-Based Feature Selection," Wireless Communications and Mobile Computing, vol. 2021, p. 17, 2021. [CrossRef]
72. H. Deng and T. Yang, "Network Intrusion Detection Based on Sparse Autoencoder and IGA-BP Network," Wireless Communications and Mobile Computing, vol. 2021, p. 11, 2021. [CrossRef]
73. S. Garg, K. Kaur, N. Kumar and J. J. P. C. Rodrigues, "Hybrid Deep-Learning-Based Anomaly Detection Scheme for Suspicious Flow Detection in SDN: A Social Multimedia Perspective," IEEE Transactions on Multimedia, vol. 21, no. 3, pp. 566-578, 2019. [CrossRef]
74. D. Javeed, T. Gao, M. T. Khan and I. Ahmad, "A Hybrid Deep Learning-Driven SDN Enabled Mechanism for Secure Communication in Internet of Things (IoT)," Sensors, vol. 21, p. 18, 2021. [CrossRef]
75. M. Khan and J. Kim, "Toward Developing Efficient Conv-AE-Based Intrusion Detection System Using Heterogeneous Dataset," Electronics, vol. 9, p. 17, 2020. [CrossRef]
76. S. Garg, K. Kaur, N. Kumar, G. Kaddoum, A. Y. Zomaya and R. Ranjan, "A Hybrid Deep Learning-Based Model for Anomaly Detection in Cloud Datacenter Networks," IEEE Transactions on Network and Service Management, vol. 16, no. 3, pp. 924-935, 2019. [CrossRef]
77. R. Li, Z. Song, W. Xie, C. Zhang, G. Zhong and X. Pei, "HALNet: A Hybrid Deep Learning Model for Encrypted C&C Malware Traffic Detection," in Network and System Security. NSS 2021. Lecture Notes in Computer Science, vol. 13041, M. Yang, C. Chen and Y. Liu, Eds., Springer, Cham., 2021.
78. V. K. A. Prabhakaran, "Hybrid semantic deep learning architecture and optimal advanced encryption standard key management scheme for secure cloud storage and intrusion detection," Neural Computing and Applications, vol. 33, pp. 14459-14479, 2021. [CrossRef]
79. I. Ullah, B. Raza, S. Ali, I. Abbasi, S. Baseer and A. Irshad, "Software Defined Network Enabled Fog-to-Things Hybrid Deep Learning Driven Cyber Threat Detection System," Security and Communication Networks, vol. 2021, p. 15, 2021. [CrossRef]
80. T. Bakhshi and B. Ghita, "Anomaly Detection in Encrypted Internet Traffic Using Hybrid Deep Learning," Security and Communication Networks, vol. 2021, p. 16, 2021. [CrossRef]
81. S. Haider, A. Akhunzada and G. R. M. Ahmed, "Deep Learning based Ensemble Convolutional Neural Network Solution for Distributed Denial of Service Detection in SDNs," in 2019 UK/ China Emerging Technologies (UCET), Glasgow, UK, 2019. [CrossRef]
82. T. Thilagam and R. Aruna, "Intrusion detection for network based cloud computing by custom RC-NN and optimization," ICT Express, vol. 7, no. 4, pp. 512-520, 2021. [CrossRef]
83. M. Roopak, G. Y. Tian and J. Chambers, "An Intrusion Detection System Against DDoS Attacks in IoT Networks," in 2020 10th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 2020. [CrossRef]
84. V. Dutta, M. Choraś, M. Pawlicki and R. Kozik, "A Deep Learning Ensemble for Network Anomaly and Cyber-Attack Detection," Sensors, vol. 20, no. 16, p. 20, 2020. [CrossRef]

Retrieval Number: 100.1/ijese.F25300510622
DOI: 10.35940/ijese.F2530.0610722
Journal Website: www.ijese.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

85. F. Sarhangian, R. Kashef and M. Jaseemuddin, “Efficient Traffic Classification Using Hybrid Deep Learning,” in 2021 IEEE International Systems Conference (SysCon), Vancouver, BC, Canada, 2021. [CrossRef]
86. I. A. Khan, D. Pi, Z. U. Khan, Y. Hussain and A. Nawaz, “HML-IDS: A Hybrid-Multilevel Anomaly Prediction Approach for Intrusion Detection in SCADA Systems,” IEEE Access, vol. 7, pp. 89507-89521, 2019. [CrossRef]
87. N. Thapa, Z. Liu, A. Shaver, A. Esterline, B. Gokaraju and K. Roy, “Secure Cyber Defense: An Analysis of Network Intrusion-Based Dataset CCD-IDSv1 with Machine Learning and Deep Learning Models,” Electronics, vol. 10, no. 15, pp. 1-13, 2021. [CrossRef]
88. S. Aldhaheri, D. Alghazzawi, L. Cheng, B. Alzahrani and A. Al-Barakati, “DeepDCA: Novel Network-Based Detection of IoT Attacks Using Artificial Immune System,” Applied Sciences, vol. 10, no. 6, 2020. [CrossRef]
89. Z. Zhao, L. Ge and G. Zhang, “A novel DBN-LSSVM ensemble method for intrusion detection system,” in ICCBN 2021: 2021 9th International Conference on Communications and Broadband Networking, 2021. [CrossRef]
90. I. Pustokhina, D. Pustokhin, E. Lydia, P. Garg, A. Kadian and K. Shankar, “Hyperparameter search based convolution neural network with Bi-LSTM model for intrusion detection system in multimedia big data environment,” Multimedia Tools and Applications, vol. 126, no. 2, 2021. [CrossRef]
91. S. Khan and A. Akhunzada, “A hybrid DL-driven intelligent SDN-enabled malware detection framework for Internet of Medical Things (IoMT),” Computer Communications, vol. 170, pp. 209-216, 2021. [CrossRef]
92. A. Azeezat, O. Adebukola, A. Abayomi-Alli and O. Olushola, “A Conceptual Hybrid Model of DeepConvolutional Neural Network (DCNN)and Long Short-Term Memory (LSTM) for Masquerade Attack Detection,” in ICTA 2020: Information and Communication Technology and Applications Third International Conference, Minna, Nigeria, 2021. [CrossRef]
93. G. Altan, “SecureDeepNet-IoT: A deep learning application for invasion detection in industrial Internet of Things sensing systems,” Transactions on Emerging Telecommunications Technologies, vol. 32, no. 4, 2021. [CrossRef]
94. H. Alkahtani and T. Aldhyani, “Botnet Attack Detection by Using CNN-LSTM Model for Internet of Things Applications,” Security and Communication Networks , vol. 2021, p. 23, 2021 . [CrossRef]
95. T.-Y. Kim and S.-B. Cho, “Web traffic anomaly detection using C-LSTM neural networks,” Expert Systems with Applications, vol. 106, pp. 66-76, 2018. [CrossRef]
96. H. Mohammad, G. Abdu, A. Ahmed, A. Majed and F. Giancarlo, “A hybrid deep learning model for efficient intrusion detection in big data environment,” Information Sciences, vol. 513, pp. 386-396, 2020. [CrossRef]
97. S. J. Bu and S. B. Cho, “A convolutional neural-based learning classifier system for detecting database intrusion via insider attack,” Information sciences, vol. 512, pp. 123-136, 2020. [CrossRef]
98. N. Aldahoul, H. A. Karim and A. Wazir, “Model fusion of deep neural networks for anomaly detection,” Journal of Big Data, vol. 8, no. 106, p. 19, 2021. [CrossRef]
99. M. A. Khan, “HCRNNIDS: Hybrid Convolutional Recurrent Neural Network-Based Network Intrusion Detection System,” Processes, vol. 9, no. 5, 2021. [CrossRef]
100. H. Liu, B. Lang, M. Liu and H. Yan, “CNN and RNN based payload classification methods for attack detection,” Knowledge-Based Systems, vol. 163, pp. 332-341, 2019. [CrossRef]
101. Y. Yang, K. Zheng, C. Wu and Y. Yang, “Improving the classification effectiveness of intrusion detection by using improved conditional variational AutoEncoder and deep neural network,” Sensors, vol. 19, no. 11, p. 20, 2019. [CrossRef]
102. K. Hara and K. Shiomoto, “Intrusion Detection System using Semi-Supervised Learning with Adversarial Auto-encoder,” in NOMS 2020 - 2020 IEEE/IFIP Network Operations and Management Symposium, Budapest, Hungary, 2020. [CrossRef]
103. C. Zhang, X. Costa-Perez and P. Patras, “Tiki-Taka: Attacking and Defending Deep Learning-based Intrusion Detection Systems,” in CCSW'20: Proceedings of the 2020 ACM SIGSAC Conference on Cloud Computing Security Workshop, New York, NY, USA, 2020. [CrossRef]
104. L. Duong, “Optimization of Cyber-Attack Detection Using the Deep Learning Network,” International Journal of Computer Science and Network Security (IJCSNS), vol. 21, no. 7, pp. 159-168, 2021.
105. V. Prabhakaran and A. Kulandasamy, “Hybrid semantic deep learning architecture and optimal advanced encryption standard key management scheme for secure cloud storage and intrusion detection,” Neural Computing and Applications, vol. 33, no. 1, 2021. [CrossRef]
106. Y. Zeng and H. W. W. &. G. Y. Gu, “Deep-Full-Range: A Deep Learning Based Network Encrypted Traffic Classification and Intrusion Detection Framework,” IEEE Access, vol. 7, pp. 45182-45190, 2019. [CrossRef]
107. N. Al-Taleb, N. Saqib, Atta-ur-Rahman and S. Dash, “Cyber Threat Intelligence for Secure Smart City,” ArXiv Preprints, vol. abs/2007.13233, 2020.
108. T. Lu, Y. Du, L. Ouyang, Q. Chen and X. Wang, “Android Malware Detection Based on a Hybrid Deep Learning Model,” Security and Communication Networks, vol. 2020, p. 11, 2020. [CrossRef]
109. L. Faling and P. Juan, “Intrusion Detection Based on CNN and WDLSTM in Big Data Environment,” Journal of Southwest China Normal University, vol. 46, no. 9, pp. 103-108, 2021.
110. P. Wu and H. Guo, “LuNET: a deep neural network for network intrusion detection,” in Proceedings of the Symposium Series on Computational Intelligence (SSCI), Xiamen, China, 2019.
111. C. M. Hsu, H. Y. Hsieh, S. W. Prakosa, M. Z. Azhari and J. S. Leu, “Using long-short-term memory based convolutional neural networks for network intrusion detection,” in Proceedings of the International Wireless Internet Conference, Taipei, Taiwan, 2018.
112. Z. Fan and Z. Cao, “An Improved Method of Network Intrusion Discovery Based on Convolutional Long-short Term Memory Network,” IEEE Access, vol. 9, p. 10, 2021. [CrossRef]
113. H. F. Calvo-Pardo, T. Mancini and J. Olmo, “Optimal Deep Neural Networks by Maximization of the Approximation Power,” arXiv Preprints, p. arXiv:2010.04044v2, 2020. [CrossRef]
114. D. Yarotsky, “Optimal approximation of continuous functions by very deep ReLU networks,” in Proceedings of the 31st Conference On Learning Theory, PMLR, 2018.
115. K. F. E. Chong, “A closer look at the approximation capabilities of neural networks,” in 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, 2020.
116. H. Mhaskar and T. Poggio, “FUNCTION APPROXIMATION BY DEEP NETWORKS,” COMMUNICATIONS ON PURE AND APPLIED ANALYSIS, vol. 19, no. 8, pp. 4085-4095, August 2020. [CrossRef]
117. M. Leshno, W. Lin, A. Pinkus and S. Schocken, “Multilayer feedforward networks with a nonpolynomial activation function can approximate any function,” Neural Networks, vol. 6, no. 6, pp. 861-867, 1993. [CrossRef]
118. G. Montufar, R. Pascanu, K. Cho and Y. Bengio, “On the number of linear regions of deep neural networks,” Advances in neural information processing systems, pp. 2924-2932, 2014.
119. O. Sharir and A. Shashua, “On the expressive power of overlapping architectures of deep learning,” arXiv preprint, p. arXiv:1703.02065 , 2017.
120. M. Ribeiro, “Lipschitz Continuity, convexity, subgradients,” 2 April 2015. [Online]. Available: https://homes.cs.washington.edu/~marcotcr/blog/lipschitz/.
121. K. Hornik, M. Stinchcombe and H. White, “Multilayer feedforward networks are universal approximators,” Neural Networks, vol. 2, no. 5, pp. 359-366, 1989. [CrossRef]
122. H. N. Mhaskar, “Approximation properties of a multilayered feedforward artificial neural network,” Advances in Computational Mathematics, vol. 1, no. 1, pp. 61-80, Feb 1993. [CrossRef]
123. K.-I. Funahashi, “On the approximate realization of continuous mappings by neural networks,” Neural Networks, vol. 2, no. 3, pp. 183-192, 1989. [CrossRef]
124. T. Nguyen-Thien and T. Tran-Cong, “Approximation of functions and their derivatives: A neural network implementation with applications,” Applied Mathematical Modelling, vol. 23, no. 9, pp. 687-704, 1999. [CrossRef]
125. R. Eldan and O. Shamir, “The power of depth for feed forward neural networks,” in Proceedings of the 29th Conference on Learning Theory (COLT 2016), New York, USA, 2016.
126. M. Raghu, B. Poole, J. Kleinberg, S. Ganguli and J. Sohl-Dickstein, “On the Expressive Power ofDeep Neural Networks,” in International Conference on Machine Learning(ICML2016), New York City, NY, USA, 2016.
127. R. Arora, A. Basu, P. Mianj and A. Mukherjee, “Understanding Deep Neural Networks with Rectified Linear Units,” in 6th International Conference on Learning Representations (ICLR 2018), Vancouver, BC, Canada,, 2018.

128. C.-F. Wang, “The Vanishing Gradient Problem: The Problem, Its Causes, Its Significance, and Its Solutions,” 8 Jan 2019. [Online]. Available: https://towardsdatascience.com/the-vanishing-gradient-problem-69bf08b15484.
129. C.-P. Lee, C. Lim and S. Wright, “A Distributed Quasi-Newton Algorithm for Empirical Risk Mini Minimization,” in KDD ’18: The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining,, London, United Kingdom, 2018.
130. A. Jung, “Machine Learning: Basic Principles,” arXiv Preprints, p. arXiv:1805.05052v13, 22 October 2020.
131. S. Clémençon, A. Bellet and I. Colin, “Scaling-up Empirical Risk Minimization: Optimization of Incomplete U-statistics,” Journal of Machine Learning Research, vol. 17, no. 76, pp. 1-36, 2016. [CrossRef]
132. Y. Zhou, “Nonconvex Optimization in Machine Learning: Convergence,Landscape, and Generalization,” Ohio State University, 2018.
133. R. Novak, Y. Bahri, D. Abolafia, J. Pennington and J. Sohl-Dickstein, “SENSITIVITY AND GENERALIZATION IN NEURAL NETWORKS: AN EMPIRICAL STUDY,” arXiv Preprints, p. arXiv:1802.08760v3, 28 June 2018.
134. P. Jain and P. Kar, “Non-convex Optimization for Machine Learning,” Foundations and Trend in Machine Learning, vol. 10, no. 3-4, pp. 142-336, 2017. [CrossRef]
135. E. Mehdi, “Non-Convex Optimization in Deep Learning,” 28 July 2020. [Online]. Available: https://medium.com/swlh/non-convex-optimization-in-deep-learning-26fa30a2b2b3.
136. D. Lopez-Paz and L. Sagun, “Easing Non-Convex Optimization with Neural Networks,” in International Conference on Learning Representations (ICLR 2018), Vancouver, BC, Canada, 2018.
137. V. Cherkassky, “Model complexity control and statistical learning theory,” Natural Computing, vol. 1, pp. 109-133, 2002. [CrossRef]
138. T. Poggio, Q. Liao and A. Banburski, “Complexity control by gradient descent in deep networks,” Nature Communications, vol. 11, no. 1027, pp. 1-6, 2020. [CrossRef] [PMCID]
139. I. Marin, A. Skelin and T. Grujic, “Empirical Evaluation of the Effect of Optimization and Regularization Techniques on the Generalization Performance of Deep Convolutional Neural Network,” Applied Sciences, vol. 10, no. 7817, pp. 1-30, 2020. [CrossRef]
140. D. Soydaner, “A Comparison of Optimization Algorithms for Deep Learning,” International Journal of Pattern Recognition and Artificial Intelligence, vol. 34, no. 13, pp. 1-26, 28 July 2020. [CrossRef]
141. D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in 3rd International Conference on Learning Representations (ICLR2015), San Diego, CA, USA, 2015.
142. Z. Tao, Q. Xia and Q. Li, “A new perspective in understanding of Adam-Type algorithms and beyond,” 25 September 2019. [Online]. Available: https://openreview.net/forum?id=SyxM51BYP. [Accessed 25 March 2022].
143. S. Reddi, M. Zaheer, D. Sachan, S. Kale and S. Kumar, “Adaptive Methods for Nonconvex Optimization,” in 32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada, 2018.
144. I. Goodfellow, Y. Bengio and A. Courville, Deep Learning, MIT Press, 2016.
145. F. Ayo, S. Folorunso, A. Abayomi-Alli, A. Adekunle and J. Awotunde, “Network intrusion detection based on deep learning model optimized with rule-based hybrid feature selection,” Information Security Journal: A Global Perspective, vol. 29, no. 6, pp. 267-283, 2020. [CrossRef]
146. F. Zhang and D. Wang, “An Effective Feature Selection Approach for Network Intrusion Detection,” in 2013 IEEE Eighth International Conference on Networking, Architecture and Storage, Xi'an, China, 2013. [CrossRef]
147. T. Zhang, “Application and Research of Deep Neural Network Model in Computer Network Intrusion Detection,” in 2019 3rd Scientific Conference on Mechatronics Engineering and Computer Science (SCMC 2019), 2019.
148. S. Kabaivanov and V. Markovska, “Hybrid deep-learning analysis for cyber anomaly detection,” IOP Conference Series: Materials Science and Engineering, vol. 878, no. 012029, pp. 1-7, 2020. [CrossRef]
149. L. Wu, Z. Zhu and E. Weinan, “Towards Understanding Generalization of Deep Learning: Perspective of Loss Landscapes,” arXiv preprint, p. arXiv:1706.10239, June 2017.
150. R. Sahu and S. Panigrahi, “Application of Deep Learning for Database Intrusion Detection,” in Advanced Computing and Intelligent Engineering, vol. 1082, B. Pati, C. Panigrahi, R. Buyya and K. Li, Eds., Springer Nature, Singapore , 2020. [CrossRef]

AUTHORS PROFILE

Stephen Kahara Wanjau, received his B.Sc. Degree in Information Sciences from Moi University, Kenya, in 2006 and MSc. Degree in Computer Systems from Jomo Kenyatta University of Agriculture and Technology, Kenya, in 2018. Currently, he is pursuing a PhD in Computer Science at Murang’a University of Technology. He is currently serving as the Director of ICT at Murang’a University of Technology, Kenya. His research interests include machine learning, network security, network intrusion detection, and big data analytics.

Geoffrey Mariga Wambugu, received his B.Sc. degree in Mathematics and Computer Science from Jomo Kenyatta University of Agriculture and Technology (JKUAT), Juja, Kenya, in 2000, the M.Sc. degree in Information Systems from The University of Nairobi, Nairobi, Kenya, in 2012, and the Ph.D. degree in Information Technology JKUAT, in 2019.He have servedfor over 10 years’ as head of department in higher education institutions in Kenya and also been involved in the design, development, review and implementation of Computing Curricula in different universities in Kenya. Currently he is a Senior Lecturer and Dean, School of Computing and Information Technology in Murang’a University of Technology. His research interests include Probabilistic Machine Learning, Text Analytics, Natural Language Processing, Data mining, and Big Data Analytics.

Aaron Mogeni Oirere, received his B.Sc. degree in Computer Science from Periyar University, Salem, Tamilnadu, India in 2007, the M.Sc. degree in Computer Science from Bharathiar University, Coimbatore, Tamilnadu, India in 2010, and the Ph.D. degree in Computer Science from Dr. Babasaheb Ambedkar Marathwada University, Maharashtra, India in 2016. He currently works at the Department of Computer Science, school of computing and Information Technology, Muranga University of Technology. His research interest include Automatic Speech Recognition, Human-computer Interaction, Information Retrieval, Database Management Systems (DBMS), Data Analytics and Hardware & Networking.