Deep Learning Models in Computer Data Mining for Intrusion Detection

Yujun Wang
Department of Science and Technology, Xi'an Siyuan University, Xi'an, Shaanxi 710038, China
E-mail: xawyj@163.com

Keywords: intrusion detection system (IDS), DL, Data mining (DM), and CNN

Received: June 15, 2023

In recent years, the expanded usage of wireless networks for the transfer of enormous amounts of data has caused a multitude of security dangers and privacy issues; accordingly, a variety of preventative and defensive measures, such as intrusion detection (ID) systems, have been developed. ID methods serve a crucial role in safeguarding computer and network systems; yet, performance remains a serious concern for many IDS.The effectiveness of IDS was analyzed by constructing an IDS dataset comprised of network traffic characteristics to identify attack patterns. ID is a classification challenge requiring the use of DL and Data Mining (DM) methods to categorize network data into regular and attack traffic. In addition, the kinds of network assaults have evolved, necessitating an upgrade of the databases used to evaluate IDS. In this study, we present a DL-based IDS that combines an optimization technique called spider monkey swarm with a convolutional neural network (SMSO-CNN). With the use of the well-known NSL-KDD dataset, the SMSO-CNN is assessed and contrasted with the following methods: DNN, k-nearest neighbor, and LSTM. The results show that the SMSO-CNN outperforms compared to other approaches in terms of accuracy.

Povzetek: Predlagana spremenjena strategija algoritma SMO (verzija AI algoritma na osnovi roja) izboljša globalno konvergenco in presega standardni SMO na 20 testnih funkcijah.

1 Introduction

An ID is the process of reviewing the system logs to check for footprints and identify any interaction. ID has been achieved throughout the years using a variety of methods, including statistical, bio-inspired, fuzzy, Markov, etc. [1]. An IDS is important because it may warn companies about potential security issues and provide early detection. It reduces the danger of data breaches, illegal possession of confidential data, and disruption of crucial systems by quickly spotting and reacting to intrusions. In addition, an IDS can help with emergency response efforts by offering important details about the type of an attack and supporting the containment, investigation, and recovery processes [2]. Distributed computer systems' complexity, significance, and informational resources have grown extremely quickly. This reality has led to an increase in computer crimes in recent years, which have increasingly targeted computers and their networks [3]. IDS was any piece of hardware, software, or a hybrid that keeps an eye out for harmful behavior on a system or network of computers. Any IDS's ultimate objective was to apprehend offenders in the act before they seriously harm resources. An IDS guards against compromise, abuse, and assault on a system. Moreover, it examined data integrity, audits network and system settings for vulnerabilities, and monitors network activities. These days, IDS was a crucial part of the security toolkit. Three services were offered by an IDS: monitoring, detection, and alarm generation. IDS are often seen as a firewall feature. Together, firewalls and IDS improve network security [4]. A foundation for computer and information security. Its fundamental objective was to distinguish between normal system activity and activity that can be considered suspicious or invasive. IDS were required since there are more reported events every year and attack methods are always evolving. The two basic kinds of IDS methods are abuse and anomaly detection. According to the misuse detection technique, an intrusion may be identified by comparing the present activity to a list of invasive patterns. Expert systems, keystroke tracking, and state transition analysis were a few examples of abuse detection. Systems for detecting anomalies assume that an intrusion should cause a departure from the system's typical behavior. Statistical approaches, neural networks, the creation of prediction patterns, association rules, and other methods may all be used to accomplish these strategy [5]. The ability of network managers to identify policy breaches has led to the widespread adoption of Network ID Systems. These policy breaches vary from insiders misusing their access to outside adversaries seeking to get illegal access [6]. For remedial action to be taken, as an alternative, detecting policy violations enables managers to spot weak spots in their defenses [7]. IDS accurately identifies the information and forecasts outcomes that may be used in the future [8]. As they proposed, intelligent network IDS based on SMSO-CNN. SMSO was an extension of the CNN approach, which relies on conditional independence assumption. CNN enhances the attack detection accuracy.

Reference	Objectives	Methods	Strength	Weakness	Results
[9]	The goal of the paper was to discuss the difficulties associated with wireless identification, including signal strength variations, noise, and interference.	DL-based IDS that combines a feature selection approach	Enhanced Model Performance, Improved Efficiency, Transferability to Different Domains.	Need a Complexity and Computational Requirements. Dependency on High-Quality Training Data and Vulnerability to Adversarial Attacks.	The experimental findings demonstrate that, in comparison to previous approaches, the FFDNN-IDS achieves a higher accuracy.
[10]	The purpose of the paper proposing the use of a BD-based (Big Data-based) DL-based system for intrusion detection is to create a practical and scalable method for identifying and containing network intrusions in large-scale and complex environments.	Big data (BD)-based Hierarchical DL System	It improves the identification rate of intrusive assaults relative to single modeling approaches by utilizing numerous DNN models	Privacy and Security Concerns, Adaptability to Evolving Attacks, Resource Requirements for Training	The outcomes shows that how well the system was able to recognise and categorise various forms of intrusions.
[11]	our study aimed to examine an extensive variety of detection of network intrusion models.	BD-related network ID algorithms	Big Data (BD)-related network intrusion detection (ID) algorithms have several strengths that make them effective in detecting and mitigating security threats in large-scale network environments.	Computational Complexity, data storage and management, Privacy and Ethical Considerations.	This survey examined research on big data analytics, focusing on its challenges and various intrusion detection techniques specifically designed for big data analytics.
[12]	It aimed to provide a thorough analysis of contemporary work on the IoT and ML, as well as intelligent approaches and ID structures in computer networks.	Machine learning based ID approaches.	Automated Detection, Enhanced Detection Accuracy, Integration with Other Security Systems.	Dependency on Training Data, Adversarial Attacks, Concept Drift and Evolving Attacks.	Over 95 pertinent works were reviewed for the study, which covered a range of topics relating to security concerns in IoT systems.

[13]	The research aimed to integrate feature selection and preprocessing strategies to improve the performance of Deep Neural Networks (DNN) for intrusion detection.	feature selection and layer design strategies	Enhance the performance of a ML model	Computational Complexity, Generalizability to New Attacks	Results demonstrate that judicious feature selection and layer configuration can effectively shorten the learning period of the model without sacrificing its overall accuracy.
[14]	To identifies the well-known security threats	DL-based ID Systems	Automated Feature Learning, Improved Detection Accuracy, Real-Time Detection and Response.	High Computational Requirements, Dependency on Large Labeled Datasets, Vulnerability to Adversarial Attacks.	The architecture of the Cisco IoT reference model was specifically discussed
[15]	It aimed to investigate and propose effective strategies for data processing and model selection in the context of network intrusion detection using machine learning techniques.	A machine-learning approaches for IDS.	Automatic Detection, Adaptability to Evolving Threats, Improved Detection Accuracy.	Dependency on Labeled training Data, Adversarial Attacks, Privacy and Ethical Considerations.	The study evaluated how nonlinear classifiers beat linear ones and how data normalization and balancing increased most classifiers' overall performance.
[16]	The aim of the study was to improve the IoT security performance.	A deep-learning approach was used to create an algorithm for identifying denial-of-service (DoS) assaults.	Automatic Feature Extraction, Improved Detection Accuracy, Adaptability to Emerging Attack Patterns.	Need lot of data Availability and Representation, Vulnerability to Adversarial Attacks.	The results show that deep-learning models provide more accuracy, enabling more efficient attack prevention on IoT networks.
[17]	The goal of the paper was to compile and assess the body of literature in the area, highlighting the various ML and DL algorithms employed for intrusion detection and the	Taxonomy for ID systems	Clarity and Organization, Decision-Making and Evaluation, Scalability and Flexibility	Security vulnerabilities, Discrimination and exclusion, Lack of standardization and interoperability.	This paper examines and improves the existing challenges and future trends in the field, serving as a valuable reference for researchers conducting

	performance in diverse application domains.				extensive studies.
[18]	Aimed to tackle the problem of interpreting firewall logs.	ML and DL methods	Ability to handle large and complex datasets, Automation and efficiency, Adaptability and generalization, Handling non-linear relationships.	Computing power needed for inference and training. Particularly DL models have a tendency to require high levels of processing, memory, and storage.	Numerous ML and DL algorithms, such as K-Nearest Neighbor (KNN), NB (NB), J48, Random Forest (RF), and Artificial Neural Network (ANN), were evaluated
[19]	The purpose of the research was to investigate and suggest effective model selection strategies and data processing techniques for ML-based network intrusion detection systems.	Numerous ML-based anomaly-based Intrusion Detection Systems (IDS)	Enhanced detection capabilities, reduced false positives, Detection of unknown attacks, Adaptability to changing threats.	Adversarial Attacks, Training Data Limitations.	Our findings show that non-linear classifiers perform better than linear ones overall, and that using data balance and normalization approaches increases the accuracy of most classifiers.
[20]	The purpose of this study was to evaluate several machine learning methods for traffic classification in relation to intrusion detection systems (IDS). The CICIDS2017 dataset, which contains bidirectional traffic flows representing both benign traffic and various modern assaults, was the main focus of the authors' work.	correlation-based feature selection (CFS) technique	Improved Classification Performance, Reduced Overfitting, Faster Training and Inference	Computational Complexity, Sensitivity to Noise, Ignores Non-Correlated Attributes.	The findings demonstrated that decision-tree-based approaches (PART, J48, and random forest) were the most effective, averaging F1 values above 0.999 for the whole dataset
[21]	The aim of this work is to evaluate a	method that combines innovative	Enhanced Feature Representation, Discovering	Complexity, Data Requirements, Interpretability	Performance indicators like precision,

	medical database of diabetes patients using a combination of innovative hierarchical decision attention network, association rules (AR), and multiclass outlier classification with the MapReduce framework.	hierarchical decision attention networks, association rules (AR), and multiclass outlier classification	Associations, Scalability and Efficiency		accuracy, recall, and F-score are used to display the results of the suggested method
[22]	The objective of this paper is to compare various deep learning (DL)-based network intrusion detection systems (IDSystems). The research also seeks to improve DL models with Generative Adversarial Networks (GANs).	Generative Adversarial Network	Technique is most suited for the NIDS application, Learning Data Representation	Training Instability, Evaluation Metrics, Sensitivity to Hyperparameters	Evaluation metrics are constructed to evaluate each algorithm's performance, and adding synthetic data produced by GAN is meant to increase the NIDS's overall accuracy.
[23]	Emphasized the value of sequential data modeling in cybersecurity, an area where temporal features were key.	Recurrent neural networks (RNNs), a subclass of artificial neural networks (ANNs)	Handling Sequential Data	Vanishing and Exploding Gradient Problems	Compared the various RNN designs to conventional machine learning classifiers, experiments showed a decreased percentage of false positives.
[24]	The purpose of this study is to outline a successful feature engineering approach for Deep Neural Networks (DNNs) in the context of Intrusion	Deep neural network	Capability to Learn Complex Patterns, End-to-End Learning	Need for Large Amounts of Labeled Data, High Computational Requirements.	Using the benchmark ID dataset, a thorough comparison of trials in DNN with various machine-learning methods was conducted

	Detection (ID) in the Internet of Medical Things (IoMT) architecture. The suggested method uses a hybrid strategy that combines Grey Wolf Optimization (GWO) with Principal Component Analysis (PCA), or PCA and PCA, respectively.				
[25]	Aimed to planning and execution of the detection system, it gives thorough information.	Intrusion detection method	Threat Detection, Real-time Monitoring	Dependence on Signature Databases	The test results show that the system satisfies wireless sensor network ID requirements with high accuracy and speed

To overcome the issues of this paper we proposed the SMSO-CNN method which shows that better the outcomes compared to this existing work.

2 Contribution of the study

Thus, this research contributes by demonstrating an implementation of the SMSO-CNN to increase its effectiveness by boosting students' interest, motivation, and the field's relevance to their lives. The following are some of the particular accomplishments of this paper:

- The approach of ID based on Spider Monkey Swarm Optimization is examined.
- To enhance the DL method of CNN which is the ability to handle large amounts of data and assess the effectiveness of the process, an efficient learning component.

3 Proposed methods

In this part, we defined the technique utilized to create the model, outlined the primary processes that were taken to construct the model, and provided an in-depth description of how the steps of the suggested model in Figure 1 were developed. This discussion is broken up into four sections: The first section is devoted to

information gathering. The discretization procedure, feature selection and extraction methods, and other data pre-processing methods are discussed in the second section. The most crucial information is presented in the third section, which details the work done to construct the suggested model and compile the foundational experiences. The fourth step is to assess the success of each current and new model by comparing their respective parameters.

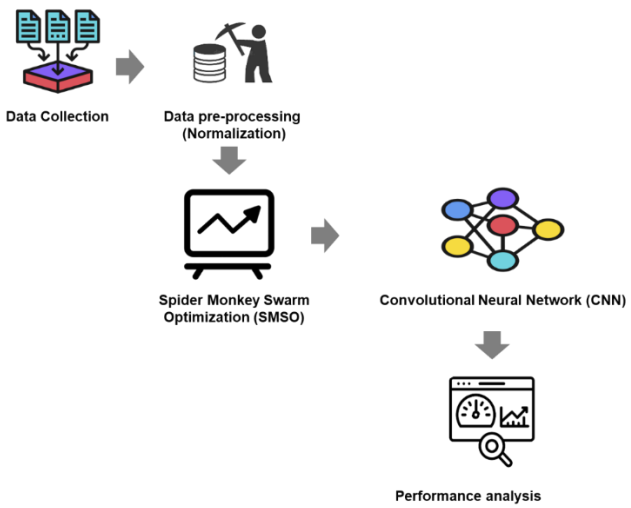

Figure 1: Proposed method framework

A. Data set

The NSL-KDD dataset is a widely used benchmark dataset in the field of network ID and DL research. It was created to addressing some of its limitations and providing a more suitable dataset for evaluating DL models in computer data mining for intrusion detection. The NSL-KDD consists of

41 characteristics, three of which are non-numeric and the other 38 numerical.

The SMSO-CNN model uses CNN to extract features from the sequential nature of SMS messages, capturing patterns and semantics particular to text data. Therefore, using the NSL-KDD dataset, which lacks SMS-specific properties and context, may not offer the essential data representation for efficiently assessing the performance of the SMSO-CNN model. A brand-new data set called NSL-KDD is suggested as a solution to these problems.

B. Data pre-processing

Use a range of preparation procedures to address missing, noisy, and data inconsistency, as well as to clean datasets. Part of the process of cleaning data involves getting rid of things like blank fields, redundant entries, syntax errors, and missing codes. Consistent data-cleaning methods have been applied to the datasets so that the cleaned data can be easily obtained and analyzed. The process of normalization requires the generation of brand-new data vectors. Reducing the likelihood of data redundancy is a major advantage of normalization. The Min-Max normalized approach is crucial for data integration and standardization. The values for each characteristic may be anywhere from 0 to 1, where 0 is the minimum and 1 is the maximum. The normalizing procedure may be expressed as,

$$W_{norm} = \frac{W_j - W_{min}}{W_{max} - W_{min}} \quad (1)$$

When dealing with data in groups, the lowest and highest values are denoted by W_{min}andW_{max}, respectively, where W_j is a data point. The algorithm known as Recursive Feature Elimination (RFE) ranks the most important features and produces a number representing their relative importance. Feature subsets are predicted to shrink when redundancies are eliminated. Sorting the variables from most essential to least interesting may be used to establish the ranking of qualities.

C. Spider monkey swarm optimization coupled with convolutional neural networks (SMSO-CNN)

The CNN's parameters or architecture are optimized using the SMSO algorithm when SMSO is used with a CNN. CNNs are a common deep learning model type for image processing and pattern recognition tasks. Finding the ideal values for various parameters, such as filter sizes, number of layers, learning rates, etc., is typically how CNNs are optimized. Particularly for complex networks or sizable datasets, this procedure can be time-consuming and computationally expensive. The algorithm can be used to automatically look for the ideal values of these parameters by integrating SMSO with CNNs.

SMSO is an algorithm proposed on the premise of swarm intelligence; it employs a cluster of SMs whose behavior is modeled after that of the foraging behavior of honey SMSs. When there are fewer monkeys in one group, fission occurs and the fusion time is established. The algorithm relies on the social structure of a set of traits from the leader, who decides to either pool resources or move in separate directions in the quest for food. Each subgroup also has a leader; however, they report to the global leader. This is the property of spider monkeys:

- There are 40- 50 spider monkeys in each band.
- The eldest woman in each group acts as a GL and makes almost all of the party's choices.
- Each smaller group is led by a local woman who is in charge of organizing the foraging schedule.

Here are more details about the main parts of Spider Monkey Optimization.

3.1 Setting up the population

Each spider monkey's starting location in the population is represented by its initial parameters, TN_{or} (o=1, 2... N), an N-D vector where N specifies the number of issue variables to be improved. Each SM pinpoints an achievable goal that might fix the issue. It is defined as Eq. for each TN_{or}(1)

$$TN_{or} = TN_{minq} + VQ(0,1) \times \left(TN_{maxq} - TN_{minq}\right) \quad (2)$$

WhereTN_{maxq} and TN_{minq} are minimum and maximum values of TN_{or} in the direction and (0, 1).

3.2 Local leader phase

At this step, the SMO updates its actual role related to the decisions of its local group and local leader (LL), and it also determines the fitness values for the positions of any newly arrived monkeys. This is the stage when Spider monkeys must increase their fitness by replacing their previous positions with new ones. The equation for the othTN's position is as follows,

$$TN_{newor} = TN_{or} + VQ(0,1) \times (KK_{kr} - TN_{or}) + VQ(-1,1) \times (TN_{qr} - TN_{or}) \quad (3)$$

In this case, the o th dimensions of the kth LL position corresponds to the r th component of the kth SM. The dimensional TN_{qr} is the rthTN picked at random from the kth group where r is less than or equal to V in the r th dimensions.

3.3 Global leader phase

Members of both the GL and LL groups share their insights to aid in the spider monkeys' stance adjustment. The coordinates may be found by,

$$TN_{newor} = TN_{or} + VQ(0,1) \times (HK_{kr} - TN_{or}) + VQ(-1,1) \times (TN_{qr} - TN_{or}) \quad (4)$$

Where (r = 1, 2,) N is a randomly chosen index and GLj is the rth dimension of the GL location. At the GLP stage, spider monkeys (TN_{or}) have their positions updated according to the ri values of the probabilities that are taken into account for calculating their fitness. This manner, the most qualified applicant may best present themselves. The

following equation may be used to determine the probability of ri:

$$ri = (fitness\ ix/fitness\ max)\ +\ 0.1 \quad (5)$$

Wherefitness max is the highest possible fitness level for the oth$N$'s group. In addition, the optimal location is selected by calculating a new fitness algorithm that relies on the created position and comparing it to the previous fitness parameter.

3.4 Global leader learning segment

In the GLL segment, the pessimistic model is used to update and perform the feature extraction. The population is used to choose and create the fitness function value. The optimal value of the place determines the value of the world leader. Instead of updating, the value is increased by one and stored in the Global Limit Count variable.

3.5 Local leader learning phase

According to the fitness values of a community organization, the LLL is changed in the SM location, making it the best possible choice for the local community. It's worth whatever the current regional authority decides it's worth. As it increases by one with each new LLC, no additional updates are supplied.

3.6 Local leader decision phase

If the LLD doesn't update its location using initial randomization or the knowledge of the GL and LL, it does so use the perturbations rate,

$$TN_{newor} = TN_{or} + VQ(0,1) \times (HK_{kr} - TN_{or}) + VQ(0,1) \times \left(TN_{qr} - KK_{or}\right) \quad (6)$$

3.7 Global leader decision phase

At this stage, the GL positioning is monitored for a certain amount of time. The GL then divides the population into subsets, always beginning with at least two and going as high as feasible. At the GLD stage, new groups are established and LLL operations are initiated to choose the LL. The GL is unable to change its location. In addition, when the optimum number of distinct groups is reached, it takes its cue from the spider monkey's fusion-splitting social structure and merges all of the smaller groups into a one, super group.

Fitness is calculated by summing the relative relevance of each attribute. Each aspect of the input data is given a score based on the goal variables. When the likelihood of reaching the node drops before it is reached, the relevance of the feature is calculated based on the impurities of the junction with the values. We may get the node's probability by dividing the ratio of the observed numbers by the total number of specimens. For optimal feature selection, we utilize to determine the fitness function.

$$fitness_{\ feature\ importance}$$

$$= \frac{Number\ of\ specimens\ that\ reach\ the\ nodes}{Total\ number\ of\ samples} \quad (7)$$

Utilizing the low-level co-evolutionary traits, the SMSO hybridized algorithm creates the hybrid mixed capability. There are merge and combine options available as part of the basic hybrid capability. Co-evolutionary is used because variations are employed sequentially, in parallel. The two types are combined, and both contribute to the creation of answers to the challenges. With this adjustment, the hierarchical SMSO generates variations using the strength of SMSO. The velocity is revised using the combined SMSO variations, as suggested,

$$u_j^{l+1} = x * (u_j^l + d_1 q_1 \left(w_{1-} w_j^l\right) + d_2 q_2 (w_{2-} w_j^l) + d_3 q_3 (w_{3-} w_j^{l+1})) \quad (8)$$

$$w_j^{l+1} = w_j^l + u_j^{l+1} \quad (9)$$

The most optimal value is chosen by maximizing the fitness value. The proposed method employs the Rosen Brock function, often called the optimization problem. With the in-built localized without a framework to guide and a proper coordinate system, the Rosen Brock product is effectively maximized,

$$e(w) = \sum_{j=1}^{M-1} \left[100(y_{j+1} - y_j^2)^2 + (1 - y_j)^2\right] \quad (10)$$

Each variable's goal function is added together to get the best possible outcome. The equation gives the generic form of the optimal solution,

$$Minimize\ or\ Y\ or\ hbest\ = \sum_{j=1}^{m} d_j Y_j \quad (11)$$

Where$Y_j$the ith control is input and DJ is the optimization problem factor for the ith parameter.

Hence, a function is used to choose the optimal set of characteristics from the subgroup, and data augmentation is calculated if there is any ambiguity among the features.

D. Convolutional neural network

The input data, convolutional, pooling surface, FC overlay, and output vector are the five components that make up a CNN. There are several layer configurations among CNNs. Figure 2 depicts the structure of the CNN that was employed in this investigation.

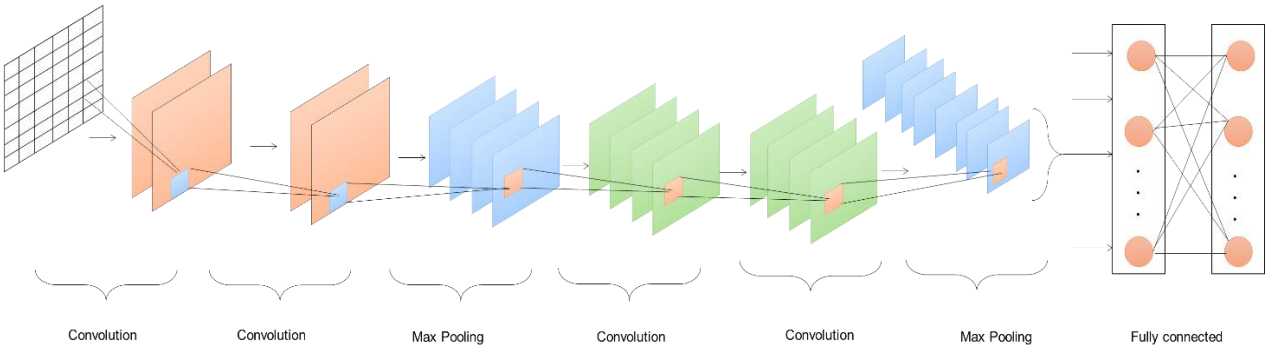

Figure 2: Structure of convolutional neural network

The convolutional gradient task is to identify interesting patterns in the data. Each convolutional kernel in its many layers is associated with a frequency and a divergence coefficient. It is assumed that u_j is the weight parameter, a_j is the divergence amount, and $V_{j-1}$is the input to convolution layers j while inversion kernel j is active. One such expression for the convolution operation is:

$$V_j = e\left(u_j \otimes V_{j-1} + a_j\right) \tag{12}$$

Where, the output result of convolution kernel j, $\otimes$ represents the convolution operation, and $e(x)$ represents the activation function.

The input data is swept repeatedly by the CNN, which then extracts the distinctive information. In addition, the multilayer layer's operational amplifier is changed to $ReLU$. The Linear transfer function is simpler to derive than the exponential, transfer function, and another training algorithm, allowing for faster model training and better protection against gradient disappearing. It is possible to write $ReLU$ as:

$$ReLU\left(V_j\right) = \begin{cases} V_j\left(V_j > 0\right) \\ 0\ \left(V_j > 0\right) \end{cases} \tag{13}$$

Downsampling data redundancy is the primary operation of the pooling layer, which also helps to attain invariance and minimize CNN complexity. Pooling layer and maximum pooling are the two most common approaches to completing pooling. If you use averaged pooling, the result is the arithmetic mean of the computation area, whereas if you use max pooling, the result is the largest value of the area.

As max pooling is superior to average pooling at preserving crucial data, it was chosen for this analysis. The mathematical formula for max pooling is:

$$R_i = max(O_i^0, O_i^1, O_i^2, O_i^3, \ldots . O_i^s,) \tag{14}$$

Where$R_i$the return outcome of the pooled region I, Max is the maximum pooling procedure, and O IS is the pooling area i's element s.

The "classifiers" of a CNN are the FC layers. Its primary purpose is to reconfigure the information from the hidden-layer space that the convolutional as well as pooling layers extracted and weighted.

4 Result and discussion

To apply the recommended methods for detecting intrusion using the Spider Monkey Swarm Optimization - Convolutional Neural Network (SMSO-CNN) approach, DL methodology was used. We employ indicators like accuracy, precision, detective rate, and false alarm rate for analysis.

Confusion matrix

When it comes to assessing IDS, this matrix is among the top options. Each section in this matrix indicates the predicted class, and each row depicts the actual section; the performance of the model is determined by several metrics. The accuracy rate of the classification is determined by comparing the actual number of records categorized with the number of anticipated records. The matrix's contents are summarized by four factors shown in Table 1.

Table 1: Confusion matrix evaluation

Present		Predictive value	
		Positive	Negative
Class	P	TP	FP
	N	FN	TN

Accuracy

Accuracy is a metric used to assess how well a classification model performs in the context of ML and statistics. It calculates the ratio of the model's accurate predictions to all of the predictions made.

$$\text{Accuracy} = (\text{Truepositives} + \text{TrueNegatives})/(\text{Truepositives} + \text{Truenegatives} + \text{Falsepositives} + \text{Falsenegatives}) = (\text{TP} + \text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN}) \quad (15)$$

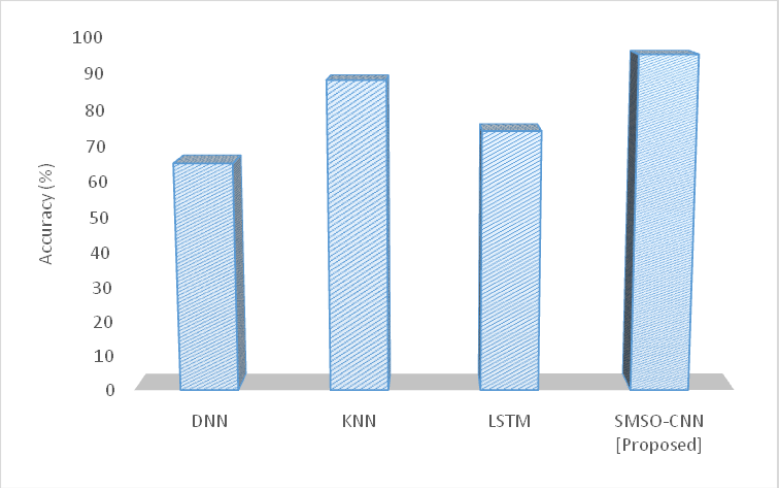

Figure 3: The Accuracy of the proposed and existing system

The accuracy of the suggested strategy is shown in Figure 3. Accuracy percentages are often provided for accuracy levels. Both the existing approach and the one that is being discussed show signs of the potential for inaccurate estimates. This threat is recognized by both systems. The recommended method, SMSO-CNN, achieves 95% accuracy in contrast to DNN [23], KNN [24], and LSTM [25], 65%, 88%, and 74%, respectively. Thus, the strategy that is suggested has the highest accuracy rate. Table 2 displays the accuracy of the suggested strategy.

Table 2: Comparison of accuracy

Methods	Accuracy (%)
DNN	65
KNN	88
LSTM	74
SMSO-CNN [Proposed]	95

Precision

Precision is a performance parameter that is used to assess the efficacy of a classification model, particularly in binary classification issues. Out of all the occurrences the model predicted as positive, it counts the percentage of accurately predicted positive instances.

$$\text{Precision} = \text{True positives}/(\text{True positives} + \text{False positives}) = \text{TP}/(\text{TP} + \text{FP}) \quad (16)$$

The precision for the suggested system is shown in Figure 4. Thus, the method that is advised has the best precision. Table 3 displays the recommended approach precision.

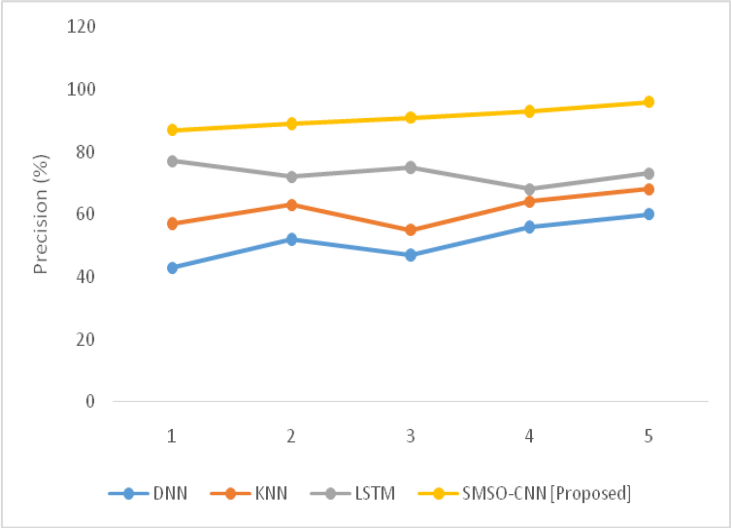

Figure 4: The precision of the proposed and existing method

Table 3: Comparison of precision

Dataset	Precision (%)			
	DNN	KNN	LSTM	SMSO-CNN [Proposed]
1	43	57	77	87
2	52	63	72	89
3	47	55	75	91
4	56	64	68	93
5	60	68	73	96

Detection rate (DR)

TP and FN are the totals for true positives and false negatives, respectively, while DR is the ratio of true positives to all nonself samples detected by the detector set.

$$\text{DetectiveRate} = \frac{\text{TP}}{\text{TP+FN}} \times 100 \quad (17)$$

There is agreement on the definition of the detection rate, is also called astrue positive rate, and Figure 5 shows the suggested technique.

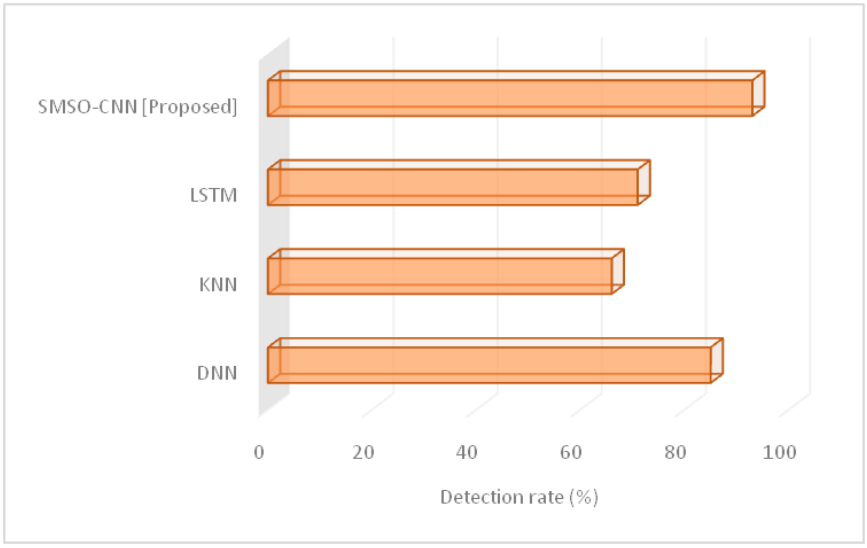

Figure 5: The detection rate of the proposed and existing method

When compared to DNN, KNN, and LSTM, the recommended approach, SMSO-CNN, obtains a 93% detection rate while only having 85%, 66%, and 71% accuracy, respectively. Thus, the strategy that is suggested has the highest accuracy rate. Table 4 displays the recommended approach detection rate.

Table 4: Detection rate comparison

Methods	Detection rate (%)
DNN	85
KNN	66
LSTM	71
SMSO-CNN [Proposed]	93

False alarm rate

The proportion of benign events that have caused a false alarm is known as the false alarm rate, also calledas the false positive rate, whereas the false alarm detection rate (FDR) gauges the percentage of irrelevant notifications.

$$\text{False alarm Rate} = \frac{\text{FP}}{\text{FP+TN}} \times 100 \quad (18)$$

A high FNR will make the system open to intrusions, while a high FPR will significantly affect how well the IDS performs. Figure6 shows the suggested False Alarm Rate calculation technique.

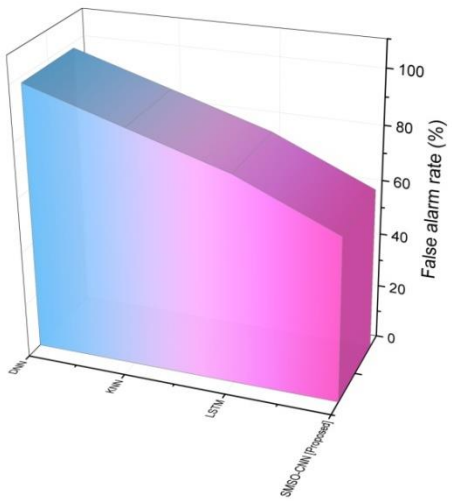

Figure 6: The false alarm rate of the proposed and existing method.

The recommended method, in contrast, obtains a detection rate of 60% whereas DNN, KNN, and LSTM have accuracy rates of 98%, 87%, and 76%, respectively. Thus, the strategy that is suggested has the highest accuracy rate. Table 4 displays the recommended approach detection rate.

Table 5: Comparison of false alarm rate

Methods	False alarm rate (%)
DNN	98
KNN	87
LSTM	76
SMSO-CNN [Proposed]	60

5 Discussion

Deep Neural Networks (DNNs) are employed in many different fields, but they also have some drawbacks that are large data requirements, Limited performance on small datasets, Vulnerability to adversarial attacks. KNN have some issues in intrusion detection includes the high computational cost, Sensitivity to irrelevant features, Imbalanced dataset challenges and the traditional work of LSTM have some drawbacks in intrusion detection that are training data imbalance, and computationally expensive. In order to overcome these issues SMSO-CNN models were used in this paper. The outcomes show that the SMSO-CNN method performs better than the others in terms of accuracy. According to the results, the suggested SMSO-CNN method performs better than previous methods and is useful for recognizing network assaults. The IDS improves its accuracy in separating ordinary traffic from attack traffic by fusing DL approaches with spider monkey swarm optimization.

6 Conclusion

In this study, we detailed the planning, development, and testing of an SMSO-based DL IDsystem. After researching several DL methods, researchers concluded that there is no one best way to do intrusion detection. Our research led us to the conclusion that the NSL-KDD collection is one of the most reliable control sets for use in simulating IDSs. By studying and assessment of the NSL-KDD dataset, the suggested hybrid model based on DL technology as the last step increases the detection performance increases accuracy, and decreases false alarms. Preparing and analyzing the dataset before the pre-processing step is crucial to building an effective model. This includes picking the right features, shrinking the dimensions, and using the discretization approach to boost IDaccuracy.Research into IDtechnologies has gained traction in response to the growing importance of network security threats. In-depth study on data mining IDmethods led to the development of an outlier-based algorithm for intrusion detection. People have long worried about keeping their networks safe and secure. There are still a lot of studies to be done and difficult problems to be addressed quickly as the network continues to evolve and cyber assaults become more varied.

References

[1] Zamani, M. and Movahedi, M., 2013. ML methods for intrusion detection. arXiv preprint arXiv:1312.2177.

[2] Tsai, C.F., Hsu, Y.F., Lin, C.Y. and Lin, W.Y., 2009. IDby ML: A review. expert systems with applications, 36(10), pp.11994-12000.

[3] Tabash, M., Abd Allah, M. and Tawfik, B., 2020. IDmodel using NB and DL technique. Int. Arab J. Inf. Technol., 17(2), pp.215-224.

[4] Vinchurkar, D.P. and Reshamwala, A., 2012. A review of an IDsystem using neural network and ML. J. Eng. Sci. Innov. Technol, 1, pp.54-63.

[5] Florez, G., Bridges, S.A. and Vaughn, R.B., 2002, June. An improved algorithm for fuzzy data mining for intrusion detection. In 2002 Annual Meeting of the North American Fuzzy Information Processing Society Proceedings. NAFIPS-FLINT 2002 (Cat. No. 02TH8622) (pp. 457-462). IEEE.

[6] Brugger, S.T., 2004. Data mining methods for network intrusion detection. University of California at Davis.

[7] Sahani, R., Rout, C., ChandrakantaBadajena, J., Jena, A.K. and Das, H., 2018. Classification of IDusing data mining methods. In Progress in Computing, Analytics, and Networking: Proceedings of ICCAN 2017 (pp. 753-764). Springer Singapore.

[8] Noel, S., Wijesekera, D. and Youman, C., 2002. Modern intrusion detection, data mining, and degrees of attack guilt. Applications of data mining in computer security, pp.1-31.

[9] Kasongo, S.M. and Sun, Y., 2019. A DL method with filter-based feature engineering for wireless IDsystem. *IEEE Access*, *7*, pp.38597-38607.

[10] Zhong, W., Yu, N. and Ai, C., 2020. Applying BD-based DL system to intrusion detection. *BD Mining and Analytics*, *3*(3), pp.181-195.

[11] Sudar, K.M., Nagaraj, P., Deepalakshmi, P. and Chinnasamy, P., 2021, January. Analysis of intruder detection in BD analytics. In *2021 International Conference on Computer Communication and Informatics (ICCCI)* (pp. 1-5). IEEE.

[12] Da Costa, K.A., Papa, J.P., Lisboa, C.O., Munoz, R. and de Albuquerque, V.H.C., 2019. IoT: A survey on

ML-based IDapproaches. *Computer Networks*, *151*, pp.147-157.

[13] Woo, J.H., Song, J.Y. and Choi, Y.J., 2019, February. Performance enhancement of DNN using feature selection and preprocessing for intrusion detection. In *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)* (pp. 415-417). IEEE.

[14] Idrissi, I., Azizi, M. and Moussaoui, O., 2020, October. IoT security with DL-based IDSystems: A systematic literature review. In *2020 Fourth international conference on intelligent computing in data sciences (ICDS)* (pp. 1-10). IEEE.

[15] Sahu, A., Mao, Z., Davis, K. and Goulart, A.E., 2020, May. Data processing and model selection for ML-based network intrusion detection. In *2020 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR)* (pp. 1-6). IEEE.

[16] Susilo, B. and Sari, R.F., 2020. IDin IoT networks using a DL algorithm. *Information*, *11*(5), p.279.

[17] Liu, H. and Lang, B., 2019. ML and DL methods for IDsystems: A survey. *applied sciences*, *9*(20), p.4396.

[18] Aljabri, M., Alahmadi, A.A., Mohammad, R.M.A., Aboulnour, M., Alomari, D.M. and Almotiri, S.H., 2022. Classification of firewall log data using multiclass ML models. *Electronics*, *11*(12), p.1851.

[19] Sahu, A., Mao, Z., Davis, K. and Goulart, A.E., 2020, May. Data processing and model selection for ML-based network intrusion detection. In 2020 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR) (pp. 1-6). IEEE.

[20] Rodríguez, M., Alesanco, Á.,Mehavilla, L. and García, J., 2022. Evaluation of ML Methods for Traffic Flow-Based Intrusion Detection. *Sensors*, *22*(23), p.9326.

[21] Jayasri, N.P. and Aruna, R., 2022. BD analytics in health care by data mining and classification methods. *ICT Express*, *8*(2), pp.250-257.

[22] Sekhar, C., Kumar, P.H., Venkata Rao, K. and Krishna Prasad, M.H.M., 2022. A Comparative Study on Network IDSystem Using DL Algorithms and Enhancement of DL Models Using Generative Adversarial Network (GAN). In *High-Performance Computing and Networking: Select Proceedings of CHSN 2021* (pp. 143-155). Singapore: Springer Singapore.

[23] Vinayakumar, R., Soman, K.P. and Poornachandran, P., 2017. Evaluation of recurrent neural network and its variants for IDsystem (IDS). *International Journal of Information System Modeling and Design (IJISMD)*, *8*(3), pp.43-63.

[24] RM, S.P., Maddikunta, P.K.R., Parimala, M., Koppu, S., Gadekallu, T.R., Chowdhary, C.L. and Alazab, M., 2020. An effective feature engineering for DNN using hybrid PCA-GWO for IDin IoMT architecture. *Computer Communications*, *160*, pp.139-149.

[25] Li, W., Yi, P., Wu, Y., Pan, L. and Li, J., 2014. A new IDsystem based on KNN classification algorithm in a wireless sensor network. *Journal of Electrical and Computer Engineering*, *2014*.