Utilizing deep learning and optimization methods to enhance the security of large datasets in cloud computing environments

S. Arvind1*, Pradeep Balasubramani2, D. Hemanand3, C. Ashokkumar4, Praseeda Ravuri5, M.N. Sharath6, and Kireet Muppavaram7

1Department of Computer Science Engineering, Hyderabad Institute of Technology and Management, India
2PS Consulting and Solutions
3Department of Computer Science and Engineering, S.A. Engineering College, Poonamallee-Avadi Road, Thiruverkadu, Chennai, Tamil Nadu, India, 600077
4Department of Computing Technologies, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Chennai, India, 603203
5Department of Computing Technologies, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Chennai, India, 603203
6Computer Science Engineer, Oregon State University, Corvallis, Oregon, USA 97331
7Department of CSE, Gitam Deemed to be University, Hyderabad, Rudararam, Sangareddy District, Telangana, India, 502329

Abstract. Many firms are outsourcing their information and computational needs because of the fast advancement of modern computing technology. Cloud-based computing systems must provide safeguards, including privacy, accessibility, and integrity, making a highly reliable platform crucial. Monitoring malware behavior throughout the whole characteristic spectrum significantly enhances security tactics compared to old methods. This research offers a novel method to improve the capacity of Cloud service suppliers to analyze users' behaviors. This research used a Particle Swarm Optimization-based Deep Learning Model (PSO-DLM) for the identification and optimization procedure. During the recognition procedure, the system transformed users' behaviors into an understandable format and identified dangerous behaviors using multi-layer neural networks. The analysis of the experimental data indicates that the suggested approach is favorable for use in security surveillance and identification of hostile activities.

1 Introduction to security in cloud computing

Cloud computing enables widespread services and access to computer resources from many vendors via a standardized interface [1]. The cloud meets users' requirements for reliable access to data and assets. Several firms have used cloud computing with capabilities including self-service, extensive network availability, pooling of resources, flexibility, and service measurement. These characteristics enable customers to focus on their business

* Corresponding author: scarvi@rediffmail.com
processes while also considering the computing power provided by the Cloud Service Provider (CSP) [2]. Cloud computing removes physical control and addresses privacy and information dependability issues by collaborating with CSP. Cloud computing is the virtualization of traditional data centers or servers. The cloud is a virtualized version of traditional data places, providing a wide range of applications and high efficiency for remote users. Managing large amounts of data poses a significant security concern. CSP security is analyzed from a storage perspective. Despite the potential benefits and earnings expected from the cloud approach, it still faced unresolved issues that impacted its credibility and widespread adoption. Cloud customers consider security a significant obstacle that prevents the adoption of the cloud paradigm.

Network security issues will surely arise in the smart grid sector due to the interconnection of communication and electricity infrastructure [3]. It accounts for the prevention and reduction strategy, which includes the random uncertainty associated with noise from communication at that time. The significant network attacks have caused substantial damage and impairment to the public's and the safety of life. Hence, studying anomalous network assault intrusion identification technology is a crucial and pressing matter. This research presents a way to maximize the hyperparameters of a Convolutional Neural Network (CNN) [4] architecture using the Particle Swarm Optimization (PSO) [5] technique, which has shown effective detection of various network attacks.

The following sections are ordered in the next order: Section 2 delves into the literature review on security concerns in cloud computing. Section 3 introduced the Particle Swarm Optimization-based Deep Learning Model (PSO-DLM) for identifying and optimizing procedures in the cloud. The simulation findings are examined and deliberated in section 4. The conclusion and conclusions are addressed in section 5.

2 Related works

Various methods have been developed to identify malicious activity in cloud platforms; however, they have yet to succeed in increasing accuracy. The cloud is vulnerable to a variety of harmful activities and security threats. These destructive actions result in link failure, incorrect decisions on forwarding, and redirection of pathways. Therefore, the indicated problem and the shortcomings of past approaches serve as incentives for creating a new method.

Thirimanne et al. introduced a Neural Network (NN) based Intrusion Detection System (IDS) that used a distributed architecture with an adaptive design to efficiently utilize cloud resources without overwhelming any one machine [6]. The suggested IDS utilizes Machine Learning (ML) capabilities from NNs to identify new threats. Testing the proposed IDS with the KDD dataset on a natural cloud-based system demonstrates its potential to identify assaults in the cloud environment. Kunhare et al. presented a novel IDS in a cloud environment that utilizes a Genetic Algorithm for choosing features and a Fuzzy Support Vector Machine (SVM) for categorization [7]. The strategy was evaluated using the KDDcup99 database and resulted in an improved detection rate of precision with a reduction in false alarms. Lata et al. created an IDS called hypervisor detectors to identify malicious behavior in a cloud-based setting at the hypervisor layer [8]. They determined that a fuzzy-based IDS tailored for target-based models would fail. They developed an adaptive technique that combines fuzzy systems with specific learning from NNs.

Dai et al. introduced an Online Sequential Extreme Learning Machine (OS-ELM) technique for intrusion detection [9]. Their technique utilizes alpha profiling to decrease temporal complexity by eliminating extraneous information via coherence and connections, reducing the state space. Beta profiling was employed rather than sampling to decrease the size of the training data set. This research examined the space and temporal difficulty,
demonstrating an improvement compared to other techniques. Otair et al. used gravity and geophysical search combined with Particle Swarm Optimisation (PSO) algorithms [10]. Lian et al. utilized an Artificial Bee Colony (ABC), a collaborative method, to identify and combat Denial of Service (DOS) assaults in a cloud setting [11]. After the bees were freed, this study extracted data from the system's traffic to train and test the classification algorithm. The detection rate of the ABC algorithm was greater than that of the algorithm. Zong et al. introduced a rapid extraction of features technique to enhance the detection of cloud irregularities [12]. An SVM had been employed for anomaly identification in this approach. The primary answer to improving the accuracy of SVM in a cloud setting was to decrease the volume of information.

3 Proposed particle swarm optimization-based deep learning model

The research provides a detailed explanation for each aspect in individual parts. Initially, the structure involves capturing sent packets by a node in the system and storing them in a database. The section will detail feature selection strategies for choosing appropriate characteristics for IDS. The selected data has to be translated into numerical values recognized by the NN. The following section involves normalizing data within the range of 0 to 1. Some of the normalized information is employed to train the multi-layer neural networks (ML-NN), while the rest is utilized for testing its performance. This data is chosen at random. The amount of information in the training and testing sets fluctuates based on the accuracy rate of their fluctuation in the tests. The PSO technique extracts the ideal weights and then trains the ML-NN employing the data for training. The categorization begins once the method reaches the predetermined number of iterations set at the beginning. Information is categorized into two categories: standard and aberrant. Information is classified as Category 2 if it shows an abnormality or incursion; alternatively, it is classified as Category 1. The research categorizes many types based on the nature of the app in question. Various assaults are analyzed in the evaluation, where multiple tests are presented. The network's management or user will get an indication or alert if a breach occurs. The architecture of the proposed PSO-DLM system is shown in Figure 1.

3.1 Datasets

The study used the KDDcup99 [13] and NSL-KDD databases [14]. The KDDcup99 database is employed for training and testing the system due to its advantageous characteristics in identifying certain behaviors throughout choosing features. The database contains 38 kinds of attacks and 42 characteristics. The following section will distinguish the techniques employed techniques and their characteristics. The NSL-KDD dataset is an enhanced iteration of the KDDcup99 dataset that removes redundant samples and introduces new types of assaults. This database has been utilized in current studies.
3.2 Feature Selection

Choosing features involves identifying and removing unnecessary and redundant information to the greatest extent feasible. It is a method of reducing dimensions that enables learning systems to operate more quickly and effectively. It enhances categorization. This work utilized correlation-based filtering to choose a subset of valuable attributes for IDS. The approach identified characteristics substantially linked with the category but not with each other as the most beneficial characteristics. This approach used a rapid redundancy reduction filter that relied on a modified correlation to choose characteristics by considering category labeling and comparing the characteristics. Twelve optimal characteristics for IDS were determined.

3.3 Data Preprocessing

Preprocessing involves two main activities: encoding the data and standardization. During data transformation (encoding), characteristics with undetectable values for an ML-NN must be transformed into characteristics with measurable values for the suggested solution. This process is referred to as encoding.
3.4 Training and Testing Datasets

K-fold cross-validation is a superior way of separating databases compared to random segregation for testing and training purposes. In this approach, K is set to 10, and the data is partitioned into ten equal segments. Every iteration involves selecting one segment as the tested information and the other segments as the data used for training. The research finally computes the average accuracy of the processes.

3.5 Extracting Optimal Bias and Weights

The research split the input information into two sets: testing and training. The training procedure begins for the data in the training set. The training procedure starts by initializing the ML-NN with various weights and then enhancing it using the PSO approach. Initially, values are assigned to the starting populace according to the number of factors in an ML-NN. The categorization error rate is computed. The method provides improved weights by advancing towards the optimum option. This occurs when the PSO method is iterated several times and reaches convergence. Once all iterations of the PSO method are completed, it will terminate. An exit circumstance arises when the best possible outcome represents the optimal overall scenario. The number of iterations is restricted and considered the exit criterion. The PSO process provides optimal values for weights and biases. If more iterations are required, the PSO procedure is restarted. The weight matrices are established, marking a crucial phase in optimizing the ML-NN. The method has been described as follows:

- Building an ML-NN by initializing PSO parameters with random values and establishing total biases and weights.
- Add weight vectors with values ranging from -1 to 1.
- Weights and bias are randomly initialized after training using various weights in the first phase.
- Assess the categorization error at each stage and adjust the biases and weights in each iteration of the method based on the categorization loss. Particles travel towards the best option at every stage, associated with a decreased rate of categorization mistakes.
- Updating the values of weights and biases involves substituting the weights from the previous step with the optimum weights.
- When the global best solution is reached, or the maximum number of iterations is reached, the ML-NN is trained using the ideal weights and then evaluated with test information. It displays the categorization accuracy %.
- The alarm is triggered in a practical setting upon detecting an assault.

3.6 Classification with Optimized ML-NN

Classification is currently being conducted using test and train data. The data is split into two categories. The suggested model's performance has to be assessed using appropriate biases and retrieved weights obtained during the training phase in the ML-NN setup. The network's results produce the Minimum Square Errors (MSEs) for training and testing data sets and the categorization and confusion matrix rates for each. This procedure is iterated over every category, and the average error of every category is then regarded as the overall error of the suggested approach.
Fig. 2 illustrates the beginning phase of setting the position variables in the PSO-DLM method. During the second phase, the fitness score of each component in the present iteration is determined by computing the cross-entropy losses function value during the first training session using ten position variables of all elements. Record the minimal fitness value for every aspect and the overall lowest fitness value, then compute the mean fitness value of the components. If the maximum number of repeats is not reached, go to the following step to complete the iteration process. If not, halt the iteration and display the minimal fitness value to conclude the procedure.

3.7 Attack or Intrusion Event

When an attack happens, alerts are shown to the user or network management in several formats. For instance, several types of notifications, such as messages on the display, error alerts, alarm sounds, and mobile or email notifications, might prompt relevant actions. For example, upon detecting a breach, every port is blocked, firewalls are reconfigured, or a honey pot is deployed. Operations suitable for the assault are chosen based on the circumstances in use. This article discusses sending attack-related requirements to a centralized server to determine the appropriate response strategy or using the data to train the ML-NN upon attack incidence and identification.

4 Simulation and outcomes

The 10-fold cross-validation approach is used to assess the suggested method. Specifically, ten specimens are randomly chosen from the dataset, each having a size of 250. One experiment is conducted for every assault and typical category. The number of activated neurons appears for the standard class, whereas the hidden layer for other categories has 15 neurons. Two hundred fifty specimens were utilized to test the ML-NN, whereas 2250 specimens were used for training. The system runs Windows 10 with 2 GB of RAM and an Intel Core CPU, and the program is performed using Python. The PSO method has enhanced the ML-NN's performance by optimizing the weights and biases.
Fig. 3. Accuracy results of the different optimization models

Fig. 3 shows the accuracy results of the different optimization models. The PSO-DLM algorithm obtained an accuracy of 96.45%, outperforming all other approaches in the research. The high precision level results from combining PSO with the DLM, which improves feature representation and efficiency. The Long Short-Term Memory (LSTM) model achieves an accuracy of 92.3%, demonstrating the efficacy of recurrent artificial neural networks in preserving temporal trends. CNN excels with an accuracy of 89.87%, showcasing its capacity to extract hierarchical information for behavior recognition. Traditional approaches like Decision Trees (DT) and Ant Colony Optimization (ACO) have correspondingly lower accuracies of 80.38% and 77.72%, respectively. The excellent results of PSO-DLM demonstrate the effectiveness of combining optimization methods with deep learning for robust security monitoring and threat detection in cloud computing settings.

Fig. 4. Precision results of the different optimization models

Fig. 4 shows the precision results of the different optimization models. PSO-DLM demonstrates exceptional accuracy of 97.12%, highlighting its remarkable capability in accurately categorizing threats. CNN achieved a high accuracy rate of 94.23%. DT and ACO have accuracy values of 87.17% and 78.11%, respectively. The remarkable accuracy of PSO-DLM highlights its efficacy in combining optimization methods with deep learning for accurate and dependable security threat detection in cloud computing settings.
Fig. 5. RMSE results of the different optimization models

Fig. 5 shows the RMSE results of the different optimization models. PSO-DLM has exceptional performance in regression analysis, with the lowest RMSE value of 3.45%. RF and LSTM provide common RMSE values of 7.44% and 7.37%, respectively, indicating their effectiveness in regression tasks. However, compared to other approaches, DT and ACO show higher RMSE values of 11.46% and 14.66%, respectively. The outstanding performance of PSO-DLM in reducing RMSE highlights its efficacy in combining optimization methods with deep learning for accurate and dependable regression tasks in cloud computing settings.

Fig. 6. Execution time results of the different optimization models

The execution time results of the different optimization models are shown in Fig. 6. PSO-DLM shows exceptional efficiency in execution time, with the fastest time recorded at 8.56 seconds, highlighting its computational speed. DT and RF have execution durations of 12.27 and 19.76 seconds, respectively. SVM, k-NN, and ACO have longer execution durations, ranging from 28.6 to 39.65 seconds. CNN, RNN, and LSTM need a longer duration, varying between 56.68 and 75.42 seconds. The quick performance of PSO-DLM highlights its usefulness for real-time security monitoring and threat detection in cloud computing settings.
5 Conclusion and findings

This study introduces a new PSO-DLM method that effectively detects various types of assaults by zombie hosts afflicted with infections. Keras assigns the hyperparameters of every layer's architecture as the location parameters of particulates, and the cross-entropy loss function's value during the first training phase of DLM is used as the fitness value for PSO-DLM. PSO iteratively updates the velocity and position of elements to look for decreasing fitness values. The momentum weight factor is adjusted based on the fitness score to prevent the PSO algorithm from becoming stuck in the local extremum and to find the appropriate DLM structural characteristics. Finally, the efficacy of the suggested PSO-DLM detection method is shown via a comparison with three widely used detection techniques. The PSO-DLM approach offers excellent performance with an accuracy of 96.45%, precision of 97.12%, RMSE of 3.45%, and an efficient execution time of 8.56 seconds, indicating its efficacy in security monitoring in cloud computing.

References