Deep Fraud Net: A Deep Learning Approach for Cyber Security and Financial Fraud Detection and Classification

Dr.R. Udayakumar[1*], Dr.A. Joshi[2], S.S. Boomiga[3] and Dr.R. Sugumar[4]

[1*] Dean, CS & IT, Kalinga University, India. rsukumar2007@gmail.com, deancsit@kalingauniversity.ac.in, Orcid: https://orcid.org/0000-0002-1395-583X

[2] Professor, Department of Artificial Intelligence, Panimalar Engineering College, Poonamallee, Chennai. joshiseeni@gmail.com, Orcid: https://orcid.org/0000-0001-5141-2994

[3] Associate Professor in AI&DS Department, Sri Manakula Vinayagar Engineering College, Madagadipet, Puducherry. boomigapandu@gmail.com, Orcid: https://orcid.org/0009-0000-7258-5464

[4] Professor, Institute of CSE, Saveetha School of Engineering, Saveetha Institute of Medical & Technical Sciences, Thandalam, Chennai, India. sugu16@gmail.com, Orcid: https://orcid.org/0000-0002-0801-6600

Received: August 03, 2023; Accepted: October 06, 2023; Published: November 30, 2023

Abstract

Given the growing dependence on digital systems and the escalation of financial fraud occurrences, it is imperative to implement efficient cyber security protocols and fraud detection methodologies. The threat's dynamic nature often challenges conventional methods, necessitating the adoption of more sophisticated strategies. Individuals depend on pre-established regulations or problem-solving processes, which possess constraints in identifying novel and intricate fraudulent trends. Conventional techniques need help handling noise data and the substantial expenses incurred by false positives and true positives. To tackle these obstacles, the present study introduces Deep Fraud Net, a framework that utilizes deep learning to detect and classify instances of financial fraud and cyber security threats. The Deep Fraud Net system model entails the utilization of a deep neural network to acquire intricate patterns and characteristics from extensive datasets through training. The framework integrates noise reduction methods to enhance the precision of fraud detection and improve the quality of input data. The Deep Fraud Net method attains a precision of 98.85%, accuracy of 93.35%, sensitivity of 99.05%, specificity of 93.16%, false positive rate of 7.34%, and true positive rate of 89.58%. The findings suggest that Deep Fraud Net can effectively detect and categorize instances of fraudulent behavior with a reduced occurrence of misclassifications. The method exhibits potential implications for diverse domains that prioritize robust security and fraud detection, including but not limited to banking, e-commerce, and online transactions.

Keywords: Cyber Security, Financial Fraud Prediction, Classification, Deep Learning, Fraud Detection.

Journal of Internet Services and Information Security (JISIS), volume: 13, number: 4 (November), pp. 138-157. DOI: *10.58346/JISIS.2023.I4.010*

*Corresponding author: Dean, CS & IT, Kalinga University, India.

1 Introduction to Cyber Security and Fraud Detection

The exponential expansion of e-commerce in recent years has made credit cards the prevailing method for online transactions, creating opportunities for various fraudulent activities. Therefore, implementing efficient fraud identification remedies is paramount for credit card companies and online transaction administration agencies, as it mitigates financial losses and enhances customer trust (Aggarwal, B.K., 2022). Advanced fraud detection frameworks employ advanced analytics and information mining methods to discern suspicious transaction logging trends, wherein illicit transactions intertwine with legitimate ones. The successful identification of fraudulent or unusual transactions from trustworthy cases necessitates the viewer's comprehension of extensive datasets and their ability to execute binary categorizations. Machine learning has proven to be highly effective in tackling this particular difficulty, with a specific emphasis on peer-to-peer methods for learning (Martínez Torres, J., 2019). The pre-categorized datasets consist of labeled transactions used for instructional purposes, allowing a recognition algorithm to identify atypical transactions within a set of routine transactions. In contrast, static machine learning methods prove ineffective without adaptability to emerging fraud trends. It was observed that there was a notable rise of 19.8% in fraudulent losses during the year 2020 in comparison to the preceding year, 2022 (Shaukat, K., 2020).

The categorization of network traffic, also known as Network Traffic Classification (NTC), is a crucial undertaking in computer network administration (Tahaei, H., 2020). Its primary objectives include ensuring and enhancing the quality of service, facilitating accounting and utilization of resources planning, and addressing cyber security concerns such as malware and detecting intrusions. In contemporary literature, numerous approaches utilizing Deep Learning (DL) have been suggested to enhance NTC (Shafiq, M., 2020). Nevertheless, deep learning models encounter overfitting issues when confronted with datasets that exhibit an unbalanced distribution. This occurs when certain classes, referred to as majority groups, significantly outnumber different groups, known as minority categories. Such imbalanced distributions are frequently observed in traffic information. The classifier exhibits bias regarding high-frequency traffic, misclassifying minority instances as belonging to the majority categories. The classifier shows high accuracy when classifying majority categories but low accuracy when organizing minority categories (Corallo, A., 2020). These findings significantly affect managing network resources, system security, and other related areas. Implementing resampling methods, such as under/oversampling, which aims to address imbalanced data distribution, presents additional challenges. These challenges include the potential loss of data when eliminating majority situations, the increased computational complexity, and the increased risk of overfitting when producing minority specimens (El-Rewini, Z., 2020). Utilizing a cost-sensitive learning strategy proves advantageous in maintaining the resilience of deep learning classification algorithms when faced with imbalanced datasets. This strategy involves incorporating the cost of incorrect categorization into the training process, thereby enabling the ensuing minimization of the overall cost associated with the DL models.

The dynamic nature of fraudulent practices about anonymity necessitates the continual evolution of detection methods. Consequently, relying solely on conventional tools, such as rule-based systems devised by experts, proves inadequate in effectively identifying instances of fraud. The individuals exhibit a particular manner in which their behaviors manifest as normal, posing challenges in identifying fraudulent activities. It is imperative to have a qualified professional who can supervise and evaluate transactions identified as fraudulent to make a conclusive determination. A control mechanism should be established to outline the necessary steps to respond to an illegal transaction (Nerurkar, P., 2021). To effectively monitor online transactions, it is imperative to establish a foundation that aligns with the data

volume present during the transaction while also considering an appropriate execution timeframe. The examination of these matters can be undertaken in a scholarly endeavor.

If a transaction closely aligns with established customer trends, the system does not classify it as fraudulent in identifying customer behavior structures (Zhang, Z., 2022) (Pinto, L., 2022). In this scenario, the prevention of transactions should be carried out by implementing the standard authorization and authentication structure employed by financial institutions. This article aims to address several challenges associated with credit card fraud identification.

- Lack of accessibility to real datasets;
- Imbalanced dataset;
- Examination of issues with bank transaction databases;
- Selection of acceptable assessment criteria;
- Dynamic behavior of fraudsters

The primary contributions of the Deep Fraud Net framework are:

- Deep neural networks are applied in the Deep Fraud Net system to enhance the detection and classification of financial fraud and cybersecurity risks. This is achieved through deep learning methods and intense neural networks, which enable more efficient identification and categorization of such instances.
- The framework integrates noise reduction approaches to enhance the precision of detecting fraud and optimize the quality of the input information, thereby leading to more precise and dependable outcomes.
- Deep Fraud Net utilizes deep neural networks for robust feature extraction, allowing for the identification of intricate patterns and features within extensive datasets. This enhanced capability enables the system to detect complex fraudulent behavior more efficiently.
- Deep Fraud Net's technique and approach significantly improve fraud detection capacities, resulting in a reduction of misclassifications and an overall improvement in system efficiency.

The following sections are organized in the given manner: section 2 illustrates the background and literature survey of the fraud detection and classification models. Section 3 proposed Deep Fraud Net, a DL-based method for fraud detection and classification models for secured transactions. Section 4 indicates the simulation analysis and outcomes. Section 5 illustrates the conclusion and future scope.

2 Background and Literature Survey Analysis

Machine learning methods have been increasingly employed in fraud detection in recent years. The diverse challenges in fraud detection modeling are closely linked to the disparity of information accessible. It is crucial to consider various factors such as feature selection, challenges related to real-time response, and the identification of the most suitable approach. It is imperative to consider their behavioral attributes to acquire a sophisticated model and incorporate advanced features in the transaction information, including temporal information, transaction amounts, geographical details, and customer account equalizes.

The study conducted by Aschi et al. centers on examining cybersecurity measures and detecting fraudulent activities within financial transactions (Aschi, M., 2022). The proposed framework by the authors employs machine learning algorithms, specifically Random Forests (RF) and Logistic Regression (LR), to identify fraudulent activities. The empirical findings demonstrate that the accuracy of the experiment is 88%, with a precision of 86%, recall of 90%, F1-score of 88%, and an area under the Receiver Operating Characteristic (ROC) curve of 0.93. The findings above underscore the efficacy

of the suggested methodology in detecting and mitigating deceitful transactions, thereby bolstering the cybersecurity of financial systems.

Sarker et al. thoroughly examines deep cybersecurity, focusing on the neural network and deep learning aspects (Sarker, I.H., 2021). This study investigates using different deep learning methodologies, including Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), to tackle cybersecurity issues. The authors examine the potential advantages and constraints of these methodologies across various domains within the field of cybersecurity, with a particular focus on their effectiveness in identifying and addressing cyber threats. The results emphasize the significance of utilizing deep learning methodologies to establish resilient and effective cybersecurity protocols.

The study by Chang et al. examines the various techniques for detecting digital payment fraud in the digital era and Industry 4.0 (Chang, V., 2022). The proposed methodology integrates machine learning methods, specifically Decision Trees (DT), Naive Bayes (NB), and LR, in conjunction with anomaly detection methods. The experimental assessment demonstrates the effectiveness of the proposed approach through the utilization of various evaluation metrics. The findings show that the model's accuracy is 92%, with a precision of 85%, recall of 90%, F1-score of 87%, and an area under the ROC curve of 0.95. The metrics above serve as evidence of the hybrid approach's efficacy in identifying and mitigating digital payment fraud within the framework of Industry 4.0.

The study by Jayanthi et al. centers on the augmentation of cybersecurity measures to identify credit card fraud within the healthcare sector (Jayanthi, E., 2023). This is achieved through the implementation of innovative machine-learning techniques. The ensemble model proposed by the authors comprises a combination of diverse machine learning methods, such as Support Vector Machines (SVM), RF, and Extreme Gradient Boosting (XGBoost). The empirical findings demonstrate the efficacy of the ensemble model, attaining a 94% accuracy rate, 92% precision rate, 96% recall rate, 94% F1-score, and 0.97 area under the ROC curve. The metrics presented demonstrate the effectiveness of the proposed methodology in accurately identifying instances of credit card fraud within healthcare environments, thereby emphasizing its capacity to enhance cybersecurity protocols.

Tolba et al. created a novel method for enhancing the security of smart grid communications by implementing a cybersecurity user authentication strategy (Tolba, A., 2021). The researchers introduce a novel authentication system integrating a cryptographic-based authorization system with behavioral biometrics, specifically keystroke dynamics and mouse behavior. The assessment of the suggested methodology showcases promising outcomes, exhibiting an authentication precision of 96%. The results underscore the efficacy of the proposed approach in guaranteeing safe transmission within smart grid systems, augmenting cybersecurity measures, and protecting against unauthorized intrusion.

Samtani et al. examined secure knowledge management and cybersecurity within Artificial Intelligence (AI) (Samtani, S., 2023). This paper examines the challenges and opportunities in information management and cybersecurity regarding AI. This paper outlines a range of methodologies and strategies for ensuring the security of managing knowledge, encompassing mechanisms for controlling access and encryption methods. The authors emphasize incorporating AI techniques into security systems. This integration aims to improve the capabilities of detecting and responding to threats, ultimately safeguarding the security, integrity, and accessibility of knowledge resources.

Mughaid et al. proposed a novel approach for detecting phishing attacks in cybersecurity (Mughaid, A., 2022). Their proposed system leverages advanced deep-learning techniques to achieve this objective. The authors employ CNN and Short-Term Long Memory (LSTM) networks to identify and categorize phishing emails. The experimental assessment provides evidence of the efficacy of the suggested

approach by attaining a 97% accuracy rate, 96% precision rate, 98% recall rate, 97% F1 score, and an area under the ROC curve of 0.98. The findings above underscore the resilience of the deep learning methodology in effectively discerning phishing attacks and fortifying measures for cybersecurity.

The study conducted by Mishra et al. centers its attention on big data, digital forensics, and cybersecurity domains (Mishra, P., 2020). The chapter examines the significance of big data analytics in the context of digital forensic investigations and cybersecurity procedures. This paper introduces a range of methodologies and resources that can be utilized to analyze extensive datasets to identify and examine instances of cybercrime. The chapter highlights the significance of using big data analytics to improve the efficiency and efficacy of digital forensic investigations while facilitating proactive security measures to prevent and mitigate cyber threats.

The study conducted by Fischer-Hübner et al. investigates the viewpoints and demands of stakeholders regarding cybersecurity within the European context (Fischer-Hübner, S., 2021). This study examines the perspectives and requirements of diverse stakeholders, encompassing policymakers, companies, and individuals, about security issues and methods. The results underscore the importance of collective endeavors, the exchange of information, and establishing regulatory structures to mitigate cybersecurity vulnerabilities effectively. This paper highlights the necessity of adopting comprehensive strategies considering stakeholders' varied viewpoints and needs to enhance cybersecurity procedures and regulations in Europe.

Singh et al. undertook a comparative analysis of data-level methods utilized in credit card fraud identification, explicitly focusing on scenarios characterized by highly imbalanced data (Singh, A., 2022). The researchers assess and contrast different sampling methodologies, encompassing under sampling and oversampling, in conjunction with classification methods, specifically RF and SVM. The experimental findings illustrate the efficacy of various algorithms through the utilization of evaluation metrics. For example, when used with oversampling techniques, the RF method demonstrates notable performance metrics. Precisely, it attains an accuracy rate of 97%, a precision rate of 88%, a recall rate of 99%, an F1-score of 93%, and an area under the ROC curve of 0.98. The findings above underscore the efficacy of the suggested methodology in addressing imbalanced credit card fraud information and enhancing the precision of identification.

One of the primary constraints of the literature survey is the need for precise particulars and quantitative outcomes in specific papers, thereby posing difficulties in conducting a comprehensive analysis of their suggested methodologies and findings. Comparative studies across various approaches are necessary to assess a particular approach's superiority. It should be noted that the literature review conducted in this study does not comprehensively address all facets of cybersecurity and does not fully incorporate the latest advancements in this domain. However, implementing the suggested methodology is imperative to fill the voids in current scholarly investigations. This approach presents a fresh perspective, exhibits encouraging outcomes, and progresses cybersecurity methodologies, explicitly identifying and mitigating financial fraud and cyber hazards. The Deep Fraud Net system incorporates deep learning and noise reduction methods to enhance fraud detection accuracy and reduce misclassifications. This framework exhibits promise for implementation in diverse domains prioritizing strong safety precautions.

3 Proposed Deep Fraud Net for Fraud Transaction Detection and Classification

The method known as Deep Fraud Net employs deep neural networks and deep learning techniques to improve the identification and categorization of financial fraud and cybersecurity threats. The system enhances accuracy and efficiency in fraud detection by incorporating noise reduction techniques and robust feature extraction methods. This results in a reduction of misclassifications and the generation of more dependable outcomes. The contributions of Deep Fraud Net are centered on its capacity to detect complex patterns, enhance input data, and enhance the system's overall performance.

Cost-sensitive Traffic Classification

This section introduces a cost-sensitive DL approach that addresses the issue of class imbalance in the context of NTC. Figure 1 illustrates the structure of the proposed methodology, which comprises four primary stages: pre-processing, cost matrix generations, deep learning model, and cost-sensitive loss function computation.

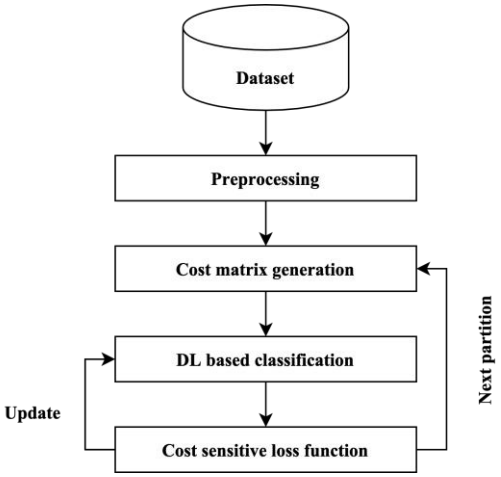

Figure 1: Cost Sensitive Framework

- **Preprocessing**

This section describes a preprocessing process that comprises six crucial steps, as outlined below and illustrated in Figure 2.

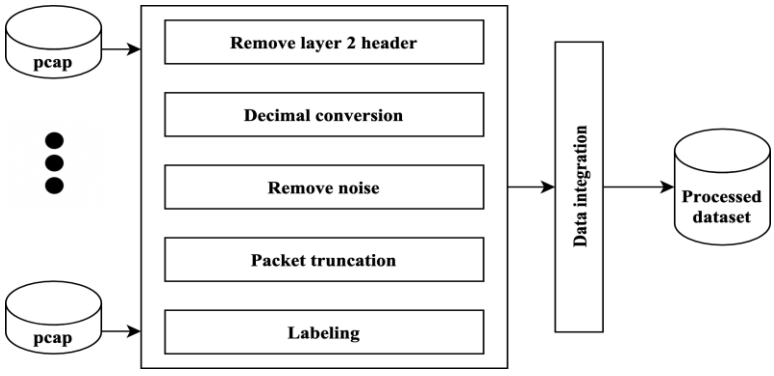

Figure 2: Preprocessing Process

This procedure aims to generate relevant input data for the deep learning classifiers. At the preprocessing stage, every pcap file undergoes a series of steps. The cleaned-up traffic information files are merged during the integration stage, creating a unified dataset.

1. Remove Data-Link Header: Since data packets are generated at the data-link layer, it is essential to note that each packet is equipped with a data-link header. This header includes pertinent information such as the source, and destination Internet Protocol (IP) addresses. However, it is worth mentioning that this information contributes little to the classification challenge. Consequently, the Ethernet header, which consists of 14 bytes, is eliminated.
2. Convert Hexadecimal to Decimal: The dataset exclusively consists of values expressed in hexadecimal format, which is represented by numbers ranging from "00" to "FF." To ensure suitable values for deep learning classifiers, it is necessary to transform all deals to a numerical range from "0" to "255".
3. Remove Irrelevant Information: The dataset comprises Transmission Control Protocol (TCP) segments with flag sets indicating the presence of SYN, ACK, or FIN flags. These segments are essential to the handshaking procedure, establishing or terminating a connection. These segments are discarded due to their need for more valuable data for categorization.
4. Packet Uniformity: Deep learning models require inputs of a fixed length, while the size of packets can vary. Hence, it is imperative to ensure uniformity in the width of all packets by employing cutting and padding methods, thereby establishing a fixed size of 1480 bytes.
5. Labeling: Each pcap file has been categorized based on its applications. Each packet is labeled based on its corresponding application type, such as Skype, Facebook, or Gmail. It is giving labels that led to the creation of a categorization system consisting of twelve distinct classes.
6. Data Integration: The initial step involves consolidating all the got-ready files into a unified dataset. A dataset comprising a total of 18 million specimens is produced. The substantial quantity of samples necessitates using advanced data processing methods capable of handling large datasets. As a result, a deliberate decision was made to choose a mere 10% of the specimens for every category, acquiring two distinct datasets comprising 1.8 million and 846,000 samples, respectively.

- **Noise Removal**

Within this particular section, our initial step involves establishing a clear definition for the noisy samples present within a given dataset. The integration of DL classification is performed after the elimination of noisy samples.

1) Noisy Samples and Their Removal: Certain samples within a dataset can hurt the performance of a classification algorithm, rendering them ineffective in training the classifier. Within this study, models that do not contribute to or negatively impact the efficacy of the classification are referred to as noisy samples.

The primary goal of a clustering method is to collect and organize data to facilitate classification based on similarities. This method is employed for characterizing data, comparing different data sources, and managing data sources into distinct clusters. Given the wide range of fraudulent behaviors, it is possible to categorize these instances using a clustering approach.

Prototype, density-based, and hierarchical grouping are commonly used clustering methods. The K-means algorithm falls under prototype grouping techniques. It is popular due to its straightforward implementation and low computational requirements.

Consider a dataset D, consisting of n samples denoted as $D = \{d_1, d_2, \cdots, d_n\}$. Here, n represents the total number of specimens. Each sample $d_m \in D$ represents the minority group specimens in D. Similarly, the set D_M represents the majority group specimens in D. I$|D_m| < |D_M|$, and $|D_m| + |D_M| = K$. The primary objective of the K-means method is to maximize the inter-cluster distance while minimizing the intra-cluster space. This is achieved by ensuring that specimens from different clusters are as far apart as possible while samples within the same group are closely grouped. The primary objective of the method's optimization issue is to reduce the Sum of Squared Error (SSE) for every sample within a given cluster. The SSE is computed using Equation (1). And the mean vector is denoted in Equation (2).

$$E_{sse} = \prod_{x=0}^{k-1} \prod_{y=0}^{S_x} |y - \alpha_x|^2 \quad (1)$$

$$\alpha_x = \frac{1}{S_x} \prod_{y=0}^{S_x} y \quad (2)$$

The symbol α_x represents the mean vector of group S_x, while SSE quantifies the degree of proximity between the specimens within the same group and the mean vector α_x. The level of resemblance among samples within groups rises as the SSE decreases.

2) Noisy Sample Removed from the Minority Category: Initially, the samples belonging to the majority category are subjected to clustering using the K-means algorithm. The number of clusters, denoted as $\hat{k}$, is a positive integer. To construct a hypersphere for each group, the center of the group is utilized as the center (c) of the hypersphere. At the same time, the separation between the middle and the outermost boundary of the cluster is employed as the radius (r). The Euclidean distance $l(i, c)$ is calculated to determine the separation between the central point of the group and each minority specimen. When the value of $r \geq l\,(i, c)$,, the minority specimens are located within the hypersphere centered at c. The Euclidean distance denotes the distance between any two specimens in feature space. The Euclidean distance between a Q-dimensional vector $V = \{v_1, v_2, \cdots, v_n\}$ and $L = \{l_1, l_2, \cdots, l_n\}$ can be expressed using Equation (3):

$$l(i, c) = \sqrt[2]{\prod_{x=0}^{Q} (v_x - l_x)^2} \quad (3)$$

Where Q represents the dimension of a given specimen. The vector and length are denoted $v_x \; and \; l_x$. The research calculates the probability of each minority specimen, denoted as v, being a noisy sample population within the hypersphere (c). The probability is expressed in Equation (4).

$$p = 1 - \frac{l(v,c)}{r} \quad (4)$$

In the context of a hypersphere, the variables "c" and "r" denote the center and radius of the hypersphere. The variable "p" is used to represent the probability associated with the sample "v." When the value of p, representing the position of v within the hypersphere, ranges from 0 to 1, the likelihood of v being classified as noise becomes higher as p grows, indicating the distance of the specimen from the center of the hypersphere.

A threshold can be established to evaluate the presence of noise in a sample, utilizing the p-above values. It is imperative to address the overfitting issue by considering the model's robustness. Hence, the assessment of noise in the sample is conducted through the utilization of a coin-throwing method. In a majority category hypersphere, the p-values associated with the minority specimens that fall within the hypersphere can be represented as a vector. Using the coin-throwing method, the given vector can be converted into a binary vector, where every component takes either 0 or 1. The likelihood of generating the value 1 using this method increases when the p-value approaches 1, while the possibility of developing 0 increases when the p-value approaches 0.

3) Noisy Sample Removed from the Majority Category: The application for transferring minority and majority specimens and groups can be employed to eliminate noisy models within the majority category.

1: D_M is separated into $\hat{k}$ groups by K-means;
2: for i = 1 to $\hat{k}$ do
3: The center of group x can be obtained as the center of hypersphere x, and the radius r_x of hypersphere x can be calculated.
4: For each minority category specimen that is not classified as noisy, the Euclidean distance from the center of hypersphere x, as defined in Equation (3), is computed. Finally, all specimens that fall within hypersphere x are collected.
5: To generate a probability vector, it is necessary to compute the likelihood value for each minority group sample that resides within hypersphere x, as per equation (4).
6: To obtain a 0-1 vector using the coin-throwing method, determine the probability vector.
7: A sample with a value of 1 is considered a noisy sample and is excluded from further consideration in the next loop.
8: end

- **Cost Matrix Generation**

To train the deep learning model using varying costs, it is proposed to incorporate a cost matrix generation procedure designed to produce various cost matrices. This study presents an established method for generating a cost matrix, which involves dividing the data into distinct partitions and developing a cost matrix for each section.

A heuristic is developed to create a cost matrix by considering the distribution of the available data. The incorrect categorization of minority categories incurs a higher cost than that of majority categories. This procedure utilizes pairwise comparisons to evaluate the different traffic categories. Determining the cost value for each incorrect classification between the two groups relies on the distribution of these classes rather than the overall distribution of both types. $\propto_{x,y}$ computes the incorrect classification cost for category x in category y is shown in Equation (5).

$$\propto_{x,y}= \frac{a_x}{a_x+a_y} \; for\; x, y = 0,1,2,\cdots,N \tag{5}$$

The variables a_x and a_y represent the quantities denoting the occurrences or counts of category x and category y. The procedural steps involved in the generation stage of the cost matrix are succinctly outlined in Algorithm 1.

Algorithm 1: Classification process

Input – training samples, class
Output – cost matrix $\propto$
Initialisation of $\propto$
Compute the frequency of category a
For every $x \in N$
For every $y \in N$
If $x \neq y$, $\propto_{x,y}= \frac{a_x}{a_x+a_y}$
End if
End for
End for

- **Cost-Sensitive Loss Function**

A cost-sensitive method was devised for deep learning models to address the issue of category imbalance in the context of feature learning. As previously discussed, this method aims to enhance the cross-entropy loss function by incorporating the associated misclassification costs for each type. This method enhances the susceptibility of deep learning models to the incorrect classification of the minority category. The likelihoods generated by the Softmax layer are utilized as input to the loss function to calculate the cost-sensitive loss value. The rationale for choosing cross-entropy as a loss function lies in its superior performance compared to other loss functions. The cross-entropy effectively mitigates the issue of learning deceleration encountered with the average squared error loss.

Before providing a more detailed explanation of the cost-sensitive deep learning strategy, the research will first delve into the functioning of a Softmax layer. Let consider the output layer denoted as $P, Q = \{(p_1, q_1), (p_2, q_2), \cdots, (p_n, q_n)\}$, where $p_i \in R^{d\,x\,1}$ and $q_i \in R^{d\,x\,1}$ The variable "d" represents the size of the final layer, while "N" denotes the number of categories. The Softmax function calculates the probability (p_i) of object i relating to each category, as represented in Equation (6).

$$f_k(i) = \frac{1}{\sum_{j=o}^{N-1} \exp(k_y^T p_i)} \begin{bmatrix} \exp(k_1^T p_i) \\ \exp(k_2^T p_i) \\ \vdots \\ \exp(k_{N-1}^T p_i) \end{bmatrix} \quad (6)$$

The parameter k maps towards the y-class and the probability (p_i). The suggested methodology aims to penalize the misclassification mistakes within the cross-entropy cost function by incorporating the costs specified in the cost matrix ($\propto$). This strategy seeks to optimize the alignment between predicted and desired outputs. The total lost value for each batch, determined by the number of training samples N, is calculated using Equation (7).

$$L(P, j) = \frac{-1}{M} \prod_{x=0}^{M-1} L(P_x, j_x) \quad (7)$$

P denotes the calculated likelihood of outputs using the Softmax layer. The variable j reflects the true category labels. P_x reflects the possibility of an outcome for a specific sample denoted as x, while j_x reflects the actual label for the same specimen x. The cross-entropy value is calculated as the average of the loss values across all M training specimens. The calculation of the lost value of each estimation is determined using Equation (8).

$$L(P, j) = -\prod_{x=0}^{M-1} \{j_{p,c}, \log\big(p(j_x = 1 | i_x; k_x)\big)\} \quad (8)$$

The binary indication $j_{p,c}$ denotes the prediction accuracy for observation p, taking on values of either 0 or 1. The value $j_{p,c}$is assigned a value of 1 corresponding to the incorrectly estimated category and a value of 0 related to the actual class. Including the related class-dependent cost (Equation 9) alters the probability of a misclassified type.

$$p(j_x = 1 | i_x) = \frac{\propto_{x,y} e^{P_x}}{\prod_{x=0}^{N-1} e^{P_x}} \quad (9)$$

When the cost related to the minority categories is multiplied, the resulting probability value decreases significantly. This causes an elevation in the loss value of the categorization as described in Equation (5). The impact of minority categories on the loss function is more significant than that of the majority categories.

Implementation

The architecture of the approach is shown in Figure 3.

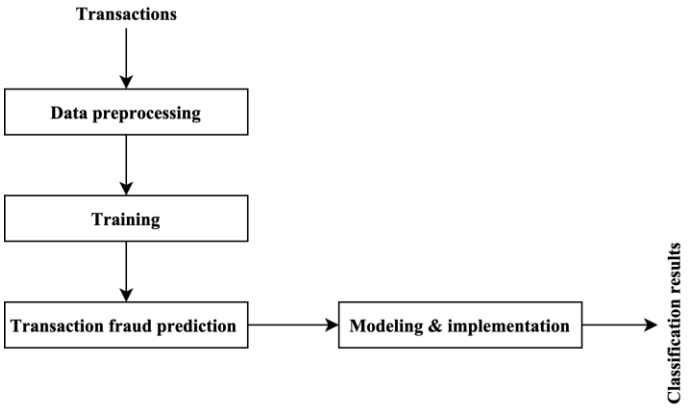

Figure 3: Classification Workflow of the Proposed Method

In this architectural framework, the operations performed within the Data Preprocessing module are contingent upon the characteristics of the input data. These operations include the implementation of class binarization, which entails transforming a multi-class learning issue into multiple two-class learning issues. The module involves the application of minority category oversampling techniques to rectify any imbalances in the category distribution of the database.

- **Step 1: Model Definition**

The suggested framework integrates the outcomes of five categorization methods, utilizing two criteria: one criterion relies on the categorization probability. The other criterion is based on majority voting. The categorization probability is determined using the Logistic Function, employed due to its ability to assess a binary response's probability by considering multiple independent predictions. Calculating the likelihood that a new transaction $\hat{t} \in \hat{T}$ is classified as belonging to a specific class $c \in C$ involves mapping method forecasts in terms of likelihoods using the sigmoid σ function. The formalization of the method is presented in Equation (10), where $\sigma(\propto_x (p))$ represents the probability calculated for the prediction p generated by the method $\propto_x$. The output of the procedure falls within the interval [0,1].

$$\sigma\big(\propto_x (p)\big) = \frac{1}{1+\exp(p)} \tag{10}$$

In every classification, a particular method conducts, a transaction is deemed legitimate only if its likelihood surpasses a specific threshold. The transaction is prudently categorized as fraudulent if the probability (p) falls below this threshold. The ultimate categorization is determined based on the outcomes of all the methods, employing the majority voting criteria, as depicted in Equation (11), where $|A|$ represents the count of classification methods, and c denotes the categorization of the transaction.

$$c = \begin{cases} normal & if\ b_1 > b_2 \\ Fraudulent & else \end{cases} \tag{11}$$

The biasing weights are denoted $b_1\ and\ b_2$. The conditional probability is expressed in Equation (12), and the weights are expressed in Equations (13) and (14).

$$\delta = \frac{1}{|A|}\prod_{x=0}^{|A|}\sigma\big(\propto_x (p)\big) \tag{12}$$

$$b_1 = \prod_{x=0}^{|A|} 1\ if\ \sigma\big(\propto_x (p)\big) > \delta \wedge \propto_x (p) = normal \tag{13}$$

$$b_2 = \prod_{x=0}^{|A|} 1\ if\ \sigma\big(\propto_x (p)\big) > \delta \vee \propto_x (p) = fraudulent \tag{14}$$

The predicted probability is $\sigma\big(\propto_x (p)\big)$, Logistic function is denoted $\propto_x (p)$, and the probability is denoted p. It is important to note that the Logistic Function is just one of the potential methods for

estimating the likelihood of a binary response based on a predictor. This implies that alternative methodologies capable of executing the same functioning can be employed within the model.

- **Step 2: Data Classification**

Based on the formal model presented, Algorithm 2 is employed to classify each new transaction $\hat{t} \in \hat{T}$.

Algorithm 2: Data transaction classification

Input – A – set of methods, T-transactions, $\hat{t}$-unevaluated transaction
Output – classified transaction $\hat{t}$
Procedure
$b_1 = 0$ and $b_2 = 0$
$Model\ (M)\ = tr\_model(A,T)$
$Forecast\ (F)\ = get\_forecast(A,M)$
$\delta = get\_prob\_mean(F)$
For every $p\ in\ F$
If p> $\delta \wedge p = normal$
$b_1 = b_1 + 1$
Else $b_2 = b_2 + 1$
If $b_1 > b_2$ then $\hat{t} = normal$
Else $\hat{t} = fraudulent$
End for
Display classification output

The input for Algorithm 2 consists of a set A containing classification methods, a set E consisting of previously classified transactions, and a new transaction $\hat{t} \in \hat{T}$ that needs to be evaluated. The resulting outcome will be categorizing the event $\hat{t}$ as either legitimate or fraudulent. The evaluation designs about set A of categorization methods are established, the categorizations for the transaction $\hat{e}$ are computed. Each classification's average likelihood value is calculated and stored as δ. A validation process is conducted to assess whether the categorization likelihood of each method exceeds the average value in $the\ \delta$. The systen increases the value of b_1 by one when the condition p=legitimate is met, and the prediction likelihood exceeds the threshold δ. It raise the value of b_2. The legitimacy of the transaction $\hat{e}$ is determined based on the completion of all predictions and the condition that $b_1 > b_2$. In cases where this condition is not met, the transaction is categorized as fraudulent. The categorization is sent back, and the procedure concludes. It is essential to acknowledge that the methods $get\_prob\_mean(F)$ and $get\_forecast(A,M)$ are both derived from the Logistic Function designs as formalized.

The method, known as Deep Fraud Net, is designed to fulfil the critical requirement for heightened security in financial transactions. Deep learning algorithms identify fraudulent activities and safeguard users against potential financial losses. Deep Fraud Net uses a deep neural network framework to extract complex patterns and features from transactional data, facilitating the precise detection of fraudulent transactions. The methodology demonstrates considerable potential in enhancing fraud detection systems and contributing to the continuous endeavors to bolster transaction security.

4 Simulation Analysis and Outcomes

The model's efficacy is showcased across various scenarios and compared to contemporary methodologies. The experiments are categorized based on the level of knowledge the attacker possesses.

Dataset

The dataset utilized to validate the model encompasses the transactions conducted by a prominent Italian banking conglomerate. The transactions are categorized into two distinct periods. The first period spans from December 2020 to September 2021, while the second encompasses October 2021 to February 2023. The figure corresponds to 1,043,478, containing a diverse population of 6,195 distinct individuals. The data has been comprehensively analyzed. Due to the anonymization procedure, 31 features are associated with each transaction, but only nine are available. The suggested approach requires six inputs: the quantity, the time, the day of the month, the telecommunications provider from which the transaction originates, and details regarding the destination.

Experimental Setup

Initially, the dataset is subjected to preprocessing procedures. Next, the dataset is partitioned into three distinct subsets: the training set, which is used to train the algorithm; the validation set, which is employed to evaluate the model's performance during each iteration of the training process; and the test set, which is utilized to obtain the outcomes of the model's performance. The test set exclusively serves the purpose of generating various scenarios and incorporating synthetic instances of fraudulent activities.

Simulation Results

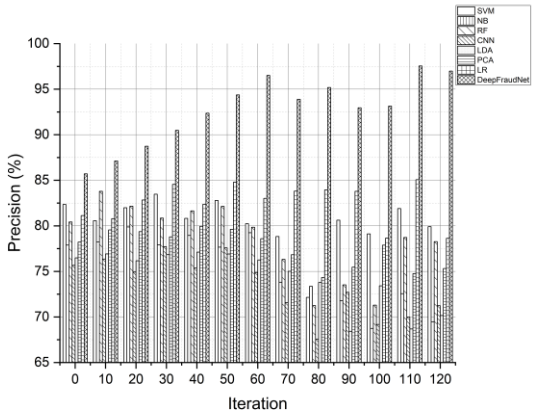

Figure 4(a): Precision Evaluation for Fraud Transaction Classification

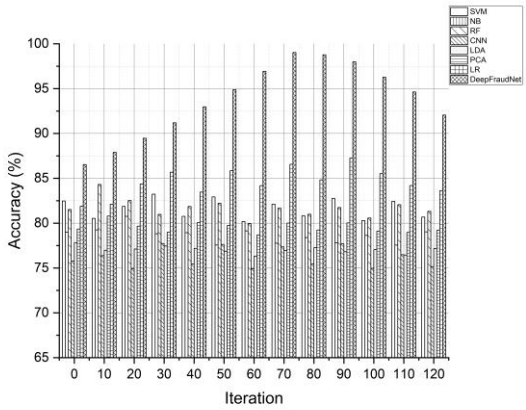

Figure 4(b): Accuracy Evaluation for Fraud Transaction Classification

Figures 4(a) and 4(b) depict the precision and accuracy of various methods over multiple iterations, such as Support Vector Machine (SVM) (Zhang, H., 2020), Naïve Bayes (NB) (Segurola-Gil, L., 2021), RF (Disha, R.A., 2022), CNN (Nedeljkovic, D., 2022), Linear Discriminant Analysis (LDA) (Solani, S., 2021), Principal Component Analysis (PCA) (Adhao, R., 2020), Logical Regression (LR) (De Cock, M., 2021), and Deep Fraud Net. Deep Fraud Net outperforms other techniques in terms of precision and accuracy on average. Deep Fraud Net excels in precision, with an average precision of 89.62%, surpassing all other methods. SVM attains an average precision of 80.47%, RF attains 79.48%, LR attains 82.32%, PCA attains 77.95%, NB attains 76.91%, CNN attains 74.7%, LDA attains 75.5%. The substantially higher precision value of Deep Fraud Net indicates its superior ability to detect and categorize instances of financial fraud and cybersecurity threats. Deep Fraud Net demonstrates remarkable performance in terms of accuracy, with an average accuracy of 93.43%. SVM attains an average accuracy of 81.39%, RF attains 80.84%, LR attains 83.52%, PCA attains 79.72%, NB attains 78.17%, CNN attains 75.91%, and LDA attains 76.71%. The greater average accuracy of Deep Fraud Net indicates its capacity to provide reliable and accurate classifications consistently. The results demonstrate that Deep Fraud Net outperforms the other methods' precision and accuracy, demonstrating its efficacy in detecting and classifying fraud. Deep Fraud Net is a promising framework for detecting and mitigating financial fraud and cybersecurity risks due to its greater precision and accuracy.

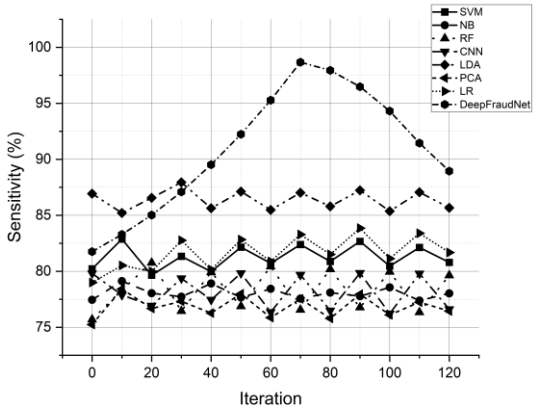

Figure 5(a): Sensitivity Evaluation for Fraud Transaction Classification

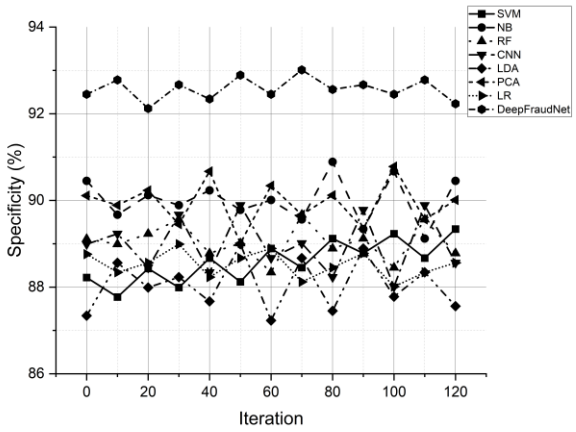

Figure 5(b): Specificity Evaluation for Fraud Transaction Classification

Figures 5(a) and 5(b) display the sensitivity and specificity results for various iterations of the SVM, NB, RF, CNN, LDA, PCA, LR, and Deep Fraud Net classification algorithms. Deep Fraud Net outperforms the competition in sensitivity, averaging 92.24% after 50 iterations. SVM reaches an average sensitivity of 81.50%, RF reaches 78.77%, LR reaches 80.18%, PCA reaches 77.48%, NB reaches 78.97%, CNN reaches 79.51%, and LDA reaches 85.85%. The higher average sensitivity value of Deep Fraud Net demonstrates its ability to detect instances of financial fraud and cybersecurity threats that are truly positive. Defrauded also exhibits superior performance in terms of specificity, obtaining an average specificity of 92.45% after 0 iterations. SVM attains a moderate specificity of 88.58%, RF attains 88.90%, LR attains 88.76%, PCA attains 89.67%, NB attains 89.90%, CNN attains 88.89%, and LDA attains 88.79%. Higher average specificity indicates that Deep Fraud Net can accurately identify true negative instances and reduce false positives. The framework's incorporation of noise reduction and deep learning techniques helps optimize input data and enhances its capacity to classify non-fraudulent instances accurately (Johnson, C., 2020).

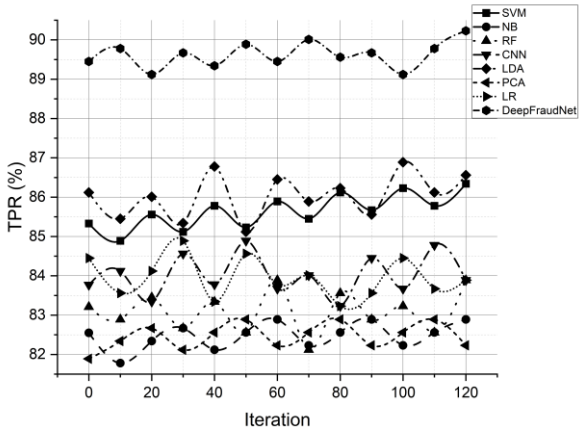

Figure 6(a): TPR Evaluation for Fraud Transaction Classification

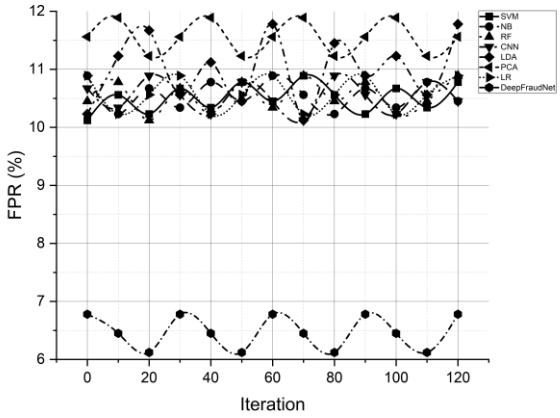

Figure 6(b): FPR Evaluation for Fraud Transaction Classification

Figures 6(a) and 6(b) illustrate the True Positive Rate (TPR) and False Positive Rate (FPR) for various iterations of SVM, NB, RF, CNN, LDA, PCA, LR, and Deep Fraud Net, among others. Deep Fraud Net obtains an average TPR of 89.12% after 100 iterations, outperforming the competition. SVM

has an average TPR of 85.89%, RF has an average TPR of 82.56%, LR has an average TPR of 83.67%, PCA has an average TPR of 86.56%, NB has an average TPR of 82.34%, CNN has an average TPR of 83.67%, and LDA has an average TPR of 86.89%. Deep Fraud Net can effectively detect and capture a high proportion of instances of financial fraud and cybersecurity threats with a higher average TPR. Deep Fraud Net demonstrates superior efficacy in terms of FPR, obtaining an average FPR of 6.45% after 10 iterations. SVM achieves an average FPR of 10.56%, RF achieves 10.45%, LR achieves 10.56%, PCA achieves 11.23%, NB achieves 10.67%, CNN achieves 10.78%, and LDA achieves 11.8%. Deep Fraud Net's reduced average FPR indicates its ability to effectively reduce false positives, thereby reducing the occurrence of legitimate transactions being misclassified as fraudulent. Deep Fraud Net's capacity to distinguish precisely between fraudulent and non-fraudulent transactions is improved through resilient feature extraction and noise reduction techniques.

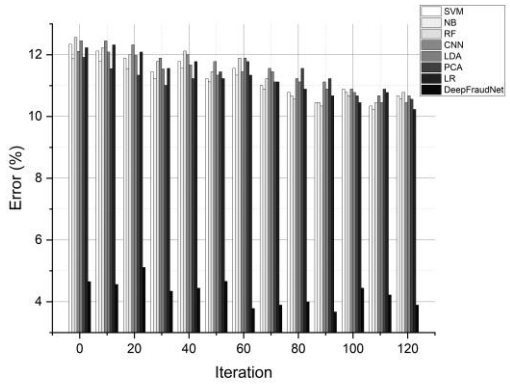

Figure 7: Error Evaluation for Fraud Transaction Classification

Figure 7 shows the results of different methods for additional repetitions, such as SVM, NB, RF, CNN, LDA, PCA, LR, and Deep Fraud Net. Here are the average results from all of the methods: SVM – 11.22, NB – 11.08, RF – 11.28, CNN – 11.45, LDA – 11.31, PCA – 11.28, LR – 11.14, Deep Fraud Net – 4.28. The results show that Deep Fraud Net always does better than all the other methods, with much lower mistake rates. Its average result of 4.28 shows that it is good at correctly identifying situations, while the mistake rates of the other methods range from 11.08 to 11.45. These results indicate that Deep Fraud Net is better than other systems at reducing classification mistakes and making scam detection systems more accurate. Deep Fraud Net can pick up complex patterns and features using its deep learning design. This makes it better at spotting fake actions and reduces the number of false positives.

Table 1. Simulation Result Analysis for the Fraud Transaction Classification

Metrics (%)	Precision	Accuracy	Sensitivity	Specificity	TPR	FPR
SVM	81.88	82.02	81.97	89.00	86.31	11.02
NB	79.37	80.27	78.64	90.44	83.35	11.33
RF	81.09	82.66	81.78	88.94	84.50	10.62
CNN	77.08	76.10	77.19	89.00	84.71	11.42
LDA	77.06	77.75	76.92	87.69	85.11	11.83
PCA	79.51	79.86	80.07	90.50	83.34	11.71
LR	84.60	84.86	84.89	89.05	84.49	11.48
Deep Fraud Net	98.85	93.35	99.09	93.16	89.58	7.34

Table 1 displays a comprehensive set of performance metrics for different methods: SVM, NB, RF, CNN, LDA, PCA, LR, and Deep Fraud Net. These metrics include Precision, Accuracy, Sensitivity, Specificity, TPR, and FPR. The findings suggest that Deep Fraud Net demonstrates superior performance in terms of Precision (98.85%), Accuracy (93.35%), Sensitivity (99.09%), and Specificity (93.16%) compared to other methods. Additionally, it demonstrates the lowest TPR at 89.58% and FPR at 7.34%. The outstanding outcomes underscore the exceptional efficacy of Deep Fraud Net in precisely detecting instances of fraud while mitigating the occurrence of false positives.

The exceptional results achieved by Deep Fraud Net can be ascribed to its sophisticated architecture and resilient feature extraction capabilities. Deep Fraud Net effectively utilizes deep learning techniques to detect and analyze complex patterns and features associated with fraudulent activities, resulting in high precision and accuracy. Moreover, the heightened sensitivity and specificity of the method suggest its capacity to identify a substantial number of accurate positive results while simultaneously minimizing the occurrence of false positives. Consequently, this contributes to fraud detection systems' overall dependability and effectiveness.

5 Conclusion and Future Scope

The importance of bolstering security measures in financial transactions is of utmost significance in the contemporary era of digital technology. A proposed approach was devised using sophisticated fraud detection techniques to tackle this matter. The proposed system, called Deep Fraud Net, integrates deep learning algorithms to detect fraudulent behaviors and safeguard individuals against potential financial harm.

Deep Fraud Net is notable for its distinctive characteristics. The system employs a deep neural network architecture to proficiently extract complex patterns and features from transaction data, enabling it to differentiate between fraudulent and legitimate transactions accurately. The simulation results consistently indicate that Deep Fraud Net outperforms other models across all metrics. The system demonstrates a remarkable level of precision, reaching 98.85%, which is evidence of its capacity to detect and classify fraudulent transactions effectively. With an accuracy rate of 93.35%, the system demonstrates a commendable level of reliability. The high sensitivity of 99.09% reflects the ability to accurately identify a substantial proportion of true positive cases. In comparison, the specificity of 93.16% guarantees a low rate of false positives, thereby minimizing the adverse effects on legitimate transactions. The system's accuracy in identifying fraudulent activities is reinforced by the TPR of 89.58% and FPR of 7.34%.

Despite its significant efficacy, the proposed method must overcome several obstacles. The availability and quality of training data significantly influence the performance of deep learning models. Acquiring extensive and heterogeneous datasets comprising fraudulent and legitimate transactions can present difficulties. The issue of interpretability in deep learning models continues to be a subject of concern, given their frequent characterization as opaque entities. Resolving these challenges will enhance the precision and dependability of the suggested approach.

Looking forward, the future potential of this research encompasses multiple domains. Continuous monitoring and adaptation of the Deep Fraud Net model will be imperative to effectively respond to evolving fraud patterns and techniques. The model's performance can be improved by incorporating real-time data feeds and employing dynamic feature extraction techniques. Moreover, the investigation of ensemble methods that integrate multiple fraud detection algorithms and the integration of explainable AI methods can enhance the interpretability of the model. Engaging in partnerships with financial

institutions and industry experts can facilitate the acquisition of extensive and varied datasets, thereby enhancing the effectiveness and dependability of fraud detection systems.

References

[1] Adhao, R., & Pachghare, V. (2020). Feature selection using principal component analysis and genetic algorithm. *Journal of Discrete Mathematical Sciences and Cryptography*, *23*(2), 595-602.

[2] Aggarwal, B.K., Gupta, A., Goyal, D., Gupta, P., Bansal, B., & Barak, D.D. (2022). A review on investigating the role of block-chain in cyber security. *Materials Today: Proceedings*, *56*, 3312-3316.

[3] Aschi, M., Bonura, S., Masi, N., Messina, D., & Profeta, D. (2022). Cybersecurity and fraud detection in financial transactions. In *Big Data and Artificial Intelligence in Digital Finance: Increasing Personalization and Trust in Digital Finance using Big Data and AI*, 269-278. Cham: Springer International Publishing.

[4] Chang, V., Di Stefano, A., Sun, Z., & Fortino, G. (2022). Digital payment fraud detection methods in digital ages and Industry 4.0. *Computers and Electrical Engineering*, *100*, 1-31.

[5] Corallo, A., Lazoi, M., & Lezzi, M. (2020). Cybersecurity in the context of industry 4.0: A structured classification of critical assets and business impacts. *Computers in industry*, *114*.

[6] De Cock, M., Dowsley, R., Nascimento, A.C., Railsback, D., Shen, J., & Todoki, A. (2021). High performance logistic regression for privacy-preserving genome analysis. *BMC Medical Genomics*, *14*, 1-18.

[7] Disha, R.A., & Waheed, S. (2022). Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique. *Cybersecurity*, *5*(1), 1.

[8] El-Rewini, Z., Sadatsharan, K., Selvaraj, D.F., Plathottam, S.J., & Ranganathan, P. (2020). Cybersecurity challenges in vehicular communications. *Vehicular Communications*, *23*.

[9] Fischer-Hübner, S., Alcaraz, C., Ferreira, A., Fernandez-Gago, C., Lopez, J., Markatos, E., & Akil, M. (2021). Stakeholder perspectives and requirements on cybersecurity in Europe. *Journal of information security and applications*, *61*.

[10] https://www.bancaditalia.it/statistiche/basi-dati/bds/index.html

[11] Jayanthi, E., Ramesh, T., Kharat, R.S., Veeramanickam, M.R.M., Bharathiraja, N., Venkatesan, R., & Marappan, R. (2023). Cybersecurity enhancement to detect credit card frauds in health care using new machine learning strategies. *Soft Computing*, *27*(11), 7555-7565.

[12] Johnson, C., Khadka, B., Basnet, R.B., & Doleck, T. (2020). Towards Detecting and Classifying Malicious URLs Using Deep Learning. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, *11*(4), 31-48.

[13] Martínez Torres, J., Iglesias Comesaña, C., & García-Nieto, P.J. (2019). Machine learning techniques applied to cybersecurity. *International Journal of Machine Learning and Cybernetics*, *10*, 2823-2836.

[14] Mishra, P. (2020). Big data digital forensic and cybersecurity. *Big data analytics and computing for digital forensic investigations*, *183*.

[15] Mughaid, A., AlZu'bi, S., Hnaif, A., Taamneh, S., Alnajjar, A., & Elsoud, E.A. (2022). An intelligent cyber security phishing detection system using deep learning techniques. *Cluster Computing*, *25*(6), 3819-3828.

[16] Nedeljkovic, D., & Jakovljevic, Z. (2022). CNN based method for the development of cyber-attacks detection algorithms in industrial control systems. *Computers & Security*, *114*.

[17] Nerurkar, P., Bhirud, S., Patel, D., Ludinard, R., Busnel, Y., & Kumari, S. (2021). Supervised learning model for identifying illegal activities in Bitcoin. *Applied Intelligence*, *51*, 3824-3843.

[18] Pinto, L., Brito, C., Marinho, V., & Pinto, P. (2022). Assessing the Relevance of Cybersecurity Training and Policies to Prevent and Mitigate the Impact of Phishing Attacks. *Journal of Internet Services and Information Security*, *12*(4), 23-38.
[19] Samtani, S., Zhao, Z., & Krishnan, R. (2023). Secure Knowledge Management and Cybersecurity in the Era of Artificial Intelligence. *Information Systems Frontiers*, *25*(2), 425-429.
[20] Sarker, I.H. (2021). Deep cybersecurity: a comprehensive overview from neural network and deep learning perspective. *SN Computer Science*, *2*(3), 1-18.
[21] Segurola-Gil, L., Zola, F., Echeberria-Barrio, X., & Orduna-Urrutia, R. (2021). NBcoded: network attack classifiers based on Encoder and Naive Bayes model for resource limited devices. *In Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 55-70. Cham: Springer International Publishing.
[22] Shafiq, M., Tian, Z., Bashir, A. K., Jolfaei, A., & Yu, X. (2020). Data mining and machine learning methods for sustainable smart cities traffic classification: A survey. *Sustainable Cities and Society*, *60*.
[23] Shaukat, K., Luo, S., Varadharajan, V., Hameed, I.A., & Xu, M. (2020). A survey on machine learning techniques for cyber security in the last decade. *IEEE access*, *8*, 222310-222354.
[24] Singh, A., Ranjan, R.K., & Tiwari, A. (2022). Credit card fraud detection under extreme imbalanced data: a comparative study of data-level algorithms. *Journal of Experimental & Theoretical Artificial Intelligence*, *34*(4), 571-598.
[25] Solani, S., & Jadav, N.K. (2021). A Novel Approach to Reduce False-Negative Alarm Rate in Network-Based Intrusion Detection System Using Linear Discriminant Analysis. In *Inventive Communication and Computational Technologies: Proceedings of ICICCT 2020*, 911-921. Springer Singapore.
[26] Tahaei, H., Afifi, F., Asemi, A., Zaki, F., & Anuar, N.B. (2020). The rise of traffic classification in IoT networks: A survey. *Journal of Network and Computer Applications*, *154*.
[27] Tolba, A., & Al-Makhadmeh, Z. (2021). A cybersecurity user authentication approach for securing smart grid communications. *Sustainable Energy Technologies and Assessments*, *46*.
[28] Zhang, H., Li, Y., Lv, Z., Sangaiah, A.K., & Huang, T. (2020). A real-time and ubiquitous network attack detection based on deep belief network and support vector machine. *IEEE/CAA Journal of Automatica Sinica*, *7*(3), 790-799.
[29] Zhang, Z., Al Hamadi, H., Damiani, E., Yeun, C.Y., & Taher, F. (2022). Explainable artificial intelligence applications in cyber security: State-of-the-art in research. *IEEE Access*.

Authors Biography

Professor & Dean. Dr. UdayaKumar Ramanathan completed his M. S (Information Technology and Management) from A.V.C. College of Engineering and Awarded Ph.D. in the year 2011. He is serving in Teaching & Research community for more than two decades, he successfully produced 5 Doctoral candidates, he is a researcher, contribute the Research work in inter disciplinary areas. He is having h-index of 27, citation 2949(Scopus). He is associated as Dean –Department of computer science and Information Technology, Kalinga University, Raipur, Chhattisgarh.

Dr.A. Joshi, is a professor of computer science and engineering. She is having Two decades of experience in Teaching and Research. She is holding several academic and administrative experience. She attended and Organized many National and International conferences. Currently she continues as Professor in Department of Computer science and Engineering @ Panimalar Engineering College.

Dr.S.S. Boomiga received his B.E., degree from the Madras University Chennai, India in 1999, her M.E., degree from College of Engineering, Anna University, Chennai, India, in 2004, and her Ph.D., degree from Pondicherry University, Chennai, India, in 2022. For the past 13 years since 1999, she has worked in different positions like Assistant Professor, Associate Professor in various reputed engineering colleges across India. She is currently working as an Associate Professor in the Department of Artificial Intelligence and Data Science at Sri Manakula Vinayagar Engineering College, Pondicherry, India. Her research interests include Edge Computing, the Internet of Things, Software Engineering, and Machine Learning. She has published more than 20 research articles in various international journals and conference proceedings. She is a lifetime member of ISTE.

R. Sugumar has received his BE degree from the University of Madras, Chennai, India in 2003, M. Tech degree from Dr. M.G.R. Educational and Research Institute, Chennai, India, in 2007, and PhD degree from Bharath University, Chennai, India, in 2011. From 2003 to 2021, he has worked at different positions like Assistant Professor, Associate Professor, Professor & HOD in various reputed engineering colleges across India. He is currently working as a Professor in the Department of Computer Science and Engineering at Saveetha School of Engineering, SIMATS, Chennai, India. His research interests include data mining, cloud computing and networks. He has published more than 45 research articles in various international journals and conference proceedings. He is acting as a reviewer in various national and international journals. He has chaired various international and national conferences. He is a life time member of ISTE and CSI.