International Journal of
LAW, ARTS AND HUMANITIES
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0523 www.zesterapublications.com Original Research Paper

INTELLIGENT CREDIT CARD FRAUD PREVENTION USING GRAPH NEURAL NETWORKS AND AUTOENCODER MODELS

SRIKANTH JINDAM[1], LIKKI LEELA VARDHAN[2], LINGAM RAM AKSHITH[3], MATURI KALYANI[4], M A SHASHANK YADAV[5], MALYALA GOUTHAM GOUMITH[6]

ASSISTANT PROFESSOR[1], UG SCHOLAR[2,3,4,5&6]

DEPARTMENT OF CSE, NARSIMHA REDDY ENGINEERING COLLEGE (UGC- AUTONOMOUS) MAISAMMAGUDA (V), KOMPALLY, SECUNDERABAD, TELANGANA-500100

ABSTRACT

Credit card fraud has become a major concern in modern digital banking due to the rapid growth of online transactions and electronic payment systems. Traditional fraud detection methods often rely on rule-based systems or conventional machine learning algorithms, which may struggle to detect complex and evolving fraud patterns in real time. To address these challenges, this study proposes an intelligent credit card fraud prevention framework using Graph Neural Networks (GNN) and Autoencoder models. The proposed approach leverages the ability of Graph Neural Networks to capture relationships and interactions between entities such as users, transactions, and merchants. By representing transaction data as a graph structure, the model can identify suspicious patterns and hidden connections that may indicate fraudulent activities. Additionally, an autoencoder-based anomaly detection mechanism is employed to learn normal transaction behavior and detect unusual deviations that may signal potential fraud. The integration of graph-based learning and deep anomaly detection enhances the system's capability to detect both known and unknown fraud patterns. The proposed framework processes transaction data in near real time, enabling faster and more accurate fraud identification while reducing false positives. Experimental evaluation demonstrates that the combined use of GNN and autoencoder models significantly improves fraud detection performance compared to traditional methods. This intelligent fraud detection system can assist financial institutions in strengthening security, minimizing financial losses, and improving trust in digital banking services by providing an effective solution for real-time credit card fraud prevention.

INTRODUCTION

The rapid growth of digital banking and online payment systems has significantly increased the use of **credit cards for financial transactions**. While these technologies provide convenience and efficiency, they have also led to a rise in **credit card fraud and financial cybercrimes**. Fraudulent activities such as unauthorized transactions, identity theft, and account takeovers cause substantial financial losses for banks and customers. Therefore, developing effective and intelligent fraud detection systems has become a critical requirement for modern banking institutions.

Traditional credit card fraud detection methods mainly rely on **rule-based systems and statistical analysis**. These approaches use predefined rules or thresholds to identify suspicious transactions. However, such systems often fail to detect sophisticated fraud patterns because fraudsters continuously adapt their techniques to bypass detection mechanisms. Moreover, rule-based systems may generate **high false positive rates**, which can inconvenience legitimate customers and reduce the efficiency of fraud prevention systems.

With the advancement of **Artificial Intelligence (AI) and Machine Learning (ML)**, modern fraud detection systems have started using data-driven approaches to identify complex patterns in transaction data. Deep learning models can analyze large volumes of financial transactions and automatically learn features that indicate fraudulent behavior. Among these approaches, **Graph Neural Networks (GNNs)** and **Autoencoder models** have gained significant attention due to their ability to capture hidden relationships and detect anomalies in large datasets.

Graph Neural Networks are particularly effective in modeling **transaction networks**, where nodes represent entities such as users, credit cards, and merchants, while edges represent financial transactions. By analyzing the relationships between these entities, GNNs can identify suspicious patterns and detect coordinated fraud activities that may not be visible through traditional methods. On the other hand, **Autoencoders** are powerful unsupervised learning models used for **anomaly detection**. They learn normal transaction patterns and identify deviations that may indicate fraudulent activity.

By combining **Graph Neural Networks with Autoencoder-based anomaly detection**, an intelligent framework can be developed for real-time credit card fraud prevention. This hybrid approach leverages the strengths of both techniques: the relational learning capability of GNNs and the anomaly detection capability of autoencoders. As a result, the system can detect complex fraud patterns, reduce false alarms, and improve the overall accuracy and efficiency of fraud detection in banking systems.

The proposed intelligent fraud prevention system aims to provide **real-time monitoring of credit card transactions**, enabling financial institutions to detect and prevent fraudulent activities quickly. Such advanced AI-driven solutions can enhance banking security, protect customer financial assets, and maintain trust in digital payment ecosystems.

LITERATURE REVIEW

Introduction to Credit Card Fraud Detection

Credit card fraud has become a major challenge for financial institutions due to the rapid growth of digital banking and online transactions. Fraudulent activities lead to significant financial losses and reduced customer trust. Traditional fraud detection systems rely on rule-based mechanisms and manual verification processes. However, these methods struggle to detect **sophisticated fraud patterns and real-time fraudulent transactions**, leading researchers to explore **machine learning and deep learning approaches** for more effective fraud detection.

International Journal of
LAW, ARTS AND HUMANITIES
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0523 www.zesterapublications.com Original Research Paper

Traditional Machine Learning Methods for Fraud Detection

Earlier research used machine learning algorithms such as **Logistic Regression, Decision Trees, Support Vector Machines (SVM), and Random Forest** to identify fraudulent transactions. These models analyze transaction attributes such as transaction amount, location, frequency, and user behavior patterns.

Studies have shown that ensemble learning methods like Random Forest can improve detection accuracy compared to single classifiers. However, traditional machine learning models often require **manual feature engineering** and may struggle to capture complex relationships among transactions, especially in large financial datasets.

Deep Learning Approaches in Fraud Detection

Deep learning models have recently gained popularity because they can automatically learn complex patterns from large-scale financial data. Techniques such as **Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN)** have been used to detect anomalies in credit card transactions.

Deep learning models can identify subtle behavioral patterns in transaction data and improve detection accuracy. However, many deep learning approaches treat transactions as independent events and fail to consider **relationships between customers, merchants, and transactions**, which are important for identifying fraud networks.

Graph Neural Networks for Fraud Detection

Graph Neural Networks (GNNs) have emerged as a powerful approach for modeling relational data. In financial systems, transactions can be represented as **graph structures**, where nodes represent entities such as customers, credit cards, and merchants, while edges represent transaction relationships.

GNN models analyze these relationships to identify suspicious patterns such as **fraud rings, coordinated fraudulent transactions, and abnormal interaction networks**. Research shows that GNN-based models significantly improve fraud detection performance by capturing complex dependencies between transaction entities.

Autoencoder-Based Anomaly Detection

Autoencoders are a type of unsupervised deep learning model used for **anomaly detection**. They learn to reconstruct normal transaction patterns from training data. When a fraudulent transaction occurs, the model produces a higher reconstruction error because the pattern differs from normal behavior.

Autoencoder-based models are particularly useful for detecting **unknown or previously unseen fraud patterns**. Variants such as **Variational Autoencoders (VAE) and Deep Autoencoders** have been applied in financial fraud detection systems to identify anomalous transactions.

Hybrid Models Combining GNN and Autoencoders

Recent research focuses on **hybrid frameworks that combine Graph Neural Networks with Autoencoders**. In these models, GNNs capture the relational structure of financial transactions, while autoencoders detect anomalies in transaction behavior.

This hybrid approach improves fraud detection by combining **structural analysis and anomaly detection**. The system can identify both known fraud patterns and emerging fraudulent activities in real time, making it suitable for modern banking systems.

Research Challenges and Future Directions

Despite significant advancements, several challenges remain in credit card fraud detection research:

- Handling **highly imbalanced datasets** where fraudulent transactions are rare
- Ensuring **real-time detection with low latency**
- Protecting **customer privacy and sensitive financial data**
- Improving model interpretability for financial regulators

Future research is expected to focus on **explainable AI, federated learning, and scalable graph-based deep learning models** to enhance fraud detection systems.

SYSTEM ANALYSIS

EXISTING SYSTEM

Traditional credit card fraud detection systems in banking mainly rely on **rule-based systems and classical machine learning algorithms**. These systems analyze transaction details such as transaction amount, location, time, merchant information, and cardholder behavior to identify suspicious activities. Banks often use techniques such as **Logistic Regression, Decision Trees, Random Forest, and Support Vector Machines (SVM)** to classify transactions as legitimate or fraudulent.

In many existing systems, predefined rules are used to detect fraud patterns. For example, transactions exceeding a certain limit or occurring in unusual locations may trigger fraud alerts. While these approaches are useful, they have several limitations. Fraud detection models often struggle to identify **complex and evolving fraud patterns**, especially when fraudsters use sophisticated techniques to mimic legitimate behavior.

Additionally, most traditional systems treat each transaction **independently**, ignoring relationships between users, merchants, devices, and transaction networks. This makes it difficult to detect **organized fraud rings or coordinated fraudulent activities**. Another challenge is **data imbalance**, as fraudulent transactions represent a very small percentage of the total transaction data, which reduces model accuracy.

PROPOSED SYSTEM

The proposed system introduces an **Intelligent Credit Card Fraud Prevention framework using Graph Neural Networks (GNN) and Autoencoder-based Deep Learning models**. This approach enhances fraud detection accuracy by analyzing both **transaction features and the relationships between entities in the transaction network**.

In this system, credit card transaction data is first converted into a **graph structure**, where nodes represent entities such as

International Journal of
LAW, ARTS AND HUMANITIES
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0523 www.zesterapublications.com Original Research Paper

customers, credit cards, merchants, and devices, and edges represent relationships between them. A **Graph Neural Network (GNN)** is then applied to capture complex interactions and hidden patterns within the transaction network.

An **Autoencoder model** is used for anomaly detection by learning the normal patterns of legitimate transactions. When a new transaction significantly deviates from these learned patterns, it is flagged as potentially fraudulent.

By combining **GNN-based relational learning with autoencoder-based anomaly detection**, the system can detect both known and previously unseen fraud patterns. The model continuously learns from new transaction data and improves its detection capability over time.

The proposed framework supports **real-time transaction monitoring**, enabling banks to quickly identify suspicious transactions and prevent financial losses.

ADVANTAGES OF THE PROPOSED SYSTEM

- Improved detection of complex and coordinated fraud patterns
- Ability to detect previously unseen fraud activities
- Better modeling of relationships between transactions and entities
- Reduced false positives compared to traditional systems
- Real-time fraud detection capability
- Scalable solution for large banking transaction datasets

IMPLEMENTATION

1. Data Collection Module

The implementation begins with collecting **credit card transaction datasets** from banking systems or publicly available sources. The dataset typically contains transaction details such as **transaction amount, time, location, merchant information, card ID, and transaction history**. These records help identify patterns of normal and fraudulent transactions.

2. Data Preprocessing Module

In this stage, the collected data is cleaned and prepared for model training. Tasks include **handling missing values, removing duplicate records, normalizing numerical values, and encoding categorical features** such as merchant category or location. Data balancing techniques such as **SMOTE (Synthetic Minority Oversampling Technique)** may also be used because fraud datasets are usually highly imbalanced.

3. Graph Construction Module

A graph structure is created from the transaction data. In this graph:

- **Nodes** represent entities such as customers, cards, merchants, and transactions.
- **Edges** represent relationships between these entities, such as a transaction between a cardholder and a merchant.

This graph-based representation helps capture hidden relationships between fraudulent transactions and suspicious entities.

4. Autoencoder-Based Anomaly Detection Module

An **autoencoder neural network** is trained on normal transaction data to learn typical transaction patterns. The autoencoder compresses the input data into a lower-dimensional representation and then reconstructs it. If a transaction has a **high reconstruction error**, it indicates that the pattern deviates from normal behavior and may be fraudulent.

5. Graph Neural Network (GNN) Training Module

The constructed graph is used to train a **Graph Neural Network**. The GNN learns complex relationships between nodes in the transaction network and identifies suspicious patterns such as fraud rings or coordinated attacks. The GNN aggregates information from neighboring nodes to improve prediction accuracy.

6. Hybrid Fraud Detection Module

In this module, the outputs from the **Autoencoder and Graph Neural Network** are combined to create a hybrid detection system. The autoencoder detects unusual transaction behavior, while the GNN identifies suspicious relationships within the transaction network. This combination improves detection accuracy and reduces false positives.

7. Model Evaluation Module

The performance of the fraud detection model is evaluated using metrics such as:

- **Accuracy**
- **Precision**
- **Recall**
- **F1-Score**
- **ROC-AUC Score**

These metrics help measure how effectively the system identifies fraudulent transactions.

8. Real-Time Fraud Detection and Deployment Module

The trained model is deployed within the **banking transaction processing system**. As new transactions occur, the system analyzes them in real time using the trained model. Suspicious transactions are flagged for further verification or automatically blocked to prevent financial loss.

9. Visualization and Monitoring Module

A monitoring dashboard is implemented to track fraud detection results. The dashboard provides insights into **fraud alerts, transaction patterns, risk scores, and system performance**, enabling banks to monitor and improve their fraud detection strategies.

10.Algorithms

International Journal of
LAW, ARTS AND HUMANITIES
Peer Reviewed, Referred & Indexed Journal
www.zesterapublications.com
ISSN: 3070-0523
Original Research Paper

DECISION TREE CLASSIFIERS

Decision tree classifiers are used successfully in many diverse areas. Their most important feature is the capability of capturing descriptive decision making knowledge from the supplied data. Decision tree can be generated from training sets. The procedure for such generation based on the set of objects (S), each belonging to one of the classes C1, C2, …, Ck is as follows:

Step 1. If all the objects in S belong to the same class, for example Ci, the decision tree for S consists of a leaf labeled with this class

Step 2. Otherwise, let T be some test with possible outcomes O1, O2,…, On. Each object in S has one outcome for T so the test partitions S into subsets S1, S2,… Sn where each object in Si has outcome Oi for T. T becomes the root of the decision tree and for each outcome Oi we build a subsidiary decision tree by invoking the same procedure recursively on the set Si.

GRADIENT BOOSTING Gradient boosting is a machine learning technique used in regression and classification tasks, among others. It gives a prediction model in the form of an ensemble of weak prediction models, which are typically decision trees.[1][2] When a decision tree is the weak learner, the resulting algorithm is called gradient-boosted trees; it usually outperforms random forest.A gradient-boosted trees model is built in a stage-wise fashion as in other boosting methods, but it generalizes the other methods by allowing optimization of an arbitrary differentiable loss function.

K-NEAREST NEIGHBORS (KNN)

- Simple, but a very powerful classification algorithm
- Classifies based on a similarity measure
- Non-parametric
- Lazy learning
- Does not “learn” until the test example is given
- Whenever we have a new data to classify, we find its K-nearest neighbors from the training data

Example

- Training dataset consists of k-closest examples in feature space
- Feature space means, space with categorization variables (non-metric variables)
- Learning based on instances, and thus also works lazily because instance close to the input vector for test or prediction may take time to occur in the training dataset

LOGISTIC REGRESSION CLASSIFIERS

Logistic regression analysis studies the association between a categorical dependent variable and a set of independent (explanatory) variables. The name *logistic regression* is used when the dependent variable has only two values, such as 0 and 1 or Yes and No. The name *multinomial logistic regression* is usually reserved for the case when the dependent variable has three or more unique values, such as Married, Single, Divorced, or Widowed. Although the type of data used for the dependent variable is different from that of multiple regression, the practical use of the procedure is similar. Logistic regression competes with discriminant analysis as a method for analyzing categorical-response variables. Many statisticians feel that logistic regression is more versatile and better suited for modeling most situations than is discriminant analysis. This is because logistic regression does not assume that the independent variables are normally distributed, as discriminant analysis does. This program computes binary logistic regression and multinomial logistic regression on both numeric and categorical independent variables. It reports on the regression equation as well as the goodness of fit, odds ratios, confidence limits, likelihood, and deviance. It performs a comprehensive residual analysis including diagnostic residual reports and plots. It can perform an independent variable subset selection search, looking for the best regression model with the fewest independent variables. It provides confidence intervals on predicted values and provides ROC curves to help determine the best cutoff point for classification. It allows you to validate your results by automatically classifying rows that are not used during the analysis.

NAÏVE BAYES

The naive bayes approach is a supervised learning method which is based on a simplistic hypothesis: it assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature .Yet, despite this, it appears robust and efficient. Its performance is comparable to other supervised learning techniques. Various reasons have been advanced in the literature. In this tutorial, we highlight an explanation based on the representation bias. The naive bayes classifier is a linear classifier, as well as linear discriminant analysis, logistic regression or linear SVM (support vector machine). The difference lies on the method of estimating the parameters of the classifier (the learning bias).While the Naive Bayes classifier is widely used in the research world, it is not widespread among practitioners which want to obtain usable results. On the one hand, the researchers found especially it is very easy to program and implement it, its parameters are easy to estimate, learning is very fast even on very large databases, its accuracy is reasonably good in comparison to the other

International Journal of
LAW, ARTS AND HUMANITIES
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0523 www.zesterapublications.com Original Research Paper

approaches. On the other hand, the final users do not obtain a model easy to interpret and deploy, they does not understand the interest of such a technique. Thus, we introduce in a new presentation of the results of the learning process. The classifier is easier to understand, and its deployment is also made easier. In the first part of this tutorial, we present some theoretical aspects of the naive bayes classifier. Then, we implement the approach on a dataset with Tanagra. We compare the obtained results (the parameters of the model) to those obtained with other linear approaches such as the logistic regression, the linear discriminant analysis and the linear SVM. We note that the results are highly consistent. This largely explains the good performance of the method in comparison to others. In the second part, we use various tools on the same dataset (Weka 3.6.0, R 2.9.2, Knime 2.1.1, Orange 2.0b and RapidMiner 4.6.0). We try above all to understand the obtained results.

RANDOM FOREST

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees. For regression tasks, the mean or average prediction of the individual trees is returned. Random decision forests correct for decision trees' habit of overfitting to their training set. Random forests generally outperform decision trees, but their accuracy is lower than gradient boosted trees. However, data characteristics can affect their performance. The first algorithm for random decision forests was created in 1995 by Tin Kam Ho[1] using the random subspace method, which, in Ho's formulation, is a way to implement the "stochastic discrimination" approach to classification proposed by Eugene Kleinberg. An extension of the algorithm was developed by Leo Breiman and Adele Cutler, who registered "Random Forests" as a trademark in 2006 (as of 2019, owned by Minitab, Inc.).The extension combines Breiman's "bagging" idea and random selection of features, introduced first by Ho[1] and later independently by Amit and Geman[13] in order to construct a collection of decision trees with controlled variance. Random forests are frequently used as "blackbox" models in businesses, as they generate reasonable predictions across a wide range of data while requiring little configuration.

SVM

In classification tasks a discriminant machine learning technique aims at finding, based on an *independent and identically distributed* (*iid*) training dataset, a discriminant function that can correctly predict labels for newly acquired instances. Unlike generative machine learning approaches, which require computations of conditional probability distributions, a discriminant classification function takes a data point x and assigns it to one of the different classes that are a part of the classification task. Less powerful than generative approaches, which are mostly used when prediction involves outlier detection, discriminant approaches require fewer computational resources and less training data, especially for a multidimensional feature space and when only posterior probabilities are needed. From a geometric perspective, learning a classifier is equivalent to finding the equation for a multidimensional surface that best separates the different classes in the feature space. SVM is a discriminant technique, and, because it solves the convex optimization problem analytically, it always returns the same optimal hyperplane parameter—in contrast to *genetic algorithms* (*GAs*) or *perceptrons*, both of which are widely used for classification in machine learning. For perceptrons, solutions are highly dependent on the initialization and termination criteria. For a specific kernel that transforms the data from the input space to the feature space, training returns uniquely defined SVM model parameters for a given training set, whereas the perceptron and GA classifier models are different each time training is initialized. The aim of GAs and perceptrons is only to minimize error during training, which will translate into several hyperplanes' meeting this requirement.

CONCLUSION

In the end, we conclude that deep learning can substitute humans, but many researchers are still not confident that robots can be instructed like humans. Globally, financial institutions are using deep learning with business intelligence to control money laundering. Our research work is based on Fraud Detection with Graph Neural Networks and Anomaly detection by using Autoencoders in Credit cards. The results of our research work help many financial organizations, especially banks, in achieving an adequate understanding of deep learning with business intelligence. Our study in future also assists the banks in Pakistan to determine the gap between current operational methods and emerging procedures that utilize deep learning and business intelligence to detect fraud. To detect credit card fraud, we define the problem as a semi-supervised node classification job and describe credit card behaviors as a temporal transaction graph. By using a brand-new attribute–driven temporal graph neural network, we can detect credit card fraud. For extracting temporal and attribute data, we specifically suggest a gated temporal attention network. So, by taking advantage of both labelled and unlabeled data, we pass attributes and risk information on the temporal transaction graph. To improve overall performance, autoencoders can be used in conjunction with other fraud detection strategies. For instance to develop a complete fraud detection solution, rule-based systems, machine learning classifiers, or network analysis algorithms can be combined with anomaly detection utilizing auto encoders. It is feasible to create a fraud detection system using a deep learning strategy, and it is possible to minimize the work required to manually label a dataset using both supervised and unsupervised learning approaches.

International Journal of
LAW, ARTS AND HUMANITIES
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0523 www.zesterapublications.com Original Research Paper

During the model update, there is a possibility to integrate the identification of novel fraud types. The method for updating the model relies on the amount of labelled data. In our research work we use Python for the analysis purpose to boost the efficiency of the business organization. Hence, in the end, we conclude that deep learning with business intelligence offers feasible explanations for the various business domains, and by using artificial intelligence with deep learning in business intelligence, we can achieve our targets more efficiently.

REFERENCES

[1] P. Tiwari, S. Mehta, N. Sakhuja, J. Kumar, and A. K. Singh, "Credit card fraud detection using machine learning: A study," 2021,
arXiv:2108.10005.

[2] T. Micro, "Deep security? Software," Tech. Rep., 2020.

[3] G. K.Kulatilleke, "Challenges and complexities in machine learning based credit card fraud detection," 2022, arXiv:2208.10943.

[4] S. K. Hashemi, S. L. Mirtaheri, and S. Greco, "Fraud detection in banking data by machine learning techniques," IEEE Access, vol. 11,
pp. 3034–3043, 2023.

[5] R. B. Sulaiman, V. Schetinin, and P. Sant, "Review of machine learning approach on credit card fraud detection," Hum.-Centric Intell. Syst., vol. 2, nos. 1–2, pp. 55–68, 2022.

[6] E. Btoush, X. Zhou, R. Gururaian, K. Chan, and X. Tao, "Asurvey on credit card fraud detection techniques in banking industry for cyber security," in Proc. 8th Int. Conf. Behav. Social Comput. (BESC), Oct. 2021, pp. 1–7.

[7] Y. Bao, G. Hilary, and B. Ke, "Artificial intelligence and fraud detection," in Innovative Technology at the Interface of Finance and Operations, vol. 1, 2022, pp. 223–247.

[8] A. Mahalle, J. Yong, X. Tao, and J. Shen, "Data privacy and system security for banking and financial services industry based on cloud computing infrastructure," in Proc. IEEE 22nd Int. Conf. Comput. Supported Cooperat. Work Design ((CSCWD)), Nanjing, China, May 2018, pp. 407–413, doi: 10.1109/CSCWD.2018.8465318.

[9] N. Karkashadze, G. Shanidze, M. Shalamberidze, and S. Mikabadze, "Modern challenges in agribusiness," Int. J. Innov. Technol. Economy, vol. 2, no. 38, Jun. 2022, doi: 10.31435/rsglobal_ijite/30062022/7828.

[10] F. Manessi, A. Rozza, and M. Manzo, "Dynamic graph convolutional networks," 2017, arXiv:1704.06199.

[11] T. Kanan, A. Mughaid, R. Al-Shalabi, M. Al-Ayyoub, M. Elbes, and O. Sadaqa, "Business intelligence using deep learning techniques for social media contents," Cluster Comput., vol. 26, no. 2, pp. 1285–1296, Apr. 2023, doi: 10.1007/s10586-022-03626-y.

[12] A. Bouguettaya, H. Zarzour, A. Kechida, and A. M. Taberkit, "Machine learning and deep learning as new tools for business analytics," in Handbook of Research on Foundations and Applications of Intelligent Bus. Analytics, 2022, doi: 10.4018/978-1-7998-9016-4.ch008.

[13] Y. Liu, Z. Sun, and W. Zhang, "Improving fraud detection via hierarchical attention-based graph neural network," J. Inf. Secur. Appl., vol. 72, Feb. 2023, Art. no. 103399, doi: 10.1016/j.jisa.2022.103399.

[14] H. A. Bukhori and R. Munir, "Inductive link prediction banking fraud detection system using homogeneous graph-based machine learning model," in Proc. IEEE 13th Annu. Comput. Commun. Workshop Conf. (CCWC), Mar. 2023, pp. 246–251, doi: 10.1109/CCWC57344.2023.10099180.

[15] Z. Li, B. Wang, J. Huang, Y. Jin, Z. Xu, J. Zhang, and J. Gao, "A graph-powered large-scale fraud detection system," Int. J. Mach. Learn.
Cybern., vol. 15, no. 1, pp. 115–128, Jan. 2024, doi: 10.1007/s13042- 023-01786