International Journal of Artificial Intelligence, Data Science, and Machine Learning
Grace Horizon Publication | Volume 4, Issue 4, 87-96, 2023
ISSN: 3050-9262 | https://doi.org/10.63282/3050-9262.IJAIDSML-V4I4P110

Original Article

Synthetic Identity Detection Using Graph Neural Networks

Sunil Anasuri
Independent Researcher, USA.

Abstract *- Synthetic identity fraud has become one of the most significant problems for financial institutions and online services. This is because synthetic identities can use a combination of real and false information, and through non-linear relationships and changing patterns, are difficult to identify using conventional procedures. This paper will explore Graph Neural Networks (GNNs) in the context of synthetic identity detection, which lies in their potential to capture the structure underlying relational data and learn complex network representations. As an adaptive way of thinking, GNNs utilise the graph structure to represent the identities of users, transaction histories, and social ties, taking into account dependencies and abnormal patterns that are not captured by traditional machine learning algorithms. We present our solution that consists of feature engineering, graph construction, and classification using GNNs, which shows its efficiency, built on benchmark datasets. There will be an increase in detection, a decrease in false positives, and an increase in interpretability. This paper proposes a contribution to the field of fraud detection by combining deep learning methods with graph-based learning to achieve a scalable and dynamic algorithm for classifying synthetic names in dynamic, networked digital environments.*

Keywords *- Synthetic Identity, Fraud Detection, Graph Neural Networks, Anomaly Detection, Machine Learning, Graph Embedding.*

1. Introduction

Synthetic identity fraud has also become one of the rapidly developing types of financial crime, presenting complex issues to credit providers, banks, and other financial entities worldwide. In comparison to the classical situation of identity theft or its misappropriation, which presupposes the theft or misuse of already existing personal information of an individual, synthetic fraud is more advanced in that it is characterized by the creation of new, fictitious identities based on the combination of pieces of some true information and fake details. For example, someone posing as a fraudster might use a valid Social Security number in conjunction with a false name, address, or date of birth to create a synthetic identity that appears realistic. [1-3] these fake identities are subsequently used to open bank accounts, obtain loans, obtain government benefits or get into all types of financial transactions. The worst aspect of synthetic identity fraud is that it can remain undetected for extended periods.

In contrast to stolen identities, where theft can usually be detected by a legitimate owner, synthetic identities will not have a back door to report peculiar activity. Moreover, the non-linearity and decentralization of financial and personal data across the industry allow them to be used by modern detection systems that rely on rule-based checks or account-level anomalous behavior standards, which are not able to detect any suspicious activity. Fraudsters are also able to establish credibility with such synthetic identities over time by keeping the fraud small and low-risk, only to later exploit the trust through large and high-risk actions, which can cause significant losses to organizations. This mix of stealthiness, elasticity, and its late isolation makes it even more critical to establish more sophisticated, analytics-driven methods of combating synthetic identity fraud at all levels.

1.1. Evolution of Synthetic Identity Detection

The detection of synthetic identity fraud has undergone a significant shift in recent years, as fraudsters have become increasingly sophisticated and the security processes in place have reached their limits. Initially, financial institutions relied on rule-based systems, through which predetermined limits such as unusually high transaction values, sudden increases in activity, or clashes of demographic details were used to identify potentially suspicious accounts. Although useful in identifying simple anomalies, these static systems were not very flexible, and fraudsters soon figured out how to exploit their various systems to ensure that they did not exceed their cutoff points. The subsequent development involved the use of supervised machine learning algorithms, such as logistic regressions, random forests, and support vector machines, based on past information on fraud to identify accounts as authentic or fraudulent. These models enhanced the accuracy of detection by modeling on the structured constructs of activity on the account, the frequency of transactions, and demographic covariates. Nevertheless, their application requires labeled datasets, and their inability to resolve cross-boundary relationships affecting multiple entities limited the possibility of counter synthetic identities that tend to exist across devices and accounts as well as networks. To overcome these weaknesses, scientists started to investigate anomaly detection tools that should detect unusual patterns without having large amounts of labeled information.

Such unsupervised methods were highly flexible but commonly generated a significant rate of false positives because they are unable to distinguish between warranted yet infrequent behavior and real fraud. In recent years, graph modelling has gained prominence, utilising relations between entities (such as accounts, devices, IP addresses, and social interactions) as interconnected networks. The paradigm enables the identification of fraud rings and concealed correlations that are impossible to discover through an on-account analysis. Empowered by this, Graph Neural Networks (GNNs) have emerged as the state-of-the-art offering a method to combine node features and the relational structure of a graph in order to define context-sensitive representations. Such evolution constitutes a transformation of rule-based detection to an advanced, data-driven, and relational approach that can support the increasing sophistication of synthetic identity fraud.

1.2. Importance of Graph Neural Networks

GNNs have become an effective paradigm to model the interconnected, complex data, and therefore, it is particularly effective in addressing synthetic identity fraud. [4,5] They are important in this sphere in various aspects:

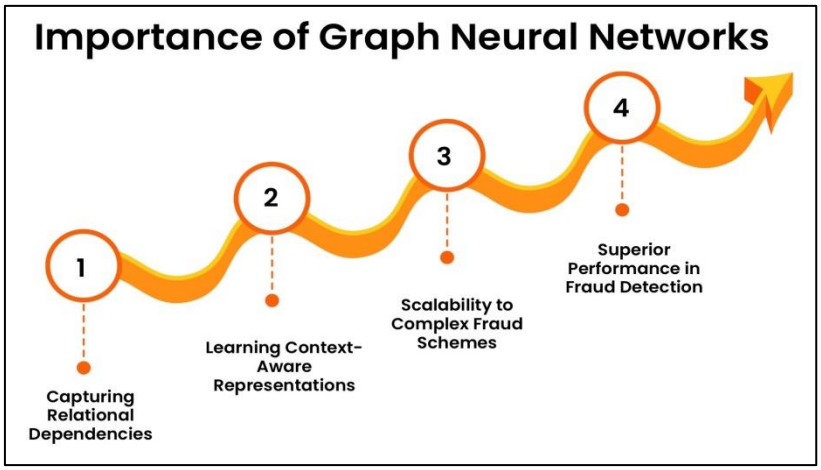

Fig 1: Importance of Graph Neural Networks

- **Capturing Relational Dependencies:** Conventional machine learning models do not consider all the associations between entities by treating each account or transaction as an independent point in the data. By contrast, GNNs have a strong ability to capture relational dependencies, as they represent the relationships in a dataset as a graph, verifying its nodes (such as users, accounts, or devices) and edges through interactions like transactions or logins. This relational modeling is critical in synthetic identity detection because fraudulent identities are commonly in possession of shared resources such as devices, IP addresses or social assets that only emerge when studied in the context of a network.
- **Learning Context-Aware Representations:** The idea of aggregating information about adjacent nodes is one of the strongest advantages of GNNs, as it enables the formation of context-aware representations. An example is that a single account can look clean on its own, and, once a connection is made to other suspicious nodes where articles were accessed on the same device or there is a large number of transactions at unexpected or uncharacteristic times, then the context of the very single account changes and poses a risk. GNNs can reveal these underlying patterns by repeatedly relaying information throughout the graph, making the process a more robust and comprehensive way of perceiving fraudulent activity.
- **Scalability to Complex Fraud Schemes:** Synthetic identity fraud often exhibits long-term dependencies between or among various entities, such as networks of accounts with interactions on multiple time scales. In contrast to shallow or rule-based models, GNNs can scale their models to connect multifaceted fraud schemes across both local and global graph structures. This capability allows them to detect fraud rings and coordinated attacks, which conventional systems often fail to detect.
- **Superior Performance in Fraud Detection:** Experiments on various models, including scanning credit card frauds, insurance claims, and internet payment frauds, have repeatedly demonstrated that GNNs outperform other models in terms of accuracy, recall, and overall robustness. By combining structural and attribute-level features, they possess the undeniable advantage of synthetic identity detection, as complex patterns and implicit correlations are crucial to the success of such systems.

2. Literature Survey

2.1. Traditional Detection Methods

Historically, synthetic identity detection has been largely dependent on rule-based, supervised learning, and anomaly detection technologies. Rule-based techniques are one of the oldest and simplest, involving the application of predefined threshold values or patterns, such as a frequency of transactions, shifting usage, and an exceptionally short account lifespan.

[6-9] They are easy to implement and interpret, and quite rigid and may not follow the evolving fraud schemes and advanced synthetic identity approaches. Supervised learning techniques, such as logistic regression, decision trees, random forests and support vector machines, have been utilized in general terms of classifying accounts as either fraudulent or genuine. These models also utilize labeled data and structured data (account activity, demographic data and transaction patterns). Although very effective with well-defined and feature-rich data, these methods fail to model complex interrelationships between different entities and, as such, cannot detect the synthetic identities that tend to have many accounts or interactions.

2.2. Graph-Based Approaches

Due to the drawbacks of conventional techniques, graphical methods have become highly popular in detecting fraud. These methods represent users, accounts, and transactions as nodes and links in a graph, allowing the identification of intricate patterns and interconnections that are otherwise difficult to discern. Graph embeddings, link prediction and community detection are frequently applied to detect anomalies and suspicious groups of activity. Other graph-structure preservative algorithms, e.g., node2vec, produce low-dimensional representations of graph nodes, enabling better similarity and anomaly analysis. Equally, Graph Convolutional Networks (GCNs) utilise the node attributes and topology of a graph in their message passing, whereby various pieces of information are aggregated from neighbours to detect minor and decentralised patterns of fraud. Such methods are especially well-suited to identifying relational dependencies and, as such, may be effective at mapping synthetic identities with coordinated or networked behaviours.

2.3. Applications of GNNs in Fraud Detection

Graph Neural Networks (GNNs) represent an extension to general graph analysis that allows to learn end-to-end on input graph-structured data with both node and edge features. In fraud detection, GNNs have proven useful in various fields, including credit card fraud, insurance premium fraud, and Internet payment fraud. As the nodes that provide information are neighbors to each other, GNNs can identify very fine and contextual anomalies, hidden correlations and model relationships that classical machine learning models might miss. For example, GNN can detect a suspicious group of transactions that share common devices, IP addresses, or mediators, which might otherwise seem harmless when viewed in isolation. The distributed nature of GNNs has repeatedly demonstrated their superiority to classic models in environments where fraud is exhibited as a coordinated effort or organisation between accounts. With the ability of GNNs to seamlessly integrate structural graph features and attribute-level features, they show a clear advantage in environments where fraud is highly networked.

2.4. Research Gap

Although the effectiveness of graph-based methods and GNNs in detecting transaction-level fraud has been demonstrated, their limited use in detecting synthetic identity fraud remains a concern. Synthetic identity fraud is necessarily more elaborate than ordinary transactional fraud, involving multiple accounts, falsified personal data, and cross-entity interdependencies that span long periods of time. Components. The existing studies tend to focus on short-term anomalies in transaction behaviour without considering multi-entity relations, as well as time series characteristics of synthetic identities. This space reveals a requirement for specialised graph-based models that can model such complex dependencies. The limitation of current technology that our approach overcomes is that a GNN-based methodology enables us to detect coordinated, multi-entity fraud that would otherwise be difficult or impossible to detect within a large dataset.

3. Methodology

3.1. Data Collection and Preprocessing

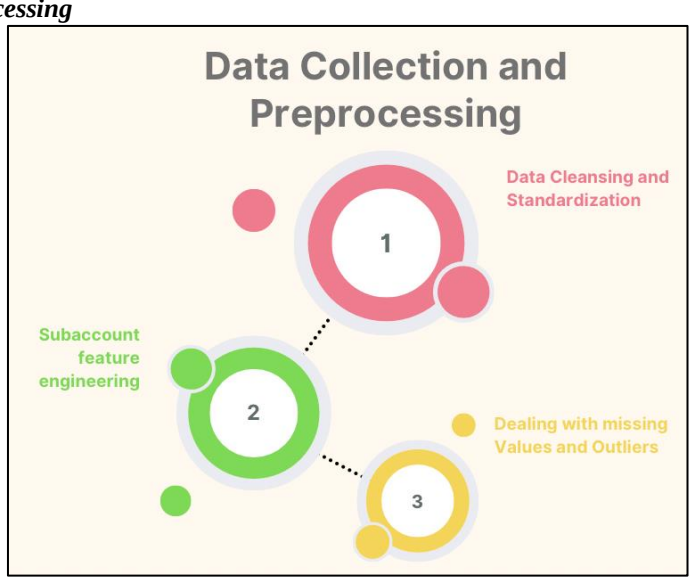

Fig 2: Data Collection and Preprocessing

The initial stage of synthetic identity detection involves extracting corresponding information using various systems. Records on transactions provide information on financial transactions made by customers, such as deposits, withdrawals, transfers, and purchases, which can reveal abnormal trends that may be signs of fraud. [10-12] The information in the user profile, such as demographical information, date of account creation and the attributes of identity, can be used to identify real accounts versus synthetic ones. Additionally, social connections and network data, such as relationships with other accounts, shared devices, or IP addresses, provide valuable context to identify coordinated fraud. Combining these various sources of data enables the model to have a comprehensive picture of user activities and potential anomalies.

- **Data Cleansing and Standardisation: Data cleaning and normalisation are essential processes to en**sure data quality and consistency. Cleaning involves the process of eliminating duplicates, correcting user data errors, and removing irrelevant and corrupted data. Normalization transforms numeric characteristics, e.g. amounts of transactions or account balances, to a comparable range so that larger-scaled features cannot disproportionately impact the process. Such measures aid in noise reduction, model convergence and the comparison of accounts and transactions across the board.
- **Subaccount feature engineering:** Feature engineering converts the original data into valuable inputs for the model. Examples of feature engineering in synthetic identity detection include the frequency of transactions, average transaction balance, account age, device usage, and network connectivity. Ratio, time trend, or clustering scores are examples of derived features that can point to hidden patterns that denote synthetic or coordinated behavior. The features chosen to run the model are well-designed to discriminate between the two accounts. This is because the model will perform better in detecting fraudulent accounts.
- **Dealing with Missing Values and Outliers: To avoid bias and enhance the robustness of a model, it is crucial to handle missing values and outliers effectively**. Missing values may exist as a result of incomplete records or user input errors and are usually corrected using imputation methods, which include mean substitution and k-neighbour methods. Transactions that introduce more outliers, e.g. unusually large transactions or excessive activity spikes, may indicate either real anomalies or fraud. Close attention to the presence and treatment of outliers (by capping, transforming outliers, and flagging) prevents an inaccurate model from being taught something different based on anomalous data points.

3.2. Graph Construction

To best represent the relationship between entities in synthetic identity detection, the dataset is treated as a heterogeneous graph. A heterogeneous graph is useful for building a complex model of interactions between users, accounts, devices, and IP addresses by enabling various types of nodes and edges. This representation is useful for identifying coordinated activities, concealed relationships, and trends that are easily undetectable in traditional tabular forms of information alone. The given framework can utilise both relational and attribute-based information to identify synthetic identities that are explicitly represented in graph form.

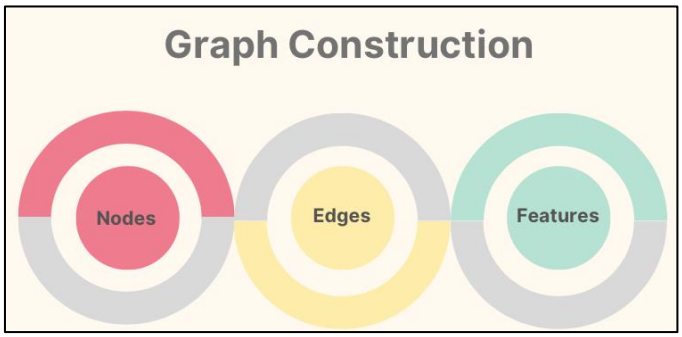

Fig 3: Graph Construction

- **Nodes:** The nodes in the graph symbolize the major objects operating in the system. These are described as users, accounts, devices, and IP addresses. Each type of node records a different dimension of the ecosystem: user nodes represent demographic and personal data, account nodes represent financial or service accounts, device nodes represent hardware identifiers, and IP nodes provide network contextual information. Modeling them as nodes allows the system to identify indirect connections and malicious patterns between different entities, multiple accounts on the same device/IP address.
- **Edges:** Edges define communication or relations between nodes and are important to model any behavioral patterns. Edges represent transactions between accounts, logins between users and accounts, or social relationships, such as similar contacts or affiliations. Timestamps and the amount of a transaction are two examples of attributes that give a graph a time dimension and a quantitative dimension, allowing the graph to distinguish not only with whom one interacts but also the nature and strength of that interaction. These edges are useful in disclosing orchestrated engagement common to synergetic identity schemes.

- **Features:** Both nodes and edges are augmented with features that characterise their nature. Features may include demographics, account age, and historical interaction levels, while edge features may contain information such as transaction amount, time/date, frequency, or type of interaction. Integrating such features within the graph enables the training of a Graph Neural Network to create meaningful representations that incorporate both structural and attribute-level data. This enables the model to uncover some of the mild correlations as well as anomalies that can help identify synthetic identities that might not be detected through traditional techniques.

3.3. Graph Neural Network Architecture

Graph Neural Networks (GNNs) are designed to leverage both the features of nodes and the graph structure to capture meaningful representations. [13-15] This is done through a Graph Convolutional Network (GCN) that propagates information along edges of connected entities to identify synthetic identities. The CNs will be especially appropriate since this will enable them to capture complex and context-sensitive relationships, which are suggestive of fraud. The architecture has an input layer, several graph convolution layers and an output layer that is customized based on a fraud probability prediction.

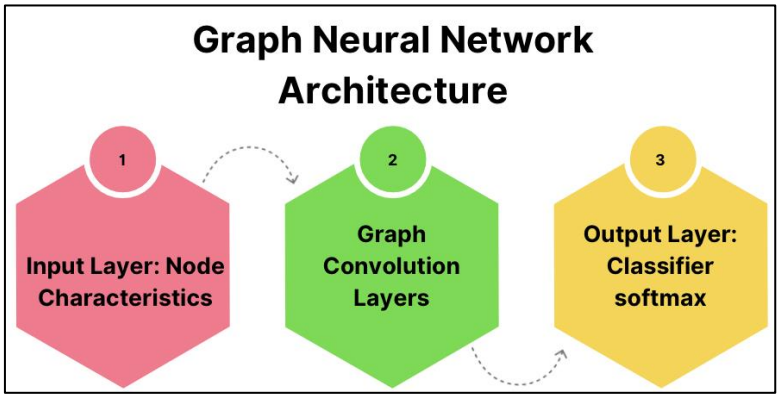

Fig 4: Graph Neural Network Architecture

- **Input Layer: Node Characteristics:** The first layer of GCN uses the node features as the initial representation. ACLs are used to provide explicit control of what nodes can do. Each node in the graph —users, accounts, devices, or IPs —is assigned characteristics in terms of demographics, account ages, activity history, and device usage. The network then uses these characteristics as input data, providing the other layers with sufficient context to learn how normal and suspicious activity occurs. This input layer encodes rich node-level information on which message passing and feature aggregation operations in the graph are based.
- **Graph Convolution Layers:** The heart of the GCN is graph convolution layers, in which the individual nodes update their node embedding by aggregating over their neighbors. The update rule is shown as

$$h_v^{(k+1)} = \sigma\left(\sum_{u \in N(v)} \frac{1}{c_{vu}} W^{(k)} h_u^{(k)} \right)$$

Here, $h_v^{(k)}$ Is the feature vector of a network node v at the level K, N(v) are the nodes adjacent to node v. $W^{(K)}$ is the weight matrix to be trained at the level K, and σ is an activation function such as ReLU. This integration enables the model to recognise the structures and relationships surrounding each node, thereby providing both direct and indirect relationships. The GCN can operate on more than a single layer and learn long-range dependencies, which is essential to capture synthetic identities across obstacles, devices, and accounts.

- **Output Layer: Classifier softmax:** The last section is a layer of softmax that adopts a classifier that predicts the chances of a node (account or user) being signed up as a fraud. Finally, the node representations after several graph convolution layers have neighbourhood information aggregated to them, and both structural and attribute patterns can be generalized. These learned embeddings are converted to probabilities using the softmax function, which allows for clear differentiation between real and fake identities. This results in effective synthetic identity detection because the GCN end-to-end is learning to assign high probabilities to nodes that are likely to be fraudulent, while normal accounts have a low probability under the training data.

3.4. Training and Evaluation

The training and evaluation of the Graph Neural Network are essential procedures to provide proper synthetic identity detection. This procedure entails a feasible loss function as well as selecting an optimization plan together with metrics of performance. The model can be trained to acquire meaningful representations of nodes and minimize the errors, but evaluation quantifies how well it can distinguish between synthetic and real accounts.

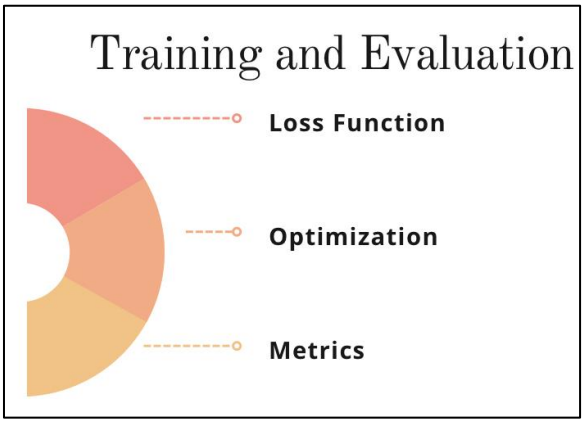

Fig 5: Training and Evaluation

- **Loss Function:** The loss function is used to measure the deviations between the resultant probabilities of fraud and the exactly annotated labels. In binary classification tasks, e.g. synthetic identity detection, the binary cross-entropy loss is often employed. This loss punishes more for misprediction, with a greater penalty on mispredictions that are distant. Reduction of the loss on training motivates the network to give correct probability estimations so that fraudulent accounts get high (predicted probabilities) and genuine accounts get low probabilities.
- **Optimization:** Optimization is the algorithm of adjusting the weights as a network is trained. Stochastic Gradient Descent (SGD) or gradient-based optimizers, including Adam, are commonly used to minimize the loss function. The optimizers tune the weights of the GCN layers through methods that calculate gradients to the loss with regard to parameters and iteratively adjust them. Learning rate scheduling and weight regularization can also enhance faster convergence and avoid overfitting, making them generalize well to unseen data.
- **Metrics:** The performance measures gauge the trainability of the sensitive model in recognizing synthetic identities. Standard measures are accuracy, precision, recall, F1-score and area under the Receiver Operating Characteristic curve (AUC-ROC). Precision indicates how accurately predicted fraudulent accounts are in reality, and recall indicates how many actual fraudulent accounts are correctly classified. The F1-score combines precision and recall in a balanced measure, and AUC-ROC illustrates the model's potential to classify classes at different thresholds. The events defined in these metrics give the model a holistic evaluation in real-life fraud detection situations.

3.5. Feature Set Description

Regarding synthetic identity discrimination, feature selection and feature design are of critical importance in encompassing behavioral as well as structural dynamics, which distinguish between authentic accounts and fake ones. [16-18] the first product is the Account Age, and it reflects how long an account has been in existence, usually expressed in terms of days. Previous accounts are usually more credible than new ones, because they have a history of transactions and they have a track record of behavior. New accounts are more likely to cause concern, especially when they are enrollments in a synthetic identity where the number of accounts is created quickly. Transaction Count is another important functionality that measures the number of transactions an account is connected to. This metric shows the overall levels of user activity for each account; abnormal numbers of transactions per account age can be a hint of abnormal activity.

Accounts that have been used to commit synthetic identity fraud typically exhibit anomalous transaction patterns, with either intense periods of transactions to replicate an authentic customer experience or rare periods of transactions to avoid detection. Device Type gets the type of device used to log in to the account, including mobile phones, tablets, or desktops. Telemetry evidence of fraudulent accounts can reveal signs of device sharing or of switching devices regularly, which may also indicate multiple synthetic accounts being accessed with a single device. Social Links quantify the size of linked objects, friends, accounts, or contact networks. Synthetic identities have fewer natural social relationships or unnatural groupings than real users, which can make this feature a significant source of information for use in graph-based detection. Finally, Average Transaction Amount presents the average Transaction value. Deviations of regular transaction size, especially in combination with other attributes like the number of transactions and social or interpersonal links, will raise suspicions. Integrating these features allows the model to model both individual account features and relationships amongst the accounts comprising the network, modeling the dynamic and realistic model necessary to determine synthetic identities.

4. Results and Discussion

4.1. Dataset Description

Synthetic identity detection. Such data contains a dense amount of information to represent the diversity of synthetic fraud, including information on user profiles, transaction records, and linkages between accounts and devices. Customer

information typically includes demographic data, such as age, gender, and location, as well as account-related attributes, including the date of account creation, verification stage, and historical activity. This information is vital in developing baseline behavioral trends and the development of deviations that would signify that synthetic identities may be present. Transaction records generate in-depth information about financial or service-related activity, including attributes such as transaction amount, timestamp, transaction type and frequency. Patterning of activity in these transactions, such as bursts of activity or recurrent patterns of transactions, can be very powerful clues to possible fraud. Additionally, the dataset provides connection data, including relationships between various entities such as shared devices, IP addresses, and social/network-based affiliations. Such relationships are especially vital to graph-based methods, as they enable the model to infer relationships and interactions that are not apparent in the disaggregated accounts. By transitioning these dependencies to edges in a heterogeneous graph format, the framework can take advantage of both structural and attribute-based patterns in identifying coordinated or collusive activity. All the benchmark datasets were thoroughly selected to contain both actual and synthetic accounts, ensuring they properly represent the level of fraud prevalence and allowing for adequate testing of the detection approaches. They are commonly characterized by a mixture of high and low interconnection densities, different account age distributions and various transactor behaviours, to reflect real-world pursuits. Such high-quality and multidimensional data can be used to extensively test the GNN model and determine whether it can generalise across patterns of synthetic identity fraud. Generally, the data set is useful in designing and testing graph-based methods of detection that can successfully detect stealthy, compassional fraud designs.

4.2. Experimental Results

The experimental assessment involves comparing the work of traditional machine learning models, including Logistic Regression and Random Forest, with the proposed Graph Convolutional Network (GCN) under conditions of artificial identity detection. The findings show that the combination of graph-based structural information and node features outperforms all baselines in terms of all measures considered.

Table 1: Experimental Results

Model	Accuracy	Precision	Recall	F1-score	ROC-AUC
Logistic Regression	0.85	0.76	0.68	0.72	0.81
Random Forest	0.88	0.79	0.71	0.75	0.84
GCN	0.94	0.88	0.85	0.86	0.91

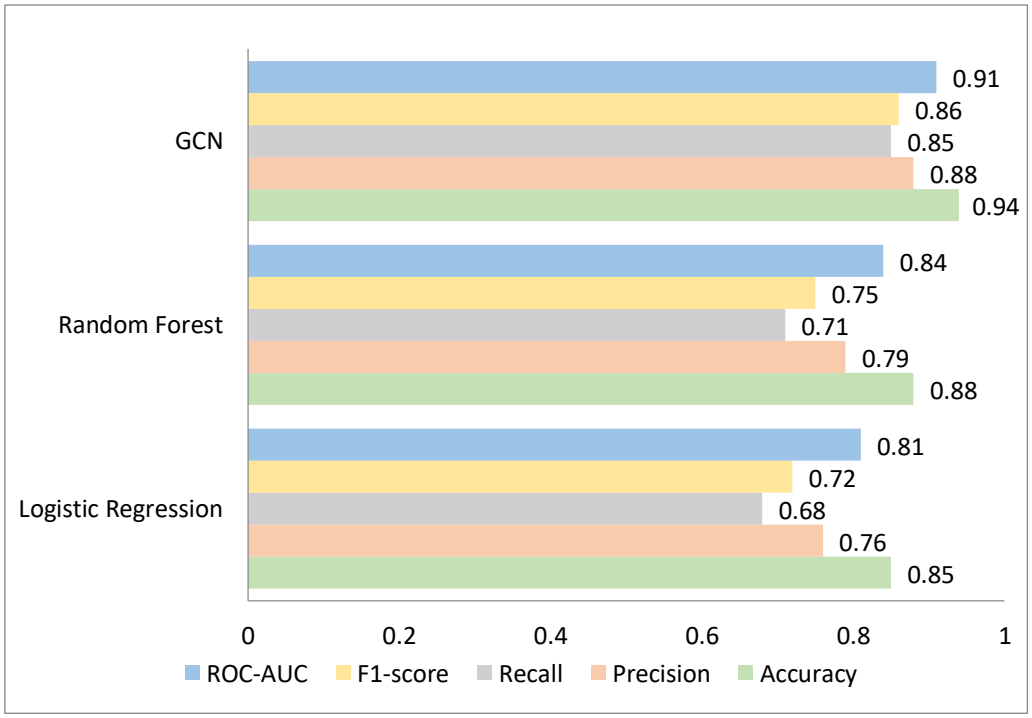

Fig 6: Graph representing Experimental Results

- **Logistic Regression:** The Logistic Regression module had an accuracy of 85 percent, precision, recall, F1-score, and the ROC-AUC of 76, 68, 72, and 81, respectively. Logistic Regression is a linear classifier that works well on structured fields but struggles to find more complex relationships between accounts, devices, and IP addresses. Its low recall rate indicates that a significant number of synthetic identities were not captured, demonstrating its shortcomings

with respect to predictive efficiency in identifying interdependent trends more characteristic of grouped criminal activities.

- **Random Forest:** Random Forest outperformed Logistic Regression with an accuracy of 0.88, precision of 0.79, recall of 0.71, F1-score of 0.75 and ROC-AUC of 0.84. Random Forest combines the advantages of non-linear models and feature combination. Although it detects more fraudulent accounts compared to Logistic Regression, it does so on an entity-by-entity basis. It thus lacks the power to establish fraud that occurs as a result of the interdependence of entities.
- **Graph Convolutional Network (GCN):** The proposed GCN model outperformed the customary methods, achieving accuracies of 0.94, 0.88, 0.85, 0.86, and 0.91, as well as corresponding precision, recall, F1-score, and ROC-AUC values of 0.88, 0.85, 0.86, and 0.91, respectively. The GCN has been able to capture structural dependencies and long-range correlations within the graph through the aggregation of information by neighbouring nodes and the integration of attribute information on nodes and edges. This enables it to detect two factors: synthetic identities that show subtle yet synchronised patterns across multiple accounts, devices, and social links. The results show clearly that graph-based learning performs well compared to the conventional methods when dealing with a complex fraud detection system.

4.3. Discussion

As seen in the experiment results, the GCN-based model drastically outperforms classical machine learning frameworks in the identification of synthetic identities, particularly in terms of parameter recall and F1-score. High recall is very critical in fraud detection, as it is based on the model's capability to discover a high proportion of genuine fraudulent accounts, rather than its overall accuracy, which is often more important in security-sensitive applications. The high performance of the GCN compared to more basic models (Logistic Regression and Random Forest) suggests its ability to retrieve even minor and sophisticated atypical and suspicious patterns that most models fail to capture. Displaying a contrast to ordinary methodologies that primarily rely on individual accounts' characteristics, the GCN utilises graph topology, which enables the integration of neighbourhood conditions across the associated nodes in the graph. This is of special interest to the synthetic identity fraud prediction, where, in order to appear legitimate, fraudulent accounts often share devices, IP addresses or social connections. By pooling data from adjacent nodes and edges, the GCN is in a position to detect groups of suspicious accounts with correlated activities, e.g., repeated logins on the same machine or between transactions involving accounts that do not seem related but are actually under the same controller.

Additionally, the model's capability to aggregate features at a node along the pattern facilitates the detection of anomalies that may otherwise go unnoticed, even when considering single accounts. This end-to-end learning method enables the network to automatically retrieve the most significant features and relationships to detect fraud and make the feature engineering less dependent. The improvement in F1-score also proves that the model not only provides high precision but also a high level of recall, meaning that it can successfully detect the accounts that commit fraud but also doesn’t leave too many false positives, which is paramount in practical deployment to ensure that users of good faith will not be report to authorities due to a mistake of the model. Taken together, the results indicate the strength of graph-based learning in considering complex and multi-entity interactions, and hence GCNs are a viable solution for implementing synthetic identity detection in dynamic and interconnected datasets.

4.4. Limitations

Although the setup using the GCN framework performs very well in identifying synthetic identities, some drawbacks should be noted. Among the main difficulties is the heavy computational cost of training and making inferences on large graphs. The memory and processing would be more and more heavy as the number of nodes and edges grows, especially in the neighbourhood aggregation of the graph convolution layers. This can constrain the scalability of the model and its inability to apply it in real-time or work with very large datasets, where there are millions of users, transactions, and interconnections. Such problems can be remedied by using sampling, graph bisection, or hierarchical modelling, which may introduce approximation errors or require additional engineering work. Another weakness is the reliance on input data quality and completeness. Graph-based models will focus on node and edge features, aiming to capture interesting patterns. Missing, noisy, or inconsistent data can negatively affect the network's capability to learn accurate representations, possibly resulting in skipped detections or false positives. The model may not be as effective, as there may be incomplete social connection data and incorrect transaction records that can blur connections between synthetic accounts. Obtaining high-quality data collection and preprocessing is, thus, a critical requirement; however, in most real-life settings, the acquisition of comprehensive and accurate data may be difficult due to privacy-related limitations, system errors, or fragmented data sources.

The other weak point is the limited interpretation of deep GNN layers. Since the network combines the data provided at lower levels, it is not easy to justify particular predictions. This black box can be a barrier to faith and adoption in regulatory or high-stakes applications where transparency is crucial. In contrast to more basic forms of modelling, such as decision trees or logistic regression, where the relative importance of features is easily interpreted, deep GNNs require specially designed tools, like attention layers, or post-hoc explanatory methods to understand how decisions are made. As a result, GCNs may yield

better predictive performance; however, addressing computation, data quality, and interpretability issues is a priority for practical and responsible application in synthetic identity detection.

5. Conclusion

Synthetic identity fraud has proved to be one of the most advanced and growing issues in the financial and digital environment. Unlike the more conventional frauds, which typically involve the direct abuse of stolen identities (either through their spendable credentials or overt transactions), synthetic identities have been created using a combination of real and fake data, making them doubly challenging to detect with conventional fraud-detection techniques. In this work, we responded to the increasing demand for sophisticated detection techniques by proposing a Graph Neural Network (GNN)- based framework to capture identities, transactions, and their interactions over the graph. Presenting the user, accounts, devices, and IP addresses as nodes and updating the relationships between them, such as transactions, logins, and social connections, as edges, the proposed framework was able to better capture the individual profiling requirements and the connections among them, allowing a more comprehensive approach to fraud detection. The practical test revealed that the GNN model performed considerably well compared to traditional machine learning models, such as Logistic Regression and Random Forest. Although traditional methods are fairly effective when fed structured tabular data, they do not perform well with the complicated and obscure associations that comprise synthetic identity schemes. However, the GNN found, through neighbourhood aggregation and feature propagation, the delicate patterns of behavior, e.g., the ability to align activities across nodes or more accounts sharing devices.

The above trends in terms of mostly significant performance gains in recall and F1-score were particularly relevant in the context of fraud detection, where recall indicates the capacity to detect fraudulent parties that are balanced against false positives, as depicted in F1-score. The results highlight the utility of graph-based learning to practical fraud problems, which are relational in nature. Although it shows positive results, the study also cites numerous limitations, such as computational complexity on very large-scale graphs, absolute dependence on high-quality and complete datasets, and the challenge of interpreting topics in deep GNN architecture. These weaknesses indicate significant refinements that need to be made to make them more scalable, robust, and trustworthy for practical deployment in the sector. Future work will aim at applying this framework to broader tasks by embracing heterogeneous graph neural networks (HetGNNs), which will have improved capabilities to deal with variability in types of nodes and edges; temporal graphs, which will incorporate time-evolutionary behavior to capture dynamic fraud patterns, and online real-time detection capabilities, which are necessary to implement into financial systems and online platforms where real-time detection is vital. This paper demonstrates that graph-based deep learning is an effective and flexible approach to the evolving and challenging issue of synthetic identity detection. GNNs also offer a way forward toward developing more intelligent, relational, and structurally based fraud detection systems, as they suggest an extension beyond individual feature analysis and into relational, structural modeling, which poses a promising path to creating models that can adapt to emerging challenges.

References

[1] “Alleviating the Inconsistency Problem of Applying Graph Neural Network to Fraud Detection” — Zhiwei Liu, Yingtong Dou, Philip S. Yu, Yutong Deng, Hao Peng. May 2020.

[2] Pourhabibi, T., Ong, K. L., Kam, B. H., & Boo, Y. L. (2020). Fraud detection: A systematic literature review of graph-based anomaly detection approaches. Decision Support Systems, 133, 113303.

[3] You, J., Gomes-Selman, J. M., Ying, R., & Leskovec, J. (2021, May). Identity-aware graph neural networks. In Proceedings of the AAAI conference on artificial intelligence (Vol. 35, No. 12, pp. 10737-10745).

[4] Zhang, W., Shu, K., Liu, H., & Wang, Y. (2019). Graph neural networks for user identity linkage. arXiv preprint arXiv:1903.02174.

[5] “Enhancing Graph Neural Network-based Fraud Detectors against Camouflaged Fraudsters (CARE-GNN)” — Yingtong Dou, Zhiwei Liu, Li Sun, Yutong Deng, Hao Peng, Philip S. Yu. August 2020.

[6] Boyaci, O., Umunnakwe, A., Sahu, A., Narimani, M. R., Ismail, M., Davis, K. R., & Serpedin, E. (2021). Graph neural networks-based detection of stealth false data injection attacks in smart grids. IEEE Systems Journal, 16(2), 2946-2957.

[7] Shen, J., Zhou, J., Xie, Y., Yu, S., & Xuan, Q. (2021, August). Identity Inference on Blockchain Using Graph Neural Networks. In International Conference on Blockchain and Trustworthy Systems (pp. 3-17). Singapore: Springer Singapore.

[8] Latchoumi, T. P., & Kannan, V. V. (2013). Synthetic Identity of Crime Detection. International Journal, 3(7), 124-129.

[9] Park, N., Kan, A., Dong, X. L., Zhao, T., & Faloutsos, C. (2019, July). Estimating node importance in knowledge graphs using graph neural networks. In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining (pp. 596-606).

[10] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Yu, P. S. (2020). A comprehensive survey on graph neural networks. IEEE transactions on neural networks and learning systems, 32(1), 4-24.

[11] Llugiqi, M., & Mayer, R. (2022, August). An empirical analysis of synthetic-data-based anomaly detection. In International Cross-Domain Conference for Machine Learning and Knowledge Extraction (pp. 306-327). Cham: Springer International Publishing.

[12] Akoglu, L., Tong, H., & Koutra, D. (2015). Graph-based anomaly detection and description: a survey. Data Mining and Knowledge Discovery, 29(3), 626-688.
[13] Yanushkevich, S., Stoica, A., Shmerko, P., Howells, W., Crockett, K., & Guest, R. (2020, July). Cognitive identity management: Synthetic data, risk and trust. In 2020 International Joint Conference on Neural Networks (IJCNN) (pp. 1-8). IEEE.
[14] Stanimirova, I., Daszykowski, M., & Walczak, B. (2007). Dealing with missing values and outliers in principal component analysis. Talanta, 72(1), 172-178.
[15] Detect financial transaction fraud using a Graph Neural Network with Amazon SageMaker, AWS, 2022. online. https://aws.amazon.com/blogs/machine-learning/detect-financial-transaction-fraud-using-a-graph-neural-network-with-amazon-sagemaker/
[16] You, J., Leskovec, J., He, K., & Xie, S. (2020, November). Graph structure of neural networks. In International Conference on Machine Learning (pp. 10881-10891). PMLR.
[17] Zhang, L., Song, H., Aletras, N., & Lu, H. (2018). Graph node-feature convolution for representation learning. arXiv preprint arXiv:1812.00086.
[18] Xu, X., Liu, C., Feng, Q., Yin, H., Song, L., & Song, D. (2017, October). Neural network-based graph embedding for cross-platform binary code similarity detection. In Proceedings of the 2017 ACM SIGSAC conference on computer and communications security (pp. 363-376).
[19] Kamath, C. N., Bukhari, S. S., & Dengel, A. (2018, August). Comparative study between traditional machine learning and deep learning approaches for text classification. In Proceedings of the ACM Symposium on Document Engineering 2018 (pp. 1-11).
[20] Wang, P., Fan, E., & Wang, P. (2021). Comparative analysis of image classification algorithms based on traditional machine learning and deep learning. Pattern recognition letters, 141, 61-67.
[21] Pappula, K. K., & Anasuri, S. (2020). A Domain-Specific Language for Automating Feature-Based Part Creation in Parametric CAD. International Journal of Emerging Research in Engineering and Technology, 1(3), 35-44. https://doi.org/10.63282/3050-922X.IJERET-V1I3P105
[22] Rahul, N. (2020). Optimizing Claims Reserves and Payments with AI: Predictive Models for Financial Accuracy. *International Journal of Emerging Trends in Computer Science and Information Technology*, *1*(3), 46-55. https://doi.org/10.63282/3050-9246.IJETCSIT-V1I3P106
[23] Enjam, G. R. (2020). Ransomware Resilience and Recovery Planning for Insurance Infrastructure. *International Journal of AI, BigData, Computational and Management Studies*, *1*(4), 29-37. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V1I4P104
[24] Pappula, K. K., Anasuri, S., & Rusum, G. P. (2021). Building Observability into Full-Stack Systems: Metrics That Matter. *International Journal of Emerging Research in Engineering and Technology*, *2*(4), 48-58. https://doi.org/10.63282/3050-922X.IJERET-V2I4P106
[25] Pedda Muntala, P. S. R., & Jangam, S. K. (2021). End-to-End Hyperautomation with Oracle ERP and Oracle Integration Cloud. *International Journal of Emerging Research in Engineering and Technology*, *2*(4), 59-67. https://doi.org/10.63282/3050-922X.IJERET-V2I4P107
[26] Enjam, G. R. (2021). Data Privacy & Encryption Practices in Cloud-Based Guidewire Deployments. *International Journal of AI, BigData, Computational and Management Studies*, *2*(3), 64-73. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V2I3P108
[27] Rusum, G. P. (2022). WebAssembly across Platforms: Running Native Apps in the Browser, Cloud, and Edge. *International Journal of Emerging Trends in Computer Science and Information Technology*, *3*(1), 107-115. https://doi.org/10.63282/3050-9246.IJETCSIT-V3I1P112
[28] Pappula, K. K. (2022). Containerized Zero-Downtime Deployments in Full-Stack Systems. International Journal of AI, BigData, Computational and Management Studies, 3(4), 60-69. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V3I4P107
[29] Jangam, S. K., & Karri, N. (2022). Potential of AI and ML to Enhance Error Detection, Prediction, and Automated Remediation in Batch Processing. *International Journal of AI, BigData, Computational and Management Studies*, *3*(4), 70-81. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V3I4P108
[30] Pedda Muntala, P. S. R. (2022). Natural Language Querying in Oracle Fusion Analytics: A Step toward Conversational BI. *International Journal of Emerging Trends in Computer Science and Information Technology*, *3*(3), 81-89. https://doi.org/10.63282/3050-9246.IJETCSIT-V3I3P109
[31] Rahul, N. (2022). Automating Claims, Policy, and Billing with AI in Guidewire: Streamlining Insurance Operations. *International Journal of Emerging Research in Engineering and Technology*, *3*(4), 75-83. https://doi.org/10.63282/3050-922X.IJERET-V3I4P109
[32] Enjam, G. R., & Tekale, K. M. (2022). Predictive Analytics for Claims Lifecycle Optimization in Cloud-Native Platforms. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *3*(1), 95-104. https://doi.org/10.63282/3050-9262.IJAIDSML-V3I1P110