International Journal of Computational and Experimental Science and ENgineering (IJCESEN)

Vol. 11-No.3 (2025) pp. 7003-7012

http://www.ijcesen.com

ISSN: 2149-9144

Research Article

Context-Aware Access Monitoring with Tracking and Alerting for Cloud-Based Invoice Processing in Financial Automation

Ranadheer Reddy Charabuddi*

Sr. Lead SAP Opentext VIM Consultant, Avventis Inc, Northlake, Texas, USA

***Corresponding Author Email** : ranadheer.pro@gmail.com - **ORCID**: 0009-0000-4842-6148

Article Info:

DOI: 10.22399/ijcesen.3954
Received : 18 July 2025
Accepted : 20 September 2025

Keywords

Risk Management
SwAT Graph Toeplitz Convolutional Networks (SGTCNs)
Cloud-based invoice processing
Tsallis Kahn's Directed Acyclic Graph (TK-DAG)
Financial Automation
Context-aware Access Monitoring

Abstract:

The proactive strategy of detecting vulnerabilities and ensuring secure processing of invoices in the cloud is termed Risk Management (RM). Yet, the existing studies didn't automatically adjust the access privileges according to contextual risk along with tracking and alerting, affecting the model. Thus, this paper presents context-aware access monitoring with tracking and alerting for cloud-based invoice processing in financial automation using Parabolic Ramp Fuzzy (PR-Fuzzy). Primarily, for accessing the invoice, the admin creates the role and access policy. Next, by using Polyphase Lifting Discrete Wavelet Transform (PLDWT), the invoice data is watermarked. Afterward, a digital signature is created for the watermarked invoice data based on the Inverse Deterministic Salt-based Digital Signature Algorithm (IDS-DSA). The digital signature is verified at the destination side, and watermarking is removed for editing the invoice. The attributes are extracted using Optical Character Recognition (OCR) from the edited invoice. Next, by using SwAT Graph Toeplitz Convolutional Networks (SGTCNs), invoice fraud detection is performed. The access is blocked if fraud is detected; otherwise, the edited invoice is watermarked, authorised, and sent to another department. Lastly, it reaches the user. Similarly, by using PR-Fuzzy, the context-aware access monitoring is performed. The access is blocked if high risk is obtained, and an alert is sent to the admin. Similarly, by using Exponential-GINI-BLOOM based Merkle-Tree (EGB-MT), the tree is constructed with invoice accessing information and then shared with the admin for tracking and monitoring. As per the results, the proposed model took a lesser rule generation time of 1024ms.

1. Introduction

Cloud-based invoice processing plays a crucial role in handling large volumes of financial documents in the era of financial automation[18]. Invoice documents are linked to the financial part of daily activities and are received every day. Also, any error in the invoice is not acceptable[2]. Yet, this shift to cloud environments presents major risks regarding unauthorized access and system reliability[20]. To prevent financial losses and ensure the continuity of operations, effective RM is important. Lately, for improved RM, many advanced techniques have been developed in cloud-based invoice processing[16][17].

Usually, to extract the key information from invoices, OCR and neural networks were employed[9]. Similarly, the Artificial neural network and support vector machine were used for effective financial fraud detection[10][1]. Likewise, for fraudulent activity detection, some existing studies used CatBoost and GNN-CNN-Long Short Term Memory (GNN-CL)[7][6]. Yet, the existing studies didn't adjust the access privileges regarding the contextual risk along with tracking and alerting. The motivation of the proposed method is to adjust the access privilege based on the low, medium, and high risk access, with tracking behavior. This improves the RM for cloud-based invoice processing in financial automation. Thus, this paper proposes PR-Fuzzy-based context-aware access monitoring and EGB-MT-based tracking and alerting for cloud-based invoice processing.

2. Problem Statement

1. None of the existing works focused on designing a model that automatically adjusts access privileges regarding contextual risk along with tracking and alerting, thus affecting real-time RM.

2. The conventional Aung et al., failed to analyze the invoice data flow and relationships between the user and vendor.
3. The existing Bisetty et al., didn't verify whether the authorized persons were accessing the invoice data or not.
4. Most of the prevailing works failed to prevent the invoice from being fake or altered by the intruder.

3. Objective

1. PR-Fuzzy is employed to perform context-aware access monitoring. Also, the invoice activities are tracked by using EGB-MT, and an alert is sent to the admin.
2. TK-DAG-based graph construction and SGTCNs-based invoice fraud detection are done for analyzing the invoice data flow and relationships.
3. IDS-DSA is introduced to verify the authorization of persons accessing the invoice data.
4. PLDWT is established for watermarking the invoice to avoid alteration.
This paper is structured as follows: the literature survey is conveyed in Section 2, the proposed methodology is elucidated in Section 3, the result is illustrated in Section 4, and finally, Section 5 concludes the proposed model with future enhancements.

4. Literature Survey

(Aung et al., 2023) offered an automated invoice processing system. Here, to extract the invoice information, the Convolutional Neural Network (CNN) and RNN techniques were employed. The model minimized the errors. Yet, it didn't analyze the invoice data flow and relationships between the user and vendor.
Bisetty et al., presented an automated invoice verification system. Here, a Machine Learning (ML) algorithm was used for improving the invoice verification process, integration of automated workflows, and data validation processes. The research attained a low processing time. Nevertheless, it failed to verify the authorization.
Ogunmokun et al., recommended a fraud risk mitigation system. Here, to identify the fraudulent activity, ML techniques (i.e., support vector machine, logistic regression, etc) and RM methods were used. The model improved the operational efficiency. Nevertheless, it didn't prevent the invoice from alteration.
Arslan, explored an end-to-end invoice processing system. Here, from the invoice image, text and table were extracted by utilizing You Only Look Once version 5 (YOLOv5) and Tesseract OCR. The model attained high accuracy. Yet, the model might not generalize well across non-standard invoices.
Tang & Liu, advanced a financial fraud detection system. Here, to detect the fraudulent activity, the distributed knowledge distillation algorithm with a neural network was used. The model achieved high performance metrics. Nevertheless, the model had lack of transparency.

Proposed Context-Aware Access Monitoring with Tracking and Alerting for Cloud-Based Invoice Processing in Financial Automation Methodology

Here, to perform context-aware access monitoring, the proposed PR-Fuzzy is established. In Figure 1, the proposed model's diagrammatic representation is shown.

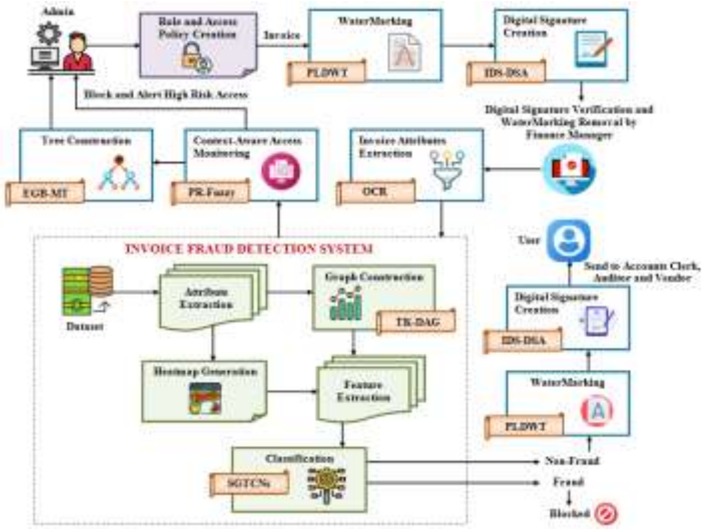

Figure 1. Diagrammatic representation of the proposed model

5. Role and Access Policy Creation

Initially, the admin creates the role and access policy $(A\rho)$ for accessing the invoice data $(I\vartheta_{\partial})$ regarding if-then conditions.

$$A\rho = \begin{cases} if \;\; \Re = \alpha\partial \;\; then \;\; \Im\mu \\ if \;\; \Re = fm \;\; then \;\; A \\ if \;\; \Re = \rho c \;\; then \;\; \zeta n \\ if \;\; \Re = \mu \;\; then \;\; Vl \\ if \;\; \Re = \upsilon \;\; then \;\; \varsigma b \end{cases} \quad (1)$$

Here, $\alpha\partial$ signifies admin, fm denotes finance manager, ρc implies accounts clerk, μ depicts auditor, υ indicates vendor, $\Im\mu$ demonstrates access to create/edit/delete invoices, A exemplifies access to approve/reject, view, and generate invoices, ζn defines access to enter and submit invoices, Vl represents view only access to invoices, and ςb specifies access to submit or view their own invoices.

Watermarking

Then, by employing PLDWT, the invoice data $(I\vartheta_{\partial})$ is watermarked to avoid alteration. Discrete Wavelet Transform (DWT) effectively hides secret data within a cover image. However, DWT involves multiple levels of matrix operations. Therefore, the Polyphase Lifting Scheme is used in DWT and is given as,

$$B_{up} = (I\vartheta_{\partial})_{\xi} + U(I\vartheta_{\partial})_{odd} \quad (2)$$

Here, $(I\vartheta_{\partial})_{\xi}$ specifies the even-indexed sample, B_{up} implies the updated invoice data, U signifies the update function, and $(I\vartheta_{\partial})_{odd}$ specifies the odd-indexed sample. Next, DWT is applied to the B_{up} as,

$$M(B_{up}) = \{ll, lh, hl, hh\} \quad (3)$$

Here, ll signifies the approximation value, and (lh, hl, hh) define the details in different directions. Next, the watermark is embedded in the hl sub-band. The watermarked invoice data is denoted as ϖ_{in}. Likewise, the inverse DWT is employed to remove the watermark from the invoice data.

Digital Signature Creation

Further, the digital signature is created based on IDS-DSA for ϖ_{in} to verify the authorization. Digital Signature Algorithm (DSA) excellently verifies the authenticity of digital documents and messages. Still, the attackers may hack the hash function, resulting in the loss of information. Thus, Inverse Deterministic Salt (IDS) is employed in DSA.

Key Generation

Initially, the public $(\wp)$ and private keys (R) are generated at the sender side.

$$\wp = |0 < \wp < \partial vs| \quad (4)$$

$$R = gn \quad mod \quad N \quad (5)$$

Here, ∂vs signifies the primary divisor, gn indicates the generator, and N represents the prime number.

Signature Generation

Then, (ϖ_{in}) are hashed employing IDS. Here, deterministic salt $(Ds(\varpi_{in}))$ is added to the (ϖ_{in}) and then the inverse operation is performed.

$$\hbar_{fn} = Hh(\varpi_{in} + Ds(\varpi_{in}))^{-1} \quad (6)$$

Here, $\hbar_{fn}$ signifies the hashed watermarked invoice at the sender side, and Hh defines the hash function. Next, the signature (ς) is computed as.

$$\varsigma = (k, \varepsilon) \quad (7)$$

$$k = (gn^{rd} \quad mod \ lp) \quad mod \ sp \quad (8)$$

$$\varepsilon = rd^{-1} \bullet (\hbar + R \cdot k) \quad mod \ sp \quad (9)$$

Where, (k, ε) signify the first and second parts of the signature, respectively, sp indicates the smaller prime number, lp exemplifies the larger prime number, and rd implies the random number. Thereafter, (ϖ_{in}) and (k, ε) are sent to the destination side.

Digital Signature Verification and Watermarking Removal

The digital signature is verified at the destination side (i.e., financial manager, accounts clerk, auditor, and vendor) by using IDS-DSA. Here, the hash is generated for the received (ϖ_{in}) based on the IDS.

$$\hbar_{ve} = Hh(\varpi_{in} + Ds(\varpi_{in}))^{-1} \quad (10)$$

Here, $\hbar_{ve}$ represents the hash generated at the destination side. Then, the modular inverse is estimated for the ε. Next, Z is checked, representing the reconstructed value of k. The signature is verified (VS) as,

$$VS = \begin{cases} if \ Z = k & v \\ if \ Z \neq k & ivl \end{cases} \quad (11)$$

The watermark is removed from the invoice data if the signature is valid (v), and the corresponding department can edit the invoice. The edited invoice document is denoted as K_j. If the signature is invalid (ivl), then the access is blocked.

Invoice Attributes Extraction

Then, the attributes are extracted from K_j by using OCR, which excellently extracts the text from the invoice documents. The extracted invoice attributes are indicated as β_κ.

Invoice Fraud Detection System

The invoice fraud detection is performed based on β_κ, and is explained in the following section.

Dataset

Mainly, the "Financial Fraud Detection" dataset is gathered and is specified as D_s.

Attribute Extraction

The features, including Step, oldbalanceOrg, nameDest, oldbalanceDest, newbalanceDest, and isFlaggedFraud, are extracted from D_s and are signified as ξx_{tt}.

Heatmap Generation

Then, to visually represent the data values, a heatmap is generated for ξx_{tt}. The generated heatmap is denoted as H_m.

Graph Construction

Similarly, by employing TK-DAG, a graph is constructed for ξx_{tt} for analysing the data flow and relationship. Directed Acyclic Graphs (DAGs) evaluate the data flow or control flow in a Financial Transaction. However, traversing a DAG to explore all possible paths can be complex. Thus, Tsallis Kahn's technique is used in DAG.
Here, the customer and vendor are represented as nodes (Θ_n), and the transaction details (T_{tal}) are considered as edges $(\exists_\tau)$.

$$\Theta_n = (\Theta_1, \Theta_2, \Theta_3, \ldots, \Theta_x) \quad (12)$$

$$\exists_\tau = \{\exists_1, \exists_2, \exists_3, \ldots, \exists_z\} \quad Here \quad \tau = (1\, to\, z) \quad (13)$$

Here, Θ_x states the x^{th} nodes and $\tau = (1\, to\, z)$ denotes the number of edges. Then, the topological sorting is done by employing Tsallis Kahn's technique, which detects all nodes with zero in-degree and places them in a queue. Thus, a desired topological ordering $(t\rho_y)$ is obtained. Subsequently, the adjacency matrix (α) is computed as,

$$\alpha = \begin{bmatrix} & \Theta_1 & \Theta_2 & \Theta_3 & \Theta_4 & \Theta_5 \\ \Theta_1 & 0 & \exists_1 & \exists_2 & 0 & 0 \\ \Theta_2 & 0 & 0 & 0 & \exists_3 & 0 \\ \Theta_3 & 0 & 0 & 0 & \exists_4 & 0 \\ \Theta_4 & 0 & 0 & 0 & 0 & \exists_5 \\ \Theta_5 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (14)$$

The constructed graph is represented as W_v.

Feature Extraction

The features, such as degree, centrality, connectivity, density, modularity, etc, are extracted from W_v. Similarly, from H_m, the features, including average intensity, entropy, activation symmetry, etc, are extracted. The extracted features are denoted as ζ_f.

Classification

The invoice fraud detection is done based on ζ_f by employing SGTCNs for avoiding unauthorized access. Graph Convolutional Networks (GCNs) preserve the connectivity and relationships between nodes and edges. Nevertheless, the GCNs have a vanishing gradient problem and low learning efficiency. Therefore, the Toeplitz Initialization technique and the SwAT activation function are employed in GCN. In Figure 2, the proposed SGTCNs diagram is depicted.

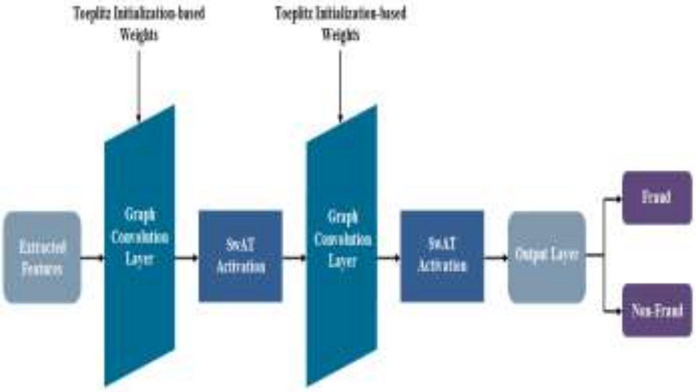

Figure 2. *SGTCNs classifier*

Initially, ζ_f are subjected to the graph convolutional layer $\left(J^{(\phi+1)}\right)$, which updates each node's feature vector by collecting information from its neighbors and converting it using learnable weights.

$$J^{(\phi+1)} = \Omega\left(\eta_{rm} \zeta_f \omega^{(\phi)}\right) \quad (15)$$

$$\Omega = \zeta_f \cdot \frac{1}{1 + exp\left(-tan^{-1} \,|\, left\left(\zeta_f\right)\right)} \quad (16)$$

$$\omega = \left(\zeta_f\right)_{p-q} \quad (17)$$

Here, Ω specifies the SwAT activation function, which solves the low learning efficiency, η_{rm} states the symmetric normalized adjacency matrix of the graph, ϕ defines the layer, ω exemplifies the learnable weight matrix, which overcomes the vanishing gradient problem, $left$ is the left operation, and p and q are the row and columns of the matrix, respectively. After the P convolution layer operation, the output layer (O_{tm}) generates the final predictions about the invoice fraud.

$$O_{tm} = J^P \quad (18)$$

$$VD = (G, F) \quad (19)$$

Where, VD signifies the invoice fraud detection outcomes, G signifies the non-fraud transaction, and F implies the fraud transaction.

Pseudocode for SGTCNs

Input: Extracted Features $\left(\zeta_f\right)$

Output: Invoice Fraud Detection Outcomes (VD)

Begin

Initialize $\left(\zeta_f\right)$

For each $\left(\zeta_f\right)$

Perform input layer

Implement

$$J^{(\phi+1)} = \Omega\left(\eta_{rm} J^{(\phi)} \omega^{(\phi)}\right)$$

Find

$$\Omega = \zeta_f \cdot \frac{1}{1 + exp\left(-tan^{-1} \,|\, left\left(\zeta_f\right)\right)}$$

Initialize weights by Toeplitz Initialization

$$\omega = \left(\zeta_f\right)_{p-q}$$

Perform (O_{tm})

End For

Obtain $VD = (G, F)$

End

The β_κ is given as input to the invoice fraud detection system in the testing time. If fraud transaction is identified, then it is blocked; otherwise, the (K_j) is watermarked by PLDWT (explained in Section 3.3). Then, the digital signature is created for the watermarked invoice data and verified at the destination side, as explained in sections 3.4 and 3.5. Finally, the invoice data reaches the user.

Context-Aware Access Monitoring

Similarly, the context-aware access monitoring is performed based on the contextual information (Φ_g) by using PR-Fuzzy for improving the real-time RM. Fuzzy effectively gives the most probable solution to complex problems. Yet, Fuzzy has a tuning difficulty with the membership function. Therefore, the Parabolic Ramp membership function is utilized in Fuzzy.

Fuzzy rules (Y) are estimated based on If-then conditions.

$$Y = \begin{cases} if \quad lg\,t = \text{normal} \;\; \& \;\; dts = \text{high} \;\; \& \;\; loc = \text{Familar} & \\ \quad \Re t = \text{high} \;\; \& \;\; ivv = \text{low} & then \quad l\gamma \\ if \quad dts = \text{medium} \;\; \& \;\; lg\,t = \text{odd} \;\; \& \;\; loc = \text{Unusual} & \\ \quad \Re t = \text{medium} \;\; \& \;\; ivv = \text{high} & then \quad m\gamma \\ if \quad dts = \text{low} \;\; \& \;\; lg\,t = \text{very odd} \;\; \& \;\; loc = \text{Foreign} & \\ \quad \Re t = \text{low} \;\; \& \;\; ivv = \text{high} & then \quad H\gamma \end{cases} \quad (20)$$

Here, $lg\,t$, dts, loc, $\Re t$, and ivv specify the login time, device trust score, location, role trust, and invoice value, respectively, $l\gamma$ indicates the low-risk access, $m\gamma$ denotes the moderate-risk access, and $H\gamma$ exemplifies the high-risk access. Next, the Parabolic Ramp membership function $(\rho\gamma)$ is computed as,

$$\rho\gamma = \begin{cases} 0 & \Phi_g \le c \\ 2\left(\dfrac{\Phi_g - c}{r - c}\right)^2 & c < \Phi_g \le \dfrac{c + r}{2} \\ 1 - 2\left(\dfrac{r - \Phi_g}{r - c}\right)^2 & \dfrac{r + c}{2} < \Phi_g < r \\ 1 & \Phi_g \ge r \end{cases} \quad (21)$$

Where, r and c implies the upper and lower thresholds, respectively. Here, the decision-making unit performs inference operations. The fuzzification process converts the crisp data into fuzzy data by using $(\rho\gamma)$. Then, to create a single fuzzy output, the output of all the rules is concatenated. Similarly, in defuzzification $(\aleph)$, the fuzzy data are converted into crisp data.

$$CA = [l\gamma, m\gamma, H\gamma] \quad (22)$$

Here, CA denotes the context-aware access monitoring outcomes. If $H\gamma$ is identified, then the access is blocked and an alert is sent to the admin.

Pseudocode for PR-Fuzzy

Input: Contextual information (Φ_g)

Output: Context-aware access monitoring outcomes (CA)

Begin
  Initialize (Φ_g)
  For each (Φ_g)
    Compute fuzzy rules (Y)
    Estimate

$$\rho\gamma = \begin{cases} 0 & \Phi_g \le c \\ 2\left(\dfrac{\Phi_g - c}{r-c}\right)^2 & c < \Phi_g \le \dfrac{c+r}{2} \\ 1-2\left(\dfrac{r-\Phi_g}{r-c}\right)^2 & \dfrac{r+c}{2} < \Phi_g < r \\ 1 & \Phi_g \ge r \end{cases}$$

    Perform decision making
    Convert crisp data to fuzzy data
    Aggregate all rules
    Perform $(\aleph)$
  End For
  Obtain (CA)
End

Therefore, the PR-Fuzzy superiorly identifies the contextual risks.

Tree Construction

Similarly, the invoice information (χ) is constructed as a tree by using EGB-MT for tracking the behavior. Merkle Tree (MT) enables efficient data integrity with minimal data. However, it needs L network accesses to verify all data blocks. Therefore, the Exponential-GINI-BLOOM is included in MT.

The hashcode (X) is created for each (χ)based on Exponential-GINI-BLOOM (eG).

$$eG = \left[(1-\partial f)\cdot S\beta(tm-1) + \partial f \cdot \left(1-\sum(\chi)^2\right)\right] \rightarrow (X_1, X_2, X_3, \ldots\ldots, X_\ell) \quad (23)$$

Here, ∂f denotes the decay factor, $S\beta(tm)$ specifies the bloom filter bucket's updated score at time (tm), and X_ℓ indicates the number of generated hashes. Next, (χ) are split into leaf hash (X_{lh})(contains the persons with performed operations), branch hash (X_{bh})(contains the department information), and root hash (X_{rh})(contains the invoice details).

$$X_{lh} = \{(X_1 + X_2), (X_3 + X_4), \ldots\ldots, (X_{(\ell-1)} + X_\ell)\} \quad (24)$$

$$X_{bh} = \{(X_1 + X_2 + X_3 + X_4), \ldots\ldots, (X_{(\ell-3)} + X_{(\ell-2)} + X_{(\ell-1)} + X_\ell)\} \quad (25)$$

$$X_{rh} = \{(X_1 + X_2 + X_3 + X_4 +, \ldots\ldots, + X_{(\ell-1)} + X_\ell)\} \quad (26)$$

The constructed tree is denoted as Ψ_h. Next, Ψ_h is shared with the admin.

6. Result And Discussion

Here, the comparative validation of the proposed and prevailing methods is done. The proposed model is implemented in the working platform of PYTHON.

Dataset Description

The proposed model employs the "Financial Fraud Detection Dataset" for detecting invoice fraudulent activity. This dataset is collected from publicly available sources, and the dataset link is provided under the reference section. Here, this dataset consists of 1048576 numbers of data. Among that, 80% (i.e., 838861 numbers of data) is employed for training and 20% (i.e., 209715 numbers of data) is used for testing.

Performance Evaluation

Here, the proposed model is weighed against existing methods.

Table 1. *Comparative evaluation based on performance metrics.*

Methods	Fuzzification Time (ms)	De-fuzzification Time (ms)	Rule Generation Time (ms)
Proposed PR-Fuzzy	4268	4362	1024
Triangular Fuzzy	7784	7865	4512
Sigmoid Fuzzy	10269	10457	7458
Gaussian Fuzzy	13587	13254	10147
Cubic Fuzzy	15847	15698	12635

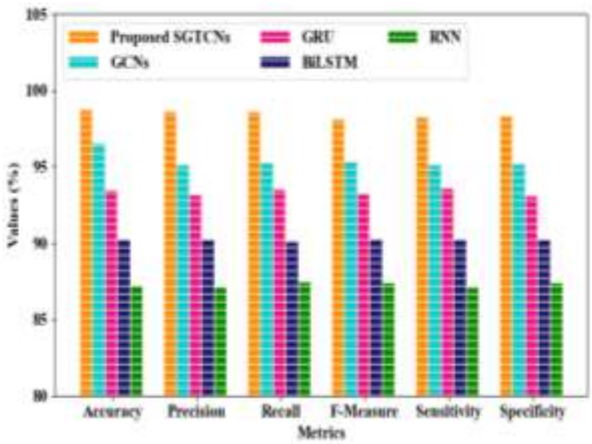

Figure 3. *Graphical analysis of the proposed and prevailing methods*

The proposed PR-Fuzzy's performance is analysed with existing methods in Table 1 and Figure 3. Here, the proposed PR-Fuzzy took minimum fuzzification, defuzzification, along with rule generation times of 4268ms, 4362ms, and 1024ms, respectively due to the inclusion of the Parabolic ramp membership function. Likewise, the existing Triangular Fuzzy, Sigmoid Fuzzy, Gaussian Fuzzy, and Cubic Fuzzy took the maximum times.

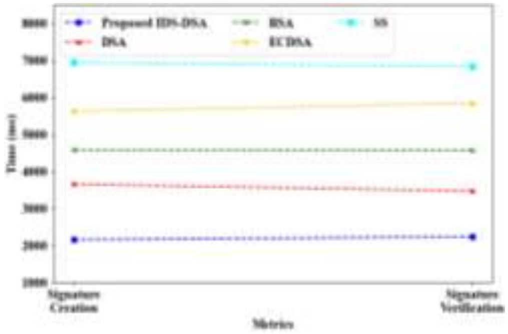

Figure 4. *Pictorial representation of signature creation and verification time*

Table 2. *Performance validation for digital signature creation*

Methods	Signature Creation Time (ms)	Signature Verification Time (ms)
Proposed IDS-DSA	2154	2236
DSA	3658	3471
RSA	4587	4569
ECDSA	5632	5847
SS	6951	6841

In Figure 4 and Table 2, the pictorial representation of the proposed IDS-DSA and prevailing techniques is shown. Here, the proposed IDS-DSA took less signature creation (2154ms) and verification time (2236ms). Nevertheless, the prevailing DSA, Rivest Shamir Adleman (RSA), Elliptic Curve Digital Signature Algorithm (ECDSA), along with Schnorr Signatures (SS) attained limited performance.

Table 3. *Performance assessment for invoice fraud detection*

Methods	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	Sensitivity (%)	Specificity (%)
Proposed SGTCNs	98.7845	98.6527	98.6325	98.1247	98.2471	98.3256
GCNs	96.5678	95.1247	95.2584	95.3656	95.1485	95.2154
GRU	93.5241	93.2148	93.5847	93.2658	93.6225	93.1248
BiLSTM	90.2547	90.2569	90.1487	90.3284	90.2598	90.2659
RNN	87.2569	87.1248	87.5269	87.4512	87.1254	87.4517

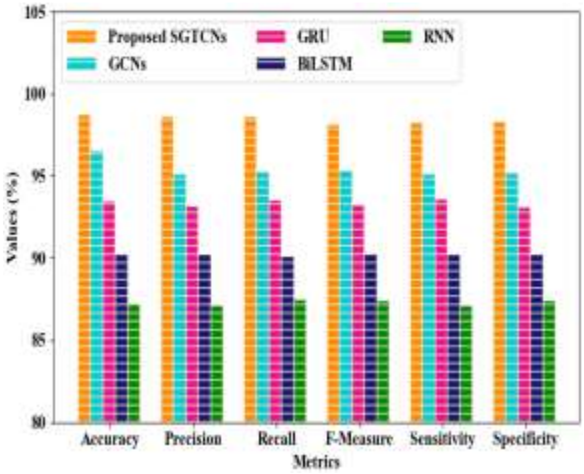

Figure 5. *Comparative validation based on performance metrics.*

The proposed SGTCNs' performance is analysed with prevailing methodologies in Table 3 and Figure 5. Regarding accuracy, precision, recall, F-Measure, sensitivity, and specificity, the SGTCNs achieved 98.7845%, 98.6527%, 98.6325%, 98.1247%, 98.2471%, and 95.2154%; while, the prevailing GCN, Gated Recurrent Unit (GRU), Bidirectional Long Short Term Memory (BiLSTM), and Recurrent Neural Network (RNN) obtained poor performance.

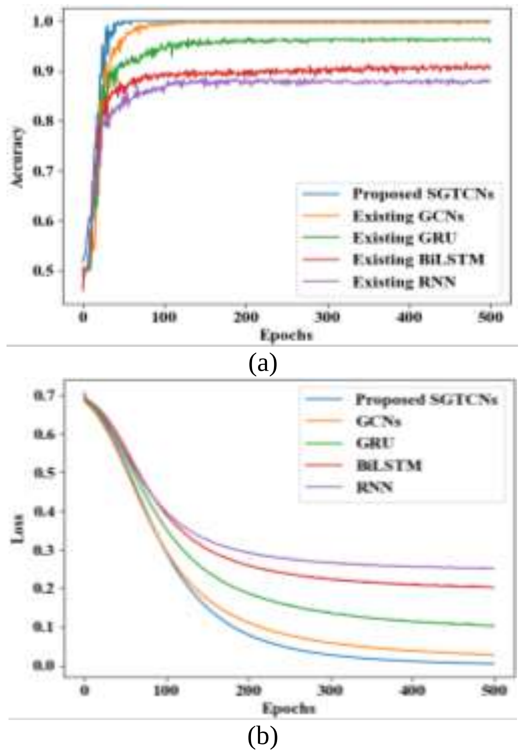

Figure 6. *Performance assessment regarding (a) accuracy vs epochs and (b) loss vs epochs*

The performance assessment regarding accuracy vs epochs and loss vs epochs is depicted in Figure 6.

Here, the accuracy vs epochs graph showed a steady increase in accuracy for different epochs. Also, the loss vs epochs graph represented the loss of the classifier for varying epochs. Thus, both graphs reflected the successful and accurate detection of invoice fraud without overfitting issues.

Table 4. *Merkle tree creation time validation*

Techniques	Merkle Tree Creation Time (ms)
Proposed EGB-MT	3478
MT	4581
RT	5629
BT	6358
VT	7541

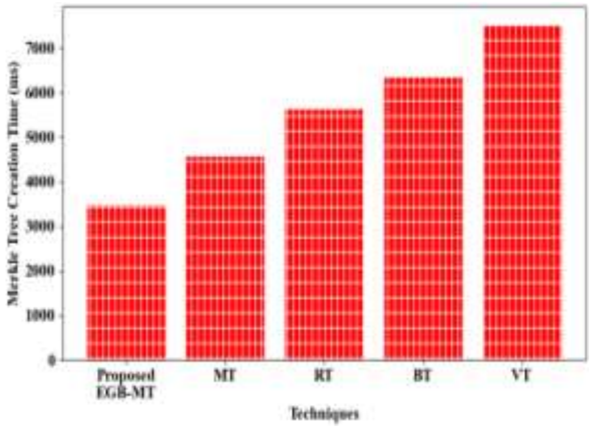

Figure 7. *Graphical representation regarding Merkle tree creation time.*

In Figure 7 and Table 4, the Merkle tree creation time validation of the proposed EGB-MT and conventional methods is shown. Here, the proposed model obtained a low Merkle tree creation time (3478 ms). However, the conventional MT, Radix Tree (RT), Binary Tree (BT), and Verkle Tree (VT) attained a maximum average Merkle Tree creation time of 6027.25ms.

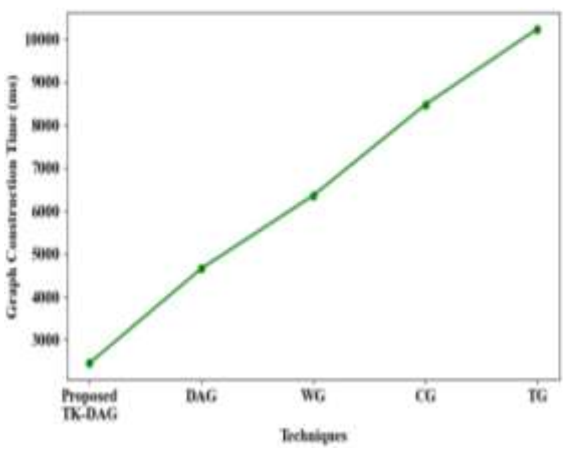

Figure 8. *Graph construction time analysis*

Table 5. *Performance analysis regarding graph construction time*

Methods	Graph Construction Time (ms)
Proposed TK-DAG	2451
DAG	4658
WG	6358
CG	8471
TG	10235

The graph construction time analysis of the proposed TK-DAG along with existing techniques are depicted in Figure 8 and Table 5. Here, the proposed TK-DAG took a minimum graph construction time (2451ms), whereas the existing DAG, Weighted Graph (WG), Connected Graph (CG), and Trivial Graph (TG) attained limited performance.

Comparative Evaluation

Here, the performance of the proposed model along with related works is compared.

Table 6. *Comparative analysis*

Authors' name	Techniques	Precision (%)	Accuracy (%)
Proposed Model	SGTCNs	98.6527	98.7845
(Innan et al., 2023)	QGNNs	95.2	92.3
(Li & Yang, 2023)	Tri-RGCN-XGBoost	89.94	95.41
(Li et al., 2023)	GL	85.2	-
(Innan et al., 2025)	QFNN	95	95
(Cui et al., 2025)	GNN-RL	64.9	-

The proposed model's performance is analysed with related works in Table 6. Here, owing to the inclusion of Toeplitz Initialization and SwAT activation, the proposed SGTCNs achieved a high precision (98.6527%) and accuracy (98.7845%). Nevertheless, the existing Graph Learning (GL) and Graph Neural Network-Reinforcement Learning (GNN-RL) attained low precision values of 85.2% and 64.9%, respectively. Likewise, the conventional Quantum Graph Neural Networks (QGNNs), Tri-Relational GCN-eXtreme Gradient Boosting (Tri-RGCN-XGBoost), and Quantum Federated Neural Networks (QFNN) obtained limited accuracy.

7. Conclusion

Here, this paper presented PR-Fuzzy-based context-aware access monitoring with tracking and alerting for cloud-based invoice processing in financial automation. Here, significant processes, including watermarking, digital signature creation, context-aware access monitoring, tree construction, and invoice fraud detection, were performed. The proposed PR-Fuzzy took minimum fuzzification and defuzzification times of 4268ms and 4362ms, respectively for context-aware access monitoring. Likewise, regarding accuracy, precision, and recall, the proposed SGTCNs achieved a high percentage of 98.7845%, 98.6527%, and 98.6325%, respectively. Thus, the proposed model attained high trustworthiness.

Future Scope

Thus, advanced techniques will be developed in the future for information extraction involving long documents with complex layouts, thus improving the model's effectiveness.

Dataset link: https://www.kaggle.com/datasets/sriharshaeedala/financial-fraud-detection-dataset

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper
- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.
- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

[1] Ali, A., Abd Razak, S., Othman, S. H., Eisa, T. A. E., Al-Dhaqm, A., Nasser, M., Elhassan, T., Elshafie, H., & Saif, A. (2022). Financial Fraud Detection Based on Machine Learning: A Systematic Literature Review. *Applied Sciences (Switzerland),* 12, 1–24. https://doi.org/10.3390/app12199637

[2] Amari, A., Makni, M., Fnaich, W., Lahmar, A., Koubaa, F., Charrad, O., Zormati, M. A., & Douss, R. Y. (2025). An Efficient Deep Learning-Based Approach to Automating Invoice Document Validation. *Arxiv*, 1–8. https://doi.org/10.1109/AICCSA63423.2024.10912544

[3] Arslan, H. (2022). End to End Invoice Processing Application Based on Key Fields Extraction. *IEEE Access*, 10, 78398–78413. https://doi.org/10.1109/ACCESS.2022.3192828

[4] Aung, T., Paw, S., & Tin, H. H. K. (2023). Automated Invoice Processing Using Image Recognition in Business Information Systems. *Economics, Commerce and Trade Management : An International Journal*, 3, 1–8. https://www.researchgate.net/publication/373041004

[5] Bisetty, S. S. S. S., Ayyagari, A., Joshi, A., Goel, O., Kumar, L., & Jain, A. (2024). Automating Invoice Verification through ERP Solutions. *International Journal of Research in Modern Engineering and Emerging Technology (IJRMEET),* 12(5), 1–26.

[6] Chen, Y., & Han, X. (2021). CatBoost for Fraud Detection in Financial Transactions. 2021 IEEE International Conference on Consumer Electronics and Computer Engineering, *ICCECE 2021*, 176–179. https://doi.org/10.1109/ICCECE51280.2021.9342475

[7] Cheng, Y., Guo, J., Long, S., Wu, Y., Sun, M., & Zhang, R. (2024). Advanced Financial Fraud Detection Using GNN-CL Model. In 2024 International Conference on Computers, *Information Processing and Advanced Education*, 453–460.

[8] Cui, Y., Han, X., Chen, J., Zhang, X., Yang, J., & Zhang, X. (2025). FraudGNN-RL: A Graph Neural Network with Reinforcement Learning for Adaptive Financial Fraud Detection. *IEEE Open Journal of the Computer Society*, 6, 426–437. https://doi.org/10.1109/OJCS.2025.3543450

[9] Hamdi, A., Carel, E., Joseph, A., Coustaty, M., & Doucet, A. (2021). Information Extraction from Invoices. *International Conference on Document Analysis and Recognition ICDAR 2021*, Sep 2021, Lausanne, Switzerland., 699–714.

[10] Hilal, W., Gadsden, S. A., & Yawney, J. (2022). Financial Fraud: A Review of Anomaly Detection Techniques and Recent Advances. *Expert Systems with Applications,* 193, 1–34. https://doi.org/10.1016/j.eswa.2021.116429

[11] Innan, N., Marchisio, A., Bennai, M., & Shafique, M. (2025). QFNN-FFD: Quantum Federated Neural Network for Financial Fraud Detection. *Arxiv,* 1–9. http://arxiv.org/abs/2404.02595

[12] Innan, N., Sawaika, A., Dhor, A., Dutta, S., Thota, S., Gokal, H., Patel, N., Khan, M. A. Z., Theodonis, I., & Bennai, M. (2023). Financial fraud detection using quantum graph neural networks. *Arxiv,* 1–15. https://doi.org/10.1007/s42484-024-00143-6

[13] Li, J., & Yang, D. (2023). Research on Financial Fraud Detection Models Integrating Multiple Relational Graphs. *Systems,* 11, 1–16. https://doi.org/10.3390/systems11110539

[14] Li, R., Liu, Z., Ma, Y., Yang, D., & Sun, S. (2023). Internet Financial Fraud Detection Based on Graph Learning. *IEEE Transactions on Computational Social Systems,* 10(3), 1394–1401.

https://doi.org/10.1109/TCSS.2022.3189368

[15] Ogunmokun, A. S., Balogun, E. D., & Ogunsola, K. O. (2022). A Strategic Fraud Risk Mitigation Framework for Corporate Finance Cost Optimization and Loss Prevention. *International Journal of Multidisciplinary Research and Growth Evaluation*, 3(1), 1--9. https://doi.org/10.54660/.IJMRGE.2022.3.1.783-790

[16] Rajagopal, M., Nayak, K. M., Balasubramanian, K., Abdul Karim Shaikh, I., Adhav, S., & Gupta, M. (2023). Application of Artificial Intelligence in the Supply Chain Finance. *Proceedings of 8th IEEE International Conference on Science, Technology, Engineering and Mathematics, ICONSTEM 2023*, 1–7. https://doi.org/10.1109/ICONSTEM56934.2023.10142286

[17] Rohaime, N. A., Abdul Razak, N. I., Thamrin, N. M., & Shyan, C. W. (2022). Integrated Invoicing Solution: A Robotic Process Automation with AI and OCR Approach. *2022 IEEE 20th Student Conference on Research and Development, SCOReD 2022*, 1–5. https://doi.org/10.1109/SCOReD57082.2022.9973841

[18] Singh, P., & Singh, H. (2025). Streamlining Approval Navigating the Invoice Approval Process in SAP OpenText VIM. *SSRN*, 1–7.

[19] Tang, Y., & Liu, Z. (2024). A Distributed Knowledge Distillation Framework for Financial Fraud Detection Based on Transformer. *IEEE Access*, 12, 62899–62911. https://doi.org/10.1109/ACCESS.2024.3387841

[20] Uzozie, O. T., Onaghinor, O., Esan, O. J., & Osho, G. O. (2023). Transforming Procurement Practices with Automation : A Review of Blockchain and RPA Integration for Enhanced Supplier Risk Management Transforming Procurement Practices with Automation : A Review of Blockchain and RPA Integration for Enhanced Supplier Ris. *International Journal of Multidisciplinary Research and Growth Evaluation*, 4(1), 1–8. https://doi.org/10.54660/.IJMRGE.2023.4.1.1151-1157