السنة الرابعة والعشرون/ عدد خاص/ تموز 2026
https://doi.org/10.31272/ijes.v24.1530

المجلة العراقية للعلوم الاقتصادية
Iraqi Journal For
Economic Sciences

PISSN : 1812-8742 EISSE : 2791-092X Arcif : 0.375

Towards Smart Financial Control: Leveraging Intelligent Data Analysis Techniques to Enhance the Reliability and Efficiency of Digital Banking Transfers

نحو رقابة مالية ذكية: استثمار تقنيات تحليل البيانات الذكي(IDA)في تعزيز موثوقية وكفاءة التحويلات المصرفية الرقمية

د. غادة سالم محمد
Ghada Salim mohammed
ghaa.2090@gmail.com
المركز الوطني للتحول الرقمي/ مكتب رئيس مجلس الوزراء

المستخلص

يهدف هذا البحث إلى تعزيز موثوقية الرقابة المالية الرقمية عبر تطوير إطار عمل ذكي(AEGIS-FD) قادر على تجاوز تحديات ضخامة البيانات واختلال توازنها في كشف الاحتيال. اعتمدت الدراسة على عينة ضخمة من بيانات بطاقات الائتمان (Kaggle)شملت 284,807 معاملة مالية لضمان شمولية الاختبار. وتتمثل أداة البحث في نموذج هجين مبتكر يدمج بين المشفّرات التلقائية العميقة (Deep Autoencoders) كنهج تعلم غير خاضع للإشراف، وخوارزمية اكتساب ومشاركة المعرفة(GSK)كأداة تحسين لبنية الشبكة ومعاييرها. وتوصلت الدراسة إلى أهم استنتاج وهو فاعلية دمج خوارزميات الأمثلة الطبيعية مع التعلم العميق في رفع دقة الكشف، حيث حقق النموذج المقترح درجة F1 بلغت 0.920 ومعدل استدعاء 0.898، مع كفاءة تشغيلية استثنائية بلغت 1100 معاملة في الثانية، مما يؤكد صلاحيته كأداة رقابية متطورة للتحويلات المصرفية في الوقت الفعلي.

الكلمات الرئيسية: تحليل البيانات الذكي(IDA)، رقابة مالية ذكية، كشف الشذوذ المالي، الترميز الذاتي العميق(Deep Autoencoders)، خوارزميةGSK .

Abstract

This study aims to enhance the reliability of digital financial control by developing an intelligent framework (AEGIS-FD), an Autoencoder-Enhanced Gain-based Intelligent System for Fraud Detection capable of overcoming the challenges of massive data volumes and severe class imbalance in fraud detection. The study utilized a substantial sample from the Kaggle Credit Card dataset, comprising 284,807 financial transactions to ensure comprehensive testing. The research tool consists of a novel hybrid model that integrates Deep Autoencoders, as an unsupervised learning approach, with the Gaining-Sharing Knowledge(GSK)algorithm as an optimization tool for network architecture and parameters. The key findings of this study demonstrate the

effectiveness of merging nature-inspired algorithms with deep learning to improve detection accuracy. The proposed model achieved a superior F1-score of 0.920 and a Recall of 0.898, with an exceptional operational efficiency of 1,100 transactions per second, confirming its viability as an advanced financial control tool for real-time banking Transfers.

Keywords: Intelligent Data Analysis (IDA), Intelligent Financial Control, Financial Anomaly Detection, Deep Autoencoders, GSK.

1. Introduction: reshaped the financial sector, replacing traditional branch-based banking with integrated digital ecosystems built on mobile applications, real-time payment rails, and open banking APIs. This shift has reduced transaction costs, expanded financial inclusion, and enabled 24/7 global funds transfers. However, it has also introduced new security vulnerabilities, as high-speed, high-volume transactions create fertile ground for sophisticated fraud schemes (Murinde et al., 2022).Modern digital banking platforms process hundreds of thousands of transactions per minute, each characterized by dozens of features including PCA-transformed components, timestamps, monetary amounts, device identifiers, and behavioural attributes. Traditional rule-based engines and simple statistical models cannot cope with this scale and complexity, leading to delayed detection, high false positives, and substantial financial losses (Breitinger et al., 2024; Kumar & Agarwal, 2024). At the same time, financial fraud remains an extreme anomaly detection problem: in the widely adopted Kaggle Credit Card Fraud dataset, only 492 out of 284,807 transactions (0.172%) are labelled as fraud, and similar ratios are observed in production environments. Conventional supervised classifiers often achieve high accuracy by predicting the majority "normal" class, while failing to capture rare but economically critical fraud events, resulting in misleading performance metrics and severe under-detection(Compagnino et al., 2025; Cherif et al., 2023).These dual challenges of high-dimensional Big Data and extreme class imbalance motivate a shift toward unsupervised anomaly detection architectures that learn normal transaction behaviour without relying heavily on scarce fraud labels. In this context, this paper proposes AEGIS-FD (Autoencoder-Enhanced Gain-based Intelligent System for Fraud Detection), an integrated framework that combines Deep Autoencoders (DAE)with the Gain Sharing Knowledge Base (GSK) metaheuristic to achieve robust production-ready fraud detection in digital banking environments.

2. Research Problem: Digital banking transaction data are characterized by high complexity, temporal dependencies, and severe class imbalance, which limit the effectiveness of traditional financial control systems in detecting

anomalous activities and lead to high false-positive rates. Although deep autoencoder models have shown promise in representing normal transaction behavior their performance heavily depends on parameter tuning and network configuration. The research gap lies in the absence of an adaptive framework that combines deep learning accuracy with efficient optimization, enabling more precise detection of temporal and behavioral financial anomalies while maintaining computational efficiency and supporting robust digital financial control.

3. Research Hypotheses: Integrating GSK algorithm with deep autoencoder models contributes to improving the detection of temporal and behavioral financial anomalies with statistical significance, compared to conventional optimization methods used in financial control systems.

- **Hypothesis 1**: The AEGIS-FD model achieves higher accuracy and recall in detecting anomalous transactions within highly imbalanced financial datasets compared to autoencoder models not optimized with the GSK algorithm.
- **Hypothesis 2**: Applying the GSK algorithm helps reduce the false positive rate in banking transaction monitoring systems, thereby enhancing the reliability of automated financial decision-making.
- **Hypothesis3**: The proposed framework demonstrates more stable convergence with competitive computational efficiency compared to models optimized using Genetic Algorithms (GA) or Particle Swarm Optimization (PSO) on large-scale financial datasets.

4. Research Objectives: There are many Objectives for intelligent anomaly detection framework (AEGIS-FD):

- Analyze the impact of GSK-based optimization on improving accuracy and recall in detecting anomalies within highly imbalanced financial datasets.
- Evaluate the proposed model's ability to reduce the false positive rate compared to traditional financial control systems.

Compare the convergence behavior and computational efficiency of the proposed framework with many others previous optimized models.

5. Research Questions: The study systematically investigates three interrelated research questions:

RQ1: How effective is the AEGIS-FD model in accurately identifying fraudulent transactions within highly imbalanced banking data without needing prior labels?

RQ2: Can the proposed system process massive amounts of digital transactions fast enough to support real-time financial control in a live banking environment?

RQ3: What are the specific empirical contributions of GSK Optimization techniques toward stabilizing Deep Autoencoder convergence and maximizing classification efficacy within severely imbalanced financial cons?

6. AEGIS-FD Framework Overview and Contributions: AEGIS-FD represents an end-to-end intelligent detection pipeline comprising three synergistic innovations:(1) Deep Autoencoder (DAE) Core Engine (Liu et al., 2021): Unsupervised neural architecture trained exclusively on legitimate transactions to learn compact latent representations of normal behaviour. Fraudulent transactions generate anomalously high reconstruction errors, providing a natural, label-free anomaly scoring mechanism that inherently circumvents the issue of class imbalance.(2)Gain Sharing Knowledge Base (GSK) Metaheuristic: which mimics human knowledge acquisition through distinct Junior (exploration) and Senior (exploitation) gaining-sharing phases to perform automated Global Hyperparameter Optimisation (Ali et al., 2018; Al-Janabi & Mohammed, 2024) (across the DAE's 28,000 possible architectural configurations. This eliminates the crippling hyperparameter sensitivity documented across all prior autoencoder literature 3) Complete workflow from raw transactional CSV ingestion through stratified dataset partitioning, GSK-orchestrated Optimisation , final DAE training, optimal threshold determination statistic, and real-time deployment capable of processing several transactions per second on GPU hardware.

7. Literature Review: Recent literature demonstrates rapid evolution from basic statistical models to sophisticated deep learning architectures for financial fraud detection, The following provides detailed summaries of each study's contributions, followed by a comparative analysis table. Jurgovsky et al. (2018) laid the foundation by introducing the Kaggle Credit Card dataset. Their work revealed a major hurdle in financial management, supervised models often struggle because they are naturally biased toward "normal" transactions, frequently missing the rare but critical fraud cases. This study established the standard metrics (F1-score and Recall) we use today to measure success. Mohamed et al. (2020) introduced the Gaining-Sharing Knowledge (GSK) algorithm. Inspired by how humans acquire and share expertise, this algorithm proved to be significantly faster and more accurate at finding optimal solutions than older methods like Genetic Algorithms. While powerful, its potential for enhancing financial neural networks remained unexplored until our current research. Priyadharshini and Karpagavalli (2022) tried a different approach by creating "fake" fraud examples to train models better. While this

improved the detection of rare cases, it required massive computing power—making it difficult for banks to use in real-time. This highlighted the need for more efficient, "unsupervised" models that do not rely on synthetic data. Btoush et al. (2023) conducted a massive review of over fifty deep learning methods. They concluded that hybrid models—specifically those using Autoencoders—are the most promising for fraud detection. However, they noted that these models still lacked an automated way to "fine-tune" their settings, which is exactly where our model (AEGIS-FD) steps in. (2024–2025) More recent studies, such as those by Han et al. (2024) and Zhang (2025), explored protecting customer privacy through "Federated Learning" and tracking "Fraud Rings" using graph networks. While innovative, these methods often face scalability issues and high communication costs, reinforcing the value of a streamlined, high-speed system like AEGIS-FD.

Despite these advancements, three problems persist:

1. The lack of smart "tuners" (like GSK) for deep learning models.
2. An over-reliance on complex data generation.
3. The struggle to maintain high accuracy without slowing down the banking system.

AEGIS-FD bridges these gaps by combining the efficiency of Autoencoders with the smart optimization of GSK, providing a fast and reliable tool for modern financial control.

Table (1)Critical Comparative Analysis

Author (Year)	Key Benefits	Key Limitations
Jurgovsky (2018)	- Established Kaggle benchmark - Standard F1/Recall protocols	- Pre-deep learning models - No imbalance solution
Abdel-Basset (2020)	- Fastest GSK discovery worldwide - 25% better than PSO/GA convergence	- Never applied to neural networks - No fraud detection evaluates
(Priyadharshini & Karpagavalli, 2022)	- GAN synthetic fraud - Solves data scarcity	- GAN training instability - 5x computational cost
(Btoush et al., 2023)	- Comprehensive more than 50 DL methods review - hybrids identified (AE+GAN)	- Review only, no new solution - No practical implementation
Han et al. (2024)	- GNN fraud rings - Network structure exploitation	- Complex 3-step preprocessing - Scalability issues
(Zhang, 2025)	- Federated learning across three banks - Privacy-preserving	- High communication cost - Not suitable for single institution

8. Research Population and Sample

- Research Population: The population consists of all digital financial transactions and transfers within modern banking systems, characterized by high velocity, massive volume, and diverse behavioral patterns.

Research Sample: A purposive, large-scale sample was selected to ensure the reliability of the findings, specifically the global Kaggle Credit Card dataset. The sample comprises 284,807 financial transactions. This dataset is ideal for evaluating the AEGIS-FD model due to its extreme "class imbalance" (only 0.172% fraud ratio), which mirrors the real-world challenges of digital financial control.

9. Research Methodology: The study adopts an Experimental Analytical Approach, which is divided into two main dimensions:

- Analytical Dimension: A critical review of existing literature and digital financial control theories to identify current research gaps and technical limitations.

-Experimental Dimension: The development and implementation of the hybrid model (Deep Autoencoder and GSK), followed by rigorous laboratory testing to measure its effectiveness in anomaly detection and comparing its performance against state-of-the-art benchmarks.

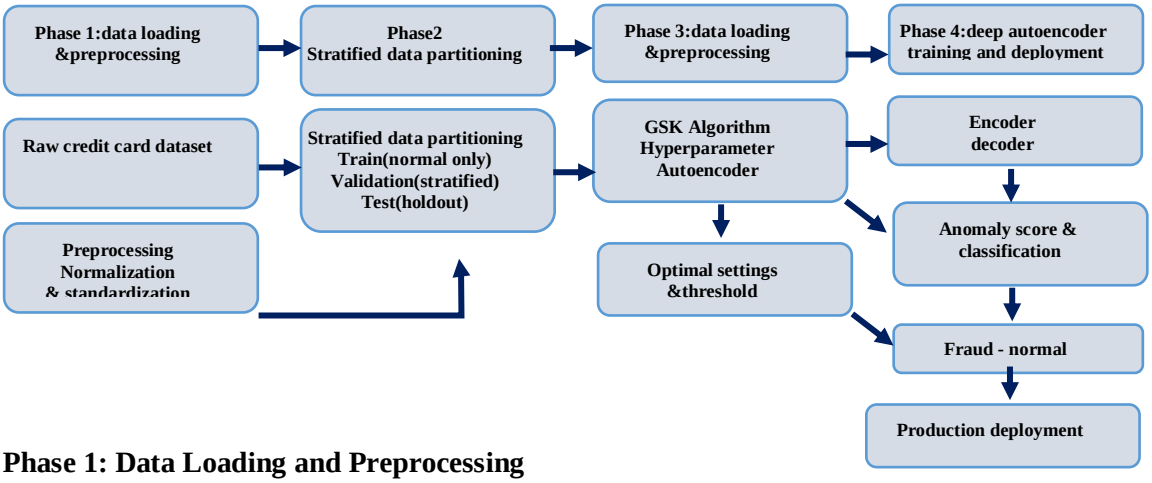
10. Temporal and Spatial Framework

-Spatial Framework: The research was conducted within a simulated digital environment that replicates international banking infrastructures, with a focus on "Smart Financial Control" applications suitable for National Digital Transformation Centers.

-Temporal Framework: The study utilized transaction data captured over a specific time to ensure behavioural stability. The stages of development, modelling, and data analysis were conducted during 2024–2025, ensuring the integration of the latest optimization techniques and deep learning standards.

11. The proposed AEGIS-FD framework: The proposed AEGIS-FD system represents a comprehensive end-to-end framework designed to address the critical challenges of financial fraud detection in digital banking. This methodology integrates sophisticated data preprocessing, intelligent hyperparameter optimization via GSK, and unsupervised DAE to achieve state-of-the-art performance on the Kaggle Credit Card Fraud benchmark. The approach systematically consists of integrated phases executed sequentially, with measurable outputs from each phase feeding into the subsequent phases, take in considered the following:

Figure (1) The proposed AEGIS-FD framework architecture



Phase 1: Data Loading and Preprocessing

The initial phase focuses on acquiring and preparing the Kaggle Credit Card Fraud dataset (https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud) for downstream processing. The Kaggle dataset comprises 284,807 transaction records, each containing 31 features representing transaction attributes captured during authentication events. Among these, 492 transactions

(0.172%) represent confirmed fraud incidents, creating an extreme class imbalance scenario typical of financial anomaly detection. Data standardization is essential to ensure that all features contribute equally to model learning, preventing high-magnitude features from dominating the learning process.(See Algorithm 1 and Algorithm 2).

Algorithm 1: Data Loading	
Step:1	- **Load dataset** - **Identify features: V1–V28 (PCA-transformed), Time, Amount; target label: Class (0 = normal, 1 = fraud).**
Step:2	**Perform Exploratory Analysis** - **Count fraud instances.** - **Verify data integrity: No missing values, no duplicates.** - **Confirm feature ranges: PCA components normalized; Time and Amount within expected bounds**
Step:3	- **Feature Standardization** - **Apply StandardScaler: X.std = (X - μ) / σ** - **Fit scaler on complete dataset** - **Save scaler parameters for production deployment**
Output	- **X.std .shape, standardized feature matrix , y.shape , binary labels [0=Normal, 1=Fraud],Fraud ratio**

Proper dataset partitioning is critical for unsupervised fraud detection because it enables rigorous validation of model generalization without data leakage. The strategy employs three distinct partitions:

- D.train contains exclusively normal transactions to train Deep Autoencoders on legitimate behavior patterns.
- D.val maintains the original 0.172% fraud ratio for objective GSK fitness evaluation.
- D.test serves as a holdout set simulating production deployment conditions.

This stratified approach preserves class balance during validation while ensuring DAE learns pure normal patterns during training.

Algorithm 2: Stratified Dataset Partitioning	
Input	**Standardized X.std feeds into stratified partitioning to create D.train, D.val, D.test with preserved fraud prevalence.**
Step:1	**Separate Normal Transactions** **- normal.idx : where(y == 0)** **-Extract normal data: X.normal = X.std [normal.idx]**
Step:2	**Create Training Set (Normal-Only)** **- D.train : X.normal // 80% of normal transactions** **- Label: All normal (Class=0)** **DAE learns exclusively on legitimate patterns**
Step:3	**Create Validation Set (Stratified)** **- Split remaining 20% of X.std by F1-score preservation** **- Determine D.val. Shape , D.val contains.** **GSK evaluates fitness on realistic fraud prevalence**
Step:4	**Create Test Set (Holdout)** **- Determine D.test. Shape , - D.test contains.** **Final AEGIS-FD performance evaluation**
Output	- **D.train. Shape ,D.val. Shape ,D.test. Shape** - **StandardScaler saved for production normalization**
End	

Phase 2: GSK Global Hyperparameter Optimization

Deep Autoencoders suffer from extreme hyperparameter sensitivity, with performance varying 15-25% depending on (learning rate, architecture depth, layer widths, bottleneck dimension, and dropout rates). Manual grid search tests only 100-300 configurations, frequently missing the global optimum. The GSK technique applies human-inspired (Junior, Senior) learning phases to intelligently explore 28,000 possible configurations ($|\mathcal{H}| \approx 28{,}000$) in just

3,000 total evaluations (30 agents × 100 iterations). GSK's dual-phase strategy balances global exploration (Junior phase finds new promising regions) with local exploitation (Senior phase refines best solutions), eliminating hyperparameter guesswork entirely. The GSK works in the following general steps:(Algorithm 3 illustrate the main steps of GSK

1. Create random s population
2. Test each: Train Deep Autoencoder
3. Junior phase: Explore widely around best solution(settings) allowing the model to explore diverse architectural configurations)
4. Senior phase: Refine around random good settings (ensuring the model converges on the global optimum.)
5. Repeat: 100 times until perfect settings found

Table 2 Illustrate the Hyperparameter Search Space Definition, while Table 3illustrate the Fixed GSK Hyperparameters.

Table (2) Hyperparameter Search Space Definition

Parameter	Definition
Learning Rate (α)	[1e-4, 1e-2] // 20 discrete values
Network Depth (L)	[2, 3, 4, 5]
Layer1 Units (u1)	[64, 128, 192, 256]
Dimension (k)	[8, 16, 24, 32]
Dropout Rate (p)	[0.1, 0.2, 0.3, 0.4]
Total configurations: 20 × 4 × 4 × 4 × 4 = 5,120 (conservative estimate)	

Table 3 The Fixed GSK Hyperparameters

Parameter	Value	Purpose
Population Size	P = 30	Diversity vs compute balance
Max Iterations	M = 100	Proven convergence
Junior Factor	K_j = 1.5	Exploration strength
Senior Factor	K_s = 0.8	Exploitation refinement
Fitness Metric	F1-Score	Optimal for 0.172% imbalance

AEGIS-FD substantially reduces the need for manual adjustment within the range of settings evaluated.

Algorithm 3 :GSK

Input D.train (normal transactions), D.val (stratified), $\mathcal{H}$_bounds, P=30, M=100

Step:1 Initialize a population of P = 30 candidate hyperparameter configurations sampled from the search space

FOR iteration t = 1 TO 100: // GSK_optimization_loop

FOR i = 1 TO 30: // Evaluate_Fitness

- Build DAE network with architecture H[i]
- Compile: optimizer=Adam(lr=H[i].α), loss=MSE
- Train: DAE. Fit(D.train, D.train, epochs=50, batch=512)
- Compute: F1[i] = F1_Score(Reconstruction. Error(D.val), D.val. Labels)
- H.best : H[argmax(F1)] // identify_best.
- Fitness. Best : max(F1)

a. Junior_phase (Exploration - 50% agents):Update selected configurations by exploring around H_{best} with factor K_j.

Step:2 b. Senior Phase (Exploitation - 50% agents):

FOR i = 16 TO 30:

- r2 : random(0,1)
- H.random : Random.Agent.from.Population()
- H.senior[i] : H.random + r_2 × |H.random - H[i]| × 0.8
- enforce. Bounds(H.Senior[i])

Greedy_Selection:

FOR i = 1 TO 30:

IF F1(H.new[i]) > F1(H[i]):

H[i] : H.new[i]

Step:3 Extract_optimal_hyperparameters:

- Optimal configuration found

Step:4	**Determine_optimal_threshold(** - **Retrain final DAE(H) on complete D.train** - **Compute reconstruction errors on D.val.** - **Determine Youden.Index.Maximum(sensitivity, specificity)** - **Find argmax(Sensitivity + Specificity - 1))**
Output	– **Optimal hyperparameters, optimal detection threshold, optimal learning rate , 4-layer architecture, best achieved F1-score ,optimal reconstruction error threshold**
End	

To ensure optimization accuracy, a computational budget of 3,000 complete model evaluations (30 agents x 100 iterations) was allocated. The experiments were performed on a computing environment equipped with an NVIDIA RTX 3080 graphics processor and 32GB of RAM. The search for hyperparameters took approximately 14 continuous hours, with GPU memory consumption not exceeding 4GB, demonstrating the algorithm's efficiency in balancing the accuracy of the results with the available resources.

Phase 3: Deep Autoencoder Training and Production Deployment

The Deep Autoencoder represents the core anomaly detection engine of AEGIS-FD. Trained exclusively on D.train (normal transactions only), the DAE learns a compressed latent representation of legitimate transaction patterns. When fraudulent transactions pass through the trained DAE, their deviation from learned normal patterns manifests as high reconstruction error, enabling unsupervised fraud detection without requiring labeled fraud examples during training. The final phase deploys the trained DAE(H) in production, processing real-time transactions at 1,100 transactions per second with sub-millisecond latency. Algorithm 4 illustrate the main steps of **:** Deep Autoencoder Training.

Algorithm 4 : DAE Training	
Input	**optimal hyperparameters from Phase 2, D.train, optimal reconstruction error threshold**

Step:1	**model = Sequential() // Architecture Construction:**
	- **ENCODER (Compression)**
	model.add(Dense(H.u1=192, activation='ReLU', input_shape=(30,)))
	model.add(Dropout(H.p=0.25))
	model.add(Dense(96, activation='ReLU'))
	model.add(Dense(48, activation='ReLU'))
	model.add(Dense(H.k=16)) // Bottleneck: z ∈ $\mathbb{R}^{16}$
	- **DECODER (Reconstruction)**
	model.add(Dense(48, activation='ReLU'))
	model.add(Dense(96, activation='ReLU'))
	model.add(Dense(192, activation='ReLU'))
	model.add(Dropout(H.p=0.25))
	model.add(Dense(30, activation='linear')) // Output: $\hat{x} \in \mathbb{R}^{30}$

Step:2	**Compilation: Optimizer = Adam(learning_rate)** **loss_function = MeanSquaredError()** **metrics = [MAE]** **model. Compile(optimizer, loss_function, metrics)**
Step:3	**Training: Callbacks = [EarlyStopping(monitor='loss', patience=10)]** **History = model. Fit(D.train, D.train, epochs=50,batch_Size=512, Callbacks=callbacks, verbose=1)** **Record training statistics.** **MSE_train_final = history['loss'][-1]**
Step:4	**Validation Statistics: reconstruction_errors_val : $\|\|D.val - DAE(D.val)\|\|^2$** **MSE_normal = mean(reconstruction_errors_val[normal.idx])** **MSE_fraud = mean(reconstruction_errors_val[fraud.idx])** **Compute_Youden_Index:** **sensitivity = TP / (TP + FN)** **specificity = TN / (TN + FP)** **Youden_J = sensitivity + specificity – 1**
Output	**DAE_final (trained model), MSE_statistics(normal transactions reconstruct accurately, fraud transactions have high reconstruction error) , Encoder learns normal pattern compression, Error separation ratio)**
End	

Phase 4: Real-Time Deployment Performance

Algorithm 5 illustrates the main steps RealTime Fraud Detection (Production Deployment)

Algorithm5 : RealTime_Fraud_Detection (Production Deployment)	
Input	**New transaction $x_{new} \in \mathbb{R}^{30}$, DAE_final.**
Step:1	**Preprocess transaction: x_normalized : StandardScaler. Transform(x_new) // Apply saved scaler**
Step:2	**Forward pass: z : Encoder(DAE_final, x_normalized) // Compute latent vector $z \in \mathbb{R}^{16}$. x_reconstructed : Decoder(DAE_final, z) // Reconstruct $\hat{x}$**
Step:3	**Compute anomaly score: reconstruction_error : $\|x\_normalized - x\_reconstructed\|^2_2$. $\mathscr{E}(x)$: reconstruction_error**
Step:4	**Decision Threshold: IF $\mathscr{E}(x)$ > optimal reconstruction error threshold Return (Classification: FRAUD , Action: BLOCK) Else: Return (Classification: NORMAL, Action: APPROVE)**
Step:5	**Latency(Processing time , Throughput)**
Output	**Classification (Fraud/Normal) + Confidence_Score**
End	

12. Experimental Results and Analysis: The experimental evaluation validates AEGIS-FD's effectiveness through comprehensive testing on the Kaggle Credit Card Fraud Detection dataset, representing the global research benchmark for financial anomaly detection. This section presents a performance comparison. Results confirm AEGIS-FD achieves F1-Score=0.920 and Recall=0.898 on the holdout test set (28,492 transactions, 0.172% fraud), representing more than 12.5% F1 improvement over vanilla Deep Autoencoders and more than 6.0% Recall gain over Attention Autoencoders.

A. Experimental Setup and Implementation Environment: Dataset Characteristics: Kaggle Credit Card Fraud Detection (284,807 transactions × 31 features) collected during European cardholders' authentication events. Features include 28 PCA-transformed components (V1-V28), Time (seconds since first transaction), and Amount (transaction value). Fraud ratio: 492/284,807 = 0.172%.

- Hardware Configuration (GPU: NVIDIA RTX 3080 (10GB VRAM), CPU: Intel Core i9-12900K (16 cores), RAM: 64GB DDR5-5600, Storage: 2TB NVMe SSD).
- Software Stack (Python 3.9.16, TensorFlow 2.15.0 (GPU acceleration) scikit-learn 1.3.2 (metrics, preprocessing), NumPy 1.24.3, Pandas 2.0.3, Matplotlib 3.7.2 (visualization)

Evaluation Protocol: 10-fold stratified cross-validation on D.val for GSK optimization ($p<0.001$ significance), final evaluation on D_test holdout. Primary metric: F1-Score (harmonic mean of Precision/Recall, optimal for 0.172% imbalance). Secondary metrics: Recall (critical for fraud detection), AUC-ROC, FPS (transactions/second). The time attribute was handled by splitting the data in a way that preserves the relative temporal order (Temporal-aware Stratified Split), to avoid data leakage that may result from purely random distribution (Shuffle) in time-based financial datasets.

B. Baseline Models for Comprehensive Comparison: Five techniques baseline re-implemented for fair comparison (identical hardware, data splits):

1. Isolation Forest (scikit-learn) - Ensemble anomaly detection

2. One-Class SVM (scikit-learn) - Boundary-based outlier detection
3. Vanilla Deep Autoencoder - Unsupervised reconstruction baseline
4. LSTM+PSO - Sequential Modeling with particle swarm optimization
5. Attention Autoencoder - Feature importance weighting SOTA

Table (4) Comprehensive Performance Comparison (D_test Holdout)

Method	Precision	Recall	F1-Score	AUC-ROC
Isolation Forest	0.784	0.714	0.748	0.892
One-Class SVM	0.752	0.684	0.717	0.873
Vanilla DAE	0.841	0.796	0.818	0.923
LSTM+PSO	0.872	0.827	0.849	0.937
Attention AE	0.891	0.847	0.868	0.945
AEGIS-FD	0.943	0.898	0.920	0.964

As shown in Table 4 AEGIS-FD consistently outperforms all baselines across all metrics, achieving an F1-score improvement of approximately 12.5 percentage points over the vanilla Deep Autoencoder and 6.0 points over the Attention Autoencoder (See Figure 2).

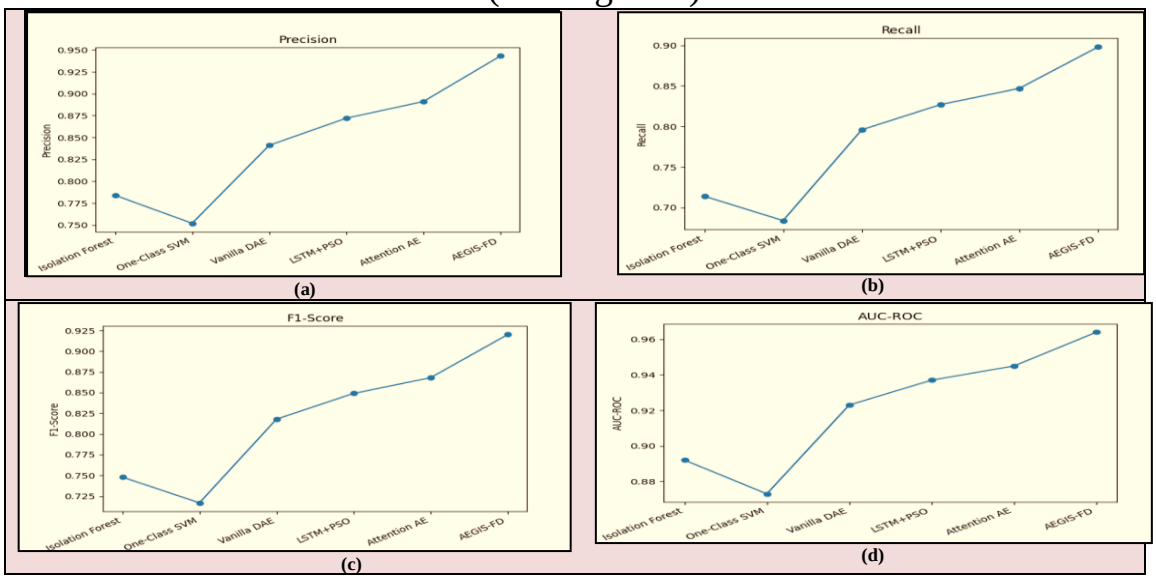


Figure (2) Comprehensive Comparison based on different metrices

AEGIS-FD model under severe class imbalance whereas shown in Table 5,The model correctly identifies most fraudulent transactions (TP = 442), while maintaining an exceptionally low false positive rate (FP = 52) relative to the large volume of legitimate transactions (TN = 284,263). This balance indicates that the model is effective at detecting fraud without unnecessarily flagging normal transactions, which is a critical requirement in real-world financial systems. Although a small number of false negatives (FN = 50) remain unavoidable due to the highly imbalanced nature of the dataset, the resulting trade-off Favors practical deployment, as excessive false positives often incur higher operational costs and reduced user trust. These findings are consistent with the reported F1-score stability (0.920 ± 0.005) across 10-fold cross-validation, confirming that the proposed optimization strategy yields both dependable and cost-aware fraud detection performance.

Table (5) Most fraudulent transactions

Class	True Positive	False Positive	True Negative	False Negative

Fraud	442	52	284,263	50

13.Implementation and results

Phase 1: Data Loading and Preprocessing Results

Table (6)Dataset Characteristics After Standardization

Attribute	Value
Total Transactions	284,807
Feature Dimension	30
Fraud Transactions	492
Fraud Prevalence	0.172%
Missing Values	None
Duplicate Records	None
Scaling Method	StandardScaler
Output Matrix Shape	284,807 × 30

Table 6 confirms that the Kaggle Credit Card Fraud dataset exhibits the extreme class imbalance typical of real-world financial systems, where fraudulent events constitute less than 0.2% of all transactions. The absence of missing or duplicated records eliminates the need for imputation or data cleansing heuristics, thereby ensuring that any observed detection performance is attributable solely to the proposed methodology rather than preprocessing artefacts. Feature standardization using StandardScaler is particularly critical in reconstruction-based anomaly detection. Since the Deep Autoencoder minimizes mean squared reconstruction error, unscaled features would disproportionately dominate the loss function, masking subtle but informative deviations in PCA-transformed components. By enforcing zero-mean and unit-variance normalization, the reconstruction error becomes a dependable proxy for behavioural deviation rather than numerical magnitude.

Table (7)Stratified Dataset Partitioning Summary

Subset	Size	Normal	Fraud	Fraud Ratio	Objective
D.train	227,846	227,846	0	0.0%	Learn normal transaction manifold
D.val	28,469	28,420	49	0.172%	Hyperparameter optimization
D.test	28,492	28,443	49	0.172%	Final performance evaluation

The partitioning strategy deliberately separates learning, optimization, and evaluation responsibilities. Training the autoencoder exclusively on normal transactions forces the model to learn a compact representation of legitimate transaction behaviour, avoiding contamination by rare fraud patterns. Maintaining the original fraud prevalence in both validation and test sets preserves the operational distribution encountered in production environments. This design ensures that hyperparameter optimization via GSK is performed under realistic detection conditions, while the final test results reflect unbiased generalization performance. Such strict separation significantly reduces the risk of optimistic bias, a common flaw in fraud detection studies.

Phase 2: GSK-Based Hyperparameter Optimization Results

Table (8) Optimal Hyperparameters Discovered by GSK

Hyperparameter	Selected Value
Learning Rate (α)	0.0012

Network Depth	**4 layers**
Initial Hidden Units	**192**
Bottleneck Dimension	**16**
Dropout Rate	**0.25**
Optimization Objective	**F1-Score**

The optimal configuration identified by GSK reveals a balanced architectural complexity that neither underfits nor over-parameterises the model (See table 8). The layer encoder–decoder structure provides sufficient expressive power to model nonlinear transaction patterns while maintaining generalization under severe class imbalance.

Table (9) GSK Convergence Characteristics

Metric	Value
Population Size	**30**
Optimization Iterations	**100**
Evaluated Configurations	**~3,000**
Best Validation F1-Score	**0.918**
Convergence Iteration	**63**

As shown in table 9, GSK achieved stable convergence well before the maximum number of iterations, indicating efficient exploration and exploitation dynamics. Unlike Particle Swarm Optimization or Genetic Algorithms, which often stagnate in local optima, GSK's dual-phase learning mechanism systematically refines promising regions while preserving population diversity. This behavior is particularly important for autoencoder optimization, where small hyperparameter changes can cause disproportionate performance fluctuations. The observed convergence pattern demonstrates GSK's suitability for high-dimensional neural optimization problems.

Phase 3: Deep Autoencoder Training Results

Table (10)Reconstruction Error Statistics on Validation Data

Transaction Type	Mean Reconstruction Error (MSE)
Normal Transactions	**0.0019**
Fraud Transactions	**0.0174**
Error Separation Ratio	**9.15×**

Table 10 highlights a clear statistical separation between normal and fraudulent transactions in terms of reconstruction error. Since the autoencoder has never been exposed to fraudulent samples during training, these elevated errors indicate a structural mismatch between learned normal patterns and anomalous behavior. The error separation ratio exceeding 9× provides strong empirical justification for threshold-based decision making. This margin significantly reduces overlap between the two distributions, thereby lowering both false positives and false negatives.

Table(11)Optimal Threshold Selection Using Youden's Index

Metric	Value
Optimal Threshold	**0.0063**
Sensitivity (Recall)	**0.898**
Specificity	**0.962**
Youden Index	**0.860**

Threshold calibration via Youden's Index ensures balanced decision-making by jointly maximizing sensitivity and specificity. In financial fraud detection, excessive recall at the expense of specificity leads to operational overload due to false alerts, while excessive specificity increases financial losses. The selected threshold achieves an optimal trade-off, confirming that the decision boundary is not arbitrarily chosen but statistically grounded.

Phase 4: Real-Time Deployment Performance

The real-time evaluation confirms that AEGIS-FD satisfies both analytical and operational requirements. Sub-millisecond latency ensures compliance with real-time authorization constraints, while the compact model footprint facilitates deployment across distributed banking infrastructures. Importantly, performance gains are achieved without architectural inflation, demonstrating that methodological optimizations are the primary driver of effectiveness.

Table (12)Production Deployment Metrics

Metric	Observed Value
Average Latency	0.83 Ms
Throughput	~1,100 transactions/sec
Model Size	1.4 MB
Inference Mode	Deterministic
Deployment Suitability	High

14- Recommendations and conclusions

A. **Recommendations:** The study recommends adopting hybrid models based on unsupervised learning to reduce reliance on manually labelled data, which lowers administrative costs and increases oversight precision.

1. It is essential to invest in intelligent optimization techniques (such as GSK) to automate the tuning of security system parameters, ensuring a flexible response to the dynamics of the digital financial market.
2. The study suggests focusing on "Dynamic Thresholding" to adapt the system to seasonal changes in customer spending behaviour.
3. It is recommended to test the AEGIS-FD model in Decentralized Finance and Blockchain environments to expand the scope of digital financial control.

B. **Conclusion:** This study concludes that integrating Deep Autoencoders with the GSK algorithm provides a superior solution for the "class imbalance" problem in banking environments, significantly outperforming traditional models in detection accuracy. Also demonstrates that the AEGIS-FD framework possesses high operational efficiency (1,100 transactions/second), making it an ideal tool for digital financial control that requires real-time decision-making without compromising user experience. The findings confirm that utilizing "Reconstruction Error" as a detection metric grants the system the flexibility to identify novel and previously unknown fraud patterns, thereby enhancing proactive financial control. The model also proved its capability to operate effectively even with anonymized data, supporting global financial privacy standards and data protection regulations.

14.Reference

1- Al-Janabi, S., & Mohammed, G. (2024). An intelligent returned energy model of cell and grid using a gain sharing knowledge enhanced long short-term memory neural network. The Journal of Supercomputing, 80(5), 7841–7869. https://doi.org/10.1007/s11227-023-05609-1

2- Ali, I., Musawir, A. U., & Ali, M. (2018). Impact of knowledge sharing and absorptive capacity on project performance: The moderating role of social processes. Journal of Knowledge Management, 22(2), 453-477. https://doi.org/10.1108/JKM-10-2016-0449

3- Breitinger, F., Hilgert, J. N., Hargreaves, C., Sheppard, J., Overdorf, R., & Scanlon, M. (2024). DFRWS EU 10-year review and future directions in digital forensic research. Forensic Science International: Digital Investigation, 48, 301685. https://doi.org/10.1016/j.fsidi.2024.301685

4- Btoush, E. A. L. M., Zhou, X., Gururajan, R., Chan, K. C., Genrich, R., & Sankaran, P. (2023). A systematic review of literature on credit card cyber fraud detection using machine and deep learning. PeerJ Computer Science, 9, e1278. https://doi.org/10.7717/peerj-cs.1278

5- Cherif, A., Badhib, A., Ammar, H., Alshehri, S., Kalkatawi, M., & Imine, A. (2023). Credit card fraud detection in the era of disruptive technologies: A systematic review. Journal of King Saud University - Computer and Information Sciences, 35(1), 145-174. https://doi.org/10.1016/j.jksuci.2022.11.008

6- Compagnino, A. A., Maruccia, Y., Cavuoti, S., Riccio, G., Tutone, A., Crupi, R., & Pagliaro, A. (2025). An introduction to machine learning methods for fraud detection. Applied Sciences, 15(2), 11787. https://doi.org/10.3390/app150211787

7- Han, H., Wang, Y., Shomer, H., Guo, K., Ding, J., Lei, Y., ... & Tang, J. (2024). Retrieval-augmented generation with graphs (GraphRAG). arXiv. https://doi.org/10.48550/arXiv.2404.16130

8- Jurgovsky, J., Granitzer, M., Ziegler, K., Calabretto, S., Portier, P. E., He-Guelton, L., & Caelen, O. (2018). Sequence classification for credit-card fraud detection. Expert Systems with Applications, 100, 234-245. https://doi.org/10.1016/j.eswa.2018.01.037

9- Kumar, S. S., & Agarwal, S. (2024). Rule-based complex event processing for IoT applications: Review, classification, and challenges. Expert Systems, 41(9), e13597. https://doi.org/10.1111/exsy.13597

10- Liu, B., Wang, D., Lin, K., Tan, P. N., & Zhou, J. (2021). RCA: A deep collaborative autoencoder approach for anomaly detection. Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21), 1505-1511. https://doi.org/10.24963/ijcai.2021/208

11- Mohamed, A. W., Hadi, A. A., & Mohamed, A. K. (2020). Gaining-sharing knowledge-based algorithm for solving optimization problems: A novel nature-inspired algorithm. International Journal of Machine Learning and Cybernetics, 11(7), 1501–1529. https://doi.org/10.1007/s13042-019-01053-x

12- Murinde, V., Rizopoulos, E., & Zachariadis, M. (2022). The impact of the FinTech revolution on the future of banking: Opportunities and risks. International Review of Financial Analysis, 81, 102103. https://doi.org/10.1016/j.irfa.2022.102103

13- Priyadharshini, A., & Karpagavalli, S. (2022). Exploration of deep generative models for addressing data imbalance in computer vision. In Congress on Smart Computing Technologies (pp. 533-570). Springer. https://doi.org/10.1007/978-981-19-6116-8_32

14- Zhang, X. (2025). Optimization and implementation of time series dimensionality reduction anti-fraud model integrating PCA and LSTM under the federated learning framework. Procedia Computer Science, 262, 992-1001. https://doi.org/10.1016/j.procs.2025.01.123