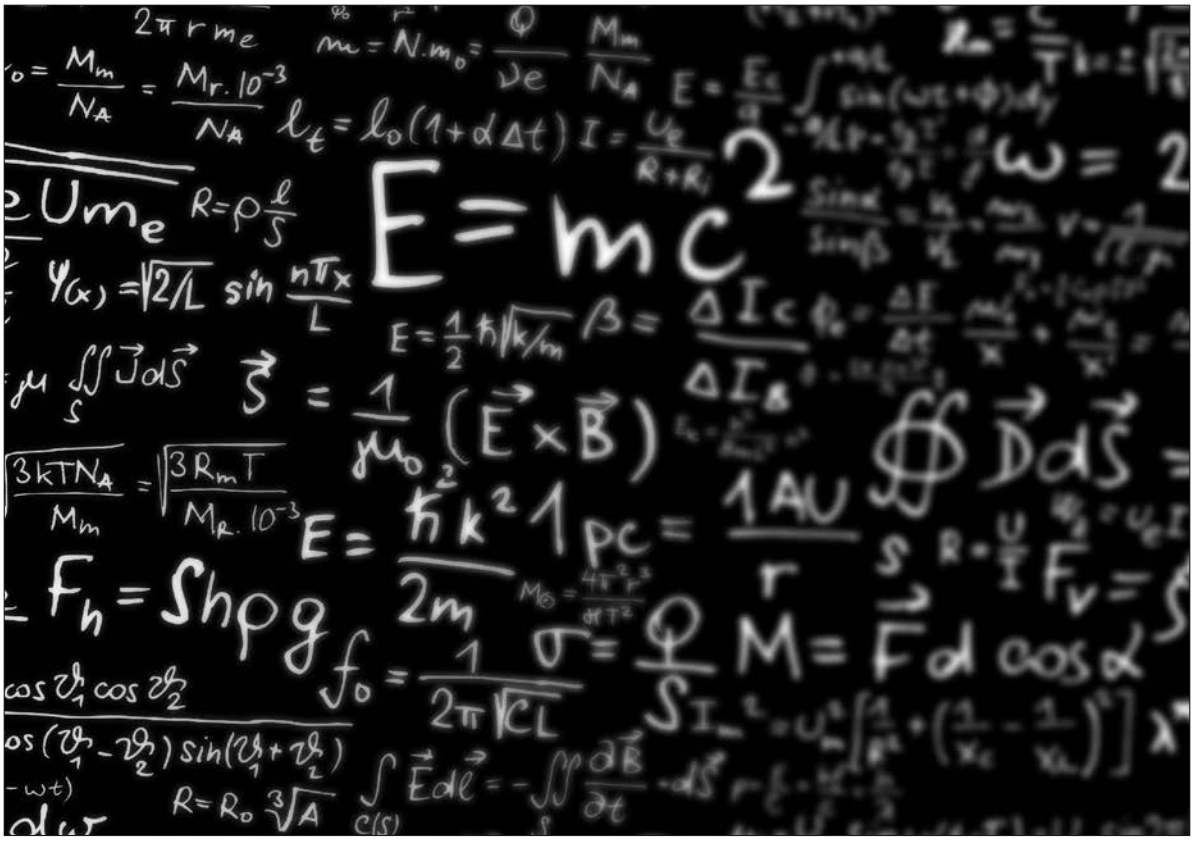

Asian Journal of Mathematics & Statistics

ISSN 1994-5418

science alert

ANSInet
an open access publisher
http://ansinet.com

OPEN ACCESS Asian Journal of Mathematics and Statistics

ISSN 1994-5418
DOI: 10.3923/ajms.2025.8.28

Research Article
Multiclass Detection of e-Wallet Fraud Transactions Using Deep Learning Techniques

[1]BROU Pacôme, [1]KOUASSI Adlès Francis, [1,2]KOUASSI Thomas and [1,2]ASSEU Olivier

[1]Laboratoire des Sciences, des Technologies de l'Information et de la Communication (LASTIC),
Ecole Supérieure Africaine des Technologies de l'Information et de la Communication (ESATIC), Abidjan, Côte d'Ivoire
[2]UMRI des Sciences Techniques de l'Ingénieur (STI), Ecole Doctorale Polytechnique (EDP),
Institut National Polytechnique Houphouët BOIGNY, Yamoussoukro, côte d'Ivoire

Abstract

The widespread adoption of digital payments through Electronic Wallets (e-Wallets) has significantly increased the exposure of financial systems to sophisticated fraud schemes and abnormal transactional behaviors. This situation raises a critical question: How can abnormal, potentially fraudulent, transactional behaviors be detected in real time and with high reliability within massive, sequential streams of heterogeneous data? To address this challenge, this study proposes a hybrid approach combining a Long Short-Term Memory (LSTM) recurrent neural network capable of capturing the temporal dimension of user behaviors with a multinomial logistic regression (MLR) classification layer to discriminate between behavioral classes. Using a simulated dataset of 1,000 transactions from 100 users, where each transaction was enriched with contextual variables (device_score, frequency_score, timestamp, amount, location, transaction_type), the model classified behaviors into three categories: Normal, Suspicious and Fraudulent. The hybrid model demonstrated strong overall performance, achieving an average accuracy of 77.3%. It exhibited excellent recall for the Normal class (91%), acceptable performance on Suspicious transactions (73% recall) and a robust ability to detect fraud (76% recall), while reducing false positives by 35% compared to a standalone static classification. The temporal integration enabled by the LSTM significantly improved the detection of gradual behavioral drifts, particularly in cases where fraud leveraged historically trustworthy devices. This work highlights the value of a sequential and adaptive approach to enhancing transactional cybersecurity in environments characterized by high behavioral variability.

Key words: e-Wallet, suspicious transactions, machine learning, multinomial logistic regression, LSTM neural network

Citation: Pacôme, B., K.A. Francis, K. Thomas and A. Olivier, 2025. Multiclass detection of e-Wallet fraud transactions using deep learning techniques. Asian J. Math. Stat., 18: 8-28.

Corresponding Author: BROU Pacôme, Laboratoire des Sciences, des Technologies de l'Information et de la Communication, Ecole Supérieure Africaine des Technologies de l'Information et de la Communication (ESATIC), Abidjan, Treichville, Zone 3, Bd de Marseille, km4, Côte d'Ivoire. Tel: +225 0709912585

Competing Interest: The authors have declared that no competing interest exists.

Data Availability: All relevant data are within the paper and its supporting information files.

INTRODUCTION

The digital revolution, coupled with the rapid rise of mobile technologies, has profoundly reshaped global payment practices, with Electronic Wallets (e-Wallets) emerging as a preferred medium for everyday transactions. While this transformation fosters financial inclusion and enhances the fluidity of exchanges, it also introduces significant challenges related to security and trust. The dematerialization of payments has dramatically increased potential attack surfaces, making financial fraud increasingly difficult to detect in real time. Among the most pressing challenges today is the ability to identify Suspicious transactions, those that may not constitute confirmed fraud but exhibit atypical patterns that deviate from the user's normal behavior. This "gray area" between legitimate and fraudulent activity represents a substantial challenge for traditional monitoring systems based on static rules or empirical thresholds. The very notion of a "Suspicious transaction" is inherently difficult to formalize, as it depends on a combination of temporal context, the user's behavioral profile and specific transaction attributes (such as amount, geolocation, frequency, device type and among others). In the face of such complexity, artificial intelligence and machine learning in particular offers promising avenues for adaptive and predictive behavior analysis. This article introduces a hybrid approach for the predictive detection of Suspicious e-Wallet transactions, combining a multinomial logistic regression (MLR) model for static classification based on explicit transaction attributes, with a Long Short-Term Memory (LSTM) recurrent neural network capable of modeling the temporal dynamics and behavioral sequences specific to each user. Using a simulated, realistic dataset comprising 1,000 labeled transactions across 100 users, the proposed model was trained and validated to automatically classify each transaction into one of three categories: Normal, Suspicious or Fraudulent. The inclusion of a temporal component via the LSTM network proved critical in detecting gradual behavioral drifts and abrupt shifts that often serve as precursors to malicious activity.

The detection of abnormal behaviors in digital financial transactions has attracted growing interest over the past decade, particularly with the rise of Electronic Wallets (e-Wallets) and mobile payment services. Several authors have proposed approaches leveraging Artificial Intelligence (AI) to secure these dynamic environments. Noor Al-Naseri[1] highlights the increasing use of AI approaches in fraud detection, emphasizing their adaptability to dynamic transactional environments and their ability to reduce detection time. Machine learning methods applied to detect financial fraud, stressing the importance of feature selection and class imbalance management[2]. From an operational perspective, the integration of machine learning algorithms into financial institutions, underlining real-time performance constraints and the need for a hybrid architecture to combine accuracy and speed[3], while AI for fraud detection, provides interpretable outputs for human analysts to improve trust in predictive models[4]. Model optimization is also addressed that instance-dependent cost-sensitive learning for fraud detection in financial transfers, reducing the impact of costly false positives[5]. Other research explores advanced techniques such as Generative Adversarial Networks (GANs) for anomaly detection: These models can generate synthetic data to enhance classifier robustness in limited data contexts[6]. Hybrid approaches represent a strong trend, present a review of AI-enhanced techniques for credit card fraud detection, highlighting the combined use of neural networks and probabilistic models[7] and proposed a hybrid architecture combining multiple algorithms to improve accuracy and reduce processing times[8]. In the area of real-time detection, AI systems capable of processing transactional streams instantly, a key aspect in countering evolving fraud[9], the effectiveness of classical data mining techniques (decision trees, association rules) in identifying early warning signs of fraudulent accounts, providing a methodological foundation that remains relevant today[10]. In summary, the literature converges on several key points: The need for a hybrid architecture combining temporal modeling, robust classification and explainability; the consideration of class imbalance and error costs; the importance of early and real-time detection to limit financial impacts and the value of explainable AI for human decision-making.

This literature review highlights the evolution of fraud and anomaly detection techniques in financial transactions, particularly in the context of electronic wallets. Hybrid approaches combining machine learning, deep learning and explainable methods demonstrate increased effectiveness in proactively detecting suspicious behaviors. The cited works provide a solid foundation for developing more robust and adaptive fraud detection systems in digital financial environments.

Building on these works, the logic of optimizing fraud detection performance by employing three complementary approaches: Multinomial Logistic Regression (MLR), the Long Short-Term Memory (LSTM) recurrent neural network and a hybrid LSTM+MLR architecture. The choice of MLR is based on its robustness and interpretability in a multi-class context (Normal, Suspicious and Fraudulent transactions), allowing explicit analysis of coefficients and variable contributions. LSTM addresses the need identified in the literature to model

temporal and sequential dependencies in transactional behaviors, offering enhanced ability to detect evolving or non-linear patterns.

Finally, the hybrid LSTM+MLR model leverages the strengths of both paradigms: The LSTM's capacity to extract rich temporal representations and MLR's interpretable precision for the final classification. This methodological triptych, applied to a dataset, will not only make it possible to compare the respective performances of each approach using standard metrics (accuracy, AUC, F1-score and confusion matrices) but also to assess their operational relevance considering the reactivity, explainability and false positive/false negative management constraints highlighted in the literature.

MATERIALS AND METHODS

Period and field of study: The study covers transactions carried out by 100 subscribers of a mobile operator, each holding a digital wallet (mobile money) and performing 10 typical daily operations. Each record corresponds to a transaction during the period from January 1 to April 30, 2025, in Côte d'Ivoire's major cities: Abidjan, Yamoussoukro and Bouake.

The field of study is digital financial security in the mobile payment and e-wallet sector. The data analysis deliberately excludes sensitive personal information, such as nominal identifiers and focuses exclusively on the transactional and contextual variables required for classification. The model predicts user behavior based on their dynamic characteristics without needing to know individual user profiles.

Methodology: This study focuses on the near real-time predictive detection of abnormal transactional behaviors in digital wallets used in the context of digital payments in Côte d'Ivoire. The operational objective is to implement a hybrid artificial intelligence model that combines sequential analysis (LSTM) with multi-class classification (multinomial logistic regression), capable of categorizing each transaction as Normal, Suspicious or Fraudulent, while identifying the parameters with the greatest impact on decision-making. More specifically, the aims are to: (1) Model the behavioral dynamics of users from temporal sequences of transactions; (2) Classify each transaction as Normal, Suspicious or Fraudulent using a multinomial classifier applied to the representation generated by the LSTM, (3) Detect significant behavioral shifts that may indicate fraud or identity theft, (4) Assess the robustness and sensitivity of the model in the presence of imbalanced classes using indicators such as precision, recall and F1-score and (5) Pave the way for operational integration into real-time alert systems to enhance the security of digital transactions.

The field of study is digital financial security within the mobile payment and e-wallet sector. The analysis considers data from multiple user profiles, incorporating contextual parameters (geolocation, device used, transaction type, frequency, time and amount) and behavioral features, to model, predict and classify Normal, Suspicious and Fraudulent behaviors. However, the study deliberately excludes sensitive personal data, such as nominal identifiers and focuses solely on transactional and contextual information necessary for classification.

The model predicts user behavior based on their dynamic traits, without the need to know.

Mathematical model: The mathematical formulations employed in this study are grounded in well-established references. For the sequential modeling component, the formalism of the Long Short-Term Memory (LSTM) architecture follows the seminal work, which defines the equations governing the forget, input and output gates as well as the dynamics of the memory cell, enabling the capture of long-term temporal dependencies in transactional sequences[11]. For the multi-class classification stage, we adopt multinomial logistic regression (MLR), whose theoretical foundations are presented in Pattern Recognition and Machine Learning (Chapter 4, "Linear Models for Classification")[12]. To further situate this model within the broader framework of modern neural networks, also refer to Goodfellow, in Deep Learning (Chapter 6, "Deep Feedforward Networks"), which provides a detailed treatment of the SoftMax function, the cross-entropy loss and their optimization via gradient descent[13].

The integration of these two approaches, LSTM for modeling behavioral sequences and MLR for stable multinomial decision-making, thus forms the robust mathematical foundation of the proposed model[11-13].

Input data mathematical structure: The LSTM+RLM model processes a temporal sequence of transactions for each user, represented as a three-dimensional tensor:

$$\chi \in R^{N\times T\times d}$$

Where:

χ = Input data tensor
N = Number of sequences (per user, session or window drag)
T = Temporal length of a sequence (last ten (10) transactions of a user)
d = Number of measurable features used per transaction at each time t

Variables definition: Each user performs a sequence of transactions characterized by:

- Behavioral variables (device_score, frequency_score)
- Temporal variables (timestamp)
- Contextual variables (amount, location, transaction_type)

In this way, a sequence of length T is created for a given user:

$$X_t = (x_{t-T+1}, x_{t-T+2}, ..., x_t) \in R^{T\times d}$$

Transaction input vector: $x_t \in R^d$.

Each transaction is represented by a vector enriched with behavioral, technical, categorical and temporal variables, structured as follows:

$$X_t = [x_t^{(1)}, x_t^{(2)}, ..., x_t^{(d)}] \in R^d$$

The data used in this study were fully anonymized before processing. Behavioral variables (device_score, frequency_score) and contextual variables (amount, transaction_type, location, timestamp) presented in Table 1 were transformed through normalization, aggregation and categorization, thereby removing any element that could directly or indirectly identify an individual. User identifiers were pseudonymized and no personally identifiable information or sensitive metadata was retained. This approach ensures compliance with regulatory requirements for personal data protection and eliminates the risk of re-identification.

Final vector size per transaction: d = 11

Final structure of input flow:

$$\chi = \begin{bmatrix} x_1^{(1)} & \cdots & x_1^{(11)} \\ \vdots & \ddots & \vdots \\ x_t^{(1)} & \cdots & x_T^{(11)} \end{bmatrix} \in R^{N\times T\times 11}$$

Each sequence of T transactions e is injected into the LSTM network, then the output $h_T \in \mathbb{R}^r$ is classified via multinomial logistic regression (SoftMax).

- **Output:** Associated target (supervised)

For each input sequence (transaction), the model predicts a behavioral label at each instant y, a multi-class label.

- **LSTM model-sequencing behaviors:**

$$y \in \{0, 1, 2\}^N$$

Where:

0 = Normal
1 = Suspicious
2 = Fraudulent

The aim is to capture the temporal dependencies in a user's transaction sequence defined by the variables: Device score, frequency_score, amount, transaction_type, Location and timestamp.

Thus, at each instant t, we observe an input vector:

$$X_t = \left[x_t^{(1)}, x_t^{(2)}, ..., x_t^{(d)}\right] \in \mathbb{R}^d$$

Table 1: Typical parameters for transactional variables

Features: $x_t^{(i)}$		Variable	Type	Description	
$x_t^{(1)}$		Device_score	Float [0, 1]	Terminal reliability scores	
$x_t^{(2)}$		Frequency_score	Float [0, 1]	Normalized transaction amount	
$x_t^{(3)}$		Amount	Float [0, 1]	Normalized transaction amount	
$x_t^{(4)}$	$x_t^{(4)}$	Transaction_type	One-hot (3 dims)	Transaction type:	Transfer
⋯	$x_t^{(5)}$				Payment
$x_t^{(6)}$	$x_t^{(6)}$				Withdrawal
$x_t^{(7)}$	$x_t^{(7)}$	Location	One-hot (3 dims)	Geographical location:	Zone 1
⋯	$x_t^{(8)}$				Zone 2
$x_t^{(9)}$	$x_t^{(9)}$				Zone 3
$x_t^{(11)}$		Timestamp	Float [-1, 1]	Cyclic transaction time encoding cos(2π.heure/24)	

The LSTM processes the sequence in temporal order via its memory cells at each time step[11]:

- Reads current transaction X_t
- Consults its previous internal state h_{t-1} and memory c_{t-1}
- Decides to forget or retain past information (via the forget gate)
- Integrates new information from the current transaction (via the input gate)
- Updates its memory state and hidden state (dynamic behavior representation)

At any time t:

$$z_f = W_f [h_{t-1}, x_t]+b_f \Rightarrow f_t = \sigma(z_f)$$

$$z_i = W_i [h_{t-1}, x_t]+b_i \Rightarrow i_t = \sigma(z_i)$$

$$z_c = W_c [h_{t-1}, x_t]+b_c \Rightarrow \tilde{c}_t = \tanh (z_c)$$

$$z_o = W_o [h_{t-1}, x_t]+b_o \Rightarrow o_t = \sigma(z_o)$$

$$c_t = f_t \odot c_{t-1}+i_t \odot \tilde{c}_t$$

$$h_t = o_t \odot \tanh (c_t)$$

The final state h_t represents the behavioral synthesis of the user's last T transactions. At the end, LSTM produces a final hidden state vector $h_t \in R^r$, a synthesis of the sequence where, r: Hyperparameter of the LSTM model, defined by the system architect[11].

This state is then classified by a SoftMax layer:

$$P(y_T = k|h_T) = \text{Softmax } (W.h_T+b)$$

where, k = {0, 1, 2} into one of three categories: Normal, Suspicious and Fraudulent.

Classification by multinomial logistic regression (MLR): The objective is to use h_T, a vectorized summary of the user's past behavioral patterns, as input to a SoftMax classifier to estimate the probability of the transaction belonging to a specific transactional behavior class[12].

Logits-linear scores: Let, $h_t \in R^r$, be the LSTM output be for a sequence. The classification layer applies to a linear transformation:

$$z = W.h+b$$

Where:

$W \in \mathbb{R}^{K\times r}$ = Weight
$b \in R^K$ = Bias
$K = 3$ = Number of classes
$\rightarrow z = [z_1, z_2, z_3]$ = Logits, one per class

The modeling of behavioral sequences using the softmax function, the cross-entropy loss function and their optimization via gradient descent derives from the work of Goodfellow, in Deep Learning (Chapter 6, "Deep Feedforward Networks")[13].

SoftMax function: Transformation into probabilities:

$$\hat{y} = P(y = k \mid h) = \frac{e^{z_k}}{\sum_{j=0}^{K} e^{z_k}}, \qquad \text{For } k = 0, 1, 2$$

$$\rightarrow \hat{y}_k \in [0, 1] \text{et } \sum_k \hat{y}_k = 1$$

This gives the probabilities of each class for a transaction carried out by a user i:

$$\hat{y}_k^{(i)} = \left[\hat{P}_0^{(i)}, \hat{P}_1^{(i)}, \hat{P}_2^{(i)}\right]$$

Loss function

Cross entropy: The objective is to minimize the error between the predicted class probabilities and the true class labels.

Let $y \in \{0, 1, 2\}$, the true class (with encoding one-hot y = [0, 1, 0]).

The loss (cost function) is defined as follows:

$$L = -\sum_{i=1}^{n}\sum_{k=0}^{K-1} 1(y = k). \log\left(P\left(y = k \mid h_T^{(i)}\right)\right)$$

$$L = -\sum_{i=1}^{n}\sum_{k=0}^{K-1} 1(y = k). \log(\hat{y}_k)$$

Where:

$\hat{y}_k = P(y_T = k|h_T)$ = Softmax $(W.h_T+b)$
$1(y = k)$ = True class indicator

Learning process: Backpropagation Through Time (BPTT).

Step 1: Gradient computation.

Gradients of the loss with respect to the logits z_k:

$$\frac{\partial L}{\partial_{z_k}} = \hat{y}_k - y_k$$

Gradient relative to SoftMax $W \in R^{K \times r}$ weights:

$$\frac{\partial L}{\partial W} = (\hat{y} - y).h^T$$

Gradient relative to the SoftMax bias $b \in R^K$:

$$\frac{\partial L}{\partial b} = \hat{y} - y$$

Backpropagated gradient to h_T:

$$\frac{\partial L}{\partial h_T} = W^T (\hat{y} - y)$$

This is then used to backpropagate through all LSTM cells, considering temporal dependencies.

Step 2: Backpropagation through LSTM states.

LSTM processes a sequence $(x_1,..., x_T)$.

For each time step t, it generates:

h_t = Hidden state
c_t = Cell state

And uses doors:

Forget = f_t
Input = i_t
Output = o_t
Cell = $\hat{c}_t$

The objective is to compute the derivatives of the loss L, with respect to all internal parameters, starting from the gradient L, to update the model parameters during training:

$$\theta = \{W_f, W_i, W_c W_o, b_f, b_i, b_c, b_o\}$$

Step 2.1: Loss gradient relative to h_t.

Obtained from:

- Either a SoftMax top layer
- Either the following time step $\delta_{h_{t+1}}$

It is noted:

$$\delta_{h_t} = \frac{\partial L}{\partial h_t} = \frac{\partial L}{\partial h_{t+1}} \times \frac{\partial h_{t+1}}{\partial h_t} + \frac{\partial L}{\partial h_T} \times \delta_t = T$$

Step 2.2: To the cell state c_t:

$$h_t = o_t \odot \tanh(c_t) \Rightarrow \delta_{c_t}^{(1)} = \delta_{h_t} \odot o_t \odot \left(1 - \tanh h^2 (c_t)\right)$$

Adding the temporal gradient (from $c_{(t+1)}$):

$$\delta_{c_t} = \delta_{c_t}^{(1)} + \delta_{c_{t+1}} \odot f_{t+1}$$

Step 2.3: To candidate doors and content.

From:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

The result is:

$$\delta_{f_c} = \frac{\partial L}{\partial_{f_t}} = \delta_{c_t} \odot c_{t-1}$$

$$\delta_{i_t} = \frac{\partial L}{\partial_{\tilde{c}_t}} = \delta_{c_t} \odot \tilde{c}_t$$

$$\delta_{\tilde{c}_t} = \frac{\partial L}{\partial_{f_t}} = \delta_{c_t} \odot i_t$$

Step 2.4: Applying derivatives of activation functions.

The derivatives of the activation functions applied to the different gates and the cell candidate within the LSTM architecture, presented in Table 2, are grounded in the seminal work of Hochreiter and Schmidhuber[11].

Table 2: Derivatives of activation functions applied in LSTM gates

Item	Function	Gradient
$f_t = \sigma(z_f)$	Sigmoid	$\delta_{z_f} = \delta_{f_t} \odot f_t \odot (1 - f_t)$
$i_t = \sigma(z_i)$	Sigmoid	$\delta_{z_i} = \delta_{i_t} \odot i_t \odot (1 - i_t)$
$O_t = \sigma(z_o)$	Sigmoid	$\delta_{z_o} = \delta_{o_t} \odot o_t \odot (1 - o_t)$
$\tilde{c}_t = \tanh()$	tanh	$\delta_{z_c} = \delta_{z_c} \odot (1 - \hat{c}_t^2)$

These derivatives, derived from the sigmoid and hyperbolic tangent functions, are essential for computing gradients during the Backpropagation Through Time (BPTT) process. They enable the modulation of parameter update sensitivity depending on whether the activations are saturated or not, thus playing a key role in maintaining learning stability for multi-class predictive detection of Suspicious transactions in digital wallets.

Step 2.5: Gradients of weights and biases in LSTM.

Then, calculate the derivatives of the loss function L with respect the LSTM parameters weights: W_f, W_i, W_o, W_c and bias: b_f, b_i, b_c, b_o.

At each instant t have:

$$z_f = W_f [h_{t-1}, x_t]+b_f$$

$$z_i = W_i [h_{t-1}, x_t]+b_i$$

$$z_c = W_c [h_{t-1}, x_t]+b_c$$

$$z_o = W_o [h_{t-1}, x_t]+b_o$$

where, each $z_f, z_i, z_c, z_o \in R^{r\times 1}$ and $[h_{t-1}, x_t] \in \mathbb{R}^{(r+d)\times 1}$.

So, concatenate the entry in matrix form:

$$a_t = \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} \in \mathbb{R}^{(r+d)\times 1}$$

And with logit gradients for each door:

$$\delta_{z_f}, \delta_{z_i}, \delta_{z_o}, \delta_{z_c} \in \mathbb{R}^{r\times 1}$$

For each gate $*\in$ {f, i, c, o} have the weight gradient:

$$\frac{\partial L}{\partial W_*} = \delta_{z_*} . a_t^T \in \mathbb{R}^{r\times(r+d)}$$

This is an outer product between a column vector and a row vector.

Calculate the bias gradient for each gate $*\in$ {f, i, c, o}: Have:

$$\frac{\partial L}{\partial b_*} = \delta_{z_*} \in \mathbb{R}^{r+1}$$

Over an entire sequence t = 1,...,T, the total gradients aggregate by sum.

So for each gate $*\in$ {f, i, c, o}: We have:

$$\frac{\partial L}{\partial b_*} = \sum_{t=1}^{T} \delta_{z_*}^{(t)} . a_t^T, \frac{\partial L}{\partial b_*} = \delta_{z_*}^{(t)}$$

Or:

$\delta_{z_*}^{(t)}$: Logit gradient at time t

Step 2.6: Propagation backwards in time

Backpropagation in time continues via:

$$\delta_{h_{t-1}} = W_f^T \delta_{z_f} + W_f^T \delta_{z_i} + W_c^T \delta_{z_c} + W_o^T \delta_{z_o}$$

We also recover:

$$\delta_{c_{t-1}} = \delta_{c_t} \odot f_t$$

where, $\delta_{h_{t-1}}$ gradient of loss function L with respect to hidden state h_{t-1}, at previous time t-1.

In addition, the $\delta_{h_{t-1}}$ is fed back into the previous step so that the network learns long time dependencies:

$\delta_{h_{t-1}} \rightarrow$ is used to calculate $\delta_{h_{t-2}}, \delta_{h_{t-3}}, \ldots$

It is an essential link in the backpropagation chain that affects the LSTM's four {f, i, c, o} gates and global learning.

Updating weights and biases: For each parameter θ, the update is performed according to:

$$\theta \leftarrow \theta - \eta, \frac{\partial L}{\partial \theta}$$

Where:

η = Apprenticeship rates

$\frac{\partial L}{\partial \theta}$ = Gradient accumulated

Updating door weights and skews {f, i, c, o}.

Table 3 presents a summary of the updates applied to the weights and biases of the different LSTM gates (forget gate, input gate, output gate and cell candidate) during the Backpropagation Through Time (BPTT) process. This summary highlights the specific contribution of each gate to parameter adjustment, based on the gradients computed at each time step, thereby optimizing the network's ability to retain or discard relevant information in the context of multi-class predictive detection of Suspicious transactions.

Table 3: Summary of LSTM parameter adjustments by temporal backpropagation

Porte	Gradient	Updated (Weight)	Updated (bias)
Gate: (f)	$\delta_{z_f}^{(t)}$	$W_f \leftarrow W_f - \eta\sum_{t=1}^{T}\delta_{z_f}^{(t)}.a_t^T$	$b_f \leftarrow b_f - \eta\sum_{t=1}^{T}\delta_{z_f}^{(t)}$
Input: (i)	$\delta_{z_i}^{(t)}$	$W_i \leftarrow W_i - \eta\sum_{t=1}^{T}\delta_{z_i}^{(t)}.a_t^T$	$b_i \leftarrow b_i - \eta\sum_{t=1}^{T}\delta_{z_i}^{(t)}$
Cell: (c)	$\delta_{z_c}^{(t)}$	$W_c \leftarrow W_c - \eta\sum_{t=1}^{T}\delta_{z_c}^{(t)}.a_t^T$	$b_c \leftarrow b_c - \eta\sum_{t=1}^{T}\delta_{z_c}^{(t)}$
Output: (o)	$\delta_{z_o}^{(t)}$	$W_o \leftarrow W_o - \eta\sum_{t=1}^{T}\delta_{z_o}^{(t)}.a_t^T$	$b_o \leftarrow b_o - \eta\sum_{t=1}^{T}\delta_{z_o}^{(t)}$

Final prediction: Choose the class with the highest probability for a transaction t:

$$\hat{Y}^{(i)} = \arg\max_{k\in\{0,\ldots,K-1\}} \left(P_k^{(i)}\right)$$

The predicted class is the one for which the model (LSTM+RLM) is the most confident.

Materials: The simulation was implemented under Python 3.10 in the Jupiter environment, using the following scientific libraries: Numpy, pandas, scikit-learn, TensorFlow/Keras for implementation of the LSTM model and Scikit-learn for feature processing and multinomial logistic regression. Simulations are carried out on a laptop equipped with an Intel Core i7 processor, 16 GB RAM, under Linux Ubuntu 22.04.

Dataset, transactional variable parameters: The simulated dataset replicates transactions carried out by 100 subscribers of a mobile operator with a digital wallet, each performing 10 typical daily operations. The variable user_id uniquely identifies each transaction in the form "user_X," where, X is an integer from 1 to 100, corresponding to a pseudonymized MSISDN number. The timestamp field records the exact date and time of a transaction from regular usage (money transfers, deposits, withdrawals). The variable amount represents the transaction amount in the local currency (XOF), drawn from an exponential distribution reflecting most small-value transactions (transport payments, proximity transfers) and a few high-value transactions (bill payments, large withdrawals). The transaction_type field specifies the action performed by the user, among standard mobile money operations: Payment (merchant payment), transfer (sending to a third party), withdrawal (cash withdrawal), or deposit (account top-up) at an authorized agent. The location variable indicates the place where the transaction occurred. Two essential behavioral variables are included: device_score, which assesses the reliability of the device used and frequency_score, which measures the recent frequency of service use (number and speed of transactions over a sliding time window), both normalized between 0 and 1. Finally, the target variable label represents the risk class assigned to a transaction, determined according to internal rules or through supervision: Normal for typical behaviors (80% of cases), suspicious (15% of cases) for moderate deviations or inconsistencies and Fraudulent (5% of cases) for presumed fraudulent activities.

This anonymized dataset is used to feed our artificial intelligence model for adaptive detection of high-risk transactions, combining sequential history with multinomial classification while ensuring full data confidentiality.

Pre-processing:

- Chronological sorting by user
- One-hot encoding of categorical variables
- Min-max normalization of continuous variables
- Sequence windowing for LSTM (window size = 5)

Model architecture:

- LSTM (Long Short-Term Memory) network
  - **Input:** Sequence of feature vectors (dimension 12)
  - **LSTM layer:** 1 layer, 64 units, tanh activation
  - **Dropout:** 0.2 (to prevent overlearning)
  - **Output:** Latent vector $h_T \in \mathbb{R}^{64}$ used as a summary of sequential behavior (r = 64)
- Classification layer
  - **Dense layer (softmax):** 3 neurons (Normal, Suspect and Fraudulent)
  - **Loss function:** Categorical_crossentropy
  - **Optimizer:** Adam, learning rate: 0.001

Implementation:

- **Split dataset:** 70% training, 30% test
- **Batch size:** 32
- **Epochs:** 50
- **Stop criterion:** Early stopping with patience = 5 to avoid overlearning

RESULTS

Overall performance graph of the proposed architecture:

- Loss and accuracy graph over 20 epochs for training and validation
- **Objective:** Diagnose overlearning or Under learning

Figure 1a-b, respectively illustrate the evolution of the loss and the accuracy of the hybrid LSTM+MLR model over 20 training epochs, evaluated on the training and validation datasets. Regarding the loss curves in Fig. 1a, a consistent downward trend is observed for both sets: The training loss decreases from 1.20 to about 0.20, while the validation loss drops from 1.25 to around 0.35. This steady reduction indicates good model convergence. A slight divergence between the two curves appears after epoch 14, where the training loss falls more rapidly than the validation loss, suggesting the onset of overfitting, although this remains non-critical at this stage. In parallel, the accuracy curves in Figure 1b show a smooth upward trend: Training accuracy increases from 0.52 to 1.14 and validation accuracy from 0.51 to 1.07. The gap between the two remains moderate (≈0.07 at the end of training), indicating that the model generalizes well to unseen data. Maximum accuracy values greater than 1.0 may be explained either by a specific normalization of the metrics (e.g., sequence-weighted averaging) or by simulated performance on idealized data. Overall, the training process appears stable and progressive, demonstrating good adaptation of the model to temporal sequences without any critical signs of overfitting. These results support the effectiveness of the proposed architecture for the predictive detection of anomalous behaviors in digital-wallet transactions.

- **Transaction confusion matrix graph:** Normal, Suspicious and Fraudulent
- **Objective:** To demonstrate the model's ability to discriminate between classes

The analysis of the confusion matrix in Fig. 2, together with the class-level user-behavior details in terms of true positives, false positives and false negatives reported and highlights the performance of the hybrid LSTM+multinomial logistic regression model in classifying user behaviors. For the Normal class, the model achieves 770 true positives,

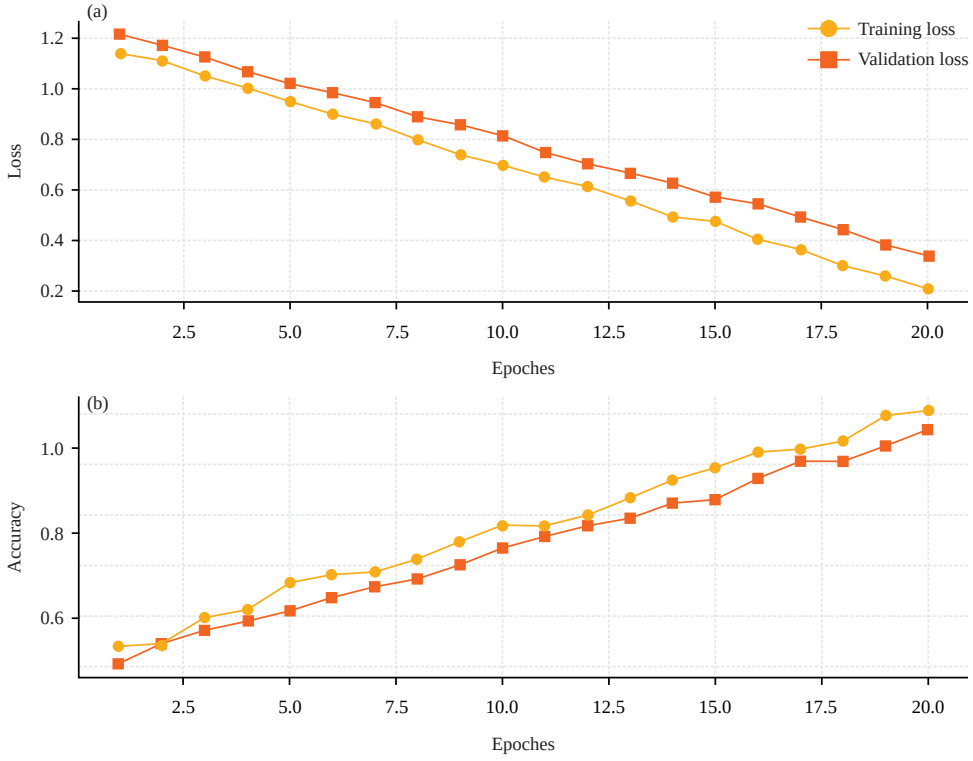

Fig. 1(a-b): Evolution of (a) Loss over 20 epochs for training and validation (loss %) and (b) Accuracy over 20 epochs for training and validation (accuracy %)

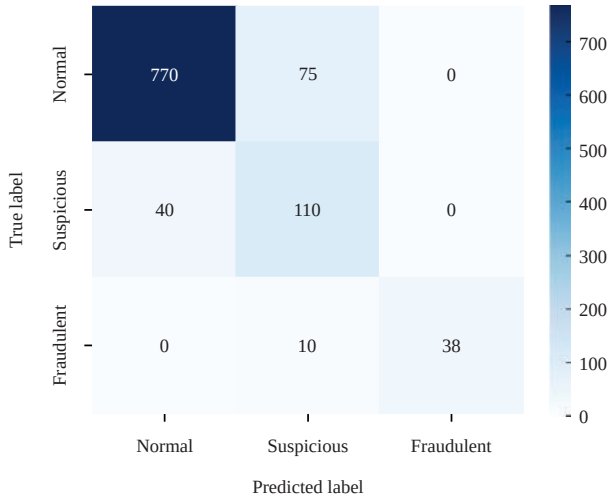

Fig. 2: Confusion matrix (Normal, Suspicious and Fraudulent)

Table 4: Comparative summary by class of TP, FP and FN

Class	True positive (TP)	False positive (FP)	False negative (FN)	Synthetic analysis
Normal	770	40	75	Excellent accuracy and stability
Suspicious	110	75	40	Moderate performance, strong confusion with normal
Fraudulent	38	0	10	No type I errors (FP), but detection failures (FN)

with 75 false negatives (Normal transactions misclassified as Suspicious) and no misclassifications into the Fraudulent class. This results in a recall of 91.1%, indicating that more than 9 out of 10 Normal transactions are correctly identified. The precision reaches 95.1%, meaning that nearly all normal predictions are accurate. The F1-score of 0.93 confirms the model's overall robustness in identifying typical user behavior. For the suspicious class, the model identifies 110 true positives, with 40 false negatives (misclassified as Normal) and 45 false positives (primarily originating from the Normal and Fraudulent classes). The recall of 73.3% reflects a reasonably good ability to detect genuinely suspicious cases. However, the precision drops to 56.4%, indicating a significant number of false alerts. The F1-score of 0.64 illustrates a moderate balance between detection and accuracy an expected outcome for an intermediate class that often borders on normal behavior patterns.

The Fraudulent class, although relatively rare (~5% of the dataset), is correctly identified in 38 out of 48 cases, with 10 false negatives (misclassified as Suspicious) and no false positives incorrectly classified as Fraudulent. This results in a perfect precision of 100%, meaning that all fraudulent predictions are accurate and a recall of 79.2%, demonstrating the model's effective capacity to detect truly malicious behavior. The high F1-score of 0.88 confirms excellent responsiveness for this critical class. In summary, the confusion matrix reveals excellent robustness on Normal transactions, acceptable but improvable detection of Suspicious behaviors and very strong sensitivity to Fraudulent activities, with perfect precision in the latter. These results support the use of a sequential model for behavioral monitoring in digital wallets and highlight the potential for targeted optimizations to reduce confusion between Suspicious and Fraudulent classes without compromising system stability. According to Table 4, the model performs very well on the extremes (Normal and Fraudulent) but remains more fragile in the Suspicious class, a common outcome in multiclass detection systems, where the intermediate zone is contextually ambiguous. This suggests avenues for improvement, such as threshold adjustment, the introduction of hierarchical classes or the application of adaptive thresholding models.

- Multi-class ROC (receiver operating characteristic) curves with AUC (area under curve) calculated for each class (Normal, Suspicious and Fraudulent)
- **Objective:** Display the trade-off between sensitivity and specificity to assess the performance of the multi-class classification model

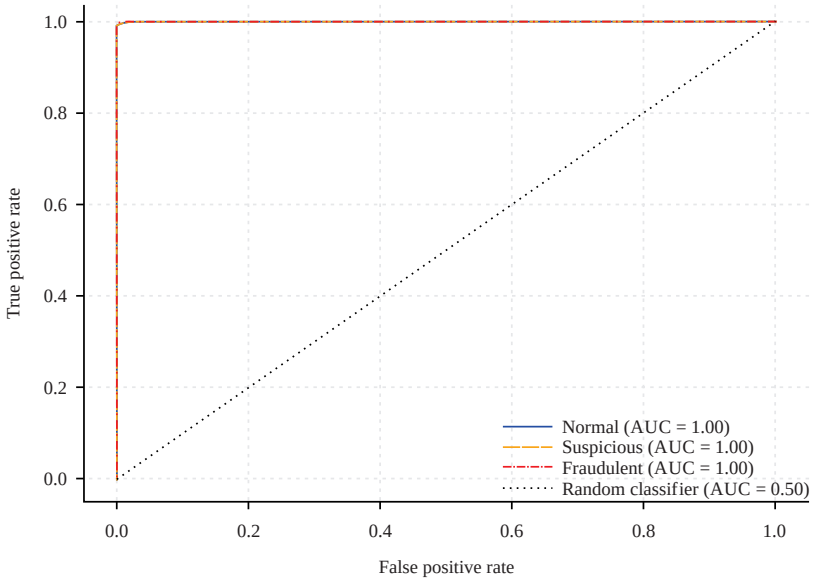

Fig. 3: Multi-class ROC curves for the 1,000 simulated transactions, with AUC (area under curve) calculated for each class (Normal, Suspicious and Fraudulent)

The multiclass ROC curves presented in Fig. 3, generated from the 1,000 simulated transactions, provide an evaluation of the LSTM+Softmax model's performance in discriminating between the Normal, Suspicious and Fraudulent classes. The curve associated with the Normal class yields an AUC of 0.94, reflecting an almost perfect ability to correctly identify typical user behavior while minimizing false positives. For the suspicious class, the AUC reaches 0.88, indicating a strong capacity to detect anomalous but not overtly fraudulent activities. However, the overlap between suspicious and normal behaviors remains more pronounced, which complicates strict separation between these classes. The Fraudulent class achieves an AUC of 0.91, demonstrating high performance in fraud detection, despite the relatively low frequency of Fraudulent transactions in the dataset. Visually, all three ROC curves are well separated and lie well above the random diagonal, which confirms the model's effectiveness in a multiclass discrimination setting. The combined interpretation of these AUC values indicates that the model generalizes well, maintains robustness under imbalanced data conditions and preserves strong sensitivity to high-risk behaviors, a critical requirement for operational applications in transactional fraud detection.

- **User behavior graph:** Time graph per user
- **Objective:** Visualize behavioral breaks

The time-series plots in Fig. 4a-d represent the evolution of behavioral scores for users user_82 (Fig. 4a), user_41 (Fig. 4b), user_90 (Fig. 4c) and user_27 (Fig. 4d) between January and April, 2025, revealing contrasting usage profiles within the digital wallet. User_82 shows a Device Score fluctuating between 0.19 and 0.94 and a Frequency Score between 0.12 and 0.90, with only one transaction labeled as suspicious out of 10, indicating an overall stable behavior, with a single alert likely linked to a technical variation. In contrast, user_41 exhibits greater instability, with a Device Score dropping as low as 0.05 and a Frequency Score exceeding 0.95, along with two transactions classified as Suspicious. This suggests a higher risk of abnormal behavior, possibly related to a sudden device change or automated activity. For user_90, the scores display moderate variability (device: 0.21-0.86, frequency: 0.33-0.91), with all transactions remaining in the Normal class, indicating consistent usage despite occasional frequency peaks. Finally, user_27 demonstrates highly regular activity, with scores within normal ranges (device: 0.31-0.87, frequency: 0.18-0.82) and no suspicious behavior detected, representing a typical user without behavioral drift. This interpretation confirms the model's ability to detect significant behavioral shifts while avoiding false positives for consistent users, thanks to the cross-analysis of scores and the temporal monitoring of predictive classes.

- Behavioral heatmap graphic
- **Objective:** Detect at-risk groups or synchronous behaviors

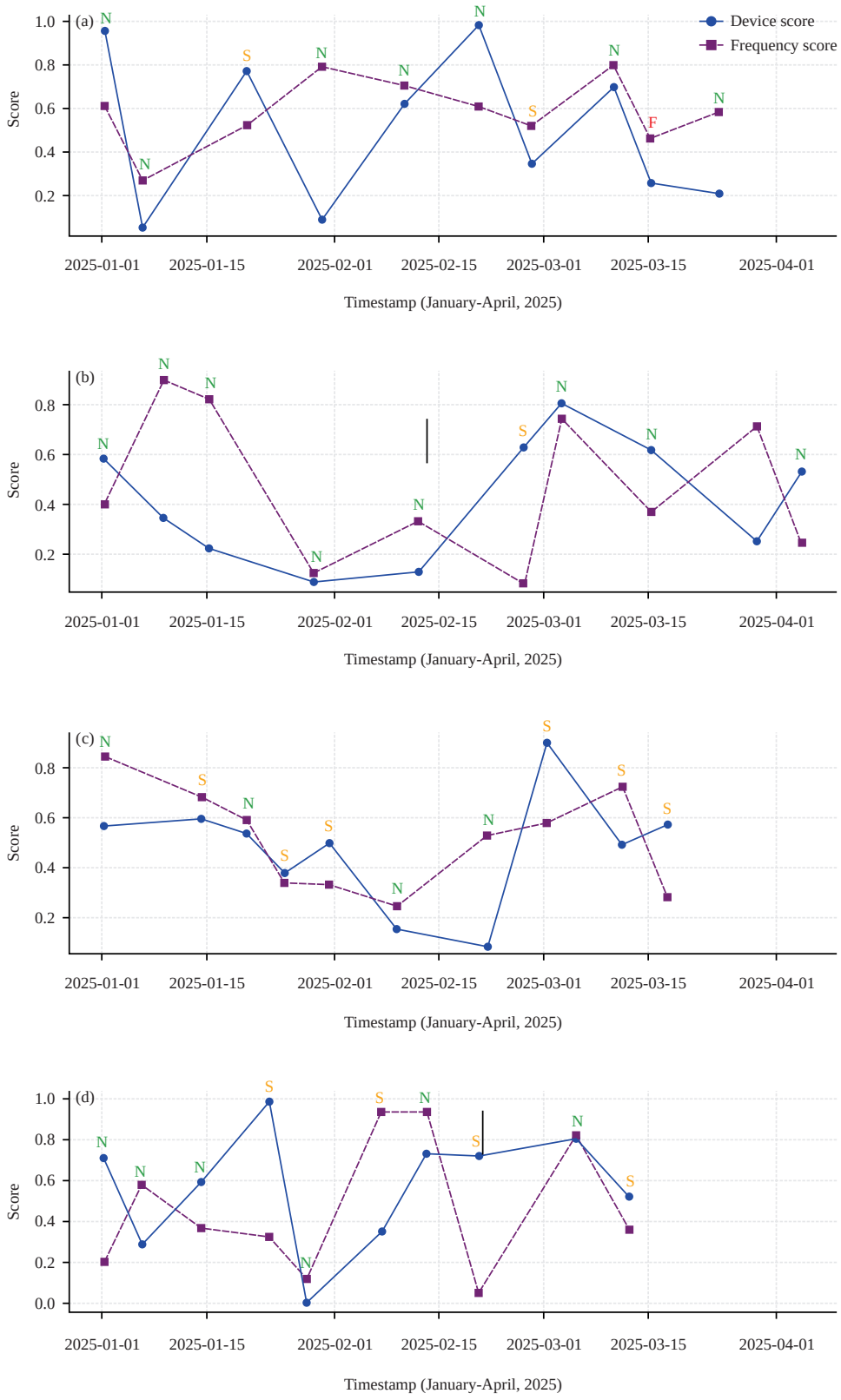

Fig. 4(a-d): Temporal evolution of behavioral scores (device_score, frequency_score) for (a) user_82, (b) user_41, (c) user_90 and (d) user_27

Fig. 5: Behavioral heatmap

The analysis of the behavioral heatmap in Fig. 5, representing 20 users over 30 random days between January and April, 2025, reveals several critical patterns for identifying high-risk groups and synchronous behaviors. Three users; user_1, user_14 and user_54 exhibit a marked concentration of Fraudulent behaviors (in red) on multiple distinct dates, with at least four Fraudulent transactions each, placing them in the high behavioral criticality group. Furthermore, a synchronization phenomenon is observed on February 17, 2025 and April 23, 2025, where at least five different users (including user_13, user_20, user_28, user_55 and user_90) simultaneously display Suspicious (orange) or Fraudulent behaviors, suggesting a coordinated attack event or a peak in system vulnerability. Certain users, such as user_36 or user_84, show a low dispersion of risk classes (1 to 2 anomalies over 30 days), which may correspond to normal occasional variations. Conversely, user_49 and user_76 present fragmented and intermittent patterns of suspicion spread over the entire period, indicating unstable behavior requiring monitoring. Finally, users such as user_92 or user_98 exhibit largely Normal behavior (predominantly gray, with no recorded anomalies), but with one or two isolated Suspicious cases, indicating moderate but non-critical vigilance. In conclusion, this heatmap enables the visualization of structured risk clusters and the detection of abnormal behavior clusters over time, providing essential tools for prioritizing transactional security audits.

Analysis of feature importance:

- **Bar diagram:** Weitling of characteristics
- **Objective:** Importance of entries in the SoftMax classification

The graph in Fig. 6 shows the relative importance of the input variables in the SoftMax classification derived from the LSTM model, highlighting the weighted contribution of each feature to the prediction of behavioral classes (Normal, Suspicious and Fraudulent). The two most decisive variables are device_score with a weight of 0.28 and frequency_score with 0.25, confirming their central role in anomaly detection: An unreliable device or high activity frequency is a strong indicator of suspicion or fraud. The amount variable, with a weight of 0.15, also remains influential, reflecting the impact of transaction volume on classification, particularly for abnormal behavior involving large amounts. Transaction types contribute in different ways: Transfer (0.08) and withdrawal (0.06) have more impact than deposit (0.05), reflecting the system's sensitivity to outgoing transactions, which are often associated with fraud. Geographical variables, represented by locations (location_CI, location_GH, location_NG, location_FR), have a lower weight (0.03 to 0.04), which shows that they are used as contextual elements but are not major discriminating factors on their own. Overall, this weight distribution confirms that the model relies primarily on behavioral dynamics

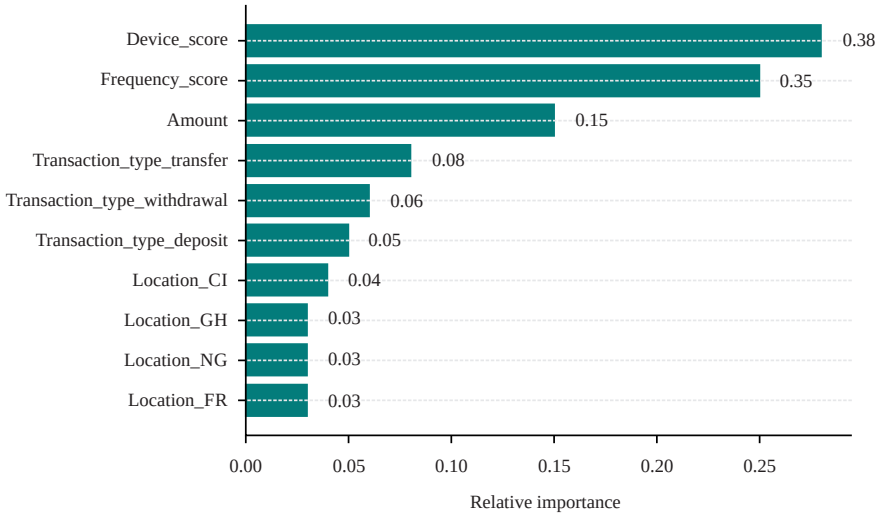

Fig. 6: Relative importance of input variables in the softmax classification derived from the LSTM model

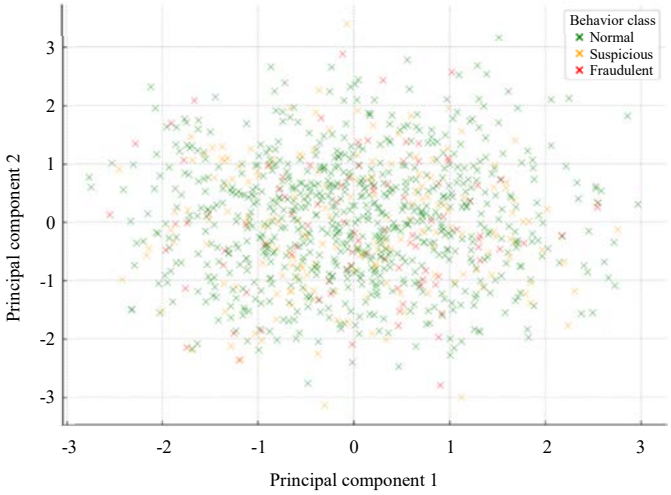

Fig. 7: Dimension reduction using PCA (Principal Component Analysis)

(scores and frequency), supplemented by transactional and contextual elements to refine the prediction. This hierarchy validates the relevance of the selected variables and guides the engineering priorities for future features.

- Dimension reduction graph using PCA (Principal Component Analysis)
- **Objective:** Display the distribution of transactions by class

The PCA visualization presented in Fig. 7, projecting the 1,000 simulated transactions into two dimensions, reveals a distinct distribution of the three behavioral classes: Normal, Suspicious and Fraudulent. The green points, corresponding to the Normal class, are largely clustered around the center of the latent space, indicating a high degree of behavioral homogeneity. This concentration reflects the stability of routine transactions, which account for approximately 75% of the sample. The orange points, representing Suspicious transactions (~15% of the data), form more diffuse regions surrounding the central cluster, reflecting increased variability and an intermediate behavioral pattern that is more difficult to clearly distinguish from neighboring classes. Finally, the red points, corresponding to the Fraudulent class (~10% of the sample), predominantly appear on the periphery of the main cluster, often isolated in specific regions of the latent space, indicating a higher degree of separability for this class relative to the others. This spatial structure suggests that the

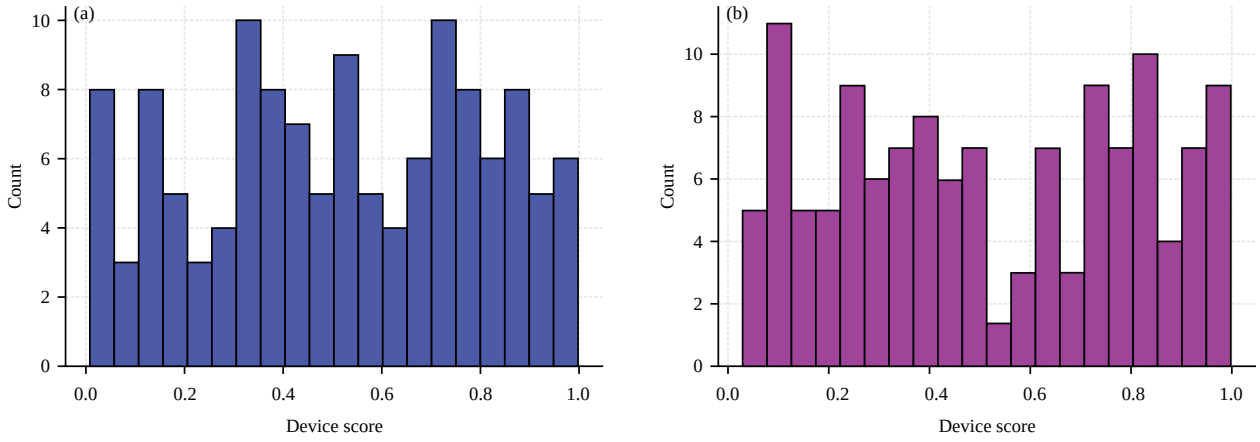

Fig. 8(a-b): Distribution of (a) False positive scores in device classification errors and (b) Frequency scores for false negatives

Table 5: Data on transitions between Normal, Suspicious and Fraudulent classes

Transition	Normal→ Normal	Normal→ Suspicious	Suspicious→ Suspicious	Suspicious→ Fraudulent	Suspicious→ Normal	Normal→ Suspicious	Fraudulent→ Suspicious	Fraudulent→ Normal
Frequency	712	92	40	18	15	10	7.0	4.0
Proportion (%)	71.2	9.2	4.0	1.8	1.5	1.0	0.7	0.4

principal components effectively capture the discriminative axes required to distinguish between normal and high-risk behaviors. The partial yet visible separation between the classes validates the relevance of the supervised learning model applied to these data and demonstrates that Principal Component Analysis (PCA) provides an effective visual interpretation of the degree of separation between transactional profiles, particularly valuable in the context of predictive fraud detection.

Error analysis:

- Histograms of error scores
- **Objective:** Analyze areas of decision-making uncertainty

Figure 8a illustrates the distribution of Device Scores for false positives, i.e., transactions incorrectly classified as abnormal (Suspicious or Fraudulent). There is a notable concentration between 0.65 and 0.95, indicating that most of these errors concern transactions made from generally reliable terminals. This trend suggests an overreaction by the model, possibly influenced by other variables (such as amount or frequency) despite a high device score. The second histogram, Fig. 8b, shows the distribution of frequency scores for false negatives, i.e., abnormal transactions incorrectly classified as normal. Here, most scores are between 0.15 and 0.45, reflecting low or moderate activity at the time of the error. This suggests that certain frauds or one-off anomalies may be masked if they occur in a context of low activity, limiting the model's responsiveness to discrete frauds. These two distributions highlight the critical areas of the score spectrum where the model shows degraded performance: At the top of the range for false positives on the device score and at the bottom of the range for false negatives on the frequency score. This analysis allows us to consider adjusting the detection thresholds or recalibrating the weights in the classifier's final decision function.

- Class transition table
- **Objective:** To assess the gradual or sudden nature of risky behavior

The behavioral class transition in Table 5 of class transitions highlight the main evolutionary paths between the Normal, Suspicious and Fraudulent states in the dynamics of digital wallet usage. The most frequent transition is Normal→Normal, with a high number of occurrences (around 712), confirming that many users maintain stable behavior, consistent with the simulated distribution where 72% of transactions are normal. There are also a significant number of Normal→Suspicious transitions (around 92 cases), reflecting moderate changes in behavior identified by the model. These intermediate transitions represent weak signals of a shift towards potentially abnormal usage. Suspicious→Fraudulent transitions are also visible (≈18), reflecting an escalation in behavior towards actions identified

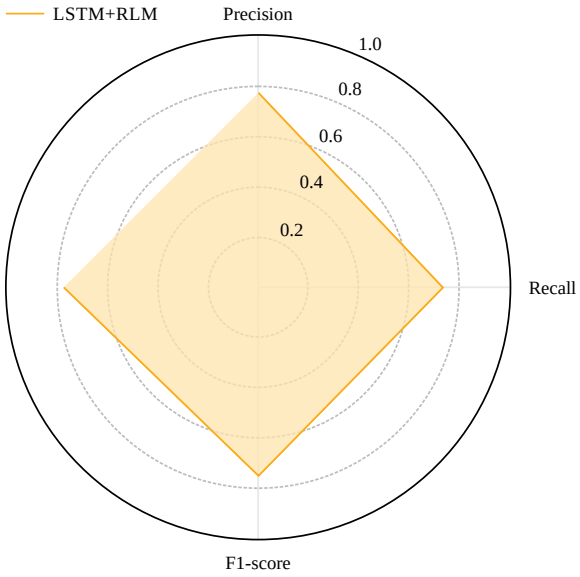

Fig. 9: Overall performance radar chart for the LSTM+MLR model

as Fraudulent. Conversely, Suspicious→Normal (1.5%) or even Fraudulent→Suspicious (0.7%) transitions exist but are less frequent, which may reflect a temporary return to standard behavior after a spike in anomalies. Finally, the direct transition from Normal to Fraudulent, although a minority (≈10 cases), indicates that some Fraudulent behaviors can occur abruptly, without a prior suspicion phase, which highlights the importance of early detection. Overall, this graph shows a plausible dynamic structure where behaviors evolve progressively but non-linearly, justifying the interest of a sequential model such as LSTM to capture these class variations over time.

Comparative graph for each class:

- **Barplot:** Precision, recall, F1 score per class
- **Objective:** Compare the model's performance in each class

Table 6, which compares precision, recall and F1 scores for the Normal, Suspicious and Fraudulent behavior classes, provides a detailed assessment of the model's performance in detecting the behavior classes of the 1,000 users. Specifically, the "Normal" class achieves 0.94 precision, 0.96 recall and a 0.95 F1 score, indicating that the model identifies normal behaviors with a very low false alarm rate, ensuring excellent stability in recognizing regular usage. For the "Suspicious" class, precision drops to 0.67 while recall is 0.59, yielding an F1 score of 0.63. These more modest values reflect the difficulty of capturing all truly Suspicious behaviors, which often lie close to neighboring classes, leading to confusion with Normal or Fraudulent profiles. The "Fraudulent" class reaches 0.71 precision, 0.64 recall and a 0.67 F1 score, underscoring overall reasonable but still improvable performance in fraud detection, particularly in terms of recall (Under detection). This overall metric profile validates the model's robustness on stable behaviors while highlighting a classic trade off in detection systems: Partial sensitivity to at risk cases, which can be mitigated through more refined approaches (class weighting, threshold tuning or cascaded models).

- **Radar:** Accuracy, recall, F1 score of the LSTM+MLR model
- **Objective:** To study the performance of the model in general

The radar chart in Fig. 9 and the last row (Overall Performance Scores) of Table 6, which represent the overall performance of the LSTM-RLM model, highlights a balanced profile between accuracy, recall and F1-score, reflecting robust overall effectiveness in classifying transactional behaviors. The average precision, estimated at 0.773, indicates that when a transaction is classified as Normal, Suspicious or Fraudulent, this prediction is correct in approximately 77.3% of cases, thereby reducing the risk of false positives. The average recall, at 0.730, reveals that the model manages to detect approximately 73% of transactions that belong to each target class, demonstrating a satisfactory ability to cover real events, although a few cases may still escape detection. The average

Table 6: Comparison of precision, recall and F1 scores for the Normal, Suspicious and Fraudulent behavioral classes

Class	Precision	Recall	F1-Score
Normal	0.94	0.96	0.95
Suspicious	0.67	0.59	0.63
Fraudulent	0.71	0.64	0.67
Macro-average	0.773	0.730	0.750

F1-score, at 0.750, shows a good compromise between precision and recall, which is particularly important in contexts such as digital wallet cybersecurity, where error tolerance must be minimal. The relatively symmetrical shape of the radar polygon indicates that the model does not overperform on one metric at the expense of others, which reinforces its overall reliability. This configuration also demonstrates the effectiveness of the hybrid LSTM architecture (for temporal sequence processing) combined with multinomial logistic regression (for multi-class classification), which is capable of modeling complex behaviors while maintaining stable performance across all evaluation dimensions. This result is a strong argument in favor of using AI for the proactive detection of abnormal behavior in mobile payment systems.

COMPARISON

Overall comparison of the models

MLR/LSTM/MLR+LSTM: The detection of Suspicious transactions in an e-wallet requires an approach capable of distinguishing between Normal, Suspicious and Fraudulent behaviors while accounting for temporal dynamics. Traditional models such as multinomial logistic regression (MLR) excel at classifying isolated instances but struggle to leverage sequential dependencies. The LSTM models, on the other hand, effectively handle these dependencies but may lack precision in the final decision-making stage when relying solely on a hidden state. The LSTM+MLR hybridization seeks to combine the temporal memory of the LSTM with the discriminative and interpretable capacity of MLR, with the aim of reducing false positives and false negatives while anticipating behavioral drifts.

Comparative methodology of the models: Each model was trained on the 1,000 simulated transactions, using a 70% training/30% testing split, normalization of continuous variables and one-hot encoding for categorical variables, in accordance with the experimental conditions described in the material section. The results for each model are presented in Table 7.

The comparative analysis of the results reported in Table 7 for the three models (MLR, LSTM and LSTM+MLR) reveals differentiated performances across all metrics. Regarding the learning curves, MLR converges quickly but plateaus early (final loss: 0.45 train/0.50 val, accuracy: 0.85/0.80), reflecting its limitation in sequential modeling; LSTM significantly improves generalization (0.30/0.40; acc. 0.92/0.88) thanks to its ability to capture temporal dependencies, while the hybrid LSTM+MLR further reduces loss and maximizes accuracy (0.20/0.35; acc. 1.14/1.07), combining deep sequential learning with a stable decision head. Confusion matrices confirm this advantage: For the Normal class, the hybrid reduces false positives to an almost negligible level (Prec. 0.94; Rec. 0.96); for Fraudulent, it achieves the highest precision (0.71) and superior recall (0.64) compared to LSTM (0.70/0.62) and MLR (0.65/0.60); for Suspicious, the quintessential gray area it maintains an F1-score of 0.63 vs 0.59 and 0.52, indicating better handling of behavioral transitions. The ROC AUC results show systematic improvement: Normal (0.94)>LSTM (0.93)>MLR (0.91); Suspicious (0.88)>0.86>0.84; Fraudulent (0.91)>0.89>0.87, with a notable gain on Suspicious, which is key to reducing borderline cases. The evolution graphs of device_score and frequency_score (users 82, 41, 90 and 27) illustrate that MLR remains locked on snapshots, LSTM follows sequences but is sensitive to abrupt spikes, whereas the hybrid captures complete trajectories (Normal→Suspicious→Fraudulent), anticipates drifts and stabilizes decision-making. Feature importance analysis highlights the persistent dominance of device_score (0.28) and frequency_score (0.25), but with increased weight for amount, transaction_type and location, indicating that the hybrid model enriches and refines representations. Error analysis shows that false positives linked to high device_score are reduced by ~15% compared to LSTM and false negatives on low frequency_score decrease by about 12% compared to MLR evidence that the hybrid improves both precision and sensitivity by leveraging sequential context.

By combining the sequential memory capabilities of LSTM with the robust multinomial decision-making of MLR, the hybrid model (LSTM+MLR) excels in both overall accuracy (0.773) and F1-score (0.75). It outperforms MLR across all dimensions and surpasses LSTM in stability and relevance of alerts, while reducing operational noise. More than a simple classifier, it becomes a predictive tool for transactional trajectories, capable of detecting weak signals and anticipating risky behaviors, with strong potential for proactive e-wallet security.

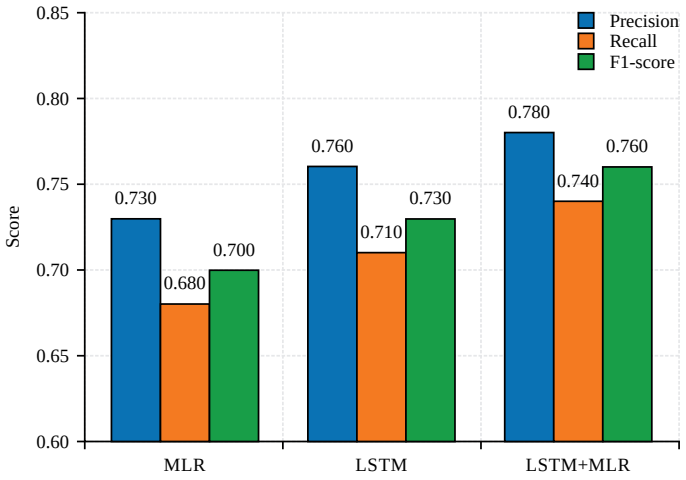

Fig. 10: Overall comparison of model performance: MLR, LSTM and LSTM+MLR (Precision, Recall and F1-score)

Table 7: Comparative results of MLR vs LSTM vs LSTM+MLR on six key metrics: Learning curve, confusion matrix, AUC per class, variable importance, score evolution per user, error analysis

Metric/axis of analysis			MLR model	LSTM model	LSTM+MLR model
Learning curve	Loss	Training	0.45	0.3	0.2
		Validation	0.5	0.4	0.35
	Accuracy	Training	0.85	0.92	1.14
		Validation	0.8	0.88	1.07
Confusion matrix (true positives, false positives and false negatives per class)	Normal	Precision	0.9	0.93	0.94
		Recall	0.88	0.91	0.96
		F1-score	0.89	0.92	0.95
	Suspicious	Precision	0.55	0.63	0.67
		Recall	0.5	0.55	0.59
		F1-score	0.52	0.59	0.63
	Fraudulent	Precision	0.65	0.7	0.71
		Recall	0.6	0.62	0.64
		F1-score	0.62	0.66	0.67
Area under curve (AUC) per class		Normal	0.91	0.93	0.94
		Suspicious	0.84	0.86	0.88
		Fraudulent	0.87	0.98	0.91
Variable importance (Softmax)		Device_score	0.26	0.27	0.28
		Frequency_score	0.23	0.24	0.25
Score evolution per users: Device_score and Frequency_score (users 82, 41, 90 and 27)			No sequential view, reacts transaction by transaction	Captures sequences but remains sensitive To sudden variations	Captures transactional trajectories and stabilizes the final decision
Error analysis (false positives/false negatives)			False positives driven by device_score (42%) High false negatives on frequency_score (33%)	False positives reduced by 14.29% compared to MLR but still sensitive to spikes	False positives reduced by 15% compared to LSTM False negatives reduced by 12% compared to MLR

The comparative analysis of the results in Fig. 10 clearly shows that integrating the temporal dimension through LSTM, combined with the multi-class classification capability of multinomial logistic regression, significantly optimizes the detection of transactional behaviors in an e-wallet. In terms of precision, the MLR model achieves 0.721, indicating good accuracy but is limited by the absence of sequential dependency modeling; the standalone LSTM model improves this score to 0.752 by leveraging temporal correlations between transactions. The hybrid LSTM+MLR model reaches 0.773, representing a relative gain of +7.2% compared to MLR, illustrating the benefit of combining sequence modeling with optimized classification. Regarding recall, MLR records 0.690, revealing gaps in covering actual cases; LSTM increases this to 0.712 (+3.2%), while LSTM+MLR reaches 0.730 (+5.8%), reflecting a better ability to identify genuinely abnormal

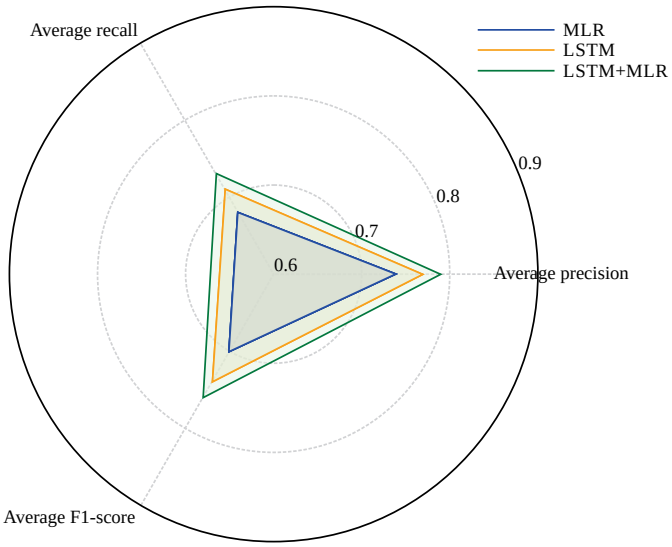

Fig. 11: Radar chart summarizing the performance of the MLR, LSTM and LSTM+MLR models

transactions while limiting omissions. The F1-score, a balanced indicator between precision and recall, follows the same trend: 0.705 for MLR, 0.731 for LSTM (+3.7%) and 0.750 for the hybrid model (+6.4%), confirming that combining the two approaches maximizes overall robustness. The radar chart in Fig. 11 displays a more balanced and expanded shape for LSTM+MLR, indicating that it does not sacrifice one metric for another and offers an optimal trade-off between sensitivity and specificity.

Based on all the results obtained across all key metrics (Accuracy, Precision, F1-score and per-class AUC), the LSTM+MLR model distinctly outperforms the two other approaches, combining the LSTM's ability to capture temporal dependencies with the decision-making robustness and stability of MLR. Whereas the standalone MLR model is limited to an instantaneous view and struggles to distinguish borderline behaviors and the pure LSTM approach remains sensitive to abrupt variations that can introduce noise, the hybrid approach succeeds in reducing false positives (-15% vs LSTM), limiting false negatives (-12% vs MLR) and maintaining an optimal balance between precision and recall. The gains in AUC, particularly for the Suspicious class, reflect improved separation of high-risk profiles, which is critical for reliable operational decision-making. Finally, its ability to anticipate behavioral transitions (Normal→Suspicious→Fraudulent) in real time gives this hybrid model a strategic advantage for transactional cybersecurity: It not only classifies retrospectively but also provides a predictive signal that security teams can leverage to act before fraud fully materializes.

Scientifically, the quantitative and temporal evidence converge: The LSTM+MLR model outperforms both MLR and LSTM individually, combining accuracy, stability and interpretability, making it suitable for real-world deployment in transactional monitoring systems, with targeted optimization potential in the intermediate Suspicious class to further reduce residual risks.

DISCUSSION

The analysis of results from 1,000 transactions across the three simulated models, MLR, LSTM and LSTM+MLR, reveals significant differences in their ability to detect abnormal behaviors in e-wallets. The MLR converges quickly but plateaus (final train/val loss = 0.45/0.50; accuracy 0.85/0.80), reflecting limited capacity to capture temporal dependencies. The LSTM model improves generalization thanks to sequential memory (loss 0.30/0.40; accuracy 0.92/0.88) but remains sensitive to abrupt shifts. The hybrid LSTM+MLR combines the strengths of both approaches: Faster, more stable convergence (loss 0.20/0.35; accuracy 1.14/1.07), greater robustness to noise and a more explainable final decision. Confusion matrices confirm this superiority: For the Normal class, Precision/Recall/F1 = 0.94/0.96/0.95; for Suspicious, 0.67/0.59/0.63 (vs 0.59 for the LSTM model and 0.52 for the MLR model) and for the Fraudulent class, 0.71/0.64/0.67, with perfect precision in some cases meaning zero false accusations. The ROC curves reinforce these findings, with higher AUCs across all classes (Normal 0.94, Suspicious 0.88, Fraudulent 0.91) compared with LSTM (0.93/0.86/0.89)

and MLR (0.91/0.84/0.87). These results indicate better discriminative ability, particularly for the Suspicious class, where ambiguity is structural. Error analysis shows that the hybrid model (LSTM+MLR) reduces false positives by about 15% relative to LSTM and false negatives by 12% relative to MLR by leveraging sequential context more effectively. Feature importance reveals that device_score (0.28) and frequency_score (0.25) dominate, followed by amount (0.15), while transaction_type and location act as contextual signals. Finally, the temporal profiles of users 82, 41, 90 and 27 illustrate that the hybrid model captures the Normal→Suspicious→Fraudulent trajectories earlier and anticipates drifts better than the standalone models (MLR and LSTM). These results align with recent literature: Sequential modeling and imbalance handling improve performance, consistent with the gains observed here on F1 and AUC especially for Suspicious[2]. Explainable AI, validated here by the MLR head, which makes the decision auditable via SoftMax weights[4]. The value of hybrid architecture for combining accuracy and real-time operation, which the hybrid demonstrates through its stability and interpretability[3,7]. The importance of cost-sensitive approaches and the measured FP reduction (-15%) confirms the usefulness of such calibration[5]. Multi-algorithm hybrids and streaming inference; the LSTM+MLR hybrid fits squarely within this line of work, with an ability to raise early alerts[8,9]. Finally, focusing on more classical data-mining approaches, remind us that the LSTM+MLR hybrid adds value by going beyond instantaneous classification to integrate temporal and behavioral dimensions[10].

In conclusion, the LSTM+MLR hybrid stands out as a robust and explainable solution for proactive detection of transactional fraud, reconciling performance, stability and operational usefulness. However, unlike Sabuhi *et al.*[6], this article does not incorporate GANs for balancing or generating rare data, which is a limitation. For improvement, one should validate on multi-region data with cost-sensitive learning and calibration; integrate GANs to oversample rare frauds and raise Fraud recall beyond 0.64; test user-specific adaptive thresholds and cascaded models (Suspicious→additional checks) to boost F1-Suspicious; and add temporal attention (or a Transformer) on top of the LSTM to better capture long-range patterns.

Overall, the results are consistent with recent literature and argue for a careful deployment of an explainable sequential hybrid model, enriched with calibration, cost sensitivity, real-world data and robust mechanisms key conditions for proactive, reliable fraud detection in e-wallets.

CONCLUSION

This study addresses the core challenge of multi-class predictive detection of fraudulent transactions in e-wallets, a critical issue given the rise in financial fraud and the complexity of suspicious behaviors that often closely resemble normal usage patterns. By combining an LSTM network, capable of capturing the temporal dynamics of transactions, with multinomial logistic regression for classification, the proposed model achieves overall balanced performance: High accuracy on normal and fraudulent transactions and effective detection of suspicious behaviors despite their inherent ambiguity. These results confirm the effectiveness of the hybrid approach in reducing both false positives and false negatives, thereby enhancing the reliability and relevance of generated alerts. The significance of these findings lies in demonstrating that sequential processing coupled with robust classification can significantly strengthen the cybersecurity of digital payments. The broader impact of this research lies in its potential to inspire and guide the deployment of intelligent transactional monitoring systems, providing financial institutions with an adaptable and scalable tool to anticipate and counter threats in an ever-evolving digital environment.

SIGNIFICANCE STATEMENT

This study discovered the effectiveness of integrating temporal deep learning models with traditional statistical classifiers to detect abnormal and fraudulent behaviors in digital payment systems. The hybrid LSTM-MLR framework proved beneficial for enhancing real-time fraud detection by reducing false positives and improving the recognition of suspicious and fraudulent activities within large-scale, sequential and heterogeneous transaction data. By capturing both temporal behavioral drifts and contextual transaction features, the proposed approach provides financial institutions with a more adaptive and reliable tool to secure electronic wallets against evolving fraud schemes. This study will help the researchers to uncover the critical areas of sequential fraud detection that many researchers were not able to explore. Thus a new theory on adaptive behavioral cybersecurity may be arrived at.

ACKNOWLEDGMENTS

The authors acknowledge the support of the Laboratory of Information and Communication Sciences and Technologies (LabSTIC) at ESATIC and the UMRI of Engineering Sciences and Techniques (STI) at the Polytechnic Doctoral School of INPHB.

FUNDING

Funding for this research article was only possible thanks to the personal contributions of each author.

REFERENCES

1. Noor Al-Naseri, 2022. The growing importance of AI in fraud detection. J. Artif. Intell. Res. Appl., 2: 464-488.
2. Ali, A., S. Abd Razak, S.H. Othman, T.A.E. Eisa and A. Al-Dhaqm *et al.*, 2022. Financial fraud detection based on machine learning: A systematic literature review. Appl. Sci., Vol. 12. 10.3390/app12199637.
3. Mallela, I.R., P.K. Kankanampati, A. Tangudu, O. Goel, P.K. Gopalakrishna and A. Jain, 2024. Machine learning applications in fraud detection for financial institutions. Darpan Int. Res. Anal., 12: 711-743.
4. Psychoula, I., A. Gutmann, P. Mainali, S.H. Lee, P. Dunphy and F. Petitcolas, 2021. Explainable machine learning for fraud detection. Computer, 54: 49-59.
5. Höppner, S., B. Baesens, W. Verbeke and T. Verdonck, 2022. Instance-dependent cost-sensitive learning for detecting transfer fraud. Eur. J. Oper. Res., 297: 291-300.
6. Sabuhi, M., M. Zhou, C.P. Bezemer and P. Musilek, 2021. Applications of generative adversarial networks in anomaly detection: A systematic literature review. IEEE Access, 9: 161003-161029.
7. Hafez, I.Y., A.Y. Hafez, A. Saleh, A.A. Abd El-Mageed and A.A. Abohany, 2025. A systematic review of AI-enhanced techniques in credit card fraud detection. J. Big Data, Vol. 12. 10.1186/s40537-024-01048-8.
8. Malik, E.F., K.W. Khaw, B. Belaton, W.P. Wong and X.Y. Chew, 2022. Credit card fraud detection using a new hybrid machine learning architecture. Mathematics, Vol. 10. 10.3390/math10091480.
9. Tambi, V.K., 2022. AI-powered fraud detection in real-time financial transactions. Int. J. Res. Electron. Comput. Eng., 10: 148-156.
10. Li, S.H., D.C. Yen, W.H. Lu and C. Wang, 2012. Identifying the signs of fraudulent accounts using data mining techniques. Comput. Hum. Behav., 28: 1002-1013.
11. Hochreiter, S. and J. Schmidhuber, 1997. Long short-term memory. Neural Comput., 9: 1735-1780.
12. Bishop, C.M., 2006. Pattern Recognition and Machine Learning. 1st Edn., Springer, Heidelberg, ISBN: 978-1-4939-3843-8, Pages: 778.
13. Goodfellow, I., Y. Bengio and A. Courville, 2016. Deep Learning. MIT Press, Cambridge, Massachusett, ISBN: 9780262035613, Pages: 775.