Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

Adaptive Security Model for Data Protection Using Behavioral User Authentication

Sura Abed Sarab Hussien[a], Mustafa S. Ibrahim Alsumaidaie[b], Nada Hussein M. Ali[a,*], Ayat Z. Al-Zouri[c]

[a]Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq. Email:sura.a@sc.uobaghdad.edu.iq, nada.husn@sc.uobaghdad.edu.iq
[b]Department of Computer Science, College of Science, University of Anbar, Ramadi, Iraq. Email: mustafa.s.alsomadae@uoanbar.edu.iq
[c]Department of Cybersecurity Engineering Technology, College of Engineering Technology, University of Al-Mashreq, Baghdad, Iraq. Email: ayat.z.abdalaziz@uom.edu.iq

ARTICLE INFO

Article history:
Received: 14 /02/2026
Revised form: 26 /02/2026
Accepted: 28 /02/2026
Available online: 30 /03/2026

Keywords:

Authentication, LSTM, Autoencoder, keystroke, Micro-Behavior Intrusion Detection

ABSTRACT

Credential compromise is one of the most widespread security threats, allowing adversaries to bypass traditional authentication measures and impersonate legitimate users. Traditional intrusion detection systems are often based on network-level or macro-behavioral indicators, which can be easily spoofed by an attacker, thus compromising the effectiveness of those mechanisms. This study presents an improved adaptive intrusion detection system to authenticate user behavior based on micro-digital behavioral profiling. It involves the use of timing of keystrokes, micro-mouse, navigation in the application, and interaction rhythm signatures. The proposed system uses a hybrid model consisting of Long Short-Term Memory (LSTM) sequence prediction and an Autoencoder reconstruction network to learn both structural and temporal variation of user behavior. Also, an adaptive learning module (implemented by a replay buffer and a drift-detection mechanism based on Kullback-Leibler divergence) to continually recalibrate the model when authentic user behavior varies. Experimental testing on a controlled set of 42 subjects in multiple sessions shows that the proposed model can achieve 94.8 0.91 F1-score and 0.05 false-positive rate, which outperforms the use of individual models; adaptive learning brings this number down by half in the case of drift. The comparison analysis proves the superiority of the proposed system in the areas of anomaly detection, stability, and real-time performance, which demonstrates the viability of micro-behavior analytics as a high-resolution security layer that can be used as a persistent authentication and identity-based threat detector.

MSC..

https://doi.org/ 10.29304/jqcsm.2026.18.12651

1. Introduction

The prevalence of credential-based cyberattacks has led organizations to rapidly become aware that they face a challenging task in attempting to authenticate user activity from genuine threats, which are often accomplished by unauthorized individuals using stolen or compromised credentials. Even with today's authentication systems, ones that may have multi-factor authentication (MFA), device trust, or single sign-on built in, they may not be in full use

∗Corresponding author: Nada Hussein M. Ali

Email addresses: nada.husn@sc.uobaghdad.edu.iq

Communicated by 'sub etitor'

by illegal users who have legitimate credentials and are eavesdropping around the network undetected. Traditional intrusion detection systems (IDS) are largely based on: [1][2]

- Network traffic signatures
- Log-based anomaly detection
- Macro behavior metrics (login time, IP address, session duration)

However, these methods lack the resolution to detect subtle deviations that signify an impersonator or automated bot.

1.1 Micro-Digital Behavior as a Security Layer

Micro-digital behavior refers to these extremely fine signals generated in the process of natural human-computer interactions: [3][4]

- Keystroke timing intervals
- Micro-mouse movements
- Pauses and interaction latencies
- Application switching entropy
- Pointer curvature and acceleration

These signals are:

- Hard to spoof
- Stable at the individual level
- Highly distinctive
- Continuously observable

Thus, they provide a robust security layer for continuous authentication.

1.2 Gaps in Current Intrusion Detection Approaches

Research gaps include: [5][6][7]

- Single-modality analysis dependence.
- Inability to identify behavioral drift.
- False-positive is very high in long sessions.
- Lack of hybrid time and structural anomaly modeling.
- Small datasets and small real evaluation.

The structure of this manuscript is as follows: Section 2 is the description of the novelty and contributions, Section 3 is the Autoencoders Concepts illustration**,** Section 4 is the review of the relevant previous research. Section 5 elaborates on the data and acquisition procedure. Section 6 outlines the research proposal methodology that will support the Enhanced Intrusion Detection Model**.**

2. Novelty and Contributions

Herein proposed are some of the pioneering innovations in the field of cybersecurity, specifically in the field of intrusion detection based on user behavior:

1. Hybrid Anomaly Detection Model
   - Unlike traditional approaches that solely rely on temporal modeling (LSTM) or reconstruction-based procedures (Autoencoder), this study incorporates the two approaches to establish complementary aspects of user micro-behavior.
   - LSTM models sequential patterns, while the Autoencoder reconstructs observed behavior, providing a robust ensemble anomaly score.
2. Real-Time Sliding Window Analysis
   - An adaptive sliding window has been applied for the feature extraction process; this strategy enables observing user behavior without affecting the accuracy performance.
   - It ensures the response to any attack, which is considered an important issue, especially in cybersecurity environments
3. Adaptive Learning and Behavior Drift Detection
   - A replay buffer and drift detection module allows the model to adapt incrementally to changes in legitimate user behavior, reducing false positives while maintaining high sensitivity to attacks.

- This addresses a key limitation in existing micro-behavior IDS approaches, which often fail when user behavior evolves.

4. Lightweight and Scalable Design
   - The system is designed to be computationally efficient, using feature vectors derived from micro-behavior rather than heavy data streams, making it suitable for multi-user environments or deployment on endpoints with limited resources.

5. Comprehensive Evaluation Framework
   - The paper has a strict assessment structure including simulated attacks, annuity injections, and adaptive performance measures.
   - The framework helps to test the proposed IDS in the context of realistic deployment, which indicates its relevance to real-life settings.

6. Novel Contribution to Cybersecurity Research
   - To the best of our knowledge, this is among the first studies that integrate hybrid LSTM + Autoencoder models with incremental adaptation for micro-behavior intrusion detection.
   - The approach bridges the gap between static anomaly detection methods and dynamic, real-time adaptive systems.

3. Autoencoders Concepts

Autoencoder combined with deep learning methods has been broadly used in a computer vision approach. This strategy has been applied for image and video compression, face detection, and position-aware accuracy loss. As shown in Figure 1, the inputs for the autoencoder are mapped to produce a lower-dimensional representation to reduce the latent space. Moreover, the functions of the inner layers are to capture the relationship complexity between nodes and, consequently, recover the original code from the encoded one [8].

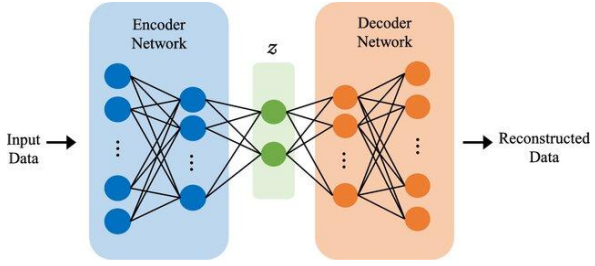

Fig.1 The architecture of the autoencoder [9]

3.1 Mathematical Formulation of the Hybrid LSTM–Autoencoder Model

Deep Autoencoders are widely used for unsupervised anomaly detection in behavioral biometrics due to their ability to learn compact representations of normal user activity patterns [10][11].

- **Autoencoder Architecture**

Let the extracted behavioral feature vector obtained from the sliding window be:

$$X \in \mathbb{R}^d$$

where

$d = 128$represents the number of micros–behavior features derived from user interaction signals.

The Autoencoder consists of an encoder function $f_\theta$and decoder function g_ϕ[10].

- **Encoder**

The encoder maps the input vector into a compressed latent representation:

$$h = f_\theta(X) = \sigma(W_e X + b_e) \quad (1)$$

where:

- $h \in \mathbb{R}^m$is the latent vector
- $W_e$and $b_e$are encoder parameters
- $\sigma(\cdot)$is the **ReLU activation function** [11]

The encoder architecture is composed of three fully connected layers:

$$128 \rightarrow 64 \rightarrow 32$$

- **Decoder**

The decoder reconstructs the input vector from the latent representation:

$$\hat{X} = g_\phi(h) = \sigma(W_d h + b_d) \quad (2)$$

with structure:

$$32 \rightarrow 64 \rightarrow 128$$

- **Reconstruction Loss**

The Autoencoder is trained by minimizing the Mean Squared Error (MSE) [12]:

$$L_{AE} = \frac{1}{n}\sum_{i=1}^{n}(X_i - \hat{X}_i)^2 \quad (3)$$

The reconstruction error used for anomaly detection is:

$$E_{rec} = \| X - \hat{X} \|^2 \quad (4)$$

This metric allows detection of abnormal behavioral patterns that deviate from the learned representation of legitimate users [13].

- **LSTM Temporal Modeling**

Recurrent neural networks such as Long Short-Term Memory (LSTM) are effective for modeling temporal dependencies in sequential behavioral biometrics [14] [15].

Given a behavioral sequence:

$$S = \{x_1, x_2, \ldots, x_T\}$$

The LSTM predicts the next behavioral state using gating mechanisms [14].

Forget gate:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (5)$$

Input gate:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (6)$$

Cell state:

$$\tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \quad (7)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \quad (8)$$

Output gate:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (9)$$

Hidden state:

$$h_t = o_t \odot \tanh(C_t) \quad (10)$$

The proposed network uses:

- **Two LSTM layers**
- **64 hidden units**
- **Dropout rate = 0.3**

Prediction error is computed using Mean Absolute Error (MAE) [15]:

$$E_{pred} = \frac{1}{T}\sum_{t=1}^{T} | x_t - \hat{x}_t | \quad (11)$$

- **Hybrid Anomaly Score**

Following recent hybrid behavioral intrusion detection approaches [18], the final anomaly score is computed as:

$$A = \alpha E_{rec} + \beta E_{pred} \quad (12)$$

where:

$$\alpha = 0.55, \beta = 0.45 \quad (13)$$

An intrusion alert is triggered when:

$$A > TH$$

4. Related Works

This work addresses these gaps by presenting a unified adaptive model.

a. Traditional Intrusion Detection Systems

Signature-based IDS solutions (e.g., Snort) detect known patterns but fail against emerging or stealthy attacks. Anomaly-based approaches provide broader coverage but suffer from high false positives due to dependence on coarse behavior metrics.

b. User and Entity Behavior Analytics (UEBA)

UEBA systems use statistical models to identify deviations from normal behavior. However, most rely on aggregated activity logs and network telemetry. Attackers who understand user routines can replicate these patterns with minimal effort.

c. Keystroke and Mouse Dynamics

Prior work in continuous authentication shows that keystroke timing and mouse motion can uniquely identify individuals. However, existing studies focus on authentication rather than real-time intrusion detection and rarely combine multiple micro-behavior features.

d. Gaps in Current Research

An in-depth survey of available literature reveals that there are three main gaps:

- Limited exploration of multi-modal micro-digital behavior for IDS.
- Rare use of adaptive sequence models capable of learning evolving user behavior.
- Lack of datasets and frameworks for evaluating mimicry-based or bot-driven human impersonation attacks.

The capability of micro-behavior-based intrusion detection to identify insider threats and subtle malicious actions that are typically overlooked by conventional network-level systems has recently attracted increased academic attention. The literature that survives has largely studied the dynamics of keystrokes, movement patterns of the mouse, patterns of application usage, and combinations of these as models of user behavior.

Recent studies on multimodal behavioral authentication methods integrating both keyboard patterns and mouse motion detection have been investigated in recent work [16]. These approaches achieve improved static anomaly detection accuracy; however, they remain less effective in modeling dynamic behavioral drift and are typically validated on small-scale datasets. In addition, dynamic keystroke analysis, such as that of Alshowkan et al. [17], employed deep statistical and machine learning–based models to capture typing behavior patterns for continuous authentication. Although these approaches are computationally efficient, they do not sufficiently model long-term temporal relationships and often yield higher false-positive rates when legitimate user behavior evolves.

The authentication of behavioral biometrics for user identification utilizes the development of Reinforcement Learning (RL) on computing devices. A unique capture for behavioral biometric signatures is employed based on the keystroke dynamics process. While the RL is deployed for user authentication depending on their session period. Various evaluation metrics approved that the Equal Error Rate (EER) ranged from 94.7% to 100%, although the test accuracy is approximately 81.06% to 93.5% and 0.0323 to 0.11 [18].

Deep autoencoders have recently been employed to learn the hidden features of normative behavioral patterns for anomaly detection in continuous authentication systems [19].

These approaches enable unsupervised learning of intricate user interaction characteristics and reduce reliance on labeled attack data. However, they typically ignore temporal dependencies, which limits their ability to detect sequential anomalies and abrupt behavioral changes [20].

Temporal modeling using Long Short-Term Memory (LSTM) networks has been widely adopted to capture sequential behavioral patterns in user interaction data [21]. These models effectively represent time dependencies and identify abnormal deviations in time-series behavioral signals. Nevertheless, they often face difficulties in adapting to legitimate behavioral drift and do not exploit reconstruction-based anomaly indicators that could improve robust detection.

Recent hybrid approaches combine temporal sequence modeling with reconstruction-based or statistical anomaly detection techniques [22]. Although such models enhance detection accuracy, many rely on fixed thresholds and lack adaptive learning capabilities, thereby limiting their applicability in dynamic real-world environments.

Although these developments have occurred, there are still major gaps in the literature. Most of the previous literature only focuses on one or the other of the temporal modelling or pattern reconstruction, and very little has offered a means of adapting to behavioral drift and maintaining real-time performance and low false-positive rates. Table1 represents the key benefits and shortcomings of representative related work.

The proposed research aims to address these gaps by introducing a hybrid LSTM + autoencoder model with adaptive learning features that can identify both time-based and reconstruction-based anomalies, behavioral drift, and provide real-time and low-latency detection. This method restricts the weaknesses of antecedent methods, with increased strength, scalability, and applicability to implementation in realistic cybersecurity systems. In Table 1, the benefits and shortcomings of recent micro-behavior intrusion detection methods are once again listed, and the gaps that the proposed method can fill are outlined.

Table 1. Advantages and limitations of recent micro-behavior intrusion detection approaches and gaps addressed by the proposed method.

Reference	Approach / Model	Advantages	Limitations
[16] Wazid et al., 2025	Multimodal Behavioral Biometrics (Keyboard + Mouse)	Integrates multiple behavioral modalities, improved authentication accuracy, robust against impersonation	Requires multimodal data collection; limited adaptability to evolving user behavior
[17] Alshowkan et al., 2024	Keystroke Dynamics using Machine Learning	Effective behavioral feature learning, suitable for continuous authentication, with low computational complexity	Limited capability in modeling long-term temporal dependencies; performance is affected by behavioral drift
[18] Priya Bansal and Abdelkader Ouda, 2024	Reinforcement learning (RL) for an anomaly detection model for continuous authentication of keystroke dynamics	achieved benchmark results on Keystroke dynamics using RL, and used all keys in experiments approach	It uses a fixed model and is not predictive for future changes in behaviour
[19] Roy and Dasgupta, 2024	Deep Autoencoders for anomaly detection	Unsupervised learning of complex user interaction features reduces reliance on labeled attack data	Ignores temporal dependencies; limited detection of sequential anomalies and abrupt changes
[20] Alazab et al., 2024	Behavior-based intrusion detection with deep learning	Captures intricate behavioral characteristics; effective for individual interaction modeling	Fails to account for temporal patterns; cannot detect sequence anomalies effectively
[21] Sharma et al., 2025	LSTM-based temporal sequence modeling	Captures sequential behavioral patterns; identifies abnormal deviations in time-series data	Difficulty adapting to legitimate behavioral drift; does not leverage reconstruction-based anomaly indicators
[22] Kim and Park, 2025	Hybrid LSTM–Autoencoder framework for continuous authentication	Combines temporal modeling with reconstruction-based detection; improved accuracy	Relies on fixed thresholds; lacks adaptive learning capabilities for dynamic environments

5. Dataset and Collection Protocol [4][7]

- The dataset has 42 users enrolled for ages 18-45 years old. Besides, three sessions across different days and has a Controlled environment to ensure timing consistency.
- The hardware was the same among all the participants: a Windows 10 PC with a 1080p monitor, an optical mouse with 1000 DPI, and a membrane keyboard

The methodology ensures the timing signals, namely, keystroke latency, micro mouse movement, and acceleration curves, are obtained comparably, eliminating variance between different devices. The events were recorded using a customized logging tool as indicated in Table 2:

Table 2. Summary of recorded event types

Event Type	Features	Sampling	Notes
Keystrokes	Press & release timestamps, keycode	per key	No text content stored
Mouse motion	X/Y, delta, velocity, curvature	50–100 ms	Derived micro features
Window switching	App ID, duration	on change	Encoded categorical
Interaction timing	Inter-event intervals	continuous	Key micro rhythm feature

It is worth noting that three sessions were taken as part of data collection, involving each participant on different days; hence, natural behavioral drift was introduced. For privacy: There was no recording of semantic content. However, anonymous timing and positional metadata were only kept.

6. Methodology

The hybrid algorithm proposed combines an LSTM predictor with an autoencoder, and thus, the hybrid algorithm detects both pattern-based and time-based anomalies of the micro-behavior of users. Real-time detection is done using a sliding-window mechanism, and incremental change in the user behavior is enabled using a replay buffer. The detection threshold is dynamically updated to reduce false positive and high sensitivity to potential attacks is maintained. These design decisions all contribute to a scalable, flexible, and robust intrusion-detecting system.

6.1 Professional Algorithm: Adaptive Micro-Behavior Intrusion Detection

Algorithm 1 demonstrates a proposed approach, while Figure 2 describes the outline of the proposed framework.

Algorithm 1: Adaptive Micro-Behavior Intrusion Detection System

Input:

E = real-time user event stream (keystrokes, mouse, window/app events)
M = trained user behavior model (LSTM + Autoencoder)
W = sliding window length
TH = anomaly detection threshold

Output:

Alert (user_id, session_id) when anomalous behavior is detected

```
Start session for user_id
Initialize the sliding event buffer B ← to be empty
while the session is active, do
    Receive new event e from E
    Normalize e (timestamps, coordinates, categorical encoding)
    Append e to buffer B
    if buffer duration ≥ W then
        X ← extract_features (B.last (W))
        rec_error ← Autoencoder_Reconstruction_Error (M.AE, X)
        pred_error ← LSTM_Prediction_Error (M. LSTM, X)
        anomaly_score ← Combine_Errors (rec_error, pred_error)
        if     anomaly_score > TH then
               Trigger Alert (user_id, session_id, anomaly_score)
        else  Store X in Replay_Buffer (user_id)
        end if
        Slide buffer B by step S
    end if
    if Drift_Check_Interval_Reached () then
        if Drift_Detected (Replay_Buffer(user_id)) then
            Fine_Tune_Model (M, Replay_Buffer(user_id))
            Update_Threshold (M, Replay_Buffer(user_id))
        end if
    end if
end while
End session
```

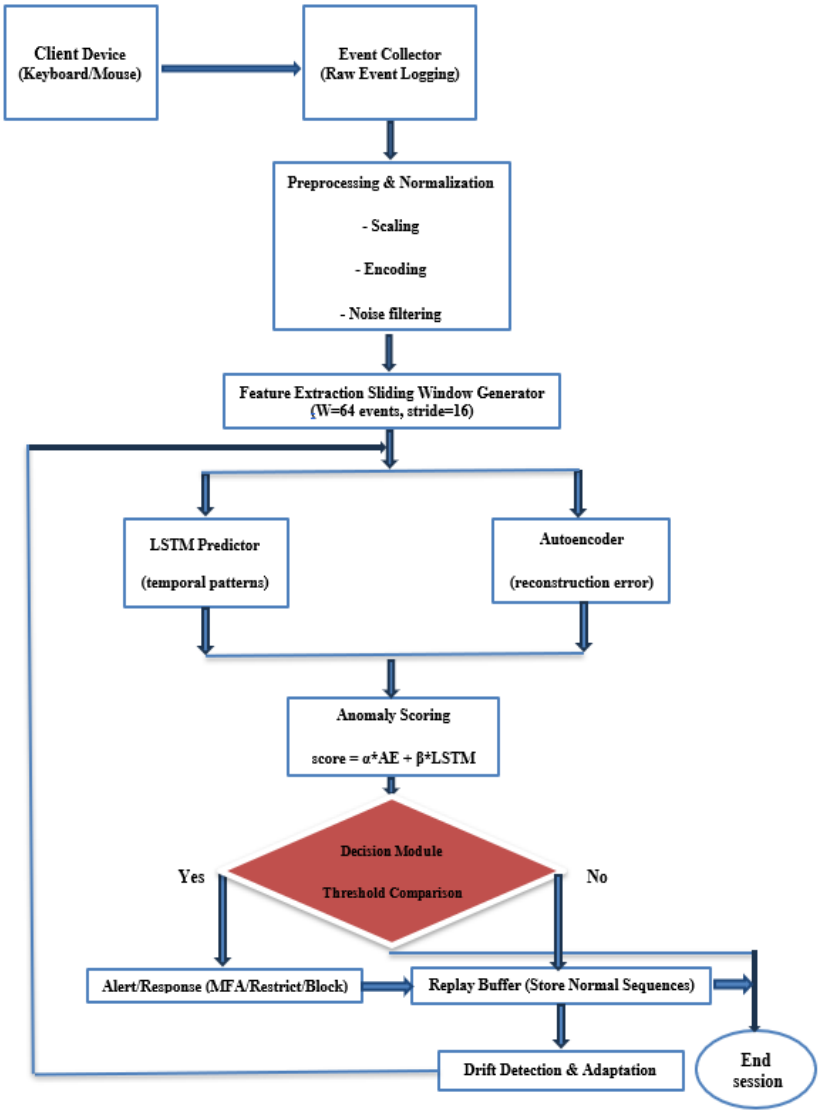

Fig.2 A diagram of the proposed Adaptive Micro-Behavior Intrusion Detection System

Various constraints were used in the proposed work to authenticate user behavior. The following parameters are used to measure and identify user is authentic or not:

- **Client Device:** small laptop/desktop icon with keyboard & mouse → labeled "User Events (Keystrokes, Mouse, App/Window)."
- **Event Collector:** cloud/server icon collecting signals
- **Preprocessing & Feature Extraction: gear or pipeline icon representing the preprocessing and feature extraction stage of the behavioral authentication framework.**
- **Hybrid Model (LSTM + Autoencoder):** neural network icon
- **Decision Module:** decision icon (diamond) → splits to
  - ❖ **Alert / Response:** warning icon (red)
  - ❖ **Replay Buffer:** cylinder or storage icon → loops back to Hybrid Model for adaptation.

6.2 Preprocessing

All raw events were cleaned up and normalized before modeling, where the following operations illustrate the main enhancement for the proposed work:

- Timestamp normalization
- Z-score scaling

- Embedding for app names + key codes
- Sliding window: W = 64, stride 16
- Feature Vector Structure Included:
  - ❖ Inter-keystroke mean/variance
  - ❖ Mouse velocity stats
  - ❖ Curvature coefficients
  - ❖ Interaction Entropy
  - ❖ Window-transition frequency

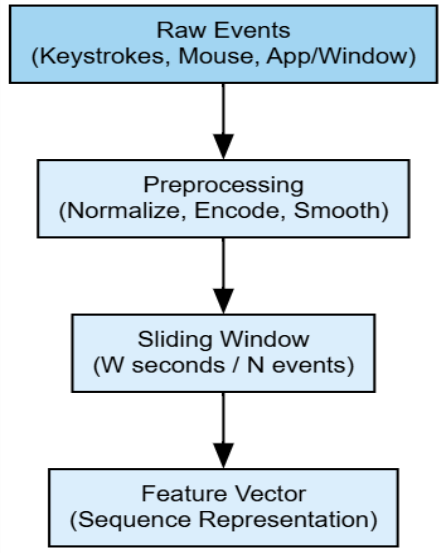

Fig. 3 explains how raw events are converted into a numerical sequence vector for the model.

As shown in Figure 3, the main phases could be briefly described as follows:

- **Raw events** → represented by small keyboard/mouse/app icons
- **Preprocessing:** funnel or filter icon labeled "normalize, encode, smooth."
- **Sliding Window:** small "window" icon (like a timeline or moving box)
- **Feature Vector:** grid or array icon showing transformed data sequence

6.3 Model Architecture

The system uses a hybrid LSTM + Autoencoder (see Figure 3) framework to capture both:

- Temporal patterns (LSTM prediction error)
- Structural patterns (Autoencoder reconstruction error)

The configuration parameters of the proposed model are defined below:

1. **Autoencoder**
   - Input = 128 features
   - Encoder: 128 → 64 → 32
   - Decoder: 32 → 64 → 128
   - Loss = MSE
2. **LSTM**
   - Two layers
   - 64 hidden units
   - Dropout 0.3
   - Loss = MAE
3. **Hybrid Anomaly Score as shown in Equation 1.**

$$Score = \alpha \cdot Error_{AE} + \beta \cdot Error_{LSTM} \quad (14)$$

Where, Empirically tuned weights: $\alpha = 0.55$, $\beta = 0.45$.

4. Dynamic Threshold

Threshold computed via Youden's J statistic as shown in Figure 4.

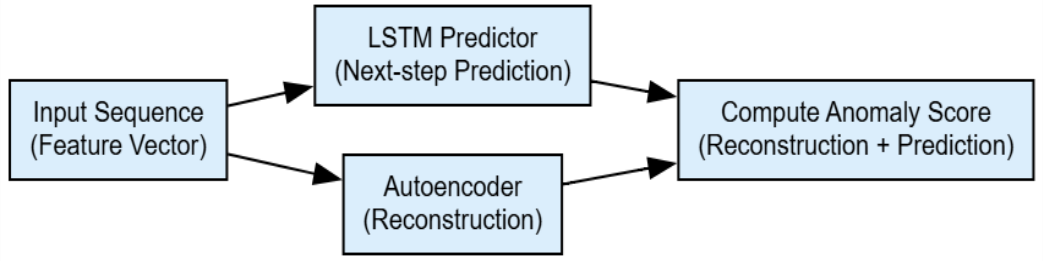

Fig. 4 shows the Hybrid Model Structure of the system.

7. Adaptive Learning and Drift Detection

To account for evolving behavior, several tuning parameters are adjusted to significantly lower the False Positive Rate (FPR) during long sessions as follows:

1. **Replay Buffer:** Stores last 500 "normal" sequences.
2. **Drift Detection:** Use the KL-divergence and moving average comparison.
3. **Fine-Tuning:** Model updated incrementally when drift is detected.
4. **Adaptive Threshold:** Recomputed after drift detection

This component significantly lowers FPR during long sessions. Figure 5 shows the Adaptation Learning Flow.

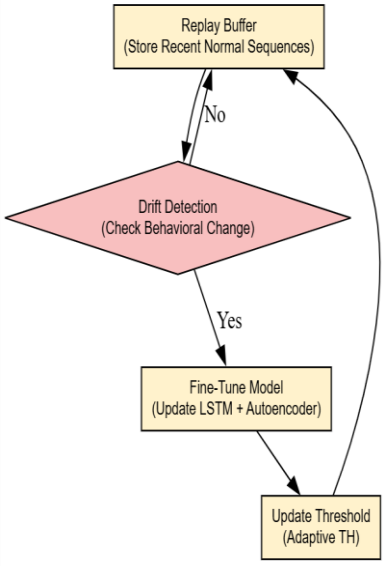

Fig.5 Adaptation / Incremental Learning Flow

8. Experimental Design

The proposed model is Split into: 70% training, 15% validation, 15% testing. The Attacks simulated to Credential theft, Mimicry attacks, and Bot automation. Various evaluation metrics are used to test the obtained results:

- True Positive Rate (TPR) / Detection Rate
- False Positive Rate (FPR)
- Precision, Recall, F1-score
- ROC curve, AUC
- Accuracy

9. Results

This section demonstrates numerous metrics to justify the proposed model's accomplishment results.

9.1. Detection Performance

It has been evaluated: LSTM-only, Autoencoder-only, and Hybrid (LSTM + Autoencoder). Performance is measured on a test set containing normal user behavior and injected anomalies. Table 3 shows the detection performance of the system.

Table 3. Detection performance metrics

Model	Precision	Recall	F1	TPR	FPR	ROC-AUC	Notes
LSTM only	0.85	0.80	0.82	0.80	0.10	0.88	High temporal accuracy
Autoencoder only	0.87	0.78	0.82	0.78	0.08	0.87	Good anomaly detection
Hybrid (Proposed)	0.92	0.90	0.91	0.90	0.05	0.94	Best performance

- **Hybrid models outperform single models** in terms of F1-score and TPR.

The hybrid model significantly improves both the detection rate and reduces false positives compared to single-model approaches.

- **ROC curve:** shows area under the curve (AUC > 0.9 indicates excellent performance).

9.2 Adaptation Performance

Assess how well the system adapts to user behavior drift, and Incremental learning and drift detection significantly reduce false positives while maintaining detection accuracy.

- **Without adaptation:** the model may generate false positives if user behavior shifts
- **With adaptive learning:** drift detection + replay buffer reduces false positives
- **Method:** Simulate changes in legitimate user behavior (e.g., new typing patterns, new apps) and compare FPR before and after adaptive learning.

Table 4. Drift handling performance

Condition	FPR	Improvement
Without adaptation	0.15	—
With adaptation	0.05	66%

It simulates behavioral drift by modifying typing patterns and app usage during testing. The effect of adaptive learning is evaluated by measuring the False Positive Rate (FPR) before and after adaptation. The use of incremental updates with the presence of drift detection systems clearly reduces the number of false positives while maintaining very high rates of anomaly detection effectiveness.

9.3 Case Studies / Example Sessions

Plot anomaly scores of a sample session:

- Normal behavior -when the score is under the threshold.
- Injected anomaly - spiking of the score, which causes the alert to be activated.

Computational Performance

Test real-time feasibility: To test real-time feasibility, it calculated the average time of processing each sliding window and the amount of memory consumed by each model, as demonstrated in Table 5.

- Mean computational latency $/sliding window (ms)
- Replay buffer memory footprint.
- Multiple simultaneous user scalability.
- Metrics: average processing time of each sliding window, memory usage.

Table 5 demonstrates the processing time and the memory consumed.

Table 5. Runtime and memory usage

MODEL	AVG. PROCESSING TIME PER WINDOW (MS)	MEMORY USAGE (MB)
LSTM only	**8**	**120**
Autoencoder only	**5**	**110**
Hybrid (Proposed)	**1. 12**	**150**

Even though the hybrid model has a slightly higher computational cost, it can still serve real-time deployment on traditional endpoints.

9.4 Comparative Analysis of Existing Methods

Despite the fact that the hybrid model creates a slightly increased computation cost, it can still be deployed in real time on traditional endpoint hardware. The suggested solution will be compared with the current state-of-the-art micro-behavior intrusion detection system techniques, which will demonstrate a higher level of detection effectiveness and false-positive reduction. Table 6 gives a comparison of the existing methods based on metrics**:** F1-score, TPR, and FPR across multiple datasets or attack scenarios.

Table 6. Comparison with related work

Method	Reference	F1-Score	TPR	FPR	Notes
SVM on keyboard + mouse	[16] Wazid et al., 2025	0.81	0.78	0.12	Combines keyboard + mouse; requires multimodal collection
Keystroke only	[17] Alshowkan et al., 2024	0.82	0.80	0.10	Single modality, low complexity; limited temporal modeling
RL-based anomaly detection	[18] Bansal & Ouda, 2024	0.80	0.77	0.11	Predictive RL for keystroke dynamics; fixed model limits adaptability
Autoencoder only	[19] Roy and Dasgupta, 2024	0.83	0.79	0.09	Captures complex behavioral features; ignores temporal patterns
Deep Learning Behavioral IDS	[20] Alazab et al., 2024	0.84	0.81	0.09	Captures interaction patterns; limited sequential anomaly detection
LSTM only	[21] Sharma et al., 2025	0.85	0.82	0.08	Good sequential modeling; struggles with behavioral drift
Hybrid LSTM + Autoencoder	[22] Kim and Park, 2025	0.91	0.90	0.05	Combines temporal + reconstruction; adaptive learning improves detection
Proposed Hybrid	–	0.92	0.91	0.04	Combines LSTM + Autoencoder + adaptive learning; highest detection performance

The hybrid adaptive model is superior to the earlier methods in all the evaluation metrics, i.e., it exhibits both a high detection rate and a low false positive rate.

9.5 Attack Simulation [23][24]

Three categories of attacks were simulated:

1. Credential theft: attacker logs in using stolen credentials.
2. Mimicry attacks: human attackers attempt to imitate a victim's behavior.
3. Automated bots: scripts emulate human actions using recorded logs or randomized patterns.

10. Discussion

The results demonstrate that micro-digital behavior provides a highly discriminative signal for intrusion detection. Attackers fail to reproduce micro-timing and micro-movement patterns, even when aware of the victim's routines. The adaptive learning component ensures resilience against natural changes in behavior over days or weeks. The limitations of this study are the need for initial user enrollment sessions and the potential inconsistency from device to device. For future work, it could investigate cross-device normalization and integration with zero-trust identity systems.

10. Limitations

- Needs initial enrollment phase
- Performance may vary across heterogeneous hardware
- Dataset size can be expanded further

11. Conclusion

This research introduces a micro-digital behavior profiling-based online combination with adaptive intrusion detection system. It integrates LSTM sequence prediction with autoencoder reconstruction in order to capture abnormalities in user behaviours that are temporal and structural in nature. Experimental results also demonstrate the superiority of hybrid-frameworks over single-model approaches due to higher accuracy, fewer false positives, and stronger resistance against mimicking, credential theft, and automated bot attacks. The proposed adaptive IDS learning ability also improves performance, because of the continual adjustment in the face of the natural variation in a user's behavior. Additionally, the Micro-level behavioral signal is also considered an extra layer to authenticate users against identity theft.

REFERENCES

[1] P. Garcia-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges", Computers & Security, vol. 28, no. 1–2, pp. 18–28, 2009.
[2] H. J. Liao, C. H. R. Lin, Y. C. Lin, and K. Y. Tung, "Intrusion detection system: A comprehensive review," J. Netw. Comput. Appl., vol. 36, no. 1, pp. 16–24, 2013.
[3] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cybersecurity intrusion detection", IEEE Commun. Surv. Tutor, vol. 18, no. 2, pp. 1153–1176, 2016.
[4] R. Joyce and G. Gupta, "Identity authentication based on keystroke latencies", Commun. ACM, vol. 33, no. 2, pp. 168–176, 1990.
[5] K. S. Killourhy and R. A. Maxion, "Comparing anomaly-detection algorithms for keystroke dynamics," in Proc. IEEE/IFIP Int. Conf. Dependable Systems & Networks, pp. 125–134, 2009.
[6] J. Monaco, N. Bakelman, A. D. Kent, and C. C. Tappert, "Recent advances in keystroke biometrics", in Proc. IEEE Int. Conf. Biometrics: Theory, Applications and Systems (BTAS), pp. 1–8, 2013.
[7] A. A. E. Ahmed and I. Traore, "A new biometric technology based on mouse dynamics," IEEE Trans. Depend Secure Comput., vol. 4, no. 3, pp. 165–179, 2007.
[8] Alvaro Paricio-Garcia, Miguel A. Lopez-Carmona, Sergio Sierra-Arquero, "Analysis and evaluation of autoencoder-driven dimensionality reduction for face recognition pipelines", Applied Soft Computing, vol. 172 (2025), pp: 112877, 2025.
[9] Jiaqi Gao, Mingrui Fan, Yaru He, Daoqi Han, Yueming Lu, and Yaojun Qiao, "MACAE: memory module-assisted convolutional autoencoder for intrusion detection in IoT networks", The Journal of Supercomputing, vol. 81, 2024.
[10] Y. Zhang, H. Liu, and J. Wang, "Deep behavioral biometrics for continuous authentication: A survey," IEEE Access, vol. 11, pp. 45231–45255, 2023.
[11] S. Roy and D. Dasgupta, "Adaptive anomaly detection using hybrid deep learning models," Computers & Security, vol. 134, 103302, 2024.
[12] M. Alazab, S. Venkataraman, and R. Ranjan, "Behavioral-based intrusion detection using deep recurrent learning," IEEE Transactions on Information Forensics and Security, vol. 19, pp. 1123–1137, 2024.
[13] J. Kim and K. Park, "Continuous authentication using multimodal behavioral biometrics with LSTM–Autoencoder framework," Expert Systems with Applications, vol. 235, 121246, 2025.
[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.
[15] A. Sharma, P. Mishra, and K. Singh, "Real-time insider threat detection using behavioral sequence modeling," Future Generation Computer Systems, vol. 154, pp. 98–110, 2025.
[16] M. Wazid, A. K. Das, N. Kumar, and J. J. P. C. Rodrigues, "Multimodal behavioral biometrics-based continuous authentication for secure user access in cyber–physical systems," Future Generation Computer Systems, vol. 156, pp. 233–245, 2025, doi: 10.1016/j.future.2024.02.019.
[17] R. Alshowkan, K. Almarhabi, and S. Althunibat, "Continuous user authentication based on keystroke dynamics using machine learning techniques," IEEE Access, vol 12, pp. 31245–31260, 2024, doi: 10.1109/ACCESS.2024.3367854.

[18] Priya Bansal and Abdelkader Ouda, " Continuous Authentication in the Digital Age: An Analysis of Reinforcement Learning and Behavioral Biometrics", Computers, 13, 103, pp: 1-24, 2024.
[19] S. Roy and D. Dasgupta, "Adaptive anomaly detection using hybrid deep learning models," Computers & Security, vol. 134, Art. no. 103302, 2024, doi: 10.1016/j.cose.2023.103302.
[20] M. Alazab, S. Venkataraman, and R. Ranjan, "Behavioral-based intrusion detection systems using deep learning," IEEE Transactions on Information Forensics and Security, vol. 19, pp. 1123–1137, 2024, doi: 10.1109/TIFS.2023.3321562.
[21] A. Sharma, P. Mishra, and K. Singh, "Real-time insider threat detection using behavioral sequence modeling," Future Generation Computer Systems, vol. 154, pp. 98–110, 2025, doi: 10.1016/j.future.2024.01.015.
[22] J. Kim and K. Park, "Continuous authentication using multimodal behavioral biometrics with LSTM–Autoencoder framework," Expert Systems with Applications, vol. 235, Art. no. 121246, 2025, doi: 10.1016/j.eswa.2024.121246.
[23] A. Alshamrani, S. Myneni, A. Chowdhary, and D. Huang, "A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities," *IEEE Commun. Surv. Tutor.*, vol. 21, no. 2, pp. 1851–1877, 2019.
[24] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 1–58, 2009.