STAP
Smart Technologies Academic Press

Journal of Cyber Security and Risk Auditing

https://www.jcsra.thestap.com/

Student Identity Fraud Detection in Online Exams Using Keystroke Dynamics: A Comparative Study of Classical Machine Learning and Deep Learning Models

Khadija Alhumaid[1*]

[1] *Research & Innovation Division, Rabdan Academy, Abu Dhabi, UAE*

ARTICLE INFO

Article History

Received: 10-02-2026
Revised: 03-03-2026
Accepted: 25-03-2026
Published: 31-03-2026

Vol.2026, No.1

DOI:

Corresponding author.
Email:
kalhumaid@ra.ac.ae

Orcid:

ABSTRACT

The process of conducting online examinations has created a new security threat which allows students to use their authentic identity verification systems to impersonate others for assessment completion. The paper presents an identity verification system which uses keystroke dynamics as its core mechanism. The approach creates a biometric typing signature by analyzing password typing through latency and hold-time features. A benchmark study is conducted using the CMU Keystroke Dynamics dataset (DSL-StrongPasswordData). The researchers trained multiple classical Machine Learning (ML) models and Deep Learning (DL) architectures to identify multiple student groups while measuring their performance with Accuracy, Precision, Recall, F1-score, and ROC curves. The classical ML methods achieved better results than DL methods when tested on the tabular behavioral biometric dataset. The Random Forest model achieves the best performance with Accuracy = 94.07%, Precision = 94.22%, Recall = 94.07%, and F1 = 94.03%. The results demonstrate that recurrent architectures outperformed convolutional architectures because BiGRU achieved an accuracy of 88.14%. Advanced visualization techniques enable users to see how identity similarity patterns and behavioral drift and fraud risk separation exist in the data. The findings support the deployment of keystroke-based authentication as a low-cost additional security layer for academic integrity in remote proctored examinations.

Keywords: Academic integrity, biometrics, deep learning, keystroke dynamics, machine learning, online exams, and student identity fraud.

How to cite the article

1. Introduction

Higher education institutions now use online examinations as their primary assessment method because distance learning and blended delivery methods and large-scale digital assessment platforms have expanded since their initial introduction [1], [2]. The implementation of remote testing methods has created a significant academic integrity problem because students now commit identity theft to take exams [3]–[5]. The attacker assumes the role of a student by using their stolen identification or through collusion to take the exam [6], [7]. The traditional authentication methods which include username and password login with ID upload and one-time proctor validation do not protect against unauthorized access because they only establish static identity verification at the start of the test and do not monitor users throughout the entire testing period [8]. Online exam environments require identity verification solutions which need to verify test takers through multiple checks while keeping operational costs down and maintaining complete system visibility control [9], [10].

The solution to this problem finds its most promising answer through keystroke dynamics technology. The system establishes a behavioral biometric system that uses three specific timing patterns to create a user's typing rhythm profile [11], [12]. Keystroke dynamics can use standard keyboards during typical user operations because it needs no extra hardware requirements, which makes it perfect for implementation in large-scale learning management systems [13], [14]. The initial benchmarking studies created testing standards with data collections that enabled researchers to evaluate different techniques for keystroke authentication and anomaly detection [9], [15], [16]. The CMU keystroke dynamics dataset from the DSN 2009 study by Killourhy and Maxion [9], [17] functions as a standard testing resource that researchers use to evaluate keystroke authentication systems and their performance metrics in various research fields. The same benchmarking system has become accessible to the public as a dataset that researchers use to study keystroke verification and identification processes.

Recent research shows that keystroke dynamics work for both authentication during logging in and continuous identity authentication which the system uses to verify user identity throughout their session [18], [19]. Keystroke based methods have shown significant security promise in realistic testing settings and are evaluated with metrics such as FAR, FRR, and EER to articulate verification trade-offs. Students in online tests are required to sustain their identity for such tests which addresses the necessity of Continuous Identity verification to do this makes keystroke dynamic a consistent solution with respect to testing period. On the one hand, reduced computational and storage overheads are challenges whereas optimizing on performance guarantees security in e-learning environments and online assessments while research describes how keystroke biometrics works during these processes. [13].

Researchers studied keystroke biometrics through two different research methods which included traditional machine learning techniques and modern deep learning methods. Through the time feature investigation SVM and Random Forest models and boosting techniques discovered their ability to create non-linear decision boundaries. Researchers developed deep learning models for keystroke signals through various methods which utilized recurrent architectures like LSTM and GRU to produce effective embedding results. [21-24]. In fact, existing research conducted in the field of keystroke dynamics consist of both broad analyses and sophisticated techniques due to its progressive development within the cybersecurity domain especially concerning access control implementations and behavioural authentication mechanisms. Previous works show that performing deep learning techniques on authentication can potentially improve accuracy, yet both data representation and model selection methods have significant effects on results.

It is evident that more comparative experimental studies in the field are necessary to test traditional ML and DL methods under like-for-like conditions to evaluate their effectiveness at detecting student identity fraud when exploiting well-established benchmark datasets. Current examination systems require identity verification of the candidates to be conducted on an ongoing basis without compromising their privacy whereas current methods rely too heavily, however, on video monitoring which brings up privacy issues and increases operational expenses. A keystroke-based system acting as a security mechanism which activities are what online test procedures can utilize to verify users through keyboard usage.

No physical faculty in remote learning environments means the problem of identity fraud during online Exams is still a challenge. The traditional authentication system that is based on a common set of login credentials and static authentication factors could not thwart both identity thefts and advanced cybersecurity threats. The current research on keystroke dynamics has limitations because it only investigates binary verification and does not include testing for

traditional machine learning techniques and deep learning methods. The development of an evaluation framework which assesses everything from identification to verification needs to combine all elements into a single integrated system. The study establishes and evaluates an identity theft detection system based on keystrokes which has been developed specifically for online assessment platforms. The research aims to create a multi-class identification system which will use a verification scoring system to perform performance evaluations of machine learning and deep learning algorithms while showing understandable behavior analysis results for fraud risk assessment.

This paper makes the following contributions. The first contribution of the study introduces an active fraud detection system for online examinations which uses keystroke patterns to identify test takers and verify their identity through a single operational procedure. The second contribution of the study offers an extensive benchmarking assessment which tests traditional ML algorithms SVM, KNN, LR, NB, DT, RF, GB, MLP and DL models CNN LSTM, BiLSTM, CNN+BiLSTM, GRU, BiGRU against the DSL-StrongPasswordData dataset. The third project provides a performance assessment which uses macro-averaged metrics to measure Accuracy Precision Recall F1-score and conducts ROC analysis while using visualization methods to study identity separation and similarity mapping and typing stability and entropy-based fraud risk assessment.

The research investigates biometric authentication through behavioral methods which use keystroke timing data from a publicly accessible dataset. The assessment takes place in a multi-class identification environment which includes supplementary tests designed for verification purposes. The research only investigates controlled dataset testing because it does not include live system tests or multiple biometric system tests.

The remainder of this paper is organized as follows. The dataset description together with preprocessing methods and experimental procedures are presented in Section 2. Section 3 presents the evaluated ML and DL models and the evaluation metrics. Section 4 reports experimental results, which include tables and visual analytics for classification and ROC curves and loss curves and fraud-risk separability. Section 5 examines how the findings affect the operational use of online examination systems. The paper ends with Section 6, which presents research limitations and future research directions.

2. Methodology

In this section, we describe the dataset used, preprocessing pipeline and feature representation along with model design and evaluation procedure followed to build and test the proposed student identity fraud detection framework. The approach generates a real-life online examination system that demands confirmation identity of the student using their typing behavior patterns.

2.1 System Overview and Threat Model

The developed system takes an online examination setup as a common environment, and the students are authenticated using LMS account and take exam remotely. The primary security concern being addressed is the impersonation of student by a rouge (masquerading) user, for accomplishing an online exam. The taxonomy of fraud detection attacks that we study in this research is shown in Figure 1. Identity fraud in online examinations can be boiled down to impersonation attacks (e.g. using false identities), behavioral imitation (e.g. reproducing another person's behavior), replay attempts and adaptive adversarial behaviors, which is shown in the following diagram: Their corresponding detection mechanism-both identity verification, behavior analysis and sequence anomaly detect, dynamic risk-are explicitly mapped to attack category in the proposed framework. This framework mapping brings clarity on how the system is addressing different fraud scenarios with respect to identification and verification level.

Two fraud scenarios are considered:
• Credential sharing fraud: The legitimate student voluntarily shares login credentials with another person.
• Credential theft fraud: The attacker obtains student credentials without consent.

In both scenarios, the attacker can pass static login verification. The proposed approach adds a second security layer based on typing behavior which is difficult to replicate consistently. The proposed model supports two operational modes:
• Identification mode (multi-class classification): The system predicts who typed the input.
• Verification mode (fraud detection): The system checks whether the typing sample matches the claimed identity.

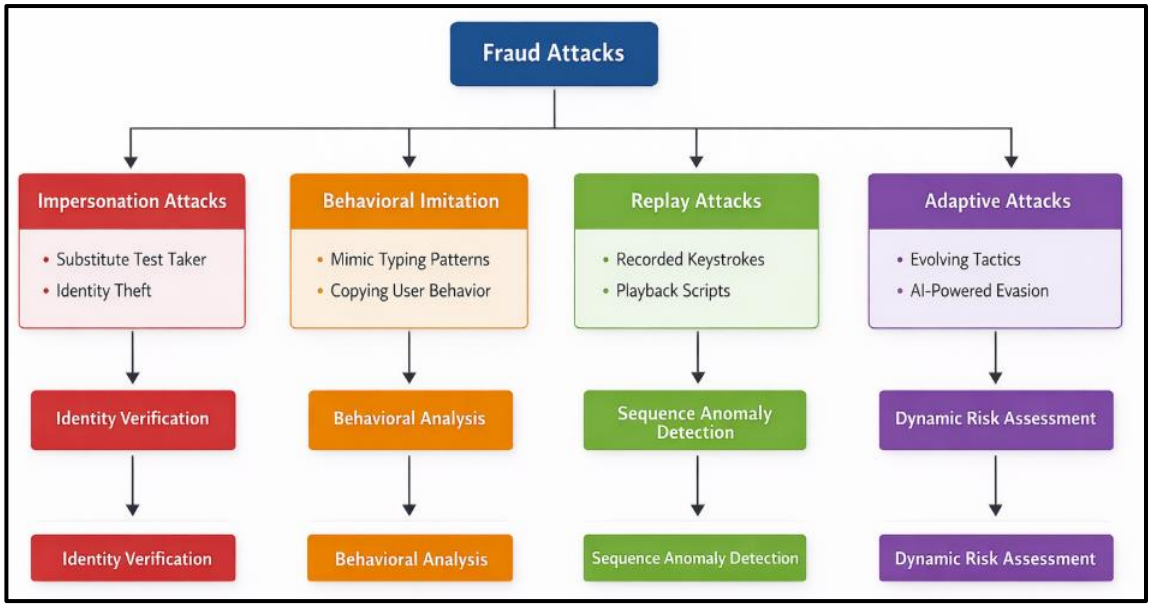

Figure 1. Taxonomy of fraud detection attacks in online examinations

2.2 Dataset Description

Experiments were conducted using the DSL-StrongPasswordData benchmark dataset, part of the CMU keystroke dynamics benchmark widely used in authentication studies [9], [17]. The dataset contains repeated typing samples of a fixed password phrase collected from multiple subjects.

The dataset version used in this work has the following properties:

• Total samples: 20,400
• Total columns: 34
• Number of identities (subjects): 51
• Label column: subject (format: s002, s003, …)
• Session column: session Index
• Repetition column: rep

Each record corresponds to one typing instance of the password. Timing features are stored as continuous numeric values (in seconds) and include:

• Hold time features (H.) Duration of pressing a key.
• Digraph features
• DD.: Down–Down latency (time between key down events)
• UD.: Up–Down latency (time between release of one key and press of the next)

These feature types capture stable behavioral characteristics that can serve as a biometric signature.

2.3 Comparison to Other Keystroke Datasets

Other than the DSL-StrongPasswordData, we have other keystroke dynamics datasets which allow for more general modelling paradigms. In Table 1, we show a brief comparison of these datasets:

Table 1. Comparison to Other Keystroke Datasets

Dataset Name	# Subjects	Sample Type	Features	Labeled	Balance	Key Use Cases
DSL-StrongPasswordData [25]	51	Fixed password typing	34 timing features (H, DD, UD)	Yes	Balanced	Multiclass identification, verification benchmark
KeyRecs [26]	100	Fixed + free-text	Inter-key latencies, session metadata	Yes	Depends on exercise	Anomaly detection, typing behavior analysis
Free-text + Synthesized [27,28]	Various	Free-text + synthetic	Keystroke sequences	Yes	Varies	Liveness detection, forgery analysis

2.4 Data Cleaning and Preprocessing

To achieve optimal learning results while testing different models, scientists developed an entire preprocessing system which they illustrated through Figure 2. The process began with raw keystroke timing data, which required the removal of non-feature metadata columns from the input space during feature extraction process. The system converted all timing features into numeric format, while it treated any invalid data entry as a missing value. Missing values were addressed using median imputation, which provides robustness with respect to outliers in continuous latency features. StandardScaler applied only to distance- and gradient-based models, and tree-based models were not scaled in order to keep their natural invariance capabilities. For multi-class training, labels were encoded and for our deep learning models one-hot encoding was used and to achieve consistent evaluation across all algorithms in general, the dataset is divided into the training-subset as well as testing-subset.

2.4.1 Feature extraction and column filtering

Three columns were treated as non-feature metadata:
• Subject used as label
• Session Index used for drift visualization and optional split analysis
• Rep repetition index
All timing features were retained as input attributes.

2.4.2 Numeric conversion

All feature values were coerced to numeric using strict conversion. Invalid entries were mapped to missing values.

2.4.3 Missing values

Missing values were handled via median imputation, which is robust against outliers and appropriate for continuous timing features.

2.4.4 Feature scaling

Scaling was applied only where needed:

• StandardScaler was applied for distance-based and gradient-based models:

• KNN, SVM, Logistic Regression, MLP (Sklearn)

• DL models

• Tree-based models did not require scaling:

• Decision Tree, Random Forest, Gradient Boosting

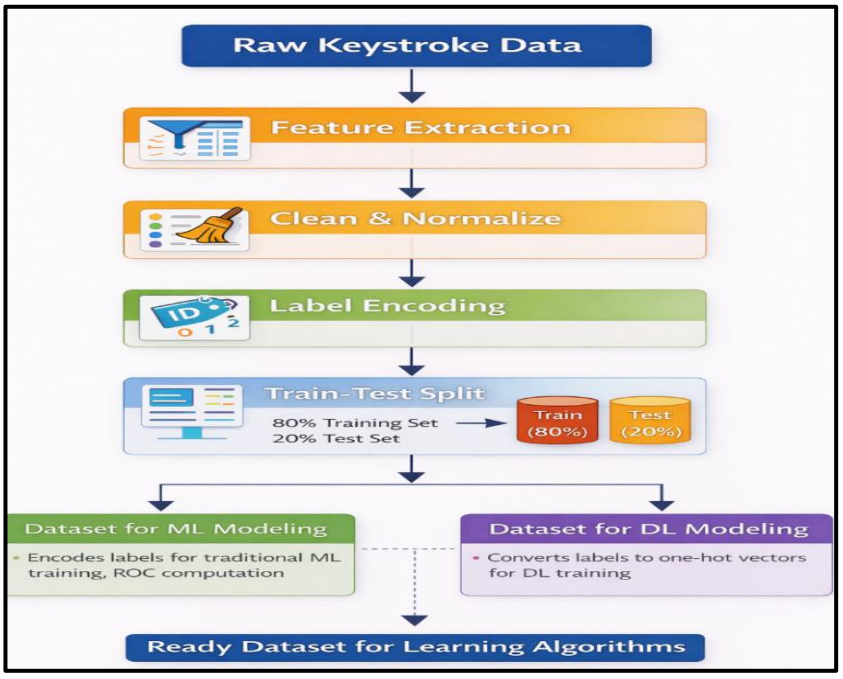

Figure 2. Workflow of keystroke data preprocessing

2.5 Label Encoding

The identity label subject is stored as string values such as s002. To train ML/DL models reliably, the labels were encoded using LabelEncoder to produce integer IDs:

$$y = \text{LabelEncoder}(subject) \quad (1)$$

This generates labels in the range:

$$y \in \{0,1,\ldots,50\} \quad (2)$$

The encoded labels were used for:

• multi-class classification training
• one-vs-rest ROC computation
• DL categorical encoding

For DL training, the labels were converted into one-hot vectors:

$$y_{onehot} = \text{to\_categorical}(y) \quad (3)$$

2.6 Train/Test Split Protocol

A stratified split was adopted to preserve the identity distribution across training and test sets.
• Train set: 80%
• Test set: 20%
• Random seed: fixed for reproducibility

$$(X_{train}, X_{test}, y_{train}, y_{test}) = \text{StratifiedSplit}(X, y) \quad (4)$$

This split design ensures each identity contributes samples to both sets, making the comparison fair across all models.

2.7 Problem Formulation : The methodology supports two related tasks.
2.7.1 Multi-class Student Identification Given a keystroke feature vector:

$$\text{x} \in \mathbb{R}^d \quad (5)$$

The model outputs one identity:

$$\hat{y} = f(\text{x}), \hat{y} \in \{1, \ldots, 51\} \quad (6)$$

This corresponds to exam identity recognition at login or checkpoints.

2.7.2 Verification-Based Fraud Detection
Given a claimed identity y_cand typing sample x, the model generates a match score:

$$s = P(y = y_c \mid \text{x}) \quad (7)$$

A decision threshold τdetermines acceptance:

$$\text{Accept if } s \geq \tau, \text{Reject if } s < \tau \quad (8)$$

2.8 Classical ML Models
A benchmark suite of classical ML models was applied to establish strong baselines and measure performance gaps relative to DL.
2.8.1 Logistic Regression
Multi-class logistic regression is trained to maximize likelihood for identity prediction:

$$P(y = k \mid \text{x}) = \frac{\exp(\text{w}_k^{\top}\text{x})}{\sum_j \exp(\text{w}_j^{\top}\text{x})} \quad (9)$$

This model provides interpretability but assumes near-linear separability.

2.8.2 SVM with RBF Kernel

SVM with RBF kernel learns non-linear boundaries:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \parallel \mathbf{x}_i - \mathbf{x}_j \parallel^2) \quad (10)$$

Probability estimates were enabled to support ROC analysis.

2.8.3 KNN Classifier

KNN classifies using nearest neighbors in standardized feature space:

$$\hat{y} = \text{mode}(y_{kNN}) \quad (11)$$

This model is simple but sensitive to feature scaling and identity overlap.

2.8.4 Naive Bayes

Gaussian Naive Bayes assumes conditional independence among features. Its performance is limited when feature correlations are high.

2.8.5 Decision Tree

Decision tree partitions feature space using greedy splits. It is interpretable but prone to overfitting.

2.8.6 Random Forest

Random Forest aggregates multiple trees trained on bootstrapped samples and random feature subsets:

$$\hat{y} = \text{majority\_vote}(T_1, \dots, T_N) \quad (12)$$

This improves generalization and captures complex feature interactions. It was the best-performing model.

2.8.7 Gradient Boosting

Gradient boosting builds an additive model of weak learners minimizing classification loss iteratively, offering strong performance on tabular features.

2.8.8 MLP (Sklearn)

A multi-layer perceptron provides non-linear modeling with fully connected layers, trained via backpropagation on scaled inputs.

2.8.9 Linear Regression Baseline

Although not suitable for multi-class classification, Linear Regression was added as a baseline to demonstrate the importance of correct task formulation. Predictions were rounded and clipped to valid class IDs.

2.9 DL Model Design

DL models were trained on scaled inputs. Since keystroke features represent sequential timing dependencies, the feature vector was reshaped into a 1D sequence:

$$\mathbf{x} \in \mathbb{R}^d \Rightarrow \mathbf{x}_{seq} \in \mathbb{R}^{d \times 1} \quad (13)$$

Each “timestep” corresponds to a feature.

2.9.1 CNN (1D Convolution)

A 1D CNN extracts local patterns from adjacent features:

- Conv1D layers
- Global Average Pooling
- Dense + Softmax output

This works best when feature ordering corresponds to real local structure.

2.9.2 LSTM

LSTM captures long dependencies via memory gates:
• forget gate
• input gate
• output gate
It models sequential dependencies across keystroke features.

2.9.3 BiLSTM
BiLSTM learns dependencies in both directions:

$$h_t = [\overrightarrow{h_t}, \overleftarrow{h_t}] \quad (14)$$

This enhances representation learning and was one of the best DL models.

2.9.4 CNN + BiLSTM Hybrid
This hybrid uses CNN as local extractor and BiLSTM as global memory model, improving performance compared to CNN alone.

2.9.5 GRU
GRU reduces complexity compared to LSTM using:
• update gate
• reset gate
GRU often converges faster with fewer parameters.

2.9.6 BiGRU
BiGRU combines bidirectional processing and gated recurrence. It achieved the best DL performance.

2.10 Training Strategy for DL
DL training followed these settings:
• Optimizer: Adam
• Learning rate: 1×10^{-3}
• Loss: categorical cross-entropy
• Early stopping:
• Monitor: validation loss
• Patience: 4
• Restore best weights

This approach reduces overfitting and improves generalization.

2.11 Evaluation Metrics

For multi-class identity prediction:
• Accuracy
• Precision (macro)
• Recall (macro)
• F1-score (macro)
Macro-averaging ensures fairness across identities by treating each subject equally regardless of sample count.

2.11.1 ROC Curve (Micro-average OvR)

Since the task is multi-class, ROC was computed using one-vs-rest binarization:

$$ROC_{micro} = ROC(\text{ravel}(Y), \text{ravel}(\hat{P})) \quad (15)$$

This provides one overall separation indicator for all classes and thresholds.

2.12 Visualization and Behavioral Analytics

The study required more evidence than its numerical performance results because multiple behavioral analytics visualizations needed to be developed for analyzing identity patterns and fraud characteristics. The researchers-built identity fingerprint heatmaps to show how different subjects distributed their centroid features among various locations. The research used session drift heatmaps to track user behavior changes which occurred during different session times. The researchers created an authentication similarity matrix which used cosine similarity to evaluate how closely two users exhibited matching patterns of behavior [29-32]. The researchers used PCA projection and t-SNE embeddings as dimensionality reduction techniques to study clustering patterns and class separation in their data. The researchers calculated typing stability index values for each subject because they used feature variance to determine how consistent their typing patterns stayed throughout different sessions. The researchers created a fraud risk map which combined model probability scores with entropy measurements to show both uncertainty and impersonation risk levels. The visualizations show how closely identities behave together with their temporal drift patterns and the stability changes that affect false rejection rates and their ability to distinguish between real and fake attempts.

3. Results and Discussion

The performance analysis of the student identity fraud detection system which uses keystroke dynamics for identifications. The study presents its findings through two distinct model categories which include classical ML and DL. The study uses quantitative metrics together with visual analytics figures to assess identity separability and system stability and identity similarity and fraud detection risk.

3.1 Classical ML Results

Table 2 displays the classification results of nine traditional machine learning models which were evaluated through macro-averaged metrics that included Accuracy and Precision and Recall and F1-score. The task requires the model to identify students from 51 different classes which makes the evaluation process more difficult because the model needs to determine the exact student identity.

Table 2. Classical ML performance comparison on DSL-StrongPasswordData dataset

Model	Accuracy	Precision	Recall	F1
Random Forest	0.940686	0.942226	0.940686	0.940348
MLP (Sklearn)	0.925735	0.926506	0.925735	0.925625
Gradient Boosting	0.919608	0.922587	0.919608	0.920092
SVM (RBF)	0.901716	0.904661	0.901716	0.902054
KNN	0.841912	0.852194	0.841912	0.841520
Logistic Regression	0.839706	0.838464	0.839706	0.837975
Decision Tree	0.724020	0.727910	0.724020	0.724493
Naive Bayes	0.675000	0.693853	0.675000	0.670804
Linear Regression	0.017892	0.019026	0.017892	0.011348

The Random Forest model gave the most robust results above 94% accuracy and a higher macro F1-score. The outcomes effectively show that timing features from keystrokes can identify stable individual patterns which can uniquely characterize 51 distinct identities. The key takeaway is: Random Forest outperforms Decision Tree because ensemble learning yields better results than a single tree-based decision boundary. However, life in the presence of non-linear decision surfaces is needed to identify identity-specific behavioral signals and therefore using Gradient Boosting as a combination with MLP gives promising results. Naive Bayes performs relatively poorly as the Gaussian independence assumption that is applied does not hold for keystroke features. Logistic Regression and KNN scores moderately which is a natural consequence of LR's linear separability limitations and KNN decision rules which rely on distance, variance and overlap in features. Linear Regression shows almost complete failure. The reason for result is that Linear Regression, which generates continuous output cannot express code in discrete multi-class face for step on 51 class boundaries in keystroke space.

3.2 ROC Curve Analysis for Classical ML

The ROC micro-averaged curves of classical ML models can be viewed in Figure 3 which displays One-vs-Rest binarization for 51 classes and micro-averaged results from all class predictions.

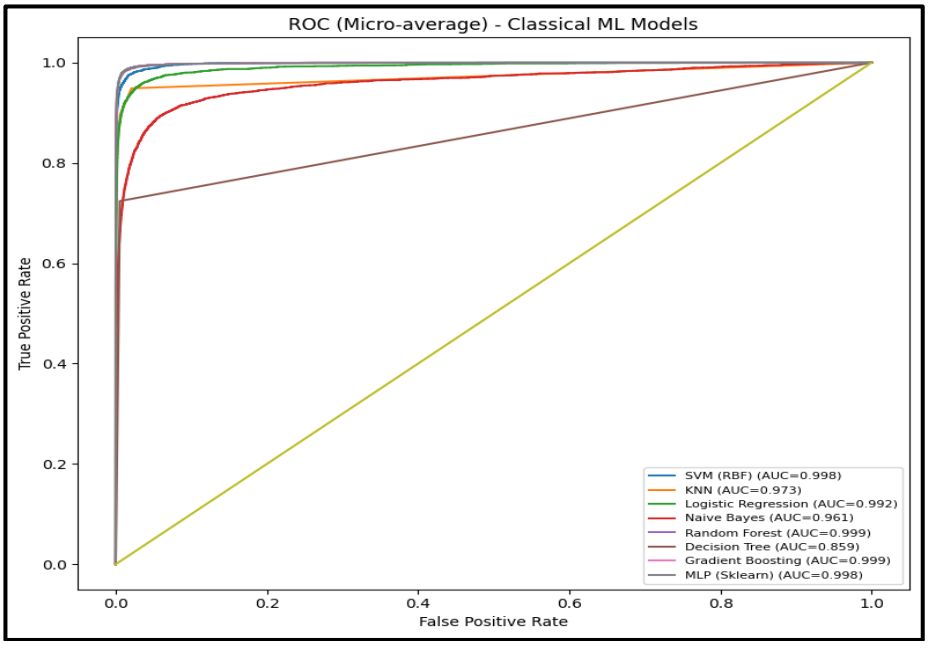

Figure 3. ROC-micro average curves computed for benchmark classical ML models for the 51-class identity classification problem.

The strongest models in Figure 3 which include Random Forest and MLP and Gradient Boosting and SVM, achieve their best performance through ROC curves which reach their peak results at the top-left corner. The models show their ability to differentiate between authentic identity patterns because they maintain accurate performance throughout all decision points. The Decision Tree curve displays its weakest performance because the model tends to over fit which results in unpredictable probability predictions. The results match this performance The ROC curve reveals more information about the model's performance than accuracy because operational authentication relies on threshold-based decisions which determine ROC performance.

3.3 DL Results

Table 3 presents performance data for six DL architectures which were trained using an identical dataset. The DL models were developed by converting tabular keystroke data into sequential feature representations. The 51 classes were used to calculate macro metrics which were then computed.

Table 3. DL performance comparison on DSL-StrongPasswordData dataset

Model	Accuracy	Precision	Recall	F1
BiGRU	0.881373	0.882801	0.881373	0.880276
BiLSTM	0.862500	0.865198	0.862500	0.862244
GRU	0.843137	0.847909	0.843137	0.843403
CNN+BiLSTM	0.823039	0.828992	0.823039	0.822158
LSTM	0.765931	0.773779	0.765931	0.763459
CNN	0.529412	0.524205	0.529412	0.509838

The model displaying the most robust results with the highest accuracy value over 94% was Random Forest and located the best macro F1-score. The findings show that stable individual keystroke timing features characterize 51 different identities. The main finding is that Random Forest outperforms Decision Tree because ensemble learning simulation provides better results than decision boundaries based on a single tree. Accuracy with Gradient Boosting combined with MLP was high because behavioral signals need to be non-linearly separable for distinguishing identities. Naive Bayes performs poorly as the independence assumptions of Gaussians do not accurately model keystroke features. Logistic Regression and KNN show a moderate performance that can be attributed to the linear separability aspect of LR (it does not work in all cases) and the fact that KNN uses distance from mean as decision rules whose effectiveness is dependent on feature variance/overlapping. Now Linear Regression shows near total failure. This happens because Linear Regression generates continuous output and cannot predict the distinct multi-class identity of 51-class limits in keystroke space. The dryland BiGRU (88.14%) is the best-performing DL model, surpassing all other DL models. This indicates the importance of recurrent memory units, especially bidirectional recurrence which models forward and backward dependencies between features. Although these features are handcrafted, they do maintain implicit temporal structure (e.g., hold times and digraph latencies).

CNN alone does badly (≈53% accuracy). This means convolutional locality assumptions (e.g. the order that engineered features are presented to model), is not a good fit for this dataset. And in the case of hybrid CNN-Bi-LSTM, the bondage of both layers improves with relevant outcomes as the CNN stage also acts as a local feature extractor whereas explores global sequence modelling and it results in better identity discrimination.
In summary, DL does not outperform classical ML for this dataset suggesting that (1) engineered tabular features are better exploited with tree ensembles and (2) DL may require much larger datasets or raw keystroke event streams instead of aggregated timing features.

3.4 ROC Curve Analysis for DL Models

All DL architecture presented in Figure 4 ROC micro-averages curve. Recurrent architectures are more separable than CNN-based models as shown in Fig. 2. Both BiGRU and BiLSTM curve shape approaches close to optimal, indicating a clear threshold-based discriminative performance and stable decision confidence level upon prediction. As expected, CNN has the worst ROC curve, due to its low accuracy and macro-F1 in Table 2.

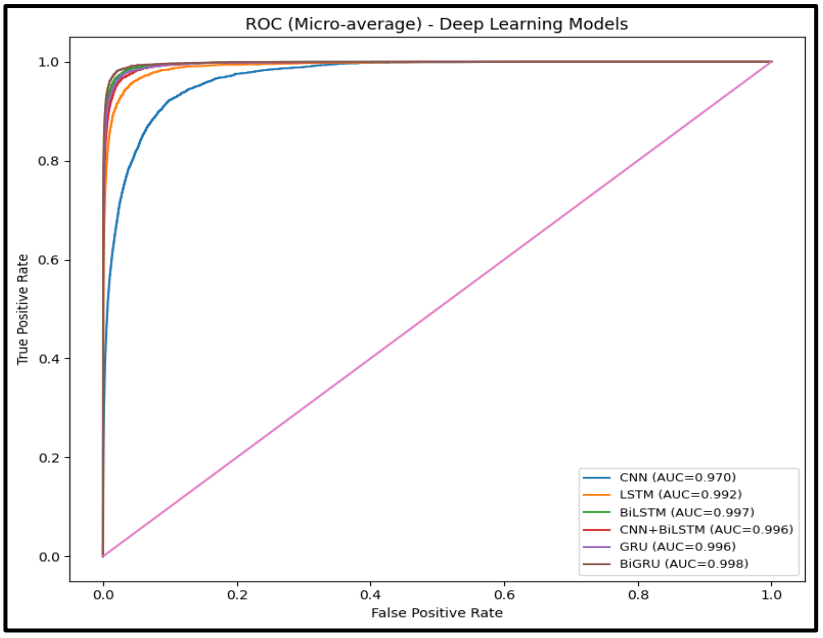

Figure 4. The micro averaged ROC curves for architectures DL (CNN, LSTM, BiLSTM, CNN-Bi-LSTM, GRU, BiGRU)

3.5 Training and Validation Loss Behavior

Figures 5 and 6 report the training loss and validation loss curves for DL models.

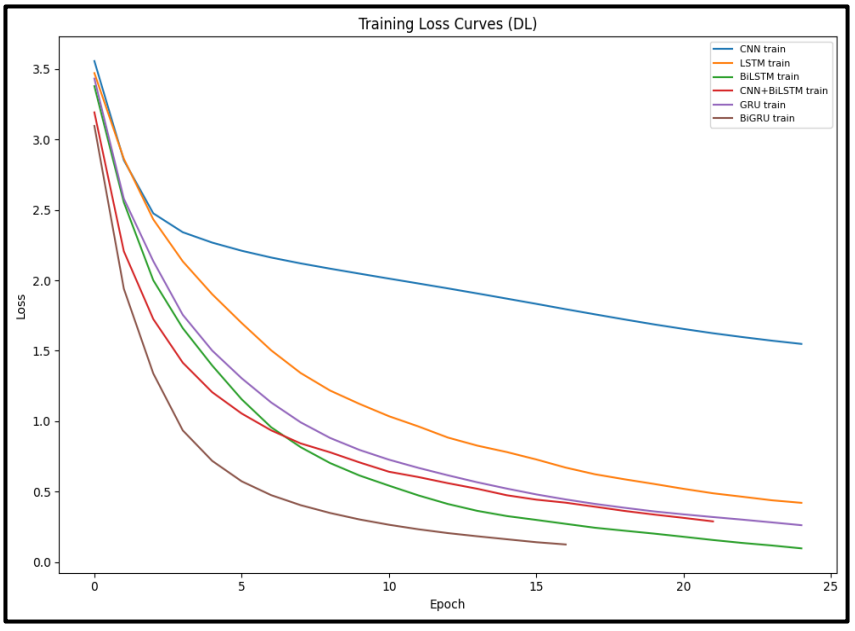

Figure 5. Training loss curves for DL models

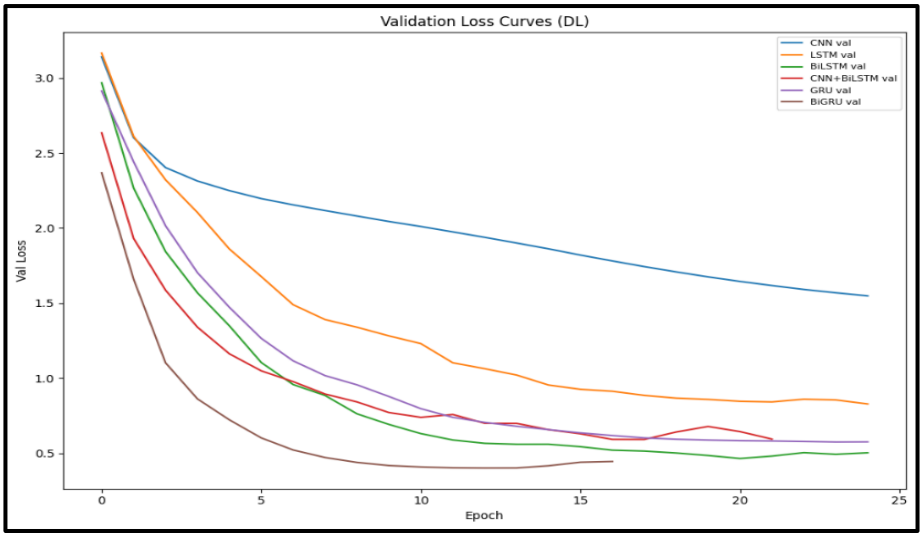

Figure 6. Validation loss curves for DL models

Keystroke dynamics learning is a complex pursuit from the operational evidence gained through the loss curves of all DL architecture models. The CNN model exhibits a gradual decrease in training and validation loss, which suggests that learning is relatively constrained when the keystroke timing features are represented as a feature-sequence. The LSTM model suffers from the same high cost of loss but converges around a higher than GRU-based architectures, showing them not only weakness in optimization but also in establishing proper dependencies within extracted features. The fast convergence and lower loss values of BiLSTM and BiGRU demonstrate stronger learning stability and the generalization capacity across identities. In terms of preventing overfitting, the validation loss curves indicate that BiGRU and BiLSTM have better generalization performance as they exhibit lower value of validation loss along with less pronounced overfitting compared to CNN-based models. Overall, the results that support this statement indicate that we will need gated recurrent architectures to better learn keystroke behavior.

3.6 Fraud Detection and Score Distribution Analysis

The study also reported a fraud detection score based on centroid-based detector with which it verified exam deployment in real world online environment. This procedure is executed as students drawing an axis of coordinate and the enrollment samples are designed a feature centroid, based on which new score samples are distance-to-centroid scored and transformed into a similarity score. The distributions of the centroid scores for both genuine and impostor cases are illustrated in Figure 7.

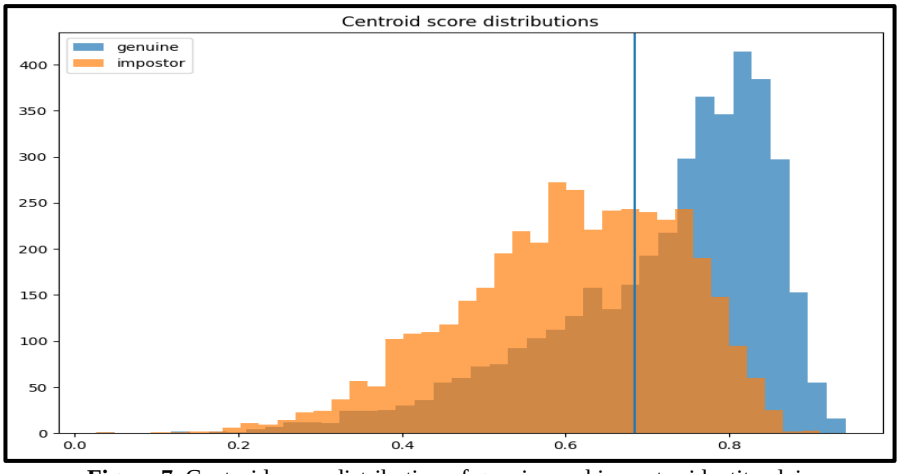

Figure 7. Centroid score distribution of genuine and impostor identity claims

The score distribution shows a clear separation between authentic and impostor identity claims. Real matches are compacted on higher scores, but a distributed tail compare the impostor contributions where will be lower matches to original identity profile. Then the line of threshold becomes a good operational decision boundary that can be used by real rejecting student and issuing flagging suspicious attempts. Such high score-based separation demonstrates that centroid verification successfully serves as an efficient continuous authentication layer augmenting the core identity classifier.

3.7 Identity Similarity and Risk of Confusion

The plot of the cosine similarities in Figure 8 makes a cognitive link between the group centroids of student identity; thereby, it creates a behavioural similarity study.

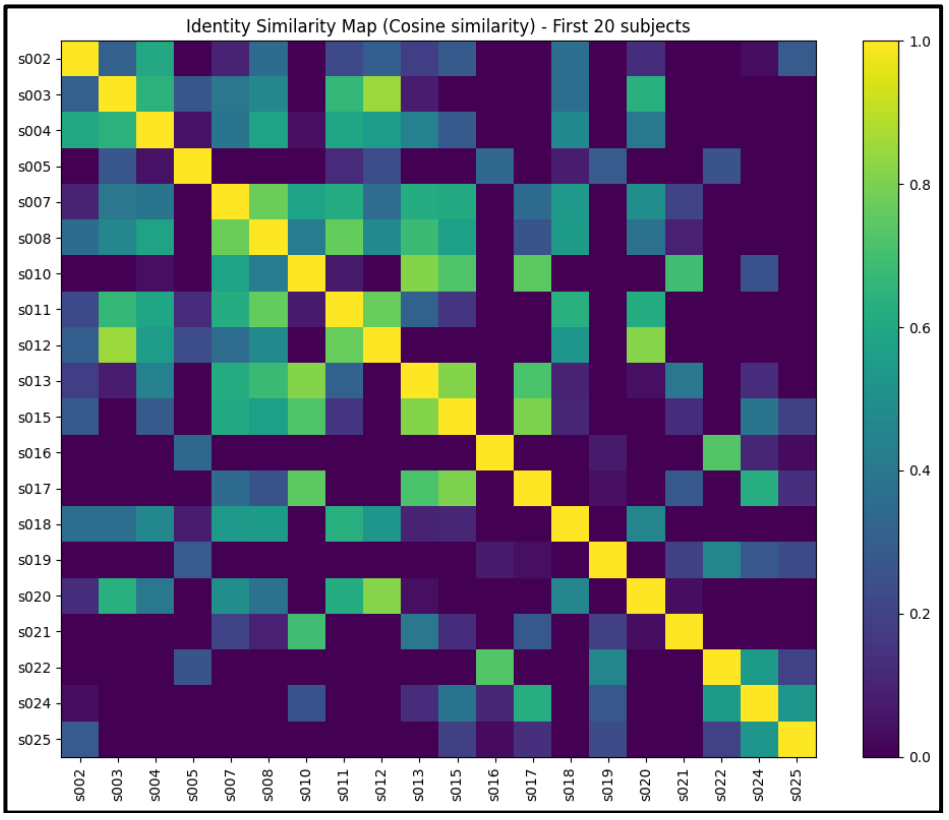

Figure 8. Identity similarity map (cosine similarity) for those 20 queries

Figure 8 demonstrates that students display unique typing patterns because their signature samples through the study produced forty-two signature patterns, which resulted in forty-two different identifiable signature signatures. The study found that only a few subject pairs within the study demonstrated more matching characteristics, which resulted in the increased chance of misidentification between subjects while making it harder to identify impersonators. The examination process gains operational benefits through similarity analysis because it enables the selection of difficult prompt phrases which require assessment during high-similarity pairs and through the requirement of identity verification which applies to users in high-risk similarity areas.

3.8 t-SNE Cluster Visualization of Typing Signatures

Figure 9 visualizes embedding clusters using t-SNE for the top 12 subjects.

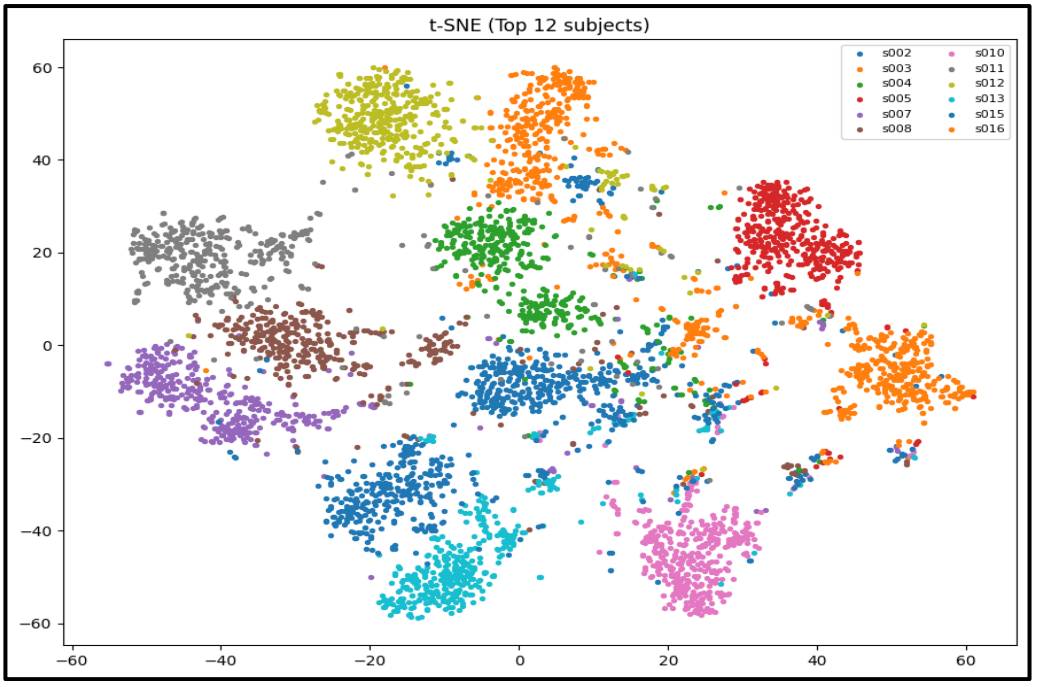

Figure 9. t-SNE cluster visualizations for keystroke samples of the top 12 subjects

Figure 9 shows distinct clustering patterns through keystroke dynamics embeddings, which produce multiple identities that create separate clusters showing distinct typing patterns. The study observes that some subjects share common behaviors which result in partial identity confusion between them. The study found a few points which had no relation to the main data because they resulted from either noisy typing execution or users showing unpredictable behavior or their conduct shifting between different sessions. The visualization confirms statistical results because it shows strong identity organization within the dataset, which includes multiple identities that stay close together in feature space, thus requiring verification thresholds and risk-based decision policies to handle identification instead of using strict class label predictions.

3.9 Typing Stability and Behavioral Drift

Figure 10 shows the reading selections with typing stability indices per subject, as measured by the average standard deviation across the features.

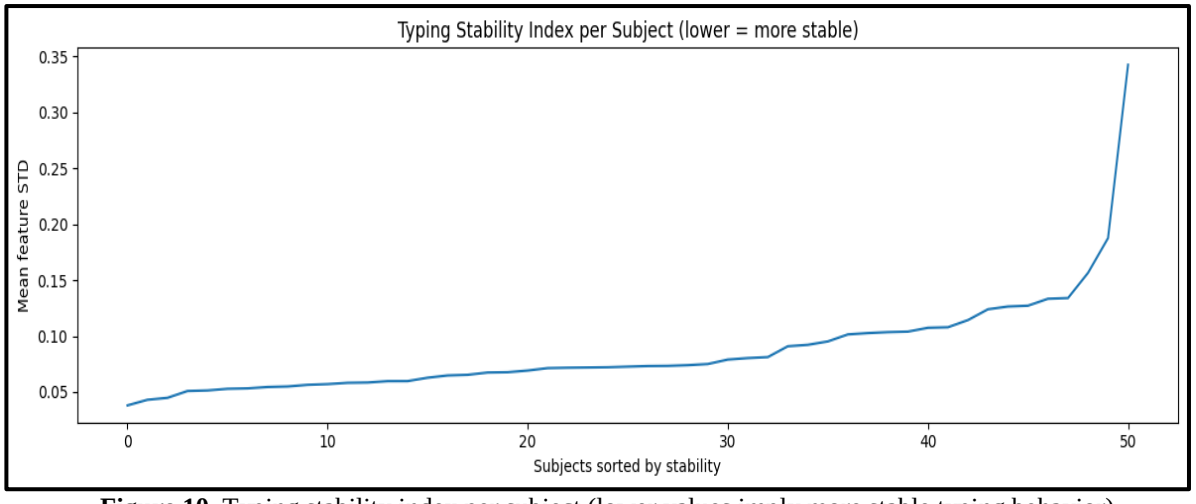

Figure 10. Typing stability index per subject (lower values imply more stable typing behavior)

Figure 10 demonstrates that different subjects show inconsistent patterns of stable behavior because of their different ways of maintaining stable conduct. The two subjects who demonstrated minimal changes in their keystroke patterns achieved better authentication results because they experienced fewer false rejections while maintaining dependable identity verification. The subjects demonstrate consistent typing patterns, yet their high variability results in increased risks of error detection through typing because of their probable mistakes. The discovery has significant operational value for online examination security systems because it demonstrates the need for adaptive authentication systems which depend on student behavior patterns to determine security thresholds and monitoring intervals during testing sessions.

3.10 Fraud Risk Map Using Entropy vs Score

Figure 11 provides the fraud risk map using probability score versus entropy (uncertainty).

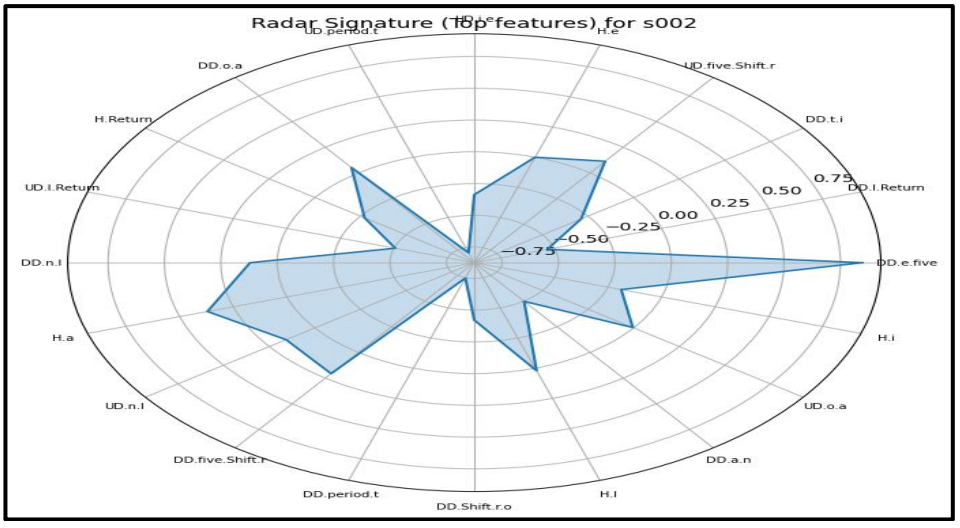

Figure 11. Fraud risk map (claimed identity probability score vs prediction entropy)

Figure 11 shows strong discrimination between true and impostor using the E (q | y, x) score along with prediction entropy. Genuine, on other hand, tend to cluster for higher score values without compromising entropy suggesting confident and consistent identity verification. Impostor attempts however, focus around zero scores with higher entropy which means the uncertainty, and behavioural based on claimed identity profile is high. The patterns show that fraud risk identification cannot only be based on the likelihood statistic, since the uncertainty provides additional security evidence. Precisely, low score but high entropy indicates strong evidence of impersonation / identity not matching and that should warrant the increased level of authentication or blocking the access on-the-fly based upon the exam security policy.

4. Conclusion

Universities use online testing methods as essential parts of their educational systems but these methods create security vulnerabilities which enable students to commit identity theft and impersonation offenses. The standard identity verification procedures that use usernames and passwords together with static login authentication methods fail to function in remote environments because users can share their credentials and hackers can steal them. The study developed a solution which used a keystroke dynamic system to verify identity through behavioral biometrics which enabled automated student authentication during online exams through normal keyboard usage without needing special equipment. The research results show that keystroke timing patterns create an effective biometric identification method which enables large-scale student identification. The system achieved high accuracy during classical ML tests which used the DSL-StrongPasswordData dataset containing 51 students. The Random Forest model produced the highest results among all classical models because it achieved an Accuracy of 94.07% and a Precision of 94.22% and a Recall of 94.07% and an F1-score of 94.03% which exceeded the performance of Gradient Boosting and SVM (RBF) and MLP. The outcomes show that ensemble learning models effectively capture non-linear relationships which exist between keystroke timing features. The problem required certain models which proved to be unsuitable. Decision Tree performance dropped significantly due to limited generalization, Naive Bayes was constrained by its independence assumptions, and Linear Regression failed

almost entirely because it is not designed for multi-class identity classification. DL models also achieved strong performance but did not surpass classical ML within this dataset configuration. Bi-directional gated recurrent architectures performed best among deep models, where BiGRU reached 88.14% accuracy, followed by BiLSTM (86.25%). The CNN-based models had weak performances, indicating that the engineered keystroke vectors do not completely conform to the inductive assumptions of convolutional architecture without sequence models like BiLSTM. The results were reinforced by visual analytics, in addition to the classification measures. We confirmed strong separability of these top-performing models over a range of decision thresholds by the ROC curve analysis, which is important for practical authenticating systems. According to the loss curves, we found that the convergence of recurrent models is better than CNN with lower validation loss and less over-fitting in gated recurrent units (GRUs), indicating that GRUs are able to capture patterns in typing behavior better. Identity similarity plots and t-SNE clustering visualization showed emulation signatures of the majority of students were very different, however few identity pairs are close to each other which correspond to higher risk for mistakes in confusion/impersonation. Moreover, the stability index of behavioural showed that some students have a high variability in their typing signature and this may lead to a low acceptance rate which indicates necessity of having personalised verification policies. Finally, the fraud risk map probability score across entropy produced clear operable insight. We see true attempts that are highly confident and show low uncertainty cluster together while impostor attempts generally fall under lower scores and generate higher entropies. It enables risk-based decision making which better reflects the huge uncertainty inherent in fraud detection than probability score alone. The work has repercussions for the security of online exam ecosystems. Keystroke dynamics is used as biometric to verify identity in LMS-based exam systems that provide a basic authentication stratum, enhancing security by minimizing the need for intrusive verification methods. The movement shows that there is no need for convoluted deep learning systems to achieve effective fraud detection, as traditional machine learning models, like Random Forest, outperform engineered keystroke features in this context. New research shows that authentication processes must be treated as perpetual systems. In this way, at risk verification will also improve exam sessions adaptive threshold systems and step up authentication do not give good turnover to the continent during high-risk identity pairs or for typers whose typing patterns did not remain stable. This means the system instead always executes an identity security process, whereas traditional cybersecurity standards are limited to single-session verification. The research yields impressive findings, but it has several significant limitations. Dataset uses hardcoded password phrase and controlled typing experiments whereas real lives exams on different devices in various typing situation and different psychological states of students which also interferes their typing. Keystroke biometrics may to be vulnerable to behavioral alterations which emerge over time due to fatigue or injury or hardware changes or device transitions between laptops and desktop computers. The study employed engineered timing features as sequence data to train deep learning models which resulted in restricted model performance because all available temporal data resources were not fully utilized. Fourth, although the study demonstrated strong separability, impersonation attacks in real settings may involve intentional mimicry, where an impostor attempts to emulate the student's typing rhythm. The current experiments developed their advanced adversarial scenarios without using direct simulation methods. The existing research needs future work which should be extended in multiple research areas. The first research direction requires researchers to obtain actual keystroke data from genuine online examination settings which should include various devices and keyboards and standard typing tests instead of using only fixed passwords. The development of improved fraud detection models will benefit from researchers investigating hybrid multi-modal frameworks which combine keystroke biometrics with mouse dynamics and device fingerprinting and browser signals and network-based behavioral patterns. Training Transformer-based temporal models and attention-based recurrent architectures on raw key-event sequences can push forward DL research as these models will capture timing dynamics better and i.e. require fewer engineered features. The second critical path involves deeper implementations of adaptive authentication systems that inform personalized threshold benchmarks contextualized along dimensions of (1) student stability metrics and (2) historical movement patterns as they predictably shift over time, thereby amplifying fairness while reducing the false rejection rate. Both adversarial testing (when an attacker invokes it using the backend test cases, etc.) and impersonation testing (Where mimicry is being used to get access through a verification process) needs to be complete before a comprehensive security assessment could be provided for defense mechanisms against replay attacks and collusion-based methods online. Also, the study proved that keystroke dynamics could be used for online examination fraud analysis as it is privacy-preserving as well as scalable. Specifically, you have shown that a classical ML method (Random Forest) provides identity recognition with high accuracy while verifying based on the risk of earned verification strength informative context dimensionality extension using uncertainty analysis, stability measures and similarity-aware controls is significantly improves verification confidence. The findings of the research show that keystroke-based continuous authentication is an effective security measure for the modern remote assessment mechanisms.

Appendix

Table A1. Abbreviations used in the manuscript

Abbreviation	Definition
BiGRU	Bidirectional Gated Recurrent Unit
BiLSTM	Bidirectional Long Short-Term Memory
CMU	Carnegie Mellon University
CNN	Convolutional Neural Network
CNN+BiLSTM	Convolutional Neural Network with Bidirectional Long Short-Term Memory
DL	Deep Learning
DNS	Domain Name System
DSL	Data Security Lab
DT	Decision Tree
EER	Equal Error Rate
FAR	False Acceptance Rate
FRR	False Rejection Rate
GB	Gradient Boosting
GRU	Gated Recurrent Unit
KNN	K-Nearest Neighbors
LMS	Learning Management System
LR	Logistic Regression
LSTM	Long Short-Term Memory
ML	Machine Learning
MLP	Multilayer Perceptron
NB	Naive Bayes
OvR	One-vs-Rest
RBF	Radial Basis Function
RF	Random Forest
ROC	Receiver Operating Characteristic
SVM	Support Vector Machine

References

[1] S. Al-Tamimi and Q. A. Al-Haija, “Security and privacy of educational computational intelligence,” in Internet of Behavior-Based Computational Intelligence for Smart Education Systems, IGI Global Scientific Publishing, 2025, pp. 301–328.
[2] S. A. Salloum, C. Mhamdi, B. Al Kurdi, and K. Shaalan, “Factors affecting the Adoption and Meaningful Use of Social Media: A Structural Equation Modeling Approach,” Int. J. Inf. Technol. Lang. Stud., vol. 2, no. 3, pp. 96–109, 2018.
[3] J. Heil and D. Ifenthaler, “Online Assessment in Higher Education: A Systematic Review.,” Online Learn., vol. 27, no. 1, pp. 187–218, 2023.

[4] S. Salloum, T. Gaber, S. Vadera, and K. Shaalan, "Phishing Email Detection Using Natural Language Processing Techniques: A Literature Survey," Procedia Comput. Sci., vol. 189, pp. 19–28, 2021.
[5] S. Salloum, T. Gaber, S. Vadera, and K. Sharan, "A Systematic Literature Review on Phishing Email Detection Using Natural Language Processing Techniques," IEEE Access, 2022.
[6] P. Dawson, Defending assessment security in a digital world: Preventing e-cheating and supporting academic integrity in higher education. Routledge, 2020.
[7] R. Al-Maroof et al., "Students' perception towards using electronic feedback after the pandemic: Post-acceptance study," Int. J. Data Netw. Sci., vol. 6, no. 4, pp. 1233–1248, 2022.
[8] G. Akçapınar, "Detecting AI-Assisted Cheating in Online Exams through Behavior Analytics," arXiv Prepr. arXiv2510.18881, 2025.
[9] K. S. Killourhy and R. A. Maxion, "Comparing anomaly-detection algorithms for keystroke dynamics," in 2009 IEEE/IFIP international conference on dependable systems & networks, 2009, pp. 125–134.
[10] S. AlTamimi, Q. A. Al-Haija, and A. AlShuaibi, "Gamification in Cybersecurity Education: Insights, Challenges, and Emerging Solutions," in 2025 IEEE 16th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2025, pp. 526–532.
[11] S. S. Abba, O. A. Obioha-Val, V. O. Ejiofor, O. M. Olaniyi, and N. R. Mayeke, "Behavioral Biometrics-Powered Continuous Authentication for Zero-trust Remote Work Environments: A Multi-factor Identity Verification Framework," Asian J. Res. Comput. Sci., vol. 18, no. 12, pp. 20–41, 2025.
[12] S. Salturk, T. E. Pamukcu, and N. Kahraman, "Contactless biometric verification from in-air signatures using deep siamese networks," Sci. Rep., vol. 16, no. 1, p. 130, 2026.
[13] R. Shadman, A. A. Wahab, M. Manno, M. Lukaszewski, D. Hou, and F. Hussain, "Keystroke dynamics: Concepts, techniques, and applications," ACM Comput. Surv., vol. 57, no. 11, pp. 1–35, 2025.
[14] A. Agrawal, "Human Behavior-Based Keystroke Password Authentication: A Behavioral Biometrics Approach," Explor. Intersect. Forensics Biometrics, pp. 61–82, 2026.
[15] R. Giot, M. El-Abed, and C. Rosenberger, "Keystroke dynamics authentication for collaborative systems," in 2009 International Symposium on Collaborative Technologies and Systems, 2009, pp. 172–179.
[16] P. S. Teh, A. B. J. Teoh, and S. Yue, "A survey of keystroke dynamics biometrics," Sci. World J., vol. 2013, no. 1, p. 408280, 2013.
[17] T. Sing, O. Sander, N. Beerenwinkel, and T. Lengauer, "ROCR: visualizing classifier performance in R," Bioinformatics, vol. 21, no. 20, pp. 3940–3941, 2005.
[18] R. Mattsson, "Keystroke dynamics for student authentication in online examinations." 2020.
[19] S. F. N. Sadikan, A. A. Ramli, and M. F. M. Fudzee, "A survey paper on keystroke dynamics authentication for current applications," in AIP conference proceedings, 2019, vol. 2173, no. 1, p. 20010.
[20] S. Al Tamimi, "Towards Beyond Technology: Reviewing Human Error (HE) as the Primary Reason of Cyber Security Breaches," in 2025 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA), 2025, pp. 1–6.
[21] X. Lu, S. Zhang, P. Hui, and P. Lio, "Continuous authentication by free-text keystroke based on CNN and RNN," Comput. Secur., vol. 96, p. 101861, 2020.
[22] E. A. Kochegurova and R. P. Zateev, "Hidden monitoring based on keystroke dynamics in online examination system," Program. Comput. Softw., vol. 48, no. 6, pp. 385–398, 2022.
[23] Y. Shi, X. Wang, K. Zheng, and S. Cao, "User authentication method based on keystroke dynamics and mouse dynamics using HDA," Multimed. Syst., vol. 29, no. 2, pp. 653–668, 2023.
[24] X. Wang, Y. Shi, K. Zheng, Y. Zhang, W. Hong, and S. Cao, "User authentication method based on keystroke dynamics and mouse dynamics with scene-irrelated features in hybrid scenes," Sensors, vol. 22, no. 17, p. 6627, 2022.
[25] Kevin, "Keystroke Dynamics - Benchmark Data Set," 2009. https://www.cs.cmu.edu/~keystroke/?utm_source=chatgpt.com.
[26] T. Dias, J. Vitorino, E. Maia, O. Sousa, and I. Praça, "KeyRecs: A keystroke dynamics and typing pattern recognition dataset," Data Br., vol. 50, p. 109509, 2023.
[27] N. González and E. Calot, "Dataset of human-written and synthesized samples of free-text keystroke dynamics to evaluate liveness detection methods," Mendeley Data, vol. 2, p. 2022, 2022.
[28] F. Hussain, "IoT Healthcare Security Dataset," Kaggle, 2023. https://www.kaggle.com/datasets/faisalmalik/iot-healthcare-security-dataset.
[29] A. Ibrahim, A. F. Kadhim, A. E. Hamzah, and M. A. Al-Shareeda, "A secure and scalable IoT home automation architecture with web and biometric control," International Journal of Cybersecurity Engineering and Innovation, vol. 2026, no. 1, 2026.

[30] M. Alshinwan, A. G. Memon, M. C. Ghanem, and M. Almaayah, "Unsupervised text feature selection approach based on improved Prairie dog algorithm for the text clustering," Jordanian Journal of Informatics and Computing, vol. 2025, no. 1, pp. 27–36, 2025. doi: 10.63180/jjic.thestap.2025.1.4.
[31] R. Almarshood and M. M. H. Rahman, "Enhancing Intrusion Detection Systems by Using Machine Learning in Smart Cities: Issues, Challenges and Future Research Direction," STAP Journal of Security Risk Management, vol. 2025, no. 1, pp. 3–21, 2025, doi: 10.63180/jsrm.thestap.2025.1.1.
[32] S. Alsahaim, M. A. Almaiah, and R. B. Sulaiman, "Security threats in mobile phones: Challenges, countermeasures, and the importance of user awareness," International Journal of Cybersecurity Engineering and Innovation, vol. 2023, no. 1, 2023.