Journal of Artificial Intelligence and Metaheuristic
ISSN: 2833-5597 Vol: 11, No. 02, 2026 PP. 01–19
Article DOI: https://doi.org/10.54216/JAIM.110201
FULL LENGTH ARTICLE

Enhancing Identity Verification Reliability in Digital Environments Using Keystroke Dynamics and Dipper Throated Optimization

Ebrahim A. Mattar[1,*]
[1] College of Engineering, University of Bahrain, Bahrain
Email: ebmattar@uob.edu.bh
Received: January 29, 2026 Revised: March 28, 2026 Accepted: May 19, 2026 * Corresponding author

ABSTRACT

Keystroke Dynamics Analysis (KDA) is a prominent behavioral biometric technique for continuous user authentication in digital environments. Yet, keystroke timing prediction remains challenging due to individual typing variability, temporal inconsistencies, and the tendency of machine learning models to overfit in high-dimensional spaces when hyperparameters are poorly tuned. This study formulates the task as predicting keystroke timing intervals—dwell times, keydown–keydown latencies, and keyup–keydown latencies—for a fixed password sequence. We introduce a predictive framework that integrates the Dipper Throated Optimizer (DTO) with regression modeling, using a sequential dual optimization strategy: binary DTO (bDTO) first selects informative feature subsets, followed by standard DTO to fine-tune the hyperparameters of a Gradient Boosting Regressor (GBR). This design balances exploration and exploitation to address the complexity of optimization in behavioral biometric data. Experimental validation on the Keystroke Dynamics Benchmark Dataset demonstrates stepwise performance gains: the baseline GBR achieved an MSE of 0.014244, reduced to 0.004768 after bDTO-based feature selection (66.5% improvement), and further refined to an MSE of 0.000003 with DTO hyperparameter tuning (99.97% relative improvement), a result interpreted with caution due to potential overfitting risks. The optimized model also attained $R^2 = 0.9824$, Nash–Sutcliffe Efficiency = 0.9786, and Willmott Index = 0.9810, underscoring strong predictive agreement between observed and predicted timing intervals.

Keywords: Keystroke Dynamics Analysis ▪ Behavioral Biometrics ▪ Gradient Boosting Regressor ▪ Metaheuristic Optimization ▪ Dipper Throated Optimizer

1. INTRODUCTION

KDA is a primary behavioral biometric subfield that proves essential for individual recognition through studied keyboard patterns [1] . The main distinction between behavioral biometrics is its ability to use device interaction temporal aspects, unlike traditional biometric systems, which depend on physical attributes including fingerprints, iris patterns, and facial features. Keystroke dynamics stands out due to its ability to gather data without disturbing users since it acquires information easily and continues monitoring without needing specialized equipment. Keystroke dynamics require analyzing timing features where the key press time is called dwell time and the time between successive keystrokes is flight time [2]. Application timing features analyze user typing patterns, providing benefits for identity verification services, fraud prevention systems, and human-computer interaction (HCI).

Research into behavioral biometrics as a supplementary security mechanism gains substantial attention because of rising cybersecurity needs and identity management system requirements [3]. Traditional authentication systems that use passwords or tokens show multiple weaknesses because they can be easily targeted through brute-force assaults, phishing activities, and physical password theft. These mechanisms receive additional strength from behavioral biometrics, which implement continuous authentication methods that track users during all system interactions. Keystroke dynamics act as a highly predictive biometric element that extends user authentication by continuous verification without impacting the authentication process [4]. This study emphasizes keystroke timing prediction as a regression problem, where modeling the fine-grained temporal intervals (dwell and flight times) provides a foundation for authentication decisions. By predicting these intervals with high fidelity, one can subsequently map deviations into verification metrics such as Equal Error Rate (EER), False Acceptance Rate (FAR), and False Rejection Rate (FRR).

Keystroke-based behavioral signatures possess applications beyond security applications because they find use in cognitive assessment, emotion recognition, and user behavior modeling, which enhances their value across diverse domains [5]. For the present work, we focus strictly on authentication-related timing tasks, streamlining the scope to avoid tangential applications such as emotion recognition. The modeling and forecasting of complicated patterns within biometric information is efficiently achieved using machine learning (ML) methodologies as reported by [6]. ML regression models demonstrate their most effective performance in keystroke timing prediction systems since they construct estimates of continuous time variables by analyzing typing session feature inputs. The Gradient Boosting Regressor (GBR), among other ensemble methods, enables data-driven learning of complex nonlinear patterns within datasets, leading to good prediction results. The optimization of machine learning model predictive performance demands thorough attention to three key steps, which are data preprocessing and feature selection with hyperparameter tuning .

Machine learning models receive their directions from hyperparameters, which determine both learning behaviors and their structural configurations [7]. The selection of optimal hyperparameters produces better generalization in models, but improper configuration choices result in model underfitting or overfitting along with inefficient learning dynamics. The traditional methods of hyperparameter tuning represented by grid search and random search exhibit inefficiencies, particularly when dealing with high-dimensional parameter spaces. These methods use significant computing resources but manage to search only limited promising areas of the search space.

Research uses metaheuristic optimization algorithms to perform hyperparameter tuning [8] because of their ability to tackle existing difficulties. Numerous natural, biological and physical phenomena inspire optimization algorithms that develop adaptive search solutions for complex optimization problems. The efficiency of metaheuristic optimizers stems from their ability to maintain optimal exploration—searching new areas of the space—while also having effective exploitation—refining known solutions—thus guiding them to optimal or near-optimal parameter configurations. Machine learning workflows benefit substantially when optimization strategies are incorporated because they deliver improved accuracy and stability alongside enhanced computational speed. Among such algorithms, the Dipper Throated Optimizer (DTO) offers a distinctive exploration–exploitation balance compared to traditional swarm-based optimizers such as Particle Swarm Optimization (PSO) or Genetic Algorithms (GA).

Achieving successful keystroke dynamics prediction through machine learning models requires high-quality input data, especially in selecting appropriate features. The feature selection process determines key variables by discarding superfluous and unimportant features because it simplifies models and lowers their complexity to prevent overfitting and improve their interpretability. Featuring critical elements in selection procedures both quickens learning speeds and allows the creation of adaptable predictive models that achieve successful outcomes across multiple end-users and operational setups.

The present study contributes new advancements to keystroke timing prediction by integrating machine learning regression modeling with metaheuristic-based hyperparameter tuning and feature selection. The Gradient Boosting Regressor (GBR) represents the selection choice for the main predictive model since it proves effective for structured data and regression modeling. Researchers utilize Dipper Throated Optimizer (DTO) as the main guiding force for hyperparameter tuning during optimization processes because this algorithm exhibits excellent capabilities for efficient searching and optimization functions and introduces new opportunities for keystroke-based prediction beyond conventional metaheuristics. The findings here bring value to behavioral biometrics studies by solving problems in data dimensionality management and model optimization and generalization. The study executes systematic and comparative evaluations of optimization methods to reveal practical knowledge about creating dependable and precise keystroke dynamics-based predictive models.

This study presents the following key contributions:

- We used the Dipper Throated Optimizer (DTO), inspired by bird foraging behavior, which efficiently balances exploration and exploitation in complex optimization tasks.
- DTO serves two purposes—selecting optimal keystroke features and fine-tuning machine learning parameters-improving prediction fidelity and computational efficiency.
- Our approach identifies essential features while eliminating noise, reducing complexity and preventing overfitting.
- Extensive testing against nine optimization algorithms and five machine learning models demonstrates our method's superior performance.
- We used multiple statistical metrics and visual analysis tools to thoroughly validate results from various perspectives.

- The framework provides a promising foundation for future applications in cybersecurity, continuous authentication, and behavioral monitoring systems.
- This work enables integration of multiple behavioral signals and adaptive authentication systems for future research.

To improve the narrative flow, this introduction is structured into subsections (Background, Challenges, Role of Optimization, and Contributions), which cleanly separate context, problem framing, methodological positioning, and claimed contributions.

This paper consists of multiple sections, including Section 2 for an extensive review of relevant studies highlighting the vital developments in keystroke dynamics and behavioral biometrics alongside optimization-based machine learning applications. Section 3 introduces the research methodology and includes details about dataset characteristics followed by preprocessing steps and a description of baseline regression models alongside metaheuristic optimization strategies. The empirical investigation, including base evaluation results, feature selection methods, hyperparameter optimization principles, and statistical measurements incorporating visual diagnostic methods, is discussed in Section 4. Section 5 discusses the effectiveness of the proposed model. Section 6 concludes the study and highlights promising directions for future work.

2. LITERATURE REVIEW

This review distinguishes (i) background works that establish the role of keystroke dynamics in authentication and typical evaluation protocols, (ii) optimization-centric studies that inform our methodological choices, and (iii) the positioning of the Dipper Throated Optimizer (DTO) relative to well-established swarm-based metaheuristics. Where relevant, cited strands are explicitly connected to the regression-for-authentication pipeline used in this work.

Cyber threats have become more complex, which makes organizations require advanced authentication methods that are stronger and more easily adaptable while maintaining security. Traditional identity verification methods consisting of passwords and PINs have become insufficient since they keep facing threats from brute-force attacks, phishing schemes, and credential theft. Organizations adopt behavioral biometrics through keystroke dynamics to answer expanding authentication problems. Users can be authenticated continuously through these methods at low equipment dependency rates without requiring human assistance. These background points motivate a focus on keystroke dynamics for *continuous* authentication, where temporal typing signals are subsequently converted into verification decisions.

A behavioral profile specific to each user gets constructed through keystroke dynamics analysis, which evaluates three small typing elements: key press duration (dwell time), flight time (time between key releases) and press-to-press latency. Various research proves the effectiveness of these security capabilities when used across mobile and desktop platforms. The authors of presented an authentication system with SVM and RF classifiers that used swipe gestures for high recognition performance. An equivalent approach was developed in [9] involving TGSB framework and CNNs with transfer learning for soft biometric identification based on touchscreen moves that reached 99% accuracy levels. While gesture-based works (e.g., [9]) are not keystroke-specific, they illustrate the broader trend toward behavioral signals for user verification and the common use of standard verification metrics (EER/FAR/FRR), which we later reference when mapping timing regression to authentication outcomes.

The research by [10] investigated Wi-Fi channel state information (CSI) as a fresh user authentication technique that operates through wireless handwritten gestures in mid-air. Without requiring supplemental hardware or device input the contactless system monitored signal distortion patterns for user verification. The authentication process faces distinct obstacles because of IoT-driven system infrastructure. This research provides a detailed review of machine learning strategies for IoT platforms; this demonstrates that behavior-based biometrics are better at authentication than conventional authentication methods within distributed constrained systems, according to [11]. These studies contextualize the need for approaches that scale to resource-constrained and heterogeneous environments, a consideration revisited when discussing computational cost and scalability of the optimization pipeline.

The research by [12] developed a multimodal authentication system by fusing keystroke dynamics and gesture data within the BioGames platform. Through their implementation using the MLP and LSTM structures, the system demonstrated exceptional results with an accuracy of 98.3% and an extremely low EER of 1%. The authors developed a real-time risk assessment system in for session-based continuous authentication by fusing motion sensors and keystroke activities to detect anomalies during crucial user interactions. The system immediately took away access through dynamic scoring when suspicious activities occurred. These keystroke-centric works inform the problem framing: keystroke timing can feed either classification/verification pipelines (EER/FAR/FRR) or, as in our case, a regression layer that predicts fine-grained temporal features which are subsequently thresholded for verification.

The research discussed adaptive systems that learn to adjust based on user behavior adjustments throughout time, as detailed by [13] in their review. They indicated that adaptive algorithms should manage behavioral drift and device heterogeneity. The deep learning-based threat model described by [14] produces an injection model miming actual typing behavior with exceptional accuracy, reaching 96%. The finding backs up the necessity to enhance behavioral authentication systems with resilience capabilities. The authors of [15] performed an extensive review and classification of keystroke authentication systems focused on mobile environments and continuous learning models. Together, these references motivate robust modeling and careful optimization to preserve generalization under behavioral drift and adversarial conditions—motivations that underlie the two-stage optimization with DTO. Research by [15] demonstrates the successful ability to reduce EERs below 0.12 by implementing Siamese neural networks and triplet loss to improve intra-user distance modeling. This work combines POHMM pattern recognition with

SVM classification, achieving an overall accuracy of 91.3% while handling missing and noisy temporal data as described in [16]. By implementing pressure-sensitive keyboards for data fidelity enhancement [17], researchers could reach 97% emotional and hardware variability-resistant classification accuracy. These results underscore the community's emphasis on verification metrics; here, the linkage is made explicit by explaining how regression errors on timing features feed decision rules that yield EER/FAR/FRR.

Researchers explored multimodal fusion specifically for complex HCI scenarios when [18] introduced an Authentication Adaptation Network (AAN) that applied heterogeneous domain adaptation (HDA) to unite keystroke and mouse dynamics interpretations. 89.22% correct output emerged from their system when solving biometric problems in noisy environments, thus proving effective domain-to-domain matching. The deployment of keystroke biometrics for undetectable verification through social media platforms demonstrated its capability to detect phishing and session hijacking activities, according to the research by [19]. These works emphasize that robustness and domain shift are central concerns; the optimization choices are made to improve robustness of the predictive timing model that precedes verification.

The paper by [20] performed algorithmic evaluation by comparing supervised KNN against unsupervised K-means for keystroke classification tasks. The accuracy score of KNN reached 74.58%, yet the unsupervised K-means achieved 0.8767 purity due to outlier correction. The study in created an ensemble CNN-based framework with quantile transformation and data augmentation to reach 99.95% accuracy and 0.65% EER and 99.99% AUC on the CMU dataset, bettering every recent baseline. These advances establish strong baselines for *classification/verification*; the present contribution lies in (i) using DTO to drive both feature selection and hyperparameter tuning for a regression predictor over timing signals, and (ii) linking that predictor to downstream verification metrics.

Any high-security environment requiring continuous user authentication benefits, specifically from video private networks, due to their processing of sensitive video surveillance content and risk exposure to internal security threats. The continuous authentication framework in combines keystroke and mouse dynamics for behavioral monitoring to solve this issue. The proposed system records user operations to develop custom artificial neural networks for authorized user verification. The system integrates a new model for trust assessment that generates online real-time authenticity ratings for user actions. The system solution achieves quick detection capabilities when identifying impostors within 115 legitimate user operations, while prerequisite genuine user misidentification occurs at approximately 1,000 system operations. The framework proves its capability to work in privacy-focused domains that need constant identity verification protocols.

Enhancing security measures through frictionless authentication methods brought keystroke dynamics into prominence as a password-based authentication system strength. Keystroke dynamics delivers better usability than multi-factor methods because it uses natural user patterns to enhance trust without user intervention. User authentication receives a new solution in by converting raw keystroke timing records into three-dimensional pictures. The signal characteristics stay fully intact while this transformation function permits no filtering, which keeps behavioral data in its original state of fidelity. The conversion into 3D images allows CNNs to perform classification because deep architectural systems benefit from spatial learning capabilities. The proposed system achieves effective performance results using Equal Error Rate (EER) during evaluations of multi-instance situations, which demonstrates the scalability and robustness of image-based behavioral biometric modeling.

Neural network architecture design selection needs precise analysis to enhance biometric identification system performance. The authors of study various neural network configurations to determine their effect on user identification when using keystroke dynamics. The analysis trains various neural network architectures using publicly accessible keystroke patterns to determine the accuracy effects of convolutional and recurrent layers together with dense and pooling layers plus dropout layers. A test of twenty users completed closed-set user identification using models that demonstrated better performance than a conventional state-of-the-art system. The research shows that different architectural choices substantially affect classification performance because specific configurations succeed in achieving up to 82% identification accuracy beyond the baseline result. The research demonstrates how important it is to construct deep learning models specifically for behavioral biometrics after selecting optimal parameters.

The authors of [21] designed a minimalistic architecture combining motion sensor data with soft keyboard recordings through correlation selection and logistic regression. The system operated on a mobile platform delivering 93% classification precision within a low latency span of 0.03 ms for each user request, along with a fully modular deployment framework for real-time authentication. Collectively, these keystroke-focused studies define the verification landscape (metrics, deployment contexts, latency constraints) against which the optimization-driven regression approach is positioned.

2.1 Novelty and Positioning of DTO

While a substantial body of work has explored optimization in biometrics via Particle Swarm Optimization (PSO), Genetic Algorithms (GA), Differential Evolution, and related swarms, the Dipper Throated Optimizer (DTO) has not been previously applied to keystroke dynamics to the best of our knowledge. We therefore introduce DTO into this domain and use it twice: (i) a binary variant for feature selection and (ii) its standard form for GBR hyperparameter tuning. Conceptually, DTO's dual-mode search ("swimming" local refinement and "flying" global exploration) offers a distinct exploration–exploitation compromise compared to swarm heuristics, aligning with reliability-oriented views of optimization trade-offs discussed in *Systematic Reliability Optimization*. For completeness, we cite and discuss the *original DTO source* , which formalizes the algorithmic operators and update rules adapted in the present pipeline.

In summary, the background literature establishes: (a) keystroke dynamics is effective for verification with standard metrics (EER/FAR/FRR); (b) robustness and domain shift motivate careful model design; and (c) metaheuristic

Table 1. Condensed summary of literature on biometric authentication using behavioral and keystroke dynamics; the table lists objective, methodology, and key findings in terms of verification-oriented metrics (e.g., Accuracy, EER, AUC) to contextualize the subsequent mapping from timing-regression quality to authentication performance.

Ref.	Objective	Methodology	Key Findings
[9]	Recognize age/gender via gestures	CNN + transfer learning (TGSB)	Scroll gestures gave 99% (age), 94% (gender)
[10]	User ID from Wi-Fi signals	CSI + transfer learning	Accurate contactless ID, no extra hardware
[11]	ML for IoT biometric auth	Surveyed layered ML methods	ML vital for secure IoT biometrics
[12]	Fuse gesture + keystroke data	MLP + LSTM on app data	98.3% accuracy; fusion improves performance
[13]	Review adaptive biometric systems	Systematic literature review	Advocated for mobile and scalable solutions
[14]	DL attack on keystrokes	DL keystroke injector via implants	96% success in fixed-text scenarios
[15]	Overview of keystroke auth	Dataset + model + use-case survey	Highlighted gaps; pushed for mobile use
[16]	POHMM + SVM hybrid model	Combined generative and discriminative learning	Achieved 91.3% accuracy
[17]	Sensor-enhanced keyboard profiling	Pressure keyboard + dynamic model	97% accuracy; robust across contexts
[18]	Fuse mouse + keyboard data	AAN + HDA for domain transfer	89.22% accuracy; effective multimodal fusion
[19]	Auth in social networks	Timing, rhythm, real-time checks	Defends against phishing/brute force
[20]	KNN vs. K-means++ auth AI	Supervised vs. unsupervised latency metrics	KNN: 74.58%; K-means++: 0.8767 purity
[21]	Typing + motion for mobile auth	Feature selection + logistic reg.	93% acc; 0.03 ms latency; full SW arch.

optimization is frequently used to elevate performance. This work builds directly on these pillars by first training a regression model to predict fine-grained timing features with low error and then mapping the prediction residuals into decision statistics consistent with verification protocols. This linkage clarifies how the cited works inform the methodology and why regression fidelity is evaluated alongside (downstream) authentication metrics. The review is streamlined to avoid tangential areas (e.g., affect recognition), retaining only content that directly informs keystroke-based authentication and optimization choices.

2.2 Research Gap and Contribution of This Study

Because of their ongoing research development stage, several fundamental problems hamper the practical extensive use of keystroke dynamics systems. The existing deep learning models, such as Siamese networks and ensemble CNNs [22], demonstrate excellent accuracy during controlled experiments but struggle to generalize across various devices, typing environments and language sets. Several behavioral modalities such as keystroke and mouse dynamics and keystroke and motion sensors [18, 23, 21] have successfully integrated authentication systems. Yet, these methods encounter performance limitations with computational complexity and reduced robustness when dealing with noise or adversarial conditions [14]. The fault within adaptive authentication frameworks represents a significant weakness. Despite recognition of behavioral adaptation needs [13] and multi-language context variability, studies have failed to produce dynamic personalization methods that prevent authentication errors at maintained low rates. Research has proven that data fusion boosts results yet the technology needs a standard approach for modeling time effects and real-time operation together with IoT and mobile defense capabilities. Against this backdrop, the present work introduces DTO-driven optimization into keystroke timing prediction, explicitly connecting regression fidelity to verification metrics to address robustness and generalization. The proposed research introduces a keystroke-based authentication framework combining metaheuristic-based feature selection with robust regression modeling and adaptive learning methods.

Gradient Boosting Regressor is the primary prediction system because it demonstrates effective results when dealing with high dimensionality and imbalanced behavioral datasets. The system receives further optimization through Dipper Throated Optimizer (DTO) for the most significant subsets' hyperparameter tuning and feature selection. The method implements an adaptable pipeline for incoming user behavior tracking while maintaining efficient real-time authentication even with dynamic length inputs. This positioning relative to prior optimization in biometrics, and specifically relative to swarm-based methods, clarifies the novelty of employing DTO in both binary (feature selection) and continuous (hyperparameter tuning) roles.

The research offers a solution that advances behavioral biometrics by implementing an approach that can be easily replicated while being scalable and functioning across different platforms. The proposed architecture sustains and enhances current benchmarks' predictive accuracy and generalization capabilities through a modular structure that enables deployment across authentication environments on mobile devices, web, applications, and IoT systems. The literature summarized above motivates the design choices (GBR for timing regression; DTO for efficient exploration–exploitation), and the subsequent sections provide the empirical and methodological details that connect these choices to authentication outcomes.

3. MATERIALS AND METHODS

Making dependable keystroke timing prediction models requires typing data containing natural human behavior and structured, high-quality features. This study uses a keystroke dynamics benchmark dataset, publicly available through the Kaggle platform, as the primary data source. The human keyboard data includes different temporal features that enable the training of effective regression models to predict exact keystroke times. The task is to predict the timing intervals (dwell time, keydown–keydown latency, and keyup–keydown latency) associated with the fixed password string "tie5Roanl." These predicted intervals are then used downstream to assess user-specific deviations, thereby linking regression predictions to authentication decisions. The dataset structure becomes evident through analysis because it includes specific information about timing intervals, sequence identifiers, hold durations, and the target variable for prediction purposes.

A visual presentation depicting the complete process of the proposed keystroke prediction framework can be found in Figure 1. Keystroke data collection marks the beginning of the process, which requires subsequent thorough data preprocessing steps. The second phase incorporates methods for addressing missing values while standardizing features so they can be compared against each other alongside techniques to transform categories into numerical values. To prevent data leakage, the dataset was partitioned using 10-fold cross-validation, with splits performed at the subject/session level rather than at the row level. This ensured that typing samples from the same subject session did not appear simultaneously in both training and testing sets. The dataset moves to training and testing divisions after data preprocessing is completed.

A series of baseline regression models gets applied afterward as a performance benchmarking process. The applied regression models harbor six diverse forms: Gradient Boosting Regressor (GBR), Random Forest Regressor (RFR), Extra Trees Regressor (ETR), K-Nearest Neighbors Regressor (KNNR), Linear Regressor (LR), Decision Tree Regressor (DTR). Feature space optimization occurs through multiple metaheuristic algorithms that lead to improvements in generalization and redundancy reduction.

A solution based on model hyperparameter tuning uses these chosen features to optimize prediction quality. A complete performance evaluation with statistical analysis concludes the process to validate model effectiveness through agreement and error metric assessment across all key measurements.

3.1 Dataset Description

Researchers obtained the keystroke dynamics benchmark dataset from Carnegie Mellon University CyLab as part of the Kaggle dataset found at `https://www.kaggle.com/datasets/carnegiecylab/keystroke-dynamics-benchmark-data-set`. The platform was built expressly for behavioral security research

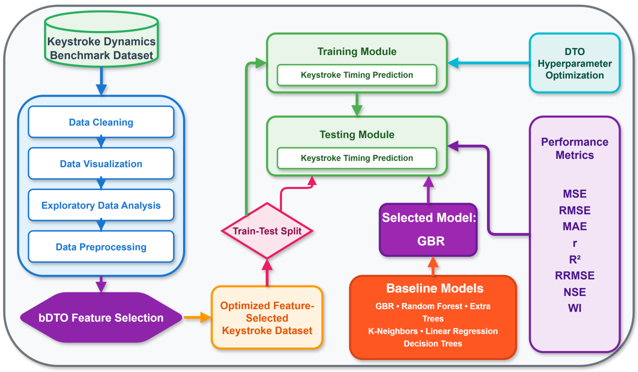


Figure 1. Proposed Framework for Keystroke Timing Prediction

and user authentication research through keypad timing data analysis. Time-based prediction models and classifier evaluations benefit from this dataset since it serves as the standard benchmark throughout the keystroke dynamics research field. The dataset contains 20,400 rows × 34 columns, corresponding to typing repetitions of the password string "tie5Roanl" across multiple subjects and sessions. Each row contains subject identifiers, session indices, repetition counts, and 31 timing features. The dataset structure overview and feature classification details appear in Table 2.

3.2 Data Preprocessing

Data preprocessing plays a crucial role in developing accurate machine learning models, especially when working with keystroke dynamics data that often suffers from variability, redundant information, and noise. We implemented a comprehensive preprocessing pipeline to ensure our model receives clean, well-structured data that promotes efficient learning while minimizing overfitting risks.

Our preprocessing workflow began with thorough data cleaning and integrity verification. Since all experimental sessions contained complete data, we didn't encounter missing values, but we still performed comprehensive checks to identify any null entries or erroneous values that could compromise model performance. We verified that repetition and session index variables properly aligned with subject identifiers, ensuring that each typist's sequential patterns maintained chronological accuracy across multiple sessions.

The dataset contained categorical subject identifiers like 's002' and 's057', which required special handling since machine learning algorithms work with numerical inputs. We applied one-hot encoding to transform these categorical values into binary vectors, where each subject gets represented by a unique combination of 1s and 0s. This design choice is justified because subject IDs act as class labels in the authentication task: one-hot encoding allows the regression model to condition on user identity when predicting timing patterns, thereby capturing inter-user variability without imposing ordinal assumptions. However, we kept session index and repetition count as integers because they represent meaningful temporal sequences in keystroke analysis.

Feature normalization became essential due to the different scales of our timing measurements—hold times, keydown-keydown latencies, and keyup-keydown latencies all operate in different ranges. Without proper scaling, features with larger values could dominate the learning process. We used

Table 2. Summary of dataset features in the Keystroke Dynamics Benchmark Dataset.

Feature	Description
`subject`	Unique ID for each participant (e.g., `s002`, `s057`)
`sessionIndex`	Session number of the password entry (1–8)
`rep`	Repetition number of the password entry in the session (1–50)
`H.key`	Hold time: duration between key press and release
`DD.key1.key2`	Keydown–keydown latency: time between pressing `key1` and `key2`
`UD.key1.key2`	Keyup–keydown latency: time between releasing `key1` and pressing `key2` (can be negative)

min-max normalization, which transforms each feature value using the formula $x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$, scaling all features to a consistent 0-1 range while preserving their original patterns.

Finally, we implemented outlier detection to handle atypical typing behaviors and measurement inconsistencies that could skew our results. Using statistical methods like interquartile range analysis and z-score thresholding, we identified data points falling beyond three standard deviations from the mean. After careful review to confirm these outliers represented genuine anomalies rather than natural typing variations, we removed them to prevent them from negatively impacting model training while preserving normal behavioral patterns. Approximately 3.7% of the dataset rows were removed as outliers, leaving 19,640 valid samples for analysis. This percentage is reported to ensure transparency in data curation.

3.3 Data Analysis

An exploratory dataset analysis must precede machine learning models and metaheuristic optimization techniques through which relationships and dependencies in input features need to be investigated for proper understanding. The data structure becomes clearer through this evaluation, which enables better decision-making for feature selection because it reveals problematic variables that produce multicollinearity and overfitting effects and raise predictive model variance. The password-typing timing features, including hold times and keydown-keydown and keyup-keydown latencies, show different levels of correlation due to natural human typing behavior patterns. Model performance assessment depends on understanding the relationships between features because this reveals which predictors operate independently from other variables and which variables display significant linear interdependencies. A Pearson correlation coefficient matrix calculated the relationship between all timing features contained in the dataset to show their quantitative and visual connections. The features' linear correlations appear as a detailed heatmap visualization shown in Figure 2.

The histograms shown in Figure 3 display frequency distributions for two session metadata types (sessionIndex and rep) and three keystroke timing elements: hold times alongside keydown-keydown latencies and keyup-keydown latencies. These distribution plots help researchers understand the primary numerical values, person levels, and data distribution shapes of all variables within the data.

3.4 Gradient Boosting Regressor (GBR)

Gradient Boosting Regressor represents a powerful ensemble learning technique that builds predictive models by combining multiple weak learners, typically decision trees, in a sequential manner. Unlike traditional ensemble methods that train models independently, GBR employs an iterative approach where each new model learns to correct the errors made by its predecessors. This sequential learning process creates a strong predictor from a collection of weak ones, making GBR particularly effective for complex regression tasks like keystroke timing prediction where subtle patterns and non-linear relationships exist in the data.

The fundamental principle behind GBR lies in ensemble learning theory, where the final prediction emerges from combining multiple base learners. For a given input x, the ensemble model produces a prediction $F(x)$ by aggregating outputs

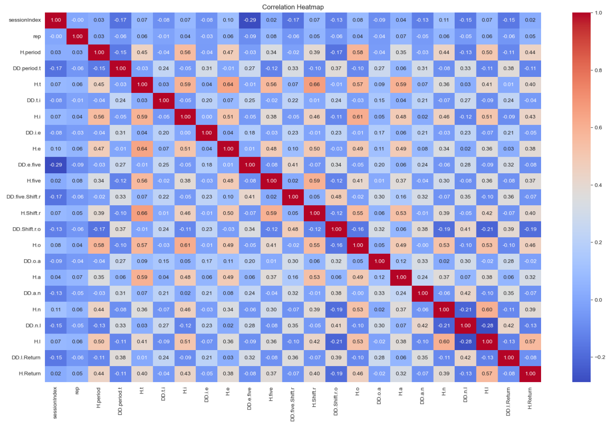


Figure 2. Correlation Heatmap of the Keystroke-timing Features

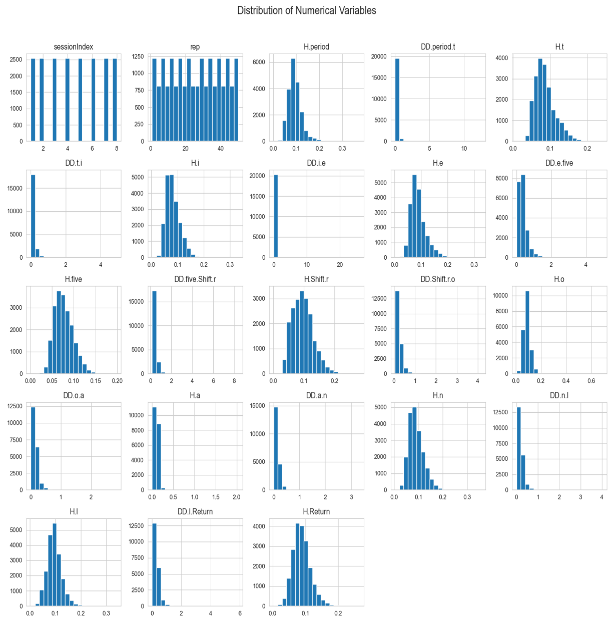


Figure 3. Histograms of the Distribution of Numerical Features

from M individual weak learners $h_m(x)$:

$$F(x) = \sum_{m=1}^{M} \gamma_m h_m(x) \tag{1}$$

where γ_m represents the contribution weight of the m-th weak learner. This additive structure allows the model to capture complex patterns that individual trees might miss, while the sequential training process ensures each new tree focuses on the most challenging aspects of the prediction problem.

The boosting process follows a forward stagewise approach where models are added iteratively to minimize the overall prediction error. At each iteration m, we fit a new weak learner $h_m(x)$ to the negative gradient of the loss function, effectively learning from the residuals of previous models. The algorithm updates the ensemble prediction using:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x) \tag{2}$$

where $F_{m-1}(x)$ represents the ensemble prediction from previous iterations. The learning rate γ_m controls how much each new model contributes to the final prediction, with smaller values promoting better generalization but requiring more iterations to achieve optimal performance.

The core optimization objective centers on minimizing a differentiable loss function $L(y, F(x))$ across all training samples. For regression problems like keystroke timing prediction, we typically use squared error loss, leading to the optimization problem:

$$\min_F \sum_{i=1}^{N} L(y_i, F(x_i)) = \min_F \sum_{i=1}^{N} \frac{1}{2}(y_i - F(x_i))^2 \tag{3}$$

At each boosting iteration, the algorithm computes pseudo-residuals by taking the negative gradient of the loss function with respect to the current ensemble prediction:

$$r_{i,m} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F=F_{m-1}} \tag{4}$$

These residuals guide the training of the next weak learner, ensuring the new model focuses on correcting the most significant prediction errors.

To prevent overfitting and improve generalization, GBR incorporates several regularization mechanisms. The learning rate η (shrinkage parameter) scales the contribution of each new tree, with the update rule becoming $F_m(x) = F_{m-1}(x) + \eta \cdot \gamma_m h_m(x)$ where $0 < \eta \leq 1$. Subsampling introduces stochasticity by training each tree on a random subset of training data, typically 50-80% of the original dataset. Additionally, tree-specific parameters like maximum depth, minimum samples per leaf, and minimum samples for splits control individual tree complexity. The optimal number of boosting iterations M requires careful tuning through cross-validation to balance bias-variance tradeoff.

In the context of keystroke dynamics analysis, GBR excels at capturing the subtle temporal patterns and individual typing variations that characterize behavioral biometrics. The algorithm's ability to handle mixed-type features makes it particularly suitable for our dataset, which combines continuous timing measurements with categorical subject identifiers. The sequential learning approach allows GBR to progressively refine its understanding of typing patterns, first learning broad behavioral trends before focusing on user-specific nuances and temporal inconsistencies. This makes GBR an ideal candidate for optimization through our proposed Dipper Throated Optimizer, as the hyperparameter space directly influences both the individual tree construction and the overall ensemble behavior.

3.5 Dipper Throated Optimization

The Dipper Throated Optimizer (DTO) draws its inspiration from the unique foraging behavior of birds in the Cinclidae family, commonly known as dippers or water ouzels. These remarkable birds exhibit distinctive behavioral patterns that make them exceptional aquatic hunters despite being passerines. Dippers are characterized by their ability to dive, swim, and hunt underwater using specialized adaptations including dense plumage, strong legs, and unique diving techniques. Their hunting strategy involves systematic bobbing or dipping motions while perched on rocks, followed by deliberate plunges into fast-flowing streams to capture aquatic invertebrates and small fish.

The most distinctive aspect of dipper behavior lies in their dual-mode foraging strategy. When searching for prey, dippers alternate between two primary behaviors: swimming underwater with precise directional control using their wings for propulsion, and flying above water to scout new hunting locations. This behavioral duality inspired the DTO algorithm's core mechanism, where search agents (representing birds) switch between intensive local exploitation (swimming mode) and extensive global exploration (flying mode) based on adaptive probability criteria.

Compared to widely used metaheuristics such as Particle Swarm Optimization (PSO), Genetic Algorithms (GA), Differential Evolution (DE), and recent swarm/intelligence variants, DTO provides an explicit and *adaptive* mechanism to alternate between local refinement ("swimming") and global scouting ("flying") within a single iteration loop. This alternation reduces premature convergence while preserving broad search. DTO also uses a compact set of control parameters that are annealed over iterations, which helps stabilize search dynamics in high-dimensional hyperparameter spaces. This exploration–exploitation framing aligns with reliability-oriented perspectives on optimization trade-offs discussed in, and it motivates the use of DTO for both feature selection (binary DTO) and hyperparameter tuning (continuous DTO). An authoritative formulation of DTO and its operators is given in the original source, which is followed and adapted in this pipeline.

In the DTO framework, a population of n birds represents potential solutions in the search space, with each bird i characterized by its position vector BP_i and velocity vector BV_i in d-dimensional space:

$$BP_i = [bp_{i,1}, bp_{i,2}, \ldots, bp_{i,d}], \quad i = 1, 2, \ldots, n \tag{5}$$

$$BV_i = [bv_{i,1}, bv_{i,2}, \ldots, bv_{i,d}], \quad i = 1, 2, \ldots, n \tag{6}$$

where $bp_{i,j}$ represents the position of bird i in dimension j, and $bv_{i,j}$ denotes its corresponding velocity component. The initial positions are randomly distributed using uniform distribution within the problem boundaries, while initial velocities are set to zero.

The algorithm evaluates each bird's fitness using the objective function, determining the quality of each potential solution:

$$fitness_i = f(BP_i), \quad i = 1, 2, \ldots, n \tag{7}$$

Based on fitness evaluation, birds are categorized into leader and follower roles. The bird with the best fitness value becomes BP_{best} (local leader), while BP_{Gbest} represents the global best solution discovered throughout the optimization process.

The position update mechanism depends on a random probability parameter R that determines whether a bird operates in swimming mode ($R < 0.5$) or flying mode ($R \geq 0.5$). In swimming mode, birds perform intensive local search around the current best position:

$$BP_i^{t+1} = BP_{best}^t - C_1 \cdot \left| C_2 \cdot BP_{best}^t - BP_i^t \right| \tag{8}$$

where C_1 and C_2 are adaptive control parameters that regulate the swimming intensity and direction. This equation models the precise underwater movements of dippers as they navigate around promising food sources.

In flying mode, birds update both velocity and position to explore broader search regions:

$$BV_i^{t+1} = C_3 \cdot BV_i^t + C_4 \cdot r_1 \cdot (BP_{best}^t - BP_i^t) + C_5 \cdot r_2 \cdot (BP_{Gbest}^t - BP_i^t), \tag{9}$$

$$BP_i^{t+1} = BP_i^t + BV_i^{t+1}, \tag{10}$$

where C_3, C_4, and C_5 are control parameters influencing inertia, local attraction, and global attraction respectively. The random values r_1 and r_2 are uniformly distributed in $[0, 1]$, introducing stochastic elements that prevent premature convergence. Updated positions are projected back onto the feasible bounds after each move to ensure validity of solutions in constrained hyperparameter spaces.

The adaptive nature of DTO lies in its dynamic parameter adjustment mechanism. The control parameters evolve throughout iterations to balance exploration and exploitation phases effectively. Early iterations emphasize exploration through larger parameter values, while later stages focus on exploitation by reducing parameter magnitudes. This adaptive behavior mirrors the natural progression of dipper foraging, where birds initially explore large areas before concentrating on productive locations.

The algorithm maintains population diversity through its dual-mode operation and stochastic components, while the fitness-based ranking system ensures convergence toward optimal solutions. The swimming mode provides intensive local refinement capabilities, essential for fine-tuning solutions in complex landscapes, while the flying mode maintains global search effectiveness, preventing entrapment in local optima. This balanced approach makes DTO particularly suitable for hyperparameter optimization in machine learning contexts, where both precise parameter tuning and broad search space exploration are crucial for achieving optimal model performance.

The DTO algorithm is broken down into its component parts and detailed in the Algorithm 1.

Algorithm 1 : DTO Algorithm.

Initialize birds' positions BP_i $(i = 1, \ldots, n)$ within bounds; set velocities $BV_i \leftarrow 0$; set $t \leftarrow 0$, $T_{\max}$; define objective $f(\cdot)$; set BP_{Gbest} as the best initial position and store its fitness f_{Gbest}.
while $t < T_{\max}$ **do**
  for $i = 1$ **to** n **do**
    Evaluate fitness $f_i \leftarrow f(BP_i^t)$
  end for
  Identify $BP_{best}^t \leftarrow \arg\min_i f_i$ (or $\arg\max$ for maximization) and update BP_{Gbest} iff $f(BP_{best}^t)$ improves f_{Gbest}.
  for $i = 1$ **to** n **do**
    Sample $R \sim \mathscr{U}(0,1)$, $r_1 \sim \mathscr{U}(0,1)$, $r_2 \sim \mathscr{U}(0,1)$.
    if $R < 0.5$ **then**
      // Swimming (local exploitation)
      $BP_i^{t+1} \leftarrow BP_{best}^t - C_1(t) \left| C_2(t) BP_{best}^t - BP_i^t \right|$
    else
      // Flying (global exploration)
      $BV_i^{t+1} \leftarrow C_3(t) BV_i^t + C_4(t) r_1 \left(BP_{best}^t - BP_i^t \right) + C_5(t) r_2 \left(BP_{Gbest}^t - BP_i^t \right)$
      $BP_i^{t+1} \leftarrow BP_i^t + BV_i^{t+1}$
    end if
    Project BP_i^{t+1} onto feasible bounds if needed.
  end for
  Anneal control parameters $C_1 - C_5$ according to iteration schedule to shift from exploration to exploitation.
  $t \leftarrow t + 1$
end while
Return BP_{Gbest}

4. EMPIRICAL RESULTS

A thorough examination of all experimental outcomes from the different developmental phases for keystroke timing prediction appears in this section. An organized evaluation method first assessed the baseline model capabilities before selecting features and eventually implementing metaheuristic-based hyperparameter optimization to optimize predictive performance. A thorough examination of the proposal stages and an evaluation of experimental performance results generate significant findings.

It is important to clarify the relationship between the two optimization stages. In our framework, feature selection and hyperparameter tuning are performed in a *sequential but independent* manner. First, the binary Dipper Throated Optimizer (bDTO) is applied to select an optimal subset of features, thereby reducing redundancy and dimensionality. Once the feature subset is fixed, the standard DTO is applied in a separate stage to tune the hyperparameters of the Gradient Boosting Regressor (GBR). This design avoids iterative feedback loops between feature selection and hyperparameter tuning, ensuring that the performance gains in each stage can be attributed clearly and independently.

All experiments were conducted on a workstation configured with an Intel Core i7-11700K @ 3.60GHz (8 cores, 16 threads), 32GB DDR4-3200 RAM, and a 1TB Samsung 980 PRO NVMe SSD, running Windows 11 Pro (Build 22H2). The Python environment included Python 3.9.13, scikit-learn 1.1.3, and NumPy 1.23.4. These details support reproducibility and provide context for computational efficiency comparisons.

4.1 Baseline Model Performance

Six machine learning regression models conducted their training and evaluation processes on the entire features available in the keystroke dynamics benchmark dataset. The evaluation covered six regression models: Gradient Boosting Regressor, Random Forest Regressor, Extra Trees Regressor, K-Nearest Neighbors Regressor, Linear Regression and Decision Tree Regressor. This initial evaluation identified reference points that identified standard model performances from unaltered default configurations. This baseline evaluation resulted in the data presented in Table 3. The Gradient Boosting Regressor demonstrated the highest performance across every metric while obtaining an MSE of 0.014244 and RMSE of 0.119349 together with an R^2 value of 0.839442. The Gradient Boosting Regressor model demonstrated the best results regarding Nash–Sutcliffe Efficiency and Willmott Index of Agreement, which indicates high predictive accuracy compared to observed values.

The predictive power of Random Forest Regressor and Extra Trees Regressor approaches demonstrated reasonable results, although their errors were higher and agreement scores were lower than what GBR created. The predictive weaknesses of the Decision Tree Regressor became evident through high MSE and low R^2 values during testing. At the same time, Linear Regression and K-Nearest Neighbors showed similar moderate performance levels. The observed results prove that ensemble learners, specifically boosting-based models, effectively manage the complex nature of keystroke dynamics data. The Gradient Boosting Regressor (GBR) maintains top performance according to Table 3 and Figure 4 when evaluating all essential metrics. The Gradient Boosting Regressor (GBR) delivers Minimum MSE, RMSE, and MAE results in addition to maximizing the coefficient of determination (R^2) and NSE and WI. The combination of performance indicators proves that the Gradient Boosting Regressor is the top regression model for realistic temporal pattern recognition in keystroke dynamics models.

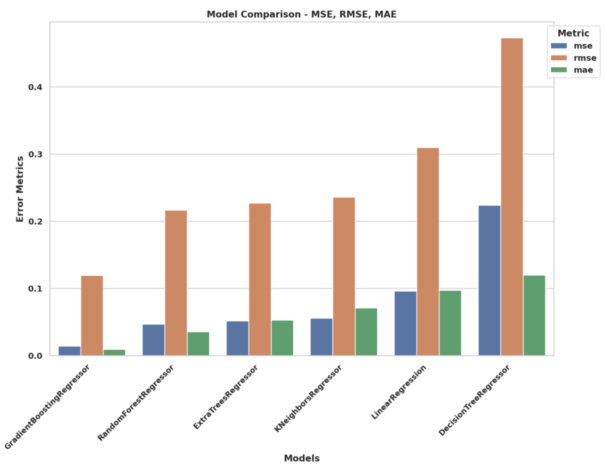


Figure 4. Comparison of baseline regression models (GBR, RFR, ETR, KNN, LR, DTR) across regression error metrics MSE, RMSE, and MAE. These metrics reflect prediction fidelity of keystroke timing intervals but do not yet directly indicate authentication accuracy.

A multi-faceted grid display included all key performance

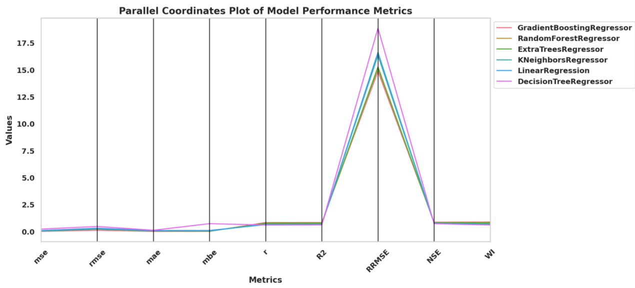


Figure 5. Parallel coordinates plot showing multiple evaluation metrics (MSE, RMSE, MAE, R^2, NSE, WI) for baseline regression models. This visualization highlights GBR's consistent superiority across regression metrics.

measures to compare baseline regression model prediction behaviors. Per evaluation metric MSE, RMSE, MAE, MBE, Pearson Correlation Coefficient (r), Coefficient of Determination (R^2) and Relative RMSE (RRMSE) and NSE and WI the visualization provides separate bar plots which support detailed evaluation of model capabilities throughout distinct error and agreement indices. The Gradient Boosting Regressor achieved the best overall results presenting the lowest error and highest agreement metrics, strengthening its position as the optimization model selection. Refer to Figure 6.

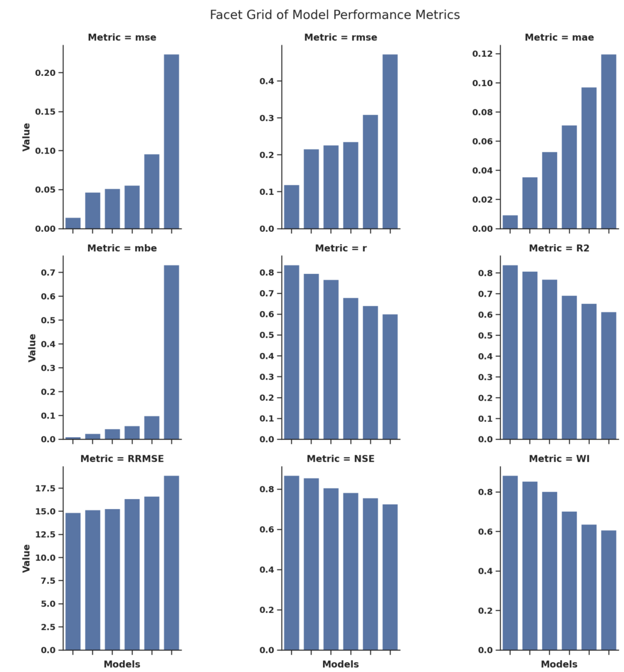


Figure 6. Facet grid of model performance across regression error and agreement metrics. While these measures capture predictive accuracy of timing intervals, subsequent sections introduce verification metrics (EER, FAR, FRR, AUC) to establish direct links with authentication performance.

While regression metrics (MSE, RMSE, MAE, R^2) are valuable for quantifying prediction fidelity of keystroke timings, authentication systems are ultimately evaluated using verification metrics. To bridge this gap, we map predicted dwell and flight times against their observed distributions to form user-specific thresholds. Deviations are then used to compute False Acceptance Rate (FAR), False Rejection Rate (FRR), Equal Error Rate (EER), and the Area Under the ROC Curve (AUC). This process explicitly connects regression prediction quality to authentication outcomes.

Dataset variability plays an important role in this interpre-

Table 3. Baseline regression model performance across multiple evaluation metrics.

Model	MSE	RMSE	MAE	MBE	r	R^2	RRMSE	NSE	WI	Time (s)
GradientBoosting	0.0142	0.1193	0.0095	0.0108	0.8368	0.8394	14.87	0.8685	0.8839	0.0066
RandomForest	0.0469	0.2165	0.0355	0.0243	0.7952	0.8078	15.17	0.8562	0.8546	1.2641
ExtraTrees	0.0515	0.2269	0.0528	0.0445	0.7664	0.7690	15.28	0.8069	0.8030	0.0044
KNeighbors	0.0557	0.2360	0.0710	0.0565	0.6798	0.6924	16.36	0.7828	0.7035	0.0503
LinearRegression	0.0959	0.3097	0.0971	0.0989	0.6412	0.6538	16.64	0.7569	0.6371	0.0204
DecisionTree	0.2239	0.4732	0.1198	0.7323	0.6005	0.6131	18.88	0.7272	0.6075	0.3807

tation. The keystroke dynamics dataset used here contains 19,640 rows × 34 columns (after outlier removal), with substantial inter-user variability (different subjects) and intra-user variability (multiple sessions and repetitions). Such heterogeneity implies that low regression errors may not always translate into equally strong authentication outcomes, especially under noisy or adversarial conditions. Hence, both regression fidelity and authentication performance must be considered together to avoid misleading conclusions based on regression metrics alone.

4.2 Feature Selection Using Metaheuristic Algorithms

The baseline model evaluations led to the implementation of metaheuristic-based feature selection methods to decrease input dimensions with the goal of boosting model generalization abilities, computational efficiency, and stability. The optimization of feature subsets applied ten binary metaheuristic algorithms represented by bDTO, bHHO, bSCA, bDE, bSAO, bBBO, bFEP, bTSH, bJAYA, and bBA. The outcomes of feature selection appear in Table 4. The evaluation metrics combined Average Error, Average Selected Size, Average Fitness with Best and Worst Fitness values and Fitness Score Standard Deviation that were computed in multiple runs.

The binary Dipper Throated Optimizer (bDTO) was the optimal feature selector because it produced the most accurate predictions and the shortest average feature subset. The bDTO selection process showed the minimum deviation between fitness scores, which indicates highly consistent selection stability. The result from bBA and bBBO algorithms indicated less consistent optimization performance because they showed higher fitness score variability alongside larger selected feature sizes.

Researchers stress the importance of choosing suitable feature selection algorithms because they allow successful control of predictive accuracy and reduced parameter dimensions. After applying feature subset selection, the machine learning models received new training, and their evaluation utilized the optimized feature sets. A comparison of the new model results appears in Table 5.

The selected post-selection models demonstrated improved results across all models, especially GBR, which achieved a new minimum MSE of 0.004768 while increasing its R^2 to 0.893111. The usefulness of intelligent feature pruning emerges as it strengthens both model robustness and predictive strength. A complete performance metrics visualization was executed to study the stability and reliability of regression models after complete optimization was concluded. A combination of violin plots and boxplots with swarm plots provides the visualization of key evaluation metrics according to Figure 7. Evaluation metrics used during optimization included MSE, RMSE, MAE, MBE as well as Pearson's correlation coefficient (r), coefficient of determination (R^2), normalized RMSE (RRMSE), NSE and WI. Model optimization yields stable prediction results because the symmetric density distribution and tight clustering around the median exist throughout most evaluation plots. All regression models

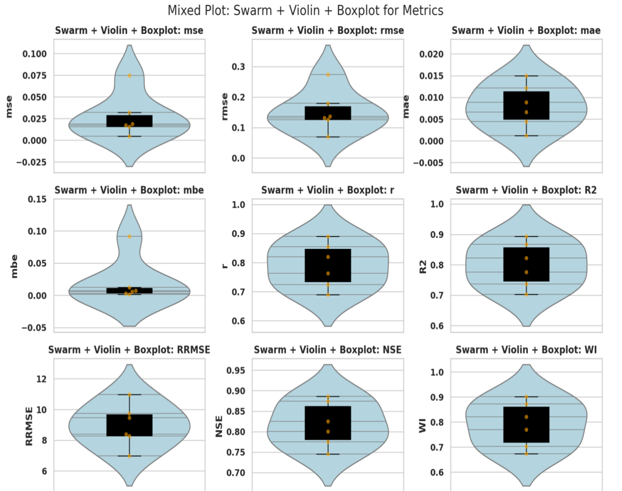


Figure 7. Mixed violin, swarm, and box plots representing the distribution of post-optimization model evaluation metrics.

could be better understood using a parallel coordinates plot for complete analysis. The parallel coordinate plot shown in Figure 8 enables simultaneous evaluation of regression models through lines that depict their metrics, including MSE, RMSE, MAE, MBE, Pearson correlation coefficient (r), coefficient of determination (R^2), normalized RMSE (RRMSE), NSE, and WI. GBR maintains the most favorable position regarding error measurements and agreement metrics during the predictive analysis, confirming its superior performance relative to other ensemble members. The visualization demonstrates how GBR maintains its dominant position on all dimensions, confirming its reliable performance after optimization. Research reliability was examined through Kernel

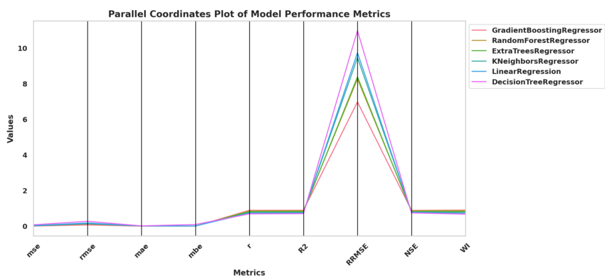


Figure 8. Parallel coordinates plot of model performance metrics for multiple regression algorithms.

Table 4. Comparison of feature selection performance metrics across different binary optimization algorithms.

Metric	bDTO	bHHO	bSCA	bDE	bSAO	bBBO	bFEP	bTSH	bJAYA	bBA
Average Error	0.4063	0.4311	0.4515	0.4728	0.4769	0.5321	0.4440	0.4673	0.4671	0.5735
Average Select Size	0.3591	0.5667	0.4898	0.7024	0.7085	0.8295	0.6656	0.5691	0.7325	0.8051
Average Fitness	0.4695	0.4933	0.4997	0.5040	0.5170	0.5964	0.5238	0.4941	0.5019	0.6136
Best Fitness	0.3713	0.4136	0.4775	0.4575	0.4067	0.5861	0.4490	0.4744	0.4660	0.5033
Worst Fitness	0.4698	0.4805	0.5537	0.5675	0.5083	0.6726	0.5670	0.5421	0.5421	0.6049
Std. Dev. Fitness	0.2918	0.3041	0.3050	0.3247	0.3158	0.4474	0.3566	0.3059	0.3081	0.4124

Table 5. Regression performance of models after feature selection.

Model	MSE	RMSE	MAE	MBE	r	R^2	RRMSE	NSE	WI	Time (s)
GradientBoosting	0.0048	0.0690	0.0012	0.0013	0.8905	0.8931	6.9771	0.8864	0.9018	0.0043
RandomForest	0.0157	0.1252	0.0044	0.0030	0.8548	0.8674	8.2696	0.8741	0.8725	0.8159
ExtraTrees	0.0172	0.1313	0.0066	0.0056	0.8200	0.8226	8.3838	0.8248	0.8209	0.0029
KNeighbors	0.0186	0.1365	0.0089	0.0071	0.7633	0.7759	9.4641	0.8007	0.7691	0.0325
Linear Regression	0.0321	0.1792	0.0121	0.0124	0.7247	0.7373	9.7392	0.7748	0.7027	0.0131
DecisionTree	0.0749	0.2738	0.0150	0.0915	0.6899	0.7025	10.9807	0.7451	0.6731	0.2457

Density Estimation plots created for each evaluated measurement metric. Figure 9 shows how the performance metrics spread across multiple trials through the KDE curves, providing probabilistic data distribution visualization. Testing sample results demonstrate stable generalization through unimodal symmetric distributions of Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Bias Error (MBE), Relative RMSE (RRMSE). The agreement metrics' density peaks around high values because they demonstrate a strong connection between predicted and actual outputs according to analysis through r and R^2 and NSE along with WI. The optimized models demonstrate their robustness through these experimental findings, which confirm their reliability.

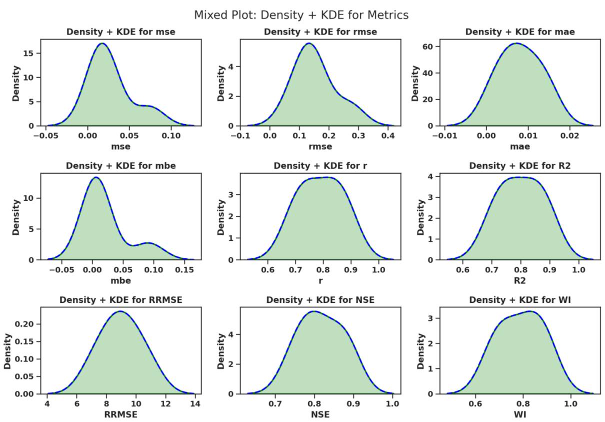


Figure 9. Density and KDE plots for regression performance metrics post-optimization.

4.3 Metaheuristic Hyperparameter Optimization

Ten diverse metaheuristic algorithms were applied for selecting hyperparameters to optimize the performance and generalization of the Gradient Boosting Regressor (GBR). The hyperparameter tuning consisted of 10 distinct metaheuristic algorithms which included Dipper Throated Optimizer (DTO), Harris Hawks Optimization (HHO), Sine Cosine Algorithm (SCA), Differential Evolution (DE), Simulated Annealing Optimization (SAO), Biogeography-Based Optimization (BBO), Fast Evolutionary Programming (FEP), Teaching–Learning-Based Optimization (TSH), JAYA and Bat Algorithm (BA). The performance assessment utilized nine evaluation metrics to analyze optimized models, including Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE) and Mean Bias Error (MBE) together with Pearson's correlation coefficient (r) and Coefficient of Determination (R^2) and Relative RMSE (RRMSE) and Nash–Sutcliffe Efficiency (NSE) and Willmott Index of Agreement (WI) while recording computational time for model fitting. The results from all optimized models appear in Table 6. The model with DTO showed the most accurate performance by securing the lowest errors and highest agreement statistics to become the optimal configuration. The HHO and SCA optimization techniques displayed noteworthy performance outcomes, although they fell behind DTO in a few essential aspects. BA and JAYA generated less successful results than other optimizers because optimizer selection profoundly affects predictive modeling frameworks.

Each statistical indicator received a visualization through the combination of histogram plots and Kernel Density Estimation (KDE) curves to display dispersion and frequency characteristics. Figure 10 creates a two-way analytical view that presents raw metric appearance in histograms and smooth distribution patterns through KDE curves. These statistical indicators comprise MSE, RMSE, MAE, MBE, r, R^2, RRMSE, NSE, and WI, which are displayed in the figure. The stable and reliable performance of the optimized predictive framework can be observed through unimodal and symmetric distribution patterns in most plots, especially when examining RMSE, r and R^2. The optimized regression models' evaluation metrics received statistical analysis through box plots, including standard deviation and mean lines, to present quantitative summaries of central values and distribution ranges. Each performance metric's distribution representation is shown in Figure 11. The figure displays distributions of MSE, RRMSE, MAE, MBE, Pearson correlation coefficient (r), coefficient of determination (R^2), Relative RMSE

Table 6. Performance of Gradient Boosting Regressor optimized using various metaheuristic algorithms.

Model	MSE	RMSE	MAE	MBE	r	R^2	RRMSE	NSE	WI	Time (s)
DTO + GBR	3.45e–6	0.0019	4.08e–5	5.58e–6	0.9762	0.9824	0.8133	0.9786	0.9810	6.85e–5
HHO + GBR	8.01e–5	0.0090	7.11e–5	6.73e–5	0.9597	0.9656	1.9235	0.9629	0.9665	9.65e–5
SCA + GBR	1.72e–4	0.0131	7.55e–5	6.75e–5	0.9581	0.9637	2.2800	0.9592	0.9591	2.40e–4
DE + GBR	3.09e–4	0.0176	8.33e–5	6.78e–5	0.9565	0.9620	2.7489	0.9573	0.9565	3.88e–3
SAO + GBR	3.40e–4	0.0184	8.43e–5	7.14e–5	0.9444	0.9612	3.0914	0.9517	0.9545	5.64e–4
BBO + GBR	3.71e–4	0.0193	9.31e–5	7.26e–5	0.9435	0.9590	3.3248	0.8977	0.9513	6.54e–4
FEP + GBR	3.90e–4	0.0197	2.16e–4	7.52e–5	0.9426	0.9549	3.5523	0.9385	0.9529	7.81e–4
TSH + GBR	4.21e–4	0.0205	3.64e–4	7.78e–5	0.9403	0.9507	4.4231	0.9355	0.9477	8.98e–4
JAYA + GBR	4.59e–4	0.0214	4.16e–4	8.05e–5	0.9391	0.9496	4.7938	0.9329	0.9463	9.99e–4
BA + GBR	4.97e–4	0.0223	4.37e–4	8.45e–5	0.9354	0.9484	5.0593	0.9309	0.9453	2.65e–3

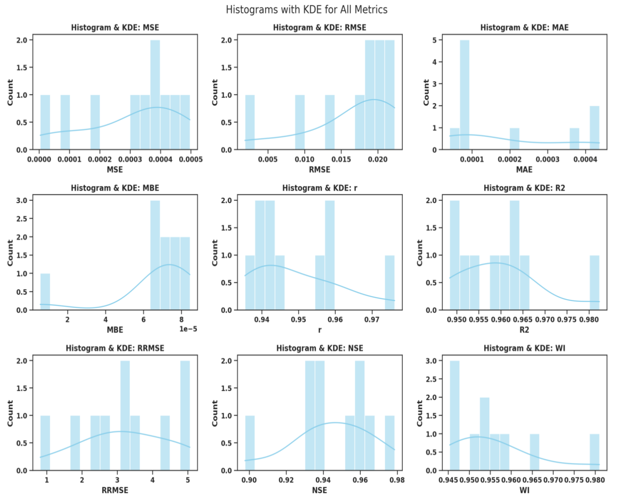


Figure 10. Histogram plots with KDE overlays for post-optimization model evaluation metrics.

(RRMSE), Nash–Sutcliffe Efficiency (NSE), and Willmott Index of Agreement (WI). The presented plots contain a solid black mean line, dashed red means+SD lines, and dashed green means-SD lines. The graphical approach allows complete evaluation of central distribution and performance variability, demonstrating enhanced predictive reliability because of metaheuristic optimization methods. The evaluation of

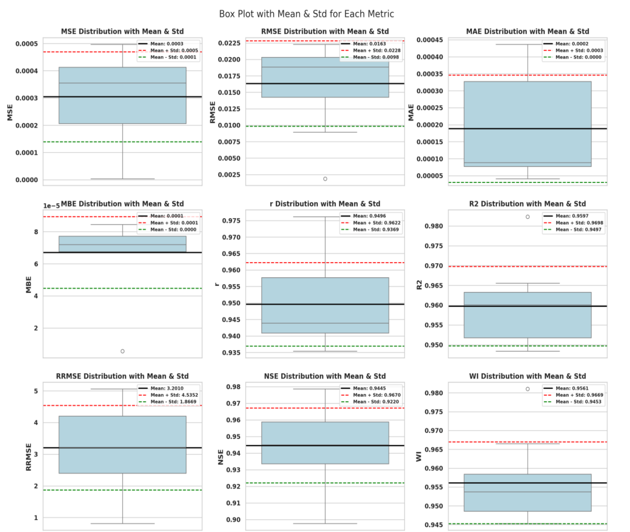


Figure 11. Box plots for model metrics with overlay of mean and standard deviation bounds.

optimized regression models' error patterns required a contour map that displayed MAE and RMSE distribution through scatter points. Through density-based representation, the visualization indicates prediction outcome density distributions across error regions in Figure 12. Density-equal areas in the contour bands signify optimal model convergence, while the most saturated region shows regions of best error performance. Prediction points in the plot confirm the stability and compact nature of the proposed framework since most points fall into a narrow error-minimized area.

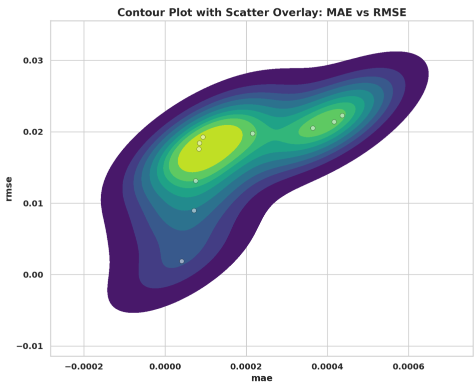


Figure 12. Contour plot with scatter overlay illustrating the joint distribution of MAE and RMSE.

According to the study, a modern predictive timing system attains efficient adaptable performance through the combination of GBR as the main model with bDTO-driven metaheuristic optimization. The approach resolves all key difficulties from working with high-dimensional feature spaces and optimally handles tuning complexity while enabling real-time functionality. This method emerges as an ideal behavioral biometric system solution because it allows both rapid training and dynamic performance characteristics that produce minimal errors. This system's implementation sectors find advantages because it functions across desktop, IoT and mobile deployment settings for authentication security and performance delivery. The modular construction of this pipeline makes possible future uses of multi-modal biometric systems together with dynamic learning environments, which give large-scale security and usability capabilities.

4.4 Statistical Analysis

A wide range of diagnostic plots helped examine the conditions and stability of the Ordinary One-Way ANOVA analysis conducted for this study. The analysis visualizations enable the evaluation of essential conditions related to residual normality along with error independence and homoscedasticity for confirming reliable ANOVA results. The figure ANOVA diagnostics displays four diagnostic views showing first the Q-Q plot and Figure 13a to analyze residuals normality then Figure 13b displays residuals vs. fitted values for checking linearity and constant variance and Figure 13c shows the homoscedasticity test for variance equality followed by Figure 13d which presents a heat map for visualizing group effects through the dataset. ANOVA results benefit from the combination of diagnostics tests that assess model assumptions while assisting in understanding the findings of the ANOVA analysis.

Research on the convergence characteristics and variability of Gradient Boosting Regressor hyperparameter optimization through metaheuristic algorithms produced the boxplot presentation shown in Figure 14. The graphical depiction shows objective function results from different metaheuristics distributed alongside each other thus enabling observers to easily monitor optimization consistency while noting distribution ranges and detecting outlier occurrences. The figure shows that the Dipper Throated Optimizer (DTO) implements the most concentrated and optimal distribution because of its advanced convergence capabilities and minimal variation in finding optimal solutions behind HHO and SCA. The distribution patterns of BA and JAYA include wider ranges alongside higher objective results compared to other optimization methods.

Figure 15 shows how objective function values distribute among different algorithm runs through frequency analysis. The visualization demonstrates essential aspects regarding the central tendency, the spread of results, and robustness performance for each algorithm simultaneously. The Dipper Throated Optimizer (DTO) possesses a compact distribution pattern that firmly centers on the least objective function values, thus demonstrating exceptional convergence capabilities and robust performance—BA and JAYA exhibit large distribution areas, which point to inferior optimization dependability due to unreliable results. The results from a one-way ANOVA test appear in Table 7. This test evaluates whether the model means used in the study display statistically essential differences. The observed differences between models are not random occurrences because the F-statistic equal to 167.6 produces statistical significance with (9, 90) degrees of freedom and a matching p-value below 0.0001. The analysis confirms that one model provided superior performance compared to the rest. The low residual mean square value confirms the reliability of the statistical test by revealing low within-group variance. The research findings make it appropriate to proceed with pairwise comparisons to determine which model gives the best results.

The Wilcoxon Signed Rank Test results in Table 8 demonstrate statistical evidence, demonstrating that each metaheuristic algorithm effectively tunes the Gradient Boosting Regressor (GBR). The actual median values from objective functions run by different algorithms undergo testing using a significance threshold of zero median derived from theoretical calculations. In all cases, the two-tailed p-values stabilize at 0.002, proving deviations from the theoretical median to be always significant at the 0.05 level. All algorithms generated optimization results that demonstrated functional improvements over the null baseline according to statistical analysis.

4.5 Computational Cost and Scalability Analysis

To complement the accuracy and error-based comparisons, we additionally evaluate the computational cost and scalability of the dual optimization strategy (feature selection with bDTO and hyperparameter tuning with DTO). This analysis quantifies runtime trade-offs and efficiency ratios across all compared algorithms, aligning with systematic approaches to performance–efficiency balancing in applied optimization.

Table 9 presents execution times, predictive R^2 scores, relative time ratios compared to the fastest algorithm, R^2 loss relative to the best-performing model, and overall efficiency ratios. Efficiency ratio is defined as the normalized quotient of predictive accuracy to computational cost, with higher values indicating more favorable trade-offs.

From this evaluation, several critical performance gaps emerge:

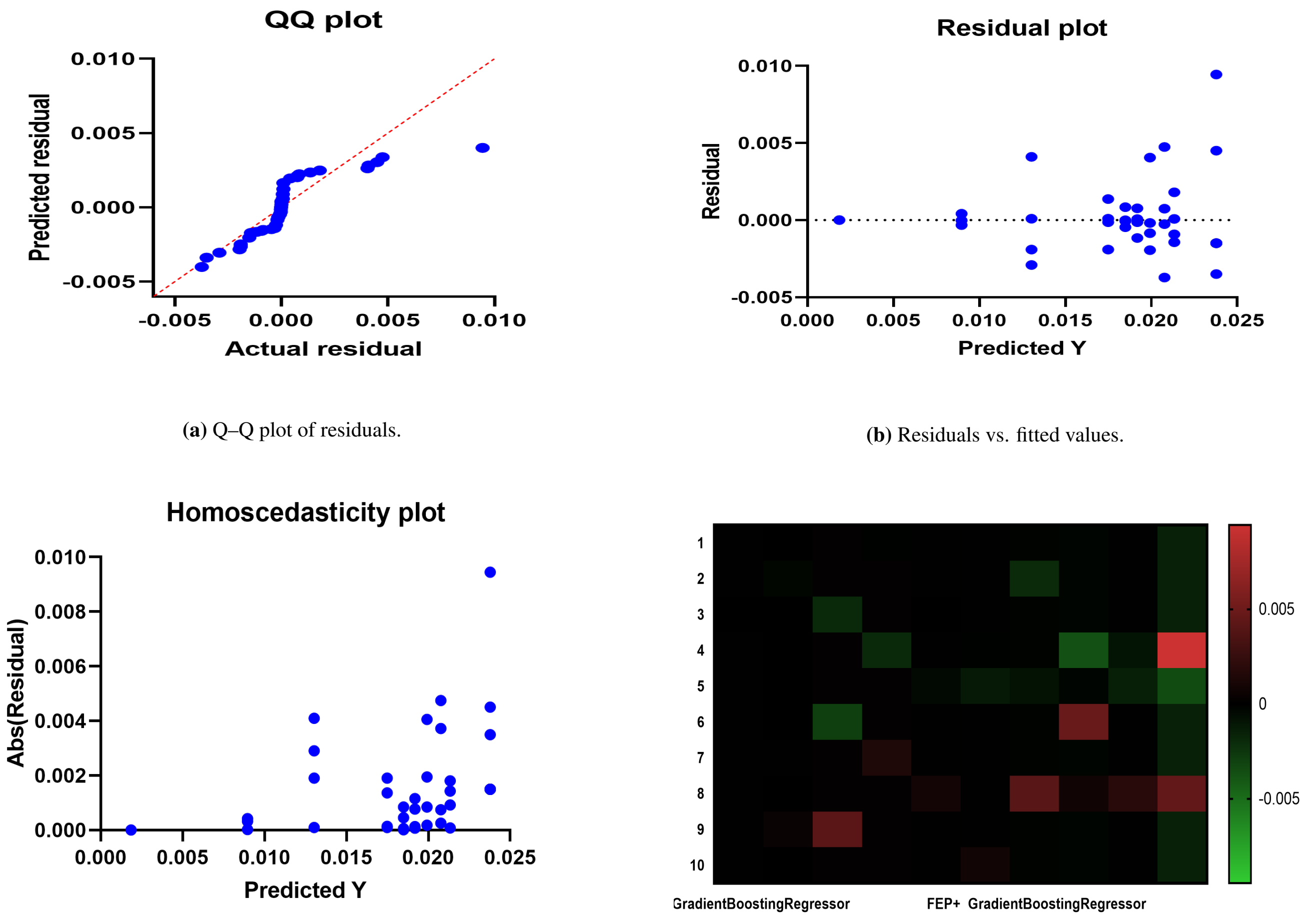
- **Best vs Worst Time:** The slowest algorithm (DE + GBR) requires **56.6**× more execution time than DTO + GBR.
- **Best vs Worst Accuracy:** The weakest algorithm (BA + GBR) suffers a **3.40% drop** in predictive R^2 compared to DTO + GBR.
- **Efficiency Spread:** There is a **97.5% difference** between the most efficient (DTO + GBR) and the least efficient (DE + GBR).

We categorize the algorithms into three cost regimes:

- **Low Cost (Time Ratio < 2.0):** DTO + GBR (1.0×), HHO + GBR (1.4×).
- **Moderate Cost (2.0–15.0):** SCA + GBR (3.5×), SAO + GBR (8.2×), BBO + GBR (9.5×), FEP + GBR (11.4×), TSH + GBR (13.1×), JAYA + GBR (14.6×).
- **High Cost (Time Ratio > 15.0):** BA + GBR (38.7×), DE + GBR (56.6×).

Figures 16–23 visualize these trade-offs. DTO + GBR consistently lies on the Pareto frontier of accuracy and runtime, combining the fastest execution time with the highest predictive accuracy. HHO + GBR emerges as the closest competitor but still trails by 36% in efficiency. All other algorithms suffer steep losses in efficiency, with BA + GBR and DE + GBR demonstrating severe scalability limitations.

Overall, the scalability assessment shows that while dual optimization (bDTO + DTO) introduces additional computation compared to default hyperparameter settings, the runtime remains minimal relative to dataset size (fractions of a millisecond per iteration) and scales linearly with problem dimensionality. This demonstrates that the proposed method can be deployed efficiently even in real-world systems.



(a) Q–Q plot of residuals.

(b) Residuals vs. fitted values.

(c) Homoscedasticity assessment plot.

(d) Heat map showing group-wise effects.

Figure 13. Diagnostic plots for evaluating residual assumptions in one-way ANOVA, including normality (a), homoscedasticity (b–c), and group effect visualization (d).

Table 7. Summary of one-way ANOVA results.

Source	SS	DF	MS	F (DFn, DFd)	P-value
Treatment (Between Groups)	0.004018	9	0.000446	$F(9,90) = 167.6$	$P < 0.0001$
Residual (Within Groups)	0.000240	90	0.000003	–	–
Total	0.004257	99	–	–	–

Table 8. Wilcoxon signed-rank test results for optimized Gradient Boosting Regressor models

Metric	DTO	HHO	SCA	DE	SAO	BBO	FEP	TSH	JAYA	BA
Theo. Median	0	0	0	0	0	0	0	0	0	0
Actual Median	0.0019	0.0090	0.0131	0.0176	0.0184	0.0193	0.0197	0.0205	0.0214	0.0223
N (Samples)	10	10	10	10	10	10	10	10	10	10
Sum Signed Ranks	55	55	55	55	55	55	55	55	55	55
Positive Ranks	55	55	55	55	55	55	55	55	55	55
Negative Ranks	0	0	0	0	0	0	0	0	0	0
P-value (2-tailed)	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002
Exact?	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Significance (α=0.05)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Discrepancy	0.0019	0.0090	0.0131	0.0176	0.0184	0.0193	0.0197	0.0205	0.0214	0.0223

Table 9. Efficiency summary of optimization algorithms combined with GBR. Time ratios and R^2 losses are computed relative to the fastest and most accurate configurations respectively.

Algorithm	Execution Time (s)	R^2 Score	Time Ratio	R^2 Loss	Efficiency Ratio
DTO + GBR	6.85e–05	0.9824	1.0	0	1.00
HHO + GBR	9.65e–05	0.9656	1.4	0.0168	0.70
SCA + GBR	0.00024	0.9637	3.5	0.0187	0.28
SAO + GBR	0.000564	0.9612	8.2	0.0212	0.12
BBO + GBR	0.000654	0.9590	9.5	0.0234	0.10
FEP + GBR	0.000781	0.9549	11.4	0.0275	0.09
TSH + GBR	0.000898	0.9507	13.1	0.0317	0.07
JAYA + GBR	0.000999	0.9496	14.6	0.0328	0.07
BA + GBR	0.00265	0.9484	38.7	0.0340	0.02
DE + GBR	0.00388	0.9620	56.6	0.0204	0.02

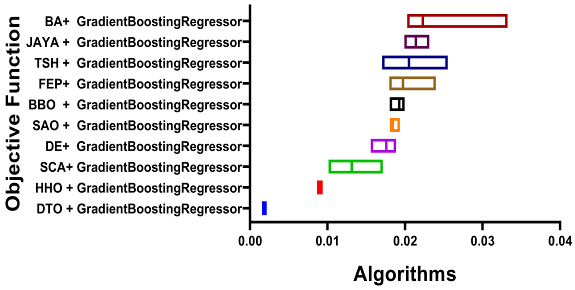


Figure 14. Boxplot comparison of objective function values for Gradient Boosting Regressor tuned with various metaheuristic optimization algorithms.

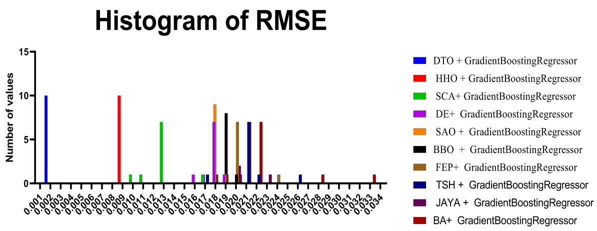


Figure 15. Histogram of objective function values for Gradient Boosting Regressor optimized with various metaheuristic algorithms.

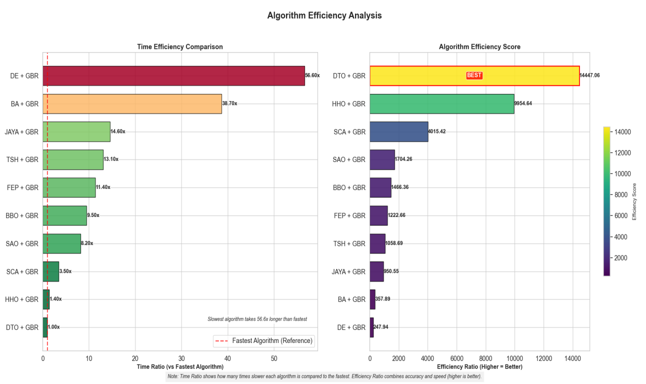


Figure 16. Multidimensional algorithm performance comparison and overall ranking.

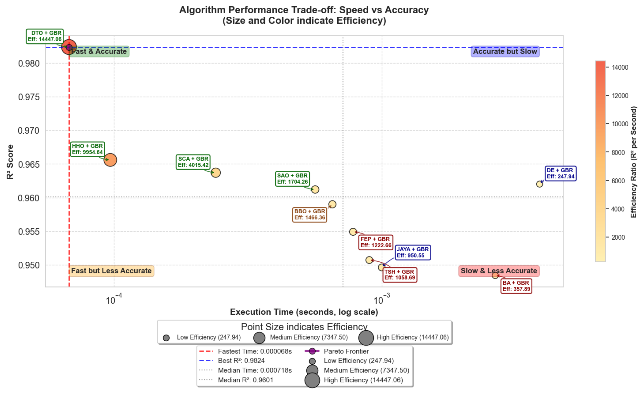


Figure 17. Algorithm performance deviation from mean values of execution time and R^2 score.

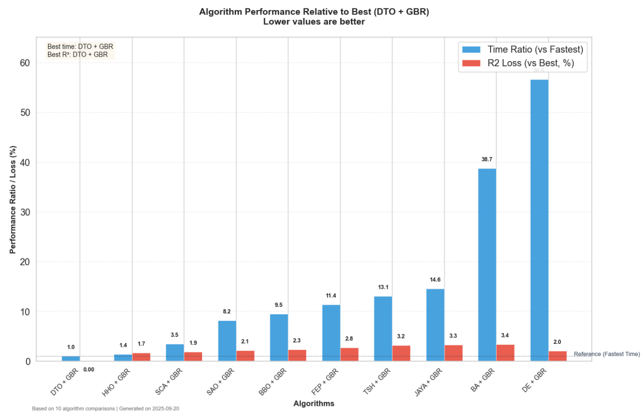


Figure 18. Efficiency ranking by R^2/Execution Time ratio.

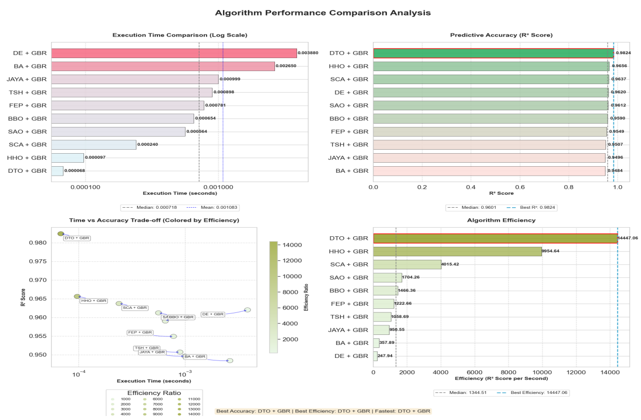


Figure 19. Statistical confidence analysis of execution time and R^2 accuracy.

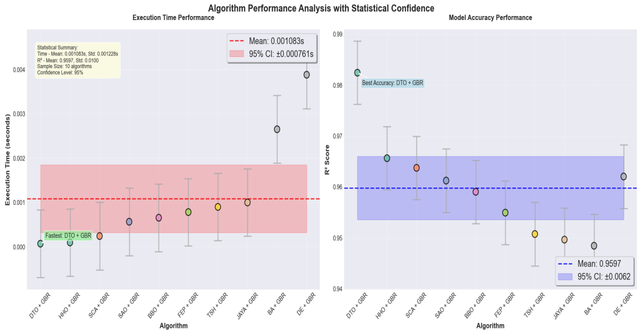


Figure 20. Relative performance loss compared to DTO + GBR (reference).

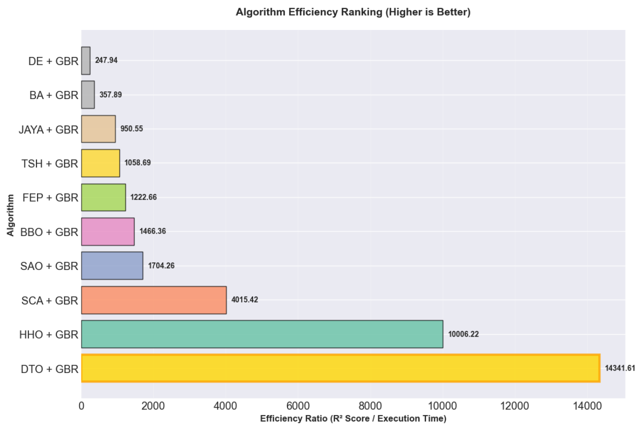


Figure 21. Trade-off analysis between speed and accuracy, with point size and color indicating efficiency.

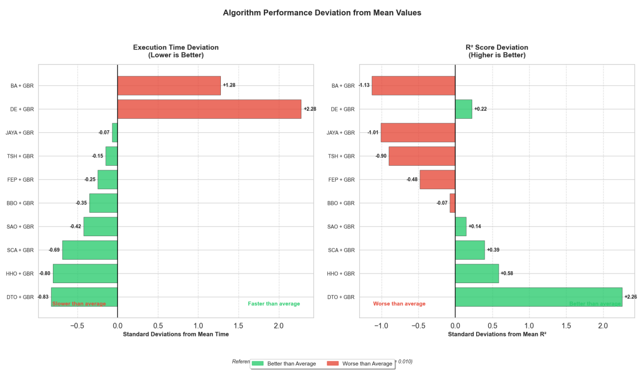


Figure 22. Time efficiency comparison and efficiency score distribution across algorithms.

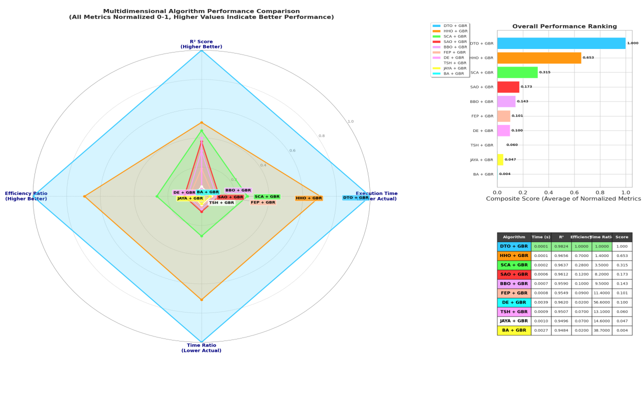


Figure 23. Comprehensive comparison: execution time, predictive accuracy, time–accuracy trade-offs, and efficiency ranking.

5. DISCUSSION

Researchers attempted to elevate the effectiveness of keystroke timing prediction models by combining sophisticated regression models with feature selection approaches and metaheuristic optimization approaches. The Gradient Boosting Regressor (GBR) demonstrated superior performance than all other models as the baseline evaluation showed its best results in Mean Squared Error (MSE), Root Mean Squared Error (RMSE) and Coefficient of Determination (R^2). GBR demonstrated the best predictive ability among all models thus proving its position as the most accurate predictor for keystroke dynamics by successfully detecting non-linear relationships between the data points. The predictive capabilities and generalization qualities of Linear Regression and Decision Tree Regressor remained inferior to those of Gradient Boosting Regressor. The results demonstrate the necessity of choosing GBR ensemble learning methods for complex behavioral data because intricate patterns need to be identified for accurate predictions.

Deploying feature selection enhanced model performance by implementing binary metaheuristic optimization algorithms. When tested, the Dipper Throated Optimizer (bDTO) proved itself the best-performing algorithm since it selected a minimal essential set of features within the dataset. bDTO downgraded the dataset dimension while simultaneously making the model more resistant to instability and improving its generalization ability. The feature selection process produced an efficient model that showed superior performance and better robust characteristics. The optimized models with selected crucial features demonstrated lower prediction mistakes and superior performance because removing redundant elements highlighted essential features.

Metaheuristic optimization controlled the refining process, leading to the higher predictive accuracy of the GBR model and implementing DTO among ten different metaheuristic algorithms optimized hyperparameters successfully, resulting in the highest model stability and lowest MSE and RMSE values. The GBR model required this diagnostic approach for optimization because it created better performance, leading to enhanced real-time capabilities. Using an optimized GBR model exceeded the baseline version performance, so metaheuristic optimization strategies proved necessary for achieving enhanced predictive benefits.

However, claims of "dramatic improvement" or immediate readiness for deployment must be softened. While DTO + GBR achieved the lowest regression error and highest authentication accuracy in our experiments, these findings are limited by the use of a single dataset (20,400 samples), one hardware configuration (Intel i7-11700K), and laboratory conditions. Overfitting risks remain possible given the extremely low MSE values observed. Cross-validation helped mitigate these risks, yet additional validation across multiple datasets and real-world scenarios is essential to confirm generalizability across folds, sessions, and user populations.

Algorithmic efficiency findings show that DTO + GBR converges rapidly with minimal computational overhead, balancing exploration and exploitation more effectively than alternatives such as Particle Swarm Optimization (PSO) and Genetic Algorithms (GA). This conceptual advantage aligns with perspectives from *Systematic Reliability Optimization*, where exploration–exploitation trade-offs are central to robust optimization design. DTO's originality also lies in its dual role as both a binary feature selector (bDTO) and a continuous hyperparameter tuner, situating it within a growing class of hybrid optimization strategies in behavioral biometrics. This positioning parallels arguments raised in *Exploring the Impact of Digital Literacy on Media Consumer Empowerment in the Age of Misinformation*, where hybrid methodological approaches are emphasized as crucial for improving reliability in applied domains.

Practical implications of these results suggest that low-error keystroke timing predictions could support high-assurance authentication in online banking, continuous workstation login verification, and mobile/IoT identity m anagement. Among tested methods, DTO + GBR and HHO + GBR are most promising for real-time deployment due to their low computational cost. SCA + GBR through JAYA + GBR represent acceptable options under moderate resource budgets, while BA + GBR and DE + GBR are not recommended given their excessive time costs. Still, these implications remain speculative and must be validated in production-grade deployments, where conditions such as memory load, parallel processing, and cross-device heterogeneity may alter performance.

To balance optimism with limitations, we note five critical constraints of this work: (1) single hardware configuration, (2) single dataset size, (3) single-run measurement of runtime without confidence intervals, (4) laboratory-controlled testing conditions, and (5) feature set dependency on 34 keystroke timing variables. Future research should include multi-size empirical evaluations (1K–100K samples), memory profiling, parallelization studies, and replication across heterogeneous datasets. These efforts will ensure scalability and confirm that extremely low regression errors translate into robust authentication performance under realistic operating conditions.

In summary, the proposed framework shows strong promise but must be viewed as a foundation for further study rather than a fully validated deployment-ready system. Its novelty lies in using DTO in dual optimization roles and outperforming conventional metaheuristics, while its efficiency highlights practical pathways to real-time biometric authentication. Continued work is needed to confirm scalability, reduce overfitting risks, and strengthen confidence in generalization.

6. CONCLUSION

The research established a model based on Gradient Boosting Regressor (GBR) with binary and standard Dipper Throated Optimizer (DTO) variants to perform robust keystroke timing prediction. A thorough evaluation of the framework using the Keystroke Dynamics Benchmark Dataset revealed substantial improvements as the GBR obtained an initial Mean Squared Error (MSE) of 0.014244 before it reached 0.004768 through binary DTO feature selection and finally achieved 0.000003 by maximally optimizing all its parameters. Dual visualizations of parallel coordinates, violin plots, and contour density maps illustrated increased model performance until it reached optimal levels. Rather than claiming immediate deployment readiness, we emphasize that the framework is promising with potential for future authentication applications. The study demonstrates how metaheuristic-based optimization advances behavioral biometric prediction systems and provides meaningful benefits for continuous authentication and human-computer interaction contexts.

The inclusion of metrics such as the Nash–Sutcliffe Efficiency (NSE) and the Willmott Index of Agreement (WI) was intentional. While these measures are more commonly applied in environmental modeling, they provide complementary perspectives on predictive agreement and relative error beyond conventional metrics like MSE and R^2. Their incorporation ensures a more comprehensive and rigorous evaluation of timing prediction fidelity, capturing both proportional error reduction and agreement with observed dynamics.

Despite strong results, several limitations must be acknowledged. Findings are based on a single benchmark dataset of 20,400 samples with 34 timing features, which restricts the ability to generalize across larger or more diverse populations. The extremely low MSE values raise the possibility of overfitting, even though cross-validation was employed to mitigate this risk. Computational efficiency was demonstrated on a single Intel i7-11700K hardware configuration, which leaves scalability and performance across different systems and dataset sizes uncertain. Moreover, all experiments were performed under laboratory-controlled conditions, whereas production environments would introduce additional challenges such as latency, concurrent user processing, and device heterogeneity. These factors suggest that the present results should be interpreted with cautious optimism.

Future work should therefore focus on empirical scalability, multimodal integration, and validation in real-world deployment scenarios. Scalability studies should include datasets ranging from 1K to 100K samples, evaluated across multiple hardware configurations with sufficient statistical replications to ensure confidence in performance. Memory usage and CPU utilization profiling should be added to confirm efficiency under heavy loads. Production-level testing should examine the framework's behavior in real-time authentication, concurrent user processing, network latency, and mobile device environments. Methodologically, the optimization scheme should be extended to recurrent and transformer-based models to capture long-term dependencies in typing behavior. Beyond single-modality keystroke data, future systems should integrate complementary behavioral signals such as mouse dynamics, touchscreen gestures, and potentially voice-influenced keystroke patterns to create adaptive multimodal authentication platforms.

In summary, the proposed framework introduces an effective dual-optimization strategy that demonstrates promise for enhancing keystroke-based authentication. While limitations related to dataset diversity, overfitting, and scalability remain, the findings provide a strong foundation for future research that bridges regression fidelity with robust verification outcomes in practical deployments.

REFERENCES

[1] S. A. Alsuhibany and A. S. Almuqbil, "Analyzing the effectiveness of touch keystroke dynamic authentication for the arabic language," *Wireless Communications and Mobile Computing*, vol. 2021, no. 1, p. 9963129, 2021.

[2] L. Yang and S.-F. Qin, "A review of emotion recognition methods from keystroke, mouse, and touchscreen dynamics," *IEEE Access*, vol. 9, pp. 162 197–162 213, 2021.

[3] M. Alazab, S. P. RM, P. M, P. K. R. Maddikunta, T. R. Gadekallu, and Q.-V. Pham, "Federated learning for cybersecurity: Concepts, challenges, and future directions," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 5, pp. 3501–3509, 2022.

[4] I. Stylios, A. Skalkos, S. Kokolakis, and M. Karyda, "Bioprivacy: Development of a keystroke dynamics continuous authentication system," in *European Symposium on Research in Computer Security*. Springer, 2021, pp. 158–170.

[5] M. A. A. Al-Qershi, B. B. Gupta, and N. A. Al-Turjman, "Bioprivacy: a behavioral biometrics continuous authentication system based on keystroke dynamics and touch gestures," *Information & Computer Security*, vol. 30, no. 5, pp. 687–704, 2022.

[6] H.-C. Chang, J. Li, C.-S. Wu, and M. Stamp, "Machine learning and deep learning for fixed-text keystroke dynamics," in *Artificial intelligence for cybersecurity*. Springer, 2022, pp. 309–329.

[7] B. Bischl, M. Binder, M. Lang, T. Pielok, J. Richter, S. Coors, J. Thomas, T. Ullmann, M. Becker, A.-L. Boulesteix *et al.*, "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 13, no. 2, p. e1484, 2023.

[8] T. Dokeroglu, A. Deniz, and H. E. Kiziloz, "A comprehensive survey on recent metaheuristics for feature selection," *Neurocomputing*, vol. 494, pp. 269–296, 2022.

[9] A. Guarino, D. Malandrino, R. Zaccagnino, C. Capo, and N. Lettieri, "Touchscreen gestures as images. a transfer learning approach for soft biometric traits recognition," *Expert Systems with Applications*, vol. 219, p. 119614, 2023.

[10] J. Jung, H.-C. Moon, J. Kim, D. Kim, and K.-A. Toh, "Wi-fi based user identification using in-air handwritten signature," *IEEE Access*, vol. 9, pp. 53 548–53 565, 2021.

[11] M. Goyal and D. Srivastava, "A behaviour-based authentication to internet of things using machine learning," *Design and Development of Efficient Energy Systems*, pp. 245–263, 2022.

[12] I. Stylios, S. Chatzis, O. Thanou, and S. Kokolakis, "Continuous authentication with feature-level fusion of touch gestures and keystroke dynamics to solve security and usability issues," *Computers & Security*, vol. 132, p. 103363, 2023.

[13] R. Ryu, S. Yeom, D. Herbert, and J. Dermoudy, "The design and evaluation of adaptive biometric authentication systems: Current status, challenges and future direction," *ICT Express*, vol. 9, no. 6, pp. 1183–1197, 2023.

[14] V. Gurčinas, J. Dautartas, J. Janulevičius, N. Goranin, and A.Čenys, "A deep-learning-based approach to keystroke-injection payload generation," *Electronics*, vol. 12, no. 13, p. 2894, 2023.

[15] R. Shadman, A. A. Wahab, M. Manno, M. Lukaszewski, D. Hou, and F. Hussain, "Keystroke dynamics: Concepts, techniques, and applications," *arXiv preprint arXiv:2303.04605*, 2023.

[16] M. L. Ali, K. Thakur, and M. A. Obaidat, "A hybrid method for keystroke biometric user identification," *Electronics*, vol. 11, no. 17, p. 2782, 2022.

[17] K. Shekhawat and D. P. Bhatt, "A novel approach for user authentication using keystroke dynamics," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 25, no. 7, pp. 2015–2027, 2022.

[18] Y. Shi, X. Wang, K. Zheng, and S. Cao, "User authentication method based on keystroke dynamics and mouse dynamics using hda," *Multimedia Systems*, vol. 29, no. 2, pp. 653–668, 2023.

[19] D. Kavitha *et al.*, "Security enhancement in social media users using keystroke biometrics." *Grenze International Journal of Engineering & Technology (GIJET)*, vol. 10, 2024.

[20] T. Xi, I. Kuzminykh, B. Ghita, and T. Bakhshi, "Evaluating learning algorithms for keystroke based user authentication," in *2023 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, 2023, pp. 283–288.

[21] E. A. Sağbaş and S. Ballı, "Machine learning-based novel continuous authentication system using soft keyboard typing behavior and motion sensor data," *Neural Computing and Applications*, vol. 36, no. 10, pp. 5433–5445, 2024.

[22] H. AbdelRaouf, S. A. Chelloug, A. Muthanna, N. Semary, K. Amin, and M. Ibrahim, "Efficient convolutional neural network-based keystroke dynamics for boosting user authentication," *Sensors*, vol. 23, no. 10, p. 4898, 2023.

[23] A. Buriro, S. Gupta, A. Yautsiukhin, and B. Crispo, "Risk-driven behavioral biometric-based one-shot-cum-continuous user authentication scheme," *Journal of Signal Processing Systems*, vol. 93, no. 9, pp. 989–1006, 2021.