2023

Unlocking User Identity: A Study on Mouse Dynamics in Dual Gaming Environments for Continuous Authentication

Marcho Setiawan Handoko
Minnesota State University, Mankato

Unlocking User Identity: A Study on Mouse Dynamics in Dual Gaming Environments for Continuous Authentication

by

Marcho Setiawan Handoko

An Alternate Plan Paper Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

In

Data Science

Minnesota State University, Mankato

Mankato, Minnesota

November 2023

11/9/2023

Unlocking User Identity: A Study on Mouse Dynamics in Dual Gaming Environments for Continuous Authentication

Marcho Setiawan Handoko

This alternate plan paper has been examined and approved by the following members of the alternate plan paper committee.

Dr. Rushit Dave
Advisor

Dr. Rajeev Bukralia
Committee Member

Dr. Mansi Bhavsar,
Committee Member

Acknowledgments

First and foremost, I would like to express my deepest gratitude to Minnesota State University, Mankato, an institution that has not only equipped me with essential knowledge and skills but has also played a pivotal role in shaping my academic and personal journey. Its diverse environment and rich culture of learning have allowed me to forge countless invaluable connections and gain insights from various perspectives. My heart is filled with appreciation for Professor Rushit Dave, whose unwavering guidance, vast expertise, and relentless support have been nothing short of transformative. His dedication to imparting knowledge, patience in addressing my queries, and consistent encouragement have fueled my passion and determination throughout this research journey. The camaraderie, feedback, and moral support I received from my peers and friends have been indispensable, offering both motivation during challenging times and joy during accomplishments. Furthermore, I'd like to acknowledge the silent supporters, those whose roles might have been indirect but whose positive vibes and faith in my capabilities kept me going. This thesis, while an individual endeavor, was truly shaped by the collective contributions, wisdom, and encouragement of many. To each one of you who has been a part of this journey, I offer my sincerest gratitude.

Glossary

Continuous Authentication	A security mechanism that verifies a user's identity continuously or at periodic intervals during a session, rather than just at the beginning.
Mouse Dynamics	The study and analysis of mouse movements and behaviors as users interact with a computer system.
Authentication	The process of verifying the identity of a user, device, or system.
Behavioral Biometrics	Metrics related to patterns of human activity, often used for authentication or identification purposes.
AUC	A performance metric used to evaluate the effectiveness of a classification model, typically used with the ROC curve.
ROC	A graphical representation that displays the true positive rate against the false positive rate for a binary classifier system as its discrimination threshold varies.
LSTM	A type of recurrent neural network that can remember patterns over long durations of time.
GRU	A type of recurrent neural network that is similar to LSTM but uses a different mechanism to remember patterns.

Sequential Model In the context of neural networks, it refers to a linear stack of layers where data flows in a sequence, from input to output.

Normalization The process of adjusting values in a dataset to a common scale, typically between 0 and 1.

Cross-Validation A statistical technique used to assess the performance of a model by partitioning the original dataset into a training set and a validation set.

F1 Score A measure of a model’s accuracy, which considers both precision and recall.

Outlier An observation that lies far away from other observations, typically seen as an anomaly.

Table of Contents

Table of Figure

Table of Table

Unlocking User Identity: A Study on Mouse Dynamics in Dual Gaming Environments for Continuous Authentication

Marcho Setiawan Handoko

An Alternate Paper Plan in Partial Fulfillment of The
Requirements for The Degree of
Master of Science in Data Science

Minnesota State University, Mankato
Mankato, Minnesota
Nov 2023

Abstract

With the surge in information management technology reliance and the looming presence of cyber threats, user authentication has become paramount in computer security. Traditional static or one-time authentication has its limitations, prompting the emergence of continuous authentication as a frontline approach for enhanced security. Continuous authentication taps into behavior-based metrics for ongoing user identity validation, predominantly utilizing machine learning techniques to continually model user behaviors. This study elucidates the potential of mouse movement dynamics as a key metric for continuous authentication. By examining mouse movement patterns across two contrasting gaming scenarios - the high-intensity "Team Fortress" and the low-intensity strategic "Poly Bridge" the research illuminates the distinct behavioral imprints users leave behind. Such consistent and unique mouse movements emphasize their credibility as reliable biometric markers. The developed sequential model in this research not only demonstrates impressive performance in user verification across these environments but also surpasses benchmarks set by prior research in the field. These findings underscore the potential of mouse movements in revolutionizing the continuous authentication domain, offering heightened security while capturing the intricacies of user behavior across diverse contexts.

1 Introduction

In today’s interconnected world, the digital realm has become an integral part of our daily lives. From online banking and shopping to social interactions and professional work, people rely heavily on digital platforms. This increasing dependency brings forth the critical importance of digital security. According to a recent study on digital security, “unauthorized access to user accounts remains a persistent and growing challenge in the digital ecosystem”. This widespread problem deeply impacts both people and businesses, causing data leaks, financial harm, and a loss of trust in online platforms [3]. Addressing this challenge is complex. Traditional authentication methods, such as passwords or security questions, have been employed to validate user identities. However, these static methods often fall short, as they can be easily compromised or forgotten. Moreover, stringent security measures can sometimes misidentify legitimate users as potential threats, leading to unnecessary barriers and frustrations. This research delves into the potential of continuous authentication, specifically focusing on mouse dynamics as a behavioral biometric. By analyzing the unique ways individuals interact with their devices through mouse movements and clicks, This study aim to develop a more nuanced and reliable method for user verification. This study will explore the efficacy of mouse dynamics in different scenarios, using games like "Poly Bridge" and "Team Fortress 2" as testbeds, and will propose solutions to enhance digital security without compromising user experience.

1.1 Continuous Authentication

Traditional authentication mechanisms have largely been built around the concept of “point-of-entry” verification. Whether it is entering a password, presenting an ID card, or providing a fingerprint, once the user is authenticated, they typically have unhindered access to the system or platform until they decide to log out or the session times out. This method is analogous to checking a guest’s invitation at the door but not monitoring their behavior once they are inside the party.

However, this approach has inherent vulnerabilities. If malicious entities gain initial access, either by stealing credentials or exploiting vulnerabilities, they can operate freely without raising alarms [2]. Furthermore, if a legitimate user’s behavior changes during a session due to coercion or any other reason, traditional systems remain oblivious to this [25].

In contrast, continuous authentication represents a paradigm shift in digital security. Instead of just using initial login details, it highlights the need for continuous checks during a user’s session. This approach has two benefits: it boosts security by constantly checking user actions and improves the user experience by cutting down on repeated logins [9].

Among the various techniques explored for continuous authentication, behavioral biometrics has gained significant attention. Behavioral biometrics focuses on the unique and often subconscious patterns in which individuals interact with their devices. Within this domain, mouse dynamics stands out as a particularly potent metric. Every user exhibits distinct patterns when using a mouse, from the trajectory and speed of movements to the rhythm of clicks. These subtle yet consistent behaviors can serve as a digital signature, offering insights into the authenticity of the user [22]. However, the real challenge is capturing and analyzing these patterns in real time without causing disruptions or false alarms [20].

1.2 Need of Research

In today's digitally connected world, ensuring security and authenticity of users remains a paramount concern. Traditional authentication mechanisms, relying primarily on static credentials, are increasingly coming under scrutiny due to their vulnerabilities. There's a growing realization of the importance of adopting dynamic and continuous authentication methods, particularly in an era rife with advanced cybersecurity threats. Within this context, mouse dynamics, a sub-domain of behavioral biometrics, emerges as a promising avenue for continuous user verification.

The uniqueness of mouse movements for each individual has positioned it as a potential robust indicator of user identity. However, a critical factor influencing these dynamics is the environment in which the mouse is being used. Different tasks or activities can elicit distinct patterns of mouse movements, and understanding these variances is crucial for the development of a reliable authentication system.

While some research has been done in this domain, a significant gap persists. Many studies have narrowly focused on mouse dynamics within a specific type of environment or task, not accounting for the vast spectrum of human-computer interactions. This research aims to explore mouse dynamics across two contrasting environments: high intensity and low intensity, providing a more holistic understanding of user behavior. Towards an era where dynamic and behavior-based authentication becomes the norm, understanding the nuances of mouse movements across varied environments is not just academically valuable, but holds profound implications for enhancing real-world cybersecurity protocols.

1.3 Objective

This study leverages data collected from two distinct video games: Poly Bridge, characterized by its low-intensity gameplay, and Team Fortress 2, known for its high-intensity action. Employing Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models. This study is driven by three primary objectives: First, to develop an effective continuous authentication system employing mouse movement data from Poly Bridge. Second, to extend this authentication approach to the high-intensity environment of Team Fortress 2. And third, to understand and accurately predict user behavior across both gaming environments, tackling the challenge posed by the contrast in intensity levels.

Considering these objectives, our research questions are: How accurately can LSTM and GRU models authenticate users based on mouse movements in the low-intensity environment of Poly Bridge? What is the prediction accuracy in the high-intensity environment of Team Fortress 2? And crucially, how effective are these models in predicting and authenticating users who engage in both gaming environments, demonstrating versatility across varied intensity levels?

Our hypotheses correspond directly to these research questions: H1 posits that the models will effectively predict and authenticate users in Poly Bridge, attributing high accuracy to the consistent and predictable mouse movements typical of low-intensity environments. H2 suggests that while the models will also authenticate users in Team Fortress 2, the prediction accuracy may be challenged by the erratic and rapid mouse movements associated with high-intensity gameplay. H3, our central hypothesis, posits that the models will be capable of predicting and authenticating users across both gaming environments, showcasing an adaptability to the contrasting intensity levels and demonstrating the robustness of our continuous authentication system. Through this exploration, our aim is to contribute valuable insights to the realm of continuous authentication, employing behavioral biometrics to enhance user security and verification processes, particularly in varied and dynamic interaction environments.

1.4 Significance of Study

This study looks at the varied ways people interact digitally, using gaming mouse movements as a lens. Recognizing that user behavior can vary significantly depending on the task at hand, games emerge as an ideal subject for examination due to their capacity to elicit a diverse array of emotions and responses.. For example, a strategy game might show careful mouse moves, whereas an action game might have quick, unpredictable ones. This research, by examining both calm and intense gaming moments, seeks to grasp a full range of user actions.

The main aim is to create a model that understands these interactions. The hope is to design a system that can recognize and confirm users by their unique mouse movements, no matter the game's pace. Such model could provide a flexible and context-aware way to verify users. What sets this study apart is its thorough approach. The data is gathered in a controlled, identical environment where users have the same setup, ensuring the reliability and consistency of the results. Furthermore, This research is not only about general trends. It digs deep into mouse movement details, aiming to spot specific patterns that distinguish one user from another. This is not only good for a reliable authentication system, but it also gives insights into how people and computers interact.

2 Literature Review

The exploration of user behaviors and their authentication in digital platforms has been a focal point of numerous studies in recent years. As digital interactions become more intricate and multifaceted, the need to understand and authenticate these interactions securely becomes paramount. Scholars have delved into various aspects of user behavior, ranging from keystroke dynamics to biometric verification. There are many authentication methods like keyboard dynamic, touch dynamic, mouse dynamic, and other methods. One particular area that has garnered attention is the examination of mouse dynamics, given its potential to offer insights into a user's unique interaction pattern. Siddiqui et al. 2021 [22], Antal et al. 2021 [4], Salman et al. 2019 [19], Chong et al. 2018 [6], Tan et al. 2017 [26], Ciaramella et al. 2022 [7], Marakhtanov et al. 2022 [14] have shown prominent result in mouse dynamic for continuous authentication This literature review seeks to provide a comprehensive overview of the prevailing research on this subject, setting the stage for a deeper dive into the nuances of mouse dynamics and some other continuous authentication.

2.1 Related Work

2.1.1 Keyboard Continuous Authentication

As technology adapts to address emerging challenges, various approaches have surfaced to keep unauthorized users at bay. Among the popular methods that are available, one of them is keystroke dynamics. By analyzing the unique rhythms and patterns of a user's typing behavior, keystroke dynamics offers a non-intrusive, real-time mechanism to authenticate and verify user identities. This method not only taps into the individual nuances of typing speed and pressure but also capitalizes on the habitual mannerisms' users display when interacting with keyboards [17,25].

A study by Aversano et al. (2021) serves as a prime example of this research direction. Their work on keystroke dynamics highlighted its viability as a cost-effective biometric technique. By introducing an ensemble learning approach, they achieved an impressive accuracy of up to 99.7%. Their research underscores the potential of using deep learning techniques for continuous user identification based on typing behavior [5].

Taking a more holistic approach, Li et al. (2020) dove into the combined realm of mouse dynamics, keystroke analysis, and even wrist motion behaviors. Their enhanced continuous authentication framework fills in the security gaps during device transitions or idle times. Their experiments demonstrated a remarkably low False Reject Rate (FRR) and False Acceptance Rate (FAR) for genuine users and attackers, respectively [11].

Meanwhile, Solano et al. (2020) focused on the challenges posed by limited training data, a common hurdle in the field of behavioral biometrics. By leveraging both mouse and keyboard dynamics during static authentication periods, such as login times, their results showcased that effective risk-based authentication is possible even with minimal training inputs [24].

On the cutting edge of neural network applications, Mao, Wang, and Ji (2022) integrated convolutional neural networks with bi-directional Long Short-Term Memory models. Their unique approach combines keystroke content and time as feature vectors, achieving low error rates that solidify the potential of advanced machine learning techniques in this realm [13].

Zeid, El Kamar, and Hassan (2022) conducted an insightful comparative study. Their research delved into the classification capabilities of various algorithms on distinct keystroke datasets. The outcomes indicated a remarkable accuracy, especially with the Random Forest classifier on fixed-text datasets [21].

While keystroke dynamics offer valuable insights for user authentication, they also come with inherent limitations. Variability in user behavior due to factors like fatigue, physical discomfort, or the use of different keyboards can introduce inconsistencies in the analysis. This makes keystroke dynamics potentially more susceptible to inaccuracies compared to more continuous methods. Moreover, continuous typing requires active user engagement, making it less effective during passive or idle states. These challenges can sometimes render keystroke dynamics less desirable compared to other behavioral biometrics, such as mouse dynamics [5,16].

2.1.2 Touch-Based Continuous Authentication.

Continuous authentication methods have evolved significantly, and one standout approach is touch dynamics. Leveraging smartphones, touch dynamics captures the unique ways users interact with their touchscreens either through taps, swipes, or pinches. These interactions, rich with behavioral patterns, offer a seamless way to continuously verify a user's identity. Embedded sensors in modern devices record this interaction data, transforming everyday smartphone usage into a powerful tool for ongoing user authentication. The potential of touch dynamics as a pivotal player in digital security becomes increasingly evident [12,28].

Abuhamad et al. (2020) embarked on an extensive survey, encompassing over 140 contemporary behavioral biometric techniques, with a spotlight on continuous authentication through smartphones' inbuilt sensors. This encompassing study ranged from motion-based methods to touch gestures, emphasizing the immense potential held by smartphones in harnessing behavioral biometrics. It presents an invaluable foundation, delineating the state-of-the-art and charting out challenges for ensuing investigations [1].

In a more application-focused approach, Durmaz Incel et al. (2021) turned their attention to the world of mobile banking. Introducing their system, DAKOTA, they harnessed touch screen and motion sensor data to understand and model user behaviors during banking transactions. With an extensive study involving 45 participants, the insights derived shed light on the viability of continuous authentication, specifically underscoring the prowess of the binary-SVM with RBF kernel [27].

Further enhancing the understanding, Zaidi et al. (2021) elucidated the core principles that steering touch-based continuous mobile device authentication. Their comprehensive study traversed key areas from data acquisition techniques to user classification methods, simultaneously flagging prevailing challenges and areas ripe for exploration. Their emphasis on the requisite for greater acceptance by the research community and market is particularly noteworthy [28].

Venturing into a multi-modal approach, Dave et al. (2022) showcased the synergistic potential of integrating touch dynamics with phone movement. By leveraging two renowned datasets, they evaluated their model's performance across multiple machine learning algorithms. Their results highlight the significant result held by fusing touch dynamics with phone movements, thereby reinforcing the prospects of this approach in bolstering continuous authentication [12].

2.1.3 Mouse Continuous Authentication

In the vast spectrum of continuous authentication techniques, mouse dynamics emerges as one of the foremost. Given the frequent use of mouse or similar pointing devices, such as trackpads, in various computing interfaces, the patterns they generate become pivotal. This is especially evident in web interfaces and applications where typing may be minimal or absent, rendering keyboard-based authentication methods ineffective. The way users move the cursor either through a mouse, trackpad, or any other pointer device creates a signature pattern, is distinct and unique to every individual. This uniqueness transforms simple mouse movements into a robust tool for authentication, seamlessly integrating security without hampering the user experience. As one delves deeper into the realm of mouse dynamics, its potential to revolutionize continuous authentication in digital interactions becomes increasingly apparent [4,22,23].

In the expansive realm of user authentication, a work by Shen et al.~~’s~~ (2013), "Using Mouse Dynamics for Continuous User Authentication," is particularly notable. They propose mouse dynamics, comprising both mouse movements and click patterns, as a behavioral biometric for user verification. Their empirical evaluation based on a diverse dataset demonstrates the method’s potential, particularly emphasized by an AUC score of 0.981. While such results are commendable, the approach does comes with challenges. Assumptions of consistent and distinctive mouse dynamics across users may not always hold true, especially when external factors like fatigue come into play. Further, the method’s broader adaptability is questionable, especially when considering users with physical constraints or those on devices that bypass traditional mouse interactions. A comparison with this research highlights differences, notably in the extended training phase the researcher required, which involved up to 2500 iterations, and the variance in data volumes considered [19].

Delving further, Antal’s (2019) article, "Intrusion Detection using Mouse Dynamics," examines user authentication using mouse dynamics. The research scrutinizes the effectiveness of various classifiers and feature selection techniques for imposter detection via mouse movements. By employing a comprehensive dataset of genuine users, Antal contrasts the performance of these methods against existing techniques. An outstanding finding from this exploration is the optimal imposter detection achieved when combining specific feature selection techniques, such as Relief, with classifiers like Random Forest. Nevertheless, the research does show some limitations. Sole reliance on mouse dynamics could leave the system vulnerable to specific challenges like replay and emulation attacks [3].

The "Not Quite Yourself Today: Behavior-Based Continuous Authentication in IoT Environments" study offers a fresh perspective on behavior-based authentication within IoT. By aggregating data from various devices, this research captures individual movements, object interactions, and even PC usage patterns, subsequently molding machine learning models for authentication. Their Continuous Authentication solution, based on non-sensitive data, boasts a remarkable 99.3% accuracy [10].

"Mouse Authentication without the Temporal Aspect – What does a 2D-CNN learn?" introduces a novel paradigm. Instead of fixed mouse feature extraction, this research advocates for a 2D-CNN that processes images as inputs. The power of transfer learning is harnessed, and the results, especially against baseline models using two public datasets, Balabit and TWOS, are promising. However, the model's pronounced sensitivity to resolution changes presents a potential challenge for real-world application [6].

"SapiMouse: Mouse Dynamics-based User Authentication Using Deep Feature Learning" further explores the potential of 2D-CNN with image inputs for mouse dynamics authentication. Contrasting with Chong's research, this study underlines the 2D-CNN's tendency to focus sparsely on curve information. This research’s results, especially when considering AUC values across various scenarios, suggest the method’s potency [4].

Tan Y’s research, "Insights from curve fitting models in mouse dynamics authentication systems," dives deep into the influence of curve-smoothing techniques on mouse movement prediction tasks. With data from the Balabit challenge, this study underscores the Autoregressive (AR) curve fitting model’s effectiveness. However, the research also emphasizes that curve-fitting solutions might not universally excel across all datasets [26].

The paper "Continuous and Silent User Authentication Through Mouse Dynamics and Explainable Deep Learning: A Proposal" presents an innovative blend of mouse dynamics data transformation with deep learning. Using the Gradient-weighted Class Activation Mapping (Grad-CAM) technique, the research achieves a unique fusion of classification prowess and model explain ability [7].

In "Mouse Dynamics Analysis Using Machine Learning to Prevent Account Stealing in Web Systems," Marakhtanov A echoes Ciaramella G’s approach but introduces the Gradient-weighted Class Activation Mapping (Grad-CAM) for model decision insight. The preliminary evaluations, based on a sample of ten users, signify the efficacy of this approach for silent, continuous user authentication [14].

Shen, Cai, and Guan’s 2012 study introduced a pattern-growth-based mining method that accentuates the stability of mouse characteristics by capturing frequent-behavior segments, achieving impressive false acceptance and rejection rates [20].

Building upon this, Mondal and Bours presented a continuous authentication scheme in 2013 where users were authenticated for each mouse event, demonstrating a robust detection rate where impostors could only perform an average of 94 mouse actions before detection [15]. The same researchers furthered this exploration in 2016, merging keystroke and mouse dynamics and proposing the Pairwise User Coupling (PUC) technique. This combined approach highlighted the power of multi-modal biometrics in enhancing identification accuracy [16].

Diving into the complexities of mouse dynamics, Hu et al. in 2019 combined mouse biobehavioral characteristics with deep learning, offering continuous authentication solutions especially valuable for thwarting insider threats in intranet environments [9]. Siddiqui, Dave, Seliya, and Vanamala in 2022 took a comprehensive look at both machine and deep learning techniques, showcasing an exceptional test accuracy with a 1D-CNN [23]. In the same vein, Antal, Fejér, and Búza in 2021 launched the SapiMouse dataset, emphasizing deep feature learning for mouse dynamics-based user authentication, demonstrating a commendable performance [4]. These studies highlight the significant potential and evolving complexities of using mouse dynamics for authentication.

However, some challenges also emerge. While Siddiqui, Dave, and Seliya's 2021 study achieved impressive accuracy rates using Minecraft's mouse dynamics data, it also highlighted the potential homogenization of data in mundane collection scenarios [22]. Gao et al.'s 2020 research underscored the challenge of limited data during classifier training, although they innovatively combined multiple algorithms to address this [8]. Almalki, Assery, and Roy's empirical evaluation of continuous authentication using mouse clickstream data analysis showcased the potential and challenges of different classifiers in identifying genuine and fraudulent users [2].

Alongside mouse dynamics, other behavioral biometrics like keystroke dynamics have also gained traction. For instance, Os, Skalkos, Kokolakis, and Karyda worked on "BioPrivacy", a keystroke dynamics continuous authentication system, which further solidifies the importance of such non-intrusive measures [25].

Mouse dynamics, although promising, face several challenges in continuous authentication. A significant limitation found in many studies is imbalanced data, where genuine user actions are over-represented compared to malicious ones. This imbalance can lead to biased classifiers, making traditional metrics like FAR and FRR less reflective of real-world performance [18]. The variability introduced by factors such as fatigue, distractions, and different device designs can also impact the consistency of mouse movements. Additionally, the generalizability of mouse dynamics models across varied environments and devices, especially when moving from traditional mice to touchscreens or styluses, is a concern [3,4,19,23].

Furthermore, the adaptability of these models in diverse contexts and user groups, especially those with physical constraints, is debatable. While metrics like accuracy are frequently highlighted, the AUC emerges as a more reliable performance indicator, especially when data is skewed [4,19,22]. For mouse dynamics to be broadly effective, refining evaluation metrics, ensuring data balance, and enhancing model adaptability are essential [4].

2.2 Literature Analysis

The provided table 1 presents a comprehensive summary of various studies focusing on continuous and behavior-based user authentication utilizing mouse dynamics. Each entry in the table outlines a unique approach, detailing the methodology employed, the contribution of the work to the field, and the results achieved. Importantly, the 'Result' column predominantly employs the Area Under the Curve (AUC) metric as a standard for evaluating performance. This choice is justified by the nature of the datasets used, which are imbalanced, rendering metrics like accuracy less reliable. Additionally, even the F1 score, which is generally more robust to imbalanced datasets, is still somewhat impacted in this context.

The studies listed employ a range of techniques, from deep learning models such as VGG16 and Convolutional Neural Networks (CNNs), to machine learning classifiers like Gaussian Naive Bayes and Random Forests. The diversity in methodologies reflects the evolving nature of this research area, as well as the variety of perspectives and approaches taken to address the challenges of user authentication through mouse dynamics. The outcomes, measured primarily using AUC, showcase the effectiveness of these methods, with several studies achieving AUCs above 0.9, indicating a high level of performance in distinguishing between authentic and fraudulent user sessions.

Table 1 offers a valuable overview of the current state of research in continuous user authentication through mouse dynamics, highlighting the prevalent use of AUC as a performance metric due to the imbalanced nature of the datasets involved, and showcasing a variety of innovative methods and contributions to the field.

Table 1: literature analysis

Title	Method	Contribution	Result
Continuous and Silent User Authentication Through Mouse Dynamics and Explainable Deep Learning: A Proposal	Data was mapped into images and deep learning model (VGG16) used for user prediction.	Proposed a method for user detection using data mapping and deep learning.	Achieved an AUC of 0.953. with the precision of 0.897 and recall of 0.896
SapiMouse: Mouse Dynamics-based User Authentication Using Deep Feature Learning	Mouse dynamics data from 120 subjects were collected and preprocessed for training on a Convolutional Neural Network (CNN).	Introduced the SapiMouse dataset for user authentication through mouse dynamics and demonstrated CNN-based user authentication.	Achieved an AUC of 0.94 from the blocks of data. The AUC start to converge on block 3
Using Mouse Dynamics for Continuous User Authentication: Volume 1	Mouse dynamics data acquisition, preprocessing, and feature extraction were performed, with Gaussian Naive Bayes classifier used for classification.	Developed a novel mouse dynamics analysis method for user authentication and compared multiple models.	Achieved an AUC of 0.981 on the benchmark test session.
Mouse Authentication without the Temporal Aspect –	Images of mouse movement sequences were generated and used for 2D-CNN	Introduced a 2D-CNN model for mouse-based user authentication and	Achieved an AUC of 0.958 from the 2D-CNN model.

What does a 2D-CNN learn	training with joint multi-label training.	compared it with baseline methods.	
Insights from Curve Fitting Models in Mouse Dynamics Authentication Systems	Data was structured into mouse event sequences and analyzed using curve fitting techniques with a Linear Support Vector Machine (LinearSVM) classifier.	Investigated the impact of curve smoothing techniques on user authentication and compared time-series forecasting models.	Achieved an AUC of 0.86 with the AR model.
Continuous Authentication Using Mouse Movements, Machine Learning, and Minecraft	Data from 10 users during Minecraft gameplay was used to create Binary Random Forest classifiers for user authentication.	Introduced a Minecraft-based mouse dynamics dataset and evaluated user authentication with Random Forest classifiers.	Achieved an average accuracy rate of 92.73%.
Not Quite Yourself Today: Behaviour-Based Continuous Authentication in IoT Environments	Data from up to twenty users encompassing various behaviors was collected for continuous authentication using machine learning models.	Focused on IoT-based continuous authentication and achieved a 99.3% accuracy rate.	Achieved an accuracy rate exceeding 86.9% in independent user authentication. And with the best of 99.3% accuracy rate.
Machine and Deep Learning Applications to Mouse Dynamics for Continuous User Authentication	Different data preprocessing methods were employed for machine learning and deep learning models, with evaluation using binary and multi-class classifiers.	Evaluated machine learning and deep learning models for user authentication using mouse dynamics data.	Achieved peak accuracy of 85.73% with 1D-CNN and 92.48% with an artificial neural network.

Continuous Authentication Using Mouse Clickstream Data Analysis	Data from 10 users with 39 behavioral features per user were used for verification and authentication with machine learning classifiers.	Demonstrated the effectiveness of machine learning classifiers for user identification with high accuracy.	Achieved AUC of 99.9% in authentication tasks using K-Nearest Neighbors, and 90.3% in Decision Tree and 92.5% in Random Forest.
Intrusion Detection Using Mouse Dynamics	Preprocessed data and performed feature extraction for impostor detection using the Balabit dataset.	Analyzed the Balabit dataset and identified the significance of drag and drop mouse actions for intrusion detection.	Achieved an AUC of 0.92 during benchmark test sessions.

3 Methodology

For this research on continuous authentication utilizing mouse dynamics, a meticulously planned three-step methodology was executed. The primary focus during the data collection stage was gathering data from two distinct games: Poly Bridge and Team Fortress 2, representing low and high-intensity environments, respectively. By choosing these contrasting environments, this study aimed to capture a wide spectrum of mouse behaviors. Transitioning to the data preprocessing and analysis stage, inconsistencies, or anomalies, possibly stemming from sensor glitches or irrelevant interactions, were purged. This curated dataset underwent extraction processes to highlight pertinent features, such as movement speed and trajectories. The normalization process was then applied, ensuring that these features maintain a consistent magnitude, regardless of their source. Following this, we applied exploratory techniques to uncover patterns and potential correlations within the data. For the modeling phase, we specifically employed GRU (Gated Recurrent

Unit) and LSTM (Long Short-Term Memory) models, given their aptitude for handling sequential data like mouse dynamics[2,7]. These models were trained using a partition of the dataset and subsequently evaluated on previously unseen data. Metrics such as accuracy, precision, and recall, along with the ROC curve and AUC values, were employed for performance assessment. Iterative refinement of these models ensured that this research effectively encapsulated the potential of mouse dynamics as a distinguishing mechanism in both high and low-intensity gaming environments for continuous authentication[2].

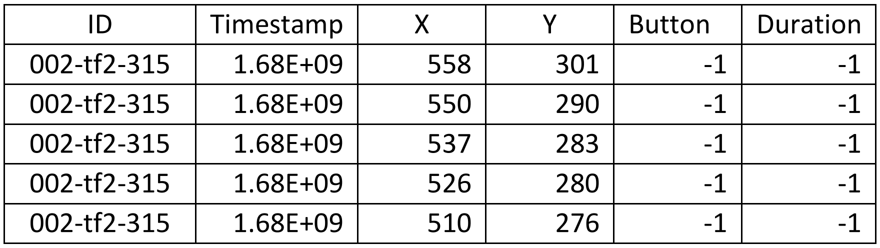
3.1 Data collection

The data gathering process was executed with precision to uphold the reliability and uniformity of the research. A total of 19 distinct users participated in this phase, all of whom were college students, both undergraduate and graduate, coming from various academic backgrounds. The majority of participants were majoring in computer-related fields. Among these participants, 11 played both games: Poly Bridge and Team Fortress 2 (TF2). However, some exclusively engaged with only one of the two games, resulting in a total of 15 users for each game.

The choice of these two games was strategic and deliberate. Poly Bridge, a low-intensity game, was selected because its gameplay mimics office tasks involving more reading and slower, deliberate mouse movements that span across the entire screen. Such movements are analogous to those made during standard office work, making it an ideal choice. The game's popularity and its simple mechanics also meant that it was recognizable to many and easy for newcomers to learn and play, ensuring that participants wouldn't struggle with unfamiliar gameplay dynamics. On the other hand, Team Fortress 2 (TF2) represents high-intensity gameplay, especially given its nature as a first-person shooter (FPS) game. This choice was made to capture data on rapid, reactive mouse movements and understand user behavior under conditions demanding quick reflexes and constant mouse adjustments. Furthermore, the fact that TF2 is freely available ensures wider accessibility and familiarity.

To maintain uniformity and negate the influence of external variables, each participant played on identical computer setups. These setups featured the same mouse, screen specifications, and other hardware conditions. Each gaming session lasted for a duration of 15 minutes. Throughout these sessions, granular mouse interaction data, including button clicks, x and y coordinates, and interaction timestamps, was diligently captured shown in Table 2. This structured approach to data collection was designed to provide a holistic representation of mouse dynamics across different gaming environments and intensities.

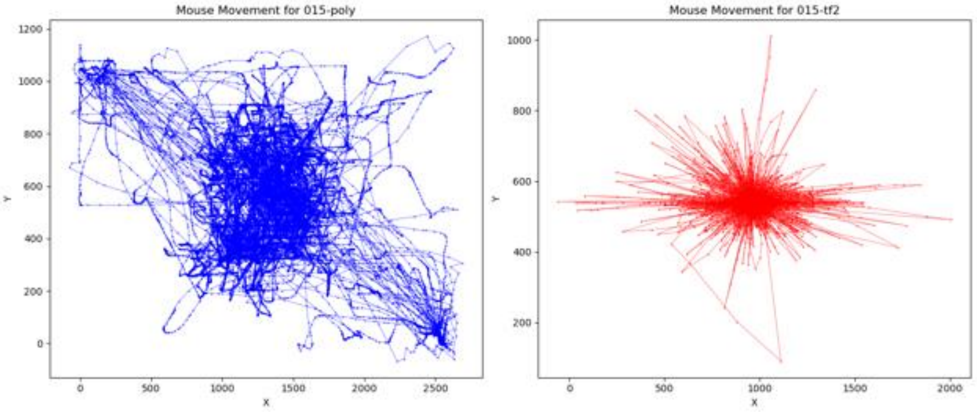
Table 2: data event



ID	Timestamp	X	Y	Button	Duration
002-tf2-315	1.68E+09	558	301	-1	-1
002-tf2-315	1.68E+09	550	290	-1	-1
002-tf2-315	1.68E+09	537	283	-1	-1
002-tf2-315	1.68E+09	526	280	-1	-1
002-tf2-315	1.68E+09	510	276	-1	-1

The mouse movement can be seen in figure 1. The graph shows a comparison of user 15 playing 2 different games: "poly" and "tf2". In the "poly" game, as shown in figure 1a, this research observes a dense, web-like pattern. This intricate map of lines suggests detailed gameplay with frequent direction changes, possibly hinting at multitasking or complex decision-making moments within the game. On the other hand, figure 1b: "TF2" game, showcased on the right side of the figure, displays pronounced radial movements emanating from a central point. These resemble a wheel, hinting at quick, reactive motions, perhaps related to precise aiming or the dynamic pace of action games.

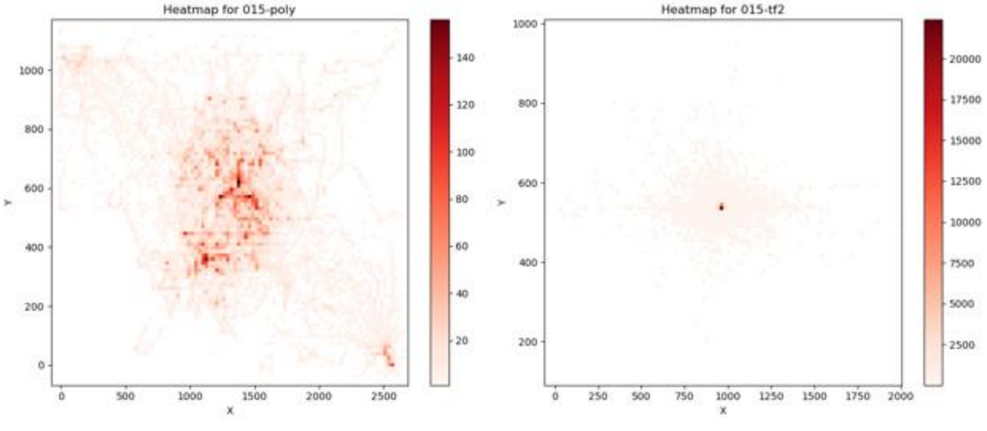
The accompanying heatmaps shown in Figure 2: mouse movement heatmap2 provide further context. Figure 2a: "poly" game displays a pronounced concentration towards the center-left, suggesting a key interaction or game area. In contrast, the heatmap for figure 2b: "tf2" centers around a distinct hotspot, indicating a recurrent return to a central position, possibly between rapid game actions.



(a) poly

(b) TF2

Figure 1: mouse movement a) poly, b)TF2



(a) poly

(a) TF2

Figure 2: mouse movement heatmap a) poly, b) TF2

3.2 Data Preprocessing

The methodology adopted for this research follows a logical and systematic approach to understanding user interactions through their gaming mouse movements shown in Figure 3. The first phase involved data collection, where raw data representing these mouse movements were amassed. Following this, the data underwent a data cleaning process to rectify any inaccuracies, inconsistencies, or anomalies present.

Post-cleaning, the Data Processing phase commenced. During this phase, the data was first normalized to ensure that features have the same scale. Subsequent to normalization, the data was transformed into a binary format through relabeling. This binary representation simplifies complex labels, making them more understandable and easier to work with. After this transformation, the data was sequenced, preserving the order of mouse movements and capturing the temporal essence of user interactions. The processed data was then partitioned into training and testing sets, ensuring a balanced representation.

Before delving into model training, a critical step undertaken was Cross-Validation. This step is indispensable as it provides insights into the model's capability to generalize on unseen data. It acts as a preliminary validation technique, ensuring that the model is not just memorizing the training data but learning to make genuine predictions.

With the insights gained from cross-validation, the final phase involved Training the Model. Using the training dataset, the model was trained to discern patterns in mouse movements, aiming to distinguish one user from another based on these patterns.

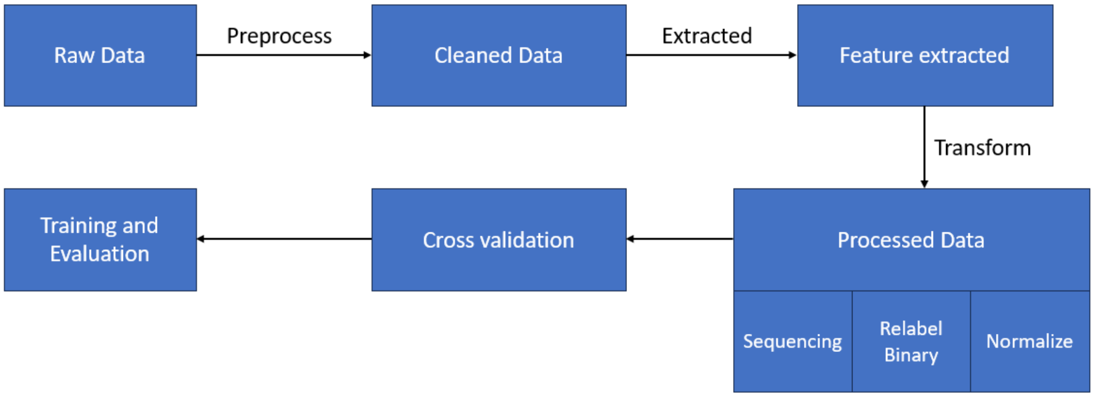


Figure 3: Data pre-processing.

3.4 Data Cleaning

The fundamental goal of data cleaning is to ensure that the dataset being utilized is free from errors and inconsistencies that could undermine the accuracy of any ensuing analysis or model. This study started by systematically examining the dataset for any missing or anomalous values. It was reassuring to discover that the dataset was comprehensive, with no missing values. This is a testament to the meticulousness with which the data collection phase was executed. Moreover, while the analysis did reveal the presence of outliers, as depicted in the box plots shown in Figure 4, a deeper inspection clarified that these outliers were not anomalies or errors. Instead, they represented genuine data points that captured the unique behavioral patterns of certain users.

Given the nature of this study, where the aim is to comprehend the nuanced interactions of users, it was imperative to retain these outliers. These data points provide a richer understanding of user behaviors, especially since they deviate from the norm. They could be pivotal in distinguishing between different user interactions, making them invaluable to the study.

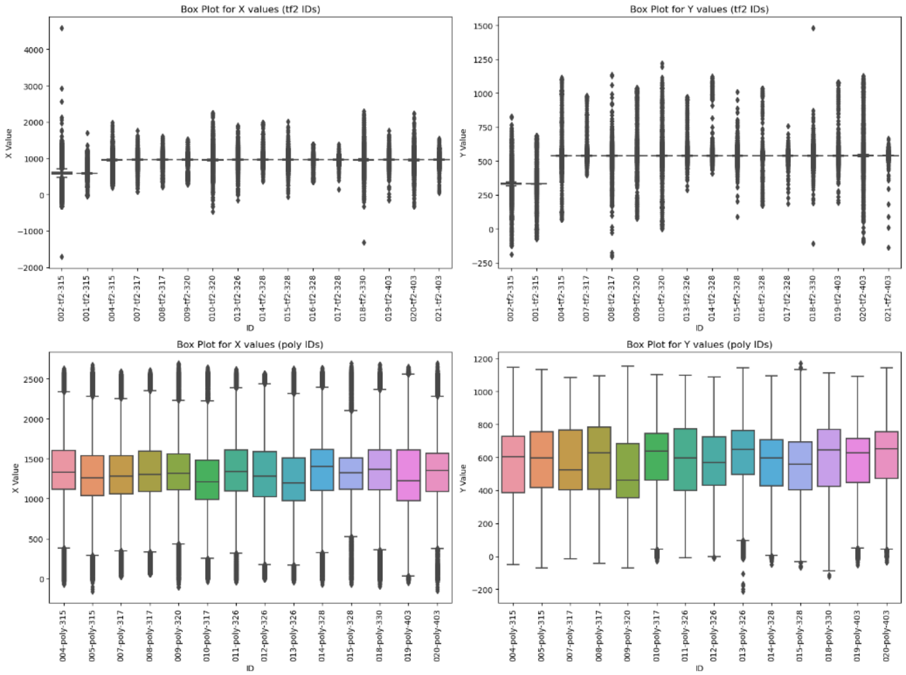


Figure 4: box plot data x and y

3.5 Feature Extraction

The feature extraction phase was instrumental in transforming the raw mouse data into actionable insights. Initially, a broad set of potential features was derived from the data, capturing various nuances of mouse movements and interactions. This included parameters like movement speed, click patterns, trajectories, and many others. Each of these extracted features was then subjected to rigorous statistical tests, including multicollinearity assessments, to ascertain their individual and combined relevance. The goal was to distill the large set of features into a more focused subset, which not only captured the essence of user behavior but also optimized the efficiency and effectiveness of the subsequent

The data is transformed into Delta X, Delta Y, Movement Distance, Velocity, Acceleration, Angle, Jerk, Curvature, Direction Change, Is Stop, and Stop Duration. Delta X and delta Y represent the differences in the X and Y coordinates between consecutive data points. They indicate how much the mouse has moved in the X and Y directions between consecutive timestamps. Movement distance is the Euclidian distance between two consecutive points, representing the straight-line distance the mouse has moved which calculated using $\sqrt{\Delta x^2 + \Delta y^2}$. Velocity is the speed which is calculated by distance divided by the time. Acceleration is the rate of change of velocity which is calculated by finding the difference in consecutive velocities. Angle is the angle of movement between consecutive data points. It is calculated using trigonometry and provides the direction of movement in a 2D plane. arctan2(ΔY, ΔX). Jerk is the rate of change of acceleration. It can capture sudden starts or stops in movement. Curvature represents the degree to which a curve (made by the mouse movement) deviates from being flat or straight. A higher curvature value could indicate more curvy or circular movement patterns, while a low curvature suggests straighter movement. It is calculated by difference in consecutive angles divided by movement distance. Direction change is a binary feature indicating if there was a change in the movement direction between consecutive points. It can be indicative of certain behavioral patterns, like erratic mouse movements. “One” is if the difference in consecutive angles is not zero, otherwise “zero”. “Is_stop” is a binary feature indicating whether the movement of the mouse has stopped. A movement is considered a "stop" if its movement distance is below the threshold (0.001). “Stop_Duration” represents the duration for which the mouse has stopped moving. If the mouse is moving, the stop duration is 0. Otherwise, it is equal to the duration for that data point.

During the preprocessing phase of the analysis, both the model’s robustness and computational efficiency are prioritized in the feature selection process. It is observed that velocity, a crucial measure of movement, was essentially mirrored by the movement distance, making it redundant. Similarly, while the changes in X and Y positions, represented by “Delta_X” and “Delta_Y”, were not linearly correlated, they were closely related to the absolute X and Y coordinates, introducing unnecessary redundancy. Further, the binary nature of the “Is_Stop” feature was found to be negatively correlated with “Stop_Duration”. Given the depth of information provided by “Stop_Duration”, which quantifies the length of each stop, and considering frequent mouse movements leading to a majority of zeros in “Stop_Duration”, “Is_Stop” is omitted. After this rigorous analysis, the refined feature set for modeling included X, Y, “Stop_Duration”, “Jerk”, “Direction_Change”, “Movement_Distance”, Acceleration, Button, and Angle shown in

Table 3. This selection ensures that the data’s essential dynamics are captured while benefiting from enhanced model interpretability and computational efficiency. Based on the analysis, it was decided to only take some features which are X, Y, stop duration, jerk, direction change, movement distance, acceleration, button, and angle. The chosen feature correlation can be seen in Figure 5 which shows no multicollinearity between features. Jerk and acceleration indeed still have a high correlation, but because it still captures the change of acceleration especially when there is a sudden stop in mouse movement.

Table 3: Features

Feature	Description
X	Horizontal position of the mouse cursor on the screen, indicating its exact location along the x-axis.
Y	Vertical position of the mouse cursor on the screen, indicating its precise location along the y-axis.
Stop_Duration	Measures the duration of pauses or stops during mouse movement, providing insights into periods of cursor inactivity.
Jerk	Quantifies abrupt changes in mouse velocity, helping identify sudden shifts in cursor movement or user actions.
Direction_Change	Tracks the frequency of changes in mouse direction, indicating how often the cursor alters its course during movement.
Movement_Distance	Calculates the total distance covered by the mouse cursor, offering insights into the extent of cursor travel.

Acceleration	Measures the rate of change in mouse velocity, gauging the speed at which the cursor accelerates or decelerates.
Button	Records mouse button press events, tracking clicks and releases to monitor user interactions with mouse hardware.
Angle	Represents the angular direction of mouse movement, measuring the angle between the mouse's start and end positions.

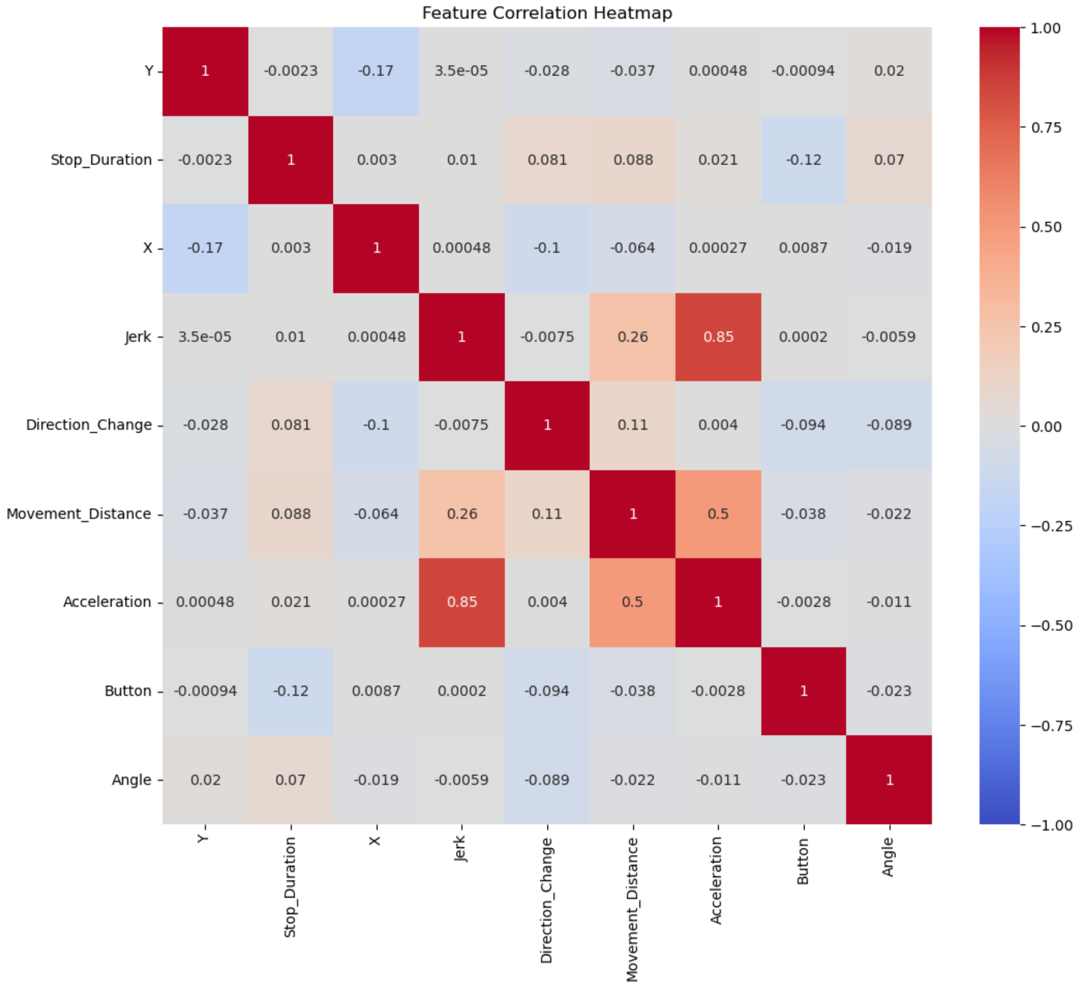


Figure 5: feature correlation

3.6 Data Transformation

The extracted data showed some null value or 0 because the calculation impacted from previous rows, which makes the first and last row extracted value tempered. For this reason, the first and last row of every user is omitted. Also to ensure the model effectively captures temporal patterns in the data, the dataset was transformed into sequences. Each sequence consists of 40 data points, providing a holistic view of short-term dynamics within sessions. The decision to use a sequence length of 40 was informed by the average session length observed in the dataset. This choice strikes a balance between encapsulating meaningful patterns and ensuring computational efficiency. By framing the data in this sequence-based structure, this study aims to harness the full potential of the chosen GRU (Gated Recurrent Unit) and LSTM (Long Short-Term Memory) model, which excels at understanding temporal dependencies.

To streamline the analysis focused on user authentication, the dataset was transformed into a binary classification format centered around a selected user, in this case, user 18. Under this transformation, data corresponding to user 18 was labeled as ‘0’, indicating the authentic user, while data from all other users was labeled as ‘1’, representing potential intruders. This binary distinction enables the model to differentiate between the genuine user’s behavior and any anomalous or intrusive patterns, making it particularly tailored for intrusion detection based on user behavior.

3.7 Model Development

Upon refining the features, the research transitioned to the intricate phase of model development. Here, the choice of GRU and LSTM models was deliberate, given their proven proficiency in handling sequential data. These models were trained to perform binary classification tasks, aiming to discern patterns and predict user behavior for each game individually. Additionally, an integrated model was developed for participants who engaged in both games, providing a holistic view of their mouse dynamics across varied gaming intensities. During the model development phase shown in Figure 6, a distinctive strategy was adopted wherein five individual models were created for each game. Each of these models was designed to predict the behavior of a specific participant (out of the chosen 5) against the collective behavior of all other participants. This binary classification approach was tailored to highlight any unique behavioral dynamics and patterns exhibited by these selected individuals as compared to the broader group of participants.

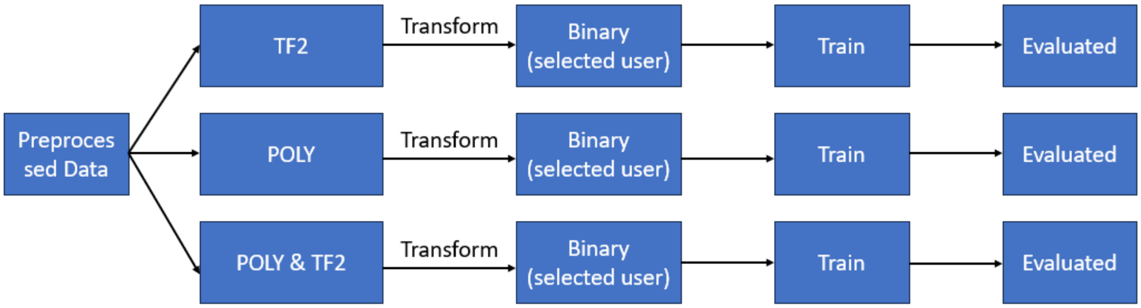


Figure 6: Model Flow

3.7.1 LSTM

LSTM, which stands for Long Short-Term Memory, is distinguished by its ability to model and remember long-range dependencies within sequential data. The foundation of LSTM lies in its intricate cell structure, which includes three key elements: the cell state, the hidden state, and three vital gating mechanisms: the input gate, the forget gate, and the output gate.

The cell state serves as the long-term memory reservoir of the LSTM. Its unique characteristic is its capability to retain and propagate information across extended sequences. This enables the LSTM to capture intricate relationships and dependencies between elements in the data that are widely separated in time. The cell state is subject to dynamic changes through interactions with the gating mechanisms, which regulate the flow of information within the LSTM. The hidden state plays the role of short-term memory within the LSTM. It captures essential information from both the current input and the previous hidden state, contributing significantly to the final predictions or outputs of the network. The functioning of the gating mechanisms is closely linked to the dynamics of the hidden state.

The gating mechanisms of LSTM are fundamental to its efficacy since they regulate the information flow within the cell. The input gate is pivotal in determining what information from the current input should be added to the cell state. It employs a sigmoid activation function to assess the relevance of the new input and then combines it with a hyperbolic tangent (tanh) transformation of the input. This process enables the LSTM to make informed decisions about how much of the new information should be incorporated into the cell state.

The forget gate assumes a critical role in deciding what information from the cell state should be discarded. It evaluates both the previous hidden state and the current input and produces a value between 0 and 1 for each element in the cell state. This value dictates how much of the existing information should be retained or forgotten, allowing the LSTM to adaptively manage its long-term memory.

The output gate is responsible for controlling which information from the cell state should be extracted and included in the hidden state. Similar to the input gate, it employs a sigmoid activation function to determine the relevant portions of the cell state that should contribute to the output, ensuring that the hidden state reflects pertinent information for the current context. LSTMs excel in tasks that demand the modeling of complex sequential patterns and the capture of long-range dependencies. These capabilities make them well-suited for sequence data.

In Figure 7, the study presents an implementation of a Long Short-Term Memory (LSTM) neural network, designed to predict user behavior based on mouse movement data. LSTM, a variant of recurrent neural network (RNN), is particularly adept at handling sequence prediction tasks due to its ability to capture long-term dependencies and sequential patterns in the input data.

The Code in Figure 7 starts by importing the requisite libraries and modules, which includes 'MinMaxScaler' from 'scikit-learn' for feature scaling, 'train_test_split' for segregating the dataset into training and testing subsets, and various components from

'TensorFlow' and 'Keras' to facilitate the construction and training of the neural network model.

Following the import statements, it proceeds to define precision, recall, and Area Under the Receiver Operating Characteristic Curve (AUC-ROC) as the evaluation metrics for the neural network model. These metrics are integral to assessing the model’s performance, providing quantitative insights into its accuracy, the balance between false positives and negatives, and its capacity to distinguish between the two classes.

The central portion of the code is devoted to the construction of the LSTM-based neural network model. The model is initialized as a sequential model, indicating a linear stack of layers. It is then comprised of three LSTM layers, each containing 50 units, with dropout layers interspersed to mitigate the risk of overfitting. The 'return_sequences' parameter is set to 'True' for the initial two LSTM layers, ensuring the entire sequence of outputs is propagated to subsequent layers, a critical requirement for maintaining the temporal dependencies in sequential data. The network concludes with a Dense layer, outfitted with a single neuron and a 'sigmoid' activation function, aligning with the requirements of binary classification tasks.

Following the assembly of the network, the 'compile' method is applied to configure the model for training, defining 'adam' as the optimization algorithm, 'binary_crossentropy' as the loss function, and including the previously established evaluation metrics.

Subsequently, the model undergoes training on the pre-processed mouse movement data, employing the 'fit' method with the number of epochs set to 15 and the batch size configured at 64. The 'validation_data' parameter is employed to provide an independent dataset for validation, facilitating ongoing assessment of the model’s performance during training.

The choice of 15 epochs is rationalized by the observation of the AUC-ROC score’s behavior, noting a plateau beyond this point. This plateauing suggests that additional epochs do not contribute meaningfully to enhancements in the model’s validation performance, thus ensuring an efficient utilization of training time and computational resources while maintaining a robust model performance.

```
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense, Dropout
import tensorflow as tf

precission = tf.keras.metrics.Precision()
recall = tf.keras.metrics.Recall()
AUC_ROC = tf.keras.metrics.AUC(curve='ROC')

# 3. Build the LSTM model
model = Sequential()
model.add(LSTM(50, input_shape=(X_train.shape[1], X_train.shape[2]),
return_sequences=True))
model.add(Dropout(0.2))
model.add(LSTM(50, return_sequences=True))
model.add(Dropout(0.2))
model.add(LSTM(50))
model.add(Dense(1, activation='sigmoid'))

model.compile(optimizer='adam', loss='binary_crossentropy',
metrics=[precission, recall, AUC_ROC])

# 4. Train the model
model.fit(X_train, y_train, epochs=15, batch_size=64,
validation_data=(X_test, y_test))
```

Figure 7: LSTM code

3.7.2 GRU

Gated Recurrent Unit (GRU) offers a simplified alternative to the LSTM architecture while retaining many of its strengths. GRUs also include a cell state and a hidden state, but they use only two gating mechanisms: the reset gate and the update gate.

The cell state and hidden state in GRUs serve analogous roles to their counterparts in LSTMs. The cell state retains long-term dependencies, while the hidden state captures short-term information and contributes to the model's predictions. The reset gate assumes a critical role in deciding what information from the previous hidden state should be reset or forgotten. It takes into account both the current input and the previous hidden state to determine a reset factor, allowing the model to decide which information from the past remains relevant to the current context. The update gate combines information from the current input and the previous hidden state to make decisions about updating the cell state. Additionally, it regulates the rate at which fresh data is incorporated into the cell state, guaranteeing a dynamic equilibrium between short- and long-term memory. Because they have fewer gating mechanisms than LSTMs, GRUs are renowned for their computational efficiency. They shine when dealing with limited computational resources or when rapid model development is essential. However, LSTMs tend to outperform GRUs on tasks involving intricate sequential patterns and long-range dependencies.

In the development of the continuous authentication system utilizing a Gated Recurrent Unit (GRU) neural network, this study analyzes user identity through mouse movement patterns show in Figure 8. The initial stage encompasses preprocessing of the dataset, utilizing the MinMaxScaler from sklearn.preprocessing, ensuring that all features within the dataset contribute uniformly to the model’s training. The train_test_split function subsequently partitions the dataset into distinct training and testing subsets, creating a foundation for robust model evaluation.

The construction of the GRU model is facilitated through TensorFlow’s Keras API, employing a Sequential model to enable a structured layer-by-layer assembly. The network integrates three GRU layers interspersed with Dropout layers, strategically placed to prevent overfitting. The first GRU layer, comprising 50 units, is configured to accept input based on the training data’s dimensions, returning sequences to maintain a continuous information flow. The subsequent GRU layers, each with 50 units, perpetuate this structure, with the second GRU layer also configured to return sequences. Dropout layers follow the first two GRU layers, with a rate of 0.2, aiming to randomly deactivate a fraction of input units at each training update.

Concluding the network’s architecture is a Dense layer, incorporating a single unit with a sigmoid activation function. This configuration facilitates the transformation of the model’s output into a probability score, indicating the likelihood of the input data being associated with a particular user. The compilation of the model utilizes the Adam optimizer and binary cross-entropy as the loss function, aligning with the binary classification requirements of the task. Metrics including precision, recall, and AUC-ROC are deployed to gauge the model’s performance, offering insights into its predictive accuracy and reliability.

The training of the model spans 15 epochs, with a batch size set at 256, and incorporates the testing dataset as validation_data. This decision for the number of epochs is informed by the observation that the AUC score, a critical metric for the model’s performance, begins to flatten post the 15-epoch mark. This plateau in the AUC score indicates that additional epochs do not yield substantial improvements in the model’s ability to distinguish between classes, and may in fact, increase the risk of overfitting to the training data. Thus, the choice of 15 epochs strikes a balance, aiming for optimal model performance on unseen data while mitigating the risk of overfitting, ultimately enhancing the robustness and reliability of the user authentication system across diverse gaming environments.

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import GRU, Dense, Dropout
import tensorflow as tf

# Metrics
precision = tf.keras.metrics.Precision()
recall = tf.keras.metrics.Recall()
AUC_ROC = tf.keras.metrics.AUC(curve='ROC')

# Build the GRU model
model = Sequential()
model.add(GRU(50, input_shape=(X_train.shape[1], X_train.shape[2]),
return_sequences=True))
model.add(Dropout(0.2))
model.add(GRU(50, return_sequences=True))
model.add(Dropout(0.2))
model.add(GRU(50))
model.add(Dense(1, activation='sigmoid'))

model.compile(optimizer='adam', loss='binary_crossentropy',
metrics=[precision, recall, AUC_ROC])

# Train the model
model.fit(X_train, y_train, epochs=15, batch_size=256,
validation_data=(X_test, y_test))
```

Figure 8: GRU code

3.8 Evaluation

Model evaluation is paramount to gauge the effectiveness and reliability of developed models. Given the inherent data imbalance introduced by the differential participation in the two games, traditional accuracy metrics could be misleading. Thus, the AUC (Area Under the Curve) and ROC (Receiver Operating Characteristic) were chosen as the primary evaluation metrics, offering insights into the model's true positive rate versus its false positive rate, especially in imbalanced scenarios. Additionally, the F1 score, which considers both precision and recall, was used as a complementary metric. While the F1 score can sometimes be influenced by data imbalances, its inclusion provided a more rounded perspective, ensuring that both the model's precision and its ability to recall true positives were assessed.

4 Result

4.1 Poly Bridge Result

This research evaluates a continuous authentication model using mouse dynamics across five users: USER20, USER13, USER4, USER15, and USER8 as shown in Table 4. Key metrics, including the F1 score, AUC, and ROC, were analyzed for both training and testing datasets. Highlights include USER20's outstanding metrics in the training phase with an F1 score of 0.98, and an AUC and ROC of 0.98 and 0.99 respectively. USER13 exhibited commendable performance, albeit with a hint of overfitting in the testing phase. Users 4 and 15 showed consistent metrics, reinforcing the model's uniformity across varying user behaviors. USER8's results reinforced the model's robustness, with metrics that rivaled those of USER20. Averaging the metrics across all users revealed an impressive F1 score of 0.98, AUC of 0.97, and ROC of 0.99 during training. In essence, the model rooted in mouse dynamics proves proficient in distinguishing genuine from deceptive mouse patterns, showcasing its promise in continuous authentication.

Table 4: result poly bridge

POLY			GRU	LSTM
USER20	Train	F1	0.98	0.97
		AUC	0.98	0.94
		ROC	0.99	0.95
	Test	F1	0.98	0.97
		AUC	0.98	0.94
		ROC	0.98	0.95
USER13	Train	F1	0.982	0.96
		AUC	0.95	0.84
		ROC	0.98	0.86
	Test	F1	0.981	0.96
		AUC	0.97	0.85
		ROC	0.97	0.86
USER4	Train	F1	0.98	0.96
		AUC	0.97	0.86
		ROC	0.99	0.89
	Test	F1	0.98	0.96
		AUC	0.98	0.87
		ROC	0.98	0.88
USER15	Train	F1	0.98	0.96
		AUC	0.96	0.84
		ROC	0.98	0.87
	Test	F1	0.98	0.96
		AUC	0.97	0.85
		ROC	0.98	0.86
USER8	Train	F1	0.98	0.98
		AUC	0.97	0.95
		ROC	0.99	0.97
	Test	F1	0.98	0.97
		AUC	0.98	0.96

		ROC	0.99	0.96
Average	**Train**	**F1**	**0.98**	**0.97**
		AUC	**0.97**	**0.89**
		ROC	**0.99**	**0.91**
	Test	**F1**	**0.98**	**0.96**
		AUC	**0.98**	**0.89**
		ROC	**0.98**	**0.90**

When assessing User 15’s performance metrics for "Poly ridge" using the GRU and LSTM models, there are distinct differences in model behaviors as shown in Figure 9Figure 9: ROC: Poly user15. The LSTM model’s confusion matrix shown in Figure 10, presents 101,950 true positives and 1,308 true negatives. However, the model encountered a noteworthy count of 6,020 false positives, which indicates occasional over-predictions of label 1 in certain instances. Contrarily, its false negatives were relatively limited at 540.

In contrast, the GRU model’s confusion matrix shown in Figure 11, showcases a more balanced performance. With 101,732 true positives and 5,066 true negatives, it significantly curtailed false positives to 2,262. This highlights the GRU model’s enhanced accuracy in predicting the 0 label, even though its false negatives increased slightly to 758.

Further insights emerge from the ROC curve. The GRU model, depicted by the green curve, achieves a near-perfect AUC of 0.98, placing it proximate to the top-left corner, a sign of optimal predictive capability. The LSTM model, represented by the blue curve, lags behind with an AUC of 0.86, suggesting a slightly lesser degree of class separability.

The result aligned with our first hypothesis. Hypothesis 1 (H1) posited that the model would exhibit high accuracy in predicting and authenticating users within the Poly Bridge environment, characterized by its low-intensity gameplay. The results from the Poly dataset affirm this hypothesis, as the GRU model achieved remarkable performance with high AUC, ROC, and F1 scores. This underscores the model’s adeptness at learning from consistent and predictable mouse movements, validating the assertion that low-intensity environments contribute positively to the accuracy of user authentication.

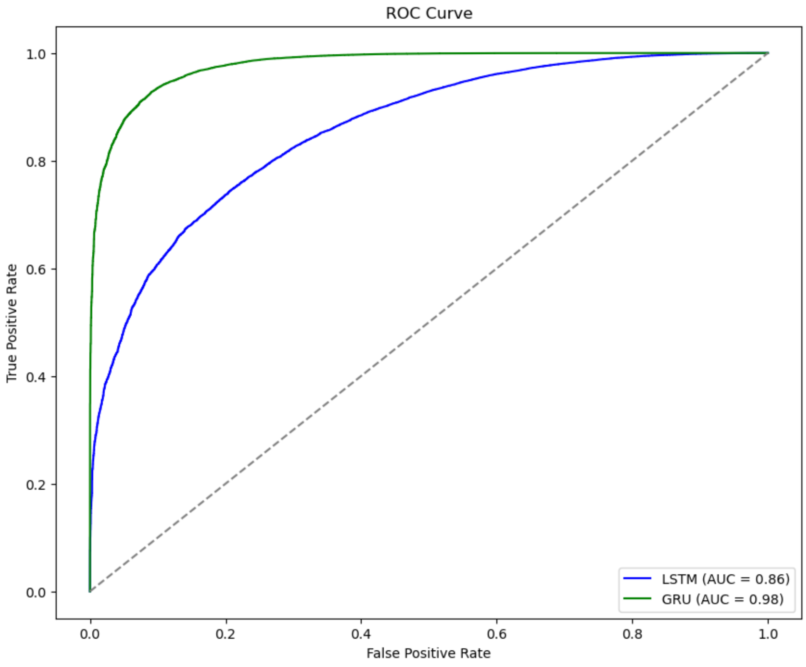


Figure 9: ROC: Poly user15

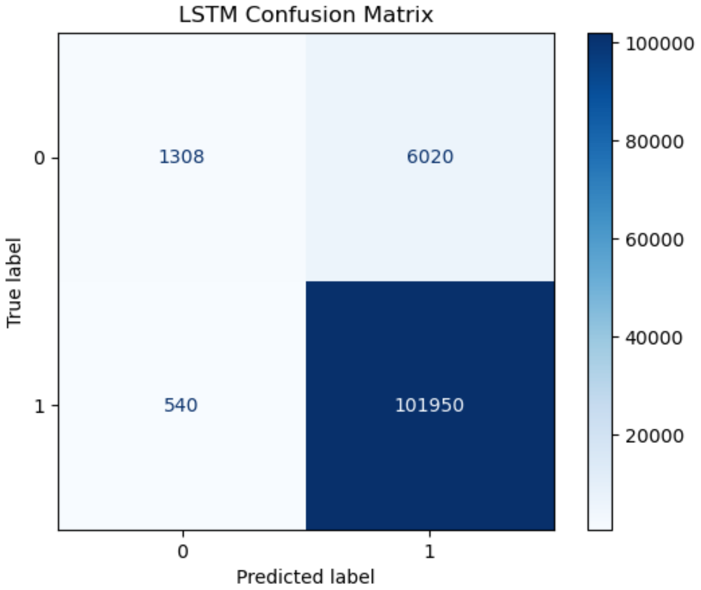


Figure 10: Poly LSTM user15

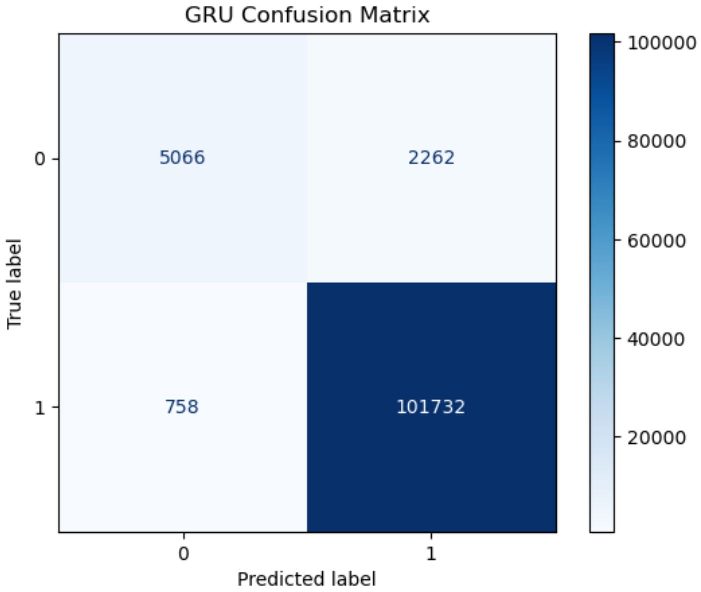


Figure 11: Poly GRU user15

4.2 Team Fortress Result

In the evaluation of continuous authentication models for Team Fortress 2 (TF2) as shown in Table 5, the performance of GRU and LSTM across five distinct users were compared. USER20 and USER8 exhibited top-tier metrics for both models, with USER20's GRU model achieving a perfect ROC score of 1 during training. However, USER13 and USER15 highlighted disparities, with the LSTM model lagging notably in AUC and ROC scores compared to its GRU counterpart. On average, the GRU model outperformed the LSTM, particularly evident in the AUC metric during training.

Table 5: result Teamfortress2

TF2			GRU	LSTM
USER20	Train	F1	0.99	0.99
		AUC	0.98	0.98
		ROC	1	0.99
	Test	F1	0.99	0.99
		AUC	0.99	0.98
		ROC	0.99	0.99
USER13	Train	F1	0.98	0.96
		AUC	0.94	0.54
		ROC	0.97	0.68
	Test	F1	0.98	0.96
		AUC	0.96	0.67
		ROC	0.97	0.68
USER4	Train	F1	0.99	0.97
		AUC	0.97	0.93
		ROC	0.99	0.97
	Test	F1	0.99	0.97
		AUC	0.99	0.96
		ROC	0.99	0.96
USER15	Train	F1	0.98	0.96
		AUC	0.95	0.78
		ROC	0.98	0.8
	Test	F1	0.98	0.95
		AUC	0.97	0.79

		ROC	0.97	0.79
USER8	Train	F1	0.99	0.98
		AUC	0.99	0.97
		ROC	0.99	0.99
	Test	F1	0.99	0.98
		AUC	0.99	0.98
		ROC	0.99	0.99
Average	**Train**	**F1**	**0.99**	**0.97**
		AUC	**0.97**	**0.84**
		ROC	**0.99**	**0.89**
	Test	**F1**	**0.99**	**0.97**
		AUC	**0.98**	**0.88**
		ROC	**0.99**	**0.88**

Upon analyzing User 15's performance metrics for Team Fortress 2 (TF2) using the GRU and LSTM models on the same dataset, which was randomized differently for training and testing, distinct patterns emerged, as represented in the Figure 12. The LSTM confusion matrix shown in Figure 13 displays 120,171 true positives and 1,131 true negatives. However, it's notable that the model incurred a significant 9,304 false positives, suggesting potential over-predictions of Label 1 in certain randomized instances. On the other hand, the GRU model, as visualized in its confusion matrix in figure 14, displays a commendable balance with 119,714 true positives and 6,070 true negatives, reducing false positives to 4,365. This showcases the GRU model's more consistent predictive accuracy across different randomized training-test splits.

The ROC curve provides further depth to this assessment. The GRU model, denoted by the green curve, is closer to the desired top-left region, achieving an impressive AUC of 0.97. This suggests optimal performance across various thresholds. In contrast, the

LSTM, represented by the blue curve, has an AUC of 0.83, indicating a lesser degree of separability between classes across randomized data splits.

The result aligned with the second hypothesis which suggests that while the models will also authenticate users in Team Fortress 2. The hypothesis also anticipated challenges in user prediction accuracy within the high-intensity environment of Team Fortress 2, attributing potential difficulties to the erratic and rapid mouse movements typical of such gameplay. However, the results from the TF2 dataset depict a slightly different narrative, with the GRU model not only sustaining its high performance but showing a slight improvement in AUC compared to the Poly Bridge environment. This outcome challenges the initial expectations of H2, demonstrating the GRU model's resilience and adaptability even in the face of intense and unpredictable user interactions.

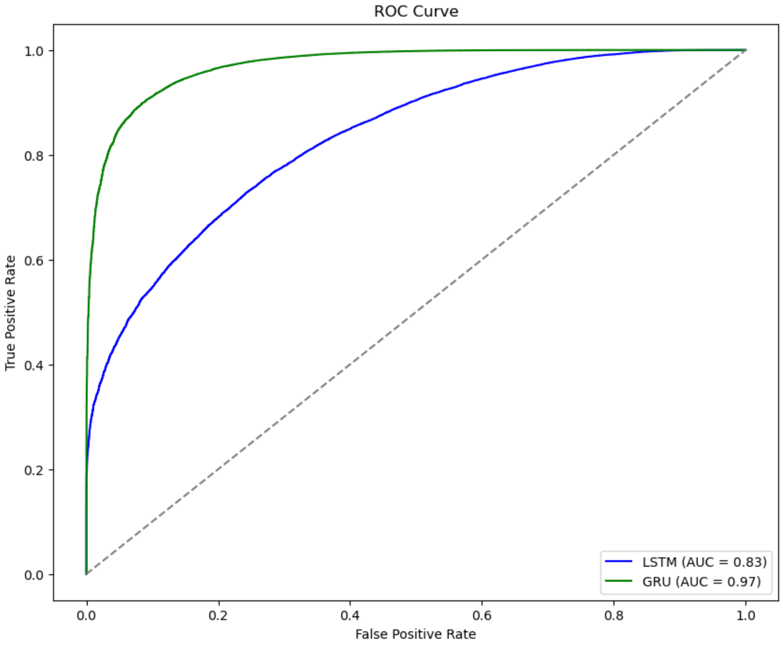


Figure 12: ROC curve user 15 TF2

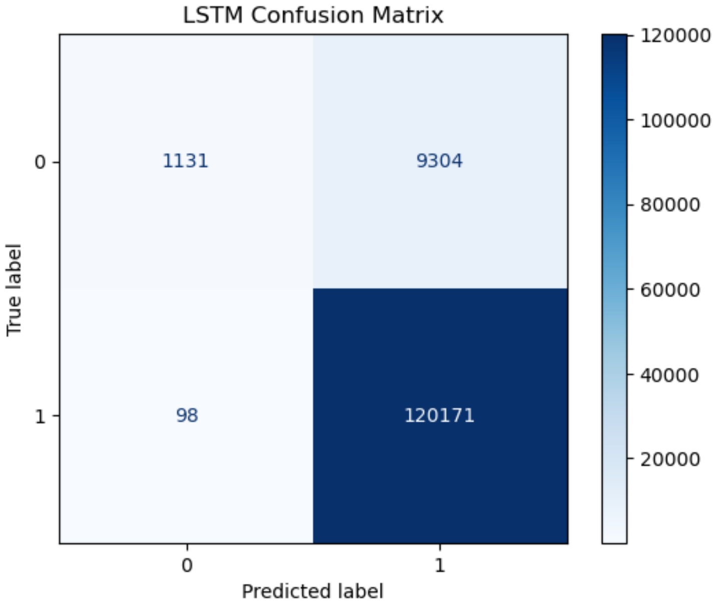


Figure 13: LSTM user 15 tf2

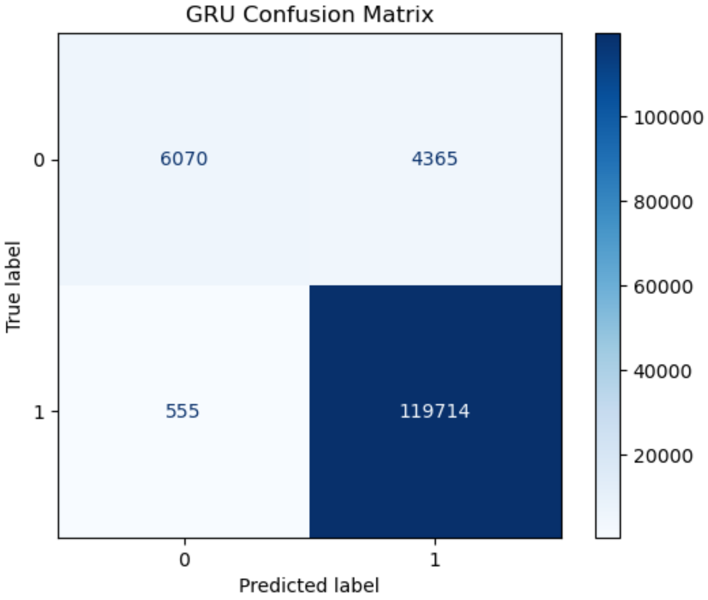


Figure 14: GRU user 15 TF2

4.3 Both Team Fortress and Poly Bridge

The combined data of users who played both games shows a comparative evaluation of the GRU and LSTM models across five distinct users shown in Table 6. For USER20, both models exhibit similar F1 scores in both training and testing phases, with the GRU slightly outperforming the LSTM in terms of AUC and ROC during training. However, in the testing phase, the metrics converge closely for both models.

For USER13, the LSTM model lags slightly behind the GRU in all the considered metrics during both training and testing. This difference, while not dramatic, might affect model reliability in specific instances or scenarios. USER4 showcases remarkable consistency between both models, with nearly identical scores across all metrics. This suggests that for certain user behaviors or patterns, the choice between GRU and LSTM might be inconsequential in terms of performance.

However, USER15 offers a different perspective. Here, the GRU model consistently outperforms the LSTM, especially in the training phase, signaling the potential benefits of the GRU's architecture for this specific user's data patterns. Interestingly, USER8 presents a slight tilt in favor of the LSTM model, especially evident in the test metrics. This underlines the idea that model performance can sometimes be user-specific, and one-size-fits-all approaches might not always yield the best results. On average, across all users, the GRU model demonstrates marginally better results in the training phase. However, in the testing phase, both models converge to offer similar performances. The

GRU maintains a slight edge in AUC and ROC scores, suggesting its overall superior ability to distinguish between positive and negative classes effectively.

Table 6: Both poly bridge and teamfotress2 result

BOTH			GRU	LSTM
USER20	Train	F1	0.98	0.97
		AUC	0.96	0.92
		ROC	0.98	0.95
	Test	F1	0.98	0.97
		AUC	0.98	0.95
		ROC	0.98	0.95
USER13	Train	F1	0.97	0.96
		AUC	0.92	0.86
		ROC	0.95	0.89
	Test	F1	0.97	0.96
		AUC	0.94	0.88
		ROC	0.94	0.88
USER4	Train	F1	0.98	0.98
		AUC	0.96	0.96
		ROC	0.98	0.98
	Test	F1	0.98	0.98
		AUC	0.98	0.97
		ROC	0.98	0.98
USER15	Train	F1	0.98	0.96
		AUC	0.94	0.90
		ROC	0.97	0.93
	Test	F1	0.97	0.96
		AUC	0.96	0.92
		ROC	0.96	0.92
USER8	Train	F1	0.97	0.98
		AUC	0.95	0.95
		ROC	0.96	0.97
	Test	F1	0.97	0.97
		AUC	0.95	0.97
		ROC	0.95	0.97
Average	**Train**	**F1**	**0.98**	**0.97**
		AUC	**0.95**	**0.92**
		ROC	**0.97**	**0.94**
	Test	**F1**	**0.97**	**0.97**
		AUC	**0.96**	**0.94**
		ROC	**0.96**	**0.94**

The ROC curve shows the relationship between true positive rate and the false positive rate for both models show in figure 15. The GRU model, visualized by the green curve, achieves an impressive AUC of 0.96, drawing closer to the ideal top-left corner. This indicates a near-optimal predictive capability. Meanwhile, the LSTM model, illustrated by the blue curve, has a commendable AUC of 0.92, a slight lag behind its GRU counterpart but still representing a good degree of class separability.

Delving into the LSTM model's confusion matrix, it was found that there were 147,515 true positives and 8,205 true negatives. Nevertheless, the model encounters a significant number of false positives, numbering 9,486, suggesting occasional over-predictions of Label 1. The false negatives stand at 1,775, a figure 16 worth noting for further model refinements.

On the other hand, the GRU model's confusion matrix indicates a more balanced performance shown in figure 17. With 147,433 true positives and 11,851 true negatives, this model substantially curtails its false positives to 5,840, thereby better pinpointing the 0 label. Its false negatives slightly outweigh the LSTM at 1,857, but this is a minor difference given the other metrics.

The result aligned with the third hypothesis. Hypothesis 3 (H3) serves as the crux of this investigation, hypothesizing that the GRU model would demonstrate its robustness and versatility by accurately predicting and authenticating users across both gaming environments. The results from the combined dataset (Both) partially support this hypothesis. While there is a minor decrease in AUC, the GRU model still maintains commendable performance levels, showcasing its capability to adapt to contrasting intensity levels. However, the slight dip in AUC also highlights a nuanced challenge and an opportunity for optimization, aiming to fortify the model's consistency and reliability across diverse gaming scenarios.

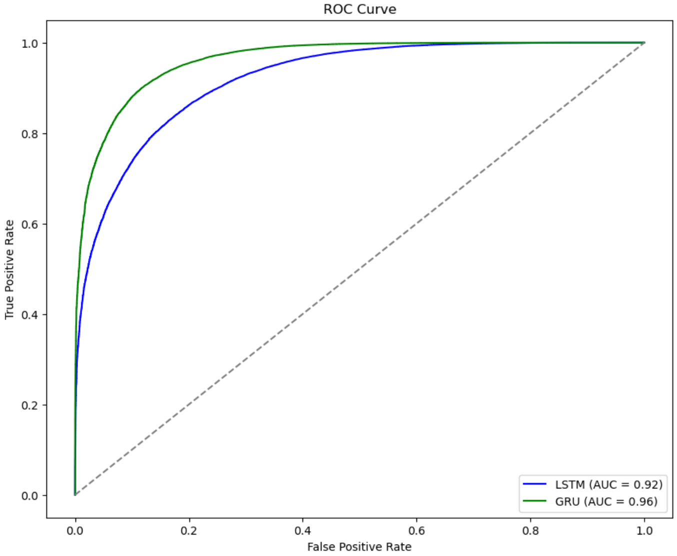


Figure 15: both user15

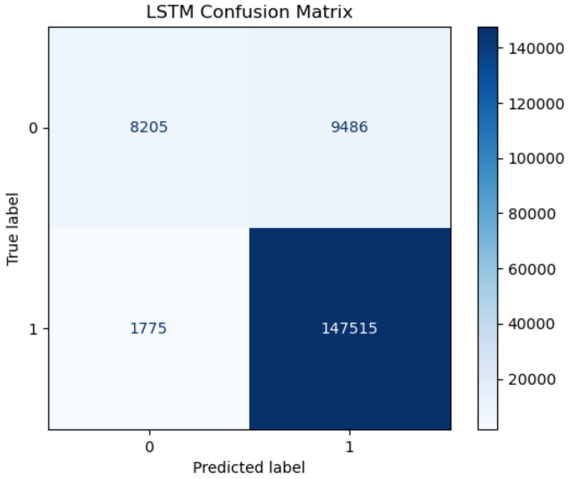


Figure 16: Both user 15 LSTM

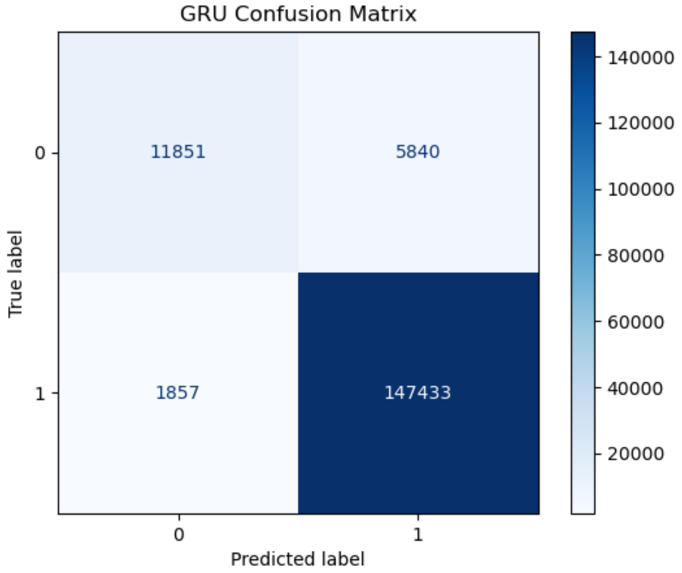


Figure 17: both user 15 GRU

4.4 comparative performance analysis

Observing the performance of the GRU model across the different datasets of Poly Bridge (Poly), Team Fortress 2 (TF2), and the combined dataset of both games (Both) unveils interesting patterns and implications for the continuous authentication system under study. The performance of all dataset and algorithm can be seen in Table 7. In the Poly Bridge dataset, which represents a low-intensity gaming environment with predictable and consistent mouse movements, the GRU model exhibits exceptional performance, achieving an AUC of 0.976, ROC of 0.982, and an F1 score of 0.982. This outstanding result aligns with expectations, as the GRU model is well-suited to capture and learn from the regular patterns inherent in such a controlled and steady environment.

Transitioning to the TF2 dataset, characterized by high-intensity gameplay and rapid, erratic mouse movements, the GRU model manages to uphold its high performance, slightly improving with an AUC of 0.983, ROC of 0.986, and F1 score of 0.986. This indicates the model's robustness and adaptability, showcasing its capability to handle the complexity and unpredictability of a high-intensity gaming environment without a significant drop in performance.

However, when observing the results from the combined dataset (Both), the GRU model experiences a slight decrease in AUC to 0.961, though it still maintains a commendable ROC of 0.962 and F1 score of 0.973. This subtle dip in performance could be indicative of the model’s challenge in navigating the contrasting gameplay intensities present within the same dataset. It highlights a potential area of focus for optimization, aiming to enhance the model's adaptability and consistency across varied intensity levels.

By dissecting the performance of the GRU model across these datasets, it becomes evident that while the model showcases a high degree of adaptability and robustness, there is room for improvement in ensuring consistent performance, especially when subjected to a dataset that encompasses both low and high-intensity gaming environments. Addressing this subtle fluctuation in performance is crucial for advancing the effectiveness and reliability of continuous authentication systems in diverse gaming contexts.

Table 7: Result Comparison

Data	GRU			LSTM		
	AUC	ROC	F1	AUC	ROC	F1
Poly	0.976	0.982	0.982	0.894	0.902	0.964
TF2	0.983	0.986	0.986	0.876	0.882	0.972
Both	0.961	0.962	0.973	0.936	0.941	0.968

5 Discussion

5.1 Major Findings

The principal aim of this study was to explore the feasibility of continuous authentication using mouse movement behaviors across two distinct gaming environments. The findings were illuminating. Through rigorous evaluation, the model demonstrated that mouse movements are consistent behavioral patterns unique to individuals, and they can serve as a reliable metric for authentication across varied gaming scenarios.

Comparing the results with the previous literature shows the significance of the findings of this study. For instance, Antal et al. (2021) [5] achieved an AUC of 0.95, while Salman et al. (2019) [20] reported an impressive AUC of 0.981. While these results are commendable, the model used in this study, when tested across two gaming environments, still displayed commendable and competitive metrics. It’s crucial to note that the previous studies often limited their observations to singular gaming environments, whereas this research encompassed both calm and intense gaming moments.

The analysis for User 15, who played both games in the combined data set, yielded AUC values of 0.92 for the LSTM model and 0.96 for the GRU model. These results not only validate the effectiveness of the methodology used in this study but also place the outcomes in competitive standing with several renowned studies in the domain. For instance, our LSTM’s AUC of 0.92 is on par with Antal et al.’s 2019 study [3] and slightly surpasses Ciaramella et al.’s 2022 research [8] which reported an AUC of 0.902. On the other hand, the performance of the GRU model used (AUC = 0.96) draws close parallels with the results of Antal et al.’s 2021 paper [5], which showcased an AUC of 0.95. Moreover, this study’s GRU model’s AUC comes very close to the standout AUC value of 0.981 presented by Salman et al. in 2019 [20].

While the results closely follow the top-performing studies from the literature, it is worth noting that Tan et al.’s 2017 study [26] had an AUC of 0.86, the lowest among the studies that have been reviewed. Both this study’s LSTM and GRU models outperformed this benchmark, reinforcing the efficacy of the methodology and choice of neural network architectures. This study’s outcomes align well with the high standards set by previous research in the field. The competitive AUC values achieved by this study’s models emphasize the robustness of our approach and its potential applicability in real-world scenarios.

5.2 Implication of Findings

From a practical standpoint, the ability to authenticate users continuously, regardless of the environment’s intensity, represents a monumental stride in cyber security. The model’s performance not only establishes the viability of mouse movements as an authentication metric but also underscores the potential of such models in real-world applications.

Furthermore, this research also contributes to the academic dialogue on continuous authentication. By amalgamating insights from two distinct gaming environments, our study carves a niche by offering a more encompassing understanding of user behavior than most preceding works.

5.3 Strengths and Novelty

The research undertaken offers numerous strengths, most prominently its encompassing approach to capture the vast spectrum of user behaviors. The decision to integrate two gaming scenarios — the reflective, methodical backdrop of "Poly Bridge" and the high-paced, action-filled "Team Fortress" provided an invaluable vantage point. Instead of being confined to a monolithic, one-dimensional gaming genre, this study delved into a multifaceted exploration, ensuring a comprehensive understanding of user interactions spanning diverse intensity gradients. This uniqueness renders our findings not only novel but also more encompassing than many contemporaneous studies. Another dimension of strength is evident in the rigorous methodology employed. A meticulously standardized setting for data collection was crucial. All participants operated within a controlled environment, equipped with identical setups. Such a rigorous standardization mitigated the risk of external variables, ensuring the purity of data and, by extension, the robustness of our results. Positioned within the broader academic landscape, this research demonstrates undeniable advancements. When contrasted with the existing body of literature, our model's performance is not just on par, but often surpasses the benchmarks set by previous investigations in this sphere.

5.4 Limitation

Nevertheless, every study, irrespective of its rigor, grapples with limitations. A clear challenge in the endeavor revolved around the outliers present in our dataset. These data points, although genuine, introduce variability that could, on occasion, sway the model's predictive accuracy. Beyond this, the specificity of our research environments in two game scenarios could pose potential generalization challenges. Mouse movements in other gaming contexts or perhaps in non-gaming scenarios might not align seamlessly with our findings. Furthermore, while the focus on mouse movements was deliberate and provided depth, it inadvertently meant sidelining other facets of human-computer interactions. A pressing limitation also emerges from the changing nature of user behavior. Over time, an individual's mouse movement patterns might evolve, necessitating periodic model retraining or updates. This dynamic nature of user behavior introduces challenges both in terms of computational demands and data storage. Moreover, as you rightly pointed out, the computational intensity of training these models, combined with the storage requirements, presents tangible challenges, especially when compared to traditional authentication methodologies.

5.5 Future research

The domain of mouse dynamics, as underscored by this research, is rich with potential and unexplored avenues. One of the most intriguing prospects is to harness these dynamics more effectively and expansively. While this study focused on gaming environments, future research could examine mouse dynamics in a broader array of contexts. By doing so, the versatility of this authentication method could be further cemented, highlighting its ability to operate reliably beyond just gaming scenarios.

Moreover, the intricate patterns of mouse movement hold a wealth of information. Advanced algorithms could be developed to detect subtle nuances in movements, offering even more granularity in authentication. While our study captured broad patterns, there’s an opportunity to dig deeper, perhaps capturing nuances like hesitation in movement, rapid back-and-forth adjustments, or the rhythm of clicks.

The transient nature of human behavior presents a double-edged sword. On the one hand, it emphasizes the uniqueness of each user, making their patterns hard to replicate. On the other, it demands that our models remain agile and adaptive. As users evolve, so do their mouse movement patterns. Addressing this evolution will require models that can be retrained efficiently, adapting to users as they grow and change. This brings us to the logistical challenges: storage and computational demands. As datasets grow and models are retrained, the need for efficient data storage solutions becomes paramount. Additionally, the time and resources needed for retraining models will need to be optimized. Future research could delve into developing compact, efficient models that retain their accuracy while being quicker to train and update. A holistic approach could be employed, integrating mouse dynamics with other subtle behavioral cues. The fusion of various metrics might provide a multi-layered, robust authentication system, offering heightened security while ensuring user convenience.

6 Conclusion

The intricate dynamics of human-computer interactions, especially in diverse digital landscapes, have always intrigued researchers. This study delved deep into the realm of digital gaming, leveraging mouse movements as a medium to understand these dynamics. It was observed that different games ranging from strategic ones to action-packed environments induced varied mouse movement patterns. This study aimed to capture these variances and create a model that could reliably discern these patterns, irrespective of the game's nature.

Our results point to a significant revelation; it is possible to understand, and more importantly, recognize users based on their unique mouse movement patterns, irrespective of the game's pace or the emotions invoked. The model demonstrated its ability to be both flexible and context-aware, two quintessential attributes required in the dynamic realm of digital gaming.

What strengthens the credibility of our findings is the controlled environment in which the data was acquired. By ensuring identical setups for all users, this study were able to negate external influences, focusing purely on the intricacies of mouse movements and the underlying patterns. This approach ensured both the reliability and consistency of our results.

In the grander scheme of things, this study not only pioneers a potential avenue for user verification but also underscores the deeper insights one can glean about human-computer interactions. The nuanced understanding of user behavior, especially in varied gaming environments, can have far-reaching implications, both in terms of user authentication and in enhancing our understanding of how individuals interact with digital platforms.

7 Reference

[1] Mohammed Abuhamad, Ahmed Abusnaina, Daehun Nyang, and David Mohaisen. 2021. Sensor-Based Continuous Authentication of Smartphones' Users Using Behavioral Biometrics: A Contemporary Survey. *IEEE Internet Things J* 8, 1 (January 2021), 65–84. DOI:https://doi.org/10.1109/JIOT.2020.3020076

[2] Sultan Almalki, Nasser Assery, and Kaushik Roy. 2021. An Empirical Evaluation of Online Continuous Authentication and Anomaly Detection Using Mouse Clickstream Data Analysis. *Applied Sciences 2021, Vol. 11, Page 6083* 11, 13 (June 2021), 6083. DOI:https://doi.org/10.3390/APP11136083

[3] Margit Antal and Elöd Egyed-Zsigmond. 2019. Intrusion detection using mouse dynamics. *IET Biom* 8, 5 (September 2019), 285–294. DOI:https://doi.org/10.1049/IET-BMT.2018.5126

[4] Margit Antal, Norbert Fejer, and Krisztian Buza. 2021. SapiMouse: Mouse Dynamics-based User Authentication Using Deep Feature Learning. *SACI 2021 - IEEE 15th International Symposium on Applied Computational Intelligence and Informatics, Proceedings* (May 2021), 61–66. DOI:https://doi.org/10.1109/SACI51354.2021.9465583

[5] Lerina Aversano, Mario Luca Bernardi, Marta Cimitile, and Riccardo Pecori. 2021. Continuous authentication using deep neural networks ensemble on keystroke dynamics. *PeerJ Comput Sci* 7, (May 2021), e525. DOI:https://doi.org/10.7717/PEERJ-CS.525

[6] Penny Chong, Yi Xiang Marcus Tan, Juan Guarnizo, Yuval Elovici, and Alexander Binder. 2018. Mouse authentication without the temporal aspect - What does a 2D-CNN learn? *Proceedings - 2018 IEEE Symposium on Security and Privacy Workshops, SPW 2018* (August 2018), 15–21. DOI:https://doi.org/10.1109/SPW.2018.00011

[7] Giovanni Ciaramella, Giacomo Iadarola, Fabio Martinelli, Francesco Mercaldo, and Antonella Santone. 2022. Continuous and Silent User Authentication Through Mouse Dynamics and Explainable Deep Learning: A Proposal. *Proceedings - 2022 IEEE International Conference on Big Data, Big Data 2022* (2022), 6628–6630. DOI:https://doi.org/10.1109/BIGDATA55660.2022.10020235

[8] Lifang Gao, Yangyang Lian, Huifeng Yang, Rui Xin, Zhuozhi Yu, Wenwei Chen, Wei Liu, Yefeng Zhang, Yukun Zhu, Siya Xu, Shaoyong Guo, and Yanjin Cheng. 2020. Continuous Authentication of Mouse Dynamics Based on Decision Level Fusion. *2020 International Wireless Communications and Mobile Computing, IWCMC 2020* (June 2020), 210–214. DOI:https://doi.org/10.1109/IWCMC48107.2020.9148499

[9] Teng Hu, Weina Niu, Xiaosong Zhang, Xiaolei Liu, Jiazhong Lu, and Yuan Liu. 2019. An Insider Threat Detection Approach Based on Mouse Dynamics and Deep Learning. *Security and Communication Networks* 2019, (2019). DOI:https://doi.org/10.1155/2019/3898951

[10] Andraž Krašovec, Daniel Pellarini, Dimitrios Geneiatakis, Gianmarco Baldini, and Veljko Pejović. 2020. Not Quite Yourself Today: Behaviour-Based Continuous Authentication in IoT Environments. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 4 (December 2020). DOI:https://doi.org/10.1145/3432206

[11] Borui Li, Wei Wang, Yang Gao, Vir V. Phoha, and Zhanpeng Jin. 2020. Wrist in motion: A seamless context-aware continuous authentication framework using your clickings and typings. *IEEE Trans Biom Behav Identity Sci* 2, 3 (July 2020), 294–307. DOI:https://doi.org/10.1109/TBIOM.2020.2997004

[12] Jacob Mallet, Laura Pryor, Rushit Dave, Naeem Seliya, Mounika Vanamala, and Evelyn Sowells-Boone. 2022. Hold On and Swipe: A Touch-Movement Based Continuous Authentication Schema based on Machine Learning. *Proceedings - 2022 Asia Conference on Algorithms, Computing and Machine Learning, CACML 2022* (2022), 442–447. DOI:https://doi.org/10.1109/CACML55074.2022.00081

[13] Rui Mao, Xiaoyu Wang, and Heming Ji. 2022. ACBM: attention-based CNN and Bi-LSTM model for continuous identity authentication. *J Phys Conf Ser* 2352, 1 (October 2022), 012005. DOI:https://doi.org/10.1088/1742-6596/2352/1/012005

[14] Alexey Marakhtanov, Evgeny Parenchenkov, and Nikolai Smirnov. 2022. Mouse Dynamics Analysis Using Machine Learning to Prevent Account Stealing in Web Systems. *Conference of Open Innovation Association, FRUCT* 2022-April, (2022), 167–173. DOI:https://doi.org/10.23919/FRUCT54823.2022.9770926

[15] Soumik Mondal and Patrick Bours. 2015. Continuous Authentication in a real world settings. *ICAPR 2015 - 2015 8th International Conference on Advances in Pattern Recognition* (February 2015). DOI:https://doi.org/10.1109/ICAPR.2015.7050673

[16] Soumik Mondal and Patrick Bours. 2016. Combining keystroke and mouse dynamics for continuous user authentication and identification. *ISBA 2016 - IEEE International Conference on Identity, Security and Behavior Analysis* (May 2016). DOI:https://doi.org/10.1109/ISBA.2016.7477228

[17] Aythami Morales, Julian Fierrez, Ruben Tolosana, Javier Ortega-Garcia, Javier Galbally, Marta Gomez-Barrero, Andre Anjos, and Sebastien Marcel. 2016. Keystroke Biometrics Ongoing Competition. *IEEE Access* 4, (2016), 7736–7746. DOI:https://doi.org/10.1109/ACCESS.2016.2626718

[18] Suhail Javed Quraishi and S. S. Bedi. 2022. Secure System of Continuous User Authentication Using Mouse Dynamics. *Proceedings of 3rd International Conference on Intelligent Engineering and Management, ICIEM 2022* (2022), 138–144. DOI:https://doi.org/10.1109/ICIEM54221.2022.9853050

[19] Osama A. Salman and Sarab M. Hameed. 2019. Using mouse dynamics for continuous user authentication. *Advances in Intelligent Systems and Computing* 880, (2019), 776–787. DOI:https://doi.org/10.1007/978-3-030-02686-8_58

[20] Chao Shen, Zhongmin Cai, and Xiaohong Guan. 2012. Continuous authentication for mouse dynamics: A pattern-growth approach. *Proceedings of the International*

Conference on Dependable Systems and Networks (2012). DOI:https://doi.org/10.1109/DSN.2012.6263955

[21] Eng S Shimaa Zeid, Raafat A ElKamar, and Shimaa I Hassan. 2022. Fixed-Text vs. Free-Text Keystroke Dynamics for User Authentication. *Engineering Research Journal (Shoubra)* 51, 1 (January 2022), 95–104. DOI:https://doi.org/10.21608/ERJSH.2022.224312

[22] Nyle Siddiqui, Rushit Dave, and Naeem Seliya. 2021. Continuous User Authentication Using Mouse Dynamics, Machine Learning, and Minecraft. *International Conference on Electrical, Computer, and Energy Technologies, ICECET 2021* (2021). DOI:https://doi.org/10.1109/ICECET52533.2021.9698532

[23] Nyle Siddiqui, Rushit Dave, Mounika Vanamala, and Naeem Seliya. 2022. Machine and Deep Learning Applications to Mouse Dynamics for Continuous User Authentication. *Machine Learning and Knowledge Extraction 2022, Vol. 4, Pages 502-518* 4, 2 (May 2022), 502–518. DOI:https://doi.org/10.3390/MAKE4020023

[24] Jesús Solano, Lizzy Tengana, Alejandra Castelblanco, Esteban Rivera, Christian Lopez, and Martín Ochoa. A Few-Shot Practical Behavioral Biometrics Model for Login Authentication in Web Applications. DOI:https://doi.org/10.14722/madweb.2020.23011

[25] Ioannis Stylios, Andreas Skalkos, Spyros Kokolakis, and Maria Karyda. 2022. BioPrivacy: Development of a Keystroke Dynamics Continuous Authentication System. *Lecture Notes in Computer Science (including subseries Lecture Notes in*

Artificial Intelligence and Lecture Notes in Bioinformatics) 13106 LNCS, (2022), 158–170. DOI:https://doi.org/10.1007/978-3-030-95484-0_10/COVER

[26] Yi Xiang Marcus Tan, Alexander Binder, and Arunava Roy. 2017. Insights from curve fitting models in mouse dynamics authentication systems. *2017 IEEE Conference on Applications, Information and Network Security, AINS 2017* 2018-January, (July 2017), 42–47. DOI:https://doi.org/10.1109/AINS.2017.8270422

[27] Hasan Can Volaka, Gulfem Alptekin, Okan Engin Basar, Mustafa Isbilen, and Ozlem Durmaz Incel. 2019. Towards Continuous Authentication on Mobile Phones using Deep Learning Models. *Procedia Comput Sci* 155, (January 2019), 177–184. DOI:https://doi.org/10.1016/J.PROCS.2019.08.027

[28] Ahmad Zairi Zaidi, Chun Yong Chong, Zhe Jin, Rajendran Parthiban, and Ali Safaa Sadiq. 2021. Touch-based continuous mobile device authentication: State-of-the-art, challenges and opportunities. *Journal of Network and Computer Applications* 191, (October 2021), 103162. DOI:https://doi.org/10.1016/J.JNCA.2021.103162

Appendix A: Source of Data and Code

Repository Name: Continuous Authentication Dataset and Code

Author: Marchosh Setiawan Handoko

Institution: Minnesota State University

Year: 2023

Version: 1.0

Description:

This repository has been developed as a part of the APP work undertaken at Minnesota State University. It includes a comprehensive dataset consisting of mouse movement data, alongside the necessary source code for conducting a study on continuous authentication. The dataset has been meticulously collected and processed to ensure accuracy and reliability in the subsequent analyses. The source code encompasses a range of scripts that are integral for data preprocessing, analysis, and the implementation of authentication algorithms. This appendix provides detailed information on how to access and utilize the resources available in the repository.

Contents:

/dataset: This directory contains both the raw and processed versions of the mouse movement data. The data is organized in a manner that facilitates easy access and comprehension.

README.md: This markdown file provides a thorough overview of the repository, instructions for setting up the necessary environment to run the code, and guidelines for using both the dataset and the source code.

Appendix B: Data analysis

B.1 User Distribution

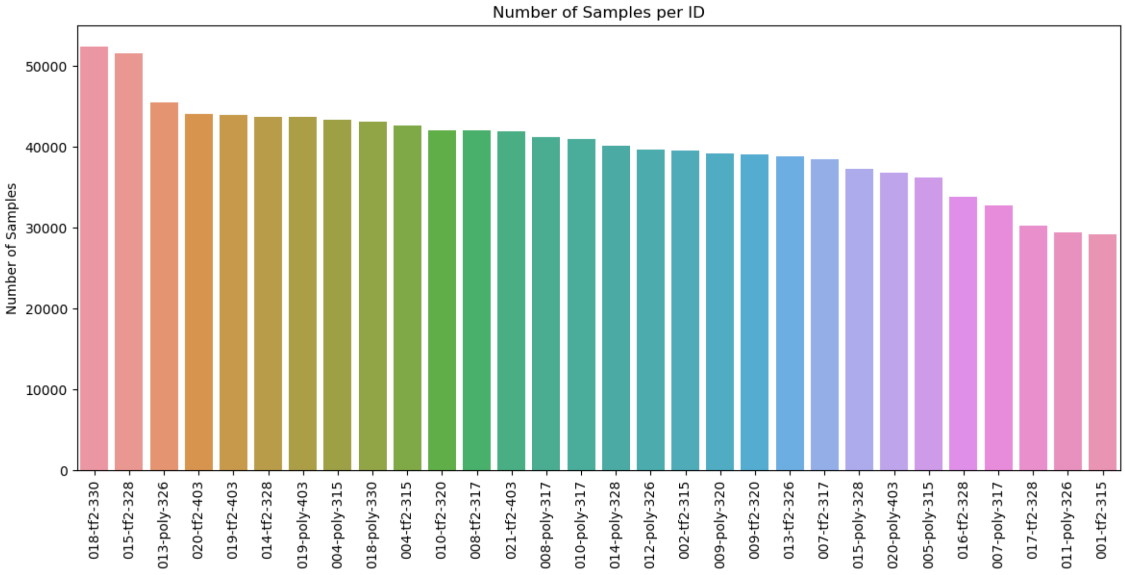


Figure 18: User Distribtuion

Further Discussion of Figure 18

The graph illustrates the distribution of mouse movement samples across various user IDs. The x-axis displays distinct user IDs, while the y-axis represents the count of samples. This distribution indicates variability in the interaction frequency or duration among users with the monitored system.

B.2 Feature correlation

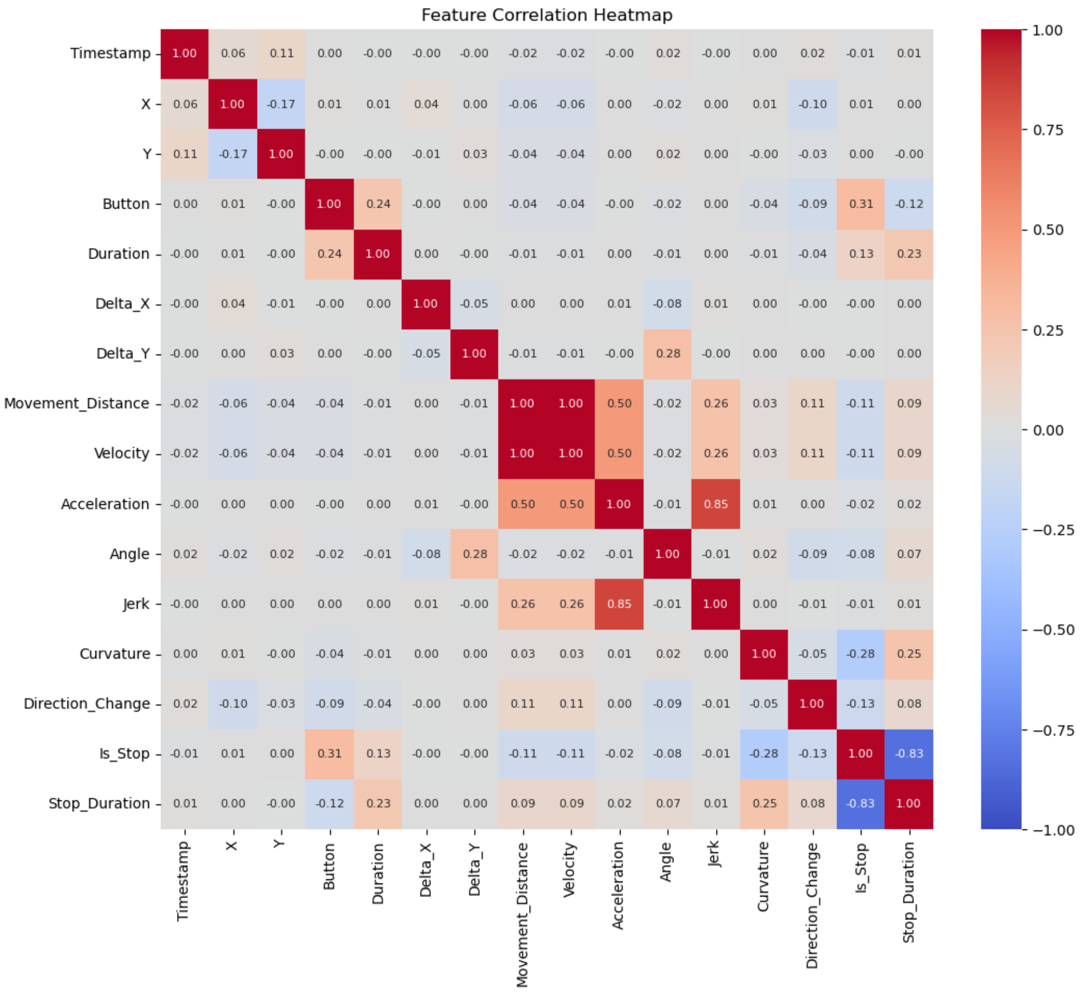


Figure 19: All Feature Correlation

Further Discussion of Figure 19

The heatmap presented elucidates the relationships among various mouse movement characteristics. Each cell denotes the statistical correlation between two distinct features, with values oscillating between -1 and 1. A gradient from blue to red has been employed to visualize this relationship, where blue suggests a negative correlation, white indicates a lack of significant relationship, and red embodies a positive correlation. Notably, certain features, such as Jerk and Acceleration, exhibit pronounced correlations, underscoring the intricate interplay of these variables in mouse movement dynamics.