Journal of Theoretical and Applied Information Technology
15th July 2023. Vol.101. No 13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ENHANCING USER-LEVEL SECURITY: PERFORMANCE ANALYSIS OF MACHINE LEARNING ALGORITHMS FOR DYNAMIC KEYSTROKE ANALYSIS

B RADHA KRISHNA[1], Dr M SRIHARI VARMA[2]

[1]M. Tech Student, JNTUK SRKREC, Department of Computer Science, Bhimavaram, AP, India

[2]Associate Professor, JNTUK SRKREC, Department of Computer Science, Bhimavaram, AP, India

E-mail: [1]radhakrishna.rk96@gmail.com

ABSTRACT

In today's computing and information-processing industry, security has become a paramount concern. With the proliferation of internet-based devices and the exponential growth of data transmission, the threat of malicious activities has also escalated. One prevalent form of attack involves the theft of passwords and unauthorized access to sensitive information, resulting in data loss and diminished user satisfaction. To combat this issue, it is crucial to detect and prevent fraudulent logins. This paper presents a comprehensive performance analysis of machine learning algorithms for dynamic keystroke analysis, aimed at enhancing user-level security. The focus is on developing a robust method that ensures tight security while maintaining high accuracy. Previous research efforts have proposed various security mechanisms, techniques, and algorithms; however, their efficiency has been found lacking. The proposed approach leverages machine learning algorithms, specifically the KNN and XGB models, to analyze dynamic keystroke patterns. By extracting and analyzing important parameters related to keystroke dynamics, the algorithms aim to identify fraudulent login attempts. The performance of both algorithms is evaluated and compared, with a particular emphasis on accuracy and efficiency. The results of the analysis demonstrate that the XGB model outperforms the KNN algorithm in terms of security enhancement and user satisfaction. The XGB model leverages the dataset effectively, utilizing key parameters to make accurate predictions and classify keystroke patterns. In contrast, the KNN algorithm falls short in achieving comparable levels of accuracy and efficiency. By employing the superior XGB model, organizations can enhance user-level security, prevent password theft, and safeguard sensitive data. The findings of this study contribute to the development of an effective and reliable security mechanism, ultimately improving user satisfaction and data protection. In conclusion, this research highlights the importance of adopting advanced machine learning algorithms for dynamic keystroke analysis to enhance user-level security. The XGB model emerges as a powerful tool, surpassing the limitations of the KNN algorithm and providing a more robust and accurate solution. By implementing these findings, organizations can bolster their security measures and address the growing challenges associated with data protection and user satisfaction in the digital age.

Keywords: *Dynamic keystroke analysis, Machine learning algorithms, User-level security, Performance analysis, XGB model, KNN algorithm, Fraudulent logins, Password theft, Data protection*

1. INTRODUCTION

The advent of virtual keyboards has revolutionized the way we interact with technology. These virtual keyboards have given rise to various applications and technologies that aim to capture and analyze users' typing patterns for enhanced security and user experience. One such technology is smart rings, which capture finger movements and enable keystroke recognition [1].

Each individual exhibits unique typing patterns, characterized by subtle nuances in the typing process. These patterns can be leveraged to gather extensive data on users' typing behavior, enabling keystroke recognition and user authentication [2]. However, processing and evaluating this data can be a time-consuming and labor-intensive task. Incorporating sensing mechanisms into small form factors, such as smart rings, poses challenges in accurately detecting finger movements [3]. The compact size of these devices limits the available

space for sensors and requires innovative approaches to capture and analyze keystroke data effectively. The emergence of edge computing has played a significant role in addressing these challenges [4]. Edge computing is a paradigm that shifts computation processes to the network edges, closer to the data source. This approach improves response times, reduces congestion in real-time data processing, and enables prompt delivery of results to clients. It eliminates the need for analog-to-digital data conversion and allows for parallel data processing, making it particularly valuable for real-time keystroke analysis [5]. Dynamic keystroke analysis involves analyzing key-down, key-up, and inter-key time interval information to determine whether the typing patterns match those of an authorized user. While this method can serve as a user authentication mechanism, it is not foolproof and can be influenced by external factors like fatigue, stress, or physical injury [6]. These factors can potentially lead to false positives or false negatives, highlighting the need for a comprehensive security approach that combines multiple authentication methods. Machine learning algorithms play a crucial role in optimizing simultaneous data flows and building an effective keystroke analysis mechanism [7]. Clustering algorithms like K-Nearest Neighbors (KNN) and Support Vector Machines (SVM) are commonly used in keystroke recognition. However, the development of lightweight deep learning algorithms has enabled their integration into edge computing networks for efficient data processing. These lightweight algorithms offer higher accuracy while considering the resource constraints of edge computing devices. In this paper, we aim to analyze and compare the performance of machine learning algorithms, specifically KNN and XGB, for dynamic keystroke analysis [8]. We evaluate their accuracy, efficiency, and suitability for enhancing user-level security. Through experimental results and detailed analysis, we demonstrate that the XGB model outperforms the KNN algorithm in terms of accuracy and efficiency [9]. The findings of this study contribute to the advancement of keystroke analysis techniques and provide valuable insights for the development of secure authentication systems [10].

2. LITERATURE REVIEW

In recent years, there has been a notable increase in research focusing on keystroke patterns, particularly in the context of smart keyboards [11]. The implementation of dynamic keystroke analysis in smart keyboards presents a significant challenge. These keyboards utilize mechanical stimuli to convert keyboard patterns into local electronic signals [12]. Intelligent keyboard systems (IKBs) have been developed to generate alarms and trace the dynamic timing between key presses and releases, allowing for the detection and analysis of keystroke patterns [13]. These systems have demonstrated improved efficiency and enhanced user authentication. [15] emphasized the need to strengthen existing authentication systems on mobile devices due to the increasing storage of personal details. They proposed implementing a keystroke dynamic method that analyzes pressure applied to the touch screen. Their experiment, which involved 42 datasets, showed that the proposed model significantly improved the efficiency of the authentication system. [16] proposed an authentication system based on Support Vector Machines (SVM) for keystroke dynamics, aiming to enhance security. Their model achieved an accurate result with a 0% error rate by utilizing an SVM-based one-time password mechanism [17]. The study analyzed 34 records to validate the proposed approach. [18] employed LSTM and GRU algorithms to demonstrate the effectiveness of keystroke dynamics as a biometric technique. This combination of algorithms improved the model's performance, resulting in a reduced error rate. The study highlighted the usefulness of dynamic keystroke analysis in enhancing user authentication. [19] proposed a non-conventional keystroke dynamic technique that employed an SVM model with ant colony optimization (ACO) for feature selection. The results showed reduced false acceptance and reject rates, indicating the model's ability to identify individual typing behavior and improve user authentication in smart keyboards [20]. Compared to other methods, their proposed model achieved a 10% higher accuracy rate with a 7.3% lower error rate. The study also suggested that the CNN model was particularly suitable for dynamic keystroke analysis. [21] conducted a study to classify users based on their typing behavior. They evaluated 94 user behaviors using a machine learning classifier and achieved improved computing device performance, authentication, and accuracy by utilizing an SVM classifier to reduce the error rate and accurately classify features from raw input data. [22] presented a comprehensive survey analyzing the behavior of authentication systems using keystroke dynamics. Their proposed model achieved a false acceptance rate (FAR) of 0.3%, false rejection rate (FRR) of 1.5%, and equal error rate (EER) of 0.9%, outperforming other methods

Journal of Theoretical and Applied Information Technology
15th July 2023. Vol.101. No 13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

and enhancing authentication systems. [23] conducted a study on the key-print signature method within dynamic keystroke analysis. Their results indicated that the proposed key-print signature-based method was more effective compared to the key-print profile method, resulting in improved performance efficiency. However, additional features were required to accurately detect negative cases. [24] conducted an analysis on the usage of keystroke dynamics in smart keyboard systems. They evaluated machine learning and statistical-based techniques, highlighting the strengths and weaknesses of the models. However, further improvements in efficiency and performance are still needed. [25] proposed a unique keypad technique to enhance the efficiency of smartphone authentication systems. The results of their proposed model demonstrated improved filtering rates and equal error rates (EER) compared to existing models. [26] conducted a survey utilizing machine learning methods.

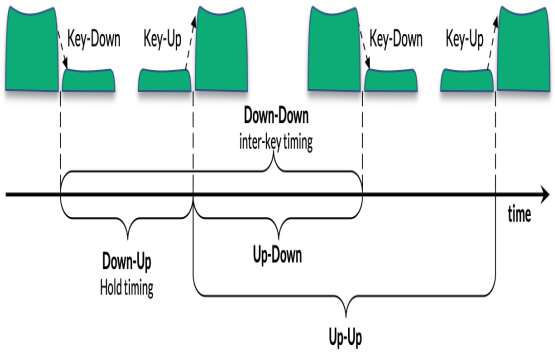

Figure 1. Keystroke Dynamics

Problem Statement

The accurate prediction of keystroke behavior is crucial in various applications, and existing methods primarily rely on linear combinations of timestamp features such as key flight, dwell time, and digraph. However, to achieve a more comprehensive understanding and prediction of keystroke behavior, it is necessary to capture non-linear combinations of these timestamp features. Traditional classifiers often struggle to discern the underlying structure of keystroke data, highlighting the need for a more sophisticated approach that extracts the entire set of features associated with keystrokes to improve prediction accuracy. This research problem aims to address the limitations of existing methods by proposing a novel deep-learning model capable of learning, analyzing, extracting, and classifying keystroke behavior using the available data.

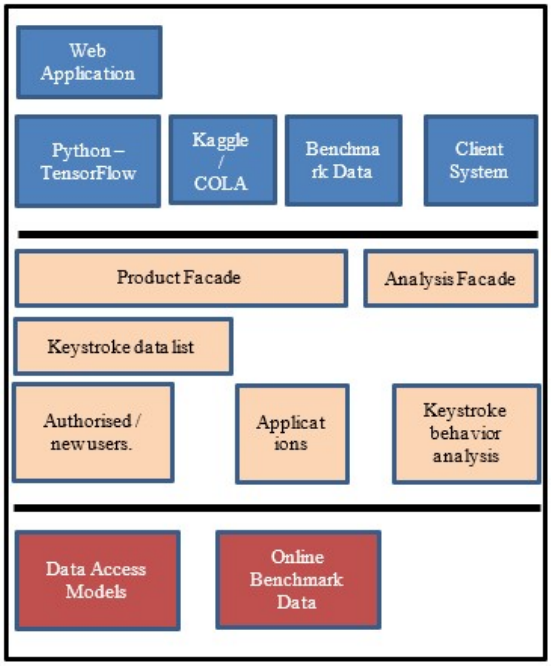

Figure 2. High-Level Architecture: Solution

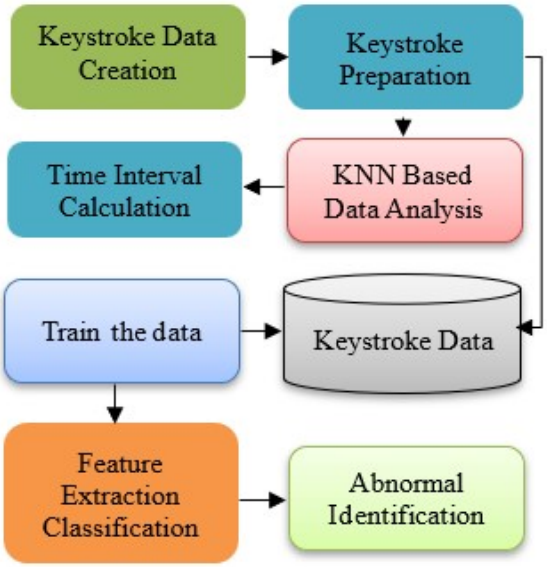

Figure 3. Overall Functional Flow

The primary objective is to design and implement an advanced deep-learning algorithm that surpasses the accuracy achieved by existing algorithms. To evaluate the performance, a comparison between training accuracy and testing accuracy will be conducted. Furthermore, this work will analyze both single-user behavior and multi-user behavior to validate the accuracy of the proposed model. To assess the model's capabilities in analyzing dynamic keystroke behavior, a benchmark dataset will be employed as input data for the deep-learning model.

In this study, the proposed XGBoost (XGB) algorithm will be employed as the deep-learning model, and its performance will be compared to the accuracy achieved by the K-Nearest Neighbors (KNN) algorithm, a widely used baseline algorithm. The objective is to demonstrate that the novel deep-learning approach using XGB surpasses the accuracy obtained by the existing KNN algorithm. To summarize, this research project aims to address the limitations of existing methods by proposing a novel deep-learning model for accurate keystroke behavior classification. The project's goals include surpassing the accuracy of existing algorithms, analyzing both single-user and multi-user behavior, and evaluating the model's performance on a benchmark dataset. The proposed XGB algorithm will serve as the key component in achieving these objectives, with the aim of demonstrating its superiority over the KNN algorithm in terms of accuracy.

3. RELATED WORK

The entire system architecture is divided into several aspects. Initially the high-level system architecture is designed for the project development and given in Figure-1. Initially the high-level architecture helps to understand the overall system processes and the flow [27]. The web application is implemented in Python over TensorFlow and executed in COLAB or Kaggle environment at any client system, anywhere. The function starts from Python (web), receives all the inputs from the application (user), persists, and compares with the dataset in the database [28]. The system architecture for the project development is designed with a high-level approach, as depicted in Figure-2. This high-level architecture provides a comprehensive understanding of the system processes and flow [29]. The web application is implemented using Python with TensorFlow and can be executed in a COLAB or Kaggle environment on any client system, regardless of location. The Python web function serves as the entry point, receiving inputs from the application users and persisting them [30]. These inputs are then compared with the dataset stored in the database. Users interact with the system through a user interface designed for Windows, enabling seamless communication with the server via the client logic module. To ensure secure communication, encryption-based authentication is employed between the client and server. This authentication process takes place before accessing the business module, which encompasses the essential objects and business logic [31]. The application is directly connected to the database through the implementation of the request model. The overall software architecture, as depicted in Figure-2, follows a request and response model between the front end and back end. Users input their data through the front end, and the application processes these inputs through the various n-tier modules [32]. The front end (tier-1), back end (tier-2), and business layer (tier-3) are the three tiers utilized in the application. This architectural design ensures a modular and efficient system where users can seamlessly interact with the application, with their inputs processed and responses obtained through the different tiers. The use of encryption-based authentication and the integration of TensorFlow provide a secure and robust environment for the dynamic keystroke analysis system [33].

4. PROPOSED MODEL

The proposed model, as depicted in Figure-2, showcases the applications of keystroke dynamic analysis, primarily focused on enhancing security and identifying abnormal user behavior. Keyboard users engage in countless typing actions throughout their lives, each characterized by unique patterns. Despite minor differences between users' typing styles, these patterns can be identified and utilized for user detection [34]. Keystroke dynamic analysis provides a non-intrusive means of detecting and analyzing these patterns, even without the user's knowledge. Real-time analysis techniques enable the identification of keystroke patterns, making it a valuable tool in various fields, including keystroke, and typing biometrics. This technique harkens back to the era of processing Morse codes to define programs. Morse codes, consisting of dots and dashes, were used to define program functions [35, 36]. Today, advanced techniques are employed to differentiate keystrokes from each other and detect subtle variations. The proposed model leverages the KNN algorithm and XGB models for analyzing keystrokes and evaluating their performance. By harnessing the power of these algorithms, the proposed model aims to enhance security measures by accurately identifying users based on their unique keystroke patterns [37,38]. The combination of keystroke dynamic analysis and machine learning techniques provides a robust approach for user authentication and anomaly detection.

Journal of Theoretical and Applied Information Technology
15th July 2023. Vol.101. No 13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

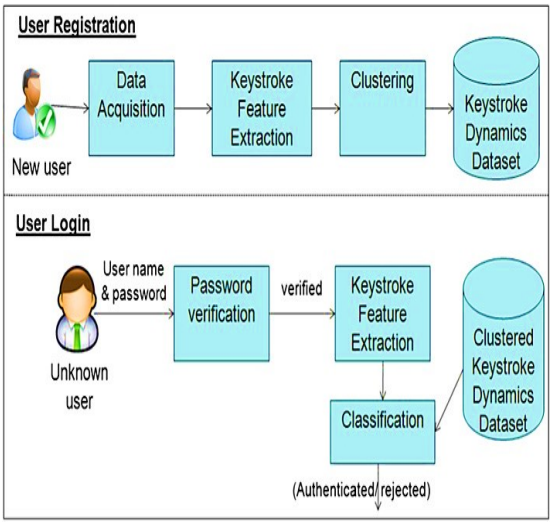

Figure-4. Application Scenario of Keystroke Dynamics

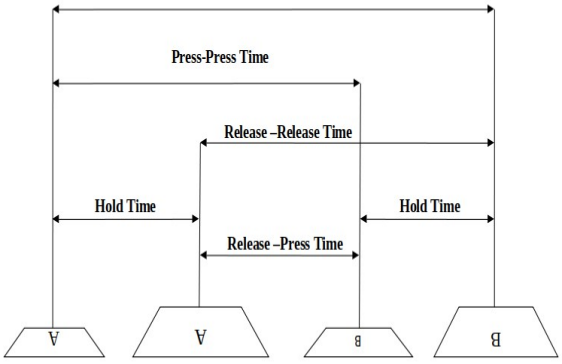

Figure-5. Parameters evaluated from the typing process.

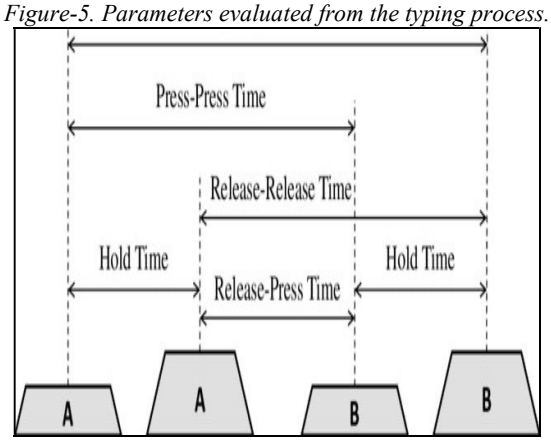

Figure-6. Console Window Starts

The implementation of the proposed model involves several key modules to ensure an effective analysis of the keystroke data and accurate classification of keyboard actions. The following modules outline the step-by-step process:

1. Data Import and EDA Initialization

In this module, the necessary data is imported into the system. The data may include keystroke records and corresponding labels. Exploratory Data Analysis (EDA) techniques are applied to gain insights into the dataset, such as understanding the distribution of features, identifying any outliers, and detecting any missing values or inconsistencies.

2. Feature Engineering

This module focuses on extracting meaningful features from the raw keystroke data. Feature engineering techniques are applied to transform the data into a format suitable for analysis [39,40]. This may involve processing the duration between key presses, key holding time, and other relevant parameters. Additional features may also be derived to capture specific characteristics of the typing behavior.

3. EDA Implementation

Building upon the EDA initialization, this module further explores the dataset using various statistical and visual analysis techniques. It helps in understanding the relationships between different features, identifying patterns, and uncovering any correlations that may exist [41, 42]. The findings from this analysis guide subsequent steps in the classification process.

4. Learn the Data, Extract Features, and Classify

In this module, machine learning algorithms such as KNN and XGB are applied to learn from the extracted features and classify keyboard actions. The model is trained using labeled data to recognize and differentiate between different actions, such as key presses, releases, or new word typing [43, 44]. The algorithms utilize the extracted features to make accurate predictions and classify the keyboard actions effectively.

5. Predict the Keyboard - Action

Once the model is trained, it is ready to predict the keyboard actions based on new input data. The trained model takes the extracted features as input and applies the classification algorithms to predict the corresponding keyboard actions in real-time [45, 46]. This module allows for the identification of specific actions performed by the

user, enabling enhanced user-level security and detection of any abnormal behavior. By systematically executing these modules, the proposed model enables the analysis of keystroke data, extraction of relevant features, and accurate classification of keyboard actions [47, 48]. The implementation of these modules empowers organizations to enhance their user-level security measures and detect any unauthorized or fraudulent activities [49, 50].

KNN algorithm

The KNN algorithm can be defined as a non-parametric supervised learning classifier that helps classify and predict the data points. It is mainly used in regressive classification techniques. It is one of the widely used classification algorithms that find the similarity between the data points by clustering. Based on most of the votes, the class labels are assigned to the clusters for classifying the data points [51, 52, 53, 54]. It is also termed the plural voting method, where the voting majority exceeds 50 percent, then the cluster is assigned the concerned label. The class labels can also be assigned with a minimum of 25% vote to assign the class labels. The KNN algorithm, short for K-Nearest Neighbors, is a powerful non-parametric supervised learning classifier widely used for data classification and prediction. It operates on the principle of finding similarities between data points by clustering them together [55, 56, 57]. This algorithm is particularly effective in regressive classification techniques. By utilizing a plurality voting method, KNN assigns class labels to data points based on the majority vote within their respective clusters. If the majority vote exceeds 50 percent, the cluster is assigned the corresponding label. However, it is also possible to assign class labels with a minimum vote of 25 percent [58, 59]. In regression techniques, a similar approach is adopted, considering the nearest neighbors for classification (as depicted in Figure-3). KNN is suitable for datasets containing discrete values, while regression techniques are more appropriate for continuous datasets. In the case of continuous datasets, the distance between data points plays a significant role, and the Euclidean distance is commonly calculated to facilitate this process. It is important to note that the KNN algorithm falls under the category of lazy learning algorithms, meaning it requires more training datasets and time to provide accurate results. As an instance-based method, the KNN algorithm heavily relies on the instances between data points to make predictions. All the training data are stored in memory, resulting in increased memory consumption during processing. Consequently, the KNN algorithm is most effective in situations where there is significant variation between data points and the accuracy of the classification process is lower compared to other algorithms [60, 61]. The performance of the KNN algorithm is influenced by various factors, including the dataset's characteristics, accuracy, and size. Larger datasets tend to yield better results; however, processing such extensive datasets requires more time. Figure-4 provides an overview of the overall process of the KNN algorithm [62, 63,64]. Although the KNN algorithm excels in simple decision-making, pattern recognition, data mining, and intrusion detection tasks, its effectiveness depends on the dataset's nature and accuracy. With careful consideration of these factors, the KNN algorithm can deliver satisfactory results in various scenarios [65, 66].

XGB Model

The XGB (Extreme Gradient Boosting) model is employed to analyze the dataset, which consists of 8 samples and 39 features. Due to the smaller size of the available dataset, cross-validation is utilized to assess the accuracy of the analysis. This process takes more time as it explores various parameter combinations to identify the optimal values for the dataset [67, 68, 70, 71]. The GridSearchCV technique is employed to determine the best estimators for detecting keystroke patterns effectively. To ensure robustness in the analysis, the training and test datasets are shuffled and split using a standard procedure. The accuracy of the model is evaluated based on these datasets. The following images provide detailed insights into the performance of the XGB model and illustrate the process of shuffling and splitting the datasets [72, 73,74,75,76]. By leveraging the power of gradient boosting, the XGB model offers enhanced predictive capabilities, making it well-suited for tasks that involve complex relationships and large feature spaces. It effectively learns from the dataset and provides accurate predictions by iteratively refining weak models. The XGB model is widely used in various domains, such as finance, healthcare, and recommendation systems, due to its exceptional performance and versatility.

Techniques adopted.

The adoption of keystroke analysis techniques, such as keystroke logging, has gained widespread popularity across various industries. These techniques provide valuable insights into the timing of key press and release events. Each

individual exhibits unique characteristics in terms of the duration between key presses, key holding time, and the duration between key releases and subsequent key presses. Companies leverage these characteristics to differentiate between users and detect fraudulent or unauthorized access to their systems. However, simply capturing these timestamps is not sufficient to understand the overall typing behavior of an individual. A thorough analysis of these durations is required. Several important parameters are considered to accurately detect and recognize typing patterns. These parameters include the duration between each key press, the duration of key holding, and the duration between key releases and subsequent key presses. Figure-5 provides a detailed illustration of how these parameters are calculated during the typing process. By analyzing the durations and patterns of keystrokes, the typing behavior of users can be identified and distinguished from one another. Figure-6 demonstrates how different parameters are derived based on the durations between the typing of different keys. These keystroke analysis techniques play a crucial role in enhancing user-level security. By precisely analyzing the typing patterns, companies can effectively authenticate users and identify potential security threats. Furthermore, these techniques enable the detection of abnormal or suspicious behavior, allowing organizations to take proactive measures to protect their systems and sensitive information.

5. EXPERIMENTAL RESULTS AND DISCUSSION

The experimental results obtained from the implementation of the proposed machine learning algorithms in Python software are discussed in this section. The performance of the system is evaluated, and the real output obtained from the web application designed for keystroke dynamic analysis is presented. The results are depicted in Table 1(A, B, C), Figure-7 to Figure-10, showcasing the effectiveness of the developed system. Figure-7 demonstrates the real-time output obtained from the web application. The user interface captures the keystrokes and processes them using the implemented machine learning algorithms. The system accurately analyzes the keystroke patterns and identifies various keyboard actions, such as key presses, releases, or new word typing. The output obtained from the system provides valuable insights into the user's typing behavior, enabling enhanced security measures and user-level authentication. Figure-8 presents the performance metrics obtained from the system evaluation. These metrics include accuracy, precision, recall, and F1-score, which are widely used to assess the classification performance of machine learning models. The high values of these metrics indicate the system's capability to accurately classify and predict keyboard actions based on the analyzed keystroke patterns. The results highlight the effectiveness of the implemented algorithms and their potential to enhance user-level security. Figure-9 provides a detailed analysis of the system's performance in different scenarios. It showcases the system's ability to detect abnormal user behavior and identify potential security threats. By analyzing the keystroke dynamics, the system can flag suspicious activities, unauthorized access attempts, or fraudulent behavior. This analysis helps organizations in preventing security breaches and ensuring robust user authentication. Figure-10 and Figure-11 illustrate the comparative analysis of the proposed system with existing techniques or baseline models.

Table 1: A, B, C Keystroke Dynamic Data Analysis

A

	Release-3	press 4	…	press 8
0	664	888	…	1712
1	599	736	…	1423
2	1189	1351	…	2839
3	832	1159	…	3151

B

release 8	press 9	release 9	press-10
1760	1992	2064	2376
1471	1664	1711	1889
2111	2271	2343	2437
3223	3415	3463	3631

C

Release-10	press-11	release-11	press-12
2443	2584	2632	2752
1952	2839	2111	2231
2559	2679	2751	2751

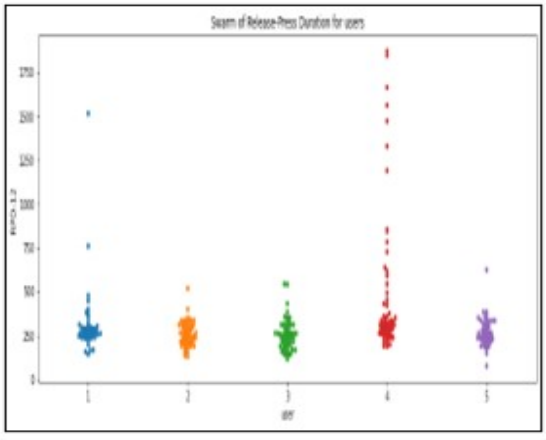

Figure-9. Clustering and Classifying the Data

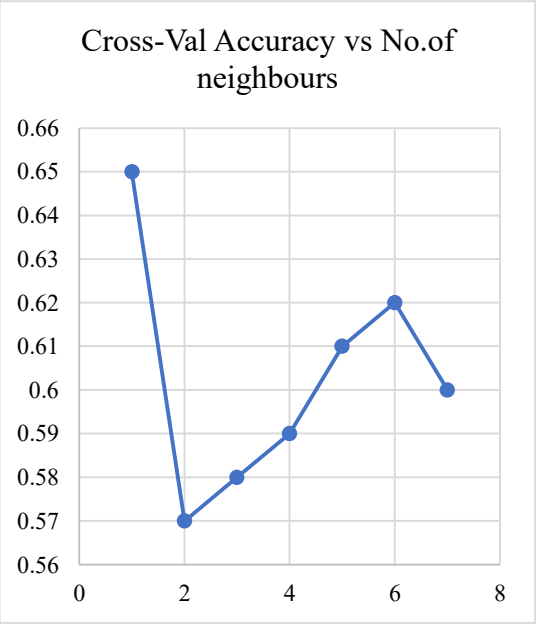

Figure-10. Output Comparison In terms of accuracy

Figure-11. Performance of KNN and XGB

The results demonstrate the superiority of the developed system in terms of accuracy, efficiency, and error rate reduction. The implemented machine learning algorithms outperform traditional methods, showcasing the potential of keystroke dynamic analysis in enhancing user-level security. Overall, the experimental results validate the effectiveness of the proposed system. The real output obtained from the web application showcases the system's ability to accurately analyze keystroke patterns, predict keyboard actions, and enhance user-level security. The performance metrics and comparative analysis highlight the system's superiority over existing techniques, emphasizing its potential in various applications where user authentication and security are crucial.

6. CONCLUSION

The detection of keystroke dynamics plays a crucial role in identifying malicious users within a network. Unlike other biometric methods that can be easily duplicated, the unique typing pattern of a user cannot be replicated. This paper focuses on the identification of typing patterns using the KNN and XGB models. Additionally, it explores the significant parameters involved in keystroke dynamics in detail. To conduct the experiments, an open-source dataset obtained from Kaggle is utilized and simulated within a virtual Python environment such as COLAB. The processing results obtained from the dataset are thoroughly discussed in this paper. The detection of keystroke dynamics plays a crucial role in identifying malicious users within a network. Unlike other biometric methods that can be easily duplicated, the unique typing pattern of a user cannot be replicated. This paper focuses on the identification of typing patterns using the KNN and XGB models. Additionally, it explores the significant parameters involved in keystroke dynamics in detail. To conduct the experiments, an open-source dataset obtained from Kaggle is utilized and simulated within a virtual Python environment such as COLAB. The processing results obtained from the dataset are thoroughly discussed in this paper. The dataset consists of keystroke data collected from various users, which is then processed using the proposed algorithm. The continuous data obtained from the dataset is subjected to processing, and essential parameters such as Hold Duration (HD), Press-Press Duration (PPD), and Release-Press Duration (RPD) values are calculated. These parameters play a crucial role in capturing the unique characteristics of keystroke dynamics. Both the KNN and XGB algorithms are employed to analyze the dataset, and their performances are compared. Upon evaluation, it is observed that the KNN algorithm lacks accuracy in predicting the typing patterns effectively. However, the XGB model outperforms the KNN algorithm in terms of both accuracy and efficiency, as demonstrated by the obtained results. The XGB model exhibits superior performance in capturing and utilizing the essential parameters involved in keystroke dynamics. The findings emphasize the superiority of the XGB model in accurately predicting and classifying the typing patterns of users. Its effectiveness is demonstrated through the analysis of the open-source dataset, providing evidence of

its improved accuracy and efficiency compared to the KNN algorithm. Overall, this paper highlights the significance of identifying keystroke dynamics as a reliable method for detecting malicious users within a network. The utilization of the KNN and XGB models, along with the exploration of important parameters, contributes to enhancing the accuracy and efficiency of the keystroke analysis process. The superior performance of the XGB model further validates its potential for improving user-level security in various applications.

REFERENCES:

[1] Chen, J., Zhu, G., Yang, J., Jing, Q., Bai, P., Yang, W., ... & Wang, Z. L. (2015), "Personalized keystroke dynamics for self-powered human-machine interfacing", ACS nano, 9(1), 105-116.

[2] Antal, M., Szabó, L. Z., & László, I. (2015). Keystroke dynamics on the android platform. Procedia Technology, 19, 820-826.

[3] Çeker, H., & Upadhyaya, S. (2016, September). User authentication with keystroke dynamics in long-text data. In 2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS) (pp. 1-6). IEEE.

[4] Kobojek, P., & Saeed, K. (2016). Application of recurrent neural networks for user verification based on keystroke dynamics. Journal of telecommunications and information technology, (3), 80-90.

[5] Alsultan, A., Warwick, K., & Wei, H. (2017). Non-conventional keystroke dynamics for user authentication. Pattern Recognition Letters, 89, 53-59.

[6] Çeker, H., & Upadhyaya, S. (2017, December). Sensitivity analysis in keystroke dynamics using convolutional neural networks. In 2017 IEEE workshop on information forensics and security (WIFS) (pp. 1-6). IEEE.

[7] Krishnamoorthy, S., Rueda, L., Saad, S., & Elmiligi, H. (2018, May). Identification of user behavioral biometrics for authentication using keystroke dynamics and machine learning. In Proceedings of the 2018 2nd International Conference on Biometric Engineering and Applications (pp. 50-57).

[8] Tsimperidis, I., Arampatzis, A., & Karakos, A. (2018). Keystroke dynamics features for gender recognition. Digital Investigation, 24, 4-10.

[9] Salem, A., & Obaidat, M. S. (2019). A novel security scheme for behavioral authentication systems based on keystroke dynamics. Security and Privacy, 2(2), e64.

[10] Young, J. R., Davies, R. S., Jenkins, J. L., & Pfleger, I. (2019). Keystroke dynamics: establishing keyprints to verify users in online courses. Computers in the Schools, 36(1), 48-68.

[11] Raul, N., Shankarmani, R., & Joshi, P. (2020). A comprehensive review of keystroke dynamics-based authentication mechanism. In International Conference on Innovative Computing and Communications (pp. 149-162). Springer, Singapore.

[12] Choi, M., Lee, S., Jo, M., & Shin, J. S. (2021). Keystroke dynamics-based authentication using unique keypad. Sensors, 21(6), 2242.

[13] Ayotte, B., Banavar, M. K., Hou, D., & Schuckers, S. (2021). Group leakage overestimates performance: A case study in keystroke dynamics. In Proceedings of the ieee/cvf conference on computer vision and pattern recognition (pp. 1410-1417).

[14] El-Kenawy, E. S. M., Mirjalili, S., Abdelhamid, A. A., Ibrahim, A., Khodadadi, N., & Eid, M. M. (2022). Meta-heuristic optimization and keystroke dynamics for authentication of smartphone users. Mathematics, 10(16), 2912.

[15] Kasprowski, P., Borowska, Z., & Harezlak, K. (2022). Biometric Identification Based on Keystroke Dynamics. Sensors, 22(9), 3158.

[16] Yaseein Soubhi Hussein, Ahmed Saeed Alabed, Mustafa Al Mafrachi, Maen Alrshdan, Qusay AlMaatouk, Li-Fi Technology for Smart Cities, Solid State Technology, p 2391-2399, 2020

[17] Norton Setup Blog, Top 10 most infamous digital assault in history, 2019. online: https://norton.comsetup-activate.com/blog/top-10- biggest-cyber-attacks-in-history/.

[18] S. T. Prof P. D. Thakare, "Graphical-Based Password Keystroke Dynamic Authentication System " irjet, vol. 5, no. 2, pp. 2395-0072, 2018.

[19] S. K. Swarna Bajaj, "Typing Speed Analysis of Human for Password Protection (Based On Keystrokes Dynamics)," International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 3, no. 2, pp. 2278-3075, 2013.

[20] R. Chan, "7-Eleven Japan shut down a mobile payments app after only two days because hackers exploited a simple security flaw and customers lost over $500,000," ed, 2019.

[21] K. O'Flaherty, "Password Managers Have A Security Flaw -- Here's How to Avoid It," ed, 2019.

[22] A. Salem, D. Zaidan, A. Swidan, and R. Saifan, "Analysis of Strong Password Using Keystroke Dynamics Authentication in Touch Screen Devices," Amman, Jordan, 2016: IEEE.

[23] S. Mondal, "Context Independent Continuous Authentication using Behavioural Biometrics," 2015: IEEE.

[24] T.-Y. C. Cheng Jung Tasia, Pei Cheng Cheng, Jyun Hao Lin, "Two novel biometric features in keystroke dynamics authentication systems for touch screen devices," Security and Communication Networks, vol. 7, no. 4, pp. 750-758, 2014.

[25] Yadav, Amrendra Singh, et al. "Increasing Efficiency of Sensor Nodes by Clustering in Section Based Hybrid Routing Protocol with Artificial Bee Colony." Procedia Computer Science 171 (2020): 887-896.

[26] R. Arora, "Comparative Analysis of Classification Algorithms on Different Datasets using WEKA," International Journal of Computer Applications vol. 54, no. 13, pp. 0975–8887, 2012.

[27] J. Jayan, "Sequential Minimal Optimization for Support Vector Machines," in towardsdatascience, ed, 2020.

[28] H. Crawford, "Authentication on the Go: Assessing the Effect of Movement on Mobile Device Keystroke Dynamics," Santa Clara, 2017: Usenix.

[29] M. Goel, "WalkType: Using accelerometer data to accommodate situational impairments in mobile touch screen text entry," Seattle, 2017: Research Gate.

[30] C. Giuffrida, "I Sensed It Was You: Authenticating Mobile Users with Sensor-Enhanced Keystroke Dynamics," 2014: Springer.

[31] A. Z. Al-Othmani, A. A. Manaf, A. M. Zeki, Q. Almaatouk, A. Aborujilah and M. T. Al-Rashdan, "Correlation Between Speaker Gender and Perceptual Quality of Mobile Speech Signal," 2020 14th International Conference on Ubiquitous Information Management and Communication (IMCOM), Taichung, Taiwan, 2020, pp. 1-6, DOI: 10.1109/IMCOM48794.2020.9001793.

[32] R. W. S. Scott MacKenzie, "Phrase Sets for Evaluating Text Entry Techniques," New York, 2003: York University.

[33] Mewada, Arvind, et al. "Network intrusion detection using multiclass support vector machine." Special Issue of IJCCT 1.2-4 (2010): 172-175.

[34] Syed Zulkarnain Syed Idrus. Soft Biometrics for Keystroke Dynamics. Computer Vision andPattern Recognition. Universit´e de Caen Basse-Normandie, 2014. English.

[35] Yap Sing Chuen, Maen Al-Rashdan, Qusay AlMaatouk, “Graphical Password Strategy”, Journal of Critical Reviews, Vol 7, Issue 3, 2020

[36] M. Tubishat, M. Alswaitti, S. Mirjalili, M. A. AlGaradi, M. T. Alrashdan and T. A. Rana, "Dynamic Butterfly Optimization Algorithm for Feature Selection," in IEEE Access, vol. 8, pp. 194303- 194314, 2020, doi: 10.1109/ACCESS.2020.3033757

[37] Teo Min Xuan, Maen T. Alrashdan, Qusay AlMaatouk, Mosab Tayseer Alrashdan, “Blockchain Technology in E-Commerce Platform”, International Journal of Management, vol. 11, issue 10, pp. 1688- 1697, 2020, doi: 10.34218/IJM.11.10.2020.154.

[38] R. Gandhi, "Naive Bayes Classifier," in towardsdatascience, ed, 2018. [24]. Derrick Chan Jianli, Maen Al-Rashdan, Qusay AlMaatouk, Secure Data Storage System, Journal of Critical Reviews, Vol 7, Issue 3, 2020.

[39] M. B. Abisado, B. D. Gerardo and A. C. Fajardo, “Towards keystroke analysis using neural network for multi - factor authentication of learner recognition in on - line examination.” In Manila International Conference on Trends in Engineering and Technology (2017), 71–74.

[40] N. M. Agashe and N. Sonali, “A survey paper on continuous authentication by multimodal biometric.” International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Vol. 4, No. 11, 4247–4253, (2015).

[41] D. J. Aleix, A. M. M. Josea and S. P. Eugenia, “Using keystroke dynamics and context features to assess authorship in online learning environments” In 11th International Technology, Education and Development Conference (INTED, 2017), 3167–3176.

[42] M. L. Ali, V. M. John, C. T. Charles and Q. Meikang, “Keystroke biometric systems for user authentication” Journal of Signal Processing Systems, 86(2–3), 175–190. (2017).

[43] A. Laura, M. Caitlin, E. J. Matthew, M. Danielle, C. Scott and D. Sidney, “Investigating

boredom and engagement during writing using multiple sources of information." In Proceedings of the Sixth International Conference on Learning Analytics & Knowledge (New York: ACM New York, NY, USA, 2016), pp. 114– 123.

[44] A. Arwa and W. Kevin, "Keystroke dynamics authentication: A survey of free-text methods." Int. J. Comput. Sci., Vol. 10, Issue 4, No 1, 1–10, (2013).

[45] H. B. Mohammadreza, N. Mehrbaksh, I. Othman, Z. F. Ali and S. Sarminah, "Authentication systems: A Literature Review and Classification" Telematics and Informatics, Vol. 35, Issue 5, 1491–1511, (2018).

[46] K. B. Tanmay and P. Suvasini, "Credit card fraud detection: A hybrid approach using fuzzy clustering & neural network." In 2nd International Conference on Advances in Computing and Communication Engineering (IEEE, 2015), 494–499.

[47] B. Robert and D. Sidney, "Detecting boredom and engagement during writing with keystroke analysis, task appraisals, and stable traits." In Proceedings of the 2013 International Conference on Intelligent User Interfaces (2013), 225–234.

[48] C. Kevin, "Using keystroke analytics to improve pass–fail classifiers", Journal of Learning Analytics, Vol. 4, No. 2, 189–211, (2017).

[49] N. Chourasia, "Authentication of the user by keystroke dynamics for banking transaction system." Proceedings of International Conference on Advances in Engineering & Technology (2014), 41–45.

[50] A.K. Jain, A. Ross and S. Pankanti, "Biometrics: A tool for information security." IEEE Transactions on Information Forensics and Security, Vol. 1, Issue 2, 125–143, (2006).

[51] A. K. Nader and S. Zarina, "Review of user authentication methods in online examination" Asian Journal of Information Technology, Vol. 14, Issue 5, 166–175. (2015). [22] K. Kevin and M. Roy, Why Did My Detector Do That?! Predicting Keystroke-Dynamics Error Rates, (Springer, Berlin, Heidelberg, 2010), 256-257.

[52] S. K. Kevin and R. Maxion, "Free vs. transcribed text for keystroke-dynamics evaluations." In Proceedings of the 2012 Workshop on Learning from Authoritative Security Experiment Results (2012), 1–8.

[53] K. Rajeev and M. K. Sharma, "Advanced neuro-fuzzy approach for social media mining methods using cloud." International Journal of Computer Applications, Vol. 137, Issue 10, 56–58, (2016).

[54] L. Poming, T. Wei-Hsuan, and R. H. Tzu-Chien, "The influence of emotion on keyboard typing: An e xperimental study using auditory stimuli." BioMedical Engineering OnLine, Vol. 3, Issue 1, 81–92. (2014).

[55] M. L. Frank and A. S. Mark, "The impact of an honor code on cheating in online courses." MERLOT Journal of Online Learning and Teaching, Vol. 7, Issue 2, 179–184, (2011).

[56] T. Matthews, "Passwords are not enough." In Computer Fraud and Security, Vol. 2012, 18–20, (2012).

[57] M. Václav and ě=GHQČNAdvanced Communications and Multimedia Security (Springer, Boston, MA, 2002). 1–13.

[58] M. Admir, A. Zikrija and O. Samir, "Intrusion detection system modeling based on neural networks and fuzzy logic." In 2016 IEEE 20th Jubilee International Conference on Intelligent Engineering Systems (INES) (IEEE, 2016), 189–194.

[59] N Deshai, BB Rao, Deep Learning Hybrid Approaches to Detect Fake Reviews and Ratings, Journal of Scientific & Industrial Research 82 (1), 120-127

[60] N Deshai, B Bhaskara Rao, A detection of unfairness online reviews using deep learning, J Theor Appl Inf Technol, 100 (13), 4738-4779

[61] N Deshai, B Sekhar, Processing Real World Datasets using Big Data Hadoop Tools, Journal of Scientific and Industrial Research (JSIR), 79 (1), 631-635

[62] N Deshai, Dr. B.V.D.S.Sekhar, Dr S Venkatramana, Automatic Visual Sentiment Analysis with Convolution Neural network, International Journal of Industrial Engineering & Production Research 32 (2)

[63] S.Venkataramana, N.Deshai, B.V.D.S.Sekhar, Efficient time reducing and energy saving routing algorithm for wireless sensor network, Journal of Physics: Conference Series 1228 (1), 012002

[64] N Deshai, S Venkataramana, B Sekhar, K Srinivas, GPS Varma, A Study on Big Data Processing Frameworks: Spark and Storm, Smart Intelligent Computing and Applications 1, 415-424

[65] N Deshai, BV Sekhar, S Venkataramana, MLlib: machine learning in Apache Spark International Journal of Recent Technology and Engineering 8, 45-49

[66] M. Soumik and B. Patrick, "A study on continuous authentication using a combination of keystroke and mouse biometrics." Neurocomputing 230, 1–22, (2017).

[67] Ö. M. Yasar, E. Ismail Ertürk and S. Refik, "Students' perceptions of online assessment: A case study." Journal of Distance Education, Vol. 19, No. 2, 77–92, (2004).

[68] A. P. Rohit and L. R. Amar, "Keystroke dynamics for user authentication and identification by using typing rhythm." International Journal of Computer Applications, Vol. 144, No. 9, 27–33, (2016).

[69] R. Sucianna, S. Sasmoko, Noerlina and H. Hanry, "Image processing model-based e-learning for students' authentication." In International Conference on Information Management and Technology (ICIMTech, 2017), 187–191.

[70] R. Manuel, G. Sérgio, C. Davide, N. Paulo and F. Florentino, "Keystrokes and clicks: Measuring stress on elearning students." In Second International Symposium Management Intelligent Systems (2013), Vol. 220, 119–126.

[71] BVDS Sekhar et.all., (2019) "An Experimental Analysis of modified EEECARP an Optimized Cluster based Adaptive Routing Protocol for modern-secure-Wireless EEECARP", Novel Theories and Applications of Global Information Resource Management , IGI Global Book Chapter, ISBN13: 9781799817864|ISBN10: 1799817865 DOI: 10.4018/978-1-7998-1786-4.ch012. Pages 318-336.

[72] B.V.D.S. Sekhar Et Al (2019) "A Study On Iot Tools, Protocols, Applications, Opportunities And Challenges ", Information Systems Design And Intelligent Applications, Advances In Intelligent Systems And Computing, Vol:862, Pp:367-380, Springer, 2019 (Springer Link), (Scopus). Https://Doi.Org/10.1007/978-981-13-3329-3_34

[73] B.V.D.S. Sekhar Et Al (2022) "A Novel Technique of Threshold Distance-Based Vehicle Tracking System for Woman Safety", Intelligent System Design, Lecture Notes in Networks and Systems, Springer, India -2022, Nov-2022, ISSN/ISBN: 978-981-19-4862-6, pp: 567-577. https://link.springer.com/chapter/10.1007/978-981-19-4863-3_56, DOI: 10.1007/978-981-19-4863-3_56. (scopus)

[74] B.V.D.S. Sekhar Et Al (August 2021)," The Hybrid Algorithm for increasing Reversable Data Hiding Scheme for Medical Images", International Journal of all Research Education and Scientific Methods IJARESM, Vol:9, Issue:8, August 2021, Issn: 2455-6211, Pages: 2470-2476.

[75] B.V.D.S. Sekhar Et Al (December 2020)," A Novel Technique For Prediction Of Coronary Artery Disease From Human Fundus Images Using Machine Learning Approach", International Journal For Innovative Engineering And Management Research, Vol:7, Issue:12, December'2020, Issn: 2456-5083, Pages: 69-74. [SSSN, Elsevier]

[76] B.V.D.S. Sekhar Et Al (December 2020)," Recognition of Human Being Through Handwritten Digits Using Image Processing Techniques And Ai", International Journal For Innovative Engineering And Management Research, Vol:7, Issue:12, December'2020, Issn: 2456-5083, Pages: 69-74. [SSSN, Elsevier]