UDC 004.056.5
DOI https://doi.org/10.32782/IT/2024-3-5

Vadym KAIDALOV
Postgraduate Student at the Department of Software Engineering, Kharkiv National University of Radio Electronics, 14, Nauky Ave., Kharkiv, Ukraine, 61166, vadym.kaidalov@gmail.com
***ORCID:** 0009-0007-0027-9207*

Vira GOLIAN
Candidate of Technical Sciences, Associate Professor, Kharkiv National University of Radio Electronics, 14, Nauky Ave., Kharkiv, Ukraine, 61166, vira.golan@nure.ua
***ORCID:** 0000-0002-7196-5286*
***Scopus Author ID:** 56008181700*

To cite this article: Kaidalov, V., Golian, V. (2024). Pokrashchennia avtentyfikatsii na osnovi dynamiky natyskannia klavish: balansuvannia tochnosti ta zruchnosti korysuvannia cherez efektyvne navchannia [Improving Keystroke Dynamics Authentication: Balancing Accuracy and User Experience Through Efficient Training]. *Information Technology: Computer Science, Software Engineering and Cyber Security*, 3, 43–50, doi: https://doi.org/10.32782/IT/2024-3-5

IMPROVING KEYSTROKE DYNAMICS AUTHENTICATION: BALANCING ACCURACY AND USER EXPERIENCE THROUGH EFFICIENT TRAINING

***The aim of this study** is to enhance the security and user experience of two-factor authentication systems through the application of keystroke dynamics, a form of behavioral biometrics. Keystroke dynamics analyze the unique typing patterns of users to offer a biometric factor for authentication, which complements the traditional knowledge-based method (username and password). This study specifically seeks to evaluate different anomaly detection algorithms to determine the minimum number of password repetitions required for effective training, optimizing both system security and user convenience.*

***Methodology.** The study replicates and extends the evaluation procedure of Killourhy and Maxion, who provided a public dataset and a detailed protocol for analyzing keystroke dynamics. The algorithms are evaluated by varying the number of password repetitions used for training, with the aim of determining the optimal training size that balances security (lower EER) and efficiency (reduced user effort).*

***Scientific Novelty.** The scientific novelty of this research lies in its investigation of the trade-off between security and user convenience in keystroke dynamics-based authentication systems. While many studies have focused on improving the accuracy of anomaly detection, this research emphasizes the importance of minimizing the training burden on users by determining the minimum number of password repetitions required for stable performance. By focusing on training efficiency and computational resource optimization, this research advances the field of behavioral biometrics and contributes to the practical deployment of keystroke dynamics in real-world authentication systems.*

***Conclusion.** The study demonstrates that keystroke dynamics can significantly improve the security of two-factor authentication systems without imposing excessive burdens on users. The findings confirm that the Manhattan (scaled) and Outlier Count (z-score) algorithms perform relatively well, particularly when the training set size is small, which is critical for practical use in authentication systems where users may be unwilling to provide numerous password repetitions. This study not only replicates the results of prior research but also contributes new insights into optimizing the training process for keystroke dynamics-based anomaly detection. Future work may explore integrating keystroke dynamics with other biometric factors, such as mouse dynamics, to develop even more secure and user-friendly multimodal authentication systems. Furthermore, continuous authentication mechanisms represent an exciting direction for future research, providing ongoing verification of user identity throughout a session rather than solely at login.*

***Key words:** keystroke dynamics, two-factor authentication, anomaly detection, behavioral biometrics, user authentication, security, continuous authentication.*

Вадим КАЙДАЛОВ
аспірант кафедри Програмної Інженерії, Харківський національний університет радіоелектроніки, пр. Науки, 14, м. Харків, Україна, 61166
***ORCID:** 0009-0007-0027-9207*

Віра ГОЛЯН
кандидат технічних наук, доцент кафедри Програмної Інженерії, Харківський національний університет радіоелектроніки, пр. Науки, 14, м. Харків, Україна, 61166
***ORCID:** 0000-0002-7196-5286*
***Scopus Author ID:** 56008181700*

Бібліографічний опис статті: Кайдалов, В., Голян, В. (2024). Покращення автентифікації на основі динаміки натискання клавіш: балансування точності та зручності користування через ефективне навчання. *Information Technology: Computer Science, Software Engineering and Cyber Security*, 3, 43–50, doi: https://doi.org/10.32782/IT/2024-3-5

ПОКРАЩЕННЯ АВТЕНТИФІКАЦІЇ НА ОСНОВІ ДИНАМІКИ НАТИСКАННЯ КЛАВІШ: БАЛАНСУВАННЯ ТОЧНОСТІ ТА ЗРУЧНОСТІ КОРИСТУВАННЯ ЧЕРЕЗ ЕФЕКТИВНЕ НАВЧАННЯ

***Мета цього дослідження** – покращити безпеку та досвід користувачів систем двофакторної аутентифікації шляхом застосування динаміки натискання клавіш, форми поведінкової біометрії. Динаміка натискання клавіш аналізує унікальні шаблони набору тексту користувачів для створення біометричного фактору автентифікації, який доповнює традиційний метод, що базується на знаннях (ім'я користувача та пароль). Це дослідження спрямоване на оцінку різних алгоритмів виявлення аномалій, щоб визначити мінімальну кількість повторень пароля, необхідну для ефективного навчання, оптимізуючи як безпеку системи, так і зручність для користувача.*

***Методологія.** Дослідження повторює і розширює процедуру оцінки Killourhy та Maxion, які надали публічний набір даних і детальний протокол для аналізу динаміки натискання клавіш. Алгоритми оцінюються шляхом варіювання кількості повторень пароля, використаних для навчання, з метою визначення оптимального розміру навчання, що збалансовує безпеку (менший EER) та ефективність (зменшене навантаження на користувача).*

***Наукова новизна.** Наукова новизна цього дослідження полягає в дослідженні компромісу між безпекою і зручністю для користувача в системах автентифікації на основі динаміки натискання клавіш. Хоча багато досліджень зосереджені на поліпшенні точності виявлення аномалій, це дослідження підкреслює важливість мінімізації навантаження на користувачів шляхом визначення мінімальної кількості повторень пароля, необхідної для стабільної роботи. Зосереджуючи увагу на ефективності навчання та оптимізації обчислювальних ресурсів, це дослідження просуває галузь поведінкової біометрії і сприяє практичному впровадженню динаміки натискання клавіш у реальних системах автентифікації.*

***Висновок.** Дослідження демонструє, що динаміка натискання клавіш може суттєво покращити безпеку систем двофакторної автентифікації без накладення надмірного навантаження на користувачів. Результати підтверджують, що алгоритми Manhattan (scaled) і Outlier Count (z-score) працюють відносно добре, особливо коли розмір навчального набору малий, що критично для практичного використання в системах аутентифікації, де користувачі можуть бути не готові надати численні повторення пароля. Це дослідження не лише повторює результати попередніх досліджень, але й вносить нові ідеї для оптимізації процесу навчання для виявлення аномалій на основі динаміки натискання клавіш. Майбутні роботи можуть досліджувати інтеграцію динаміки натискання клавіш з іншими біометричними факторами, такими як динаміка миші, щоб розробити ще більш безпечні та зручні багатомодальні системи автентифікації. Крім того, механізми безперервної автентифікації є захоплюючим напрямком для майбутніх досліджень, забезпечуючи постійну перевірку ідентичності користувача протягом сесії, а не тільки при вході.*

***Ключові слова:** динаміка натискання клавіш, двофакторна автентифікація, виявлення аномалій, поведінкова біометрія, автентифікація користувачів, безпека, безперервна автентифікація.*

Introduction. User authentication is the process of verifying a user's identity widely used in computer systems to protect data from unauthorized access. Typically, a username and password pair is used as a piece of the knowledge that only the genuine user should know to confirm their identity. However, such an approach is often not secure enough, leading to the introduction of the so-called «second factor» in the authentication process. Generally, there are three types of the factors: knowledge-based (something a user knows, e.g., the username and password values), possession-based (something a user has, e.g., their personal smartphone), and biometric-based (something a user is, e.g., their iris scan). The introduction of the second factor makes it harder for impostors to deceive the authentication system. However, the possession-based factors usually require additional effort from the genuine user, such as unlocking their smartphone and entering a PIN code or responding to a notification, thus worsening the user experience. Biometric authentication uses unique biometric traits of individuals to verify their identity. These traits can be broadly divided into physiological (e.g., iris scan, fingerprint, voice) and behavioural (e.g., typing rhythms, mouse or touchscreen navigation patterns). In particular, keystroke dynamics is the process of identifying individual users on the basis of their typing rhythms, which are in turn derived from the timestamps of key-press and key-release events on the keyboard (Maxion & Killourhy, 2010, p. 201-210). This form of authentication uses behavioural biometric traits of users. If the authentication process is performed once when the user enters the system, it is called «static» authentication. If the biometric authentication is performed continuously during the user's session, it is called the «continuous» authentication (Ryu et al., 2021, p. 34541-34557). The current study examines a two-factor authentication system that verifies a user's identity by confirming their password as something that only the genuine user knows and applying keystroke dynamics on the timestamps of the keyboard events produced during the password entry. Such a system does not require any additional effort from users while being more secure due to the use of a second factor for authentication.

Related works. Killourhy and Maxion enabled the comparison of different anomaly detectors' performance across studies in the keystroke dynamics literature. They publicly shared a data set, developed an evaluation procedure, and measured the performance of various anomaly-detection algorithms on an equal basis (Maxion & Killourhy, 2010, p. 201-210). Typing data was collected from 51 subjects, each typing the same password 400 times. The researchers extracted various timing features from the raw data, such as keydown-keydown times and hold times. Fourteen anomaly detectors from the literature were reimplemented and evaluated according to a well-defined procedure. Another study by the same authors analysed factors influencing the accuracy of anomaly-detection algorithms: the algorithm itself, amount of training, choice of features, use of updating, impostor practice, and typist-to-typist variation (Killourhy & Maxion, 2010, p. 256-276). Their results indicated that the algorithm, amount of training, and use of updating were highly influential while impostor practice and feature set had minor effect. Some typists were significantly easier to distinguish than others. The researchers considered training amounts of 5, 50, 100, and 200 password repetitions done by the genuine user during the model training stage. While their study aimed to determine whether the amount of training could significantly influence the results in general, the problem of achieving stable accuracy rates with minimal training was not covered. Comparing the number of password repetitions required from users to train different anomaly-detection models can help select better models in terms of user experience, which is covered in this study. Additionally, determining the minimum number of repetitions needed to train an anomaly-detection model can help choose a reasonable size for the sliding window used in updating typing profiles, thus optimizing the use of computational resources.

Purpose. The purpose of this study is to reproduce the evaluation procedure described by Killourhy and Maxion, develop a method for comparing the minimum number of password repetitions needed to achieve stable accuracy rates for the best-performing anomaly detectors identified by the researchers, and share the Python source code with the research community to facilitate reproducibility. By determining the minimum number of repetitions required for training, the study aims to optimize the size of the sliding window used in updating typing profiles, thereby conserving computational resources.

Methodology. The data set shared by Killourhy and Maxion consists of keystroke-timing data collected from 51 subjects over 8 sessions. Each subject was asked to type the same password, «.tie5Roanl», 50 times during each session, providing timing information for a total of 400 password entries. The rationale for the choice of the password and other aspects of the data collection were described in detail by the researchers (Killourhy & Maxion, 2009, p. 125-134).

The raw typing data, such as key events and timestamps, cannot be used directly by an anomaly detector. Instead, sets of timing features are extracted from this raw data and organized into a vector of times, known as a timing vector. These features are used to train and test the detectors. The time between the key presses of consecutive keys (Keydown-Keydown), the time between the release of one key and the press of the next (Keyup-Keydown), and the time between the press and release of each key (Hold) are all available as features in the timing vectors provided by the mentioned data set. However, a study has shown there is no difference in anomaly detection accuracy among feature sets that include the Hold features and at least either Keydown-Keydown or Keyup-Keydown features (Killourhy & Maxion, 2010, p. 256-276). Therefore, only the Hold and Keyup-Keydown features are used from the data set for the evaluation procedure of the current study to conserve computational resources. The Enter key is also considered part of the password, so its Keyup-Keydown and Hold features are included as well. Given the aforementioned password and the Enter key inclusion, each timing vector consists of 21 features: 11 Hold features and 10 Keyup-Keydown features.

Killourhy and Maxion implemented and evaluated 14 anomaly detectors from the keystroke-dynamics and pattern-recognition literature (Killourhy & Maxion, 2009, p. 125-134). They observed a clear division between seven detectors that were competitive in their evaluation and seven that were not. In the current study, the 5 best-ranked anomaly detectors have been reproduced according to the descriptions provided by the researchers: «Manhattan (scaled)», «Nearest Neighbour (Mahalanobis)», «Outlier Count (z-score)», one-class SVM and «Mahalanobis».

Typing data produced by an impostor should be detected as anomalous by an anomaly detector. In this context, the presence of an anomaly is referred to as a positive outcome, while its absence is called a negative outcome. When an anomaly detector incorrectly identifies a timing vector produced by the genuine user as anomalous, this mistake is termed a «false positive». All possible outcomes are shown in Table 1.

An anomaly detector produces a numeric value as a score assigned to the input timing vector. A threshold value must be chosen so that the detector marks the timing vector as anomalous if the score exceeds the threshold. The choice of a threshold value greatly influences the performance rates of detectors, so a range of threshold values should be used for performance measurements.

Given a certain threshold value, the True Positive Rate (TPR) and the False Positive Rate (FPR) are defined as follows:

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$

In the keystroke-dynamics authentication literature, the True Positive Rate (TPR), also known as the «Hit Rate», indicates the frequency with which a detector correctly identifies an impostor. The «Miss Rate», defined as (1 – TPR), reflects the rate at which an impostor is mistakenly identified as a genuine user. The False Positive Rate (FPR), or «False-Alarm Rate» (FAR), denotes the frequency with which a genuine user is incorrectly rejected.

Given a constant set of the Hold and Keyup-Keydown features selected, the evaluation procedure described by Killourhy and Maxion (Killourhy & Maxion, 2010, p. 256-276) uses the first T training repetitions produced by a genuine-user subject S to train an anomaly-detection algorithm A. Here T can be 5, 50, 100 or 200 repetitions, S can be any of the subject identifiers specified in the data set, and A can be one of the mentioned algorithms. At the scoring stage, the genuine-user test data is composed of the subsequent 200 repetitions from (T + 1) to (T + 200). The «practiced» impostor test data consists of the last 5 password repetitions from each of the remaining 50 users, resulting in a total of 250 impostor timing vectors. As shown by the researchers, impostor practice represents a minor threat to the accuracy of keystroke-dynamics detectors but is still included in the current evaluation procedure to ensure realism.

The evaluation procedure for the sliding-window updating case is more complex. The concept involves sliding a window of size T over the genuine user's typing data, advancing the window in increments of 5 repetitions for computational efficiency. For each window, the detector is trained on the repetitions within that window and then tested using the next five repetitions. This process is repeated as the window is incremented. Since there are 200 repetitions of genuine-user test data, this results in 40 cycles of training and testing (200/5). The evaluation procedure is well-defined by the researchers (Killourhy & Maxion, 2010, p. 256-276) and is accurately reproduced in the current study, as shown in the following sections. According to their results, the use of updating is a highly influential factor.

Across the keystroke-dynamics authentication literature, such anomaly detectors performance measures as the Equal Error Rate (EER) and the Zero-Miss False Alarm Rate (ZMFAR) have been used. They both give an understanding of a detector's performance over a range of different treshold values. The EER is defined as the point at which the Miss Rate and the False-Alarm Rate are equal. At this point, the system's rate of incorrectly rejecting genuine users is equal to the rate

Table 1

Table of possible outcomes of an anomaly detector's work

Anomaly detection result	Anomaly is actually present (Impostor)	No anomaly is actually present (Genuine user)
Anomaly is detected	True Positive (TP)	False Positive (FP)
No anomaly is detected	False Negative (FN)	True Negative (TN)

of incorrectly accepting impostors. The EER provides a single value that reflects the overall accuracy of the system, with a lower EER indicating better performance. The ZMFAR is defined as the minimum False-Alarm Rate when the threshold is set to ensure the Miss Rate is zero. This metric reflects a detector's performance under the condition of zero tolerance for impostors involved in the performance evaluation.

A Receiver Operating Characteristic (ROC) curve is used to visualize a detector's performance across various threshold values. Figure 1 illustrates an example of a ROC curve for the «Outlier Count (z-score)» algorithm, trained on data from the subject «s010» (excluding Keydown-Keydown features, with a sliding window enabled, and a training set size of 15).

Intuitively, a larger area under the ROC curve indicates higher overall performance of an anomaly detector. While minor variations in the ratio of outcomes can significantly affect the ZMFAR value, EER is a more robust and balanced performance measure that represents the trade-off between false alarms and misses. Consequently, EER has been selected to compare the performance of detectors, as demonstrated in the following sections.

The anomaly-detection algorithms and the described evaluation procedure have been implemented using Python, NumPy, Pandas, and scikit-learn. These implementations have been shared on GitHub to support further research (Kaidalov, 2024).

Findings. Firstly, it was crucial to replicate the EER and ZMFAR values reported by Killourhy and Maxion from their initial comparison of anomaly detectors (Killourhy & Maxion, 2009, p. 125-134). The values reported by the researchers and those reproduced in the current study are presented in Table 2.

While most of the rates match with insignificant differences, there is an almost 2% difference in the

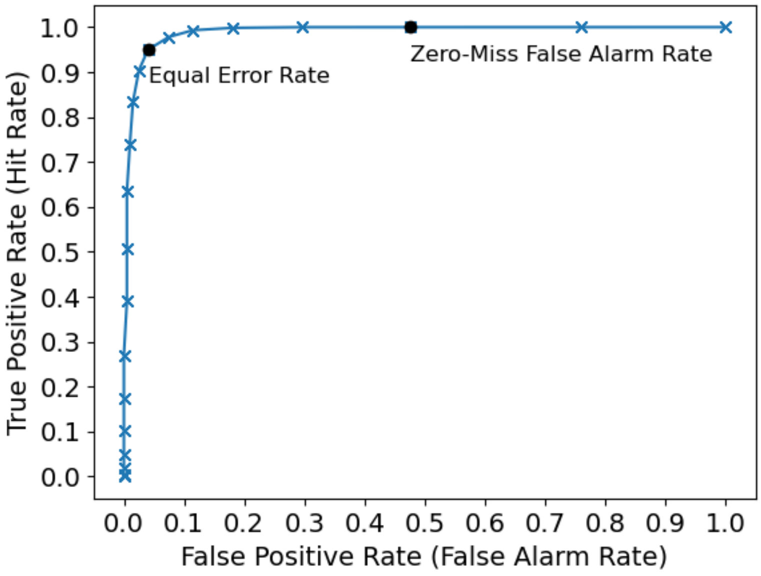

Fig. 1. An example ROC curve for the «Outlier Count (z-score)» algorithm

Table 2

Table of the EER and ZMFAR values reported in (Killourhy & Maxion, 2009, p. 125-134) and reproduced in the current study

Anomaly-detection algorithm	Average EER in (Killourhy & Maxion, 2009, p. 125-134)	Reproduced average EER	Average ZMFAR in (Killourhy & Maxion, 2009, p. 125-134)	Reproduced average ZMFAR
Manhattan (scaled)	0.096 (0.069)	0.098 (0.068)	0.601 (0.337)	0.610 (0.333)
Nearest Neighbour (Mahalanobis)	0.100 (0.064)	0.100 (0.063)	0.468 (0.272)	0.468 (0.270)
Outlier Count (z-score)	0.102 (0.077)	0.101 (0.076)	0.782 (0.306)	0.782 (0.303)
One-class SVM	0.102 (0.065)	0.119 (0.059)	0.504 (0.316)	0.500 (0.282)
Mahalanobis	0.110 (0.065)	0.110 (0.064)	0.482 (0.273)	0.482 (0.270)

EER for the one-class SVM algorithm. This discrepancy can be attributed to differences in implementations. Although the researchers specified the ν parameter as 0.5, they did not indicate which kernel was used. In the current study, the default values of the scikit-learn package were used for the remaining parameters of OneClassSVM during evaluation.

The extended evaluation procedure published by Killorhy and Maxion (Killourhy & Maxion, 2010, p. 256-276) had introduced such parameters as sliding windows updates, training amount, impostors practice, and a feature set. The exact EER values for the 'Manhattan (scaled)' algorithm were reported as 7.1% for unpracticed impostors and 9.7% for practiced impostors. Sliding window updates were used, Keydown-Keyup features and Enter features were excluded, and the training amount was set to 100. The evaluation procedure reproduced in this study successfully replicates the EER values reported by the researchers.

The implemented anomaly detectors were evaluated with training set sizes ranging from 5 to 100, increasing by increments of 5 password repetitions. Given its significant impact, updating was enabled. The feature set included Hold and Keyup-Keydown features, as well as Enter key features. To ensure realism, impostor practice was enabled. The obtained EER values are presented in Table 3, and a visual comparison of EER values at different training set sizes is shown in Figure 2.

Figure 2 presents a line plot illustrating the Equal Error Rate (EER) of various anomaly detection algorithms as a function of training set size. The x-axis represents the training set size, ranging from 5 to 100, while the y-axis indicates the EER, ranging from 0 to 0.35.

The «Manhattan (scaled)» algorithm shows a gradual decline in EER as the training set size increases, indicating improved performance with more training data. In contrast, the «Outlier count (z-score)» algorithm initially decreases but then shows an increase in EER, suggesting possible overtraining as the model becomes too specialized to the training data and loses generalization capability.

The «Nearest Neighbour (Mahalanobis)» algorithm demonstrates only minor improvements over the Mahalanobis algorithm, despite requiring significantly higher computational resources. This suggests that the algorithm's additional complexity may not be justified given the marginal performance gains. The one-class SVM algorithm maintains a relatively constant high EER across all training set sizes.

Notably, the EER values for the «Manhattan (scaled)» and «Outlier count (z-score)» algorithms are relatively lower at smaller training set sizes,

Table 3

Table of EER values for each reproduced anomaly detector at training set sizes ranging from 5 to 100

Training set size	Manhattan (scaled)	Nearest Neighbour (Mahalanobis)	Outlier Count (z-score)	One-class SVM	Mahalanobis
5	0.135	0.302	0.100	0.282	0.287
10	0.117	0.267	0.093	0.269	0.263
15	0.110	0.228	0.092	0.262	0.227
20	0.109	0.237	0.095	0.257	0.236
25	0.106	0.209	0.099	0.254	0.210
30	0.105	0.173	0.100	0.251	0.173
35	0.104	0.160	0.101	0.248	0.159
40	0.102	0.152	0.101	0.242	0.152
45	0.099	0.143	0.100	0.236	0.143
50	0.097	0.136	0.100	0.232	0.136
55	0.095	0.130	0.099	0.229	0.131
60	0.094	0.125	0.101	0.229	0.128
65	0.093	0.123	0.102	0.223	0.125
70	0.093	0.123	0.102	0.222	0.126
75	0.093	0.122	0.103	0.222	0.126
80	0.093	0.120	0.103	0.220	0.125
85	0.093	0.118	0.103	0.217	0.124
90	0.093	0.118	0.104	0.215	0.123
95	0.092	0.115	0.102	0.212	0.121
100	0.091	0.114	0.103	0.209	0.119

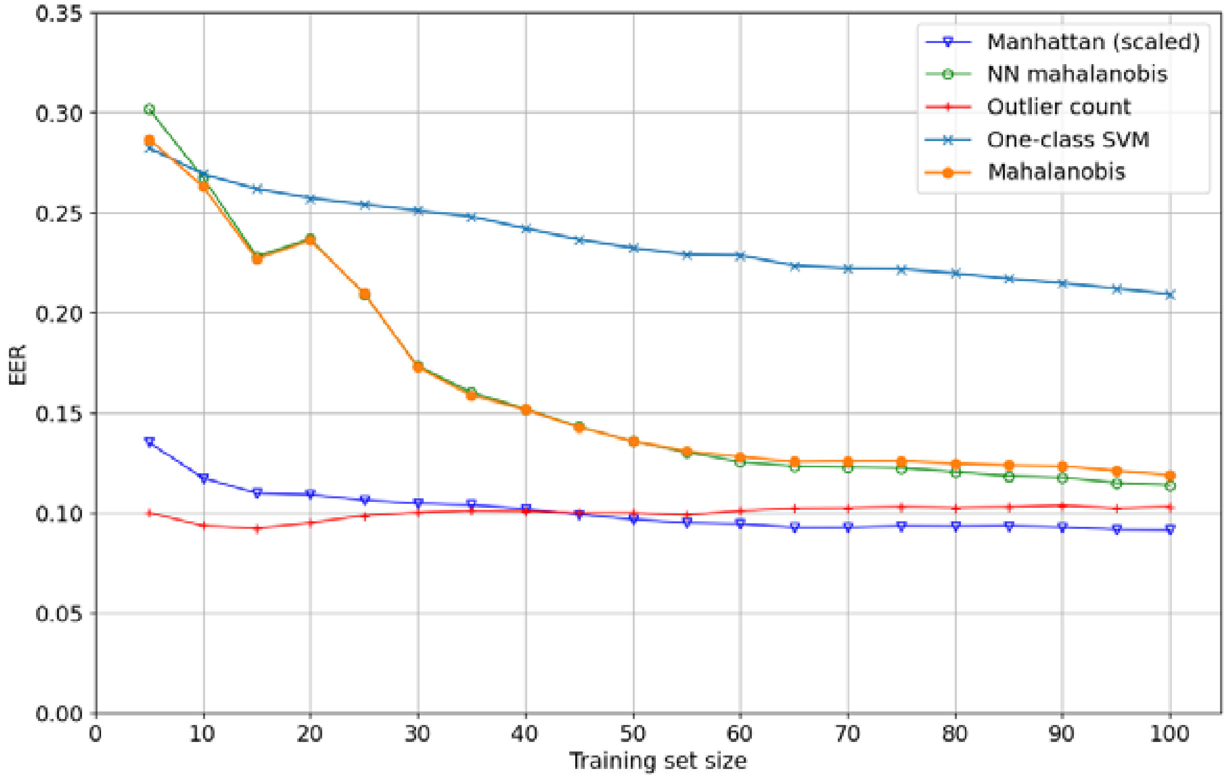

Fig. 2. The graph of the EER values for each reproduced anomaly detector at training set sizes ranging from 5 to 100

suggesting higher practical value when limited data is available.

Conclusion. This study has successfully replicated and extended the evaluation of anomaly detection algorithms applied to keystroke dynamics for two-factor authentication systems. By reproducing the performance metrics (EER and ZMFAR) for various algorithms as initially reported by Killourhy and Maxion, we have confirmed the reliability of their findings. Among the algorithms tested, the «Manhattan (scaled)» and «Outlier Count (z-score)» showed promising results, especially when using smaller training sets, which is critical for practical deployment.

One significant outcome of this research is the determination of the minimum number of password repetitions needed to achieve stable accuracy rates. This result is particularly valuable for reducing user effort in the training phase and optimizing computational resources during real-time authentication processes. The methodology and Python code made available further contribute to reproducibility, a key aspect of advancing research in keystroke dynamics.

Looking forward, several promising areas for future exploration have emerged. First, while the current study focused on a single password input, expanding the analysis to include more diverse passwords could improve the robustness of anomaly detection models. Second, exploring the integration of keystroke dynamics with other biometric factors, such as mouse dynamics or touchscreen patterns, could lead to the development of multimodal biometric systems, offering enhanced security and user experience. Additionally, improving the performance of machine learning algorithms, particularly through the use of deep learning models, might yield better generalization capabilities for unseen data.

Another avenue for future work includes investigating the potential for continuous authentication mechanisms. Rather than only verifying users at login, continuous authentication can monitor user behaviour throughout their session, providing an added layer of security. Lastly, optimizing the computational efficiency of the system remains a critical challenge, especially for real-time applications. Exploring lightweight models or optimizing feature extraction processes could help deploy keystroke-based authentication systems in resource-constrained environments.

In conclusion, the findings of this study contribute to the ongoing development of secure and user-friendly authentication systems. Future research should focus on extending the methods presented here and addressing the emerging challenges in keystroke dynamics and behavioural biometrics.

BIBLIOGRAPHY:

1. R. A. Maxion, K. S. Killourhy. Keystroke biometrics with number-pad input. 2010 IEEE/IFIP International Conference on Dependable Systems & Networks (DSN). 2010. P. 201–210. URL: https://doi.org/10.1109/DSN.2010.5544311 (date of access 19.09.2024)

2. R. Ryu, S. Yeom, S.-H. Kim, D. Herbert. Continuous Multimodal Biometric Authentication Schemes: A Systematic Review. IEEE Access. Vol. 9. P. 34541–34557. 2021. URL: https://doi.org/10.1109/ACCESS.2021.3061589 (date of access 19.09.2024).

3. K. S. Killourhy, R. A. Maxion. Why Did My Detector Do *That*?! Recent Advances in Intrusion Detection. RAID 2010. Lecture Notes in Computer Science. 2010. Vol 6307. P. 256–276. URL: https://doi.org/10.1007/978-3-642-15512-3_14 (date of access 19.09.2024).

4. K. S. Killourhy, R. A. Maxion. Comparing anomaly-detection algorithms for keystroke dynamics. 2009 IEEE/IFIP International Conference on Dependable Systems & Networks. P. 125–134. URL: https://doi.org/10.1109/DSN.2009.5270346 (date of access 19.09.2024).

5. V. D. Kaidalov. GitHub – vkaidalov/keystroke-dynamics-authentication. GitHub. 2024. URL: https://github.com/vkaidalov/keystroke-dynamics-authentication (date of access: 19.09.2024).

REFERENCES:

1. Maxion, R. A., Killourhy, K. S. (2010). Keystroke biometrics with number-pad input. 2010 IEEE/IFIP International Conference on Dependable Systems & Networks (DSN). P. 201–210. Retrieved from: https://doi.org/10.1109/DSN.2010.5544311 (date of access 19.09.2024)

2. Ryu, R., Yeom, S., Kim, S. -H., Herbert, D. (2021). Continuous Multimodal Biometric Authentication Schemes: A Systematic Review. IEEE Access. Vol. 9. P. 34541–34557. Retrieved from: https://doi.org/10.1109/ACCESS.2021.3061589 (date of access 19.09.2024).

3. Killourhy, K. S., Maxion, R. A. (2010). Why Did My Detector Do *That*?! Recent Advances in Intrusion Detection. RAID 2010. Lecture Notes in Computer Science. Vol 6307. P. 256–276. Retrieved from: https://doi.org/10.1007/978-3-642-15512-3_14 (date of access 19.09.2024).

4. Killourhy, K. S., Maxion, R. A. (2009). Comparing anomaly-detection algorithms for keystroke dynamics. 2009 IEEE/IFIP International Conference on Dependable Systems & Networks. P. 125–134. Retrieved from: https://doi.org/10.1109/DSN.2009.5270346 (date of access 19.09.2024).

5. Kaidalov, V. D. (2024) GitHub – vkaidalov/keystroke-dynamics-authentication. *GitHub*. Retrieved from: https://github.com/vkaidalov/keystroke-dynamics-authentication (date of access: 19.09.2024).