

## Article

# Investigation of the Impact of Damaged Smartphone Sensors' Readings on the Quality of Behavioral Biometric Models

Paweł Rybka <sup>1</sup>, Tomasz Bąk <sup>1</sup>, Paweł Sobel <sup>1</sup> and Damian Grzechca <sup>2,\*</sup> <sup>1</sup> Digital Fingerprints, ul. Żeliwna 38, 40-599 Katowice, Poland<sup>2</sup> Faculty of Automatic Control, Electronics and Computer Science, Silesian University of Technology, ul. Akademicka 16, 44-100 Gliwice, Poland

\* Correspondence: dgrzechca@polsl.pl

**Abstract:** Cybersecurity companies from around the world use state-of-the-art technology to provide the best protection against malicious software. Recent times have seen behavioral biometry becoming one of the most popular and widely used components in MFA (Multi-Factor Authentication). The effectiveness and lack of impact on UX (User Experience) is making its popularity rapidly increase among branches in the area of confidential data handling, such as banking, insurance companies, the government, or the military. Although behavioral biometric methods show a high degree of protection against fraudsters, they are susceptible to the quality of input data. The selected behavioral biometrics are strongly dependent on mobile phone IMU sensors. This paper investigates the harmful effects of gaps in data on the behavioral biometry model's accuracy in order to propose suitable countermeasures for this issue.

**Keywords:** behavioral biometrics; machine learning; MEMS; Multi-Factor Authentication



**Citation:** Rybka, P.; Bąk, T.; Sobel, P.; Grzechca, D. Investigation of the Impact of Damaged Smartphone Sensors' Readings on the Quality of Behavioral Biometric Models. *Sensors* **2022**, *22*, 9580. <https://doi.org/10.3390/s22249580>

Academic Editors: Shih-Hau Fang, Wei Wei, Hsiao-Chun Wu and Kun Yan

Received: 14 October 2022

Accepted: 2 December 2022

Published: 7 December 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

According to the FBI's (Federal Bureau of Investigation) 2021 report on Internet crime [1], the number of phishing attacks reported to the IC3 (Internet Crime Complaint Center) doubled in 2021 (241,342 incidents) compared to 2020 (114,702 incidents) and was almost ten-times higher than in 2019 (26,379 incidents). This data clearly shows how popular phishing attacks [2,3] are and how the demand for phishing countermeasures is growing in the government, banking, and military sectors. Although a multitude of companies and governments organize staff training on cybersecurity, accounts are still being hacked as fraudster attacks become smarter and better targeted [4,5]. Fortunately, even if the user's login and password have been voluntarily provided to the fraudster, there are still a number of ways to protect accounts against being hijacked; one of them is behavioral biometrics.

Smartphone sensors (for example, accelerometers, magnetometers, gyroscopes) find a lot of applications when it comes to mobile apps—from entertainment (mobile games) to monitoring user's behavior (pedometers, sleep monitoring, and others). Recently, the use of smartphone sensors has found applications in more advanced systems such as health monitoring [6] or cybersecurity. As the consumption of multimedia and mobile resources access rises day-to-day [7], the topic of behavioral biometry and its impact on cybersecurity has recently been present in many research papers covering both desktop [8] and mobile [9] device usage.

The latest studies [10,11] have shown that the use of behavioral biometrics has become an increasingly popular part of MFA (Multi-Factor Authentication) [12]. Over the last couple of years, institutions for confidential data handling have been more prone to reach for users' behavioral patterns (e.g., keyboard strokes, mouse movements, or mobile device handling) when implementing identity theft countermeasures, as this sort of data does not require any additional user involvement harmful for the UX (User Experience) [13].

Moreover, behavioral biometric models show high resistance to fraudsters as their behavior is vastly different from what users tend to do; as such, the combination of suspicious activity on user's account (e.g., a transfer for a high amount) with unusual keyboard or smartphone readings may indicate an attack [14,15].

Although the individual user behavioral biometrics model is a powerful weapon against account hijacking, it is still vulnerable to low data quality delivered from used devices. Swapping mobile phones or damaging certain sensors may lead to reduced quality of fraud detection. For the sake of maintaining the high quality of behavioral biometry authentication services, it is a must to implement precautionary rules. This paper compares how exemplary behavioral models based on aggregated accelerometer and gyroscope readings deal with incomplete anonymized user data. The first stage for applying quality drop countermeasures would be labeling certain users' behavioral data readings as "low quality" to prevent data damage that reduces the model's classification accuracy below established thresholds. The next stages would be more complex—for example, providing user models resistant to sensor damage or applying additional safe mode pipelines.

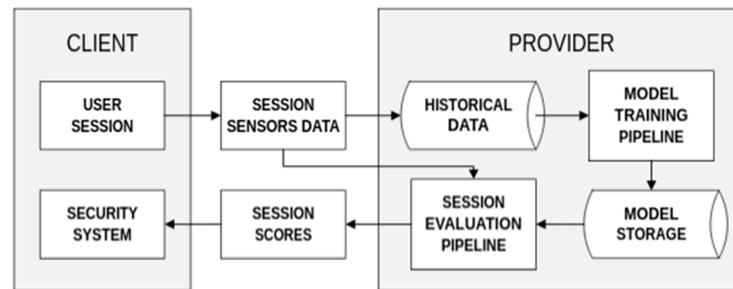
The most recent papers published on the matter of behavioral biometrics show experimental attitudes towards data and introduce results which do not cover real-life scenario difficulties. The following analysis presents a real-life industrial-based case study on behavioral authentication for one of the leading national banks. This paper covers all the data processing pipeline issues and problems regarding the quality and quantity of certain users' data as well as the struggle encountered with the use of multiple devices and OS versions.

At the beginning of the manuscript, the dataset is introduced; then, the mathematical background of the behavioral model (input vectors, structure, hyperparameters etc.) is presented. In the end, the models' input vectors are artificially disturbed with commonly occurring data damage and their quality is examined to verify whether the presented solution is susceptible to certain types of data absence.

The related papers the authors highly recommend getting familiarized with are [16], where researchers present a proposal of a continuous authentication system for smartphone user classification based on interactions with the device, and [17], which provides BehavePassDB—a public database for mobile behavioral biometrics solution benchmarking. This database can be used as a sandbox for testing new features and classification algorithms before feeding further data. The use of behavioral biometry in mobile devices provides reliable security for zero price when UX is considered, and numerous researchers emphasize the importance of maintaining UX of the highest quality [18]. It is the clients themselves (banks and other institutions covered by behavioral biometry cybersecurity solutions) that insist on keeping the system user-friendly. Additionally, as the global COVID-19 pandemic and its repercussions caused severe changes in the use of digital resources, behavioral biometry has proven to be a high-quality cyberattack countermeasure in fields where other security systems have failed [19].

## 2. Materials and Methods

The data used for this research was acquired from the accelerometer and gyroscope sensors of smartphone devices running banking applications on the Android operating system [20], coming from users of one of the leading national banks. Sensor readings were collected from the beginning until the end of use of the banking application. In this case, the data was sent to the upstream node and evaluated in real time. Whenever fraudulent behavior is detected at any stage of the session, an alert signal is sent to the mobile application provider (usually the bank's department of security). The alerting system does not take any additional meta-data apart from an historical behavioral profile. No information on age, gender, banking history, device type, OS, or other data is stored or analyzed. The following block diagram presents how the information exchange between the client and the provided security system is organized (Figure 1):



**Figure 1.** Data exchange pipeline for the behavioral biometry security system.

### 2.1. Dataset Structure

Users  $u$  perform a certain number  $SN_u$  of connections called sessions  $S^u$  (1) with the server via the banking application. Each user session  $s_i^u$  that lasts for  $TS_{s_i^u}$  seconds, consists of feature vectors whose count is a number denoted by  $FC_{s_i^u}$  (2). Every single user feature vector  $D$  in the entire population consists of a fixed number ( $N = 6$ ) of features  $d$  (3). The feature vector used for the analysis was formed in the following manner: accelerometer  $x$  axis; accelerometer  $y$  axis; accelerometer  $z$  axis; gyroscope  $x$  axis; gyroscope  $y$  axis; gyroscope  $z$  axis.

$$S^u = [s_1^u, s_2^u, \dots, s_k^u, \dots, s_{SN_u}^u] \quad (1)$$

$$s_i^u = [D_1, D_2, \dots, D_{FC_{s_i^u}}] \quad (2)$$

$$D_i = [d_1, d_2, \dots, d_N] \quad (3)$$

For the sake of applying user classification, the sensor readings from a single session  $s_i^u$  are aggregated column-wise into windows  $W$  of intervals  $WI = 20$  s (4). Each element of window vector  $W$  denotes a single window where all feature vectors from a certain interval are stored. The sampling frequency is not uniform and varies based on the user's device and data pipeline processing issues (e.g., packet losses). The aggregates are responsible for converting the data into smaller chunks and for immunizing it from being sampling frequency-susceptible.

$$W = [W_1, W_2, \dots, W_{N_W}], \text{ where } N_W = \lfloor \frac{TS_{s_i^u}}{WI} \rfloor \quad (4)$$

$t_0$  indicates the session starting time and  $t_{D_x}$  indicates the time which passed since  $t_0$  until the creation of a feature vector  $D_x$  (5). The number of all feature vectors in a single window is denoted by  $N_M$  (6).

$$W_j = s_i^u, \text{ where } \left\lfloor \frac{t_{D_x}}{WI} \right\rfloor = j \quad (5)$$

$$W_j = [w_1, w_2, \dots, w_{N_M}] \quad (6)$$

The following Formulas (7)–(9) show how the aggregated features vector  $F$  is created:

$$W_j^k = [w_1(k), w_2(k), \dots, w_{N_M}(k)] \quad (7)$$

$$F = [f_1, f_2, \dots, f_{4N}] \quad (8)$$

$$f_k = \text{aggregate}_{k \bmod 4} (W_j^k) \quad (9)$$

The total length of the aggregated feature vector equals the length of the original feature vector times the number of all the aggregating methods (8)—standard deviation,

arithmetic mean, amplitude, and median—which is denoted by  $4N$ . The formula on which the aggregates are based is shown below (10):

$$\text{aggregate}_{type}(X) = \begin{cases} \frac{1}{P} \sum_{j=1}^P (x_j - \mu)^2, \text{ where } \mu = \frac{1}{P} \sum_{j=1}^P (x_j), & \text{for type} = 0 \\ \frac{1}{P} \sum_{j=1}^P (x_j), & \text{for type} = 1 \\ \max(X) - \min(X) & \text{for type} = 2 \\ \begin{cases} sx_{\frac{P}{2}} & \text{if } P \bmod 2 = 0, \\ \frac{sx_{\frac{|P|}{2}} + sx_{\frac{|P|}{2} + 1}}{2} & \text{if } P \bmod 2 = 1 \end{cases} & \text{where } SX = \text{sort}(X) \text{ for type} = 3 \end{cases} \quad (10)$$

### 2.2. Data Preparation

In order to ensure reliable training and testing datasets, only those users with more than a certain number of unique sessions ( $SN_u > 12$ ), and ones for which a model could be built (undamaged data) were considered. Each viable session had to last for a fixed amount of time  $TS_{min}$  or longer ( $TS_{S_i^u} > TS_{min} = 100$  s) to assure the occurrence of at least  $WC_{min} = 5$  windows lasting for  $WI$ . Each window generated a sub-score; further sub-score processing resulted in score generation. The user choice rule described above is presented in a flow chart below (Figure 2):



Figure 2. The user choice rules pipeline.

From a total of 264 users, only 127 fulfilled the requirements (118 users did not meet data quantity needs and 19 users failed training, resulting in generation of a low-quality model). The exemplary aggregate  $W$  of the sensor readings of 4 random users are presented in Figure 3.

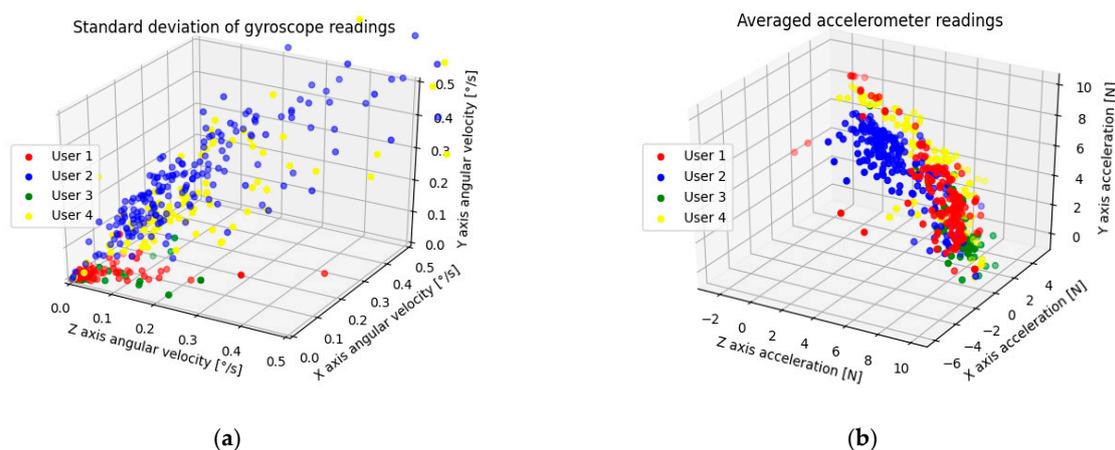


Figure 3. Exemplary user data on average accelerometer readings (a) and standard deviation of gyroscope readings (b) from 20-second intervals.

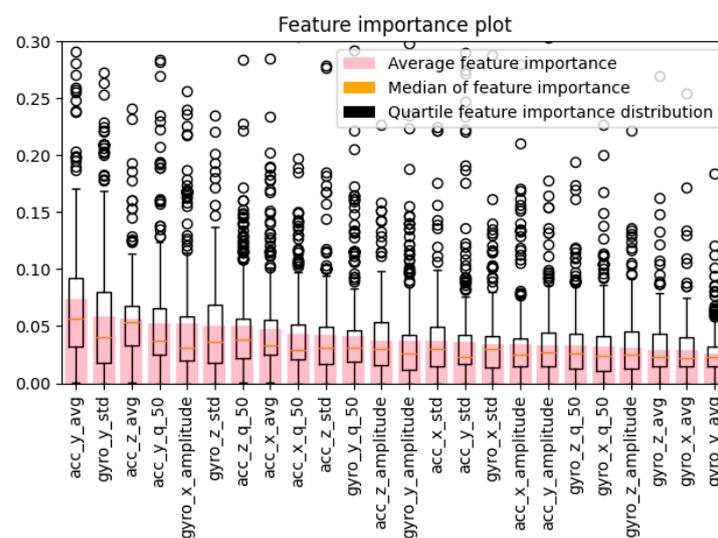
Each session consisted of at least 5 sub-scores, varying from 0 to 1 (indicating the similarity measure coming from the classifier’s output)—to indicate that the behavior was user-like. In order to correctly evaluate the session, the final score—consisting of  $M$  sub-scores—was calculated as their average (11). Whenever the final score exceeded the

user-defined threshold (evaluated based on certain business requirements of the bank), the session was considered fraudulent (12).

$$score = \frac{\sum_{i=1}^M \text{subscore}_i}{M}, \text{ where } M \geq 5 \quad (11)$$

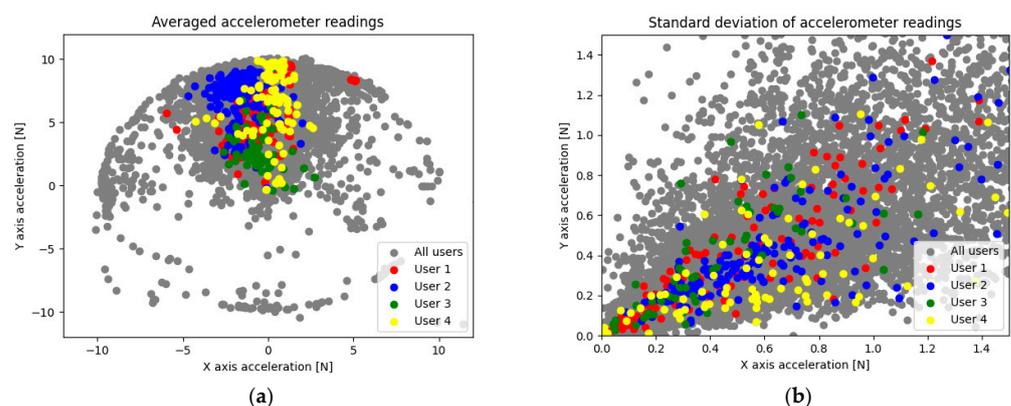
$$\text{assignment} = \begin{cases} 0 (\text{user}), & \text{if } score < thr \\ 1 (\text{fraudster}), & \text{if } score \geq thr \end{cases} \quad (12)$$

To provide in-depth data insight, the feature importance [21–23] for 24 aggregates was calculated. The feature significance was estimated using the XGBoost (eXtreme Gradient Boosting) classification algorithm, with its hyperparameters heuristically optimized. [24]. The boxplot below (Figure 4) shows the feature importance distribution, sorted by descending average importance.

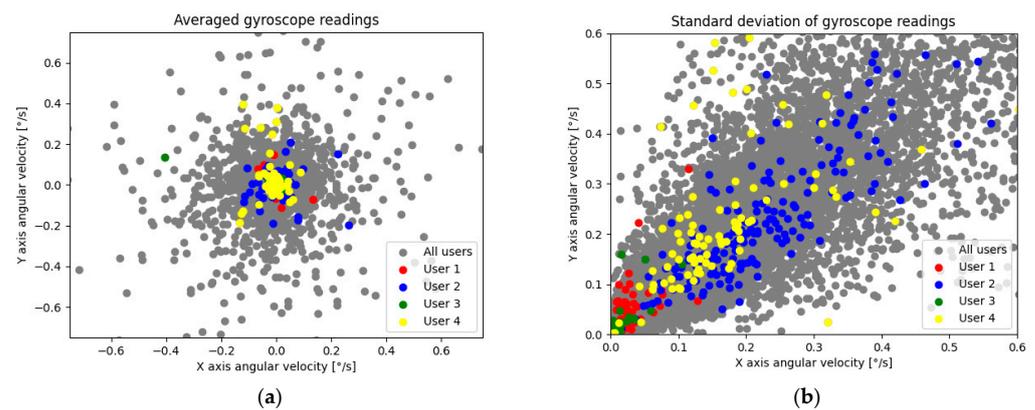


**Figure 4.** Feature importance of aggregated data (averages, standard deviations, medians, and amplitudes) of the  $x$ ,  $y$ , and  $z$  axes of the accelerometer and gyroscope readings.

By analyzing the data presented in Figure 2, we can notice a relationship—the accelerometer data worked best with averages while the gyroscope provided the best diagnostic value (higher average and median feature importance) with standard deviations. This is caused by the nature of the data provided by those sensors; a certain user distinguishability depending on the sensor type and aggregation method can be seen in Figure 5 (accelerometer) and Figure 6 (gyroscope). In order to provide a clearer visualization, 4 randomly chosen users were highlighted.



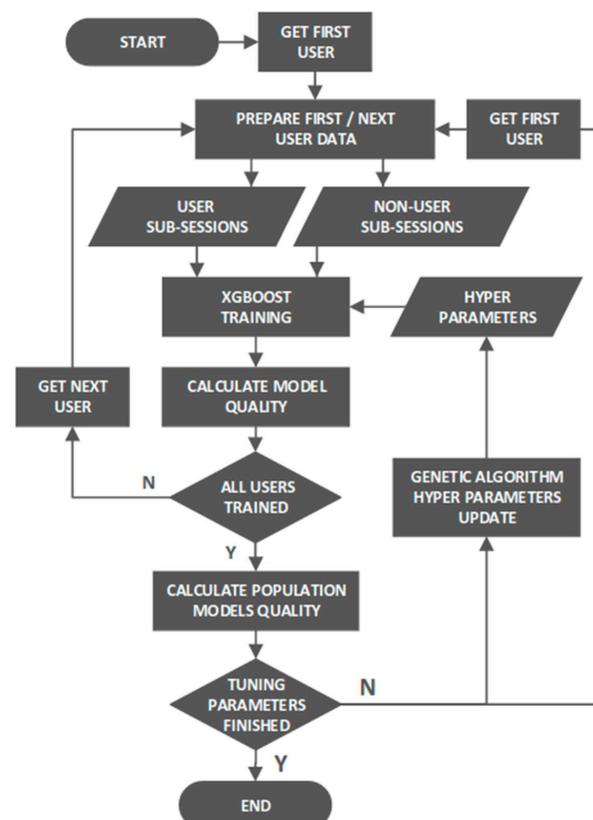
**Figure 5.** Average (a) and standard deviation (b) of accelerometers'  $x$  and  $y$  axes (all users).



**Figure 6.** Average (a) and standard deviation (b) of gyroscopes'  $x$  and  $y$  axes (all users).

### 2.3. XGBoost Training

To train a model, the user's data (sub-sessions) were labeled as an authorized session and the data considering the remaining 126 users were labeled as a fraudulent one. Subsequently, the data set was passed to the XGBoost classifier—the boosting estimator based on decision trees [25–30], in which the trees are expanded to a forest where each estimator is built on the residual value of the previous classification. To reach the best possible model, the XGBoost's hyperparameters were tuned for the whole population [31]. The method used for reaching the best possible model quality is presented in a flow chart in Figure 7. The method trains users stored in a queue with certain hyperparameters set. When all the users are trained, their mean model quality is calculated and is treated as fitness. This procedure was reproduced a fixed number of times with a different set of hyperparameters (which came from the evolutionary algorithm). The best hyperparameter set did not change and was treated as the target set.

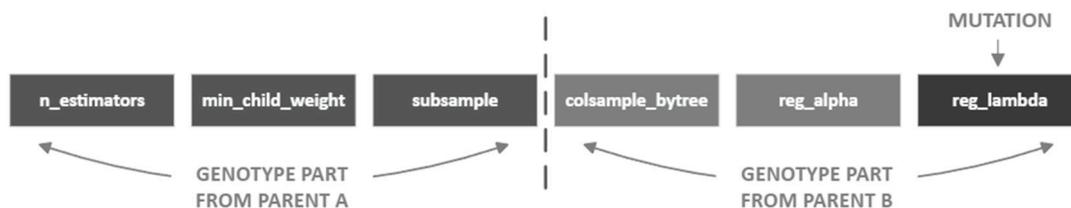


**Figure 7.** Flow chart representing the XGBoost classifier hyperparameter optimization process.

The classifier hyperparameters were optimized using an evolutionary algorithm whose outcome is presented in Table 1 below (non-listed parameters were set to default). The evolutionary algorithm started with a population of 10 randomly chosen values of 6 hyperparameters taken randomly from the uniform distribution, with upper and lower boundaries denoted as “min” and “max” in Table 1) and performed 10 steps of vector crossing (selecting 2 equal subparts of two different hyperparameters’ dictionaries and combining them) and a one-value mutation (randomly reselecting a particular parameter’s value from the specified domain), leaving only the top set of hyperparameter values at each step. Its fitness function was the same as the classifier’s objective function, denoted by Formula (13). The vector that provided the best model qualities among the entire population was kept for further consideration. The exemplary genotype division and mutation are presented in Figure 8.

**Table 1.** XGBoost classifier hyperparameters optimized with the use of a genetic algorithm.

Hyperparameter	Value	Min	Max	Description
n_estimators	220	1	350	Number of weak classifiers (gradient-boosted trees)
min_child_weight	7	2	7	Minimum sum of instance weight (hessian) needed in a child
subsample	0.658	0.1	0.99	Subsample ratio of the training instance
colsample_bytree	0.791	0.5	1.0	Subsample ratio of columns when constructing each tree
reg_alpha	0.415	0.0	2.0	L1 regularization term on weights
reg_lambda	0.566	0.0	2.0	L2 regularization term on weights

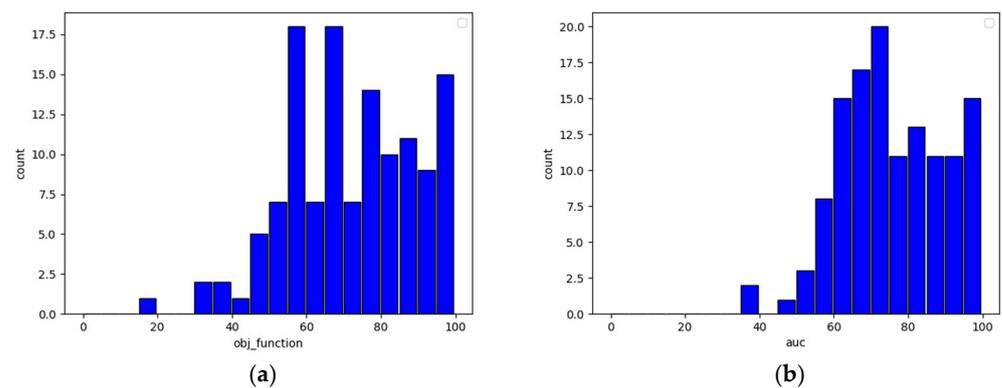


**Figure 8.** Exemplary hyperparameter child genotype from the evolutionary algorithm.

The model validation data set consists of 30% of the total data. To reproduce real-life model usage, as well as to prevent data leakage, the evaluation was run only on the most recent readings. The objective function used for the model training is represented by Formula (13), where TP stands for “true positive” prediction, TN for “true negative”, FP for “false positive”, and FN for “false negative”:

$$fitness = e^{-\frac{\log(\frac{TP}{TP+FN}) + \log(\frac{TN}{TN+FP})}{2}} \quad (13)$$

To illustrate what an undamaged sensor’s model quality looks like, let us examine Figure 9, showing the model quality distribution in terms of the objective function and the receiver operating characteristic’s area under the curve (ROC-AUC) for the entire population. The higher the value of both the objective function and the receiver operating characteristic, the better the model quality is. Any damage done to models (for example, by applying data with missing values for training) will result in the decay of both metrics.



**Figure 9.** Model quality distribution for all users—objective function (a) and ROC-AUC (b).

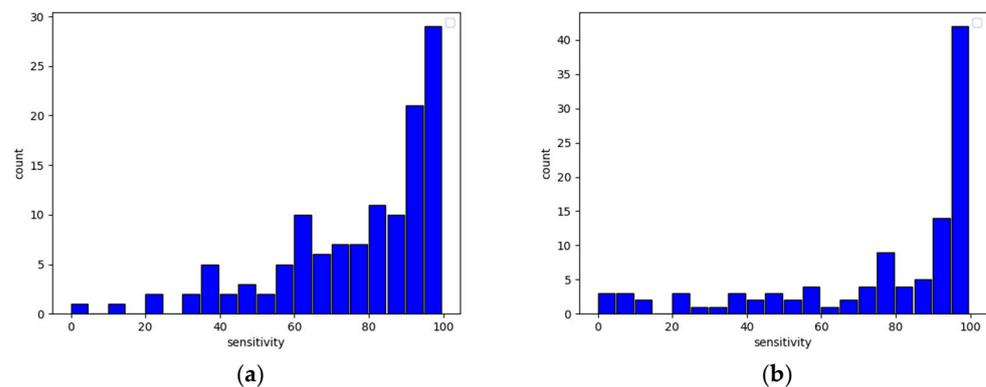
### 3. Results

The experiment investigating the harmful effects of sensor damage on behavioral biometrics model quality was run by replacing certain axis data with zero values. The idea behind conducting such an analysis derives from the necessity of knowing whether the model output is still valid, meaning the authentication provided by the behavioral biometry can be trusted. For minor damage—e.g., one gyroscope axis—the model could still provide valuable information, while deleting the data coming from all the axes may cause the model to become utterly useless. To find those boundaries, the two most common data collection failures were considered: damaging the data from one axis of the sensor and damaging the data from all axes of the sensor. Table 2 provides information on how average models' quality decays due to the zeroing of certain vector values.

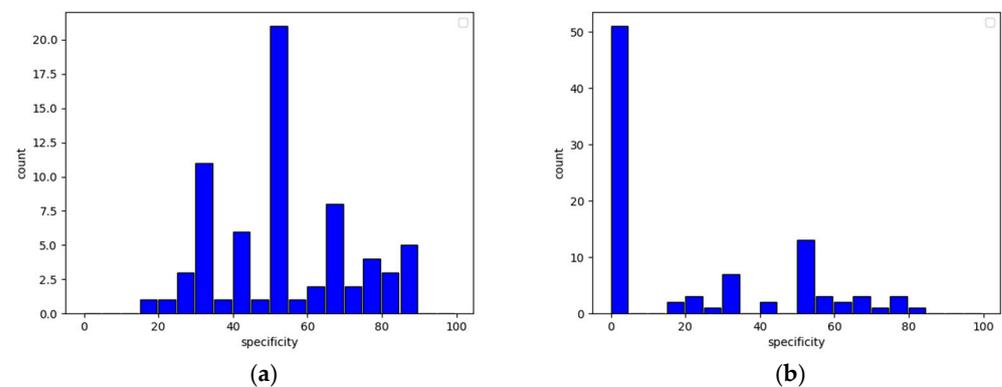
**Table 2.** The influence of sensor data damage on average model quality.

Zeroed Feature	Sensitivity (True Positive Rate)	Specificity (True Negative Rate)	Objective Function	ROC-AUC
None (undamaged data)	0.78	0.74	0.72	0.76
Accelerometer ( <i>x</i> axis)	0.71	0.65	0.56	0.68
Accelerometer ( <i>y</i> axis)	0.70	0.61	0.50	0.66
Accelerometer ( <i>z</i> axis)	0.80	0.43	0.38	0.61
Accelerometer ( <i>x, y, z</i> axes)	0.71	0.33	0.14	0.52
Gyroscope ( <i>x</i> axis)	0.77	0.61	0.58	0.69
Gyroscope ( <i>y</i> axis)	0.78	0.60	0.58	0.69
Gyroscope ( <i>z</i> axis)	0.77	0.68	0.63	0.72
Gyroscope ( <i>x, y, z</i> axes)	0.77	0.45	0.38	0.61

The presented table explicitly proves that the lack of even one axis may severely damage the quality of predictions. What is also worth noticing is that the true negative rate heavily dropped due to data changes while the true positive rate remained almost the same. Such behavior causes models to produce an increased amount of false positive output, and this leads to overwhelming of the system with false fraudster alerts (correct user detection is in most cases the same though). The histograms presented in Figure 10 (sensitivity) and Figure 11 (specificity) clearly show this relationship—the higher the specificity, the less likely the model is to classify a fraudster as a user, and the higher the sensitivity, the more reliable user detection becomes.



**Figure 10.** User sensitivity histogram with undamaged data (a) and data without accelerometer axis z (b).



**Figure 11.** User specificity histogram with undamaged data (a) and data without accelerometer axis z (b).

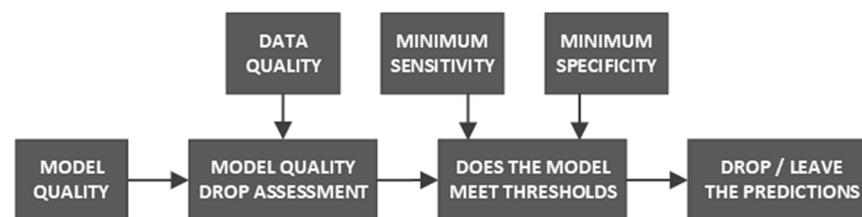
#### 4. Discussion

The presented paper investigates commonly occurring issues with mobile sensor data used for behavioral biometry. To properly approach faulty data handling (while detecting, e.g., zero values on a certain axis), it is necessary to know the model quality drop for particular-axis damage. The results of the analysis showed that when certain sensor axis data is missing, then vectors used for user authentication may cause a major drop in model accuracy. What is worth noticing and what derives from both the feature importance plot and from the damage influence table is the fact that the most harmful factor for user identification is the loss of accelerometer readings (especially the z axis—34 p.p. compared to the original objective function). Such damage causes the same ROC-AUC drop as losing all the gyroscope's readings (ROC-AUC lowered by 0.15). This is caused by the fact that the accelerometer's z axis provides user-distinctive data—it is highly responsible for indicating at what position the smartphone or tablet is held by the user and how his/her grip changes over time. On the other hand, when it comes to the gyroscope's z axis, this parameter holds the lowest amount of user-distinctive data (objective function only dropping by 9 p.p.). This may be caused by the fact that there exists no substantial angular movement in this axis, or because all the movements are repeatable over the entire population.

#### 5. Conclusions

The true negative ratio is the most affected metric, and this means that the model's ability to correctly distinguish a user from the rest of the population will not work well—the model will classify user sessions as fraudulent ones. Such behavior will heavily deteriorate security systems by setting off false alerts. To prevent this, we can change the classification threshold level by increasing specificity at the cost of sensitivity; yet, this approach will not

improve the total model accuracy. These minor specificity drops should be compensated for by increasing the classification threshold by a predefined factor (which will result in increases in objective function), while higher drops should raise a flag indicating that the evaluation score is invalid. A different approach to dealing with lower-quality models rests with decreasing their weights in an authentication system. Usually, authorization via behavioral biometry uses several models that measure several types of activity—if we can assess a certain model’s quality for certain data, we can lower the contribution provided by this model in generating the final score. Yet another way of evading quality drop is using different classifiers—those less susceptible to data damage or those using data preprocessing methods that immunize models against data damage. We can as well think of modifying the classification pipeline—whenever nothing but zero values are present on a specific axis, we can decide whether to evaluate the session or to skip the evaluation (for example, the authentication process is run only if sensitivity did not drop below 70% and specificity did not drop below 60% after introducing certain-axis damage). The additional session evaluation block would take model statistics, as well as a data structure, and assess the prediction reliability (Figure 12).



**Figure 12.** Prediction quality assessment pipeline proposal.

What may be worth noticing is that further examination of the model quality drop can be used to estimate numerous device/user-related issues, e.g., to identify device damage. If the user did not show any symptoms of classification problems, and after some time generates numerous false alerts, we may conclude that the sensors do not work correctly anymore or that there are different issues (e.g., user illness or malware attack). These assumptions, however, require further studies.

Future work will mostly be focused on building high-quality damaged-sensor handling pipelines. In case of damaged data occurrence, we must be able to quickly assess the quality of incoming information and to find an efficient way of detecting session hijacking, even if the most valuable information is lost due to data collection errors.

Additional study directions to take should be focused on immunizing the system to erroneous data as well as improving the system’s overall quality by introducing novel noise-resistant classifiers and data processors. What should also be kept in mind is the fact that different devices (keyboard or mouse) and different features may not respond similarly to what the analysis has shown. Quality drops caused by missing data be separately examined and countermeasures introduced to them may differ from the ones implemented for mobile devices.

**Author Contributions:** Conceptualization, P.R.; methodology, P.R. and D.G.; software, P.R., T.B. and P.S.; validation, P.R. and D.G.; formal analysis, P.R., T.B. and D.G.; investigation, P.R.; resources, P.R.; data curation, P.R. and P.S.; writing—original draft preparation, P.R.; writing—review and editing, T.B. and D.G.; visualization, P.R.; supervision, D.G.; project administration, T.B.; funding acquisition, P.R. and T.B. and D.G. All authors have read and agreed to the published version of the manuscript.

**Funding:** The article was carried out under the project no. POIR.01.01.01-00-0082/20 “Development and verification of new methods of user authentication based on behavioral biometrics and machine learning methods”, co-financed by the European Regional Development Fund under Measure 1.1 of the Operational Programme Smart Growth 2014-2020, and was partially supported by Statutory Research for Young Researchers funds and partially by Statutory Activity from the Faculty of Automatic Control, Electronics and Computer Science, Silesian University of Technology, Gliwice, Poland.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

## References

1. Internet Crime Report. Internet Crime Complaint Center (IC3). 2020. Available online: [https://www.ic3.gov/Media/PDF/AnnualReport/2020\\_IC3Report.pdf](https://www.ic3.gov/Media/PDF/AnnualReport/2020_IC3Report.pdf) (accessed on 1 November 2022).
2. Desolda, G.; Ferro, L.S.; Marrella, A.; Catarci, T.; Costabile, M.F. Human Factors in Phishing Attacks: A Systematic Literature Review. *ACM Comput. Surv.* **2022**, *54*, 173. [\[CrossRef\]](#)
3. Alkhalil, Z.; Hewage, C.; Nawaf, L.; Khan, I. Phishing Attacks: A Recent Comprehensive Study and a New Anatomy. *Front. Comput. Sci.* **2021**, *3*, 563060. [\[CrossRef\]](#)
4. Shahbaznezhad, H.; Kolini, F.; Rashidirad, M. Employees' Behavior in Phishing Attacks: What Individual, Organizational, and Technological Factors Matter? *J. Comput. Inf. Syst.* **2021**, *61*, 539–550. [\[CrossRef\]](#)
5. Aneke, J.; Ardito, C.; Desolda, G. Help the User Recognize a Phishing Scam: Design of Explanation Messages in Warning Interfaces for Phishing Attacks. In Proceedings of the International Conference on Human-Computer Interaction, Málaga, Spain, 22–24 September 2021; Springer: Cham, Switzerland, 2021; pp. 403–416. [\[CrossRef\]](#)
6. Majumder, S.; Deen, M.J. Smartphone Sensors for Health Monitoring and Diagnosis. *Sensors* **2019**, *19*, 2164. [\[CrossRef\]](#)
7. Falkowski-Gilski, P. On the Consumption of Multimedia Content Using Mobile Devices: A Year to Year User Case Study. *Arch. Acoust.* **2020**, *45*, 321–328. [\[CrossRef\]](#)
8. Teh, P.S.; Teoh, A.B.J.; Yue, S. A Survey of Keystroke Dynamics Biometrics. *Sci. World J.* **2013**, *2013*, 408280. [\[CrossRef\]](#)
9. Stylios, I.; Kokolakis, S.; Thanou, O.; Chatzis, S. Behavioral biometrics & continuous user authentication on mobile devices: A survey. *Inf. Fusion* **2021**, *66*, 76–99. [\[CrossRef\]](#)
10. Sahdev, S.L.; Singh, S.; Kaur, N.; Siddiqui, L. Behavioral Biometrics for Adaptive Authentication in Digital Banking—Guard Against Flawless Privacy. In Proceedings of the 2021 International Conference on Innovative Practices in Technology and Management (ICIPTM), Noida, India, 17–19 February 2021; pp. 261–265. [\[CrossRef\]](#)
11. Almalki, S.; Assery, N.; Roy, K. An Empirical Evaluation of Online Continuous Authentication and Anomaly Detection Using Mouse Clickstream Data Analysis. *Appl. Sci.* **2021**, *11*, 6083. [\[CrossRef\]](#)
12. Ometov, A.; Bezzateev, S.; Mäkitalo, N.; Andreev, S.; Mikkonen, T.; Koucheryavy, Y. Multi-Factor Authentication: A Survey. *Cryptography* **2018**, *2*, 1. [\[CrossRef\]](#)
13. Chalhoub, G.; Flechais, I.; Nthala, N.; Abu-Salma, R.; Tom, E. Factoring User Experience into the Security and Privacy Design of Smart Home Devices: A Case Study. In Proceedings of the Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, 25–30 April 2020. [\[CrossRef\]](#)
14. Matsuoka, K.; Irvan, M.; Kobayashi, R.; Yamaguchi, R.S. A Score Fusion Method by Neural Network in Multi-Factor Authentication. In Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy, Orleans, LA, USA, 16–18 March 2020. [\[CrossRef\]](#)
15. Miyazawa, A.; Thao, T.P.; Yamaguchi, R.S. Multi-factor Behavioral Authentication Using Correlations Enhanced by Neural Network-based Score Fusion. In Proceedings of the 2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 8–11 January 2022; pp. 569–577. [\[CrossRef\]](#)
16. Rocha, R.; Carneiro, D.; Costa, R.; Analide, C. Continuous Authentication in Mobile Devices Using Behavioral Biometrics. In Proceedings of the Ambient Intelligence—Software and Applications—10th International Symposium on Ambient Intelligence, Ávila, Spain, 26–28 June 2019; Novais, P., Lloret, J., Chamoso, P., Carneiro, D., Navarro, E., Omatu, S., Eds.; Springer International Publishing: Cham, Switzerland, 2020; Volume 1006, pp. 191–198. [\[CrossRef\]](#)
17. Stragapede, G.; Vera-Rodriguez, R.; Tolosana, R.; Morales, A. BehavePassDB: Public Database for Mobile Behavioral Biometrics and Benchmark Evaluation. *Pattern Recognit.* **2023**, *134*, 109089. [\[CrossRef\]](#)
18. Falkowski-Gilski, P.; Stefański, J. Quality Expectations of Mobile Subscribers. *J. Telecommun. Inf. Technol.* **2015**, *1*, 15–19.
19. Chyzhevska, M.; Romanovska, N.; Ramskyi, A.; Venger, V.; Obushnyi, M. Behavioral Biometry as a Cyber Security Tool. In Proceedings of the Cybersecurity Providing in Information and Telecommunication Systems II, Kyiv, Ukrain, 26 October 2021; Volume 2, pp. 88–97.
20. Falkowski-Gilski, P.; Stefański, J. Android OS: A Review. *Tem J.* **2015**, *4*, 116–120.
21. Cenggoro, T.W.; Mahesworo, B.; Budiarto, A.; Baurley, J.; Suparyanto, T.; Pardamean, B. Features Importance in Classification Models for Colorectal Cancer Cases Phenotype in Indonesia. *Procedia Comput. Sci.* **2019**, *157*, 313–320. [\[CrossRef\]](#)
22. Chen, C.; Shi, H.; Jiang, Z.; Salhi, A.; Chen, R.; Cui, X.; Yu, B. DNN-DTIs: Improved drug-target interactions prediction using XGBoost feature selection and deep neural network. *Comput. Biol. Med.* **2021**, *136*, 104676. [\[CrossRef\]](#)

23. Muslim, M.A.; Dasril, Y. Company bankruptcy prediction framework based on the most influential features using XGBoost and stacking ensemble learning. *Int. J. Electr. Comput. Eng. (IJECE)* **2021**, *11*, 5549–5557. [[CrossRef](#)]
24. Putatunda, S.; Rama, K. A Comparative Analysis of Hyperopt as Against Other Approaches for Hyper-Parameter Optimization of XGBoost. In Proceedings of the 2018 International Conference on Signal Processing and Machine Learning, Shanghai, China, 28–30 November 2018; pp. 6–10. [[CrossRef](#)]
25. Ogunleye, A.A.; Wang, Q.-G. XGBoost Model for Chronic Kidney Disease Diagnosis. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2020**, *17*, 2131–2140. [[CrossRef](#)]
26. Dhaliwal, S.S.; Nahid, A.-A.; Abbas, R. Effective Intrusion Detection System Using XGBoost. *Information* **2018**, *9*, 149. [[CrossRef](#)]
27. Jing, X.; Zou, Q.; Yan, J.; Dong, Y.; Li, B. Remote Sensing Monitoring of Winter Wheat Stripe Rust Based on mRMR-XGBoost Algorithm. *Remote Sens.* **2022**, *14*, 756. [[CrossRef](#)]
28. Sanders, W.; Li, D.; Li, W.; Fang, Z.N. Data-Driven Flood Alert System (FAS) Using Extreme Gradient Boosting (XGBoost) to Forecast Flood Stages. *Water* **2022**, *14*, 747. [[CrossRef](#)]
29. Liu, Y.; Wang, H.; Fei, Y.; Liu, Y.; Shen, L.; Zhuang, Z.; Zhang, X. Research on the Prediction of Green Plum Acidity Based on Improved XGBoost. *Sensors* **2021**, *21*, 930. [[CrossRef](#)]
30. Shahbazi, Z.; Byun, Y.-C. Knowledge Discovery on Cryptocurrency Exchange Rate Prediction Using Machine Learning Pipelines. *Sensors* **2022**, *22*, 1740. [[CrossRef](#)]
31. Chen, J.; Zhao, F.; Sun, Y.; Yin, Y. Improved XGBoost model based on genetic algorithm. *Int. J. Comput. Appl. Technol.* **2020**, *62*, 240. [[CrossRef](#)]