Generating Datasets Based on the HuMIdb Dataset for Risk-based User Authentication on Smartphones

Maria Papaioannou
Instituto de Telecomunicações
Aveiro, Portugal
Faculty of Engineering and Science,
University of Greenwich
Chatham Maritime, UK
m.papaioannou@av.it.pt

Georgios Zachos
Instituto de Telecomunicações
Aveiro, Portugal
Faculty of Engineering and Science,
University of Greenwich
Chatham Maritime, UK
g.zachos@av.it.pt

Georgios Mantas
Instituto de Telecomunicações
Aveiro, Portugal
Faculty of Engineering and Science,
University of Greenwich
Chatham Maritime, UK
gimantas@av.it.pt

Aliyah Essop
Faculty of Engineering and Science,
University of Greenwich
Chatham Maritime, UK
a.b.essop@greenwich.ac.uk

Abdulkareem Karasuwa
Faculty of Computing, Engineering and
Science, University of South Wales
Pontypridd, UK
abdulkareem.karasuwa@southwales.ac.uk

Jonathan Rodriguez
Instituto de Telecomunicações
Aveiro, Portugal
Faculty of Computing, Engineering and
Science, University of South Wales
Pontypridd, UK
jonathan@av.it.pt

Abstract— **User authentication acts as the first line of defense verifying the identity of a mobile user, often as a prerequisite to allow access to resources in a mobile device. Risk-based user authentication based on behavioral biometrics appears to have the potential to increase mobile authentication security without sacrificing usability. Nevertheless, in order to precisely evaluate classification and/or novelty detection algorithms for risk-based user authentication, it is of utmost importance to make use of quality datasets to train and test these algorithms. To the best of our knowledge, there is a lack of up-to-date, representative and comprehensive datasets that are publicly available to the research community for effective training and evaluation of classification and/or novelty detection algorithms suitable for risk-based user authentication. Toward this direction, in this paper, the aim is to provide details on how we generate datasets based on HuMIdb dataset for training and testing classification and novelty detection algorithms for risk-based adaptive user authentication. The HuMIdb dataset is the most recent and publicly available dataset for behavioral user authentication.**

Keywords—mobile device security, risk-based adaptive user authentication, dataset generation, record selection

I. Introduction

Authentication acts as the first line of defense verifying the identity of a user, process, or device, often as a prerequisite to allowing access to resources in an information system. In smartphone devices, user authentication is essential to protect smartphone users' data privacy [1]–[6]. Risk-based user authentication based on behavioral biometrics appears to have the potential to increase mobile authentication security without sacrificing usability [7], [8]. In particular, risk-based user authentication mechanisms have been proposed in the literature to dynamically verify the identity of a user during their entire interaction with the smartphone device, based on a risk score calculated in real-time. In this way, it is suggested that risk-based user authentication mechanisms enhance the reliability of whole authentication process without interrupting the smartphone user's regular activity [9]. In our previous publications [10]–[13], we have presented: (i) a thorough review of related work on user authentication solutions for public safety and mobile devices, (ii) the security vs. usability challenge, (iii) the concept of the risk-based user authentication, as well as (iv) the HuMIdb dataset, which, to the best of our knowledge, is the most recent and publicly available dataset for behavioral user authentication [14], [15], as well as (v) a comprehensive work [13] on the design of a risk-based adaptive user authentication mechanism for mobile devices that comprises a novel secure and usable user authentication solution ensuring continuous user authentication behind-the-scenes and invisible to the user.

On top of that, in [13], we also modified effectively the ''HuMIdb'' dataset files, and we trained and tested the following most popular classification algorithms for risk-based authentication: K-NN, Decision Trees (DT), Support Vector Machine (SVM), and Naïve Bayes (NB) over the ''HuMIdb'' dataset using ten-fold cross validation. Nevertheless, the evaluation results for these classification algorithms demonstrated impact of overfitting and therefore, we considered the concept of novelty detection to overcome this challenge. Thus, we tested and evaluated the following novelty detection algorithms: Local Outlier Factor (LOF), one-class Support Vector Machine (OneClassSVM), and KNN_average (i.e., KNN configured properly for novelty detection). All of them demonstrated a high performance for the same part of the ''HuMIdb'' dataset that was also used for the evaluation of the classification algorithms, when applied to distinguish between a known legitimate user and an unknown malicious user. To the best of our knowledge, this was the first time that novelty detection algorithms have been considered for risk-based adaptive user authentication demonstrating promising results.

As it can be observed, to precisely evaluate classification and/or novelty detection algorithms for the proposed risk-

The research work leading to this publication has received funding from the European Union's Horizon 2020 Research and Innovation programme under grant agreement H2020-MSCA-RISE-2019-eBORDER-872878.

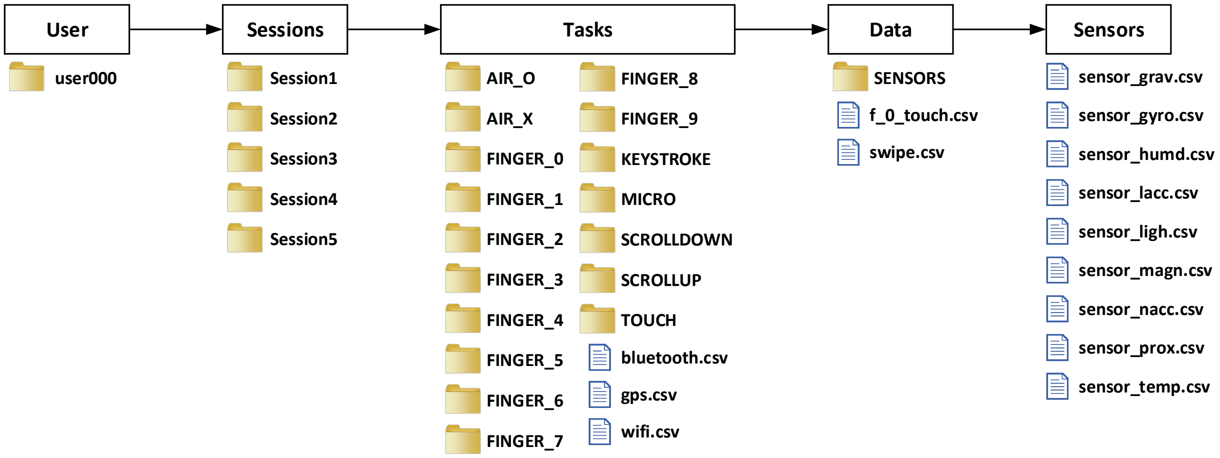


Figure 1: An Overview of the Structure of the HuMIdb Dataset.

based user authentication, it is of utmost important to make use of quality datasets to train and test the classification and/or novelty detection algorithms. Nevertheless, there is a lack of up-to-date, representative and comprehensive datasets that are publicly available to the research community and considered as benchmark datasets for effective training and evaluation of classification and/or novelty detection algorithms suitable for risk-based user authentication. This lack of benchmark risk-based user authentication specific datasets constitutes a significant research challenge that should be addressed so as to develop more accurate and efficient risk-based user authentication for mobile devices. Toward this direction, in this paper, our aim is to provide details on how we generate datasets based on HuMIdb dataset for training and testing classification and novelty detection algorithms for risk-based adaptive user authentication. To the best of our knowledge the HuMIdb dataset is the most recent and publicly available dataset for behavioral user authentication [14], [15].

Following the Introduction, the rest of the paper is organized as follows. Section II presents the structure of HuMIdb dataset, while Section III provides details on how we generated datasets based on the HuMIdb dataset for risk-based adaptive user authentication. Finally, the paper is concluded in Section IV.

II. HuMIdb Dataset Structure

A. Data acquisition

The data provided by the HuMIdb dataset (Human Mobile Interaction database) were captured by 14 sensors that are typically integrated in a mobile device, namely accelerometer, linear accelerometer, GPS, WiFi, magnetometer, gyroscope, gravity, orientation, light, proximity, keystroke, touchscreen, Bluetooth, and microphone, throughout natural human-mobile interaction performed by more than 600 smartphone users [14], [15]. For the data acquisition, the authors in [14], [15] created an Android application that collects sensor signals when smartphone users perform eight effortless tasks with their own devices and without any guidance or supervision (i.e., the users could be being indoors or outdoors, at daytime or night, walking, sitting, or standing, etc.).

In particular, the designed tasks consisted of the following activities: a) swiping up, b) swiping down, c) keystroking, d) tapping and double tapping, e) cross hand gesturing, f) circle hand gesturing, g) finger handwriting, and h) voice recording. Their acquisition protocol involved 5 sessions with at least 1 day gap among them (i.e., the minimum time between one user finishes a session and the next time the app allows to have the next session). When the smartphone user starts a task, the application displays a brief pop-up message giving instructions on the procedure how to complete the task. On top of that, the app also captured the smartphone orientation (e.g., landscape/portrait), the screen size, resolution, the date when the session was captured, as well as the model of the smartphone device.

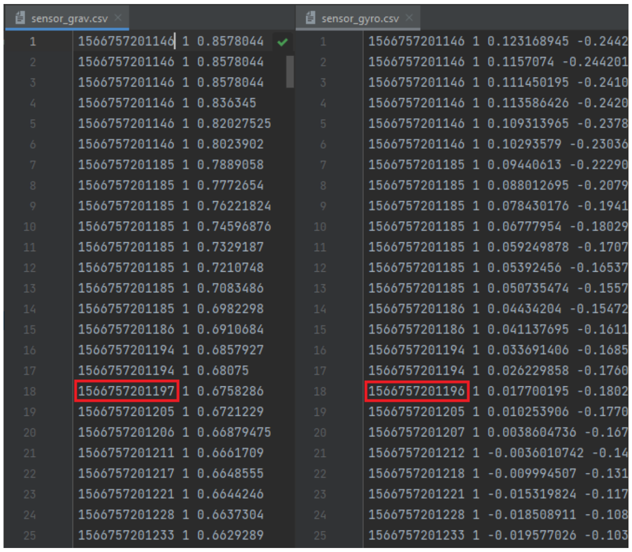
The authors launched their developed Android application on Google Play Store and advertised it in their research web site as well as and numerous research mailing lists. Afterwards, participants were self-selected world widely, establishing a diverse network of people compared to previous state-of-the-art mobile databases. It is important to highlight that all captured data have been anonymized with prior participant consent according to the GDPR (General Data Protection Regulation) and then were stored in private servers [14], [15].

B. HuMIdb dataset structure

The data provided by HuMIdb dataset are stored in nested folders with the ID number to identify each smartphone user´s folder. The structure of HuMIdb nested folders is depicted in Fig. 1. As we can observe, inside the user´s folder, there are five ``session´´ sub-folders corresponding to the five different sessions the user has completed. Each ``session´´ sub-folder contains a set of ``task´´ sub-folders (e.g., keystroking, swiping up and down, tapping and double tapping) and three CSV files with the Bluetooth, WiFi and GPS data signals acquired during the given session. Besides that, each ``task´´ sub-folder includes a ``sensors´´ sub-folder including data from the various sensors required for the particular task. Additionally, each ``task´´ sub-folder contains CSV files involving data related to the touch gestures of the user during the particular task. depicts the structure of the nested folders and files of the HuMIdb dataset.

III. Generating Datasets

The purpose of the generation of datasets based on the HuMIdb dataset is the fusion of the various categories of



```
1566757201146 1 0.8578044
1566757201146 1 0.8578044
1566757201146 1 0.8578044
1566757201146 1 0.836345
1566757201146 1 0.82027525
1566757201146 1 0.8023902
1566757201185 1 0.7889058
1566757201185 1 0.7772654
1566757201185 1 0.76221824
1566757201185 1 0.74596876
1566757201185 1 0.7329187
1566757201185 1 0.7210748
1566757201185 1 0.7083486
1566757201185 1 0.6982298
1566757201186 1 0.6910684
1566757201194 1 0.6857927
1566757201194 1 0.68075
1566757201197 1 0.6758286
1566757201205 1 0.6721229
1566757201206 1 0.66879475
1566757201211 1 0.6661709
1566757201217 1 0.6648555
1566757201221 1 0.6644246
1566757201228 1 0.6637304
1566757201233 1 0.6629289
```

```
1566757201146 1 0.123168945 -0.2442
1566757201146 1 0.1157074 -0.244201
1566757201146 1 0.111450195 -0.2410
1566757201146 1 0.113586426 -0.2420
1566757201146 1 0.109313965 -0.2378
1566757201146 1 0.10293579 -0.23036
1566757201185 1 0.09440613 -0.22290
1566757201185 1 0.088012695 -0.2079
1566757201185 1 0.078430176 -0.1941
1566757201185 1 0.06777954 -0.18029
1566757201185 1 0.059249878 -0.1707
1566757201185 1 0.05392456 -0.16537
1566757201185 1 0.050735474 -0.1557
1566757201186 1 0.04434204 -0.15472
1566757201186 1 0.041137695 -0.1611
1566757201194 1 0.033691406 -0.1685
1566757201194 1 0.026229858 -0.1760
1566757201196 1 0.017700195 -0.1802
1566757201205 1 0.010253906 -0.1770
1566757201207 1 0.0038604736 -0.167
1566757201212 1 -0.0036010742 -0.14
1566757201218 1 -0.009994507 -0.131
1566757201221 1 -0.015319824 -0.117
1566757201228 1 -0.018508911 -0.108
1566757201233 1 -0.019577026 -0.103
```

Figure 2: "sensor_grav.csv" and "sensor_gyro.csv" Sensor Files.

data that are present in many different dataset folders and files. The goal is to produce dataset records related to the interaction between human and mobile phone device and containing information from as many as possible different dataset files. The process of generating datasets based on the HuMIdb dataset can be divided in the following steps: (a) analyzing and understanding the content of the available data, (b) determining which data could be fused, and (c) combining the various types of data.

A. Analysis of the available data

Fig. 1 showed an overview of the folders and files of the HuMIdb dataset. At this point, it is important to explore the dataset files in the various folders and check the content of these files. We started with the first user (i.e., "user000" folder), its first session (i.e., "Session1" folder) and the circle hand gesturing task (i.e., "AIR_O" folder). We checked the content of the different sensor files (i.e., "sensor_grav.csv", "sensor_gyro.csv", "sensor_humd.csv", "sensor_lacc.csv", "sensor_ligh.csv", "sensor_magn.csv", "sensor_nacc.csv", "sensor_prox.csv", "sensor_temp.csv") in the "Sensors" folder as well as the content of the touch gesture file (i.e., "swipe.csv" file) and the microphone (i.e., "micro.npy") for the circle hand gesturing task. Table I. describes the features of the sensor files along with the features of the touch gesture files.

TABLE I. FEATURES OF THE SENSOR FILES ALONG WITH THE FEATURES OF THE TOUCH GESTURE FILES

Features	Description
	Features of the Sensor files
	Features of "sensor_grav.csv"
1	Timestamp in milliseconds.
2	Orientation of the mobile device (portrait=1, landscape=0).
3	Gravity data.
	Features of "sensor_gyro.csv"
1	Timestamp in milliseconds.
2	Orientation of the mobile device (portrait=1, landscape=0).
3	X axis data of the gyroscope of the mobile device.
4	Y axis data of the gyroscope of the mobile device.
5	Z axis data of the gyroscope of the mobile device.
	Features of "sensor_humd.csv"
1	Timestamp in milliseconds.
2	Orientation of the mobile device (portrait=1, landscape=0).
3	Humidity data.
	Features of "sensor_lacc.csv"
1	Timestamp in milliseconds.
2	Orientation of the mobile device (portrait=1, landscape=0).
3	X axis data of the linear accelerometer of the mobile device.
4	Y axis data of the linear accelerometer of the mobile device.
5	Z axis data of the linear accelerometer of the mobile device.
	Features of "sensor_ligh.csv"
1	Timestamp in milliseconds.
2	Orientation of the mobile device (portrait=1, landscape=0).
3	Light level data.
	Features of "sensor_magn.csv"
1	Timestamp in milliseconds.
2	Orientation of the mobile device (portrait=1, landscape=0).
3	X axis data of the magnetometer of the mobile device.
4	Y axis data of the magnetometer of the mobile device.
5	Z axis data of the magnetometer of the mobile device.
	Features of "sensor_nacc.csv"
1	Timestamp in milliseconds.
2	Orientation of the mobile device (portrait=1, landscape=0).
3	X axis data of the accelerometer of the mobile device.
4	Y axis data of the accelerometer of the mobile device.
5	Z axis data of the accelerometer of the mobile device.
	Features of "sensor_prox.csv"
1	Timestamp in milliseconds.
2	Orientation of the mobile device (portrait=1, landscape=0).
3	Proximity data.
	Features of "sensor_temp.csv"
1	Timestamp in milliseconds.
2	Orientation of the mobile device (portrait=1, landscape=0).
3	Temperature data.
	Features of the Touch Gesture Files
1	Timestamp in milliseconds.
2	Orientation of the mobile device (portrait=1, landscape=0).
3	X coordinate data regarding to the use of the touchscreen.
4	Y coordinate data regarding to the use of the touchscreen.

Features	Description
5	Pressure data regarding the use of the touchscreen.
6	Action data regarding the touchscreen (finger_down= 0, movement= 2, finger_up= 1).

The first observation is that both the sensors files and the touch gesture file contain records with the only common features the following: i) "feature 1" representing the timestamp in milliseconds (ms), and ii) "feature 2" representing the orientation of the mobile device (i.e., portrait or landscape).

Secondly, two sensor files (e.g., "sensor_grav.csv", "sensor_gyro.csv") are placed one beside the other. The first 25 lines are shown in Fig. 2. We can see that, in "sensor_grav.csv" file, multiple records share the same value for the first feature (i.e., "timestamp" feature) while being different in the values of the rest of the features. In addition, the "timestamp" values are not the exact same in both sensor files. For instance, the "timestamp" value in line 18 in the "sensor_grav.csv" file is not present as a value in any of the records of the "sensor_gyro.csv" file. Vice-versa, the "timestamp" value in line 18 in the "sensor_gyro.csv" file is not present as a value in any of the records of the "sensor_grav.csv" file.

Furthermore, it is possible to observe that some sensor files (i.e., "sensor_humd.csv", "sensor_temp.csv") are empty, containing no data. Next, the "sensor_prox.csv" file contains only three records, and by considering the three "timestamp" values (i.e., "1566757201185", "1566757214291", "1566757214715"), we understand that the records refer only to three moments (i.e., 1185 ms, 14291 ms, 14715 ms) during the specific task (i.e., the "AIR_O" task in this case). Thus, this specific "sensor_prox.csv" file contains little information.

Moreover, we continue the exploration of the content of the sensor files and the touch gesture files in the remaining tasks of the first session (i.e., "Session1") of the first user (i.e., "user000") and we observed trends similar to the ones described before.

After that, we explored the information of the "bluetooth.csv", "gps.csv" and "wifi.csv" files of the "Session1" folder. The "bluetooth.csv" and "wifi.csv" files contain both alphanumerical and numerical features and by observing the values of the features of various records, it was understood that the values of the "bluetooth.csv" file and the "wifi.csv" file highly depend on the specific mobile device of the user and/or the WiFi network the user is connected to. In addition, the records of the "gps.csv" contain mostly geolocation information related to the user. Finally, we checked the content of the remaining sessions (i.e., sessions 2 to 5) of the first user, and we observed similar trends in the remaining sessions.

B. Selection of data to be fused

After understanding the content of the various folders and file of the "HuMIdb" dataset, the next step is to select which data and how they could be fused. Since the dataset relates to the interaction between a human and a mobile device, it is essential to focus on the type of data where this interaction is more pronounced. This type of data is the touch gesture data included in the touch gesture files (e.g., "f_0_touch.csv", "swipe.csv") as shown in Fig. 1. This is because the normal method used by humans to interact with their mobile phone devices is through the touchscreen.

Thus, our intention is to enhance the information contained in every record of the touch gesture files, by leveraging the information included in all the other files of the "HuMIdb" dataset. At this point, it is worthwhile mentioning that the records related to the humidity, proximity, temperature, Bluetooth, WiFi, and micro files were not used to expand the information of the touch gesture records. This was because these files: (i) suffered from lack of records, (ii) contained records with alphanumeric values that did not allow further processing, or (iii) contained records that were closely related to specific device characteristics (e.g., MAC address) whose values were always fixed.

Therefore, data selected to be fused include the following: the records of the touch gesture file (e.g., "swipe.csv", "f_0_touch.csv") with the records in the following five sensor files: (a) the "sensor_nacc.csv" files containing records related to the accelerometer sensor of the mobile device, (b) the "sensor_grav.csv" files containing records related to the gravity sensor of the mobile device, (c) the "sensor_gyro.csv" files containing records related to the gyroscope sensor of the mobile device, (d) the "sensor_lacc.csv" files containing records related to the linear accelerometer sensor of the mobile device, and (e) the "sensor_magn.csv" files containing records related to the magnetometer sensor of the mobile device.

C. Data fusion

The steps of the fusion process are described for the touch gesture file and the sensor files that are present in the "AIR_O" task folder inside the "Session1" session folder of the "user000" user folder. Exactly the same steps can be used to fuse the records of the touch gesture files with the records of the five sensor files related to any other task (e.g., "AIR_X", "FINGER_0") of other sessions (e.g., "Session2", "Session3") of other users (e.g., "user001"). The fusion process consists of the following steps:

Step 1: An empty list of records (i.e., "dtst_List") is initialized and the records of the touch gesture file (i.e., "swipe.csv") are added, without being modified, to the empty list of records.

Step 2: An empty list of sensor files (i.e., "sensor_files" list) is initialized and the names of the sensor files (i.e., "sensor_nacc.csv", "sensor_grav.csv", "sensor_gyro.csv", "sensor_lacc.csv", "sensor_magn.csv") in the "Sensors" folder, whose records are to be fused with the touch gesture records, are added to the empty "sensor_files" list.

Step 3: We select the first sensor file (i.e., "sensor_nacc.csv") in the "sensor_files" list, and we proceed to Step 4. At this point, it is worthwhile mentioning that the processes described in Steps 4 to 13 are repeated for every sensor file in the "sensor_files" list.

Step 4: We select the first touch gesture record (i.e., "tg_record") in the "dtst_List" list, and we proceed to Step 5. At this point, it should be noted that the processes described in Steps 5 to 12 are repeated for every "tg_record" in the "dtst_List" list.

Step 5: Assuming one specific “tg_record” in the “dtst_List” list and one specific sensor file (e.g., “sensor_nacc.csv”), we perform a search process using the timestamp of the specific “tg_record” and the timestamps of all the records of the specific sensor file. We search to find all the records of the specific sensor file whose timestamps match with the timestamp of the specific “tg_record”. If among the records of the specific sensor file, there is at least one record whose timestamp matches with the timestamp of the specific “tg_record”, then we proceed to Step 6. Otherwise, we proceed to Step 8.

Step 6: If the number of records of the specific sensor file whose timestamps match with the timestamp of the specific “tg_record” is equal to one, then we proceed to Step 7a. Otherwise, if the number of matching records is greater than one, we proceed to Step 7b.

Step 7a: Based on which one is the specific sensor file (e.g., “sensor_nacc.csv”), the values of the non-common features (i.e., features 3, 4, and 5) of the only matching record of the sensor file are added as values of extra features to the specific “tg_record” in the “dtst_List” list. Then, we proceed to Step 12.

Step 7b: Based on which one is the specific sensor file (e.g., “sensor_nacc.csv”), the values of the non-common features (i.e., features 3, 4, and 5) of all the matching records of the sensor file are averaged. Then, the averaged values are added as values of extra features to the specific “tg_record” in the “dtst_List” list. Afterwards, we proceed to Step 12.

Step 8: Assuming one specific “tg_record” in the “dtst_List” list and one specific sensor file (e.g., “sensor_nacc.csv”), we perform a search process using the timestamp of the specific “tg_record” and the timestamps of all the records of the specific sensor file. The search process is different from the one performed in Step 5. In this case, we search for at maximum two records of the specific sensor file. The first record (i.e., “next_smaller” record) is required to have the timestamp that is the next smaller timestamp present in the sensor file compared to the timestamp of the specific “tg_record”. The second record (i.e., “next_greater” record) is required to have the timestamp that is the next greater timestamp present in the sensor file compared to the timestamp of the specific “tg_record”. If two records that satisfy these two conditions are found in the specific sensor file, then we proceed to Step 9. Otherwise, if one of these two records cannot be found in the sensor file, we proceed to Step 10.

Step 9: Based on which one is the specific sensor file (e.g., “sensor_nacc.csv”), we use the values of the non-common features (i.e., features 3, 4, and 5) of the “next_smaller” record and the “next_greater” record to calculate the values of the non-common features that will be added to the specific “tg_record”. This calculation is possible through the interpolation method because we need to find the values of the non-common features for a timestamp that lies in between the timestamps of the “next_smaller” record and the “next_greater” record. At this point, it is worthwhile mentioning that the use of the interpolation is the best choice since the timestamp of the specific “tg_record” may not be equal to the average value of the timestamps of the “next_smaller” record and the “next_greater” record. Then, the calculated values are added as values of extra features to the specific “tg_record” in the “dtst_List” list. Then, we proceed to Step 12.

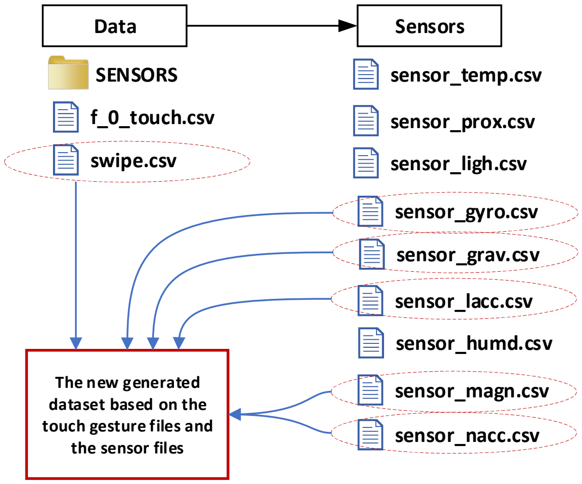


Figure 3: The New Generated Dataset.

Step 10: If the “next_smaller” record cannot be found in the sensor file, we proceed to Step 11a. Otherwise, if the “next_greater” record cannot be found in the sensor file, we proceed to Step 11b.

Step 11a: As the “next_smaller” record cannot be found in the sensor file, the interpolation method cannot be used in order to calculate the values of the non-common features of the specific “tg_record”. As a result, based on which one is the specific sensor file (e.g., “sensor_nacc.csv”), the values of the non-common features (i.e., features 3, 4, and 5) of the “next_greater” record of the sensor file are added as values of extra features to the specific “tg_record” in the “dtst_List” list. Then, we proceed to Step 12.

Step 11b: As the “next_greater” record cannot be found in the sensor file, the interpolation method cannot be used in order to calculate the values of the non-common features of the specific “tg_record”. As a result, based on which one is the specific sensor file (e.g., “sensor_nacc.csv”), the values of the non-common features (i.e., features 3, 4, and 5) of the “next_smaller” record of the sensor file are added as values of extra features to the specific “tg_record” in the “dtst_List” list. Then, we proceed to Step 12.

Step 12: If the specific “tg_record” is not the last record in the “dtst_List” list, then we select the next “tg_record” in the list and proceed to Step 5. Otherwise, we proceed to Step 13.

Step 13: If the specific sensor file is not the last one (i.e., “sensor_magn.csv” file) in the “sensor_files” list, then we select the next sensor file in the list and proceed to Step 4. Otherwise, we proceed to Step 14.

Step 14: In this Step, the fusion of the touch gesture record with the records of the sensor files for this specific task folder is considered as completed.

Table II. summarizes the features of the dataset generated from the above steps. In addition, Fig. 3 gives an overview of the process followed for the data fusion, leading to the generation of the new dataset based on the touch gesture files and the sensor files of the “HuMIdb” dataset.

TABLE II. FEATURES OF THE GENERATED DATASET

Features	Description
Touch gesture files	
ts	Timestamp in milliseconds.
orientation	Orientation of the mobile device (portrait=1, landscape=0).
x_coord	X coordinate data regarding to the use of the touchscreen.
y_coord	Y coordinate data regarding to the use of the touchscreen.
pressure	Pressure data regarding the use of the touchscreen.
fingerActions	Action data regarding the touchscreen (finger_down= 0, movement= 2, finger_up= 1).
"sensor_nacc.csv"	
x_axis_nacc	X axis data of the accelerometer of the mobile device.
y_axis_nacc	Y axis data of the accelerometer of the mobile device.
z_axis_nacc	Z axis data of the accelerometer of the mobile device.
"sensor_grav.csv"	
grav_data	Gravity data.
"sensor_gyro.csv"	
x_axis_gyro	X axis data of the gyroscope of the mobile device.
y_axis_gyro	Y axis data of the gyroscope of the mobile device.
z_axis_gyro	Z axis data of the gyroscope of the mobile device.
"sensor_lacc.csv"	
x_axis_lacc	X axis data of the linear accelerometer of the mobile device.
y_axis_lacc	Y axis data of the linear accelerometer of the mobile device.
z_axis_lacc	Z axis data of the linear accelerometer of the mobile device.
"sensor_magn.csv"	
x_axis_magn	X axis data of the magnetometer of the mobile device.
y_axis_magn	Y axis data of the magnetometer of the mobile device.
z_axis_magn	Z axis data of the magnetometer of the mobile device.

IV. CONCLUSIONS

User authentication acts as the first line of defense verifying the identity of a mobile user, often as a prerequisite to allow access to resources in a mobile device. Risk-based user authentication based on behavioral biometrics appears to have the potential to increase the authentication security level without sacrificing usability.

In our more recent work [13], we have provided a comprehensive work on the design of a risk-based adaptive user authentication mechanism that comprises a novel secure and usable user authentication solution. On top of that, we also modified adequately the "HuMIdb" dataset files, and we trained and tested the following most popular classification algorithms for risk-based authentication: K-NN, DT, SVM, and NB over the "HuMIdb" dataset using ten-fold cross validation. However, the evaluation results demonstrated impact of overfitting and therefore, we considered the concept of novelty detection to overcome this challenge. Thus, we tested and evaluated the following novelty detection algorithms: OneClassSVM, LOF, and KNN_average. All of them demonstrated a high performance for the same part of the ''HuMIdb'' dataset that was also used for the evaluation of the classification algorithms.

As it can be observed, to accurately evaluate classification and/or novelty detection algorithms for the proposed risk-based user authentication, it is important to utilize quality datasets to train and test the classification and/or novelty detection algorithms. Nevertheless, there is a lack of up-to-date, representative and comprehensive datasets that are publicly available to the research community and considered as benchmark datasets for effective training and evaluation of classification and/or novelty detection algorithms suitable for risk-based user authentication. Toward this direction, in this paper, the aim was to provide details on the methodology we followed to generate datasets based on "HuMIdb" dataset for training and testing classification and novelty detection algorithms for risk-based adaptive user authentication.

REFERENCES

[1] C. Liang, C. Yu, and X. Wei, "Auth+track: Enabling authentication free interaction on smartphone by continuous user tracking," *Conf. Hum. Factors Comput. Syst. - Proc.*, 2021.

[2] V. Sucasas, G. Mantas, M. Papaioannou, and J. Rodriguez, "Attribute-based pseudonymity for privacy-preserving authentication in cloud services," *IEEE Trans. Cloud Comput.*, vol. 7161, no. c, 2021.

[3] M. Papaioannou *et al.*, "A survey on security threats and countermeasures in Internet of Medical Things (IoMT)," *Trans. Emerg. Telecommun. Technol.*, no. May, pp. 1–15, 2020.

[4] M. Papaioannou, J. C. Ribeiro, V. Monteiro, V. Sucasas, G. Mantas, and J. Rodriguez, "A privacy-preserving user authentication mechanism for smart city mobile apps," in *IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (IEEE CAMAD)*, 2021, pp. 1–5.

[5] I. Essop, J. C. Ribeiro, M. Papaioannou, J. Rodriguez, G. Zachos, and G. Mantas, "Generating datasets for anomaly-based intrusion detection systems in IoT and industrial IoT networks," *Sensors*, vol. 21, no. 4, pp. 1–31, 2021.

[6] F. Pelekoudas-Oikonomou *et al.*, "Blockchain-Based Security Mechanisms for IoMT Edge Networks in IoMT-Based Healthcare Monitoring Systems," *Sensors*, vol. 22, no. 7, 2022.

[7] S. Wiefling, L. Lo Iacono, and M. and Dürmuth, "Is this really you? An empirical study on risk-based authentication applied in the wild.," in *In IFIP International Conference on ICT Systems Security and Privacy Protection (pp. 134-148). Springer, Cham.*

[8] S. Wiefling, M. Dürmuth, and L. Lo Iacono, "Verify It's You: How Users Perceive Risk-Based Authentication," *IEEE Secur. Priv.*, vol. 19, n, no. December, pp. 47–57, 2021.

[9] S. Gupta, A. Buriro, and B. Crispo, "Demystifying authentication concepts in smartphones: Ways and types to secure access," *Hindawi Mob. Inf. Syst.*, vol. 2018, 2018.

[10] M. Papaioannou, G. Mantas, A. Essop, P. Cox, I. E. Otung, and J. Rodriguez, "Risk-based adaptive user authentication for mobile passenger ID devices for land/sea border control," in *IEEE 26th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 2021, pp. 1–6.

[11] M. Papaioannou, G. Mantas, and J. Rodriguez, "Risk-based user authentication for mobile passenger ID devices for land and sea border control," in *2021 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*, 2021, pp. 180–185.

[12] M. Papaioannou, G. Mantas, D. Lymberopoulos, and J. Rodriguez, "User authentication and authorization for next generation mobile passenger ID devices for land and sea border control," in *12th International Symposium on Communication Systems, Networks and Digital Signal Processing, CSNDSP 2020*, 2020, pp. 8–13.

[13] M. Papaioannou, G. Zachos, I. Essop, G. Mantas, and J. Rodriguez, "Towards a Secure and Usable User Authentication for Mobile Passenger ID Devices for Land/Sea Border Control," *IEEE Access*, vol. 10, pp. 38832–38849, 2022.

[14] A. Acien, A. Morales, J. Fierrez, R. Vera-Rodriguez, and I. Bartolome, "BeCAPTCHA: Detecting human behavior in smartphone interaction using multiple inbuilt sensors," *arXiv Prepr. arXiv2002.00918*, 2020.

[15] A. Acien, A. Morales, J. Fierrez, R. Vera-Rodriguez, and O. Delgado-Mohatar, "BeCAPTCHA: Bot detection in smartphone interaction using touchscreen biometrics and mobile sensors," *arXiv Prepr. arXiv2005.13655*, no. May, 2020.