Alexandria Engineering Journal (2020) xxx, xxx–xxx

HOSTED BY

Alexandria University

Alexandria Engineering Journal

www.elsevier.com/locate/aej
www.sciencedirect.com

Implicit authentication method for smartphone users based on rank aggregation and random forest

Mohamed W. Abo El-Soud [a,d], **Tarek Gaber** [b,d,*], **Fayez AlFayez** [a], **Mohamed Meselhy Eltoukhy** [c,d]

[a] *Department of Computer Science and Information, College of Science, Majmaah University, Zulfi, Saudi Arabia*
[b] *School of Science, Engineering, and Environment, University of Salford, UK*
[c] *University of Jeddah, College of Computing and Information Technology at Khulais, Department of Information Technology, Jeddah, Saudi Arabia*
[d] *Faculty of Computers and Informatics, Suez Canal University, Ismailia 41522, Egypt*

Received 31 March 2020; revised 7 July 2020; accepted 12 August 2020

KEYWORDS

Implicit authentication;
Smartphone authentication;
Feature selection;
Classification;
Machine learning;
Random forest

Abstract Currently, the smartphone devices have become an essential part of our daily activities. Smartphone' users run various essential applications (such as banking and e-health Apps), which contains very confidential information (e.g., credit card number and its PIN). Typically, the smartphone's user authentication is achieved using mechanisms (password or security pattern) to verify the user identity. Although these mechanisms are cheap, simple, and quick enough for frequent logins, they are vulnerable to attacks such as shoulder surfing or smudge attack. This problem could be addressed by authenticating the users using their behaviour (i.e., touch behaviour) while using their smartphones. Such behaviours include finger's pressure, size, and pressure time while tapping keys. Selecting features (from these behaviours) could play an important role in the authentication process's performance. This paper aims to propose an efficient authentication method providing an implicit authentication for smartphone users while not imposing an additional cost of special hardware and addressing the limited smartphone capabilities. We first investigated feature selection techniques from the filter and wrapper approaches and then used the best one to propose our implicit authentication method. The random forest classifier is used to evaluate these techniques. It is also used to achieve the classification task in our authentication method. Using a public dataset, the experimental results showed that the filter-based technique (i.e., rank aggregation) is the best feature selection to build an implicit authentication method for the smartphone environment. It showed accuracy results around 97.80% using only 25 features out of 53 features (i.e., require less mobile resources (memory and processing power) to authenticate users. At the same time, the

* Corresponding author at: School of Science, Engineering, and Environment, University of Salford, UK, and Faculty of Computers and Informatics, Suez Canal University, Ismailia, Egypt.
E-mail addresses: m.wagieh@mu.edu.sa (M.W. Abo El-Soud), t.m.a.gaber@salford.ac.uk (T. Gaber), f.alfayez@mu.edu.sa (F. AlFayez), mohamed_eltokhy@ci.suez.edu.eg (M.M. Eltoukhy).

Peer review under responsibility of Faculty of Engineering, Alexandria University.

https://doi.org/10.1016/j.aej.2020.08.006

Please cite this article in press as: M.W. Abo El-Soud et al., Implicit authentication method for smartphone users based on rank aggregation and random forest, Alexandria Eng. J. (2020), https://doi.org/10.1016/j.aej.2020.08.006

results showed that our method has less error rate: 2.03 FAR, 0.04 FRR, and 1.04 ERR, comparing to the related work. These promising results would be used to develop a mobile application that allows implicit authentication of legitimate owners while avoiding the traditional authentication problems and using fewer smartphone resources.

1. Introduction

Mobile devices became very important in our life. According to [1], the number of smartphone users, all over the world, are increasing from a year-to-year. It is currently (2020), around 3.5 billion users. These users usually use their phones to store crucial data such as private pictures, videos, and credit card numbers. These data are confidential. To protect these data, mobile security has become one of the most important trends in research. Users' authentication comes as the first step of mobile security. Authentication is considered the main point of entry to smartphones. Once a user (an owner or imposter) gets into the mobile, they can perform nearly all tasks, access sensitive information, and other personal apps.

Traditional methods for authenticating users' identity to mobile devices are built on explicit authentication mechanisms such as passwords, PINs, or secret patterns [2,3]. These types of authentication are known as knowledge-based authentication methods. Although these methods are simple, cheap, and quick, they are subject to be forgotten, re-used, guessed, or shared with others [4]. Moreover, it is hard for humans to find a balance between password/PIN usability (i.e., user's memory load) and the security of this password/PIN (i.e., its hardship to be cracked) for these transitional methods. Furthermore, these methods are subject to attacks, such as shoulder surfing attack [5] and smudge attack [6].

Another mobile authentication approach is based on users' biometrics. This approach depends on unique human characteristics such as fingerprint, IRIS, or face unlock. These kinds of authentication are also known as physiological biometric [3]. Although these methods are effective and safe to authenticate the owner of the mobile (i.e., addressing some of the limitations of the traditional methods above), they have some disadvantages, including high computational cost, accuracy, and usability during the unlocked state. They also require special hardware devices such as sensors for iris scans, fingerprints, and face recognition. Furthermore, these methods are not practical for frequent logins/authentication as the case of smartphone authentication [3].

Smartphones are subject to continuous logins due to different reasons such as make phone calls, accessing social media, or phone banking. It would be convenient (more useable) if the authentication method is done implicitly (without either Password/PIN or face or fingerprints). The implicit authentication aims to offer another way to prevent unauthorized access to mobile devices [7]. This method works in the background of the device to decide to continue or to lock the device. It is divided into two phases. Firstly, the user accesses his mobile normally, and the system records behaviour-based features (e.g., how they touch the touchscreen) of the user. Secondly, after the user logs into her/his mobile, by any of the simple authentication methods, the system continuously compares current user behavior with the learned user model to take a decision of allowing the access or keep the phone locked [3,4]. This is known as behavioral biometrics. It includes analysis of information like the shape of handwriting, the timing of keystrokes, speech, unique patterns inherent in one's gait, and other features of one's general behavior without additional hardware [8]. Mobile authentication based on behavioral biometrics has advantages over physiological ones as the former could be used to efficiently support continuous and transparent authentication systems. Also, behavioral biometrics do not need any special hardware while collecting behavioral data, i.e., it is very cost-effective.

This paper focuses on proposing an efficient authentication method that is providing an implicit authentication for smartphone users while not imposing additional cost for special hardware and addressing the smartphone computational limitations. To achieve this aim, using a behaviour public dataset [9] (i.e., keystroke of touch screens), we investigated different feature selection techniques that could improve the authentication accuracy while addressing the smartphones computational limitations (limited hardware, battery life, and memory size). The contribution of this paper is twofold:

1. Investigating the impact of different feature selection techniques on the performance of the implicit authentication method. These techniques are from the two feature selection approaches, filter approach, and wrapper approach. A thorough evaluation was conducted.
2. Proposing, evaluating, and analyzing a new implicit authentication method that employs the best feature selection technique identified from the above contribution. The evaluation and analysis showed that this (the proposed method) method could address smartphones' computational limitations mentioned above.

The rest of the paper is organized as follows. A brief survey of the literature is given in Section 2. Section 3 introduces an overview of the techniques and algorithms used in the proposed method. The proposed method is presented in Section 4. Section 5 discusses the experimental work and its results. Finally, the conclusions are highlighted in Section 6.

Please cite this article in press as: M.W. Abo El-Soud et al., Implicit authentication method for smartphone users based on rank aggregation and random forest, Alexandria Eng. J. (2020), https://doi.org/10.1016/j.aej.2020.08.006

2. Related work

The most recent behaviour-dependent methods rely on the assumption that individuals tend to have consistent and stable behaviour profiles, such as gait movement [10], handwriting [11,12] and GPS trends [13]. These behaviour patterns are then recognized and used to achieve authentication. Below, a brief survey of implicit authentication for smartphone users will be discussed, and then the research gap will be identified.

Kayacik et al. [14] proposed an implicit authentication method employing the user profile concept. The implicit authentication method utilizes smartphone sensors data, including application use, cell towers, wifi networks, battery usage, GPS readings, and accelerometer output. Although the authors demonstrated that the model's attacks could be detected within approximately 900 s, authentication performance was not reported. Premkumar et al. [15] suggested another implicit authentication approach based on features collected from users' interactions with smartphones. These features include locations, pressure, and touch, using a set of behavioural patterns linked to a user interface without the use of external hardware. Using both random forest classifier and support vector machine (SVM) were employed for users' authentication. The reported accuracy of this system was 99.87%. However, a small dataset was used, only data from 20 participants.

Yao et al. introduced an implicit authentication model based on fuzzy logic. [16]. This model uses features derived from different events of day-to-day user routines (i.e., user behavior). Such events include features from the outgoing call, incoming call, outgoing SMS, incoming SMS, WIFI history, and browser history. They then conducted a preliminary study with data collected from two users. They have then conducted various performance experiments which their best results were an average of 99% for Gaussian-based schemes. Nonetheless, this method has only used data from two users.

Lee and Lee [7] proposed a re-authentication framework that uses several sensors integrated into the user's mobile, assisted by support information from a wearable device, such as a smartwatch. The system continuously monitors sensors data for authentication without intervention by the user. Their proposed system achieved over 92% accuracy with less than 2% battery consumption. Following the same approach as Lee and Lee [7], Yao et al. [17] suggested an adaptive neuro-fuzzy inference system (ANFIS) based framework for implicit authentication for smartphones. Using a set of data from 5 users, their system demonstrated a 95% accuracy rate by comparison to a non-ANFIS system, which achieved 90% in recognition performance.

Lee et al. [18] developed a new system for authentication of smartphone users implicitly. This system is based on the behaviour features related to the way a user's phone is picked-up. In other words, the behaviour of a user bending his/her arm whenever he/she picks up their smartphone to use it. Their work used data from 24 users for system evaluation. They reported 96.3% authentication accuracy using a similarity-based classier.

Jain et al. [19] also proposed a keystroke authentication method utilizing touchscreen events containing action (press or release), screen coordinates, timestamps, and pressure. A dataset of feature was built and one-class SVM was used to achieve the authentication process. Their method gave an Equal Error Rates (EER) of 10% for data related to keystrokes, 3.5% for touchscreen data, and 3% for both types of data. However, only data from 30 participants were used, which is also a small dataset.

Alshanketi et al. [20] presented an implicit authentication method for smartphone users based on their keystroke dynamic on a touchscreen. They were using a random forest classifier and a pubic RHU keystroke dataset in [9], consisting of 51 individuals with 985 samples, a result of an Equal Error Rate (EER), 5.8%. Using the same dataset, Ulinskas et al. [21] investigated the impact of using feature selection on the implicit authentication. They used the *t-test* for feature ranking and k-nearest neighbour classifier to analyze the problem of fatigue recognition when keystroke dynamics data is used for implicit authentication. Their investigation showed that 91% accuracy rate has been achieved using one class of the data, which is "key release-release" (RR).

Recently, Tharwat et al. [8] used genetic algorithm (GA) for feature selection and bagging classifier or classifiercation for user identification using public keystroke dynamics database, the touch-based keystroke dataset in [9]. Their results showed that a set of 10 features achieved an accuracy of 83.8%, which could be improved.

From the above literature survey, the following remarks can be drawn. Firstly, the implicit authentication accuracy can still be improved. The best-achieved accuracy was 96.3% [18], but this paper did not utilize the users' touch behaviour. It used the user's smartphone pick-up pattern. From the results of the implicit authentication based on the user's touch behaviour data, i.e., [20,21,8], the best accuracy was 91%, and it was achieved in [21] which could be improved. Secondly, almost all the analyzed articles above did not conduct a comprehensive evaluation study. These articles did not evaluate their solutions in terms of accuracy, FAR, FRR, and ERR. Based on these findings, this paper aims to propose a touch-based implicit authentication method addressing the above limitations.

3. Preliminaries

This section gives an overview of the techniques/algorithms used in the proposed method. It highlights three different feature selection techniques: rank aggregation, OneR attribute selection, and correlation attribute evaluation. It also summarized the used classifier, i.e., random forest algorithm.

3.1. Feature selection techniques

Feature selection plays an essential role in the process of data classification. In general, the dataset includes several attributes. Since, all features are not relevant and sometimes affect the classification process. Hence, feature selection is an essen-

Please cite this article in press as: M.W. Abo El-Soud et al., Implicit authentication method for smartphone users based on rank aggregation and random forest, Alexandria Eng. J. (2020), https://doi.org/10.1016/j.aej.2020.08.006

tial step to develop an efficient system. The proposed solution in this paper aims to develop an efficient authentication method for smartphone users. The irrelevant features of this authentication method may influence the method performance in terms of the accuracy rate and/or the processing time as well as memory space and battery consumption.

The filter approach and the wrapper approach are two types of feature selection techniques. In the filter approach, the features are evaluated on the basis of selection metrics with respect to the characteristics of the dataset. This approach uses mathematical methods for feature set evaluation. The Filter-based methods require class labels in order to determine the significance of features and thus require labeled data. The filter-based techniques are independent of the classification algorithm and are typically less computationally intense than wrapper methods [22]. On the other side, in the wrapper approach, the features are evaluated based on a specific machine learning algorithm. The wrapper-based methods rank features using the results obtained from performing a classification process on a dataset using various feature subsets. The wrapper-based methods depend mainly on the type of classifier. These methods can be more computationally intensive than the filter-based methods. Also, the wrapper-based methods suffer from over-fitting and complexity of computational time [23].

The idea of using the feature selection in our proposed method is to improve the performance by finding the most discriminating features that have high capability to identify the user uniquely. To achieve this, we investigated feature selection methods from wrapper and filter-based methods. From the filter approach, we used the rank aggregation [24] and correlation attribute evaluation (CA) [25], while from the wrapper approach, OneR attribute selection [26] was used. In the following subsection, each of these methods is briefly summarised.

3.1.1. Rank aggregation

The idea of rank aggregation is to rank the features with respect to their relationship. Feature ranking methods have advantages of scalability, simplicity, and good empirical success. Let S be a matrix containing n instances $s_i = (s_{i1}; \ldots; s_{id}) \in \mathbb{R}^d$. Let $v_i = (v_1; \ldots; v_n)$ is the vector of class labels for n instances. The set of features (F): $f_j = (f_1; \ldots; f_d)$. The features are ranked according to the scoring function $H(j)$ which is calculated using the values of s_{ij} and v_i. The idea is to suppose that a high score is indicative of a high capability of the feature. Hence, the feature are sorted in decreasing order according to its $H(j)$ value [24].

When applying scoring functions, the algorithm finds a way to various aggregate lists of features. To get a better ordering, the rank aggregation is used to combine many different rank orderings on the same set of alternatives to obtain the best ordering list. The goal of rank aggregation is to find the best list when dealing with feature selection, which is able to achieve the highest classification rate [24].

This is can be seen as an optimization problem, when we look at $argmin(C; \sigma)$, where *argmin* gives a list of σ at which the distance C is minimized with a randomly selected ordered list. In this optimization problem, the objective function is represented by [24]:

$$F(\sigma) = \sum_{i=1}^{n} w_i \times C(\sigma, L_i). \tag{1}$$

where the weights associated with the lists L_i is represented by w_i, C is the distance between a pair of ordered lists (a distance function measuring) and L_i is the i^{th} ordered list of cardinality L. Hence, the best solution is to define the value of σ^* that would minimize the sum of distances between L_i and σ^* as given by [24]:

$$\sigma^* = \arg\min \sum_{i=1}^{n} w_i \times C(\sigma, L_i). \tag{2}$$

3.1.2. Correlation Attribute Evaluation (CA)

In this method, the weights of the attributes are determined according to their correlation with the target class. Pearson's correlation method is used to measure the correlation between each attribute and the target class. Pearson correlation coefficient (PCC) [25] measures the linear correlation between two random features. It is symmetric in nature. The PCC value falls in a definitely closed interval [−1,1]. The value of PCC close to either −1 or 1 indicates the strong relationship between the two variables. PCC value close to 0 infers the weaker relationship between them. PCC value 0 indicates no relationship between them. PCC quantifies the degree to which a relationship between two variables can be described as shown in Eq. 3 [25].

$$p(y, z) = \frac{\sum (y - \bar{y})(z - \bar{z})}{\sqrt{\sum (y - \bar{y})^2 (z - \bar{z})^2}} \tag{3}$$

3.1.3. OneR attribute selection

This method evaluates each attribute individually by using the OneR classifier [26,27]. The idea of this classifier is based only on the attribute values and labels. This algorithm utilizes OneR classifier to give a weight for each attribute. For each attribute, it then creates a rule utilizing the same attribute and also measures its error rate afterward [28]. It chooses the rule having the least error from the generated rules for training. It generates rules for whole features and chooses a baseline (the best) performance as a benchmark for various training techniques.

3.2. Random forest algorithm

The random forest algorithm [29] has been highly effective as a public objective classification and regression technique. It combines many randomized decision trees and gathers their predictions by averaging. Furthermore, it is applied to large-scale problems, returns measures of variable importance, and is easily suitable to different ad-hoc learning tasks. The random forest algorithm is described as illustrated in Algorithm 1.

Algorithm 1. Random Forest Algorithm

Inputs: T_n: Training set, $S_n \in \{1, \cdots n\}$: The number of sampled dataset in each tree, $R > 0$: The number of trees, nodesize $\nu \in \{1, \cdots n\}$: The number of examples in every cell that is not split, $ntry \in \{1, \cdots p\}$: The number of probable directions for splitting at every node of every tree, $v \in [1, 2, \cdots U]^p$: Predicted value, where U=51 user;
Output: The random forest prediction at v
for $i=1, \cdots R$ **do**
  Choose points S_n *uniformly with replacement according to k-fold in* T_n
  Set $P_0 = \{[1, 2, \cdots U]^p\}$ *The partition related to the tree root*
  For all $1 \le \kappa \le S_n$, $P_\kappa = \phi$
  level $\ell = 0$ *and* $n_{ofnodes} = 1$ **while** $S_n > n_{ofnodes}$ **do**
    if $P_\ell = \phi$ **then**
      $\ell = \ell + 1$
    else
      Let the first element in P_ℓ *is* B **if** $B < \nu$ **then**
        $P_\ell \leftarrow P_\ell \backslash \{B\}$
        $P_{\ell+1} \leftarrow P_{\ell+1} \cup \{B\}$
      else
        Choose a subset uniformly, without replacement
        $ntry \in \{1, \cdots p\}$ *of cardinality ntry*
        Choose the best split in B by optimizing the CART-split
        Criterion along the ntry coordinates
        Call B_R *and* B_L. *Cut the cell B related to the best split*
        the resulting cells:
        $P_\ell \leftarrow P_\ell \backslash \{B\}$
        $P_{\ell+1} \leftarrow P_{\ell+1} \cup \{B_R\} \cup \{B_L\}$
        $n_{ofnodes} = n_{ofnodes} + 1$
  The average of Z_i *falling in the cell of* v *in partition* $P_\ell \cup P_{\ell+1}$ *= calculate the predicted value* $m_n(v; \Theta_j; T_n)$ *at* v.
Calculate the random forest estimate $m_{R,n}(v; \Theta_1; \cdots; \Theta_R; T_n)$

4. The proposed method

The proposed method, depicted in Fig. 1, consists of three main phases: feature selection, feature sampling, and classification. In the feature selection phase, the features are ranked using the three techniques. In other words, the ranking process is achieved with three different techniques: ranking aggregation [24], correlation attribute evaluation [25], and OneR attribute selection [26] mentioned above. The output of this phase is three ranked features sets, i.e., *ranked features 1*, *ranked features 2* and *ranked features 3*. The hypothesis here is that each of these ranked features would give different classification results according to the ranking technique used for sorting the features.

The feature sampling phase and classification phase are interleaved, so they will be explained together. The backward elimination method is used to find the best set of features. *N sets* of the ranked features are produced. The aim here is to investigate which subset of features would give the highest classification rate (i.e., identifying the legitimate owner of a smartphone) while minimizing the processing time and using less resources. To achieve this aim, all the available features, which are 53 features (see Section 5 for more details), are ranked according to each different feature ranking technique. Then, we started to remove the lowest ranked feature one by one. Saying that all feature size $= n$) are used. Then, one feature is removed out and the rest is used, i.e., *(n-1)*, for testing. This process is continued until the best result is achieved. Hence, by evaluating the different sets of ranked features, $n, n-1, n-2, \ldots 1$ sets, we would identify the best set of features giving the best results. As it can be seen in Fig. 1, each ranked set is given to the classifier (i.e., the random forest algorithm) individually and then the results (accuracy, False Acceptance Rate (FAR), False Rejection Rate (FRR) and Equal Error Rate (ERR)) are recorded. Then, the best result is identified from each ranking algorithm, i.e., three results should be identified. Finally, these three results are compared to determine the final best accuracy result. This means that one of the feature selection algorithms (rank aggregation, correlation attribute selection, and OneR attribute selection) and one subset of the features will be reported to be the best combination achieving the highest result.

Please cite this article in press as: M.W. Abo El-Soud et al., Implicit authentication method for smartphone users based on rank aggregation and random forest, Alexandria Eng. J. (2020), https://doi.org/10.1016/j.aej.2020.08.006

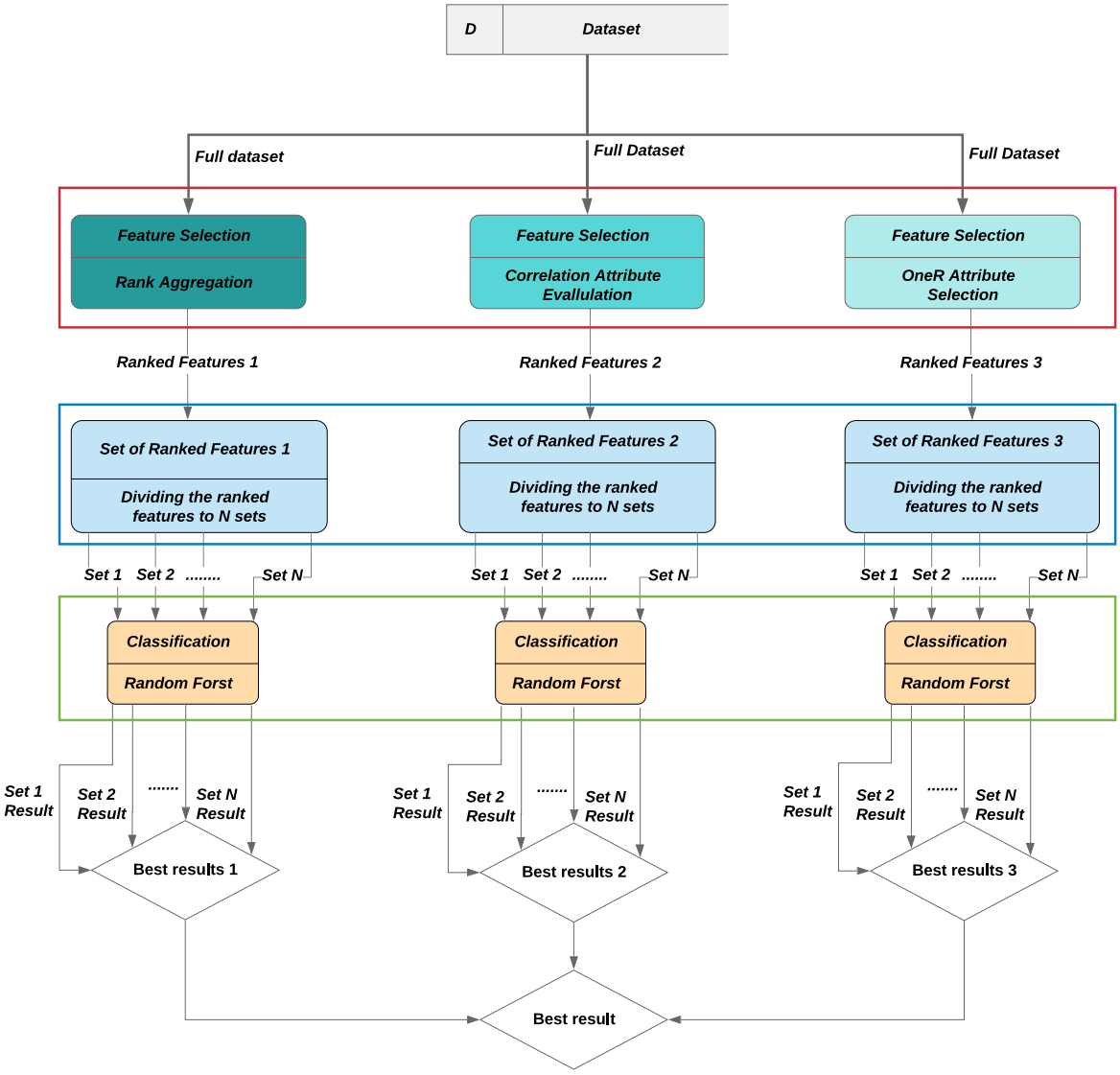

Fig. 1 The architecture of the proposed method.

5. Results and analysis

To evaluate the proposed method, we designed two main scenarios and a public dataset has been used (see more details below). The results of these scenarios are measured using a number of well-know evaluation measures for user authentication techniques. We used the measures: accuracy, False Acceptance Rate (FAR), False Rejection Rate (FRR) and Equal Error Rate (EER). All these measures are computed under the 10-fold cross-validation method which means that every data sample will be in testing samples exactly once while it will be in training samples 9 times. The 10-fold cross-validation method has been used to validate that the results of statistical analysis could be generalized to any independent datasets.

5.1. Dataset

The dataset used in this study is a public dataset known as RHU touch mobile keystroke which was published in [9]. This dataset is collected from 51 persons, each of whom has been requested to type a password "rhu.university" 15 times during three different sessions and the average time between each session was five days. In each session, the data are separately collected. Generally, the collected database includes both males and females with different ages and consists of 954 samples. From this database, four pieces of information are extracted from each user, including the time between key pressure and key release (PR), the time between key release and key pressure (RP), the time between two key pressures (PP), and the time between two key releases RR) [29]. The main features (PR, RP, PP, RR) further consist of sub-features describing the different times' values of entering a password: RR, RP, and PP compose of 13 sub-features while PR consists of 14 sub-features. Hence the total number of features is 53 features for each user (i.e., class).

5.2. Experimental scenarios

As described in the proposed system, different feature selection methods and one classifier (random forest) were used. So, we

Table 1 The average of 10-fold cross-validation results (Accuracy, FAR, FRR, ERR and Run Time) of features selection techniques using random forest

No. Fe.	Rank Aggregation					Correlation Attribute					OneR Attribute				
	Accur.	FAR	FRR	ERR	Time	Accur.	FAR	FRR	ERR	Time	Accur.	FAR	FRR	ERR	Time
53	93.17	5.80	0.14	2.97	100.63	93.17	5.53	0.15	2.84	114.80	92.55	5.45	0.16	2.80	122.43
52	92.84	5.75	0.15	2.95	100.50	93.42	4.88	0.14	2.51	114.30	93.81	5.22	0.13	2.67	121.23
51	93.48	5.45	0.14	2.79	102.62	93.60	5.50	0.14	2.82	111.07	92.21	5.82	0.17	2.99	121.98
50	93.61	5.15	0.14	2.64	101.21	93.64	4.85	0.13	2.49	110.38	93.42	5.27	0.14	2.70	119.57
49	93.71	5.10	0.13	2.62	97.05	94.13	4.88	0.12	2.50	113.39	93.48	5.12	0.14	2.63	108.26
48	93.17	5.65	0.14	2.90	98.96	93.71	5.22	0.13	2.67	107.37	93.89	4.48	0.13	2.31	104.87
47	93.41	5.75	0.14	2.94	91.91	92.97	5.98	0.15	3.07	105.90	93.59	5.02	0.14	2.58	107.73
46	92.86	5.50	0.15	2.83	95.37	93.60	5.32	0.13	2.73	108.35	93.50	4.77	0.14	2.45	106.56
45	94.23	4.70	0.12	2.41	93.51	93.92	5.50	0.13	2.81	105.16	93.71	5.40	0.13	2.77	105.01
44	94.23	4.52	0.12	2.32	92.25	94.46	4.70	0.12	2.41	101.85	94.67	4.65	0.11	2.38	105.45
43	94.23	4.75	0.12	2.44	95.75	94.54	4.33	0.12	2.22	104.37	95.29	4.10	0.10	2.10	103.50
42	93.90	5.30	0.13	2.71	95.43	94.34	4.88	0.12	2.50	101.89	95.08	4.35	0.10	2.23	101.33
41	94.55	4.30	0.11	2.21	92.05	94.88	4.35	0.11	2.23	103.51	94.85	4.50	0.11	2.30	102.60
40	94.22	4.85	0.12	2.49	94.59	94.67	4.43	0.11	2.27	102.31	94.55	4.47	0.11	2.29	100.72
39	94.29	4.50	0.12	2.31	97.48	94.54	4.65	0.11	2.38	100.04	94.34	4.75	0.12	2.43	100.74
38	94.75	4.35	0.11	2.23	95.78	94.36	4.75	0.12	2.43	99.66	94.33	4.60	0.12	2.36	98.79
37	95.62	3.78	0.09	1.94	97.54	94.97	4.52	0.10	2.31	95.97	95.32	3.83	0.10	1.97	97.09
36	95.73	3.75	0.09	1.92	96.84	95.52	3.95	0.09	2.02	96.81	95.68	3.95	0.09	2.02	96.02
35	94.97	4.35	0.11	2.23	92.43	95.06	4.12	0.10	2.11	111.54	94.98	4.20	0.10	2.15	97.18
34	94.45	4.63	0.12	2.37	88.70	94.64	4.58	0.11	2.34	105.42	94.54	4.60	0.11	2.36	99.66
33	95.07	4.30	0.10	2.20	86.84	94.99	4.38	0.10	2.24	99.37	94.65	4.13	0.11	2.12	95.47
32	95.07	4.25	0.10	2.18	91.20	95.06	4.28	0.10	2.19	97.20	95.79	3.80	0.09	1.94	94.25
31	95.16	4.25	0.10	2.18	90.45	95.08	4.25	0.10	2.18	95.07	95.69	3.93	0.09	2.01	90.98
30	96.76	2.80	0.07	1.43	81.03	94.56	4.67	0.11	2.39	93.19	96.64	3.15	0.07	1.61	91.77
29	95.92	3.75	0.08	1.92	80.28	95.57	4.00	0.09	2.05	91.04	96.85	2.80	0.06	1.43	88.98
28	96.12	3.30	0.08	1.69	81.53	96.34	3.22	0.08	1.65	86.73	96.33	3.25	0.08	1.66	89.43
27	96.76	3.00	0.07	1.53	79.92	96.32	3.35	0.08	1.71	85.64	96.54	3.00	0.07	1.54	86.19
26	97.28	2.30	0.06	1.18	80.76	96.96	2.80	0.06	1.43	82.33	96.55	3.00	0.07	1.54	88.04
25	**97.80**	**2.03**	**0.04**	**1.04**	**77.55**	96.75	3.05	0.07	1.56	84.10	**97.38**	**2.40**	**0.05**	**1.23**	**90.11**
24	95.91	3.25	0.04	1.67	76.41	**96.95**	**2.60**	**0.06**	**1.33**	**87.06**	95.81	3.90	0.09	1.99	87.72
23	95.59	3.88	0.09	1.99	75.03	95.70	3.95	0.09	2.02	86.10	95.28	4.10	0.10	2.10	85.76
22	94.54	4.90	0.11	2.51	75.07	95.38	4.10	0.10	2.10	85.69	95.59	3.73	0.09	1.91	84.87
21	95.39	4.25	0.10	2.17	76.15	95.16	4.25	0.10	2.18	86.28	95.69	3.60	0.09	1.84	90.50
20	95.61	3.60	0.09	1.85	78.79	95.39	4.15	0.10	2.12	81.80	95.49	4.10	0.09	2.10	88.53
19	95.69	3.85	0.09	1.97	74.68	95.61	3.90	0.09	2.00	80.00	95.91	3.50	0.08	1.79	86.93
18	94.66	4.90	0.11	2.51	72.22	95.10	4.05	0.10	2.08	79.53	95.30	4.35	0.10	2.22	87.30
17	94.56	4.39	0.11	2.25	72.03	93.71	5.40	0.13	2.77	74.65	94.86	4.70	0.11	2.40	76.56
16	94.74	4.65	0.11	2.38	70.62	94.94	4.08	0.11	2.09	72.10	94.83	4.35	0.11	2.23	74.73
15	92.15	6.50	0.17	3.33	66.90	95.38	4.05	0.10	2.07	69.78	92.15	6.25	0.17	3.21	74.77
14	89.39	7.75	0.24	3.99	66.26	91.74	6.72	0.18	3.45	73.46	89.73	8.08	0.23	4.15	73.69

The bold values are highlighting the the best values of the results.

designed two main scenarios to evaluate the impact of the feature size of each of these selection methods and the configuration of the random forest (i.e., the size of the decision tree). These scenarios are described in the following subsections.

Scenario ONE: Determination of the feature set size

This scenario aims to determine the best size of the features that gives the highest performance. To achieve this aim, different set of features with different sizes have been tested. Firstly, all features (saying that all feature size = n) are tested. Then, one feature is omitted and the rest is tested, i.e., *(n-1)* features were tested. This process is continued until the best result is achieved. This scenario is investigated under the three feature selection algorithms described earlier. The expected output of this scenario is the highest classification result using a set of features less than n and the best feature ranking algorithm under the default configuration of the random forest classifier.

Scenario TWO: Random Forest Configuration

Given that Scenario ONE resulted in a set of features less than n and the best feature ranking algorithm, in Scenario TWO, we aim to investigate whether the decision tree value of the random forest classifier has an impact on improving the classification results. The output of this scenario would be the best decision tree value contributing to the best classification (authenticating the mobile owner).

All experiments were conducted on a laptop/PC Core i5-2400 CPU 3.10 GHz with 4.00 GB. The implementation was compiled using MATLAB R2015a under Windows 10.

5.2.1. Results of scenario ONE

The results of this scenario are summarized in Table 1. It is worth mentioning that the order (top–bottom) of these results is following the order of the ranked features obtained from

Table 2 The set of selected features.

Rank Aggregation	Correlation Attribute	OneR Attribute
RP_13	RR_11	RP_11
PR_6	RP_11	RP_1
PR_4	PP_11	RP_8
PR_3	PP_1	PP_8
PR_2	RP_8	RR_11
PR_5	PP_8	RR_1
PR_7	RR_8	RP_3
PR_13	RR_5	RP_10
PR_8	PP_5	PP_11
PR_11	RR_1	PP_5
PR_10	RP_5	PP_10
PR_9	PP_2	RP_6
PR_1	RR_2	PP_1
PP_13	RP_1	PP_4
PP_12	RR_12	RR_9
PP_11	RP_12	PP_6
PP_4	PP_12	RR_12
PP_3	RR_6	RR_6
PP_2	RP_2	RR_8
PP_5	PP_6	RR_2
PP_6	RP_6	RR_10
PP_7	PR_11	PP_3
PP_10	PR_14	PP_9
PP_9	PR_12	PP_12
PP_8		RR_5

each different feature selection method. So, by evaluating the different sets of ranked features, $n, n-1, n-2, \ldots 1$ sets, would identify the best set of features giving the best results in terms of the criteria given above. From this table, it can be noticed that the best results were obtained when the number of features was 25 features using the rank aggregation or OneR techniques, and 24 features using correlation attribute technique. The list of the significant features that were selected by each of these feature selection techniques are listed in Table 2. Although the difference is not significant between the rank aggregation and the oneR attribute selection results but the former is the best among the three final results especially at the FAR and ERR measures. To conclude, it can be said that the random forest with the rank aggregation technique could help in the authentication process of smartphone users using only 25 out of 53 features with 97.80% accuracy, 2.03 FAR, 0.04 FRR, and 1.04 ERR. In addition, the rank aggregation based results are the best in terms of the computational time, taking 77.55 MS.

From the identified sub-features above, it could be noticed that the rank-aggregation with the random forest could be used to build an efficient implicit authentication method for smartphone applications while meeting its low computational capabilities. The obtained results support the claim that filter-based features selection methods (the rank aggregation in our case) are faster than the wrapper based method (i.e., OneR technique) [23].

Also, from Fig. 2, it can been seen that the accuracy almost increases when the number of features decreases until it reached a level (i.e., 25 features), while the accuracy started to decrease after using 24 features or less. The interpretation of this could be that the high impacted features (up till features 25) are having significant contribution in the user authentication process (i.e., classification). Once starting the use of low

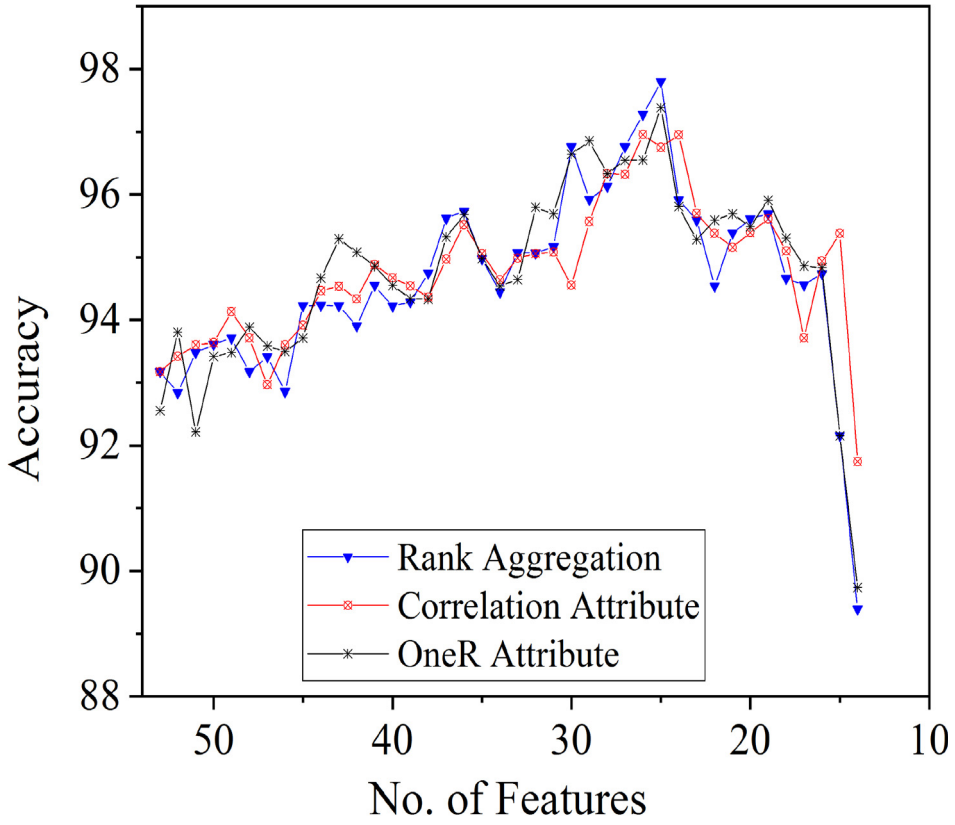

Fig. 2 Classification accuracy rates versus different number of features.

Table 3 Determining the best value of the random forest parameter (decision tree size).

Decision Tree	Average of 10 Cross-Validation R.				
	Accur.	FAR	FRR	ERR	Time
3	96.96	2.80	0.06	1.43	69.10
5	**97.80**	**2.03**	**0.04**	**1.04**	**77.55**
10	97.27	2.2	0.06	1.13	125.85
5	96.86	2.95	0.06	1.51	152.76
20	96.54	3.15	0.07	1.61	173.91
25	97.09	2.80	0.06	1.43	184.82
30	97.17	2.55	0.06	1.30	191.24

The bold values are highlighting the the best values of the results.

Table 4 Comparison with related work.

Reference	No. of Users	Feature Selection	No. of Features	Dataset	Class.	Acc.	FAR	FRR	ERR
Lee and Lee [7]	20	N/A	N/A	Private	k-NN	92.1	7.5	8.3	–
Jain et al. [19]	30	N/A	31	Private	SVM	–	–	–	10
Yao et al. [17]	5	N/A	N/A	Private	ANFIS	95	–	–	–
Alshanketi et al. [20]	51	N/A	53	Public [9]	RF	–	5.8	5.8	5.8
Ulinskas et al. [21]	51	t-test	47	Public [9]	KNN	91	–	–	–
Tharwat et al. [8]	51	GA	10	Public [9]	Bagging	83.8	–	–	–
Our Method	51	Rank	25	Public [9]	RF	**97.80**	**2.03**	**0.04**	**1.04**

The bold values are highlighting the the best values of the results.

ranked features, the results went down as these low ranked-features have in-significant impact on differentiating different users.

5.2.2. The results of scenario TWO

The results of Scenario ONE showed that the use of the rank aggregation feature selection algorithm could improve the classification (the authentication) results. To further investigate whether the random forest parameter (decision tree size) would further improve the classification results, in Scenario TWO, the best set of features are used, i.e. the 25 features gave the best results in Scenario ONE and the decision tree values of the random forest, i.e., 3, 5, 10, 15, 25 and 30, are investigated. The results of this scenario are summarized in Table 3. As discussed earlier, all these results are computed using 10-fold cross-validation method.

From these results, it can be noticed that the accuracy did not improve when the decision tree size was increased. The other performance metrics (FAR, FRR, and ERR) did not improved too. In addition, the computational time increased while the number of decision trees increased. The main conclusion of this scenario is that the best decision tree size giving the best classification rate was 5-trees.

5.2.3. Comparison and discussion

To further evaluate our obtained results, we compared them with the results of the related work discussed in Section (2). The compared work is chosen such that they proposed implicit authentication for smartphones using private or public dataset. A summary of this comparison is shown in Table 4. From this table, the following remarks can be drawn. Firstly, using the private dataset, the best achieved accuracy was 95% by [17], but as discussed in Section (2) this study only used data from 5 participants which is very limited and the data is not public for the community to evaluate. Furthermore, the results in [17] did not report any results for FAR, FRR nor ERR which are very important to evaluate any authentication method.

Secondly, with respect to the comparison with the work that used the same datset (i.e., [9]) and applying feature selection techniques [20,21,8], our proposed method achieved the best results using 25 features. Although, Tharwat et al. [8] used only 10 features (i.e., less than the features used in our method) but they accomplished accuracy of 83.8% which is less than ours with 14%. In addition, Tharwat's method has not been evaluated in terms of FAR, FRR nor ERR.

Generally, this comparison means that our proposed method is feasible in terms of computation cost (efficiency) which measures how many resource this method would use to achieve an implicit authentication. As discussed earlier, implicit authentication is a continuous process running in the smartphone background. Therefore, it would be empirical if such authentication mechanism consumes as less as possible amount of smartphone resources, e.g., processing power and memory. One could say that the next generation of smartphones would improve mobile processors and memories. However, as reported in [30], the capacity of the mobile batteries will remain to be constrained because of the limitations in space and heat transfer. This means, it is expected that batteries life would be a big barrier for mobile computing efficiency in coming years. Therefore, it would be practical to adopt mobile authentication techniques which use as less resources as possible. Having our proposed method achieved implicit users authentication using only 25 features out of 53 features, this means that our method is efficient.

This further could be discussed under two models of mobile application developments: thick-client and thin-client model. If our proposed method is implemented on the smartphones,

following the thick-client model, it would save the battery life and the memory space while running this continuous authentication process. In addition, if it is implemented on the smartphones, following the thin-client model, it would minimize the communication overhead where less data (number of features) representing the smartphone's users would be sent over the network. This will also minimize the computation cost of the server side (processing 25 features instead of 53). In case of thin-client model, our method could be used in many applications such security sector, banks, healthcare, and government administration where an extra layer of security is required to protect their users sensitive data.

In terms of security issues of user authentication, the comparison presented in Table 4, shows that our method achieved less FAR, FRR and ERR rates which are important for not allowing illegitimate users to be granted access to the smartphone containing private and confidential information such as banking and personal data.

Like other touch-based implicit authentication methods, [20,21,8], our method is also not subject to the shoulder surfing attack. This attack could not be mounted by outsider/random attackers but also by malicious insiders [31]. Examples of malicious insiders include friends, relatives, and colleagues. One could be very careful from outsiders but with malicious insiders, they may take it easy and give confidence to their family and friends. However, the insiders could abuse this confidence and take the advantage of it and mount a shoulder surfing attack. As our proposed method do not use any of the traditional authentication methods, PIN, graphical or non-graphical password, our proposed method is secured against the shoulder surfing attacks.

In addition to the malicious insiders identified in [31], we argue that our proposed implicit authentication method could help parents to have a strong control over their smartphone from their kids. The children can abuse unintended and unprotected parents' smartphones making unauthenticated online transactions, e.g., by accessing their banking App or e-commerce App. This could be prevented using our touch-based implicit authentication method.

6. Conclusion and limitation

With the widespread of smartphone devices, there should be a seamless way to authenticate their owners. This paper proposed an efficient implicit authentication method addressing this problem. The proposed method is based on using the rank aggregation (filter-based technique) feature selection technique and random forest classifier to implicitly authenticate users using their touch behaviour. Implicit features were extracted from the users' touch behaviour and then ranked according to their impact on the classification results using the rank aggregation technique. The ranked features are then fed to the random forest classifier to identify which user is accessing the smartphone. The rank aggregation is chosen in our proposed method after a comparison with OneR attribute selection (wrapper-based method) and correlation attribute evaluation (filter-based method). The results of the evaluation showed that a smartphone's user could be implicitly authenticated using fewer features (25 out of 53) selected by the rank aggregation technique and classified by the random forest while still achieving less error rate: 2.03 FAR, 0.04 FRR, and 1.04 ERR. Using these fewer features means that the smartphone's limitations (i.e., memory size, battery life, and computational capability) could be addressed. In addition, security attacks, e.g., shoulder surfing, could be thwarted too.

The limitation of this study is that it is data-based study. Testing the proposed method using another dataset might affect the obtained results. The results reported in this paper might be confirmed or changed if different datasets are used in terms of size or features. In future work, it is planned to investigate whether deep learning techniques would further improve the accuracy results and other metrics.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to thank the deanship of scientific research and the research center for Engineering and Applied Sciences, Majmaah University, Saudi Arabia, for their support and encouragement, Project No. 38/80.

References

[1] Statista, 2020: Number of smartphone users worldwide from 2016 to 2021 (in billions) available at https://www.statista.com/statistics/330695/number-of-smartphone-usersworldwide/ Accessed June 2020..

[2] M. Qi, Y. Lu, J. Li, X. Li, J. Kong, User-specific iris authentication based on feature selection, in: 2008 International Conference on Computer Science and Software Engineering, pp. 1040–1043, Hubei, 2008. doi: 10.1109/CSSE.2008.1060..

[3] Abdulaziz Alzubaidi, Jugal Kalita, Authentication of Smartphone Users Using Behavioral Biometrics, J. IEEE Commun. Surveys Tut. 18 (3) (2016) 1998–2026.

[4] Wei-Han Lee, Ruby B. Lee, Multi-sensor authentication to improve smartphone security, in: International Conference on Information Systems Security and Privacy (ICISSP), Angers, pp. 1–11, 9 March 2017..

[5] N.H. Zakaria, D. Griffiths, S. Brostoff, J. Yan, Shoulder surfing defence for recall-based graphical passwords, in: SOUPS '11: Proceedings of the Seventh Symposium on Usable Privacy and Security, Article No.: 6, pp. 1–12, July 2011, doi: 10.1145/2078827.2078835..

[6] S. Cha, S. Kwag, H. Kim, J.H. Huh, Boosting the guessing attack performance on android lock patterns with smudge attacks, in: ASIA CCS17, ACM, April 2017, pp. 313–326..

[7] Wei-Han Lee, Ruby Lee, Implicit sensor-based authentication of smartphone users with smartwatch, in: HASP, ACM, No. 9, 18 June 2016, pp. 1–8, ISBN 978-1-4503-4769-3/16/06..

[8] A. Tharwat, A. Ibrahim, T. Gaber, A.E. Hassanien, Personal identification based on mobile-based keystroke dynamics, in: Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2018, AISI 2018, Advances in Intelligent Systems and Computing, vol 845, Springer, cham, 2019, pp. 457–466. doi.org/10.1007/978-3-319-99010-1-42..

[9] M. El-Abed, M. Dafer, R.E. Khayat, RHU Keystroke: A mobile-based benchmark for keystroke dynamics systems, in: 2014 International Carnahan Conference on Security

Technology (ICCST), Rome, 2014, pp. 1–4. doi: 10.1109/CCST.2014.6986984..
[10] C. Nickel, T. Wirtl, C. Busch, Authentication of smartphone users based on the way they walk using k-NN algorithm, in: 2012 Eighth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Piraeus, 2012, pp. 16–20, doi: 10.1109/IIH-MSP.2012.11..
[11] M. Trojahn, F. Ortmeier, Toward mobile authentication with keystroke dynamics on mobile phones and tablets, in: 27th International Conference on Advanced Information Networking and Applications Workshops, Barcelona, 2013, pp. 697–702, https://doi.org/10.1109/WAINA.2013.36.
[12] M. Shahzad, A.X. Liu, A. Samuel, Secure unlocking of mobile touch screen devices by simple gestures: you can see it but you cannot do it, in: MobiCom, September 2013, pp. 39–50, doi:10.1145/2500423.2500434..
[13] S. Buthpitiya, Y. Zhang, A.K. Dey, M. Griss, n-gram geo-trace modeling, in: Proceedings of the 9th International Conference on Pervasive Computing, vol. 6696, June 2011, pp. 97–114. doi: 10.1007/978-3-642-21726-5-7..
[14] H.G. Kayacík, M. Just, L. Baillie, D. Aspinall, N. Micallef, Data driven authentication: on the effectiveness of user behaviour modelling with mobile device sensors, Mob. Security Technol. (MoST), 28 Oct. 2014..
[15] S. Premkumar, C. Samuel, H.P.C. Duen, Z. Hongyuan, Latentgesture, active user authentication through background touch analysis, in: Proceedings of the Second International Symposium of Chinese CHI. Chinese CHI'14, New York, NY, USA, ACM, 2014, pp. 110–113. doi:10.1145/2592235.2592252..
[16] F. Yao, S.Y. Yerima, B. Kang, S. Sezer, Fuzzy logic-based implicit authentication for mobile access control, in: SAI Computing Conference (SAI), London, 2016, pp. 968–975, https://doi.org/10.1109/SAI.2016.7556097.
[17] F. Yao, S.Y. Yerima, B. Kang, S. Sezer, Continuous implicit authentication for mobile devices based on adaptive neuro-fuzzy inference system, in: 2017 International Conference on Cyber Security and Protection Of Digital Services (Cyber Security), London, 2017, pp. 1–7, https://doi.org/10.1109/CyberSecPODS.2017.8074846.
[18] Wei-Han Lee, Xiaochen Liu, Yilin Shen, Hongxia Jin, Ruby B. Lee, Secure pick up: implicit authentication when you start using the smartphone, in: Procee. of SACMAT'17, Indianapolis, IN, USA, 21–23, June 2017..
[19] Jain Lohit, V.M. John, J.C. Michael, C.T. Charles, Passcode keystroke biometric performance on smartphone touchscreens is superior to that on hardware keyboards, Int. J. Res. Comput. Appl. Inf. Technol. 2 (2014) 29–33. ISSN Online:2347-5099, Print:2348-0009..
[20] F. Alshanketi, I. Traore, A.A. Ahmed, Improving Performance and Usability in Mobile Keystroke Dynamic Biometric Authentication, in: 2016 IEEE Security and Privacy Workshops (SPW), San Jose, CA, 2016, pp. 66–73, https://doi.org/10.1109/SPW.2016.12.
[21] M. Ulinskas, M. Woźniak, R. Damaševičius, Analysis of keystroke dynamics for fatigue recognition, in: O. Gervasi, et al. (eds.), Computational Science and Its Applications – ICCSA 2017. ICCSA 2017. Lecture Notes in Computer Science, vol 10408. Springer, Cham, 2017..
[22] J. Handl, J. Knowles, Feature subset selection in unsupervised learning via multiobjective optimization, Int. J. Comput. Intell. Res. 2 (3) (2006) 217–238.
[23] M. Bennasar, Y. Hicks, R. Setchi, Feature selection using joint mutual information maximisation, Expert Syst. Appl. 42 (22) (2015) 8520–8532, https://doi.org/10.1016/j.eswa.2015.07.007.
[24] N. Ailon, M. Charikar, A. Newman, Aggregating inconsistent information: ranking and clustering, in: Proceedings of the ACM Symposium on Theory of Computing (STOC), ACM, 2005, pp. 684–693.
[25] Mark A. Hall, Correlation-based Feature Selection for Machine Learning, Thesis is submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy at The University of Waikato, April 1999..
[26] Aditi Mahajan, Anita Ganpati, Performance evaluation of rule based classification algorithms, Int. J. Adv. Res. Comput. Eng. Technol. (IJARCET) 3 (10) (2014).
[27] Cristiana Neto, Maria Brito, Vítor Lopes, Hugo Peixoto, António Abelha, José Machado, Application of data mining for the prediction of mortality and occurrence of complications for gastric cancer patients, Entropy, MDPI 21(12) (2019) 1163. doi:10.3390/e21121163..
[28] Shawkat Ali, Kate A. Smith, On learning algorithm selection for classification, Appl. Soft Comput. 6 (2) (2006) 119–138, https://doi.org/10.1016/j.asoc.2004.12.002.
[29] Gerard Biau, Erwan Scornet, A Random Forest Guided Tour, Mathematics Subject Classification, Springer, 62G05, 62G20, 2010..
[30] D. Li, S. Hao, J. Gui, W.G.J. Halfond, An empirical study of the energy consumption of android applications, in: 2014 IEEE International Conference on Software Maintenance and Evolution, Victoria, BC, 2014, pp. 121–130, doi: 10.1109/ICSME.2014.34..
[31] I. Muslukhov, Y. Boshmaf, C. Kuo, J. Lester, K. Beznosov, Know your enemy: the risk of unauthorized access in smartphones by insiders, in: 15th International Conference on Human Computer Interaction with Mobile Devices and Services, ACM, 2013, https://doi.org/10.1145/2493190.2493223.