Neural Computing and Applications (2020) 32:13651–13678
https://doi.org/10.1007/s00521-020-04775-0

ORIGINAL ARTICLE

Machine learning algorithms for improving security on touch screen devices: a survey, challenges and new perspectives

Auwal Ahmed Bello[1] · Haruna Chiroma[2] · Abdulsalam Ya'u Gital[1] · Lubna A. Gabralla[3] · Shafi'i M. Abdulhamid[4] · Liyana Shuib[5]

Received: 21 November 2018 / Accepted: 5 February 2020 / Published online: 17 February 2020

Abstract
Mobile phone touch screen devices are equipped with high processing power and high memory. This led to users not only storing photos or videos but stored sensitive application such as banking applications. As a result of that the security system of the mobile phone touch screen devices becomes sacrosanct. The application of machine learning algorithms in enhancing security on mobile phone touch screen devices is gaining a tremendous popularity in both academia and the industry. However, notwithstanding the growing popularity, up to date no comprehensive survey has been conducted on machine learning algorithms solutions to improve the security of mobile phone touch screen devices. This survey aims to connect this gap by conducting a comprehensive survey on the solutions of machine learning algorithms to improve the security of mobile phone touch screen devices including the analysis and synthesis of the algorithms and methodologies provided for those solutions. This article presents a comprehensive survey and a new taxonomy of the state-of-the-art literature on machine learning algorithms in improving the security of mobile phone touch screen devices. The limitation of the methodology in each article reviewed is pointed out. Challenges of the existing approaches and new perspective of future research directions for developing more accurate and robust solutions to mobile phone touch screen security are discussed. In particular, the survey found that exploring of different aspects of deep learning solutions to improve the security of mobile phone touch screen devices is under-explored.

Keywords Machine learning algorithms · Deep learning · Mobile phone touch screen · Android · Support vector machine · Command attention · Security

1 Introduction

Mobile phones started as a basic cellular device where calls are made and limited text messages. Currently, touch screen mobile phone devices have revolutionized the mobile phone market as well as dominated the user-input technologies for the mobile phone devices. The revolution and dominance of the mobile phone touch screen devices are attributed to their high level of flexibility and good usability [1]. The use of mobile phone touch screen devices has become part of people lives including people with

✉ Shafi'i M. Abdulhamid
shafii.abdulhamid@futminna.edu.ng

Auwal Ahmed Bello
ahmadbello18@gmail.com

Haruna Chiroma
freedonchi@yahoo.com; chiromaharun@fcetgombe.edu.ng

Abdulsalam Ya'u Gital
asgital@gmail.com

Lubna A. Gabralla
lubnagabralla@gmail.com

Liyana Shuib
liyanashuib@um.edu.my

[1] Department of Mathematical Sciences, Abubakar Tafawa Balewa University, Bauchi, Nigeria

[2] Department of Computer Science, Federal College of Education (Technical), Gombe, Gombe, Nigeria

[3] Department of Computer Science and Information Technology, Community College, Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia

[4] Department of Cyber Security Science, Federal University of Technology, Minna, Minna, Nigeria

[5] Department of Information System, Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia

disability [2, 3]. The mobile phone touch screen devices have gained unprecedented attention because of the availability of millions of appealing applications [4]. It is estimated that by 2020, the number of mobile phone users will hit 4.78 billion [5]. The mobile phones are equipped with touch screen input, more flexible operating system such as android, iOS and windows, higher memory space of up to 256 GB and higher-speed processors. Those functionalities result from the advancement of the mobile computing, mobile technology and mobile networks [6]. As they possess all these functionalities, it prompted users of mobile phone to stores sensitive and private information including emails, photos [7, 8], private video files, mobile banking applications and also use them for official work. In addition, these mobile phone touch screen devices deliver and access media and digital content services to the mobile device users ubiquitously [9]. The increasing ubiquitous of mobile phone touch screen has prompted a serious security concern. The data stored and accessed from mobile phone touch screen devices are mainly protected by authentication at the point of login [10]. The protection of the private information on the mobile phone touch screen is highly critical [1], and the storage of sensitive information raises more security concerns.

As a result of security concerned, many authentication techniques have been devised and used for the protection of the mobile phone touch screen devices from intrusion by attackers. The authentication technique includes but not limited to password, graphical password, PIN, biometrics, gesture/pattern, lego rebot, location-based authentication, lengthy and free text strings, two-factor face and multi-factor-based authentications. Despite the improvement recorded by these authentication schemes in protecting the mobile devices, they are still vulnerable to attacks such as shoulder surfing, impersonation, smudge attacks, dictionary, brute force, keylogger and guessing.

To further improve the security of mobile phone touch screen devices, machine learning algorithms are applied to enhance the security of the mobile phone touch screen devices and are gaining a tremendous popularity. However, notwithstanding the growing popularity, up to date, no dedicated comprehensive survey has been conducted on machine learning algorithm solutions to improve the security of mobile phone touch screen devices.

This survey aims to connect this gap by conducting a comprehensive survey on the solutions of machine learning algorithms to improve the security of mobile phone touch screen including the analysis and synthesis of the algorithms and methodologies provided for those solutions. This survey can serve as a starting point for new researchers intending to dwell into touch screen mobile phone authentication using machine learning algorithms. Also, it is intended for experience researchers to easily identify areas of weakness and propose a new authentication scheme for the touch screen mobile phone based on machine learning algorithms.

The summary of the six key contributions and novel idea to the artificial intelligence in this survey are as follows:

- The survey provides a brief tutorial on machine learning procedure for improving the security of mobile phone touch screen devices and provides a comprehensive dedicated review on major contributions from different projects and limitation from each of the projects.
- Based on the fundamental flow of the application of machine learning algorithms in improving the mobile phone devices security, the paper organizes the literature into the nine primary areas of support vector machine, artificial neural networks, K-nearest neighbor, Bayesian network, particle swarm optimization, random forest, dynamic time warping, hybrid algorithms and others.
- In addition, the survey categorizes the reviewed studies on machine learning algorithms for improving mobile phone devices security in terms of feature engineering, operating system and mobile device manufacturers, performance metrics, datasets and implementation platforms.
- A new taxonomy is created based on the applications of machine learning algorithms to advance the security of mobile phone touch screen devices.
- A comprehensive discussion of the challenges in improving the security of mobile phone touch screen devices is presented, and new perspective for future research directions was identified. Prominent among the challenges faced by the existing machine learning algorithm solutions to improve security of mobile phone devices includes: (1) requiring separate feature engineering technique before applying the machine learning algorithm which is cumbersome, time-consuming and can lead to biased, (2) old technology, e.g., attention commands can be used to compromise the security of Android, (3) existing machine learning solutions to the mobile devices security are limited to a particular operating system without compatibility across different operating system platforms and (4) under-exploration of different aspects of deep learning solutions to improve the security of mobile phone touch screen devices despite the effectiveness, robustness and efficiency of the deep learning.
- To provide opportunity for enhancing research in machine learning algorithm solutions for improving the security of future mobile phone touch screen devices, the survey discusses new perspectives of possible solution methodologies to each of the challenges pointed out in the survey.

The other sections of the paper are arranged as follows: Sect. 2 presents the rudiment of the machine learning

algorithms. Section 3 ushers some mobile phone security breaches. Section 4 provides the procedure for applying machine learning in authenticating mobile phone devices. Section 5 presents contributions from different machine learning security authentication projects. Section 6 provides feature engineering. Sections 7, 8, 9 and 10 present operating system and mobile device manufacturers, datasets, performance evaluation parameters and implementation platforms, respectively. Lastly, Sect. 11 presents the general discussion of the research area. Section 12 points out challenges and new perspective for future research before concluding the paper in Sect. 13.

2 Machine learning algorithms rudiment

This section presents the machine learning algorithms found to be used in the literature for authenticating mobile phone touch screen devices. There are many other machine learning algorithms but this survey only provided those that were used for authentication in mobile phone touch screen devices. A brief background of the algorithms is provided to show interesting readers how the algorithms operate to achieve their goal; especially, new researchers and expert researchers are interested to switch to this research area. In addition, it can make the survey to be self-contained. The background is very brief because there is an extensive description of the algorithm in the literature. Interesting reader can go for further reading in the cited references.

2.1 Support vector machine

The support vector machine (SVM) is a binary classifier [11] where data samples are represented in space as a point. Categories of different data samples are separated by a far distance. New samples of data are categorized into the same space and classified based on which side of the gab they belong to. Two different classes of data are separated by increasing the margin of the point closest to each of the class, and this closest point is called support vector. The center of the margin is the hyperplane which separates the two classes optimally [12]. The training set T of the data in SVM can be expressed as [12]:

$$T = \{(X_1, X_1), (X_2, X_2), \ldots, (X_m, X_m)\} \tag{1}$$

where X_i is an n-dimensional vector, and Y_i is either 1 or -1 representing the class which the point X_i belongs. The general SVM classification function is represented as:

$$F(x) = w \cdot x - b \tag{2}$$

where w is the weight vector, b is the bias which will be calculated during training process by the SVM.

To classify a training set correctly, $F(.)$ (or w and b) must return positive number for positive data points and negative numbers for negative data points, for every point X_i in T.

$$\begin{aligned} w \cdot x_i - b > 0 \quad & \text{if } y_i = 1 \text{ and} \\ w \cdot x_i - b > 0 \quad & \text{if } y_i = -1 \end{aligned} \tag{3}$$

2.2 Artificial neural network

The artificial neural networks (ANN) is structured in layers, it has one input and output layer, multiple or non-multiple hidden layer [13]. On the ANN, each layer has a certain number of nodes, and each node is connected to all the nodes in the layer(s) adjacent to it. Each connection carries a weight which represents the connection popularity and strength. When training the network, input variables are fed to the input layer and are gradually passed to the output layer. The training is done iteratively, and the differences between the current output of the network and the desired output in the output layer are minimized as it iterates [14]. The equation for an input (d), with hidden nodes (M) and output nodes (c) can be expressed as [14]:

$$a_j = \sum_{i=1}^{d} w_{ji}^{(1)} x_i + w_{jo}^{(1)} \tag{4}$$

where $w_{ji}^{(1)}$ and $w_{jo}^{(1)}$ are the weights in the input node layer connecting node i and j, and a bias of the hidden node j including x_o with a constant of (1), Eq. (4) becomes:

$$a_j = \sum_{i=0}^{d} w_{ji}^{(1)} x_i \tag{5}$$

is transform using the activation function $f(\cdot)$ to get Eq. (6):

$$z_{j=g(a_j)} \tag{6}$$

For each output node k, Eq. (7) can be expressed as:

$$a_k = \sum_{j=1}^{M} w_{kj}^{(2)} x_i + w_{ko}^{(2)} \tag{7}$$

The weight will then absorb the bias as shown in Eq. (8):

$$a_k = \sum_{j=1}^{M} w_{kj}^{(2)} z_j \tag{8}$$

Using nonlinear activation function, the activation function of kth output node is obtained by transforming Eq. (8):

$$y_{k=\delta(a_k)} \tag{9}$$

The notation $\delta(\cdot)$ is used for the activation at the output node to show that the activation at the hidden nodes and output nodes is not the same. The complete equation can be obtained by combining (5)–(9)

$$y_k = \delta\left(\sum_{j=1}^{M} w_{kj}^{(2)} g\left(\sum_{i=0}^{d} w_{ji}^{(1)} x_i\right)\right) \tag{10}$$

If activation is to be used in the output nodes, then $\delta(a) = a$.

2.3 Particle swarm optimization

The particle swarm optimization (PSO) is an optimization algorithm which finds the optimal solution by moving particles to different position in the swarm (search space). The PSO initially starts with the set of particles which are randomly distributed; the velocity vector for each particle is computed in the search space, and updates each particle position using the velocity vector in the search space until optimum solution is found [15, 16]. In PSO, the *i*th particle in the search space is represented as $X_i = (X_{i1}, X_{i2}, X_{i3} \ldots, X_{iD})$, and the velocity for each particle is represented as $Vi = (V_{i1}, V_{i2}, V_{i3} \ldots, V_{iD})$. The particles are manipulated using the following equations:

$$V_{id} = v_{id} + c1 * \text{rand}() * (P_{id} - x_{id}) + c2 * \text{Rand}() * \left(P_{gd} - x_{id}\right) \tag{11}$$

$$X_{id} = X_{id} + X_{id} \tag{12}$$

where c_1 and c_2 are two positive constants, and rand() and Rand() are two random functions within the range [1, − 1].

For balancing the local and global search, an inertia weight w parameter is introduced to the equation:

$$V_{id} = w * v_{id} + c1 * \text{rand}() * (P_{id} - x_{id}) + c2 * \text{Rand}() * \left(P_{gd} - x_{id}\right) \tag{13}$$

$$X_{id} = X_{id} + X_{id} \tag{14}$$

2.4 K-nearest neighbor

The K-nearest neighbor (KNN) is a learning algorithm used for classification and regression. In KNN, an object is classified using its nearest neighbor majority vote [17]. The KNN is one of the simplest classifiers because computations are postponed until classification. Therefore, it is also referred to as lazy learning algorithm. The KNN can be the best choice for classification especially when there is little or none knowledge of the data distribution; it uses a supervised method of classification that is label. It is a distance-based classifier that is based on the distance between the training and the test samples [17]. KNN classifier procedure can be expressed as;

Let $T = \{(x_i, y_i)\}_{i=1}^{N}$ represents the training set, where x_i is the training vector and y_i is the corresponding class of x_i. Given a query x', its unknown class label y' can be obtained by the following two steps:

Firstly, we identify a set of k similar labeled target neighbors for the query x^i. Denote the training set $T' = \left\{\left(x_i^{NN}, y_i^{NN}\right)\right\}_{i=1}^{k}$, arranged in ascending order in terms of Euclidean distance $d\left(x', x_i^{NN}\right)$ between $x' and x_i^{NN}$ [18].

$$d\left(x', x_i^{NN}\right) = \sqrt{\left(x' - x_i^{NN}\right)^T \left(x' - x_i^{NN}\right)} \tag{15}$$

Secondly, the class label is predicted by its nearest neighbor majority votes:

$$y' = \text{argmax}_y \sum_{\left(x_i^{NN}, y_i^{NN}\right) \in T'} \delta\left(y = y_i^{NN}\right) \tag{16}$$

where y is the class label, y_i^{NN} is the *i*th nearest neighbor among its K-nearest neighbors. $\delta\left(y = y_i^{NN}\right)$, is the Dirac delta function, takes a value of zero if $y \neq y_i^{NN}$ and one if $y = y_i^{NN}$ [18].

2.5 Random forest

The random forest (RF) is a classification and regression tree built using training data bootstrap samples and selected random features [19]. Data samples are classified in random forest based on aggregation of majority vote. The RF built multiple decision trees and outputs the classification or prediction of each tree during training [19]. RF can be expressed mathematically as:

Given a training set $X = x_1, \ldots, x_n$ with classes $Y = y_1, \ldots, y_n$, a random sample of the training set will be selected and trees will be fit to these samples: For *b = 1,…,B*

After training classification for samples, x will be made by taking the majority vote.

$$f = \frac{1}{B}\sum_{b=1}^{B} f_b(x) \tag{17}$$

Also, the random forest classifier uses the Gini index which determines the impurity of an attribute in respect to the class. For a given training set T, selecting a class at random and assuming it belongs to the class C_i; the Gini index can be expressed as:

$$\sum\sum_{j/i} (f(C_i, T)/|T|)\left(f\left(C_j, T\right)/|T|\right) \tag{18}$$

where $f\left(C_j, T\right)/|T|$ is the probability that the selected case belongs to the class C_i [20].

2.6 Dynamic time warping

The dynamic time warping (DTW) is an algorithm that measures the similarity between two different sequences P and Q that may vary [21] such as audio, video and walking acceleration. The DTW measures the distance between two times series by calculating the optimal match between them. The "warp" distance indicates how similar a set is to another set, and warp value of zero indicates absolute similarity between the two sets [22]. Therefore, the higher the warp distance, the higher the differences between the two sets. DTW can be expressed mathematically as:

$$D_{DTW}(P,Q) = D(L_P, L_Q) \tag{19}$$

where P and Q are sequences and L_P and L_Q are length of P and Q, respectively. D is the cumulative distance matrix for P and Q.

The major challenge of DTW is to determine the optimal warping path. The warping path can be represented as $W = \{w(1), w(2), \ldots.., w(k), \ldots.., w(K),\}$ where each w is a pair of points for the samples being matched, i.e., $(w(k) = [i(k), j(k)])$. The warping function is required to minimize the cost function [23].

$$D = \sum_{k=1}^{k} d[(w(k))] \tag{20}$$

where

$$d[(w(k))] = d(T(i(k)), R(j(k))) \tag{21}$$

is the distance between $i(k)$ of the test pattern and $j(k)$ of the reference pattern [23].

2.7 Bayesian network

The Bayesian networks (BN) are used to represent probability distribution dependencies based on the concept of a directed acyclic graph [24]. In a Bayesian network model, each feature in a given domain is represented as a node in the graph, while dependencies between those features are represented as edges connecting the respective nodes. Independences between features are represented as zero edge connecting the nodes. The BN is one of the most effective classifiers due to its high performance in prediction [25]. As the number of features increases, the BN becomes time-consuming [24].

Let $D = \{u_1, \ldots, u_N\}$ represent the training set; in classification of each element, u_i in the training set is a tuple of the form $a_1^i, \ldots a_n^i, c^i$ that assigns values to both the attributes $A_1, \ldots, A_n$ and the class variable C.

$$LL(B|D) = \sum_{i=1}^{N} logP_{B\left(c^i|a_1^i,\ldots.a_n^i\right)} + \sum_{i=1}^{N} logP_B\left(a_1^i, \ldots a_n^i\right) \tag{22}$$

The first term of the equation estimates how effectively B estimates the probability of the class given the features, and the second term of the equation determines how well B estimates the distribution of the features.

2.8 Sensitivity with respect to the models

The literature survey shows that different alternative machine learning algorithms were used for improving the accuracy of touch screen mobile phones. Different models can be realized depending on the machine learning algorithm applied. The performance of the alternative models depends on parameter settings, and the models are sensitive to the parameters. Sections 2.1 to 2.7 present the machine learning algorithms, and the discussion of their sensitivity to parameter settings is as follows:

The SVM is sensitive to parameter C (the penalty factor) which if it is too large, it might cause overfitting, while very small value might cause underfitting [26]. It also controls the trade-off between error on training data and maximization of margin. The SVM is also sensitive to the epsilon (ε) that relies on the target values of the training data and determines the accuracy of the function. The ANN is a sensitive parameter setting, and the ANN requires the setting of the number of input neurons, hidden neurons and output neurons; initial weights are associated with the connections, learning rate, bias and activation function. The kernel parameter (σ) of the SVM determines the stability of the general performance. The KNN is sensitive to the parameter K. Large number of k minimizes the effect of noise in classification but makes the differences between classes less distinct. The PSO is sensitive to quit a number of parameters such as the current position of the particle, particle velocity, best position found by a particle so far, inertia weight and acceleration constants. The RF is sensitive to the following parameter settings: number of trees, the node size and number of different descriptors tied at each split. The BN is sensitive to kernel estimator.

3 Mobile phone touch screen security authentication

The security of mobile phone serves as a protective mechanism between a user and access to a mobile phone that authenticates users based on information provided by the user. Building mobile phone security requires collecting user data, and trains the data based on the collected features to build a profile. Finally, it authenticates the user based on the built profile. A user is authenticated according to the following expression [27]:

$$1 - \frac{\sum_{i=1}^{N}\left(\frac{\text{Occurance of Feature}_{ix}}{\sum_{x=1}^{M}\text{Occurance of Feature}_{ix}} \times W_i\right)}{N} \geq \text{threshold} \tag{23}$$

where x = Feature$_i$ value, M = total number of values for Feature$_i$, N = total number of features, W_i = the weight associated with Feature$_i$ $(0 < W_i \leq 1)$, threshold = predefined value between 0 and 1. In addition, for a mobile phone security to authenticate a user, a subject will provide the requested input and claim the identity of the user U. From the obtained input, feature vectors say $f^{\text{PR}}, f^{\text{PP}}, f^{\text{RP}}$ and f^{RR} are evaluated and compared with the trained profile of user U to produce the dissimilarity distance [28]:

$$s^{\Delta} = \begin{cases} \sum_{K=1}^{K} \dfrac{|f_u^{\Delta}(k) - \mu_u^{\Delta}(k)|}{\sigma_u^{\Delta}(k)}, & \Delta = \text{PR} \\ \sum_{K=1}^{K-1}, \dfrac{|f_u^{\Delta}(k) - \mu_u^{\Delta}(k)|}{\sigma_u^{\Delta}(k)}, & \Delta \in \{\text{PP}, \text{RP}, \text{RP}\} \end{cases} \tag{24}$$

which are used to generate a global distance s. Decision on whether the input is from the real user is made by comparing the matching result with the threshold. If the distance s is less than the selected threshold, the user is identified as the legitimate owner of the claimed identity; otherwise, the user is rejected [28]. The computed score in (24) can then be normalized according to the length of the features expressed as:

$$r^{\Delta} = \begin{cases} \dfrac{s^{\Delta}}{K}, & \Delta = \text{PR} \\ \dfrac{s^{\Delta}}{K-1}, & \Delta \in \{\text{PP}, \text{RP}, \text{RR}\} \end{cases} \tag{25}$$

and the global distance s is calculated as [28]:

$$s = s^{\text{PR}} + s^{\text{PP}} + s^{\text{RP}} + s^{\text{RR}} \tag{26}$$

3.1 Security breaches in mobile phone touch screen devices

The mobile phone touch screen devices are susceptible to different kinds of security breaches. Many security breaches for mobile phone exist; but this section provides few examples to the reader on different approaches of compromising the security of mobile phone touch screen but not in any way exhaustive because is beyond the scope of the study, for example, impersonations. The impersonations are in different ways all in an effort to compromise the security of the mobile phone touch screen device. Impersonation scenario in mobile authentication occurs when an intruder tries to gain access to a mobile device using known identity of the genuine user. For example, an attacker can try to gain access to mobile phone touch screen by drawing the user's pattern trying to mimic the user's electroencephalography (EEG) signal. This is done by wearing the EEG signal headset while drawing the pattern and deceiving the system to allow gaining access to the mobile device [29]. Figure 1 depicts an impersonation scenario using EEG signal headset.

A shoulder surfing attack is the situation where an attacker is provided with a video of the genuine user drawing the correct gesture. Based on the video, the attacker tries to learn and redraw the correct gesture several times to gain access to the device [1]. Another shoulder surfing attack as pointed out in [30], an attacker monitor as the genuine user draws a PIN. Then, learns the genuine user correct PIN and tries to gain access to the mobile device by drawing the users correct PIN. In another scenario, if the intruder has access to a video of how the user is drawing a PIN, the intruder will try to learn and imitate the genuine user's PIN drawing behavior to gain access to the mobile device. An attacking scenario on lock pattern starts by video recording on how the real user draws the pattern while unlocking the device after which the intruder studies the video to come up with number of patterns to try and gain access to the device [31]. Song et al. [32] considers zero effort impersonation scenario where the attacker has similar hand shape with the user, knows the gestures to perform, but without knowing the behavior attached to performing the gestures. The attacker will then try to gain access to the device by performing those gestures. Smudge attack is the situation where the genuine user gesture traces

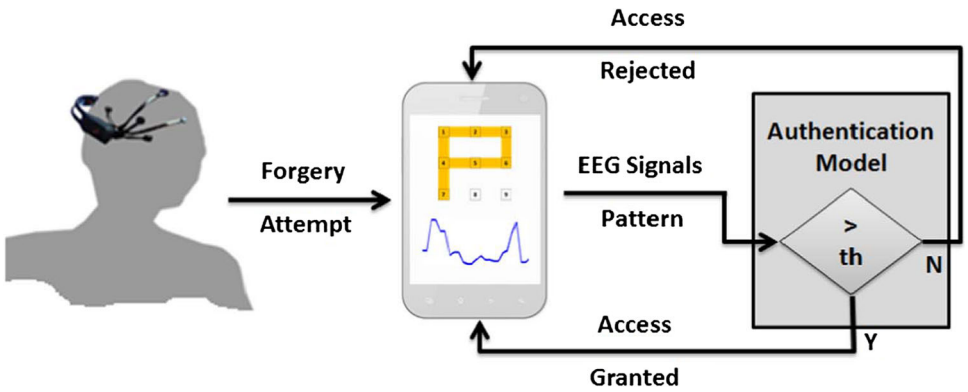

Fig. 1 Impersonation scenario using EEG signal headset [29]

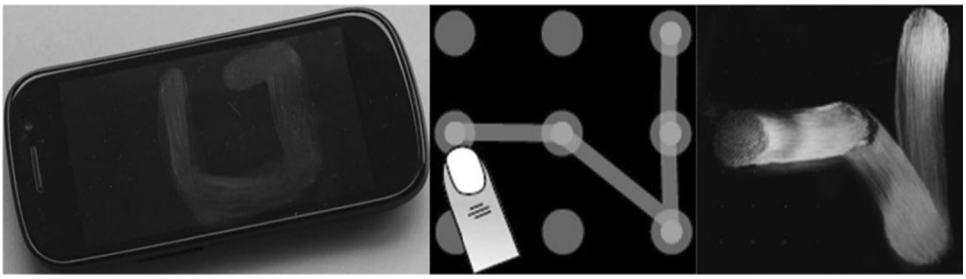

Fig. 2 Scenario of smudge attack [33]

are left clearly on the screen. The attacker replicates the gesture traces to gain access to the device. Shoulder surfing attack is the situation where the finger movement of the genuine user performs the gesture as shown to the attacker in an animated video several times so that the attacker can try to mimic the behavior to gain access to the device. Su et al. [33] considers multiple shoulder surfing through which multiple attackers combine to record the user's authentication identity. The recording is through a video recording device after which the recordings are analyzed to guess the correct user identity. Figure 2 is the scenario of smudge attack.

Liang et al. [34] an attacker can develop a malicious android application similar to a paid application and place it on play store for free. When the application is downloaded by a careless user and the user launches the application, it starts recording the user's sensor data. Subsequently, send it to the attacker where the attacker extracts features from the sensor information. The extracted features are used by the attacker to train a model and compared to the original model to gain access.

Shen et al. [35] assume that the attacker already knows the password and has already accessed the device, while the imposter is trying to perform some actions, and the usage behavior is compared. If the behavior of the imposter does not match the genuine user behavior, the operating system gets notified, and the attacker is logged out of the device. Mehrnezhad et al. [36] an intruder impersonates a user by stealing the PIN of the user once the user accesses a website controlled by the imposter. An embedded JavaScript code starts recording the user's sensor information without the user's permission and can use the information to gain access to the device of the genuine user. Table 1 summarizes the attacks on mobile phone touch screen devices as presented in this section.

Table 1 Summary of attacks on mobile phone touch screen devices

References	Type of attack
[29]	EEG signal mimicking
[1, 30, 31]	Shoulder surfing
[32]	Smudge attack
[33]	Shoulder surfing and smudge attack
[34]	Malicious application attack
[35]	Mimicking usage behavior
[36]	Website's hidden JavaScript program attack

3.2 Pros and cons of machine learning algorithms on the security system of mobile phone touch screen

The pros and cons of the machine learning-based security systems are provided in this section to give readers the opportunity to understand both the pros and cons of the machine learning on security systems. For example, the training data of the security system can be poisoned by adversary derives and injected to reduce the system classification accuracy. The poising of the training data is done by inserting cautiously created samples into the training data. Therefore, the classification can be distorted during the training phase allowing the attacker to redefine the classification as the attacker wishes. The failure of a machine learning-based security system can cause damage to the system or even help cybercrimes to be perpetrated [37]. A machine learning-based intrusion detection system can be evaded to compromise the security system by way of encoding the attack payload, and the target of the data can be able to decode it. Similarly, it could be that the attackers' mission is to cause a concept drift from the security system that can lead to retraining continuously, as such, reduce the performance of the system significantly [38]. Machine learning security model can be compromised by insidious attack such as the backdoor key or Trojan attack. In this case, the machine learning model can be carefully adversary poisons by inserting a backdoor key to distort the model to malfunction when the model detects the presents of the backdoor key but it still works well on the standard dataset [39]. The false negative of a learning system can be compromised to cause denial of service. The false negative can be exploited to compromise the integrity of the learning system. There is a lack of systematic mechanism to measure the actual quantity of risk related to the value of information that is leaked from the learning system to the attackers [40]. The machine learning algorithm trained on a particular mobile malware is biased

toward the malware. The algorithm lacks the ability to detect different mobile malwares with different dynamic behaviors and characteristics, therefore, bypass the machine learning algorithm to penetrate the system [41]. A learning system for detecting security bridge can be obsolete as computer technology evolves because the scale and characteristics of the data will remain the same, while computer technological advancement changes the pattern of the crime leading to a new data with different scales and characteristics. Therefore, the efficiency and effectiveness of the learning system will be degraded paving way for bypassing the security system [42]. The conventional machine learning algorithms have high computational cost because they require independent feature extraction technique before feeding the extracted features to the shallow machine learning algorithms. As the dataset increases, the performance of the shallow machine learning algorithms decreases. The performance of the algorithms lacks consistency across different dataset in view of the fact that the performance of the algorithms depends on the nature of the dataset among other factors. Despite the limitations of the machine learning-based security system as discussed, it has a lot of advantages as follows:

The machine learning algorithms have proven to be feasible in improving the security of mobile touch screen devices; the major advantages of machine learning algorithms in mobile phone security includes ability to accept biometric and sensor data to improve security of mobile phone. The shallow machine learning algorithms work effectively on small size dataset and require less convergence speed. Most machine learning algorithms require very little authentication time which makes it convenient for users as mobile phone users tend to avoid security that takes time before authentication. Machine learning algorithms have proven to be reliable in improving security of mobile phone [43], and therefore provide end point security of the mobile device and offer fast and secured user login.

The machine learning can use data from user profile, network traffic, services, access events, etc., to develop a system with different profiles that define normal and abnormal patterns. The learning system triggers security alarm to call the attention of expert if it detects abnormal pattern. Therefore, it prevents activities that are strange on the system. The international mobile subscriber identity-catcher can be used to violet the privacy of mobile user by surveillance of mobile phone user, interception of calls and mobile SMS. Machine learning can be applied to develop network-based international mobile subscriber identity-catcher detector to detect the presence of international mobile subscriber identity-catcher [44]. The machine learning algorithms is used to develop security authentication system of mobile phone devices that are very strong and difficult to penetrate based on gesture (e.g., finger gesture, finger typing gesture, gesture behavior, etc.), behavior (e.g., finger touch behavior, hand geometry and behavioral information, thumb stroke, etc.), screen touch (e.g., touch interaction, touch operation, touch pattern, etc.) and rhythm (e.g., tap and slide, etc.).

4 Machine learning procedure for improving the security of mobile phone touch screen devices

The procedure typically started with the acquisition of data from mobile phone touch screen devices or other data gathering device which could be from any manufacturing company, e.g., Samsung, Huawei, LG, etc., and installed with a particular mobile operating system such as Android, iOS or Windows. This kind of data is real-life data but the data can be a benchmark data where it can be collected from a public data repository. Subsequently, the raw data are obtained and stored in a central repository. The next stage involved data engineering where features are extracted using any suitable feature extraction technique to extract only relevant features and discard the irrelevant features. This is to have a quality data as quality data plays an important role on the performance of the machine learning algorithms.

No matter how excellent is an algorithm, low-quality data can reduce the effectiveness and efficiency of the algorithm. The data are partitioned into training and test datasets for several fold cross-validation. Then, modeling is based on the machine learning algorithm using the training dataset and the optimization of the algorithm parameters through trial and error or any other means. The proposed approach based on the machine learning algorithm is validated and evaluated based on performance metrics to measure the effectiveness and efficiency of the approach. At the comparison stage, the performance of the newly propose model is compared with the classical algorithms to show the advantage of the proposed approach before deploying it into the mobile phone touch screen to strengthen its security. The complete procedure is shown in Fig. 3.

5 Application of machine learning algorithms in improving the security of mobile phone touch screen

The application of machine learning algorithms for improving the security of mobile phone touch screen devices is presented in this section. Figure 4 is a taxonomy created based on the applications of the machine learning algorithms to improve the authentication of mobile phone

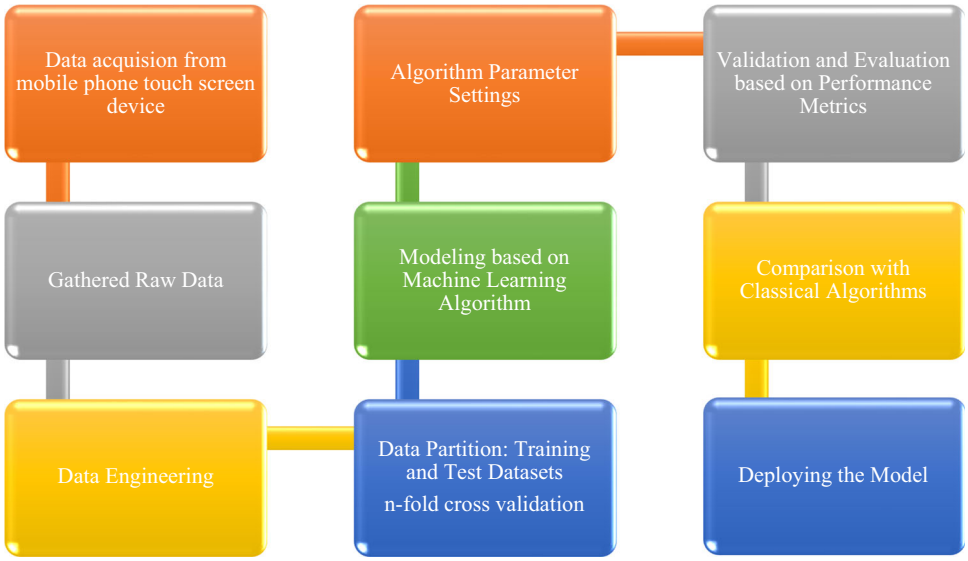

Fig. 3 Procedure for modeling and deploying machine learning authentication scheme

touch screen devices. The classification is obtained from the reviewed papers included in this research.

5.1 Support vector machine

The SVM is one of the algorithms applied to improve the security of mobile phone touch screen. The SVM is applied in different form, for example, [29] combined EEG signal with existing password pattern based to improve the security of a mobile phone touch screen. The EEG signal of the users is recorded concurrently while unlocking the device through the password pattern. Hidden Markov model (HMM) and SVM were used for the authentication where HMM is used to model the EEG signal and SVM is used to build the classifier for authentication. Results of the experiment indicated that the algorithm of the proposed SVM–HMM classifier performs better than the Naïve Bayes (NB) and cosine similarity. The system is limited to authentication of remote user using brain signal transmitted over a secured network. Shahzad et al. [43] proposed the use of support vector distribution estimation (SVDE) to develop a gesture-based authentication scheme for mobile devices. Gesture samples were collected as the specified gesture is displayed on the screen. The SVDE is used to build a classifier based on the multiple gestures data collected from the user. Experimental result shows that the proposed authentication scheme achieves a high classification accuracy than the existing swipe schemes. The proposed method is limited to small size of training samples. Frank et al. [45] proposed the use of KNN and SVM to determine if a classifier can be used for continuous authentication based on user interaction with mobile phone touch screen devices. The framework trains the user touch screen interaction data using KNN and SVM to develop the continuous classification model and evaluate the performance of both the KNN and SVM classifiers. The experimental result shows that the SVM always provides better accuracy than the KNN. The proposed authentication scheme does not consider attributes such as location, accelerometer data and images from the front camera which might have affected the accuracy of the scheme.

Bo et al. [46] proposed the use of SVM to build a classification model based on touch behavior biometrics of mobile phone users. The developed classification model updates the SVM model by adding new observed features through self-learning to improve the classification accuracy. Results show that the proposed authentication scheme is fast and accurate in identification and consume less power. The proposed method only considers behavioral features, neglecting context features such as running application context. Saravanan et al. [47] evaluated a set of machine learning classifiers, namely SVM, RF, NB, J48 and BN classifiers to determine the optimal model for both unary and multi-class authentications. The classifiers were trained to build the classification model based on touch interaction data with user interface elements. Results of the experiment show that the SVM and BN classifiers are optimal for unary and multi-class classification, respectively. The study does not investigate how touch data can be influenced by different external features such as usage while standing or walking. Ehatisham-ul-Haq et al. [48] employed three different algorithms to build three classifiers: SVM, DT and KNN for user authentication. The algorithms were trained to build a classification scheme based on physical activity performed by different mobile phone users for their recognition. Validation and evaluation were conducted for the three classifiers to determine the best among the classifiers. It is found that the overall performance of SVM in recognizing users in five different phone positions is better than the DT and KNN.

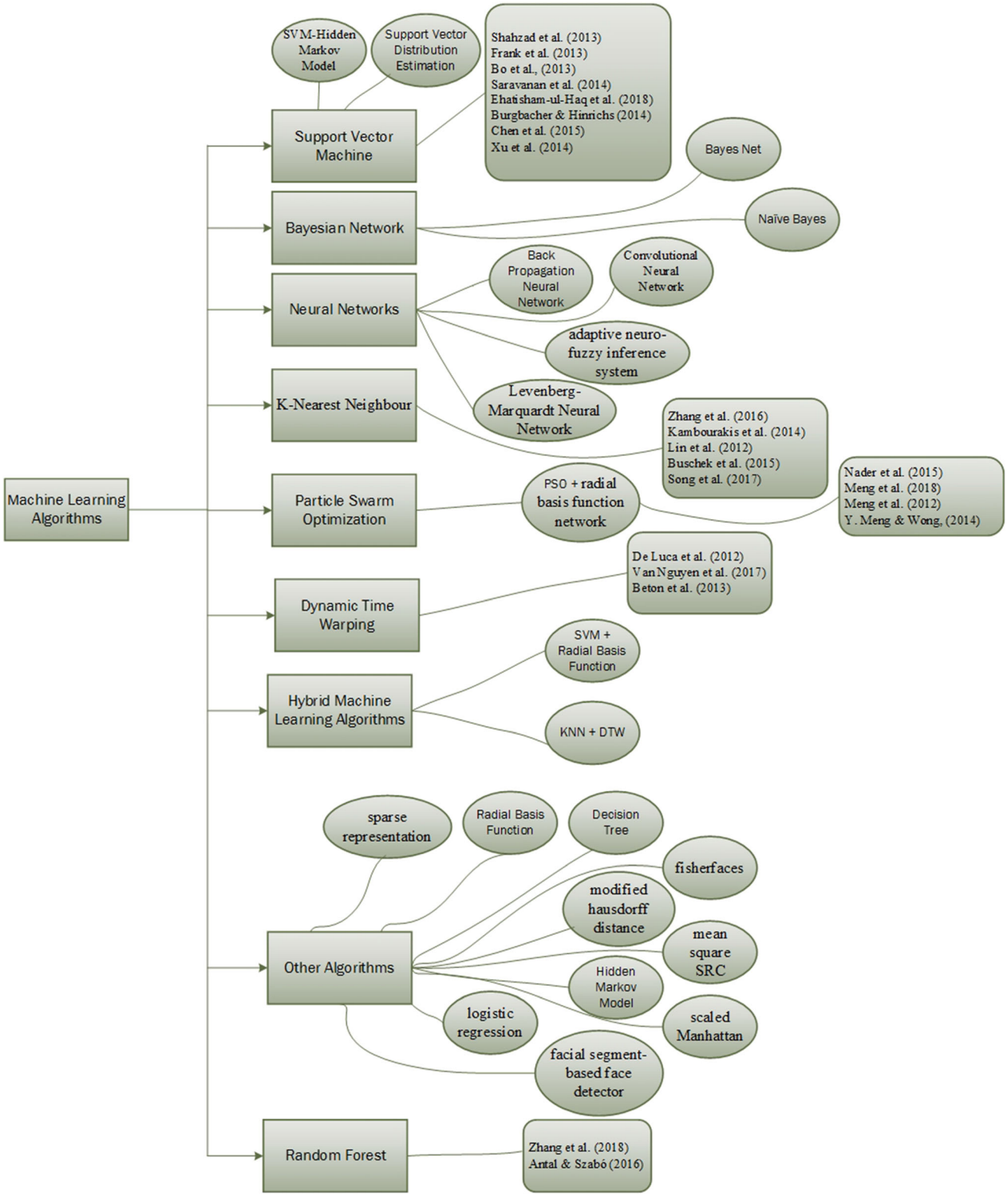

Fig. 4 Taxonomy of machine learning algorithms in improving the security of mobile devices

The proposed work did not consider physiological and motion sensors as well as contextual information for the authentication.

Burgbacher and Hinrichs [49] proposed the use of SVM to create a classifier that verifies if a message was written by a genuine user using user's gesture typing behavior. The

SVM classifier was built based on behavioral biometrics from user's fingertips movement on a gesture keyboard. The experiment results indicated that the SVM classifier can use gesture typing behavior to distinguish genuine users from imposters on a touch screen smart phone. The proposed framework did not investigate more test subjects, and no comparison was made to other classifiers to evaluate the performance of the scheme. Chen et al. [50] proposed SVM to develop an authentication framework based on taps and slide rhythm of users on touch screen. The SVM classifier is trained using the taps/slide rhythm of the users to create an SVM model. Comparisons are made between the taps, slide authentication and single-finger and multiple-finger authentications. Results shows that the finger tap provides a better performance than finger slide, and multiple-finger authentication is more effective than single-finger authentication. The proposed framework performs better only when the user has a better rhyme memory, and the SVM was not compared with other classifiers to evaluate the performance. Xu et al. [51] proposed the use of SVM with RBF kernel function to build a continuous authentication framework based on user's operations on mobile touch screen. Touch operation data were collected concurrently as the user performs basic operations. The SVM–RBF was trained on the collected data to develop a continuous classification scheme. Results show that the touch behavioral biometric can be used as a promising technique for continuous authentication. The proposed approach lacks an optimized adaptation technique to the unstable collected data lacks in comparison with other classifiers. Table 2 presents summary of the mobile phone touch screen authentication scheme based on SVM classifiers.

5.2 K-nearest neighbor

The KNN is applied to strengthen the authentication scheme of mobile phone touch screen devices. For example, [52] evaluate the performance of DT, NB, SVM, ANN, KNN-DTW, random forest (RF) and AdaBoost (AB) in building authentication models for continuous and transparent authentication based on user's behavioral touch dynamics on QWERTY keyboard. The classifiers were used to authenticate users by monitoring their interaction with touch screen of mobile device on Escape keyboard. Performance of the classifiers was evaluated, and the results show that the KNN is more effective for continuous authentication than the compared algorithms. This

Table 2 Summary of the authentication schemes that applied SVM as the classification algorithm

References	Algorithm	Comparison Algorithm	Findings	Limitation
[29]	SVM-HMM	NB and cosine similarity	The proposed SVM-HMM outperform the NB and cosine similarity	The proposed system is limited to authentication using brain signal over a network
[43]	SVDE	Swipe schemes	The proposed method achieves a high classification accuracy than the existing swipe scheme	The proposed authentication scheme is limited to small size of training samples
[45]	SVM	KNN	The SVM achieves better accuracy than the KNN	The proposed authentication scheme does not consider attributes such as location, accelerometer data, etc., which might have affected the accuracy of the scheme
[46]	SVM	Not reported	The proposed method consumes less power while proving fast and accurate authentication	The proposed method neglects context features which may have an influence on the accuracy of the model
[47]	SVM, BN	NB, J48, RF	The SVM proves optimal in unary class authentication, while BN proves optimal for multi-class authentication	The proposed method has limited investigations on how touch data can be influenced by external factors
[48]	SVM	DT, KNN	The SVM has better performance than the DT and KNN in different phone positions	The proposed approach explores only a certain number of sensors and did not consider user contextual information
[49]	SVM	Not reported	The SVM can use gesture typing behavior to distinguish smart phone users	The proposed study uses limited number of subjects
[50]	SVM	Not reported	The SVM classifier shows that the taps and multiple-finger authentication proves to be better than slide and single-finger authentication	The proposed frameworks only provide better performance when the user has a better memory for rhymes
[51]	SVM	Not reported	The SVM shows the potential of continuous authentication using touch biometrics	The proposed scheme lacks optimized method to handle unstable data and no evaluation

Table 3 Summary of the authentication schemes that uses K-nearest neighbor

References	Proposed algorithm	Comparison algorithm	Findings	Limitation
[52]	KNN	DT, NB, SVM, NN, KNN, RF and AB	The KNN proves more effective for continuous authentication than the compared algorithms	This approach is limited to investigate how usability of text entry can be improved to solve the usability–security trade-off
[53]	KNN	RF, MLP	The KNN outperforms the compared algorithms	The study lacks the ability to automatically adjust touch-struck behavior to different touch screen keyboards
[54]	KNN	Not reported	Behavioral biometrics obtained from orientation sensors can be used to improve security of mobile devices	The proposed authentication scheme uses small number of participant and did not consider position of the user such as standing
[56]	KNN	KNN, GM, LSAD, NB and SVM	Classification algorithms are more powerful than anomaly detectors in authenticating users based on typing behavior	The proposed model didn't explore touch-specific features to improve the model performance
[32]	KNN	SVM	KNN provide better performance in all the cases than SVM	The proposed study analyzed small number of gestures and used small number of subjects for the experiment

approach is limited to investigate how usability of text entry can be improved to solve the usability–security trade-off. Kambourakis et al. [53] compared three classification algorithms, RF, KNN and multilayer perceptron (MLP), to determine the most appropriate classifier to build a classification model based on touch-struck dynamics. The algorithms were used to build user's biometric profile in respect to user's typing pattern for authentication. The three classifiers were evaluated. Results of the experiment show that the KNN is superior in terms of both efficiency and performance compared to RF and MLP. The study lacks the ability to automatically adjust user typing behavior to different soft keyboards.

Lin et al. [54] proposed KNN to determine the feasibility of authenticating mobile phone users using behavioral biometrics extracted from orientation sensors of mobile devices. The KNN classifier is trained on the behavioral features to construct an authentication mechanism. The results of the experiment suggested that the proposed KNN classifier achieves a low error rate. Therefore, it can be integrated with the existing mechanism to provide more robust authentication for handheld devices. The proposed approach did not consider user's different positions, and it was not compared with other classifiers. However, the typical practice is to compare the performance of a propose algorithm with the classical algorithm to show its advantage, or otherwise, detail can be found in [55]. Buschek et al. [56] proposed three anomaly detectors, namely KNN, Gaussian model (GM) and least squares anomaly detection (LSAD), and three classifiers are based on KNN, NB and SVM to improve continuous authentication accuracy based on user's key typing behavior. The KNN, GM and LSAD are used to create an anomaly detection model while KNNC, NB and SVM are used to build an authentication framework in respect of combining temporal and spatial touch features. The classifiers and the anomaly detectors were evaluated. Results indicated that the classifiers are more efficient and powerful than the anomaly detectors. The proposed model did not explore touch-specific features such as typing a text message freely.

Song et al. [32] proposed KNN and SVM to develop a reliable, simple and fast authentication system using user's physiological and behavioral biometrics on multi-touch devices. Legitimate user's hand geometry and behavioral information are used to build a one-class classification model using the SVM and KNN. Result of the experiment shows that KNN outperform SVM in almost all the different cases even though, the proposed framework used limited number of experimental subjects and analyzed small number of gestures. The summary of the studies is given in Table 3.

5.3 Bayesian network

The BN has been applied to improve the security of mobile phone touch screen devices. A number of articles were found to be used for the security authentication. For example, [57] evaluated the application of three classification algorithms in mobile phone touch screen authentication, and the algorithms are as follows: BN, decision tree (DT) and RF. The algorithms were used for building classifier models based on a multi-touch finger gestures dataset to improve security for mobile devices. The three classifiers DT, RF and BN were evaluated. Result of the experiment shows that the BN classifier was chosen because it outperforms RF and DT. The proposed authentication method is limited to user's finger gestures without considering other common gestures. Feng et al. [58]

Table 4 Summary of the authentication schemes that used Bayesian network

References	Algorithm	Comparison Algorithm	Findings	Limitation
[57]	BN	DT, RF	The BN classifier performs better authentication in terms of FRR than RF and DT	The proposed authentication method is limited to user's finger gestures without considering other common gestures
[58]	BN	DT, RF	The BN classifier is faster and effective in detection of an intruder than the compared DT and RF classifiers	The proposed authentication scheme does not consider the effect of different conditions on the touch behavior of users
[59]	BN	SVM, KNN	The BN has minimum computational cost and provide the best identification accuracy than SVM and KNN	The proposed work could not provide different access levels to different users based on their identified behavioral traits
[60]	NB	Not reported	The NB provide more accuracy on touch behavior and has less computational complexity than other classifiers	The proposed framework uses limited number of touch features and machine learning classifiers

evaluated the effectiveness of three classification algorithms BN, DT and RF on improving security system of mobile phone touch screen devices. The three algorithms were used to develop a classification model to classify a user based on virtual key typing behavioral biometric. The performance of the algorithms was evaluated. It is found that the BN classifier detects unauthorized user more effectively and faster than the DT and RF. The proposed authentication scheme does not consider the influence of different circumstances on the touch behavior such as typing while working.

Khalidd [59] proposed BN, SVM and KNN to compare and evaluate the performance of the three algorithms on user authentication. The classifiers were trained to develop an authentication framework for identifying mobile users based on the physical activity they performed. The three classifiers accuracy was compared. Results show that the BN provides the best accuracy rate and is computationally cheap compared to the SVM and KNN. The proposed approach could not provide different access level to different users once recognized based on their behavioral biometrics. Kolly et al. [60] proposed NB to build an authentication framework to determine if a user can be identified based on touch screen behavior. The NB classifier was trained on a number of features to create a classification profile based on the way user touch a screen. Results show that the NB provide the good performance and has a low computational complexity. The proposed concept uses a smaller number of touch dynamics features. Table 4 summarized the BN authentication scheme for the mobile phone touch screen devices.

5.4 Random forest

The RF is one of the algorithms used by researchers to enhance the security of the mobile phone touch screen devices. For example, Zhang et al. [61] evaluated three classification algorithms: RF, SVM and NB to determine the most efficient and effective algorithm for classification. Based on the magnetometer data and touch screen sensor data, the RF, SVM and NB were used to build classification models and evaluated. Results suggested that the RF provides better performance than the SVM and NB. This work is limited to exploit implicit authentication techniques for device identification.

Antal and Szabó [62] evaluated two sets of classifiers: one-class classifier comprising of Parzen density estimator (PDE), Gaussian mixtures method (GMM), support vector data description method (SVDDM) and the KNN. Secondly, two-class classifier consisting of RF, BN and KNN build an authentication system for mobile device. The algorithms were used to develop an authentication model based on touch behavioral biometric of mobile phone users. The classifiers were evaluated on multiple datasets. Experimental result shows that the RF and SVDDM provide the best performance for two-class classifiers and one-class classifiers, respectively. The proposed study did not harmonize user personality type with user-specific error rate to obtain the best authentication performance. Table 5 presents the summary of the studies.

5.5 Artificial neural networks

The ANN is one of the powerful machine learning algorithms that received tremendous attention from the research community. The ANN is found to enhance the security of mobile phone touch screen devices. The ANN exists in different forms. For example, [63] proposed adaptive neuro-fuzzy inference system (ANFIS) classifier to improved pattern password security of touch screen mobile devices. The ANFIS is used to build a classification model using touch duration as behavioral traits. Evaluations of the ANFIS classifier were conducted. Results indicated that the ANFIS enhance the security of the mobile phone touch

Table 5 Summary of the authentication schemes that used Random Forest

References	Algorithm	Comparison Algorithm	Findings	Limitation
[61]	RF	SVM, NB	The RF provides the best performance among the three algorithms	The proposed authentication scheme is limited to expanding the application of implicit authentication scheme for device identification purpose
[62]	RF	PDE, GGM, NN, BN, KNN and SVDDM	The RF and SVDDM proves to provide better performance for one-class and two-class classifications, respectively	The proposed study did not correlate user personality type and user-specific error rate to obtain the best performance for both classes

screen compared to ANN and red green blue (RGB) histogram-based. The proposed scheme did not consider other features such as touch pressure and touch stroke interval. Zhouet al. [64] proposed backpropagation ANN (BPNN) to improve the security of a mobile phone touch screen devices. The BPNN is used to create an authentication scheme using thumb stroke behavior. The performance of the BPNN classifier is evaluated, and the authentication models were compared with keystroke dynamics authentication. It was found that the BPNN and RF classifiers provide better accuracy than the DT, NB, SVM, KNN-DTW, RF and AB in all the different settings. The proposed classifier provides an enhance security and usability than keystroke dynamics model. The scheme is limited to reduce the password complexity on the security and usability of the model.

Alpar and Krejcar [65] proposed Levenberg–Marquardt ANN (LM-NN) to create an improved pattern password authentication system using touch location as biometrics. The touch location data are used to train the LM-ANN and Gauss–Newton algorithms ANN (GN-ANN) to build the classification models. Result of the experiment shows that the LM-NN is better and faster in authentication than the GM-ANN. Though the ANN has the possibility of been stuck in local minima, Liang et al. [34] proposed convolutional neural network (ConvNet) to predict tap sequence and application usage behavior of users. Sensor data were collected as users interact with the device on different applications, and a classification model is build using the sensor data based on the ConvNet, SVM, KNN and DT. Evaluation was conducted, and results show that the ConvNet provide better compared to the SVM, DT and KNN. The proposed approach did not consider more complex CovNet model to obtain stable and improved results. The summary of the studies that applied ANN for enhancing security of mobile phone touch screen devices is presented in Table 6.

5.6 Dynamic time warping

The DTW has been found to improve the security of the mobile phone touch screen devices. Few numbers of the DTW were applied for the authentication. For example, [22] deployed DTW to determine if implicit authentication works based on how a user enters a pattern password. The DTW was used to build an authentication model to analyze and find the similarities between two behavioral sets of data. Experimental result shows that the implicit authentication based on how user performs a pattern password on a touch screen smart phone can be used to improve mobile phone security. The proposed study was not compared to other approaches to determine its effectiveness. Nguyen

Table 6 Summary of the authentication schemes that used Artificial Neural Networks

References	Algorithm	Comparison Algorithm	Findings	Limitation
[63]	ANFIS	ANN, RGB Histogram	The ANFIS produce the lowest error rate than the compared ANN and RBG histogram	The proposed model considers just small number of features
[64]	BPNN	DT, NB, SVM, KNN-DTW, RF and AB	The ANN and RF outperform the other classifiers	The proposed study could not moderate the complexity of password on security and usability of the scheme
[65]	LM-NN	GM-NN	LM-ANN provide better performance in both optimization and authentication than the GM-NN	The ANN has the possibility of been stuck in local minima
[34]	ConvNet	SVM, KNN and DT	ConvNet provides the best accuracy in predicting user's tap behavior than the compared algorithms	The CovNet require too much layers to get the complete features hierarchy

Table 7 Summary of the authentication schemes that used Dynamic Time Warping

References	Algorithm	Comparison Algorithm	Findings	Limitation
[22]	DTW	Not reported	Drawing pattern behavior proved promising for implicit user authentication	The proposed approach was not compared with other approaches
[30]	DTW	Not reported	Pin drawing behavior can be used for smart phone user authentication using DTW	The proposed scheme is limited to using small number of subjects and no evaluation
[66]	DTW	PC	DTW proves to be more efficient than PC in terms of user authentication	The proposed model did not investigate more algorithms, features and benchmark datasets to improve the scheme accuracy

et al. [30] applied DTW to develop a scheme that authenticates users based on how a user draws a PIN on a touch screen instead of typing it. The DTW algorithm is used to compare and find the similarities between the two PINs drawing sample of users based on the user's PIN drawing behavior. Results of the experiment confirm that the PIN drawing behavior of users can be used for user verification, and the DTW can provide a promising performance to support the proposed model. The proposed study uses limited number of experimental subjects and was not compared with other classification algorithms for evaluation.

Beton et al. [66] employed DTW to construct a biometric authentication system which combines pattern password representation and the associated behavior while drawing it. The DTW is deployed to compare and measure the similarities between two sets of behavioral biometrics data to determine successful authentication. Experimental result shows that the DWT proves to be more efficient in authenticating users than the Pearson correlation (PC). The proposed method did not explore other matching features. Table 7 shows the summary of the studies that applied DTW for mobile phone touch screen authentication.

5.7 Particle swarm optimization

The PSO in this domain is not used alone but in hybrid form with radial basis function network, a class of ANN. The PSO is a nature-inspired algorithm from the swarm intelligence. The PSO is found to enhance the effectiveness of the RBFN, and hence improves the security of the mobile phone touch screen. For example, Nader et al. [67] proposed hybridized PSO and radial basis function network (PSO-RBFN) to develop an authentication scheme based on mobile touch screen user's behavioral gesture. The touch gesture behavioral biometrics were collected, while users were interacting with the mobile phone. The PSO-RBFN is applied to build a classifier for the recognition of the authenticate user's profile. Result of the experiment shows that the PSO-RBFN classifier performs better authentication than the RBFN, DT, J48, NB, BPNN and repeated incremental pruning (RIP) classifiers. The proposed system is limited to a smaller number of participants, and unadjusted features were used. Meng et al. [68] applied PSO-RBFN to build a classification model based on user's behavioral touch gesture on web browser. The PSO optimizes the weighted sum of the RBFN to enable the classifier handle variation in behavioral touch data. The user authentication model was created based on PSO-RBFN. Results suggested that the PSO-RBFN achieves a better authentication accuracy in handling unstable data than the J48, NB, Kstar, RBFN and BPNN. The proposed study is restricted to usage in only web browser running application context.

Meng et al. [69] applied PSO-RBFN to create an authentication scheme based on touch dynamics of users on mobile phone touch screen devices. The RBFN was optimized by PSO to build a classification model that deals with variation in user's behavioral biometrics data. The performance of PSO-RBFN was explored on the collected touch behavior data. Experimental result shows that PSO-RBFN is faster and provides the best authentication accuracy compared to J48, NB, Kstar, RBFN and BPNN. The proposed research work did not explore the scheme on other mobile operating systems; it is only restricted to android. Meng et al. [70] evaluated five classification algorithms, namely J48, NB, RBFN, BPNN and PSO-RBFN to determine which classifier best suit a touch dynamics authentication scheme. The authentication model was build using the five classifiers based on user's touch behavior using minimum number of features to reduce processing time. The performance of the algorithms was evaluated. The results show that the PSO-RBFN provides the best authentication performance than the compared classifiers. The proposed study is restricted to the behavioral biometric only and did not consider combining with other traditional method to improve performance. Table 8 presents the summary of the studies that used PSO-RBFN for improving the authentication scheme of a mobile phone

Table 8 Summary of the authentication schemes that hybridized Particle Swarm Optimization with Radial Basis Function Network

References	Algorithm	Comparison Algorithm	Findings	Limitation
[67]	PSO-RBFN	RBFN, J48, NB, BPNN and RIP	The proposed PSO-RBF classifiers perform better than the compared classifiers	The proposed system is limited to a smaller number of participant and unadjusted features were used
[68]	PSO-RBFN	J48, NB, Kstar, RBFN and BPNN	PSO-RBFN is more accurate in authentication than J48, RBFN, BPNN, NB and Kstar	The proposed scheme restricts its usage in web browser only ignoring other running application context
[69]	PSO-RBFN	J48, NB, Kstar, RBFN and BPNN	PSO-RBFN is faster and more accurate in authentication than the compared classifiers	The proposed scheme did not consider the feasibility of the scheme on other platforms
[70]	PSO-RBFN	J48, NB, BPNN and RBFN	The PSO-RFBN authentication scheme based on touch dynamics provide the best performance than the compared classifiers	The proposed approach could not combine the behavioral based and other traditional authentication method to improve user authentication

touch screen devices. Table 8 presents the summary of the studies that applied PSO-RBFN.

5.8 Hybrid machine learning algorithms

To enhance the effectiveness of the algorithm for application in enhancing security of mobile phone touch screen, the algorithms are hybridized. The machine learning algorithms are combined together or machine learning with a conventional technique. For example, [71] hybridized SVM and RBF to build a classifier for authenticating mobile phone touch screen device users. The SVM-RBF classifier is build based on multiple facial attributes extracted from the mobile device users. The efficiency of the SVM-RBF classifier is evaluated. The result shows that the SVM-RBF classifier has more robustness, consumes less memory and performs better under different conditions than the compared PCA-RBFSVM and local binary pattern (LBP). The proposed approach lacked the capability of adapting to attribute change of the user such as aging. Feng et al. [72] combined one KNN (1KNN) and DTW (1KNN-DTW) to develop authentication scheme for mobile touch screen device. The DTW measures the similarities between two time series of touch input data, while the one KNN is used to build the classifier for identity recognition. Result of the proposed authentication scheme shows the potential of achieving a strong touch-based identity recognition in natural usage and provides better accuracy rate than the current user-input authentication such as password and pattern. The proposed scheme could not provide a trade-off between an effective continuous authentication and minimum computational cost. Table 9 presents the summary of the studies.

5.9 Other machine learning algorithms

This section presents algorithms used for enhancing mobile phone security that could not be classified under any of the algorithms discussed in the previous sections. As a result of that those algorithms are placed under this section as other algorithms. For example, [73] evaluated the performance of five classification algorithms, namely J48, ANN, RBF, BN and RF to determine the classifier that can provide the best authentication based on user's acceleration data on mobile phone. The acceleration data were collected as the user move while holding the phone in three states: in

Table 9 Summary of the authentication schemes that used hybridized machine learning algorithm

References	Algorithm	Comparison Algorithm	Findings	Limitation
[71]	SVM + RBF	LBP, PCA + RBFSVM	The proposed authentication method shows more robustness and consume less memory than the compared authentication method	The propose approach lack the capability of adapting to attribute change of the user such as aging
[72]	1KNN-DTW	password and pattern	Experiment shows a strong implicit authentication scheme can be achieved in natural usage	The proposed method could not provide a balance between providing minimum computational cost and strong continuous authentication

pocket, making a call and touching the screen. The classification model was built by the five classifiers, and their performance was evaluated. Results confirmed that the RBF is the best, and J48 provides the worst performance. The proposed research work did not consider the phone's orientation in estimating the holding state to authenticate users. Crawford et al. [74] proposed DT to construct a continuous transparent authentication framework on mobile phone touch screen devices based on keystroke dynamics and voice utterances of the users. The two behavioral biometric traits are fed to the DT classifier to build the legitimate user model that can classify feature vector for the two biometric traits. Results show that the proposed system minimizes explicit authentication by less disruption of users unless it is unsure of the user's identity. The proposed model was not compared with other classification scheme and did not consider how complex interaction with the mobile device may affect the security of the device over time.

Mahbub et al. [75] proposed facial segment-based face detector algorithm (FSFD), RF and mobility Markov chains (MMC) to construct a multimodal active authentication scheme using multiple mobile phone sensors data. The FSFD is used to build a model for detecting partial faces, and the RF classifier is trained on a multimodal dataset to create a classification model based on swipe behavior, while the MMC is used to model the mobility behavior of user from one state to another. The algorithms were compared and evaluated with other algorithms. Results indicate that the FSFD outperforms Viola-Jones (VJ), deep pyramid deformable part model (DPDPM) and deformable part-based model (DPM), while RF performs better than the RBFSVM, KNN, NB, LR and gradient boosting model (GBM). The proposed study provides average high error rate which can possibly affect the overall performance of the model. Roy et al. [76] proposed HMM algorithm to construct a continuous user authentication which requires training data from the smart phone owner using touch behavior. The behavioral profile of the user is built based on user's touch behavior pattern using the HMM. The HMM algorithm is compared to touch analytics algorithm. Result shows that HMM provides better performance in different test scenarios than the compared algorithm. The proposed study did not evaluate the HMM on multiple dataset to determine its feasibility. Fathy et al. [77] evaluated four still image-based classifiers involving fisherfaces (FF), sparse representation (SRC), large margin nearest neighbor (LMNN) and eigenfaces (EF). In addition, five image set-based algorithms, namely convex hull-based image set distance (CHISD), affine hull-based image set distance (AHISD), sparse approximated nearest points (SANP), dictionary-based face recognition from video (DFRV) and mean square SRC (MSSRC) were evaluated to determine the best algorithms for face authentication using video extracted from front camera of smart phone. The algorithms were used to detect faces using the video recordings from front camera. The algorithms were evaluated, and the results show that the FF, MSSRC, SRC provide the best performance in authentication than the compared algorithms. The proposed method could not investigate the best features that can deal with variations in data.

Jain and Kanhangad [78] proposed modified Hausdorff distance (MHD) to build an authentication model using behavioral traits extracted from smart phone orientation and acceleration sensors. The MHD is used for matching of extracted features from multiple gestures performed by the users. The MHD performance is compared with DTW. The outcome suggested that the MHD outperforms DTW for all the considered gestures. Though the proposed model uses limited gesture samples per user and limited experimental subjects which may have an effect on the performance, Serwadda et al. [79] evaluated the performance of ten classification algorithms, namely SVM, NB, BN, RF, KNN, J48, ANN, logistic regression (LR), scaled Manhattan verifier (SMV) and Euclidean verifier (EV) under a common experimental scenario to understand the potential of touch biometrics for touch screen authentication. The ten algorithms were trained on a touch biometric dataset to build an authentication scheme. The performance of the algorithms were analyzed and evaluated. Results show that the LR has the lowest error rate, and therefore provides the best performance than other compared algorithms. The proposed scheme did not explore how multiple biometric traits of poor users can affect the performance of the scheme. Sitová et al. [80] deployed scaled Manhattan (SM), scaled Euclidian (SE) and SVM to develop a framework for authenticating smart phone users continuously with respect to user's behavioral traits in two different conditions. The SM, SE and SVM algorithms are used to build a model that will authenticate users based on user's hand movement, grasp and orientation. Experimental result shows that SM produces the lowest error rate in both sitting and walking condition of the user. The proposed study did not investigate the accuracy of the scheme on multiple usage conditions, weather conditions and application context. Table 10 shows the summary of the other algorithms used for improving the security of mobile phone touch screen devices.

6 Feature engineering

Feature engineering is the process of transforming collected raw data into non-redundant features which provides distinctive human information that can be used for

Table 10 Summary of the authentication schemes that used other algorithms

References	Algorithm	Comparison Algorithm	Findings	Limitation
[73]	RBF	J48, NN, RF, BN	The RBF is the best classifier for authentication based on user acceleration and J48 is the worst	The proposed system neglect phone orientation in estimating the holding state for authentication
[74]	DT	Not reported	Transparent continuous authentication based on behavioral biometric trait provide trade-off between usability and security	The proposed approach did not consider complex interaction of the behaviors which might affect the security over time
[75]	FSFD	DPDPM, DPM, VJ, RF, MMC, RBFSVM, KNN, NB, LR, GBM	The FSFD provides low processing time than the compared face detection algorithms while RF outperforms the compared classifiers in respect to swipe gestures authentication	The proposed study provides low overall performance due to higher error rate
[76]	HMM	Touch analytics algorithms	HMM provides the best performance in multiple scenarios than the compared touch analytics algorithms	The proposed model did not investigate the feasibility of other newly generated datasets on the model
[77]	FF, MSSRC, SRC	LMNN, EF, CHISD, AHISD, SANP, DFRV,	The FF, MSSRC, SRC provide the best performance in authenticating users from video captured using front camera	The proposed study could not investigate better algorithms that can improve the accuracy by dealing with the data variation
[78]	MHD	DTW	MHD outperforms DTW in matching features gotten from gestures made by the users	The proposed scheme uses limited subjects for experiment and limited gestures per user
[79]	LR	SVM, NB, BN, RF, KNN, J48, NN, SMV, EV	The LR achieves a better accuracy and lower error rate than the compared classifiers	The proposed study did not explore how biometric modalities of multiple poor users can affect the accuracy of the scheme
[80]	SM	SE, SVM	The SM outperform SE and SVM on feature collected in both walking and sitting usage conditions	The proposed approach did not consider multiple usage condition, weather and application context

classification. Complex data require much memory and consume lots of power to process. So, feature engineering is required to reduce the number of the features while maintaining its accuracy and original content. In feature engineering, irrelevant features are discarded because those irrelevant features can increase computational cost and reduce the effectiveness of the algorithms.

Multiple feature extraction techniques exist to help in selecting appropriate and efficient features from a large set of data. Example of the feature engineering techniques involves discrete Fourier transform, stepwise linear regression, fisher score ranking, etc. Machine learning algorithms require quality data to produce quality output. In this survey, a lot of the studies use feature engineering before applying the machine learning algorithm to develop the mobile phone devices security authentication schemes. Table 11 shows the feature engineering techniques used by various studies and the number of selected features, though some studies did not report the number of features and feature extraction techniques, for example [51, 76], etc.

7 Operating systems and mobile device manufacturers

The analysis of the operating systems used by different studies is depicted in Fig. 5 showing the visual representation of the types of operating systems used. Figure 5 indicates that Android operating system has the longest bar signifying that most of the research used Android operating system for their experiment. The iOS and windows operating systems do not attract much attention from researchers. The operating systems shown in Fig. 5 are extracted from the papers that reported their devices operating systems. However, some of the studies did not reveal the operating systems on their devices.

Android operating system is widely used likely because of its flexibility, scalability and is the most common touch screen operating system in mobile devices. Therefore, it is easier to perform certain experiment on Android mobile devices as indicated in the studies. Figure 6 shows the visual representation of the type of mobile device used by researchers. It is indicated that Samsung is the most used mobile device to perform security authentication

Table 11 Summary of the feature engineering algorithms used to select the optimum features

References	Feature engineering technique	Number of features used
[29]	Discrete Fourier Transform	5
[67]	Not reported	14
[71]	VLFeat toolbox	4
[43]	Not reported	7
[57]	Not reported	6
[45]	Correlation coefficient	30
[72]	Not reported	3
[46]	Not reported	4
[47]	Not reported	2
[58]	Not reported	40, 41, 41
[61]	Not reported	16
[68]	Not reported	21
[48]	Not reported	16
[62]	Not reported	22
[59]	Fourier transform	8
[79]	Not reported	28
[73]	Not reported	43
[69]	Not reported	21
[52]	Not reported	6
[53]	Not reported	4
[63]	Not reported	2
[22]	Not reported	5
[60]	Not reported	6
[54]	Stepwise linear regression	53
[70]	Not reported	8
[75]	Not reported	24
[80]	Fisher score ranking	17, 13, 16
[66]	Not reported	5
[64]	Not reported	6
[65]	Not reported	2
[50]	Not reported	15
[77]	Not reported	3
[78]	Not reported	6
[34]	Not reported	10
[32]	Fisher Score	12

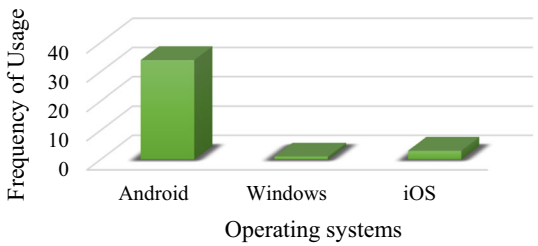

Fig. 5 Operating systems used in different studies

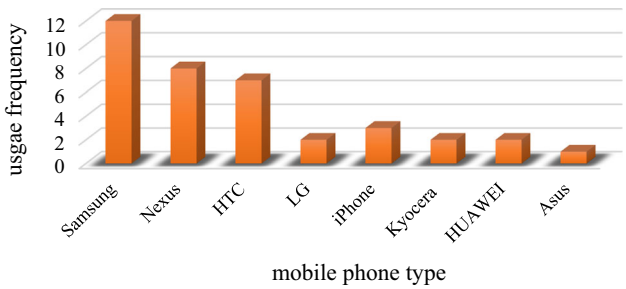

Fig. 6 Mobile phone device manufacturers

experiment because it has the highest bar. The attention attracted by Samsung is likely because it has a powerful embedded sensor and is commonly used among customers. The second most used mobile devices are the Nexus followed by HTC and others.

Table 12 Summary of the datasets found in the survey

References	Data	Data gathering device	Type of data
[29]	EEG signal	Emotive headset	Real-world data
[67]	Behavioral touch gesture	WebTouch and gesture pattern	Real-world data
[71]	Facial attributes	AA01 and MOBIO dataset	Benchmark dataset
[43]	Touch gesture	Touch screen	Real-world data
[57]	Multi-touch gesture	Digital sensor glove and android program	Real-world data
[45, 47, 51–53, 60, 61, 68–70, 76, 79]	Behavioral touch data	Touch screen	Real-world data
[46]	Sensor data	Touch screen	Real-world data
[58]	Virtual key typing data	Touch screen	Real-world data
[72]	Touch and magnetometer data	Touch screen and magnetic ring	Real-world data
[48, 59]	Sensor data	Magnetometer, accelerometer, and gyroscope	Benchmark dataset
[62]	Touch and motion data	Touch screen and motion sensors	Real-world data
[73]	Acceleration and rotation data	Phone accelerometer	Real-world data
[63]	Touch duration data	Touch screen	Real-world data
[22, 30, 49, 50, 54, 56, 66, 74, 78]	Behavioral biometric data	Touch screen	Real-world data
[75]	Mobile sensors data	Mobile phone sensors	Benchmark datasets
[80]	Mobile sensor data	Accelerometer, gyroscope and magnetometer	Real-world data
[64]	Thumb stroke behavior data	Thumb stroke virtual keyboard	Real-world data
[65]	Touch location data	Touch screen	Real-world data
[77]	Video data	Mobile phone front camera	Benchmark dataset
[34]	Sensor data	Mobile phone embedded sensors	Real-world data
[32]	Multi-touch data	Touch screen	Real-world data

In some of the studies, e.g., in [29, 46, 48, 53, 58, 62, 65, 66, 75–77], the mobile devices used were not revealed.

8 Dataset

Datasets are collection of related data corresponding to the content of a particular database table where each column represents a variable (feature) and each row represent the complete data of a particular subject corresponding to an experiment [81]. Different data used for the authentication exist but recent studies show that biometric data are the most used data (this is evident in Table 12) and efficient way to authenticate users.

Biometric data consist of physiological biometrics which involve physical characteristics of a particular subject such as hand, finger, face, iris and others, and behavioral biometrics data define how a user performs different activities such as but not limited to keystroking, gesture drawing, and touch behavior. Behavioral biometrics prove to be the most efficient data for classifying users because it is very difficult to mimic an individual's behavior Shahzad et al. [43].

Datasets are classified into benchmark and real-life data datasets. A benchmark dataset is a ready-made data already processed, no variation of data, and all noise removed ready to be used for experiment, while a real data is obtained by performing a real experiment involving multiple subjects to be performing a particular task as the data are recorded. A real data has to be processed because it may be too large to be fed to the algorithm [82]. Benchmark dataset is easy and less expensive to obtain but lacks the real-world experience, while real-world data are difficult and expensive to gather but the data have the real-world experience. A real-world data requires more resources and takes longer time to obtain because it involves acquisition of devices and multiple experimental subjects. Table 12 presents the datasets used by various projects reviewed in this survey. Table 12 clearly indicated that very few studies used benchmark dataset, while overwhelming majority of the studies gathered real-world dataset for their research work.

9 Performance evaluation parameters

The performance metrics is the measure to determine the effectiveness of the machine learning algorithm used in developing the authentication scheme. Table 13 shows the performance metrics used by different studies as extracted during the survey. Each study has it is own choice of performance metrics; there is no uniformity on the choice of the metrics, and some use 2, 3 or 4 depending on the

Table 13 Performance evaluation metrics used

References	HTER	FRR	FAR	TPR	AER	EER	TAR	FNR	FPR	TNR	AR	RMSE	F-M	RR
[29]	√	√	√	√	X	X	X	X	X	X	X	X	X	X
[67]	X	√	√	X	√	X	X	X	X	X	X	X	X	X
[71]	X	X	√	X	X	X	√	X	X	X	X	X	X	X
[43]	X	X	X	√	X	√	X	√	√	X	X	X	X	X
[57]	X	√	√	X	X	X	X	X	X	X	X	X	X	X
[45]	X	√	√	X	X	√	X	X	X	X	X	X	X	X
[72]	X	X	X	√	X	X	X	X	X	√	X	X	X	X
[46]	X	√	√	X	X	X	X	X	X	X	X	X	X	X
[47]	X	X	X	√	X	X	X	X	X	√	X	X	X	X
[58]	X	√	√	X	X	X	X	X	X	X	X	X	X	X
[61]	X	√	√	X	X	X	X	X	X	X	√	X	X	X
[68]	X	√	√	X	X	X	X	X	X	X	X	X	X	X
[48]	X	X	X	X	X	√	X	X	X	X	√	X	X	X
[62]	X	X	X	X	X	√	X	X	X	X	X	X	X	X
[59]	X	X	X	X	X	X	X	X	X	X	√	√	√	X
[79]	X	X	X	X	X	√	X	X	X	X	X	X	X	X
[51]	X	√	√	X	X	√	X	X	X	X	X	X	X	X
[73]	X	√	√	X	X	X	X	X	X	X	X	X	X	X
[69]	X	√	√	X	X	X	X	X	X	X	X	X	X	X
[52]	X	X	X	X	X	X	X	X	X	X	√	X	X	X
[53]	X	√	√	X	X	√	X	X	X	X	X	X	X	X
[63]	X	√	√	X	X	√	X	X	X	X	X	X	X	X
[49]	X	√	√	X	X	√	X	X	X	X	X	X	X	X
[22]	X	√	√	√	X	X	X	X	X	√	X	X	X	X
[60]	X	√	√	X	X	X	X	X	X	X	X	X	X	X
[54]	X	√	√	X	X	√	X	X	X	X	X	X	X	X
[70]	X	√	√	X	√	X	X	X	X	X	X	X	X	X
[74]	X	√	√	X	X	√	X	X	X	X	X	X	X	X
[75]	X	X	X	X	X	√	X	X	X	X	X	X	√	X
[80]	X	X	X	X	X	√	X	X	X	X	X	X	X	X
[76]	X	√	√	X	X	√	X	X	X	X	X	X	X	X
[30]	X	X	X	√	X	√	X	√	√	X	X	X	X	X
[66]	X	√	√	X	X	√	X	X	X	X	X	X	X	X
[56]	X	X	X	X	X	√	X	X	X	X	X	X	X	X
[64]	X	√	√	X	X	√	X	X	X	X	X	X	X	X
[63, 65]	X	√	√	X	X	√	X	X	X	X	X	X	X	X
[50]	X	X	X	√	X	X	X	X	√	X	X	X	X	X
[77]	X	X	X	X	X	X	X	X	X	X	X	X	X	√
[78]	X	√	√	X	X	√	X	X	X	X	X	X	X	X
[34]	X	X	X	X	X	X	X	X	X	X	√	X	X	X
[32]	X	√	√	X	X	X	X	X	X	X	X	X	X	X

study. What is common among the studies about the performance metrics is that is hardly to find the justification on the choice of the performance metrics. The metrics found in the survey includes half total error rate (HTER), false reject rate (FRR), false accept rate (FAR), true positive rate (TPR), average error rate (AER), accuracy rate (AR), true acceptance rate (TAR), equal error rates (EER), false negative rates (FNR), false positive rates (FPR), true negative Rate (TNR), root-mean-squared error (RMSE), F-measure (F-M) and recognition rate (RR).

The FAR and FRR rate has the highest number of used because the best way to measure the accuracy of an

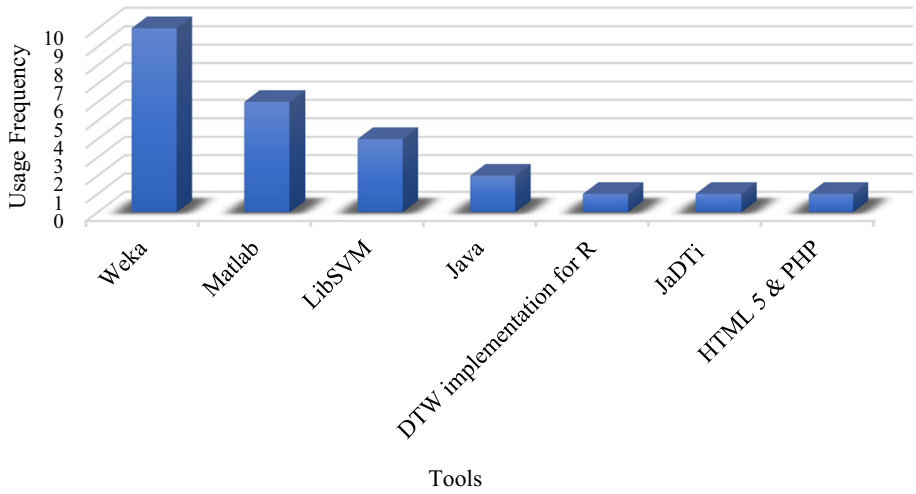

Fig. 7 Visual representation of the implementation platforms

authentication system is by determining the rate at which the system falsely accept a non-genuine user which is FAR, and the rate at which the system falsely rejects a genuine user who should have gain access which is FRR [49] followed by the EER which is the equality between FAR and FRR and also has a significant number of usages.

10 Implementation platforms

The machine learning algorithms for improving the security of the mobile phone touch screen devices are implemented on a platform. The survey shows that different platforms were used for the implementation. Figure 7 depicts the visual representation of the implementation platforms used to conduct experiment by different studies. As indicated in Fig. 7, WEKA is the most frequently used platform followed by MATLAB. Both the WEKA and MATLAB are well-known platforms for implementing machine learning propose conceptual frameworks.

The survey shows that some of the studies do not report platform used for their implementation. For example, the following studies did not report their implementation platforms: [29, 34, 45, 46, 48, 54, 56–59, 61, 64, 66, 75, 80].

11 Domain of applications

The machine learning algorithms were found to be applied in different domain within the area of the mobile phone security authentication. Table 14 shows the domain column with the corresponding feature, algorithm and reference. It is clearly indicated that biometrics received tremendous attention from researchers, and passwords seem to be relegated to the background.

12 General discussion of the research area

The survey reported the applications of machine learning algorithms to improve the security of mobile phone touch screen devices, (see Tables 2, 3, 4, 5, 6, 7, 8, 9, 10). Feature engineering (Table 11), operating systems (Fig. 5), manufacturers of the mobile phone devices (Fig. 6), performance metrics (Table 13), dataset (Table 12) and implementation platforms (Fig. 7) are discussed in the survey. The SVM, KNN, FR, ANN and DTW are the main machine learning algorithms that received tremendous attention from the research community. It is found that the machine learning algorithms can significantly improve the authentication scheme of the mobile phone touch screen devices for different operating systems and mobile devices from different manufacturers. Those machine learning algorithms work well especially when the dataset is of small size. The algorithms are found to perform on both the real life and benchmark datasets. Those algorithms have minimum computational cost, simple to implement and provide good performance for the security of the mobile devices. The performance of the algorithms varies depending on the data type, e.g., text, audio or images.

Multiple number of researches have been conducted, and the research outputs have been published on yearly bases. Figure 8 depicts the publications of the research outputs in this domain from 2012 to 2018 to show the publication trend.

Figure 8 represents the trend of papers published on improving authentication system of mobile devices in recent years. The trend indicated that mobile phone security authentication is one of the trending research topics in the literature between 2012 and 2018. It also shows that researchers are dwelling into mobile authentication system. The trend is expected to grow faster in the future especially with the new research perspective unveiled in this survey.

Table 14 Summary of the application domain with corresponding feature and algorithm

Domain	Feature	Algorithm	References
Biometric	Finger + EEG signal	SVM	[29]
	Facial	SVM-RBF	[71]
		FSFD, RF	[75]
		FF, MSSRC, SRC	[77]
	Behavioral	KNN	[54]
		RF	[62]
		DTW	[22]
		SM	[80]
	Keystroke dynamics and voice utterances	DT	[74]
	Touch	LR	[79]
Gesture	Finger gesture	SVDE	[43]
		BN	[57]
		MHD	[78]
	Finger typing gesture	SVM	[49]
	Gesture behavior	PSO + RBFN	[67, 68]
Behavior	Finger touch behavior	SVM	[46]
		NB	[60]
		HMM	[76]
		1NN + DTW	[72]
	Key typing	KNN	[56]
		BN	[58]
	Hand geometry and behavioral information	KNN	[32]
	Physical activity	SVM	[48]
		BN	[59]
	Thumb stroke	ANN	[64]
	PIN drawing behavior	DTW	[30]
	Pattern drawing behavior	DTW	[66]
	Acceleration	RBF	[73]
	Tab sequence	ConvNet	[34]
	User interaction	SVM	[45]
Touch	Touch interaction	SVM	[47]
	Touch operation	SVM-RBF	[51]
	Touch dynamics	KNN	[52, 53]
		PSO-RFBN	[69, 70]
	Touch pattern	RF	[61]
	Touch duration	ANFIS	[63]
	Touch location	LM-ANN	[65]
Rhythm	Tap and slide	SVM	[50]

The survey depicted the trend of the machine learning algorithms as shown in Fig. 9.

Figure 9 shows the popularity of each machine learning algorithm in mobile device security authentication. It is found that the SVM has the highest popularity in mobile authentication followed by the KNN. This trend clearly shows that researchers heavily depend on SVM to improve the security authentication of mobile phone devices.

13 Challenges and new perspective for future research

Despite the progress witness in enhancing the security of mobile phone touch screen devices based on machine learning algorithms, there are a lot of room for improvement. In this section, we discussed the challenges facing the research area and suggest new perspectives for future research opportunities. The challenges and the suggestions for future research are as follows:

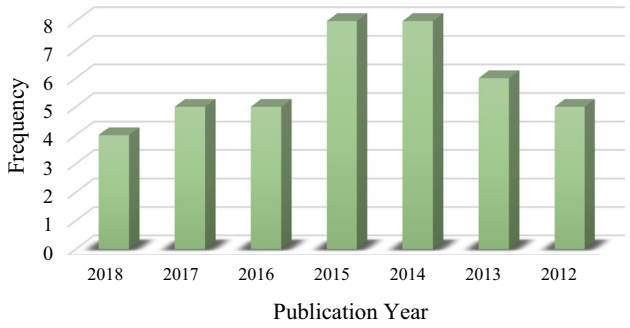

Fig. 8 Publication trend of the published outputs

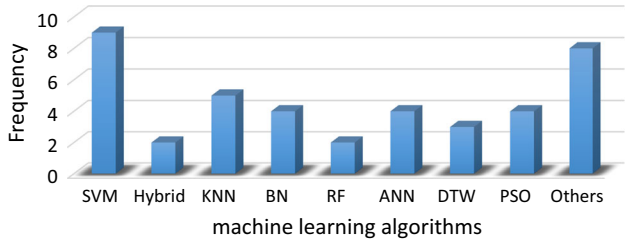

Fig. 9 Trend of the machine learning algorithms

As shown in the survey (see Table 11), the machine learning algorithms used for developing authentication scheme in mobile phone touch screen devices heavily depend on feature extraction method before the machine learning algorithms are applied to the extracted features. This means that before the application of the machine learning algorithm, separate technique for extracting features is required. Therefore, the performance of the machine learning algorithms used in this heavily relied on feature engineering. However, [83] argued that feature engineering might cause biased and lead to the extraction of features that are not complete for the modeling. Inefficient extracted features may lead to awful classification and also affect the accuracy of the scheme. Expert data analysts cannot boost of efficient feature extraction; therefore, effective techniques have to be used to improve feature extraction. The feature engineering makes the process a double work and requires significant human intervention in the process. To eliminate the double work typically practiced by researchers in this domain, we recommend the exploration of deep learning architectures that do not require separate technique for feature extraction. The feature extraction in deep learning is performed automatically, and therefore significantly reducing human intervention in the process. With the exception of [34] that explore ConvNet, it is obvious that the application of machine learning algorithms in mobile phone touch screen devices to improve security of the devices do not witness massive exploration of deep learning. This is despite the excellent performance witness by deep learning in feature selection as shown in [84], yet, its application is highly limited in the domain of mobile phone touch screen devices. The following aspect of the deep learning remain un-explored in improving the security of mobile phone touch screen devices despite their potentials to produce superior and robust solutions: neural abstract machine, generative adversarial network, interaction based deep network, efficient inference technique, deep belief network, attention models, deep recurrent networks, memory augmented neural network, biologically plausible deep network, deep extreme learning machine, stacked autoencoders, few short learning and other innovative deep learning architectures. We urge researchers in the machine learning community to massively explore the potentials of this untapped aspect of deep learning to propose novel authentication system for enhancing the security of mobile phone touch screen devices.

The studies use different operating system for the authentication scheme based on the machine learning algorithm. Figure 4 depicts the operation used by different studies. In most cases, the studies are limited to a particular operating system without considering multiple operating systems when proposing machine learning-based authentication scheme. As a result of that the authentication scheme is limited to only the operating system used for the study, and the result cannot be generalized or operated on another operating system. As such, it can be deployed to similar operating system. For example, the authentication scheme developed for iOS cannot be deployed to Android or Windows platforms. Therefore, compatibility of mobile phone touch screen authentication scheme based on machine learning algorithms across different operating system is significant in improving the security of mobile phone touch screen devices. In the future, we suggest researchers to develop a machine learning-based authentication scheme with potential to operate across different operating system platforms. Hybrid authentication scheme developed with data from different operating system platforms could provide a feasible solution. An authentication system should be able to work across multiple platforms. Therefore, authentication model should be developed and tested across different mobile device and operating system to provide similar level of accuracy. The ability of an authentication system to be compatible to different platform makes it efficient and effective.

Figure 9 indicated that the literature heavily depends on the conventional SVM in proposing authentication scheme for mobile phone touch screen devices. However, SVM has several limitations pointed out as follows: The SVM do not work effectively with noisy dataset. As such, training the SVM to optimize its parameters becomes difficult. Realizing the optimal kernel parameter values has no systematic approach. The accelerated particle swarm

optimization (APSO) is applied to avoid the issues raised against the SVM to create SVM-APSO, and it is found to perform better than the conventional SVM [85]. The optimization of SVM based on cuckoo search algorithm (SVM-CSA) has better performance than the conventional SVM [86]. In addition, it is found that the twin SVM (TSVM) that uses two non-parallel hyperplane for classification has better performance than the conventional SVM in dealing with classification problems [87]. We suggest the exploration of SVM-CSA, SVM-APSO and TSVM in the future for developing an authentication scheme for mobile phone touch screen devices instead of the heavily used conventional SVM.

Despite the fact that the machine learning algorithms have proven to improve the security of mobile phone touch screen, yet, there is no consistency on the performance of the algorithms. The performance of the algorithms, e.g., SVM, RF, ANN, BN and KNN is not consistent as evidently revealed from the survey. Evidence of the inconsistent performance of the algorithms is shown in Tables 2, 3, 4, 5 and 6. Though it is a known fact that algorithms lack uniform performance across different domain of applications, it can be concluded that having an algorithm with uniform performance across different domain of applications remain an open research problem. We suggest researchers to attempt to answer the following research question: How can a machine learning algorithm maintain consistent performance across different domains of applications?

The survey revealed that the research projects that propose machine learning algorithm solutions to mobile device security authentication do not consider old technology in their methodology. The methodologies of the projects are limited to the modern technology. However, old technology has the potential to compromise the security of a mobile phone device. As shown in Fig. 5, most of the mobile phone touch screen devices use for the projects used Android as the mobile operating system. However, decade old technology, attention commands can be used to compromise the security of Android mobile phone touch screen devices. The attention commands have been a legacy technology dated back to the 1980s. No matter the sophistication of today's modern smart phone touch screen devices is based on an old technology. As a result of technological advancement, the use of the attention commands has been extended to text messages, 3G, controlling touch screen, launching of camera and LTE. It was found that many of the mainstream smart phone touch screen devices leave the commands accessible to third party through the USB port of the devices when in the hands of consumers. As such, an intruder can decide to set up charging station that is malicious or engage in the distribution of contaminated charging cables. Therefore, use the contaminated charging cables or malicious charging station to initiate an attack with capability to take control of the mobile phone touch screen devices, access data and penetrate the lock screen protection [88]. As a result of that the machine learning-based authentication scheme proposed by several researchers (see Sect. 5) cannot be able to handle the security breaches arising from the attention comments. This is in view of the fact that the attention comments were not considered in the methodology, as such making the authentication scheme vulnerable to the attention comments. We suggest that from now henceforth when developing machine learning approach to improve the security of mobile phone touch screen devices, the legacy technology of the device should be put into consideration to be an integrated part of the authentication scheme. The methodology should not be concentrated on the modern aspect alone.

The machine learning algorithms currently in used in this domain lacks the capability to effectively handle very large-scale data except the ConvNet. In the future, deep learning should be used to process very large-scale data that emanated from mobile phone touch screen devices because deep learning has the capability to process very large-scale data [89, 90].

14 Conclusions

This survey attempts to present a dedicated survey on exploring machine algorithms to improve the security of mobile phone touch screen. This study provides a comprehensive survey on the different machine learning algorithms used in improving the security systems of mobile phone touch screen and how they perform in different scenarios. It reports the recent progress in touch screen authentication, different impersonation scenarios, the limitation of the current approaches and new perspective for future research directions. It was observed that the methods proposed by the previous researchers to improve the security system of touch screen mobile phone opened different research problems. This paper can be used by experienced researchers to easily identify area that require improvement and propose a more effective, robust and efficient security scheme for mobile devices. On the other hand, it can serve as a starting point for new researchers intending to apply machine learning for improving the security of mobile phone touch screen devices.

Acknowledgement This research is supported by TETFund Institutional Based Research Grant through Federal College of Education (Technical), Gombe, Nigeria.

Compliance with ethical standards

Conflict of interest The authors declared that there is no conflict of interest.

Ethical approval This is a literature review article and does not involve human subject for data collection. There is no need for ethical approval.

References

1. Shahzad M, Liu AX, Samuel A (2017) Behavior based human authentication on touch screen devices using gestures and signatures. IEEE Trans Mob Comput 16:2726–2741
2. Ernst M, Swan T, Cheung V, Girouard A (2017) Typhlex: exploring deformable input for blind users controlling a mobile screen reader. IEEE Pervasive Comput 16:28–35
3. Lai J, Zhang D (2015) ExtendedThumb: a target acquisition approach for one-handed interaction with touch-screen mobile phones. IEEE Trans Hum–Mach Syst 45:362–370
4. Yu J, Han H, Zhu H, Chen Y, Yang J, Zhu Y et al (2015) Sensing human-screen interaction for energy-efficient frame rate adaptation on smartphones. IEEE Trans Mob Comput 14:1698–1711
5. Statista (2018) Number of mobile phone users worldwide from 2015 to 2020 (in billions). https://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide/. Accessed 23 Dec 2019
6. Francese R, Risi M, Tortora G, Tucci M (2016) Visual mobile computing for mobile end-users. IEEE Trans Mob Comput 15:1033–1046
7. Arteaga-Falconi JS, Al Osman H, El Saddik A (2016) ECG authentication for mobile devices. IEEE Trans Instrum Meas 65:591–600
8. Clark GD, Lindqvist J (2015) Engineering gesture-based authentication systems. IEEE Pervasive Comput 14:18–25
9. Vu T, Baid A, Gao S, Gruteser M, Howard R, Lindqvist J et al (2014) Capacitive touch communication: a technique to input data through devices' touch screen. IEEE Trans Mob Comput 13:4–19
10. Zhao X, Feng T, Shi W, Kakadiaris IA (2014) Mobile user authentication using statistical touch dynamics images. IEEE Trans Inf Forensics Secur 9:1780–1789
11. Smola AJ, Schölkopf B (2004) A tutorial on support vector regression. Stat Comput 14:199–222
12. Meyer D, Wien FT (2001) Support vector machines. R News 1:23–26
13. Zhou L, Burgoon JK, Twitchell DP, Qin T, Nunamaker JF Jr (2004) A comparison of classification methods for predicting deception in computer-mediated communication. J Manag Inf Syst 20:139–166
14. Bishop CM (1995) Neural networks for pattern recognition. Oxford University Press, Oxford
15. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of IEEE international conference on neural networks, Piscataway December
16. Venter G, Sobieszczanski-Sobieski J (2003) Particle swarm optimization. AIAA J 41:1583–1589
17. Peterson LE (2009) K-nearest neighbor. Scholarpedia 4:1883
18. Gou J, Du L, Zhang Y, Xiong T (2012) A new distance-weighted k-nearest neighbor classifier. J Inf Comput Sci 9(6):1429–1436
19. Svetnik V, Liaw A, Tong C, Culberson JC, Sheridan RP, Feuston BP (2003) Random forest: a classification and regression tool for compound classification and QSAR modeling. J Chem Inf Comput Sci 43:1947–1958
20. Pal M (2005) Random forest classifier for remote sensing classification. Int J Remote Sens 26(1):217–222
21. Keogh EJ, Pazzani MJ (2001) Derivative dynamic time warping. In: Proceedings of the 2001 SIAM international conference on data mining, pp 1–11
22. De Luca A, Hang A, Brudy F, Lindner C, Hussmann H (2012) Touch me once and i know it's you!: implicit authentication based on touch screen patterns. In: Proceedings of the SIGCHI conference on human factors in computing systems, pp 987–996
23. Youssef AM, Abdel-Galil TK, El-Saadany EF, Salama MMA (2004) Disturbance classification utilizing dynamic time warping classifier. IEEE Trans Power Delivery 19(1):272–278
24. Sahami M (1996) Learning limited dependence bayesian classifiers. In: KDD, pp 335–338
25. Friedman N, Geiger D, Goldszmidt M (1997) Bayesian network classifiers. Mach Learn 29:131–163
26. Alpaydin E (2004) Support vector machines. Introd Mach Learn 2004:218–225
27. Li F, Clarke N, Papadaki M, Dowland P (2011) Behaviour profiling for transparent authentication for mobile devices. In: Edith Cowan University Research Online, pp 307–314
28. Campisi P, Maiorana E, Bosco ML, Neri A (2009) User authentication using keystroke dynamics for cellular phones. IET Signal Proc 3(4):333–341
29. Kumar P, Saini R, Roy PP, Dogra DP (2017) A bio-signal based framework to secure mobile devices. J Netw Comput Appl 89:62–71
30. Van Nguyen T, Sae-Bae N, Memon N (2017) DRAW-A-PIN: authentication using finger-drawn PIN on touch devices. Comput Secur 66:115–128
31. Ye G, Tang Z, Fang D, Chen X, Kim KI, Taylor B et al (2017) Cracking android pattern lock in five attempts. In: Proceedings of the 2017 network and distributed system security symposium 2017 (NDSS 17). Internet Society, pp 1–15. https://doi.org/10.14722/ndss.2017.23130
32. Song Y, Cai Z, Zhang Z-L (2017) Multi-touch authentication using hand geometry and behavioral information. In: 2017 IEEE symposium on security and privacy (SP), pp 357–372
33. Su X, Wang B, Zhang X, Wang Y, Choi D (2018) User biometric information-based secure method for smart devices. Concurr Comput Pract Exp 30:e4150
34. Liang Y, Cai Z, Yu J, Han Q, Li Y (2018) Deep learning based inference of private information using embedded sensors in smart devices. IEEE Netw 32:8–14
35. Shen S-S, Kang T-H, Lin S-H, Chien W (2017) Random graphic user password authentication scheme in mobile devices. In: 2017 International conference on applied system innovation (ICASI), pp 1251–1254
36. Mehrnezhad M, Toreini E, Shahandashti SF, Hao F (2018) Stealing PINs via mobile sensors: actual risk versus user perception. Int J Inf Secur 17:291–313
37. Pitropakis N, Panaousis E, Giannetsos T, Anastasiadis E, Loukas G (2019) A taxonomy and survey of attacks against machine learning. Comput Sci Rev 34:100199
38. Banerjee N, Giannetsos T, Panaousis E, Took CC (2018) Unsupervised Learning for Trustworthy IoT. In: 2018 IEEE international conference on fuzzy systems (FUZZ-IEEE), pp 1–8
39. Chen B, Carvalho W, Baracaldo N, Ludwig H, Edwards B, Lee T et al (2018) Detecting backdoor attacks on deep neural networks by activation clustering. arXiv preprint arXiv:1811.03728
40. Barreno M, Nelson B, Joseph AD, Tygar JD (2010) The security of machine learning. Mach Learn 81:121–148
41. Zhou Y, Wang Z, Zhou W, Jiang X (2012) Hey, you, get off of my market: detecting malicious apps in official and alternative android markets. In: NDSS, pp 50–52
42. Abubakar AI, Chiroma H, Muaz SA, Ila LB (2015) A review of the advances in cyber security benchmark datasets for evaluating data-driven based intrusion detection systems. Procedia Comput Sci 62:221–227
43. Shahzad M, Liu AX, Samuel A (2013) Secure unlocking of mobile touch screen devices by simple gestures: you can see it

but you can not do it. In: Proceedings of the 19th annual international conference on mobile computing and networking, pp 39–50
44. Engelstad P, Feng B, van Do T (2016) Strengthening mobile network security using machine learning. In: International conference on mobile web and information systems, pp 173–183
45. Frank M, Biedert R, Ma E, Martinovic I, Song D (2013) Touchalytics: on the applicability of touchscreen input as a behavioral biometric for continuous authentication. IEEE Trans Inf Forensics Secur 8:136–148
46. Bo C, Zhang L, Li X-Y, Huang Q, Wang Y (2013) Silentsense: silent user identification via touch and movement behavioral biometrics. In: Proceedings of the 19th annual international conference on Mobile computing and networking, pp 187–190
47. Saravanan P, Clarke S, Chau DHP, Zha H (2014) Latentgesture: active user authentication through background touch analysis. In: Proceedings of the second international symposium of Chinese CHI, pp 110–113
48. Ehatisham-ul-Haq M, Azam MA, Naeem U, Amin Y, Loo J (2018) Continuous authentication of smartphone users based on activity pattern recognition using passive mobile sensing. J Netw Comput Appl 109:24–35
49. Burgbacher U, Hinrichs K(2014) An implicit author verification system for text messages based on gesture typing biometrics. In: Proceedings of the SIGCHI conference on human factors in computing systems, pp 2951–2954
50. Chen Y, Sun J, Zhang R, Zhang Y (2015) Your song your way: rhythm-based two-factor authentication for multi-touch mobile devices. In: 2015 IEEE conference on computer communications (INFOCOM), pp 2686–2694
51. Xu H, Zhou Y, Lyu MR (2014) Towards continuous and passive authentication via touch biometrics: An experimental study on smartphones. In: Symposium on usable privacy and security, SOUPS, pp 187–198
52. Zhang D, Kang Y, Zhou L, Lai J (2016) Continuous user authentication on touch-screen mobile phones: toward more secure and usable M-commerce. In: Workshop on E-business, pp 225–236
53. Kambourakis G, Damopoulos D, Papamartzivanos D, Pavlidakis E (2016) Introducing touchstroke: keystroke-based authentication system for smartphones. Secur Commun Netw 9:542–554
54. Lin C-C, Chang C-C, Liang D, Yang C-H (2012) A new non-intrusive authentication method based on the orientation sensor for smartphone users. In: 2012 IEEE sixth international conference on software security and reliability (SERE), pp 245–252
55. Buczak AL, Guven E (2016) A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Commun Surv Tutor 18:1153–1176
56. Buschek D, De Luca A, Alt F (2015) Improving accuracy, applicability and usability of keystroke biometrics on mobile touchscreen devices. In: Proceedings of the 33rd annual ACM conference on human factors in computing systems, pp 1393–1402
57. Feng T, Liu Z, Kwon K-A, Shi W, Carbunar B, Jiang Y et al (2012) Continuous mobile authentication using touchscreen gestures. In: 2012 IEEE conference on technologies for homeland security (HST), pp 451–456
58. Feng T, Zhao X, Carbunar B, Shi W (2013) Continuous mobile authentication using virtual key typing biometrics. In: 2013 12th IEEE international conference on trust, security and privacy in computing and communications (TrustCom), pp 1547–1552
59. Khalidd A (2017) Identifying smartphone users based on their activity patterns via mobile sensing. Procedia Comput Sci 113(2017):202–209. https://doi.org/10.1016/j.procs.2017.08.349
60. Kolly SM, Wattenhofer R, Welten S (2012) A personal touch: recognizing users based on touch screen behavior. In: Proceedings of the third international workshop on sensing applications on mobile phones, p 1
61. Zhang Y, Yang M, Ling Z, Liu Y, Wu W (2018) FingerAuth: 3D magnetic finger motion pattern based implicit authentication for mobile devices. Future Gener Comput Syst. https://doi.org/10.1016/j.future.2018.02.006
62. Antal M, Szabó LZ (2016) Biometric authentication based on touchscreen swipe patterns. Procedia Technol 22:862–869
63. Alpar O (2015) Intelligent biometric pattern password authentication systems for touchscreens. Expert Syst Appl 42:6286–6294
64. Zhou L, Kang Y, Zhang D, Lai J (2016) Harmonized authentication based on ThumbStroke dynamics on touch screen mobile phones. Decis Support Syst 92:14–24
65. Alpar O, Krejcar O (2015) Pattern password authentication based on touching location. In: International conference on intelligent data engineering and automated learning, pp 395–403
66. Beton M, Marie V, Rosenberger C (2013) Biometric secret path for mobile user authentication: a preliminary study. In: 2013 World congress on computer and information technology (WCCIT), pp 1–6
67. Nader J, Alsadoon A, Prasad P, Singh A, Elchouemi A (2015) Designing touch-based hybrid authentication method for smartphones. Procedia Comput Sci 70:198–204
68. Meng W, Wang Y, Wong DS, Wen S, Xiang Y (2018) TouchWB: touch behavioral user authentication based on web browsing on smartphones. J Netw Comput Appl 117:1–9
69. Meng Y, Wong DS, Schlegel R (2012) Touch gestures based biometric authentication scheme for touchscreen mobile phones. In: International conference on information security and cryptology, pp 331–350
70. Meng Y, Wong DS (2014) Design of touch dynamics based user authentication with an adaptive mechanism on mobile phones. In: Proceedings of the 29th annual ACM symposium on applied computing, pp 1680–1687
71. Samangouei P, Patel VM, Chellappa R (2017) Facial attributes for active authentication on mobile devices. Image Vis Comput 58:181–192
72. Feng T, Yang J, Yan Z, Tapia EM, Shi W (2014) Tips: context-aware implicit user identification using touch screen in uncontrolled environments. In: Proceedings of the 15th workshop on mobile computing systems and applications, p 9
73. Watanabe Y (2014) Influence of holding smart phone for acceleration-based gait authentication. In: 2014 Fifth international conference on emerging security technologies (EST), pp 30–33
74. Crawford H, Renaud K, Storer T (2013) A framework for continuous, transparent mobile device authentication. Comput Secur 39:127–136
75. Mahbub U, Sarkar S, Patel VM, Chellappa R (2016) Active user authentication for smartphones: a challenge data set and benchmark results. In: 2016 IEEE 8th international conference on biometrics theory, applications and systems (BTAS), pp 1–8
76. Roy A, Halevi T, Memon N (2014) An HMM-based behavior modeling approach for continuous mobile authentication. In: 2014 IEEE international conference on acoustics, speech and signal processing (ICASSP), pp 3789–3793
77. Fathy ME, Patel VM, Chellappa R (2015) Face-based active authentication on mobile devices. In: 2015 IEEE international conference on acoustics, speech and signal processing (ICASSP), pp 1687–1691
78. Jain A, Kanhangad V (2015) Exploring orientation and accelerometer sensor data for personal authentication in smartphones using touchscreen gestures. Pattern Recogn Lett 68:351–360

79. Serwadda A, Phoha VV, Wang Z (2013) Which verifiers work? A benchmark evaluation of touch-based authentication algorithms. In: 2013 IEEE sixth international conference on biometrics: theory, applications and systems (BTAS), pp 1–8
80. Sitová Z, Šeděnka J, Yang Q, Peng G, Zhou G, Gasti P et al (2016) HMOG: New behavioral biometric features for continuous authentication of smartphone users. IEEE Trans Inf Forensics Secur 11:877–892
81. Renear AH, Sacchi S, Wickett KM (2010) Definitions of dataset in the scientific and technical literature. In: Proceedings of the 73rd ASIS&T annual meeting on navigating streams in an information ecosystem, vol 47, p 81
82. Snijders C, Matzat U, Reips U-D (2012) " Big Data": big gaps of knowledge in the field of internet science. Int J Internet Sci 7:1–5
83. He F, Bao L, Wang R, Li J, Xu D, Zhao X (2017) A multimodal deep architecture for large-scale protein ubiquitylation site prediction. In 2017 IEEE international conference on bioinformatics and biomedicine (BIBM), pp 108–113
84. Ibrahim R, Yousri NA, Ismail MA, El-Makky NM (2014) Multi-level gene/MiRNA feature selection using deep belief nets and active learning. In: 2014 36th annual international conference of the IEEE engineering in medicine and biology society (EMBC), pp 3957–3960
85. Yang X-S, Deb S, Fong S (2011) Accelerated particle swarm optimization and support vector machine for business optimization and applications. In: International conference on networked digital technologies, pp 53–66
86. Zhang X, Wang J, Zhang K (2017) Short-term electric load forecasting based on singular spectrum analysis and support vector machine optimized by Cuckoo search algorithm. Electr Power Syst Res 146:270–285
87. Tomar D, Agarwal S (2015) Twin support vector machine: a review from 2007 to 2014. Egypt Inform J 16:55–69
88. Newman LH (2018) Exploiting decades-old telephone tech to break into android devices. https://www.wired.com. Accessed 20 Dec 2019
89. Zhang Q, Yang LT, Chen Z, Li P (2018) A survey on deep learning for big data. Inf Fusion 42:146–157
90. Wang H, Raj B (2015) A survey: time travel in deep learning space: an introduction to deep learning models and how deep learning models evolved from the initial ideas. arXiv preprint arXiv:1510.04781