International Journal of Safety and Security Engineering
Vol. 13, No. 5, October, 2023, pp. 903-915
Journal homepage: http://iieta.org/journals/ijsse

An Adaptive Context-Aware Authentication System on Smartphones Using Machine Learning

Aiman M. Ayyal Awwad

Department of Information Technology, Faculty of Information and Communications Technology, Tafila Technical University, Tafila 66110, Jordan

Corresponding Author Email: AimanAwwad@ttu.edu.jo

https://doi.org/10.18280/ijsse.130514

Received: 15 March 2023
Revised: 16 October 2023
Accepted: 24 October 2023
Available online: 10 November 2023

Keywords:
authentication, mobile computing, context-awareness, machine learning

ABSTRACT

The authentication method for unlocking screens is an essential security feature of smartphones to avoid unauthorized access. Various mechanisms have been integrated into smartphones to authenticate users to reduce privacy infringement. Contextual awareness is now a key requirement for mobile computing to make intelligent decisions and provide an adaptable and convenient authentication model. Therefore, in this paper, a context-aware authentication system based on a machine learning technique is proposed. The proposed system takes the user's body postures, location, SMS contents, and ambient environmental conditions as context. To enhance privacy protection, these context factors are selected to guide the authentication process and enable the authentication system to understand users and their surrounding environment. However, to provide the most appropriate authentication method, a machine learning model is designed. The model is trained and tested with the user's context datasets. An appropriate dataset for the machine learning model is generated, and features that affect end-user interaction during the authentication process are identified. A total of 25 responses from smartphone owners were collected to evaluate the proposed system. After conducting a sentiment analysis, we found that 72 percent of users have a positive sentiment regarding the proposed system, which means that context-aware technology helps improve authentication adaptability and provides a convenient authentication method. The performance of the model was tested, and the results show that the proposed model effectively achieves a Mean Absolute Error (MAE) of 1.299, a Root Mean Square Error (RMSE) of 1.437, and an R Square of 76.78. Therefore, the system can improve the reliability and adaptability of the authentication process.

1. INTRODUCTION

With the rapid development of mobile computing and information technology, smartphones have become an integral part of individuals' daily lives. The use and popularity of smartphones are rapidly increasing, and more customers are trusting these devices, as a direct way to access the Internet [1, 2]. According to data collected by the Statista portal in October 2022, the number of mobile devices operating around the world is about 15.96 billion, and it is anticipated to reach almost 18.22 billion by 2025 [3]. Moreover, the number of smartphone subscribers worldwide surpassed six billion in 2022 and is expected to increase by one hundred million in the next few years [4].

Smartphones have convenient features that allow users to navigate and interact with applications. The diversity of applications has increased the connection between users and their smartphones, and a large volume of private data is kept on smartphones [5]. This increasing proliferation of smartphones makes controlling access to devices and data security crucial. At the same time, the continuous and rapid growth of mobile applications and cloud services is driving the demand for authentication. Therefore, to decrease the risk of privacy disclosure, various authentication models (such as PIN code, fingerprint, pattern, etc.) have been commonly integrated into smartphones [2, 6].

Meanwhile, it is doable for unauthorized users to gain access to the system. Biometrics technology (such as a fingerprint, iris, or face ID) is widely used due to its simplicity and convenience, and at the same time, it is vulnerable to various attacks [6]. For instance, a fingerprint pattern is vulnerable to impersonation attacks from artificial fingers. Specifically, fingerprints can be robbed using a special type of sticker, and there is a lack of authenticating the user's identity in case of damage or wound or when fingerprints are covered with dirt or oil. The password-based authentication method has a major problem in that users are not really specialized in memorizing text code. Additionally, sometimes passwords can be easy to guess or steal, particularly if they are not strong or predictable [1, 2]. Furthermore, facial recognition methods can be fooled with a 3D model of the owner's head. The user is accustomed to using one of the authentication methods, but to enhance the security of the protection mechanisms, we need to consider unlocking techniques that might be more convenient, adaptive, and secure [6].

For this very reason, it makes sense to have a dynamic and smart mechanism in place to provide an authentication model based on some criteria and user requirements. Typically, context-aware authentication is a useful approach that utilizes contextual information of the user regarding authentication [5].

A context-aware service works when both physical and virtual sensors handle contextual information on a smartphone (e.g., temperature and humidity) and a user (e.g., physical location, body postures, or user preferences) [7]. Context-aware technology enables services that consider the demands and needs of different users [8, 9]. These demands and interests can be presented in terms of their relevance to users and in terms of giving their main areas of interest. Context-aware applications infer what, when, and where the feature should be provided to the end-user in a smart way [8-10]. Then, based on a given set of parameters for the user, relevant contextual information is presented to the appropriate user at the appropriate time through a user interface [11, 12].

Context-aware authentication models are becoming more impressive as mobile devices are using more widespread computing environments. Context-aware authentication is usually a reasonable approach that uses the results of user conventions for authentication purposes. Unlike traditional authentication methods, the context-aware authentication method is secure against attacks since there is a lot of information representing the user's context and it is difficult to speculate the context of the user's device [1, 2, 5]. However, to make a useful prediction from context data, a machine learning model is essential.

Machine learning focuses on using data and algorithms to imitate the way people learn. To estimate the relationship between two or more contextual factors and a single authentication model, a Multiple Linear Regression (MLR) model is used. MLR is a statistical technique that uses multiple independent descriptive variables to anticipate the outcomes (i.e., dependent variable) by fitting the linear relationship to the observed data [13, 14].

Therefore, in this paper, a machine learning authentication model based on context awareness is proposed. It can be applied in various ways and ensure a secure authentication process by analyzing the user's context from his device. The context data of the owner is used to train the authentication model to enhance the security of the protection mechanism and correctly distinguish the owner from all fraudsters.

The aim of this research is to provide adaptability in mobile authentication system based on user's preferences and their context. Therefore, the research question that this paper aims to answer is: Can we enhance privacy protection and provide the most appropriate authentication method when the authentication process understands the user and is guided by contextual factors?

The proposed context-aware authentication system detects information and adapts it accordingly to deliver a more relevant and interactive user experience. It is an intelligent approach that provides useful services based on geolocation, request, desire, and intent. In the proposed system, a multiple linear regression model is designed to predict the authentication method based on features. The dataset is constructed to contain features that describe the user context. Features include body postures, location, calendar, alarm, weather, and SMS. The proposed system utilizes a user's context, however, depending on the device's ambient environment, the appropriate authentication model is provided to the end user. The contribution of our work is as follows:

- Both physical and virtual sensors on a smartphone are used to gather contextual information about users and information about their surrounding environment.
- General factors are gathered to determine the user's body postures or activity, geographic location, and ambient noise level. The Calendar provider API, Weather conditions API, AlarmManager class, and SMS content provider are used to collect user's context and preferences.
- A machine learning algorithm (i.e., MLR) is used to identify the authentication model to unlock a smartphone screen and provide an appropriate and adaptable authentication model. To the best of our knowledge, our work is the first that primarily uses the above-mentioned contextual factors to guide the authentication process in order to enhance the reliability and adaptability of the authentication process.
- Since emojis have become widespread, many researchers have developed ways to integrate them into existing sentiment analysis methodologies [15]. Therefore, in this study, user sentiment (that is, positive or negative) is analyzed using expressive sentiment responses from users immediately after the authentication process is performed.

2. AUTHENTICATION METHODS TO UNLOCK SMARTPHONE'S SCREEN

Recently, the smartphone has been equipped with various sensors to enable users to unlock the smartphone screen. Authentication to unlock the screen is the mode that occurs when the user turns on the smartphone or wakes it from sleep mode. To remove the lock, the user must use one of the supported authentication methods. Authentication refers to verifying a user's credentials, and authorization occurs after the system successfully authenticates the user's identity [5, 16].

2.1 Symbolic password

This method of unlocking an Android-powered device requires the user to enter a combination of letters, numbers, or characters. The longer the password, the higher the security level of the smartphone. Users typically prefer passwords that are highly guessable, write them down, never changed, or only be changed among a few alternatives, making them vulnerable to attack. The password should be unique and memorable, and at the same time, it is easy for an unknown person to spy on the password the moment the user unlocks the screen [1, 2]. In this method, the user can always type the letters, numbers, and characters to create a complex password to unlock the screen. While this can be more secure than a PIN code or a pattern. Also, it would be very annoying to do it over and over again, especially, when the user's hands are busy with something.

2.2 PIN code

As with the symbolic password, the PIN code enables the user to quickly unlock the smartphone. To unlock an Android smartphone or tablet, the user needs to enter a four-digit code. A PIN code is a reasonable authentication method that balances convenience and security. Users can type a longer PIN code faster than with a drawing pattern, and much faster if a simpler keypad is used. PINs might be guessed, shared, or robbed, especially if they are not kept confidential. For the anonymous, it is easy to spy to get the code. One of the main drawbacks of PIN codes is the limited level of security, particularly when compared to more recently developed authentication methods such as biometrics (fingerprint or face recognition). For a PIN code, users tend to pick the code that is easy to guess. Most PIN codes are relatively short, usually four to six digits long,

making them vulnerable to brute force attack [1, 5].

2.3 Graphic or pattern lock

Pattern lock has become a popular method to authenticate a smartphone's user. In this case, the user needs to draw a simple shape with his/her finger. It is a convenient and secure method to unlock the smartphone or tablet. The pattern is usually a 3X3 matrix of dots. The pattern method seems to be more secure than the conventional PIN code and less secure compared to biometric methods. However, swiping a complex pattern takes longer than a PIN code, therefore, the more secure your smartphone is, the more difficult it is to unlock as well [1, 2]. Patterns are easy to guess or notice, especially if the attacker watches the user while drawing the pattern.

2.4 Fingerprint

Fingerprints are intricate, almost unique, difficult to change, and persist throughout an individual's life, making them appropriate as a long-term feature of an individual's identity. Fingerprint verification is a process for identity authentication used to verify an individual's identity by comparing a captured sample with a previously stored sample. When the samples match, the verification process is passed. In this case, the user's fingerprint pattern is only recorded once, which makes it easy to verify the fake fingerprint successfully. Fingerprint can be used to open the lock easily and quickly. It is a reliable security method that is hard to tamper with. The user needs to touch the sensor with his/her finger. It also does not require the genuine user to remember a password. Unlocking has become much easier with the advent of fingerprint sensors in smartphones [2, 17].

2.5 Face recognition

This method enables genuine users to unlock their smartphones and verify their identity simply by placing their faces directly in front of their smartphone screen. Face images are used to build the user's face model, and the face model is stored on the smartphone. Facial recognition is a secure and convenient method to unlock a smartphone device. This method uses facial recognition technology to identify the owner of the smartphone. It takes and analyzes the owner's unique facial features to build a template of the face. During the setup process, the smartphone owner will be asked to record his/her face by taking a sequence of images. Then, the smartphone builds a model of the owner's face that can be used in the comparison process. Although the facial recognition method is convenient, it is not secure as compared to other methods such as fingerprint or a PIN code. For instance, some systems can be tricked by models or videos of the user's facial features [6, 18].

2.6 Voice unlock

This method allows genuine users to save voice samples on the smartphone and when the user speaks the words loudly, the smartphone will check the sound to unlock the screen [18]. Voice unlocks systems recognize and verify the smartphone owner's voice by analyzing various voiced traits, such as tone, pitch, and speech prints. Although voice unlock is convenient, it is not secure as compared to other biometric methods such as fingerprint or face ID. An attacker can impersonate the owner or record the owner's voice. Voice unlocks is often used for hands-free access to smartphones. Users can just say a passphrase to quickly unlock their device. It is a very appropriate approach, especially when the user's hands are busy eating or driving. Regarding privacy concerns, the genuine user should be aware of potential concerns as voice samples are usually kept and processed on the device [18].

Certainly, smartphone unlocking methods differ in terms of security, convenience, and accessibility. A comparison of common smartphone screen unlocking methods highlighting the strengths and weaknesses of each method is illustrated in Table 1.

Table 1. A comparative summary of smartphone unlocking methods

	Security	Convenience	Accessibility
Passcode /PIN	Relatively secure	Require entering a four-digit code or alphanumeric password	Support individuals with disabilities
Pattern lock	Moderate security	Balances security and convenience	May not support individuals with certain motor skills disabilities
Fingerprint	High security	Convenient and fast	May not support individuals with certain disabilities
Facial recognition	High security	Very convenient and fast	May not operate well in low-light situations
Voice unlock	Moderate security	Convenient, may not operate well in noisy places	Support individuals with certain disabilities, such as poor eyesight

3. RELATED WORKS

Authentication methods, particularly smartphone unlocking, play a vital role in how users interact with PCs, smartphones, tablets, etc. Although various technologies have been adopted to unlock screens such as pattern, fingerprint, and face ID, they remain subject to some limitations. In order to better understand how authentication is used in smartphones and how to deliver a context-aware authentication system, various studies on authentication methods have been conducted [17, 19].

For pattern locking, a fine-grained and context-aware biometrics system was proposed by Shi et al. [2]. The combination of implicit and traditional authentication systems to build a two-factor authentication method is useful for enhancing the security of the protection mechanism. During pattern unlocking, the proposed scheme uses different behavior information of the user as additional fingerprint authentication, so even if the password information is exposed to the attacker, the system is still safe and secure. The authors proposed a context-aware module to differentiate between any of these contexts (for example, body postures when entering the pattern) and use them to direct the authentication process. Also, the authors use a single-class classifier to train the authentication model. The classifier does not require fraudulent data during the training process for the authentication classifier. Finally, the effectiveness of the proposed system was verified, and the authors pre-assigned a combination of passwords based on user preferences. The results show that the proposed system

can effectively enhance the ability of the pattern lock to withstand attacks [2].

Wang and Tao proposed a scheme for context-awareness implicit authentication. Their assumption is to improve the PIN authentication method. This approach uses three motion sensors (accelerometer, magnetometer, gyroscope) and some associated touch time and pressure context (timestamp, touch size, pressure) to capture features of the user. The proposed scheme is based on touch and gesture features obtained using both distance and statistical measurement methods. The authors infer that these sensors can not only recognize movement habits to further differentiate users but also capture various kinds of user postures. The scheme also adopts a multi-expert fusion process to merge the authentication results into the score level. The effectiveness of the proposed contextual awareness scheme is verified by testing the authentication method using different passwords. The results show that the proposed method can effectively achieve the best equal error rate (EER). Therefore, it can improve the reliability and implicit authentication process [1].

A context-aware authentication system was proposed by Benzekki et al. [5]. The system is easy to implement and cost-effective. Ambient environmental information is sent when the users authenticate themselves making the system more efficient and lightweight. The authentication process goes through various stages in order to make a decision concerning access. The proposed system is capable to work as a second authentication to enhance the authentication process using a symbolic password or act as a primary authentication model to entirely replace passwords in a cost-effective and easy-to-use manner and offer further security levels for financial transactions such as online payments using credit cards or PayPal.

Nowadays, the combination of gestures with authentication systems has proven to be more successful, therefore, Ganesh et al. proposed a gesture-based authentication scheme to securely unlock smartphones with touch screens. This authentication scheme is based on the fuzzy inference system that produces decisions based on the user's finger pressure and inclination of the smartphone while it is unlocked. The scheme is trained using appropriate datasets that contain the required features from the smartphone owner. During an actual authentication process, the fuzzy system produces smart decisions by using the input variables with a threshold value to unlock the smartphone securely. This work differs from previous works in two aspects. First, the proposed pattern is safe against both smearing and shoulder surfing attacks. Second, the time complexity of the authentication system has decreased [20].

Feng et al. introduced a touch-based identity protection service that authenticates users implicitly and unobtrusively in the background by constantly studying touchscreen gestures in a running application. The proposed scheme analyzed real-world touch data as well as contextual information needed for persistent user authentication. The authors evaluated their system based on collected data from 23 phone owners. The system achieved more than 90 percent accuracy in real-world conditions under minimal computational load and 6 percent of battery usage. The proposed system is effective not only in the offline simulation but also in the practical testing of the device [7].

The main purpose of the proposed system is to sense and collect the user's context and provide an adaptive authentication model. To the best of our knowledge, our work is the first to use a context-aware technology in the authentication process primarily. To enhance privacy protection, these context factors are selected to guide the authentication process and enable the authentication system to understand users and their surrounding environment. The proposed system provides a balance between security and convenience and offers an authentication method that is more suitable for the current situation. The system uses a machine learning algorithm to predict the authentication method, which has not been researched in the above works.

4. SYSTEM DESIGN

In this section, a framework architecture for an adaptive context-aware mobile authentication system is presented. The proposed framework consists of a set of layers that can sense context changes and adapt the system's authentication model according to the context as well as the user profile and preferences.

Figure 1 provides an overview of the context-aware authentication system, which consists of two stages: training and authentication. In the training stage, the proposed system calls the motions, location, and other sensors built into the smartphone to gather the owner's data when he wants to unlock the smartphone screen. In addition, from the collected data we will extract contextual features and define user context according to the contextual awareness classifier. The detected context features are fed into the machine learning model to provide an authentication method.

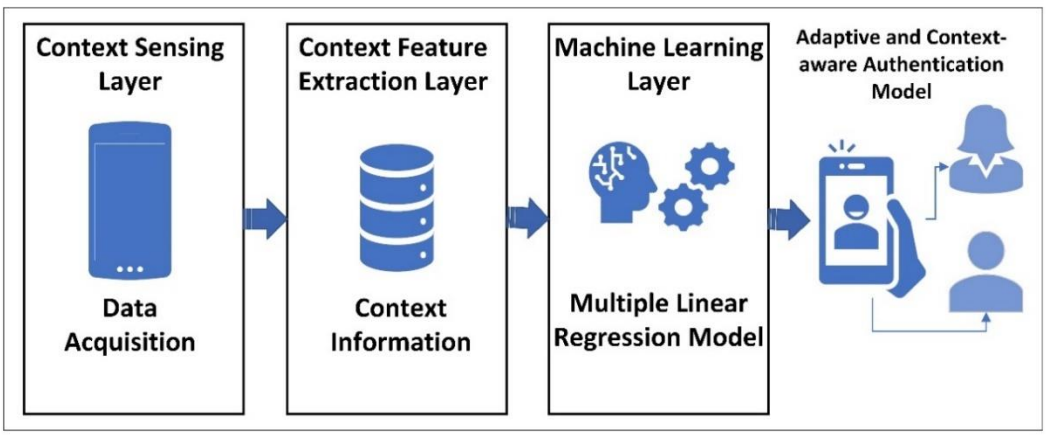


Figure 1. Architecture of the proposed authentication system framework

In the authentication stage, the user just needs to unlock the smartphone screen as usual to finish the authentication process. During system execution, the obtained features and captured context information are provided in the trained proposed model. In the next part, we will describe the main components of the proposed approach, namely data acquisition, context awareness, machine learning model, and authentication model construction.

4.1 Context sensing layer

The first step is to obtain the user's input data. After the transformation of the context, the second step generates the customized outcome according to the information that reflects the authentication model. Contextual information can be categorized into seven elemental kinds: geographical location, alarm settings, calendar events, SMS messages, noise levels, weather conditions, and user's physical activity as shown in Figure 2.

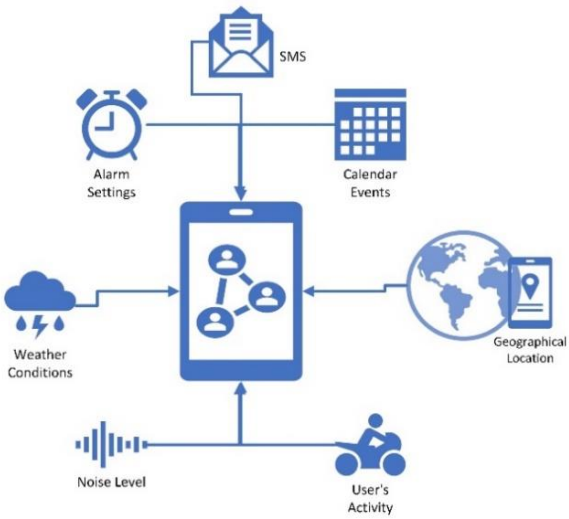


Figure 2. Types of contextual information

In our mobile authentication system, contextual information originates from manifold physical and virtual sensors. The component performs a user-centered operation using smartphones. Smartphones can use context information to render a personalized authentication model. Virtual and physical sensors can be used to obtain the user's context information as follows [21]:

4.1.1 Virtual sensors

Virtual sensors can obtain virtual information from mobile apps. For instance, applications such as Calendars, Alarms, Weather, and SMS can be deemed as virtual sensors. Specifically, we can use the Alarm App to get a wake-up alarm and sleeping schedule for the user. In our model, the user's virtual sensor contexts are composed of four factors: calendar events, weather conditions, alarm settings, and SMS messages. Reading from virtual sensors can involve different potential challenges, including:

- Calendars: Scheduling conflicts, time zone differences, event updates, privacy concerns.
- Weather conditions: Inaccuracy, changing conditions, regional variations.
- Alarm settings: Alarm types, privacy and security, setting customization.
- SMS messages: User permissions, message formats, message encoding schemes, length limitations, privacy concerns.

4.1.2 Physical sensors

The Android smartphone is equipped with sensors that detect movement, direction, and various environmental states. Android sensors relay data from various physical sensors such as accelerometers, ambient temperature, magnetometers, gyroscopes, heart rate, light, proximity, pressure, and relative humidity sensors. Reading data from physical sensors can come with several potential challenges. Here are some common problems associated with these sensors:

- Location sensor: Inaccuracy, signal interference, indoor use, battery drain, privacy concerns, real-time updates, etc.
- Accelerometers sensor: Noise and signal filtering, vibration and shock handling, power consumption, calibration, etc.
- Gyroscopes sensor: Noise and drift, temperature sensitivity, vibration and shock, power consumption, latency, calibration, etc.
- Noise level sensor: Accuracy and calibration, environmental interference, privacy concerns, sensor location, stable power supply, etc.

The proposed approach utilizes three broad categories of sensors as follows [22]:

4.1.3 Motion sensors

These sensors (including accelerometers, gyroscopes, gravity sensors, and rotation vector sensors) are used to monitor the user's physical activity such as immobile, walking, running, etc. The accelerometer is used to measure the appropriate acceleration or precise movement of the smartphone and detect the shake and inclination of the device. A gyroscope is a sensor that can measure and maintain the rotational rate and angular velocity of a smartphone. The rotation vector sensor can be used to detect gestures and measure relative changes in the smartphone's orientation. Thus, after the user's context information is collected by the smartphone, the proposed approach will segment the context information according to the source sensor type.

4.1.4 Environmental sensors

The smartphone provides sensors to measure different environmental attributes. These sensors are used to monitor the ambient temperature, relative humidity, illuminance, noise level, and pressure, in which the user is operating and located. All environmental sensors are hardware-dependent and are only available if the smartphone manufacturer has included them in a device.

4.1.5 Position sensors

These sensors are used for monitoring the physical position of a smartphone device (e.g., home, office, library, etc.). Android-powered devices use two sensors to determine the location of a smart device: an accelerometer and a geomagnetic field sensor.

4.2 Context feature extraction layer

In this layer, context feature extraction is responsible for tuning and analyzing contextual information collected by virtual and physical sensors. The raw sensor data gathered by these sensors are processed in this layer, and the output

information is retained for use in the machine learning layer. The goal is to determine the appropriate context for our model requirements and the degree of context relevance that matches the user's context.

To perform better user authentication, we use the extraction method to extract features from the obtained user data. In this system, context features are heterogeneous and from various sources (i.e., virtual and physical sensors). We need to define the dataset in machine learning, how to gather the data, and what features the appropriate dataset has. That is why, after data acquisition, it is essential to pre-process it (data cleaning), as well as annotate the data by labeling it so that can be understood by the system. In addition, we suppose that the user's context when using the smartphone will be helpful for authentication model construction.

Similarly, in our approach, motion sensors (accelerometer, gyroscope, and rotation vector sensor) are used to get the motion attributes of the smartphone while unlocking the screen. Moreover, by analyzing the user's context information while unlocking the screen, it can be seen that the uniqueness of various users' context is not only deemed in the motion sensing information of the smartphone but also in the location information, ambient environment state (noise level), and software context (i.e., weather condition, calendar events, and alarm settings).

4.3 Machine learning layer

Machine learning is a key component of the proposed approach. Using statistical techniques, algorithms are trained to perform decision processes (i.e., prediction or classification). A machine learning model is a mathematical function that takes features as inputs, makes predictions as outputs (the predicted value is the label), and learns how best to correspond predicted values with patterns reported from the training data. In machine learning, the learning process is just a mapping function that maps features to targets and this function is used for data whose labels are not known. In the proposed approach, the authentication model is one of the most critical parts for selecting the adaptive and most related authentication method [13, 23].

4.3.1 Multiple linear regression

Multiple linear regression is a machine learning technique used to model the relationship between two or more independent features (i.e., inputs or predictors) and responses to one dependent variable (i.e., the output). MLR is a statistical model that produces coefficients that can be used to predict inputs by fitting a linear equation to the observed data related to each user. MLR is used when the system needs to predict the outcome of a variable based on the value of other independent variables. It is an extension of linear regression models that takes only one independent variable. In MLR, the target is to understand how the dependent variable is affected by two or more independent variables. It can be defined using the following formula [14]:

$$y = \mathrm{R}_0 + \mathrm{R}_1 X_1 + \cdots + \mathrm{R}_n X_n + \partial \qquad (1)$$

where:

y=The dependent, i.e., predicted response.

R_0=The y-intercept, i.e., the value of y when all other factors are zero.

$\mathrm{R}_1 X_1$=The coefficient of regression (R_1) of the independent input variable (X_1) (It is also known as the impact of increasing the value of the independent variable on the predicted value of y).

$\mathrm{R}_n X_n$=The coefficient of regression for the last independent input.

∂=A model error (that is, how much variance is there in the estimation of y).

Regression is used to find relationships between independent variables as well as map features to others. The proposed model attempts to find out how users' authentication methods depend on their features. We use multiple regression to predict the authentication model based on the user's physical activity, geographical location, weather conditions, and calendar information. With multiple regression, we can use more variables, such as alarm settings, and SMS messages to make the prediction more accurate [14].

4.3.2 Dataset

The first thing we need to do is decide which sources we will use to collect data. Typically, there are three types of sources you can select from: open-source datasets available online, the Internet, and artificial data generators. Dataset collection sources vary and depend heavily on the targeted project. The best option for our model is to collect data from smartphone sensors that are directly related to the system objectives [24].

Datasets are especially essential for machine learning algorithms to learn from. The dataset is an instance of how a machine learning algorithm can make predictions, with tags representing the result of a given prediction (pass or fail). A dataset is just a collection of data that can be utilized to make predictions about future events or outcomes based on historical data. Datasets are usually labeled before being used by machine learning algorithms, so the algorithm understands which results to predict or classify [14, 24].

Raw data is a useful source for starting and feeding a machine learning algorithm to deliver users with an appropriate authentication model according to the user's context. Machine learning datasets take various forms and can be obtained from several sources. Typically, the three kinds of machine learning datasets are text data, multimedia data, and sensor data.

Furthermore, it is necessary to make sure that the data fragments are of sufficient quality. While there are manners to clean up the data and make it standardized and understandable before the two processes: annotation and training, it is best that the data matches the list of desired features. There are a few steps we need to perform before the dataset can be usable [25] (see Figure 3).

- Collect: The first thing we need to do is identify data sources. In our case, the raw sensor data is the source of the dataset for the proposed approach.
- Pre-process: We need to define which features make up a suitable dataset. High quality is the key thing to consider when collecting a dataset for a machine learning model. First, the pieces of data must be relevant to our objective. Second, the data should be cleaned by fixing bad data in the dataset, including, empty cells, wrong formatting, duplicates, etc.
- Annotate: After we make sure that the data is clean and relevant, we also need to ensure that the data is understandable so that the computer can process it.

Thus, to make smart decisions and deliver an adaptive, secure, and context-aware authentication model, context

awareness is now a crucial requirement for mobile computing and services. To build our model using machine learning, we must create a new dataset. Because it is a very specific problem, we must create a domain dataset, clean it, and understand its relevance to acquire the outcome. Next, it is time to train the proposed model. The basic principle is to get the annotated data and feed it into the algorithm. In fact, we will need to repeat this process several times. This step is a training of the machine learning model. Training means feeding our proposed model with bunches of cleaned and annotated data for a distinct number of epochs (i.e., the number of times the algorithm is trained on the entire project).

Figure 3. Three steps to processing data in machine learning

4.3.3 Machine learning features

A suitable dataset is required for the machine learning model; therefore, we must determine the features of the collected data. For this reason, physical and virtual sensors are chosen to collect the following contextual factors.

4.3.4 User's body postures and physical activity

Smartphones are not always immobile, and users may use their smartphones while they are in various body postures and physical activities, such as running, walking, sitting, or driving. The context of physical activity refers to the user's actual postures. For example, an increase in smartphone mobility is less favorable toward a PIN code or password authentication model, and thus, face recognition is more favorable. According to the user's mobility, we classify the user's posture and activity into four levels: sitting, walking, running, and driving. Each level is defined by a numerical value as shown in Table 2.

Table 2. User's posture and activity levels

User's Postures	Value
Sitting	1
Walking	2
Running	3
Driving	4

4.3.5 Noise levels

The smartphone is used to compute noise levels in the surrounding environment such as crowded spots in cities or workplaces, and quiet places in the countryside or at home. When the ambient environment noise is loud, voice-based authentication is not usable. Noise levels affect the user's concentration regarding achieving authentication. The factor will be used as one of the machine learning features. Three levels of noise are defined as shown in Table 3.

Table 3. The values of noise levels

Noise Levels	Value
High	1
Medium	2
Low	3

4.3.6 Geographical location

In the proposed authentication model, the geolocation context, defined by the location sensor, is an influential feature in providing acceptable authentication as we assume. This approach uses location-tracking technologies to determine the owner's physical location. For example, in a library, the fingerprint model is more favorable to be delivered, or a user can unlock her smartphone once she enters a certain location. We classified an individual's physical location into four categories. They are expressed in integer values as shown in Table 4.

Table 4. Physical location parameter values

Physical Location	Value
Home	1
Office	2
Park	3
Shopping	4
Restaurant	5

4.3.7 Alarm App settings

Users usually set a wake-up alarm and sleep schedule (including bedtimes, wake-up times, and mealtimes) in the Alarm App. If the user does not want to set up a sleep schedule, he/she can set a regular alarm for the wake-up time. For instance, when a user is having a meal, a voice-based authentication method is more convenient. We categorize an individual's sleep schedule into three categories as shown in Table 5.

Table 5. Alarm settings values

Alarm App Settings	Value
Bedtimes	5
Wake-up times	10
Meal timing	15

4.3.8 Calendar App events

The Calendar App is a storehouse of user events. The Calendar App automatically synchronizes any events and dates linked with the user's Google Account. On Android devices, the Calendar application displays events and is most typically used to set reminders for events. However, when the calendar event is a meeting or conference, the fingerprint method is preferable. Two types of Calendar events and their corresponding values are identified in Table 6.

Table 6. The values of calendar events

Calendar Events	Value
Idle	0
Meeting, Conference, Seminar	1

4.3.9 Weather conditions

Nowadays, weather forecast apps are an important part of smartphones and are one of the most frequently visited apps. The proposed approach uses a Weather App to check the weather conditions. We can get various details like temperature, precipitation, humidity, cloudiness, and so on. For example, the fingerprint sensor cannot read your fingerprint if your hand is wet, damp, or oily, so facial or voice-based recognition is more convenient. However, in our model, we considered three different types of weather including Sunny, Rainy, and Snowy as shown in Table 7.

Table 7. Types of weather values

Weather Types	Value
Sunny	1
Rainy	2
Snowy	3

4.3.10 SMS App

Today, any smartphone has SMS capabilities and can send and receive text messages to or from any other smartphone through the messaging app. In the proposed model, when the smartphone is stolen or lost, the genuine user sends the following SMS message to his/her smartphone "My phone is stolen or lost." We considered two different situations of a smartphone including stolen/lost or existent as shown in Table 8.

Table 8. The two different situations of a smartphone

Smartphone Situation	Value
Existent	0
Stolen/Lost	1

All the above features are important for the authentication model as they determine the authentication method to be selected.

4.4 Authentication model

An important component of this layer is the authentication model required to be defined. In the proposed model, we assume that the authentication model depends on the user's current context. Using context factors, we designed an authentication model whose type is determined by the value of motion, location, and noise level sensors as well as the software context (Weather, SMS, Alarm, and Calendar Apps) as shown in Figure 4. At this point in our model, we assume that the type of any authentication model describes a combination of all context factors.

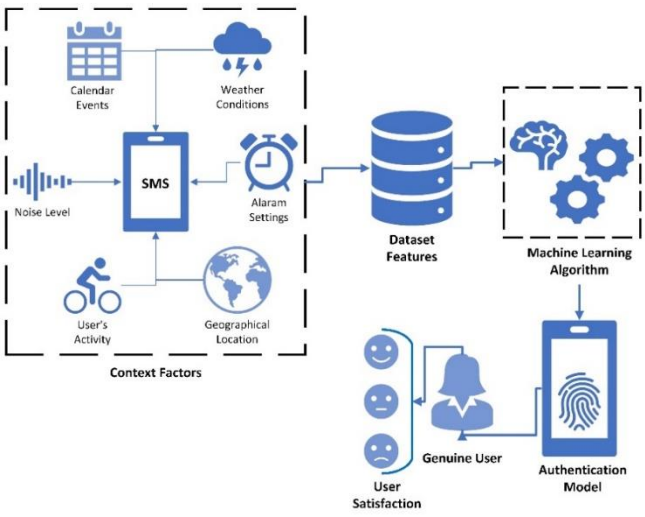


Figure 4. The process of designing an authentication model using machine learning algorithm

To find out features from the raw data, users are asked to go through the authentication process to unlock the smartphone screen so that we can get their responses according to the generated authentication model. To increase the accuracy of the proposed model, the dataset is updated each time the authentication process is performed. When a user inputs a PIN code, draws a pattern, and enters a fingerprint, the proposed model records the user's response using emoji sentiment (see Figure 4). Six methods of authentication methods and their corresponding values are identified in Table 9.

Table 9. The value describes the authentication method

Authentication Methods	Value
Password	1
PIN code	2
Pattern	3
Fingerprint	4
Voice-based	5
Face ID	6
Password, fingerprint, face ID	7

Machine learning algorithms are only as useful as the data used for training. The more data samples used during training, the more useful your algorithm will be. This is why it is necessary to have a large volume of processed datasets when working on artificial intelligence systems– so you can train the model effectively and gain the best outcomes. When we deal with machine learning models, data is fundamental. Without data, we cannot perform typical model training, and no insights are achieved.

We need to build a model that can predict the most appropriate and secure authentication method for a smartphone based on the user's context. In our case, the target variable (or label) is the authentication method, and the features are the user's body postures, location, noise level, weather conditions, calendar events, SMS messages, and alarm settings.

We will collect data from different users when they unlock their smartphone screens, with their corresponding inputs and outputs. This data is labeled and it's in the form of a user's context for each user. Pre-processing is performed to fill in any missing values or address outliers that might impact the accuracy of our model.

Depending on the user's context and surrounding information, the appropriate authentication model is selected, adapted, and delivered to perform the authentication process. For instance, the increase in smartphone mobility is less favorable towards a PIN code or password authentication model, so, when the user is running or driving, the most appropriate authentication method will be facial recognition or voice unlock respectively.

The smartphone detects the user's location and changes its authentication method, for example, automatically changing to pattern mode if the user is in a meeting, changing to fingerprint if the user is at home, and changing to voice ID whether the user is walking, running, or traveling by car. The smartphone detects whether a place is crowded such as an airport, railway station, or mall, and automatically changes the device authentication method to provide a non-voice command model.

However, if the smartphone is stolen or lost, the attacker may utilize lighting conditions to unlock the smartphone by picking up smudges remaining on the screen by the genuine user. Therefore, in our model, we add an extra level of security along with the password-based authentication method. We have implemented three levels of authentication. The first level is the symbolic password, the second one is the user's fingerprint, and the last stage is facial recognition. Therefore, by going through these stages, the genuine user can be identified, and access granted.

5. SYSTEM PROTOTYPE IMPLEMENTATION

5.1 Android App

To collect users' context data when the user frequently unlocks the smartphone screen, an Android background service is implemented to collect the user's context implicitly, and an Android App is developed to provide a convenient and adaptive authentication method as shown in Figure 5. All context information is collected from the application and system levels. The proposed system was installed by 25 users (15 males and 10 females) on their smartphones. The App uses sensors to capture real-time readings of sensors such as motion characteristics, physical location information, noise level, and software context (i.e., Calendar, Alarm, SMS, and Weather information).

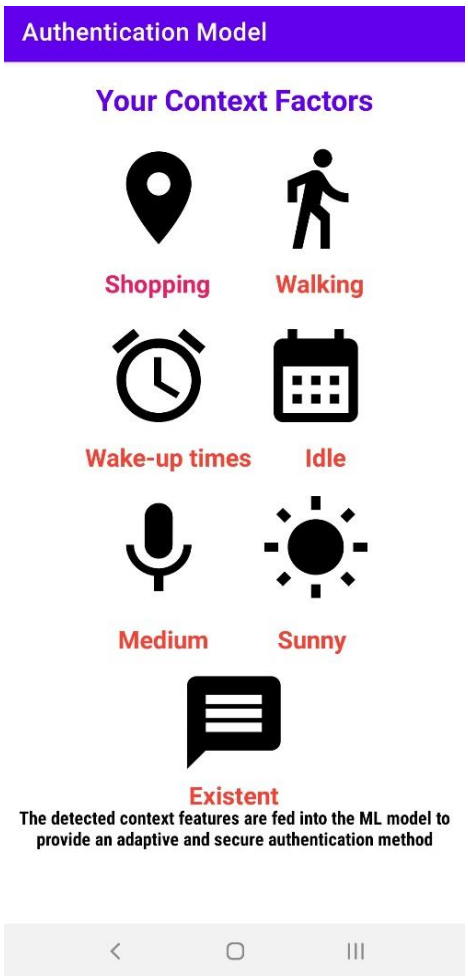
The Android environment offers a Sensor Manager class that enables developers to maintain the raw data for sensors at a static delay after registering sensors with listener methods. Using the Android sensor framework, we accessed the available sensors on the smartphone and obtained raw sensor data. The Android sensor framework offers many classes and interfaces to accomplish a variety of sensor-related operations. However, we have utilized the sensor framework to perform the following:

- Determine which sensors are available on a smartphone.
- Get raw data from sensors and define the minimum rate at which we get sensor data.

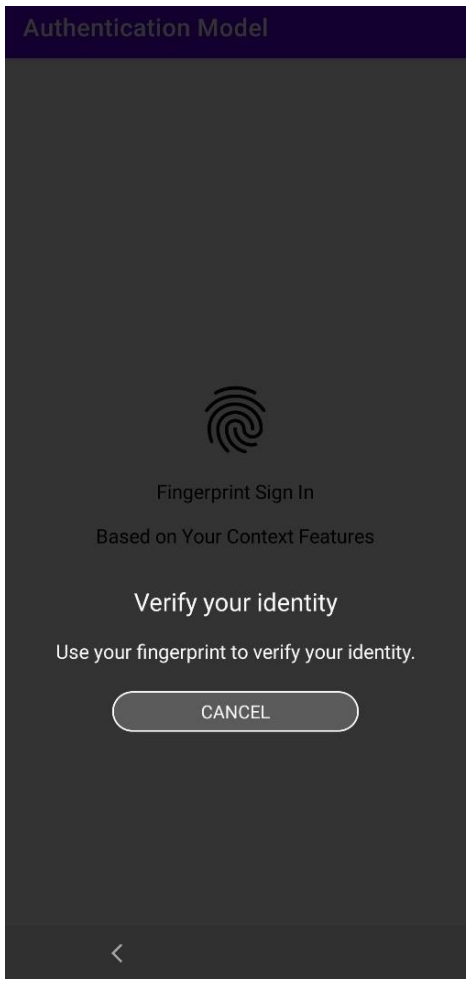
In general, the rate at which sensor data is acquired is usually referred to as the "sampling rate." It is usually expressed in Hertz (Hz), which refers to the number of data points collected per second. The minimum rate at which we get sensor data relies on the sensors used in the App. However, different sensors have different sampling rates. In the proposed system a high sampling rate is declared for the motion sensor and the sampling rate is limited to 200 Hz for others which results in more power consumption and storage.

Sensors are designed to collect the user's context and environmental information around her. Certain attributes are collected to define the user's body postures and activity, physical location, noise level, and software context. Significant user movement changes the user's geographic location; for example, hiking, running, cycling, or sitting in a moving vehicle.

Regarding weather conditions, there are numerous vendors that provide weather as an API. These APIs provide basic weather conditions such as temperature, probability of rain, humidity, pressure, cloudiness, wind, etc. In addition, Android allows calendar events to be read using the Calendar Provider API. The Calendar API manages access to a repository containing information about calendars. The Calendar Provider API enables users to perform the four basic operations (query, update, insert, and delete) on calendars, events, reminders, and so on. Calendar content providers typically expose information as a collection of relational database tables [26].



(a) User's context information



(b) Adaptive authentication method

Figure 5. Screenshots for the proposed adaptive authentication system

	A	B	C	D	E	F	G	H	I
1		Postures	Location	Noise	Weather	Alarm	Calendar	Existent	Model
2	0	1	1	1	1	10	0	0	2
3	1	1	1	3	2	5	0	0	3
4	2	1	2	2	1	10	0	0	2
5	3	2	4	3	2	10	0	0	4
6	4	4	3	3	1	10	0	0	5
7	5	3	3	2	1	10	0	0	6
8	6	2	3	2	3	10	0	0	1
9	7	1	5	3	1	15	0	0	5
10	8	1	1	2	1	15	0	0	5
11	9	1	2	1	1	15	0	0	5
12	10	2	4	3	3	10	0	0	6
13	11	3	3	3	2	10	0	1	7
14	12	1	2	2	3	10	1	0	4
15	13	1	2	2	1	15	1	1	7
16	14	2	4	2	1	10	0	0	4

Figure 6. Snapshot of the authentication model dataset

Moreover, user experience for the proposed authentication system is a critical part of security and usability. When the proposed system provides a dynamic and adaptive way to offer an authentication method based on user context factors, it is necessary to provide a more personalized and relevant experience to users. Therefore, the proposed system take into account achieving a balance between security and ease of use, as the system will provide a more secure, convenient, and accessible authentication method for users.

5.2 Machine learning algorithm: Multiple linear regression

To predict the authentication method based on the user context. In this case, there are seven independent variables, that is, the features, and one dependent variable, that is, the label, which is the value to predict. To work with datasets, the Pandas Python library is used. Pandas has functions for analyzing, cleaning, exploring, and manipulating data [14].

In the proposed model, a new machine learning dataset is created, and this collection of data is used to train our model (see Figure 6). A dataset includes several separate elements of data to train a machine learning algorithm with the purpose of discovering predictable patterns. The data is marked or annotated first so the machine learning algorithm understands what the result should be.

To perform multiple linear regression in Python, the sklearn library is imported [14]. Our dataset is divided into training and testing datasets. This is achieved by including (train_test_split) from the (sklearn.model_selection) library. We keep 80% of the data in the training dataset and the remaining 20% in the test dataset. The training dataset is one of the most critical subsets. This set contains the data initially used to train the model. In other words, it helps guide the algorithm on what to look for in the data.

Our dataset is labeled before being used by the machine learning algorithm, so that the algorithm knows what result to expect or classify as an anomaly. For example, when we try to predict the authentication method for unlocking the screen, we label the dataset with the names of the authentication models so that the machine learning algorithm can learn from the previous data. Since all context factors are described by distinctive values, the authentication model also is described by them.

We build our model based on the training dataset and test it on the test dataset. The test dataset is the input in the final phases of the training process. This subset is used to test the model's accuracy and show how much the model has learned.

6. EXPERIMENTAL RESULTS AND PERFORMANCE ANALYSIS

6.1 Participants

To collect data, 25 participants were invited to participate in the study. Participation is entirely voluntary. All participants are smartphone savvy and use authentication mechanisms to protect their personal information. Participants ranged in age from 18 to 45 years, with a good background on the direction of the study. The demographics of participants are shown in Table 10. These participants were asked to operate on their Android smartphones. Participants in a research study were given instructions via email, and these instructions were designed to ensure that participants fully understood what was expected of them. In addition, we take any ethical issues into account to protect user privacy.

Table 10. Demographic characteristics of the participants

Variable	Description	Percentage
Gender	Male	60
	Female	40
Age	18-22	28
	23-45	72
Background	Academia	24
	Public	76

6.2 Authentication methods setup

To evaluate the effectiveness of the proposed system, every time users turn on their smartphone or wake up the screen, they will be asked to unlock their smartphone, therefore, each participant needs to set up the screen lock in the following ways:

M 1: Pattern; Draw a simple figure with user's finger.
M 2: PIN; Enter four or more numeric code.
M 3: Password: Enter four or more characters or numbers.
M 4: Voice; Detects the user's voice and unlocks the smartphone when the user says the correct password.

M 5: Fingerprint; The sensor reads the user's fingerprint pattern.
M 6: Face ID; Takes images of user face from different angles. The face model is stored on smartphone.

6.3 Operational environment

Because this study is based on context-aware authentication, we consider different operational environments to collect data. To verify the relevance of the authentication model based on contextual awareness, we examine the performance of authentication in various usage contexts. Authentication experiments are performed in the following physical activities:
A1: Training sets are obtained from the data collected while a user is in a non-moving position.
A2: Training sets are obtained from the data collected during the user's walk.
A3: Training sets are obtained from data collected while the user is running.
A4: Training sets are obtained from data collected while the user is driving.

To make the collected data realistically reflect the user's attitude habits, all participants are required to use the smartphone at different time intervals: alarm information (bedtimes, wake-up times, etc.) and calendar events (meetings, conferences, etc.) with various weather conditions (rainy, sunny, etc.). Furthermore, the 25 participants need to use a smartphone to authenticate themselves while they are in different physical locations (home, office, park, etc.) and at different noise levels.

Collecting contextual data (i.e., personal or private information) may present ethical issues and needs compliance with privacy rules. The proposed system ensures that users know what data is collected, how it is used, and that they have provided their consent to data collection. However, the data is collected only for the specific goal for which individuals are informed. The system avoids using contextual data for unapproved purposes.

6.4 Evaluation criteria

6.4.1 Sentiment analysis

Due to the widespread popularity of emojis, many researchers have developed ways to integrate them into existing sentiment analysis methodologies [15]. The proposed model allows users to express their views and opinions using the most popular form of expression (i.e., Emojis). Each opinion expressed has an associated emoji called a sentiment (see Figure 7). Users were asked to provide their satisfaction or dissatisfaction regarding the authentication model. A total of 25 responses were collected as illustrated in Figure 8.

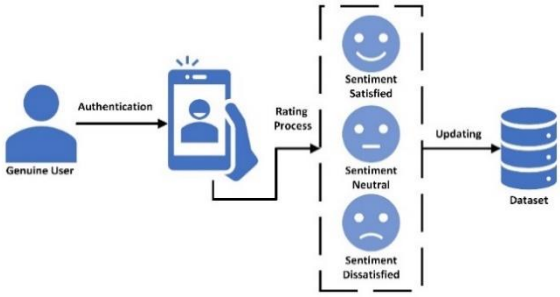


Figure 7. The process of expressing opinions using emojis

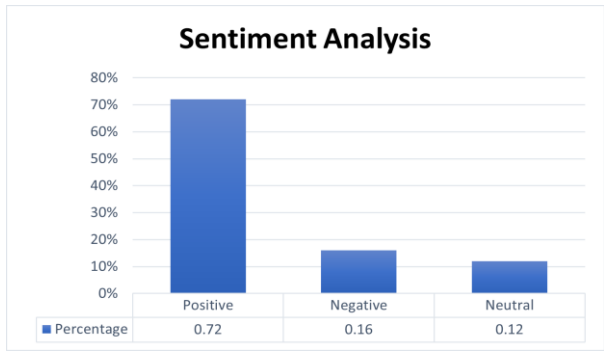


Figure 8. A percentage value for each sentiment emoticon

We analyzed user's satisfaction level with the proposed system. The sentiment ratio is calculated by comparing the number of negative sentiments emojis with positive or neutral emotions. From the analyzed data, we noticed that the sentiment ratio is 72 percent (see Figure 8), which means that context-aware technology is helpful in improving authentication adaptivity and provides a convenient authentication method.

6.4.2 Training and testing

To evaluate the performance of the proposed system, Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and R Square metrics are adopted. More specifically, MAE is the absolute difference between actual and expected values. The lower the value, the higher the performance of the model, which is defined as follows [14, 24]:

$$MAE = \frac{\sum_{i=1}^{n}|y_i - x_i|}{n} \quad (2)$$

where, y_i are the expected values, x_i are the actual values, and n is a total number of data points.

Whereas RMSE is the standard deviation of the errors that occur when making a prediction on a dataset. The RMSE is taken into account when determining the model accuracy. The lower the value, the higher the performance of the model. The RMSE formula is defined as follows [14, 24]:

$$\text{RMSE} = \sqrt{\sum_{i=1}^{N}(x_i - \hat{x}_i)^2} \quad (3)$$

where, N is the number of non-missing data points, xi is the actual time series of observations, and $\hat{x}_i$ is the estimated time series.

Finally, R-squared (R^2) is a statistical metric that describes the proportion of variance in a predicted variable explained by an input variable. It is the determination coefficient. It informs us how many points lie on the regression line. R^2 is defined as follows [14, 24]:

$$R^2 = 1 - \frac{\text{Sum Squared Regression (SSR)}}{\text{Sum of Squares Total (SST)}} \quad (4)$$

where, SSR is the sum of the squares of the residual values, and SST is the sum of the distance the data is completely away from the squared mean.

The MAE obtained for the proposed model is 1.299, which means it is pretty good because it is close to zero. An MAE of

zero indicates that the model is an excellent predictor of the outcomes. The calculated RMSE of the proposed model is 1.437, which is also pretty good. The R-squared value is 76.78, which means that 76.78% of the data fit the regression model as illustrated in Table 11.

Table 11. Performance of the proposed authentication system

#	Evaluation Metrics	Value
1.	Mean absolute error	1.299
2.	Root mean square error	1.437
3.	R squared	76.78

7. CONCLUSION

Context-aware authentication methods are a kind of security technique that deals with different contextual factors when giving access to individuals. These methods strive to enhance privacy protection, by understanding users and their surrounding environment. We have introduced a context-aware authentication system that takes into account environmental information that represents the user's context. This context information is used to create a user model to perform authentication. However, in our model, users tend to unlock their smartphones in a certain context. The proposed system is aware of contextual information regarding weather conditions, noise level, alarm settings, calendar information, body postures, and the physical location of the user. The system is trained using an appropriate dataset containing the features required by the authorized user. Before the actual authentication process, the proposed system makes smart decisions using user context information to provide a more convenient and secure authentication method. In the proposed model, we deal with multilinear regression and build a dataset that contains context information about the users.

The proposed system allows phone owners to express their opinions using sentiment Emojis. The work presented in this paper calculated the sentiment ratio expressed by owners immediately after the authenticating process. Experimental results showed that context-aware technology is helpful in improving adaptation performance. The performance of the model was tested, and the results showed that the authentication model effectively achieves an MAE of 1.299, an RMSE of 1.437, and an R-squared of 76.78.

However, the proposed system adds more complexity to the authentication process, which may make it less user-friendly. The system also relies on some technologies such as GPS, API, biometrics, etc. If data from physical or virtual sensors fails to be read, the authentication process may be compromised. For future work, there is a need to conduct extensive research on this context-aware authentication system by involving more users and capturing and analyzing various contextual factors.

REFERENCES

[1] Wang, R.Z., Tao, D. (2019). Context-aware implicit authentication of smartphone users based on multi-sensor behavior. IEEE Access, 7: 119654-119667. https://doi.org/10.1109/ACCESS.2019.2936034

[2] Shi, D., Tao, D., Wang, J.T., Yao, M.Y., Wang, Z.B., Chen, H.J., Helal, S. (2021). Fine-grained and context-aware behavioral biometrics for pattern lock on smartphones. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 5(1): 1-30. https://doi.org/10.1145/3448080

[3] Federica Laricchia, Number of mobile devices worldwide 2020-2025, the Statistics Portal, Statista. https://www.statista.com/statistics/245501/multiple-mobile-device-ownership-worldwide/, accessed on Jan. 20, 2023.

[4] Petroc Taylor, Smartphone subscriptions worldwide 2016-2021, with forecasts from 2022 to 2027, the Statistics Portal, Statista. https://www.statista.com/statistics/330695/number-of-smartphone users-worldwide/, accessed on Dec. 18, 2022.

[5] Benzekki, K., El Fergougui, A., ElAlaoui, A.E. (2018). A context-aware authentication system for mobile cloud computing. Procedia Computer Science, 127: 379-387. https://doi.org/10.1016/j.procs.2018.01.135

[6] Fan, B., Liu, X., Su, X., Hui, P., Niu, J. (2020). EmgAuth: An EMG-based smartphone unlocking system using Siamese network. In 2020 IEEE International Conference on Pervasive Computing and Communications (PerCom), Austin, TX, USA, pp. 1-10. https://doi.org/10.1109/PerCom45495.2020.9127387

[7] Feng, T., Yang, J., Yan, Z., Tapia, E.M. Shi, W. (2014). TIPS: Context-aware implicit user identification using touch screen in uncontrolled environments. HotMobile, 9: 1-9.

[8] Gandodhar, P.S., Chaware, S.M. (2018). Context aware computing systems: A survey. In 2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2018 2nd International Conference on, Palladam, India, pp. 605-608. https://doi.org/10.1109/I-SMAC.2018.8653786

[9] Bibri, S.E., Krogstie, J. (2017). The core enabling technologies of big data analytics and context-aware computing for smart sustainable cities: A review and synthesis. Journal of Big Data, 4(1): 38. https://doi.org/10.1186/s40537-017-0091-6

[10] Rivero-Rodriguez, A., Pileggi, P. and Nykänen, O.A. (2016). Mobile context-aware systems: Technologies, resources and applications. International Journal of Interactive Mobile Technologies (iJIM), 10(2): 25-32. https://doi.org/10.3991/ijim.v10i2.5367

[11] Pinjari, H., Paul, A., Jeon, G., Rho, S. (2018). Context-driven mobile learning using fog computing. In 2018 International Conference on Platform Technology and Service (PlatCon), Jeju, Korea (South), pp. 1-6. https://doi.org/10.1109/PlatCon.2018.8472763

[12] Masango, M., Mouton, F., Nottingham, A., Mtsweni, J. (2016). Context aware mobile application for mobile devices. In Information Security for South Africa (ISSA), Johannesburg, South Africa, pp. 85-90. https://doi.org/10.1109/ISSA.2016.7802933

[13] Skansi, S. (2018). Introduction to Deep Learning: From Logical Calculus to Artificial Intelligence. Springer.

[14] Igual, L., Seguí, S. (2017). Introduction to Data Science: A Python Approach to Concepts. Techniques and Applications, Springer.

[15] Godard, R., Holtzman, S. (2022). The multidimensional lexicon of emojis: A new tool to assess the emotional content of emojis. Frontiers in Psychology, 13. https://doi.org/10.3389/fpsyg.2022.921388.

[16] Chakraborty, B., Nakano, K., Tokoi, Y., Hashimoto, T. (2019). An approach for designing low cost deep neural

network based biometric authentication model for smartphone user. In TENCON 2019 - IEEE Region 10 Conference (TENCON), Kochi, India, pp. 772-777. https://doi.org/10.1109/TENCON.2019.8929241

[17] Filina, A.N., Kogos, K.G. (2017). Mobile authentication over hand-waving. In 2017 International Conference "Quality Management, Transport and Information Security, Information Technologies" (IT&QM&IS), St. Petersburg, Russia, pp. 69-74. https://doi.org/10.1109/ITMQIS.2017.8085764

[18] Zhang, X., Cheng, D., Dai, Y., Xu, X. (2018). Multimodal biometric authentication system for smartphone based on face and voice using matching level fusion. In IEEE 4th International Conference on Computer and Communications (ICCC), Chengdu, China, pp. 1468-1472. https://doi.org/10.1109/CompComm.2018.8780935

[19] Irvan, M., Nakata, T., Yamaguchi, R.S. (2020). User authentication based on smartphone application usage patterns through learning classifier systems. In IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, pp. 4462-4466. https://doi.org/10.1109/BigData50022.2020.9378172

[20] Ganesh, S.M., Vijayakumar, P., Deborah, L.J. (2017). A secure gesture based authentication scheme to unlock the smartphones. In Second International Conference on Recent Trends and Challenges in Computational Models (ICRTCCM), Tindivanam, India, pp. 153-158. https://doi.org/10.1109/ICRTCCM.2017.31

[21] Baldauf, M., Dustdar, S., Rosenberg, F. (2007). A survey on contextaware systems. International Journal of Ad Hoc and Ubiquitous Computing, 2(4): 263-277. https://doi.org/10.1504/IJAHUC.2007.014070

[22] Android Developers, Sensors Overview. https://developer.android.com/guide/topics/sensors/sensors_overview, accessed on Jul. 10, 2022.

[23] Ertel, W., Black, N.T. (2018). Introduction to Artificial Intelligence. Springer.

[24] Zafarani, R., Abbasi, M.A., Liu, H. (2014). Social Media Mining: An Introduction. Cambridge University Press.

[25] Han, J.W., Kamber, M., Pei, J. (2011). Data Mining: Concepts and Techniques. Morgan Kaufmann.

[26] Android Developer, Calendar provider overview. https://developer.android.com/guide/topics/providers/calendar-provider, accessed on May. 15, 2022.