M.Tech. Capstone Project
Continuous User Authentication

Supervised by: Arun Balaji Buduru

Rakesh Singh Rawat
MT17046

***PROBLEM STATEMENT*—Continuous User Authentication is a problem in which we have to continuously verify the genuineness of the user throughout the session. If any imposter, malware or bot is able to take over the access of the session then the model should be able to detect the vulnerability. Although, this can be done through consistent password prompts after a time interval but this deteriorates seamless experience, irritating the user. Present techniques focus on one time training of model, while this is insufficient as user behaviour may evolve over time. One time training doesnt address this issue, So the proposed method is aimed to provide lifelong learning (online learning) to handle the such scenarios that can occur due to change in user behavior throughout the usage of the device (mobile, a personal computer).**

I. Introduction

Passwords are used everywhere, from social networking to sensitive activities like banking transactions. Passwords are a great form of security by authenticating the user identity but passwords can only provide one-time security as these techniques rely on one-time authentication. One time authentication methods have several drawbacks: such as passwords can be stolen and misused. Also, if malware or imposter takes control of the system after one-time authentication, then this vulnerability can be dangerous. These vulnerabilities can be avoided by employing a password prompt after a fixed time interval to ensure that the user is genuine. However this naive technique has its own disadvantages such as it deteriorates the user experience. If the password was stolen, then imposter can easily bypass this feature. Also, due to multiple password user can get frustrated. These problems can be avoided by biometric-based continuous user authentication, which doesn't require the user to interrupt his work and pay attention to authentication procedure as all these processes are performed in the background. Traditional continuous authentication techniques rely heavily on machine learning models that are trained at the enrollment phase and then these models are used in the continuous authentication phase. However, these types of model training pose another challenge i.e. one-time training. User behavior is complex and can evolve over the period, learning this complex behavior involves the use of models that can learn the complex hypothesis and learning nature should be continuous. Also, as the user uses the device, more data is available over time that can help models to learn the user characteristics. In this project, these issues are solved by using online learning in the authentication pipeline. Recent advancements in online learning [18] shows great utility in these types of scenarios.

II. Dataset

Continuous authentication is performed on various types of devices. Datasets are available from these types of devices. Some of the target devices are:

- Smartphones [5] [6] [7] [11] [10] [15].
- Personal Computer/Laptop using webcam, keyboard and mouse dynamics [1] [2] [4] [13] [14] [15] [16].
- Wearable devices such as Google Glass [3].

For this work, Smartphone based Hand Movement Orientation and Grasp(HMOG) [21] dataset is used. HMOG uses accelerometer, gyroscope, magnetometer and touch screen events to construct behavioral profile through these signals. Accelerometer, gyroscope and magnetometer samples readings at 100hz. Data from three usage scenarios of smartphones were involved such as :

- Reading
- Texting
- Navigation on map

The dataset contains 100 users data of individual sessions. Each sessions records meaningful activities such as reading, navigation for the duration of 5 to 15 mins. For each user, 24 sessions of 3 different tasks are recorded. Dataset consists of 9 categories of data out of which 5 most prevalent categories are used in this project: Accelerometer, Magnetometer, Gyroscope, Touch events and Activities

Touch events contains several useful features like touch pressure, location of contact, pointer count, contact size etc. Activity records gesture scenarios such as sitting and walking which are one of the most important features.

III. Literature Review

Various work done till now can be summarized on the basis of various stages of continuous authentication.

A. Training Phase/Registration Phase/Enrollment phase

Initially, a genuine user profile is made by providing various tasks to the user so that useful features can be extracted.

*1) **Types of biometrics***

For this phase, different types of biometrics are used in the past. Soumik et al. [1] used mouse events to verify the user. Koichiro [2] and Koichiro [16] used soft biometrics such as face color histograms and clothes color from webcam. This mainly targets the problems that occur due to posture of the user. Mario [14] assumes that each users touchscreen usage can be unique as touch dynamics correlate with online signature verification. Mario [7] used only accelerometer while

[15] also used gyroscope and magnetometer for getting motion patterns. Mario [6] and Xiao [10] used touchscreen swipe and gesture data. Nawaf et al., [14] used gait based acceleration data along with smartwatch based gait and swiping and phone movements. Different from these biometrics, Upal [11] used application usage pattern of users.

2) Features

From these biometrics, useful features are extracted such as : Soumik [1] took 63-dimensional features of the mouse behavior such as:

- Length and number of movement
- Curvature and inflection
- Curve straightness characteristics

They considered four mouse actions such as mouse movement, point and click, and drag and drop for the mouse behavior. Instead of making features out of raw data, they only used raw data. Mario et al., [14] used touch strokes and geometric patterns such as start and end of a stroke, velocity etc. Mario et al., [7] divided accelerometer data into blocks and then used autoencoder based feature reconstruction. Mario [6] considers context such as sitting or walking for feature extraction and concluded that phone usage context boosts performance.

3) Feature selection/construction methods

In some techniques, important feature selection/construction is performed like Mario et al., [14] used Mutual Information for determining important features. Mario [6] used univariate feature selection out of 211 features. Koichiro [16] did PCA based feature reconstruction while Mario [15] used Siamese CNN for feature construction from motion data and Mario et al., [7] used autoencoders.

4) Training methods

Generally, training data is preprared by using genuine user samples and imposter samples (samples from other users) unlike Rajesh et al., [13] who used only genuine samples and modeled the task as outlier detection problem. They used fusion of many one class classifiers and compared their performance with multi class classifiers. Generally, classifiers such as SVM were used.

B. Identification phase/Continuous Authentication Phase

In this, CA is done by collecting the information in real time and testing on the trained model while maintaining genuineness score such as Trust Value [1], decay function [2], etc. Unsafe behavior will decrease the score; if it reaches beyond a particular threshold, then the user is prompted again for verification through hard biometrics. This re-verification is also called a termination step. Some techniques [2] use a weighted linear combination of features from soft biometrics and generate the probability of action being safe/unsafe.

Some other techniques that slightly different from above paradigm: Upal et al., [11] used smartphone app usage pattern as a biometric for user verification where unknown apps are handled by using smoothing in Markovian process. Models for different users are different based on their pattern of usage. Here, observation states are formed from application data and binned time data. Technique [10] studies the transfer of trained model across devices for seamless experience (One person uses multiple devices).

IV. Methodology

A. Online Learning Methods

All methods used one-time training which is done at the initial phase, after that the model is used in every session for testing purpose. But users can evolve their way of usage over time as they can slowly change the way they control their devices [14]. Also, one-time training involves a limited amount of training data and thus model may not be able to handle the concept drift that can occur long time after initial phase [18]. Also, the data from successful sessions (where there is no imposter alert in the sessions) can be leveraged for learning more complex functions. Hence, methods that can learn continuously on the fly(Online learning methods) can tackle these problems. Some of commonly used online learning methods are reinforcement learning and neural networks. However, continuous learning is difficult to achieve through traditional neural networks as the depth and capacity of networks has to be decided apriori, which leads to convergence issues [18]. There is another technique based on Deep Neural networks proposed by Doyen et al., [18] for continuous learning settings. In this technique, shallow network is involved in the begining that does fast convergence even on small number of instances. It then, gradually switches to a deeper model when more data is received. This adaptive capacity from shallow to deep is acheived through Hedge Backpropagation algorithm [18]. In this project, Hedge Backpropagation is used for learning complex user profile through continuous data.

B. Hedge Backpropagation

Doyen et al., [18] described their method as expert learning model through hedging strategy. Their model follows online setting where stream of data is used for continuously training the network. HBP uses overcomplete network where every hidden layer is attached to a classifier. Here, final prediction eq(1) is given by weighted combination of classifiers learned using hidden layers which resemble in the ensemble of classifiers.

$$F(x) = \sum_{l=0}^{L} \alpha^{(l)} f^{(l)} \tag{1}$$

$$\begin{gathered} f^{(l)} = softmax(h^{(l)}\Theta^{(l)}) \\ h^{(l)} = \sigma(W^{(l)}h^{(l-1)}) \\ h^{(0)} = x \end{gathered} \tag{2}$$

Here, $\theta^{(l)}$ are lth classifier parameter, $h^{(l)}$ is the hidden layer for classifier l, α is to be learnt using hedging algorithm. Loss function and α update is given by eq(3) and eq(4) respectively.

$$\mathcal{L}(F(x), y) = \sum_{l=0}^{L} \alpha^{(l)} \mathcal{L}(f^{(l)}(x), y) \tag{3}$$

$$\alpha_{t+1}^{(l)} \longleftarrow \alpha_t^{(l)} \beta^{\mathcal{L}(f^{(l)}(x),y)} \tag{4}$$

At every iteration, all α weights sum is equal to one. Gradient updates of classifiers are calculated in slightly different method eq(5).

$$\theta_{t+1}^{(l)} \leftarrow \theta_t^{(l)} - \eta \alpha^{(l)} \nabla_{\theta_t^{(l)}} \mathcal{L}(f^{(l)}(x), y_t) \tag{5}$$

However representation weights W updation is done is given by equation(6):summation is done from layer $j = l$. Weight updation of classifiers is followed by updation of α equation(7). This updation maintains a balance between exploration and exploitation.

$$W_{t+1}^{(l)} \leftarrow W_t^{(l)} - \eta \sum_{j=l}^{L} \alpha^{(j)} \nabla_{W_t^{(l)}} \mathcal{L}(f^{(j)}(x), y_t) \tag{6}$$

$$\alpha^{(l)} \longleftarrow max\left(\alpha^{(l)}, \frac{s}{L}\right) \tag{7}$$

Hedge Backpropagation has several advantages over other methods such as:(1) It makes predictions based on the ensemble decision with expert advice. (2) It enjoys properties of shallow as well as deep classifiers. (3) It can learn complex functions by switching into deep classifiers.

This technique is used extensively in this project by using HBP based model to predict imposter vs genuine user using features of HMOG dataset.

C. Feature Extraction

HMOG dataset contains smartphone sensor readings from accelerometer, gyroscope, magnetometer, and touchscreen. Accelerometer, magnetometer and gyroscope samples x,y and z coordinates of the device at 100hz. Many previous works [5] [7] [10] demonstrated significant performance by using only accelerometer and gyroscope data. Accelerometer determines the way user handles the device movements [21]. These three sensors collectively detect how the user can do hand micro-movements. Along with these readings, touch events such as single vs two finger usage, touch pressure also presents biometric traits of the user. Phone orientation and contact size provide information about phone handling style and contact area preference.The fusion of touch events and sensors provides enough evidence for unique signature of genuine user. The fusion of touch event and sensor readings are taken at a time window of 10ms, 50ms and 100ms. All readings which come under a common time window is summarized into meaningful features for the classifier. There are two categories of readings:(1) *Continuous* and (2) *Categorical*. Sensors provide continuous x,y and z data, contact size, touch location also provides continuous readings while some touch event such as pointer count, orientation etc provides categorical data. Based on these types of readings, features are extracted over a time window. For continuous data such as x,y and z coordinates, their mean, max and min values are taken. This is based on the intuition that these parameters can approximately identify the nature of coordinates over the time window. For categorical features most frequent value is taken into account as it gives the overall idea during that time window.

D. Feature Selection

All the features from fusion of touch events and sensors were not significantly correlated to the labels. Some of the features were less informative and redundant. Hence they are removed by Mutual Information score. Mutual Information is filter-based feature selection technique which gives ”the amount of information” about random variable X learned from other random variable Y. In Figure 2. some of the features are relatively low in MI score. These low scored features are removed before classification.

V. Experiments and Results

A. Type of features

Experiments were conducted on different feature combinations:(1) accelerometer and gyroscope, (2) On Touchevent data, (3) Fusion of touchevent, accelerometer, gyroscope and activities data.

B. Time Window

Experiments were performed on the three windows of $10ms$, $50ms$, $100ms$, out of which $50ms$ performed best for the sensor data alone. The performance is compared with offline Multi Layer perceptron with online learning [18], as shown in the Figure 1.

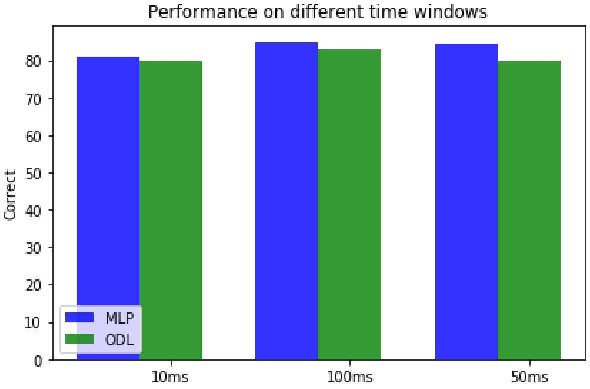

Fig. 1: Performance: static vs online learning on accelerometer + gyroscope

C. Training Data

Firstly, datasets were prepared from data of 6 users. Each time 1 user is considered as genuine class with 50% samples from genuine class, while others are considered as imposters.

D. Intermediate Results

Complete data of each user was split into 20% data for online learning and 80% for initial training. Out of initial training data, 70% train, 13% validation and 17% test data. This 17% test data was used for both initial training and online training as test data.

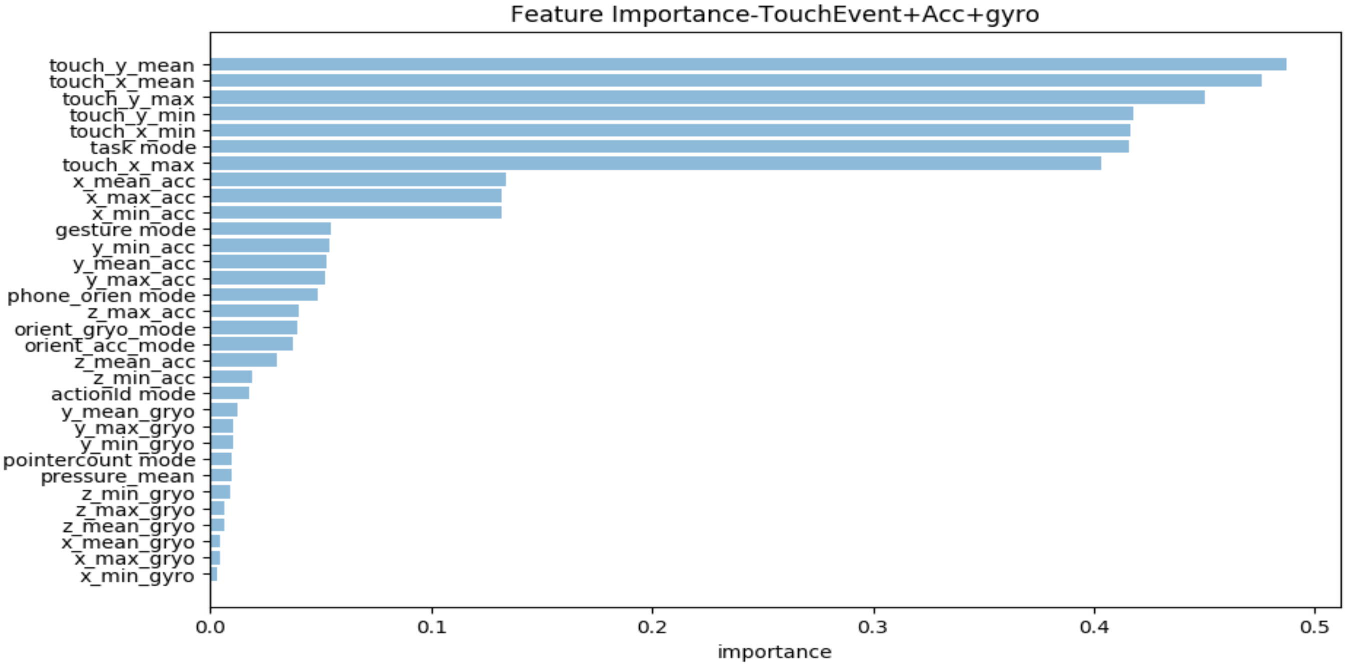

Fig. 2: All 32 Features and their importance according to Mutual Information

1) Accelerometer + Gyroscope Data

With Accelerometer and gyroscope data trained and tested on time windows of $10ms$, $50ms$ and $100ms$, model achieved best error rate of 17% on $50ms$ without feature selection.

2) TouchEvent Data

With TouchEvent data, the error rate was 27% which was significantly higher than accelerometer and gyroscope performance.

3) Fusion of Touchevent, Accelerometer and Gyroscope Data

Fusion of data gave 27.5% error without any feature selection. The total number of features of fusion data was 32 as shown in Figure 2., which after feature selection drops to 21. With 21 top selected features, classifier gave 18.3% error rate on 10 epochs.

E. Final Model

For best performing final model, user profiles were trained for each user with similar distribution described above(50% genuine, 50% others). Feature selection was done on fusion data which reduced redundant features, which reduced total features from 32 to 21 final features. The model is given initial boost by training it for 65 epochs. Complete data of each user was split into 70% train, 13% validation and 17% test data.

1) Metrics

To assess the performance of final models, ROC curve, AUC and Equal Error Rate is used along with error rate.

2) Results

Final model average error rate(averaged for all six users) after initial training for 60 epochs was 15.5. Equal error rate,AUC and ROC curve after initial training for each user profile is shown in figure 4. After online training on extra 20% data for only five epochs, the average equal error dropped to 13.4% as shown in figure 5. In terms of test data accuracies, the average of six users after initial and online training is 86.6% with highest being 89.3% and lowest being 85.3%.

3) Online Learning

To demonstrate online learning capability 20% data was kept aside for online training and an over-complete architecture was taken. Figure 3 shows the online learning in two settings where: (1)Initial data was trained for 1 epoch and extra data was trained for 4 epochs. (2) Initial data was given boost by training it to 60 epochs and extra data was trained for 5 epochs. From Figure [4][5], its clear that error rate further decreases after online training, extra data shown by dotted red line in figure 3. This proves that model after initial training boost further learns by training on extra data for only 5 epochs in an online setting.

VI. Conclusion

This project demonstrated Continuous Authentication in an online setting through the use of Hedge Backpropagation [18]. Thus, data from valid user sessions can be used to update the learning of a model and to leverage the additional data for modeling complex behavior. Feature selection resulted in significant boost in the performance. However, there is a very large scope for improvement through the use of RNN with Hedge Backpropagation, this will eliminate time window step and features can be more richer. However, we kept these improvements for future work.

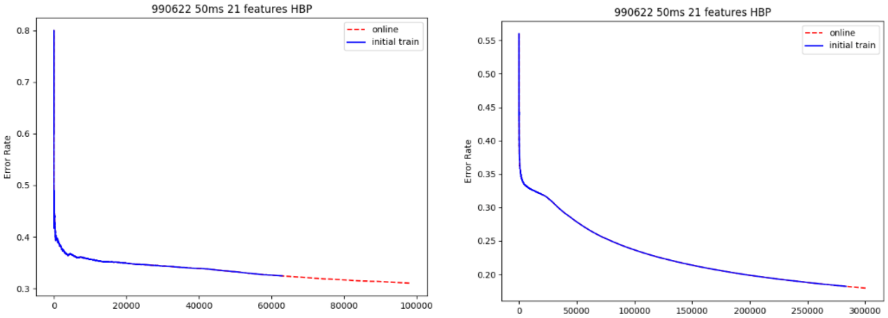

Fig. 3: (a) Left: 1 initial epoch + 3 epochs on extra data (b) Right: 45 initial epoch + 15 epochs on extra data

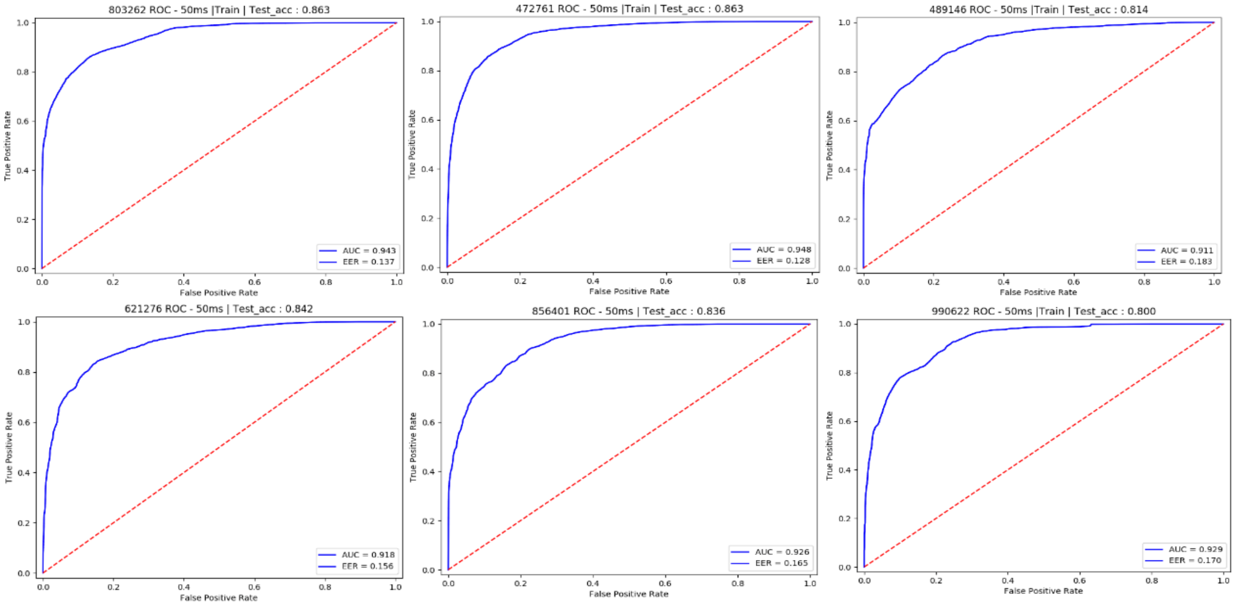

Fig. 4: Training data AUC, EER and ROC for six user profiles

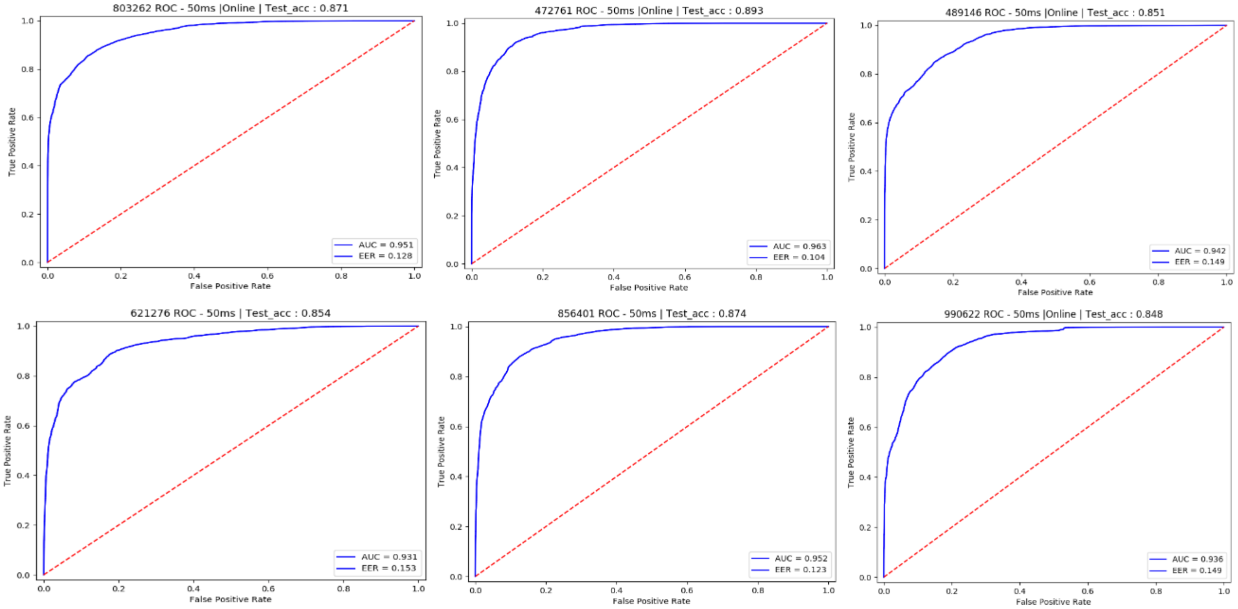

Fig. 5: Extra data (Online) AUC, EER and ROC for six user profiles

REFERENCES

[1] Soumik Mondal and Patrick,“Continuous authentication using mouse dynamics.

[2] Koichiro Niinuma, Unsang Park, Member, IEEE, and Anil K. Jain, Fellow, IEEE., “Soft Biometric Traits for Continuous User Authentication.

[3] Ge Peng, Gang Zhou and David T. Nguyen, Xin Qi, Qing Yang, and Shuangquan Wang., “Continuous Authentication With Touch Behavioral Biometrics and Voice on Wearable Glasses.

[4] Abdullah Alshehri, Frans Coenen and Danushka Bollegala, “ Towards Keystroke Continuous Authentication Using Time Series Analytics.

[5] Pouya Samangouei, Vishal M. Patel, and Rama Chellappa., “ Attribute-based Continuous User Authentication on Mobile Devices.

[6] Mario Parreno Centeno,Aad van Moorsel, and Stefano Castruccio, “ Effect of context in swipe gesture-based continuous authentication on smartphones.

[7] Mario Parreno Centeno, Aad van Moorsel and Stefano Castruccio.,“ Smartphone Continuous Authentication Using Deep Learning Autoencoders.

[8] Guannan Wu, Jian Wang , Yongrong Zhang and Shuai Jiang.,“A Continuous Identity Authentication Scheme Based on Physiological and Behavioral Characteristics.

[9] Ioannis C. Stylios,“ A Review of Continuous Authentication Using Behavioral Biometrics.

[10] Xiao Wang, Tong Yu, Ole Mengshoel and Patrick Tague.,“Towards Continuous and Passive Authentication Across Mobile Devices: An Empirical Study.

[11] Upal Mahbub, Jukka Komulaineny, Denzil Ferreiray and Rama Chellappa.,“Continuous Authentication of Smartphones Based on Application Usage.

[12] Nawaf Aljohani1, Joseph Shelton, Kaushik Roy.,“ Continuous Authentication on Smartphones Using An Artificial Immune System.

[13] Rajesh kumar Partha Pratim Kundu and Vir V. Phoha.,“Continuous authentication using one class classifiers and their fusion.

[14] Mario Frank, Ralf Biedert, Eugene Ma, Ivan Martinovic, and Dawn Song,“Touchalytics: On the Applicability of Touchscreen Input as a Behavioral Biometric for Continuous Authentication.

[15] Mario Parreo Centeno, Yu Guan, and Aad van Moorsel.,“ Mobile Based Continuous Authentication Using Deep Features.

[16] Koichiro Niinuma, Anil K. Jain.,“Continuous User Authentication Using Temporal Information.

[17] Sina Tabakhi, Parham Moradi.,“An unsupervised feature selection algorithm based on ant colony optimization.

[18] Doyen Sahoo, Quang Pham, Jing Lu, Steven C.H. Hoi.“Online Deep Learning: Learning Deep Neural Networks on the Fly. *IJCAI 2018*.

[19] Biocatch,“www.biocatch.com.

[20] Behaviosec,“www.behaviosec.com.

[21] Zdenka Sitov, Jaroslav Sedenka, Qing Yang,.“HMOG: New Behavioral Biometric Features for Continuous Authentication of Smartphone Users.