ISSN XXXX XXXX © 2017 IJESC

Research Article Volume 7 Issue No.6

Enhancing Performance and Usability of Keystroke Dynamics Authentication on Mobile Touchscreen Devices using Features Extraction Scheme

Dr.T.Pandikumar[1], Abraham Fekede[2], Capt.Zinabu Haile[3]
Associate Professor[1], M.Tech[2], Lecturer[3]
Department of Computer & IT
College of Engineering, Defence University, Debre Zeyit, Ethiopia

Abstract:
Behavioural biometric focuses on how and what we does using our body parts. Keystroke dynamics technology follows typing rhythm to distinguish genuine users from impostors. From cost perspective keystroke dynamics is very cost effective and can be integrated and work with different platforms. The dynamic feature of today's technology especially on IT increased the burdens on the end users by forcing them to remember different passwords. Now a day's the usage of smart phones become rapidly increasing in line with that the threats are also increasing as different studies showing. On this research thesis work we will implement keystroke dynamics for touchscreen devices having timing, size, pressure and location features to enhance authentication mechanisms. We will look using keystroke dynamics as an additional feature to enhance security of authentication, by using a smart phone having a touch screen as input method. The security mechanism for smart phones mostly depend on PIN(Personal Identification Number) which don't fulfil the current standard password requirements.by adding the above mentioned features as well as features extraction using Artificial Neural Network for the training and classifications purpose we implemented a keystroke dynamics on touchscreen devices for authentication and data collection we collected datasets from the implemented system determine which are the best features was analysed using random forest algorithm and Finally we evaluated the system by using standard metrics' EER, FAR and FRR also analyse and compare them with the previous similar experiments conducted and publicly available datasets we achieved the an EER of 0.75 .

Keywords: Behavioral biometric, Keystroke Dynamics, Mobile devices, Touchscreen devices

1. INTRODUCTION

1.1 Background

Most mobile phone services are provided via the internet making it ubiquitous, with a potential of unauthorized users getting unlimited access to the device. This may lead to data that is private and sensitive to the owner be stolen or abused. When a mobile phone in particular a smartphone is stolen, a lot of private and sensitive data can be compromised and be exploited for malicious activities, as the users of such phones usually are concerned about their sensitive data stored in the phone than the phone itself. In essence, when a smartphone is lost, the consequences that come with it are dire; they include privacy intrusion, user impersonation, and sometimes severe financial loss. As a first defense step, user authentication is essential to protect a system. Currently, user authentication systems for mobile phones are mainly based on three techniques: passwords, physiological biometrics and behavioral biometrics. Password authentication has well-known drawbacks, for instance, passwords can often easily be guessed and stolen through "shoulder surfing". Moreover, password authentication method brings insufficient security level because writing them down, using simple passwords, or reusing passwords make them easy to break. In this way, it is important to make and grow uniquely adjusted systems, which are intended to be usable under these limitations. Previous studies have reported that biometric based person recognition is a good alternative to overcome the difficulties of password authentication.

Biometric authentication is an authentication mechanism that uses human behavioral or physiological characteristics that are measurable, to define and represent the identity of a user. Human physiological biometrics are the physical human body characteristics that uniquely identify a person, these include fingerprints, retina and human face. Such biometrics are known to offer a consistent performance, however, they are also known to have a common disadvantage of being non-standardized and costly. In addition, physical biometrics are difficult and intrusive for collectability, low degree of user acceptability. In contrast, behavioral biometrics authentication relies upon a person's actions or habits to uniquely identify that person. Behavioral biometrics authentication may include signature recognition, mouse dynamics, touch gesture and keystroke dynamics. Behavioral biometrics can be an alternative to physical biometrics, therefore address some of the earlier pitfalls of physical biometrics. In addition, behavioral biometrics are easily implementable since they can be implemented at the software level. These biometrics can be unobtrusive and easily collected, without the user's knowledge. In addition, collection of data about the behavioral biometrics does not often require any special hardware and therefore it is cost effective. Furthermore, embedding fingerprint sensors into touchscreens behind gorilla glass is challenging, and has not been demonstrated. Another traditional biometric method is keystroke authentication that used traditional keyboards. This method could only provide temporal information, such as time interval between keystroke and time interval of a key being pressed. With the increased popularity of touchscreen mobile phones, touch gesture

behavior is increasingly becoming important in comparison to its counterpart the keystroke behavior, since almost all smartphones use the touchscreen as the main input method.

1.1 Statement of the Problem

Computing and communication technology usage are transforming from conventional desktop computers to mobile devices. In recent years, mobile applications related to online financial transactions have been developed to provide convenience to its users. There are recent innovations in mobile commerce that have enabled users to do transactions using their mobile device. These applications include purchasing goods, banking, and process point-of-sale payments. However, users are concerned with the security of their data. Due to the sensitivity of information, some users are afraid that their personal information will be stolen or hacked, so instead, they prefer not to use these applications and go the traditional way, thus defeating the purpose of mobility as well Knowledge-based authentication methods, such as passwords, PINs or pattern locks, are still the primary methods used to authenticate mobile users. However, these methods are vulnerable to a number of security threats or attacks, including brute force attacks, shoulder surfing, and smudge attacks.

On this research thesis work the aforementioned problems will be addressed using measurable features of individual and study the use of keystroke dynamics on touch mobile devices, as an authentication approach, based on experimental data collection and the newly prepared dataset.

1.5 Methodology

There are different sets of approaches will be followed among them:

- Conducting literature survey will to grasp the adequate knowledge needed to carry out this study.
- Collection of readymade datasets online.
- Developing sample dataset using java.
- Evaluating the system using WEKA simulating tool.
- The implementation of the technique and the algorithms will be carried out using simulator.
- Use different metrics to benchmark and analyze the system.

.

II. LITERATURE REVIEW

The keystroke dynamics research area has evolved into several branches of specializations covering keystroke features, anomaly detection models and classifiers, physical desktop keyboard studies, touch mobile devices studies, dataset collection studies, and multi-modality studies. In this section we will discuss selected research work that represents key areas of the keystroke dynamics area. On the research paper presented by (Alshanketi Issa Traore Ahmed Awad E., A 2016). A more secure alternative option which has gained interest recently is extracting keystroke dynamic biometrics from supplied passwords for mobile authentication. In this paper, they show that using random forests classifier, improved accuracy performance can be achieved for mobile Keystroke dynamic biometric authentication. They also propose a new algorithm for handling typos, which is an essential step in improving usability. They study both timing features and Pressure-based features. The best performance, obtained by combining timing and pressure Features, is an Equal Error Rate (EER) of 2.3% for a population of 42 users. In this study they considered both the standard features (dwell and flight times) and the pressure features (finger pressure and finger size). For the flight time, they consider the following variations: release-to-press (RP) (the duration of the time interval between a key released and a key pressed), press to-press (PP) (the duration of the time interval between two keys pressed), and release-to-release (RR) (the duration of the time interval between two keys released).

(Ala Abdulhakim Alariki , Azizah Abdul Manaf , Seyed The aim of touch gesture is developing a scheme to enhance the authentication accuracy and performance of determination users based on their touch gesture behavioral biometrics. From the existing works, most of the researcher collected and tested their methods in small group of users and advices. In addition, most of the schemes facing problem with accuracy based on score classifiers, which is the percentage of correct predictions. Extracting and testing more touch features such as finger pressure, finger size, touch minor and touch major would help to get better authentication accuracy. Furthermore, implementing one of the artificial intelligence classification methods other than Naïve Bayes will help to get better accuracy result. This research thesis develops a scheme to extract and study more touch gesture features and tested in large group of users by using multiple classification techniques, hence this will increase the accuracy with good performance authentication.

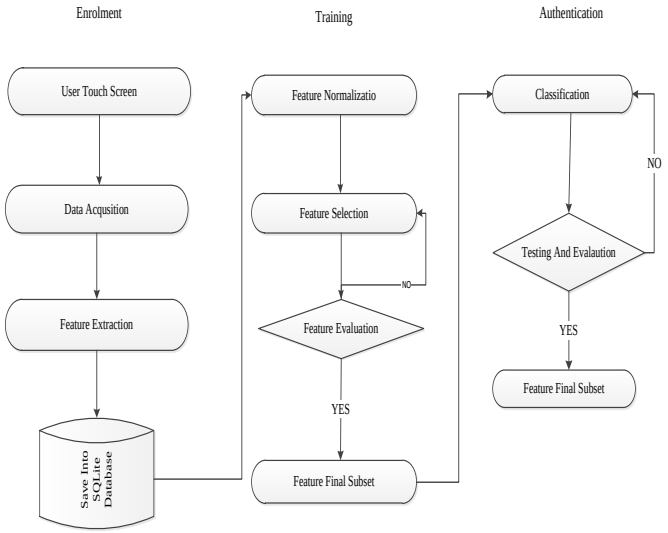

Figure.1. The framework consists of enrolment, training and authentication

They developed for Data Acquisition New dataset is created to evaluate the proposed system. To collect user touch gesture-data, they developed an Android program, which captures touch gestures using a standard API of Android system. For Feature selection also known as attribute selection or variable selection in statistics as well as machine learning, is done after feature normalization for the purpose of choosing the most significant feature in the dataset.

III.MATERIALS AND METHODS/METHODOLOGY

3.1 Data Collection and Feature Extraction

3.1.1 Data collection

In this research thesis we will analyze touch data collected from public dataset .An Android application was implemented in order to collect users' behavioral data. Besides touchscreen data such as touch position, pressure and finger area data were

also been collected and user specific features were extracted from the raw data.

3.1.2 Feature extraction

Three sets of features are defined in order to determine the effect of touchscreen biometrics in Addition to the well-studied keystroke biometrics. The three feature sets include: timing features, touchscreen features, and both timing and touchscreen features combined. The duration of each soft keystroke, and transitions between press-release and release-press events will be taken, for the total features. The non-timing touchscreen features are calculated similarly to the timing features. The pressure and the position of screen-touch x- y coordinates are defined as touchscreen measurements, for a total of three measurements. We listed touch behavior features which can be extracted from touch behavior biometrics. There are several different features of touch behavior biometrics which can be used when the user presses the touch screen.

3.1.2.1 X-COORDINATE

This parameter is a sequence of numbers which stores the finger position on X-axis on the Touchscreen while gesturing.

3.1.2.2 Y-COORDINATE

This feature is equal as the previous but it refers to the finger position on Y-axis on the touchscreen while gesturing.

3.1.2.3 FINGER PRESSURE

This parameter, like X and Y coordinates, keeps track of the finger pressure on the touchscreen. Pressure can be obtained by using Android API Motion Event. getpressure().The returned pressure measurements are of an abstract unit, ranging from 0 (no pressure at all) to 1 (normal pressure), however the values higher than 1 could occur depending on the calibration of the input device according to Android API documents .

3.1.2.4 FINGER SIZE

Size can be obtained by using Similar to Android API call MotionEvent.getsize() measures the touched size, associated with each touch event. According to Android document, it returns a scaled value of the approximate size for the given pointer index. This represents the approximation of the screen area being pressed. The actual value in pixels corresponding to the touch is normalized with the device's specific range and is scaled to a value between 0 and 1.

3.1.2.5 FINGER TIME

Time can be obtained by using Android API MotionEvent. getEventTime();.it retrieves the time this event occurred.

3.2 Authentication Classification

Classification is to find the best class that is closest to the classified pattern. Artificial neural network algorithm is used to classify the features in classification phase. We will define and use a threshold to decide if the user is the genuine one or an impostor. We choose to evaluate Artificial Neural Network algorithm for Authentication classification.

3.3 Technologies

3.3.1 Requirements

Latest Android OS with firmware has been chosen for the development. The language used in Android OS is Java. Hence, Java SDK is also required to develop software on Android OS.

Database used for mobile devices is primarily SQLite. Unlike most other SQL Databases, SQLite does not have a separate server process. SQLite reads and writes directly to ordinary disk files. All mobile devices having Android OS installed comes with touch pad. Thus touch keyboard should be used for giving inputs to the software.

3.3.2 JAVA

The Keystroke Dynamics software is built using Java Software Development Kit on Android OS. The integrated development environment used for the writing Java classes is Android Studio 2.3.2. The reasons behind choosing Java over other software languages include the following. The Android user space is largely dominated by Java technologies that run on top of Google's custom Dalvik Java virtual machine Java is simple, easy to implement and object oriented. Java provides high performance using its very large set of application programming interfaces (APIs).is robust and secure Java can provide multi-threaded programming so that the program execution is faster and it is dynamic. Java is platform independent, architecturally neutral and highly interpretable. Java has excellent set of Graphical user interface APIs in form of its abstract window toolkit (AWT) class as well Java Swing class. [Margit Antal , Laszlo Szabo ,Izabella Laszlo , October 2014]

3.3.3 Android

Android is a software stack for mobile devices that includes an operating system, middleware and key applications. The Android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language. Android was built from the ground-up to enable developers to create compelling mobile applications that take full advantage of all a handset has to offer. It was built to be truly open. For example, an application can call upon any of the phone's core functionality such as making calls, sending text messages, or using the camera, allowing developers to create richer and more cohesive experiences for users. Android is built on the open Linux Kernel. Furthermore, it utilizes a custom virtual machine that was designed to optimize memory and hardware resources in a mobile environment. Android is open source; it can be liberally extended to incorporate new cutting edge technologies as they emerge. The platform will continue to evolve as the developer community works together to build innovative mobile applications.

3.4.4 SQLITE Database

SQLite is an in-process library that implements a self-contained, server less, zero configurations, Transactional SQL database engine. The code for SQLite is in the public domain and is thus free for use for any purpose, commercial or private. SQLite is currently found in more applications than we can count, including several high-profile projects. SQLite is an embedded SQL database engine. Unlike most other SQL databases, SQLite does not have a separate server process. SQLite reads and writes directly to ordinary disk files. A complete SQL database with multiple tables, indices, triggers, and views, is contained in a single disk file. The database file format is cross-platform - you can freely copy a database between 32-bit and 64-bit systems or between big-endian and little-endian architectures. These features make SQLite a popular choice as an Application File Format. As mentioned previously, Android offers full support of the SQLite Database, with direct Java interfaces to access the API. However the recommended method to create a new SQLite

database is to create a subclass of SQLiteOpenHelper and override the onCreate() method, in which you can execute a SQLite command to create tables in the database. The SQLite Open Helper class is a helper class to manage database creation and version management. Using this class as parent offers an easy way to create the database when the application is started for the first time and each time the database schema is updated. (Noor Mahmood Shakir Al-Obaidi, 2016)

3.4.5 WEKA simulating tool.

WEKA tool is a collection of machine learning algorithms for data analysis and data mining tasks used for the evaluation in this research. The algorithms are applied directly to the collected input pattern touch-gesture data. WEKA implements algorithms for data pre-processing, feature selection, classification, and includes visualization tools.

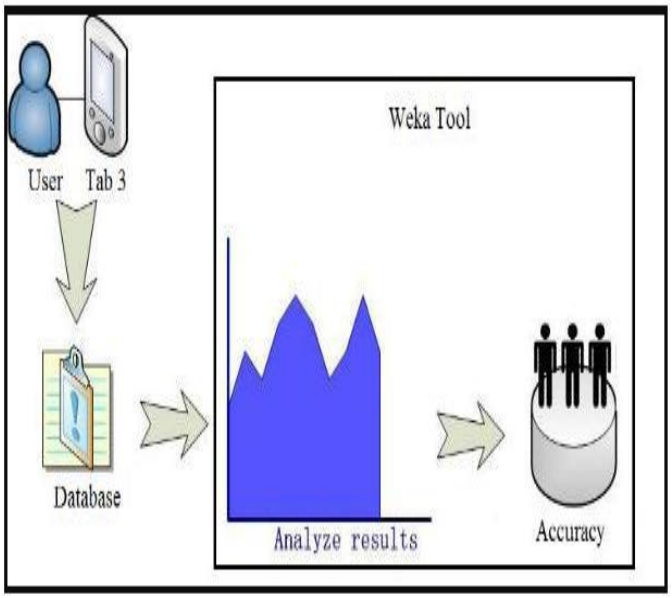

Figure. 2. Overview of evaluation

3.4.6 Android AVD Emulator

The implementation of the technique and the algorithms will be carried out using the simulation tool called android AVD emulator and gunny motion of independent than the studio environment. As well as we will install it on the real devices for showing the system.

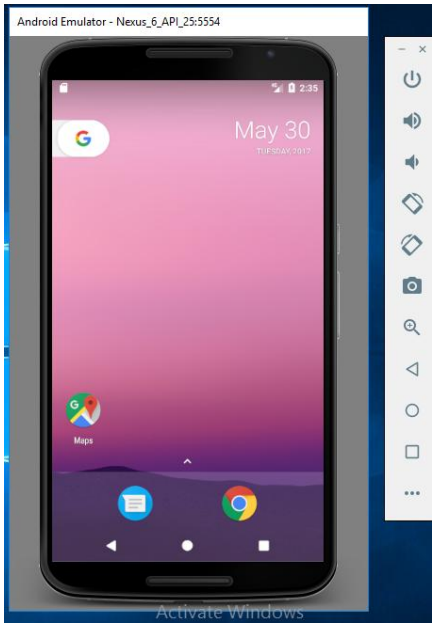

Figure .3. Gunny Motion Emulator

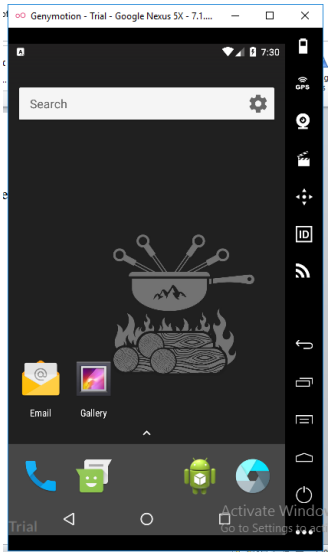

Figure. 4. Android built in Emulator API 25

3.4.7 DB Browser for SQLite

DB Browser for SQLite is an open source, freeware visual tool used to create, design and edit SQLite database files. Using DB Browser you can browse your data exported from real devices or any emulator you used you can import and export and also helps for editing your data as you like easy to use and it also have tool for drawing some graphs.

IV. DATA AND DATA ANALYSIS

4.1 The Proposed Keystroke Dynamics Model for Mobile Devices

Here we are presenting an authentication scheme that enhances performance, usability and implementations of keystroke dynamics on mobile touchscreen devices. We considered the recent works which have good results in the common metrics EER, FAR and FRR. For User authentication on computers using behavioural biometrics depends on training and feature extractions as well as classifications. The classification phase of an authentication system relies on pre stored training data on the selected feature set. On This chapter the analysis of the public dataset, design of the new model, and description of the implemented mobile KSD system will be discussed.

4.2 Feature Set for Touch Mobile Devices

In Mobile touchscreen devices have additional features that can be measured; such as pressure, finger area locations. In this research thesis we are adopting the same feature set of the SU work, with the addition of other features like 2 – graphs and 3 – graphs. Feature that covers the complete time of two successive and three keys.

4.3 Description of the Proposed System

The proposed mobile keystroke dynamics for touchscreen devices use artificial neural network to train the system and use the same for classifying and authenticate the user.

Have one basic interface and provides two main functions:

- Users are asked to register their password, and data will be collected on during the training phase.
- User authentication (testing phase).

4.3.1 Training Algorithm

The training algorithm performs the tasks of registering a new user, collecting keystroke data and storing the resulting training input layer.

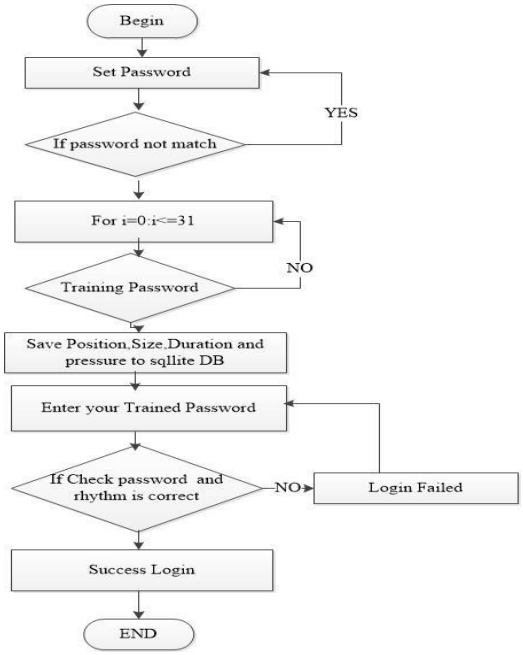

Figure. 5. Training Algorithm

4.4 Selected Model for KSD Artificial Neural Network

On the field of computation machine learning become a solution to simple to complex networks. It is used as dynamic learning tool from different types of machine learning we choose Multilayer perception (also called Back Propagated Delta Rule Networks) as our model for this research thesis work. MLP is feed forward Neural Network that maps sets input layer datasets on to sets of output layer and consists of a node having multilayer network. A feed forward neural network is an artificial neural network wherein connections between the units do not form a cycle. Each node is a processing element and unique with the activation function except the input node.

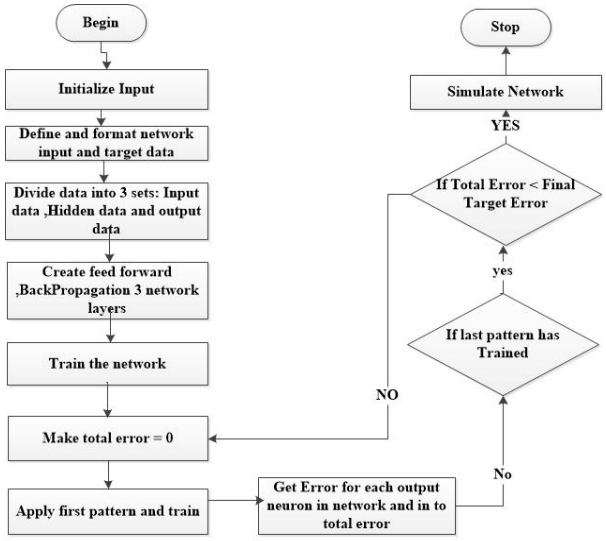

Figure. 6. Flow Chart Artificial Neuron network

MLP Multi-layer networks use a variety of learning techniques, the most popular being back-propagation. Here, the output values are compared with the correct answer to compute the value of some predefined error-function. By various techniques, the error is then fed back through the network. MLP utilizes a supervised learning technique algorithm called back propagation for training and testing. Supervised learning technique Networks that need a teacher to tell the network what the desired output should be while the unsupervised once adapts purely in response to its inputs. MLP is a modification of the standard linear perceptron which can distinguish data that are not linearly separable as in our case for typing behaviour of the users. MLP is a NN model that learns nonlinear function mappings and also capable of learning typing behaviour characteristics. This network was introduced around 1986 with the advent of the back-propagation algorithm. Until then there was no rule via which we could train neural networks with more than one layer a multi-layer perceptron can also learn non – linear functions. As the name implies, a Multi-layer Perceptron is just that, a network that is comprised of many neurons, divided in layers. These layers are divided as follows:

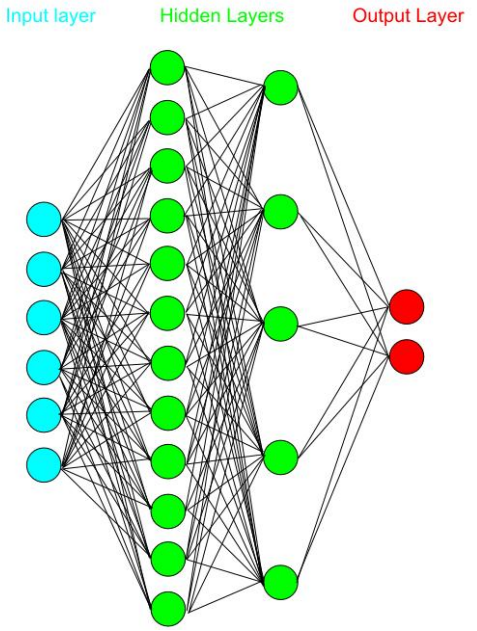

Figure.7. Structure of a Multilayer perceptron

- The **input layer**, where the input of the network goes. The number of neurons here depends on the number of inputs we want our network to get

- One or more **hidden layers**. These layers come between the input and the output and their number can vary. The function that the hidden layer serves is to encode the input and map it to the output. It has been proven that a multi-layer perceptron with only one hidden layer can approximate any function that connects its input with its outputs if such a function exists.

- The **output layer**, where the outcome of the network can be seen. The number of neurons here depends on the problem we want the neural net to learn

- The Multi-layer perceptron differs from the simple perceptron in many ways. The same part is that of weight randomization. All weights are given random values between a certain range, usually [-0.5, 0.5].

- Calculating the output, in this phase the output of the network will be calculated. For each layer, calculate the firing value of each neuron by getting the sum of the products of the multiplications of all the neurons connected to say neuron from the previous layer and their corresponding weights.

We gradually propagate forward in the network until we reach the output layer, and create some output values. Just like the perceptron these values are initially completely random and have nothing to do with our goal values. But it is here that the back-propagation learning algorithm comes.

V. RESULTS AND DISCUSSION

5.1 Overview

On This chapter how the keystroke dynamics is implemented on mobile touchscreen device using the ANN feature extraction and training will be discussed and also we present the experimental results the proposed system will have a data collection, Training and authentication schemes and also we present the feature selection in evaluation and for determining very important features using random forest algorithm.

5.2 Evaluation Methods and Metrics

The achieved result are evaluated using the three well known and used by almost all researchers on biometrics use them this are the error metrics equal error rate , false acceptance rate and false rejection rate. The three types of evaluations was discussed as below

FRR = number of refused genuine / Total number of genuine

FAR = number of accepted imposters / Total number of imposters

ERR = (FRR+FAR) / 2

On the Proposed KSD system keyboard prototype the collection of a user's input rhythm can be achieved by developing our own virtual keyboard, which we designed such that it could be easily installed on the selected mobile device. Our chosen mobile platform was Android OS. Data collection was performed with Genymotion emulator Google Nexus 5X - 7.1.0 –API 25 – 1080x1920, with the Android version (7.1.1) and Galaxy Note 3 with android version 5.0. In the case of Android, MotionEvent and Gesture classes provide functions to collect input data: input time duration, pressure level, size, and position coordinates.

5.3 System evaluation

We used WEKA for training and testing. It is a data mining tool developed at the University of Waikato in New Zealand. It employs a collection of Machine Learning (ML) algorithms for data mining tasks. The algorithms are applied directly to a dataset in different formats (.txt, .xls, and .csv). WEKA implements algorithms for data pre-processing, classification, and clustering; it also includes a visualization tools and GUI.

5.4 KSD Experiments

On This section we present the experimental part of the proposed KSD system. It also implements the following experiments:-

Experiment 1 implementation: On this part we present an implementation's of keystroke dynamics on mobile touchscreen devices is discussed.

Experiment 1 Result analysis: When we analyse the KSD FAR, and FRR, were used. These measures could be calculated form experiments results provided by WEKA.

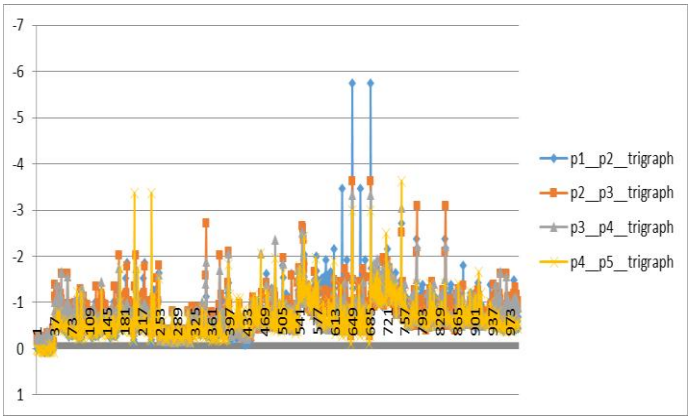

Figure .8. Trigraph of all Users

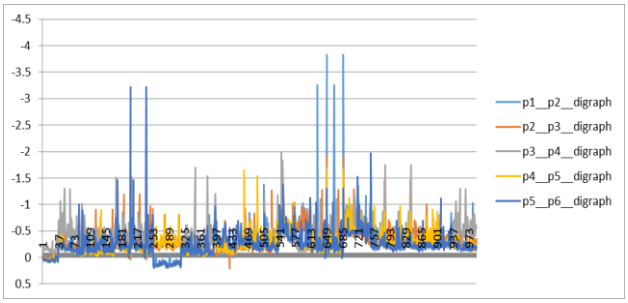

Figure .9. digraph of all users

Figure .10. Shows size touch event of user 1

VI. CONCLUSIONS

Our main purpose on this research thesis work is to propose keystroke dynamics feature for Android mobile touch screen devices. For achieving this, we developed a virtual keyboard which was used for gathering timing and non-timing features: time duration, pressure, size, and position of each character. WEKA was used for building the MLP model based on our own dataset. Keystroke dynamics for touchscreen devices is getting very successful .KSD provides acceptable level in performance measures as a second factor authentication. A model for defining a reliable and secure KSD authentication system is proposed. This model is the cheapest second factor authentication techniques, because no additional hardware is required. By focusing on Non-timing features we can increase the security level for our touchscreen Devices .The price advantage of it and Usability makes KSD in mobile devices preferable. We have compared and analysed different scholar's job and observed that significant changes are there in mobile touchscreen devices keystroke dynamics authentication. We contributed a new datasets for others interested on KSD mobile touchscreen devices.

6.1 Future Work

As a future work we believe that the accuracy we have achieved is good if it is repeated with different scenario and improved performance will be achieved. And the public datasets available for touchscreen devices is not adequate.

VIII. REFERENCES

[1]. Alshanketi Issa Traore Ahmed Awad E. A. "Improving Performance and Usability in Mobile Keystroke Dynamic Biometric Authentication" 2016 *IEEE Security and Privacy Workshop*

[2]. Ala Abdulhakim Alariki , Azizah Abdul Manaf , Seyed Mojtaba Mousavi "Features Extraction Scheme for Behavioural Biometric Authentication in Touchscreen Mobile Devices" *International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 18 (2016) pp 9331-9344*

[3] . Kyle R. Corpus, Ralph Joseph DL. Gonzales , Alvin Scott Morada , "Mobile User Identification through Authentication using Keystroke Dynamics and Accelerometer Biometrics **"** 2016 *IEEE/ACM International Conference on Mobile Software Engineering and Systems*

[4]. Daniel Buschek, Alexander De Luca Florian , Alt CHI 2015, *Crossings, Seoul, Korea* "Improving Accuracy, Applicability and Usability of Keystroke Biometrics on Mobile Touchscreen Devices"

[5]. Matthias Trojahn Frank Ortmeier**,** "Toward mobile authentication with keystroke dynamics on mobile phones and tablets "on 2013 27th *International Conference on Advanced Information Networking and Applications Workshops*

[6]. TaiChung, Taiwan , Emanuele Maiorana, Patrizio Campisi, Noelia González-Carballo, Alessandro Neri , **"**Keystroke Dynamics Authentication for Mobile Phones"

[7]. Rosy Vinayak, Komal Arora , "A Survey of User Authentication using Keystroke Dynamics" *International Journal of Scientific Research Engineering & Technology (IJSRET*), ISSN 2278 – 0882 Volume 4, Issue 4, April 2015

[8]. Pin Shen Teh a,*, Ning Zhang a, Andrew Beng Jin Teoh b, Ke Chen a "A survey on touch dynamics authentication in mobile devices" *computers & s e c u r i t y 59 (2 0 1 6) 210–235*

[9]. Rohit A. Patil Amar L. Renke , " Keystroke Dynamics for User Authentication and Identification by using Typing Rhythm*" International Journal of Computer Applications (0975 – 8887) Volume 144 – No.9, June 2016*

[10]. D. W. Salil Partha Banerjee, "Biometric authentication and identification using keystroke dynamics: A survey," *Journal of Pattern RecognitionmResearch., vol. 7, no. 1, 2012.*

[11]. "Keystroke dynamics on android platform," Procedia Technology, vol. 19, pp. 820 – 826, 2015. [Online]. Available*: http://www.ms.sapientia.ro/~manyi/keystroke.html*

[12]. Mradul Shrivasatva "keystroke dynamics for mobile devices– algorithm and authentication" Master of Science in *Computer Science San Diego State University, 2011*

[13]. Noor Mahmood Shakir Al-Obaidi "A New Statistical Anomaly Detector Model for Keystroke Dynamics on Touch Mobile Devices "*Middle East University May, 2016*

[14]. Shea Allison Ryan "Mobile keystroke dynamics: assessment and implementation *"December 2014*

[15]. http://www.coolestech.com/rhu-keystroke/

[16]. http://www.cs.cmu.edu/~keystroke/

[17]. https://www.ms.sapientia.ro/~manyi/mobikey.html

[18].https://www.quora.com/What-is-the-sigmoid-function-and-what-is-its-use-in-machine-learnings-neural-networks

[19]. https://anuradhasrinivas.files.wordpress.com/.../error-backpropagation-algorithm1.doc