A Simple Authentication Method with Multilayer Feedforward Neural Network Using Keystroke Dynamics

Ahmet Melih Gedikli(✉) and Mehmet Önder Efe

Department of Computer Engineering of Hacettepe University, Ankara, Turkey
{ahmet.gedikli,onderefe}@hacettepe.edu.tr

Abstract. Keystroke dynamics is a widely accepted user recognition and verification behavioral biometric, which has been studied nearly for a century. Intrinsically, this biometric is used together with id/password authentication forming multi-factor authentication. There are several anomaly detection algorithms that have been proposed for this task. While some proposals handle this problem with measuring data distance by taking correlation and dependence into account, some models use complex and time-consuming models deep neural networks to train to reach the right approximation. Our paper addresses a simple, accurate and lightweight method for user authentication. We show the effectiveness of our approach through comparisons with existing methods, which have also used the CMU keystroke dynamics benchmark dataset used here too. Using feed forward multilayer neural network with resilient backpropagation, we obtained an Equal Error Rate (ERR) equal to 0.049 for authentication with overall identification accuracy of 94.7%.

Keywords: Feed forward multilayer neural networks · Keystroke dynamics · User recognition and authentication · Biometrics · Resilient backpropagation

1 Introduction

Today's world requires fast, reliable, secure and easy to use/access to information. Security concerns paved the way for many different techniques uses user information such as passwords and user details. However, such information brings the threat together if they are not used combined with other techniques. A well-discussed, foolproof, automated and proven technique is biometrics, which uses personal characteristics and unique individual behaviors such as voice, fingerprint patterns [1]. Combining password security with biometrics forms a multi-factor authentication [2]. Physiology based and behavior based systems together forms the biometrics systems. While fingerprint, 2D face and voice authentication systems are physiology based ones, on the other hand, behavior based approach is comprised of keystroke dynamics on keyboard, touch screens and mouse click patterns. A behavioral biometric method used in user verification and identification is keystroke dynamics, which analyses typing rhythms of users and classifying them according to their keystroke behaviors. Each individual has unique keystroke timing patterns which can form a user protective

C. Djeddi et al. (Eds.): MedPRAI 2019, CCIS 1144, pp. 9–23, 2020.
https://doi.org/10.1007/978-3-030-37548-5_2

evaluation. Using this protective evaluation to authenticate using a compromised password could be detected and rejected immediately because evaluation consists an undoubtedly different pattern from genuine one. Typing behavior subject is firstly touched on as idiosyncratic behavioral characteristics in 1936 [3]. Back in the 19th century, telegraph operators could recognize each other based on their typing rhythms [4]. Also comparing other biometric systems, keystroke dynamics has more advantages like being user-friendly and non-intrusive. Continuous authentication is possible with no need of user awareness and additional required hardware equipment. There are numerous research that physiology based authentications like fingerprints, 2D face and voice can be imitated easily [5–9]. Also, it is proven that a weak password with keystroke dynamics supported authentication too can be attacked by imitators [10]. However, the attacker has to know the whole typing behavior of the subject to intrude. On the other hand, a strong password can't be imitated easily with keystroke dynamics [10]. In most cases the username and password are leaked, but the typing behavior information is not easy to be captured. Therefore, keystroke dynamics is one of the most secure biometric which can't be imitated if a strong password is used along with it. In summary, keystroke dynamics biometrics is cut out mechanism for user authentication since it is software based, easy, cheap and online [2].

Keystroke dynamics of an individual shows inconstancy due to external factors like input keyboards, different keyboard layouts etc. and transient internal factors such as emotion, stress, drowsiness [11]. These results basically show that a genuine individual will be eliminated when s/he is under a threat and forced to be authenticated due to affected neurophysiological pathway.

Keystroke dynamics features are extracted using timing information of key up, key down and key hold events. These features forms digraphs, which are the time latencies between two successive keystrokes, trigraphs, which are the time latencies between every three consecutive keys and n-graphs which are time latencies between every n consecutive keys. Digraphs, trigraphs and n-graphs are discriminative at word-specific level. These extracted features especially used in user classification. In this paper, we use a static text for verification and authentication.

Neural network based models are frequently used in the field of computer vision, speech signal processing, text representation and automatic control systems. They are also widely used and adopted to computer security domains. Being different from classical methods, which rely on complex distance metrics or manual feature engineering, neural network models have some advantages like simplification of the whole process and getting a scalable problem definition with property of high performance. With motivation of superior performance of neural network models in some problem sets, in this paper, we used a multilayer simple feedforward neural network for user authentication. We have trained three different neural network structure with CMU keystroke dynamics benchmark dataset, each of which has different number of hidden layers and activation functions between hidden layers.

This paper is organized as follows. In Sect. 2, we give the background. In Sect. 3, we described CMU Keystroke Dynamics Benchmark Dataset. In Sect. 4, we introduced our neural network model. In the fifth section, we explained the training mechanism and evaluation of the neural network structures. In the sixth part of the paper, we discussed the results and the conclusions are given at the end of the paper.

2 Related Works

Distinguishing users via keystroke patterns was first discovered in 1970s [12, 13]. These works were focused on static type of text. The long passage text identification by keystroke dynamics was also considered in [14]. Later digraph and trigraphs' mean and variances were used first to extract the keystroke features by Monrose and Rubin [15]. They used the Euclidean distance metric with Bayesian-like classifiers and observed quite successful results. A number of detailed survey papers were published from 2009 to 2015 each of which takes different perspective of keystroke dynamics [2, 16–22]. To extract keystroke features, relative order of duration times for different *n*-graphs is proven to be stronger to intra-class differences than absolute timing [23].

2.1 Distance Based Classification

In this approach, feature vectors are extracted from typing behavior. These vectors are then classified for authentication and verification procedures. Euclidean distance is used in early days due to its simplicity, but it has drawbacks. It was highly sensitive to scale differences in the extracted features and it cannot deal with correlation between the vectors. Mahalanobis distance, however, takes covariances of data to reduce heterogeneity in real data. Mahalanobis distance is widely used for comparing features via disconnecting the interactions between features based on their covariance matrix [24]. Another method which is the Manhattan distance have become prominent with simple computation and easy breakdown into contributions made by each variable. From this perspective, it is hard to be influenced by outliers when compared to higher order distance metrics including Euclidean distance and Mahalanobis distance. A performance comparison study showed that the Manhattan distance pointed out that the top performers are classifiers using scaled Manhattan distance with an equal error rate of 0.096, and the nearest neighbor classifier using the Mahalanobis distance with an equal error rate of 0.10 [25].

2.2 Advanced Machine Learning Based Classification

Over the years, keystroke biometrics research has taken advantage of many existing classification techniques including K-means methods, K-nearest neighbor classifiers, Bayesian classifiers, fuzzy logic, boost learning and random forests. Support vector machines are used to accommodate non-linear decision boundaries for complicated classification issues. The persistent features of keystroke dynamics are extracted using SVMs and used in classifying user typing [26]. Deep learning techniques has also been used in classification and reported that it outperforms before mentioned techniques [31, 33, 34]. In these models, Deep Learning structure is fed with timing features of keystroke dynamics, since the training procedure can take quite a long time, ADAM optimization and Leaky Rectified Linear Unit are used for faster learning process [33]. Besides faster convergence, the most valuable EER is obtained in [33]. Another research used Deep Belief Nets to extract hidden feature detectors and those feature detectors were used building a pretrained Artificial Neural Network for real training process instead of starting to the training with a random model [31]. More complex Deep

Learning models were also used recently in this research area. For example, Recurrent Neural Network (RNN) with Convolutional Neural Network (CNN) was used in a research in whose model 5 different sequence length of texts from 10 to 100 (10, 30, 50, 70, 100) and 3 keystroke time characteristics were used for evaluation [35]. The CNN was used for extracting high level timing features, later, these features were provided as inputs to the RNN. It is stated that using 30 sequence length of texts and 3 keystroke time characteristics results better than other cases. In addition to Deep Learning techniques, a recent research used NeuroEvolution of Augmenting Topologies (NEAT), which is a type of Genetic Algorithm, resulted highest identification accuracy, however, they have built their own dataset and tested their models with that [36].

3 CMU Keystroke Dynamics Benchmark Dataset [25]

The dataset provided in CMU Keystroke Dynamics Benchmark consists of three types of timing information named the hold time, key down-key down time and key up-key down time. This timing information was collected for one static password, which is `.tie5Roanl` with keystroke timing information of 51 users. Whole data for one user is collected in 8 different sessions with 50 repetitions on each one of them. Figure 1 illustrates keystroke timing types. Hold time or dwell time in Fig. 1 represent the duration of time during pressing a key, concisely it is calculated as the difference time between release and press of a single key. In Fig. 1, Hold Time represents this type. Key-down key-down time is the time from pressing a key to pressing a consecutive key which is represented Down-Down Time in Fig. 1. Finally, key-up key-down time is the time from releasing a key to pressing a consecutive key. This type of time can be negative because user might have not released the former pressed key while pressing last consecutive key. In Fig. 1, Up-Down Time demonstrates this type of timing. There are 31 features for each trial of user with 11 of them are hold time ending with introducing enter key, 10 of them are key-down key-down time, finally 10 of them are key-up key-down time.

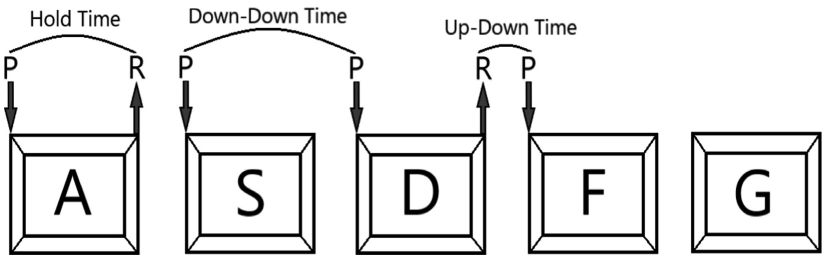

Fig. 1. Demonstration of features in keystroke dynamics benchmark data set

4 Neural Network Model

CMU Keystroke Dynamics Benchmark Dataset is used as input to our neural network layout. We start with a subject considering as genuine user. We took other 50 subjects as impostors to that user's authentication. The first 200 timing features of genuine user and the impostors' features are used as training data for a genuine user. This anomaly detection process has been repeated for each user taking that specific user as genuine and the others as impostors. Using impostors' data in training process leverages capability of differentiating features extracted by the neural network model. The other 200 timing features of genuine user is used as validation and randomly selected 5 timing features for each impostor, in total 250, are used as test data. These evaluation criteria for CMU Keystroke Dynamics Benchmark Dataset are mentioned in [25]. For each user, there exists a trained neural network model produces output between 0 and 1 as response to timing features. We designed three different neural network models. We started by designating one of our 51 subjects as the genuine user, and the rest as impostors. We train an anomaly detector by extracting 200 initial timing feature vectors for a genuine user from the dataset. We repeat this process, designating each of the other subjects as the genuine user and the remainders are as impostors. Thus, number of

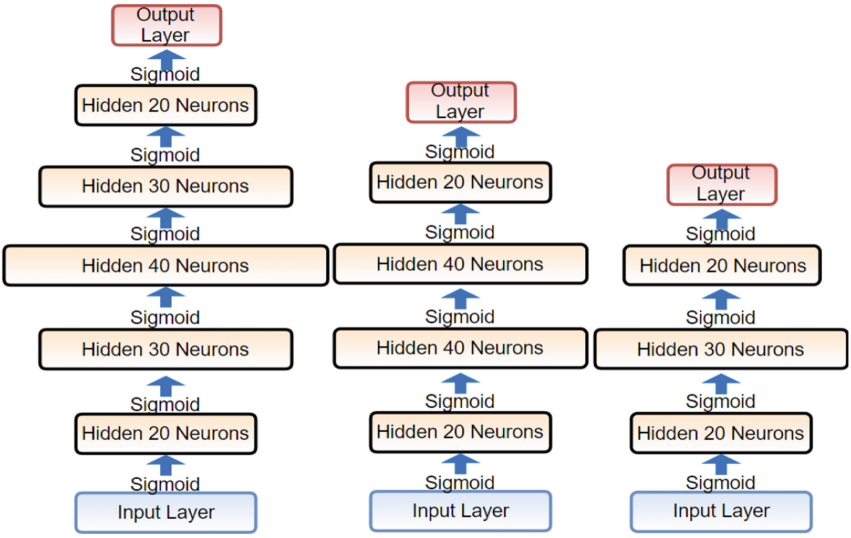

Fig. 2. Illustration of three neural network models

created models equal to number of distinct subjects. Three different neural network models are represented in Fig. 2. Training process were done in each model for every user for 10 times. We used resilient backpropagation (Rprop) method changing the weights with using momentum factor to diminish the fluctuations in weight changes over consecutive iterations. Equation (1) describes this procedure where $E(\omega)$ is the loss function, ω is the weight vector and η is the learning rate.

$$\Delta\omega_i(t+1) = -\eta \frac{\partial E}{\partial \omega_{i,j}} + \alpha \Delta\omega_i(t) \tag{1}$$

There are numerous research outcomes that prove resilient backpropagation is more successful than plain error backpropagation [27, 28]. Also, we decreased the learning rate parameter in each epoch towards a predefined minimum value. The algorithm in Fig. 3 states Rprop backpropagation process with weight momentum factor and decreasing learning rate. While Δ denotes Rprop weight changes, E denotes mean square error in one epoch and $\omega_{i,j}$ denotes a weight between neurons. Neural network hyperparameters used in algorithm shown in Fig. 3 are listed in Table 1 with the corresponding initial values. All three neural network models in Fig. 2 were initialized to the values listed in Table 1.

Table 1. Hyper parameters used in neural network model with their initial values

Hyperparameter name	Value
Learning Rate (η)	0.15
Minimum Learning Rate (η_m)	0.05
Learning Rate Decrease Value (η_d)	0.0001
Momentum Alpha (α)	0.05
Rprop Learning Rate Plus (η^+)	1.2
Rprop Learning Rate Minus (η^-)	0.5
Rprop Minimum Delta Weight (Δ_{min})	10^{-6}
Rprop Maximum Delta Weight (Δ_{max})	10
Initial Rprop Delta Weight (Δ_0)	0.9

$\forall i,j:\ \Delta_{i,j}(t) = \Delta_0$

$\forall i,j:\ \frac{\partial E}{\partial \omega_{i,j}}(t-1) = 0$

Repeat
Compute Gradient $\frac{\partial E}{\partial \omega}(t)$:
For all weights and biases:

IF $\frac{\partial E}{\partial \omega_{i,j}}(t-1)\frac{\partial E}{\partial \omega_{i,j}}(t) > 0$

$\Delta_{i,j}(t) = \min(\Delta_{i,j}(t-1)\eta^+, \Delta_{max})$

ELSE IF $\frac{\partial E}{\partial \omega_{i,j}}(t-1)\frac{\partial E}{\partial \omega_{i,j}}(t) < 0$

$\Delta_{i,j}(t) = \max(\Delta_{i,j}(t-1)\eta^-, \Delta_{min})$

$\Delta\omega_{i,j}(t) = -sign\left(\frac{\partial E}{\partial \omega_{i,j}}(t)\right)\Delta_{i,j}(t)$

$\omega_{i,j}(t+1) = \Delta\omega_{i,j}(t) + \left(\eta\Delta\omega_{i,j}(t)\right) + \alpha\Delta\omega_{i,j}(t-1)$

$\eta = \max(\eta - \eta_d, \eta_m)$
$t = t + 1$
Until Convergence

Fig. 3. A mathematical notation of Rprop backpropagation algorithm

5 Training and Evaluation

5.1 Evaluation Methodology

To measure model performance in biometrics some rates are used to measure performance. Table 2 shows the confusion matrix of any binary classifier.

Then we can express all rates used in the evaluations as given in below.

$$True\ Positive\ Rate\ (TPR) = \frac{TP}{TP + FN} \tag{2}$$

$$True\ Negative\ Rate\ (TNR) = \frac{TN}{TN + FP} \tag{3}$$

$$False\ Positive\ Rate\ (FPR) = \frac{FP}{TP + FN} \tag{4}$$

$$False\ Negative\ Rate\ (FNR) = \frac{FN}{TP + FN} \tag{5}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{6}$$

Table 2. Confusion matrix of binary classifier

		Predicted Class	
		Positive	Negative
Actual class	Positive	True Positives (TP)	False Negatives (FN)
	Negative	False Positives (FP)	True Negatives (TN)

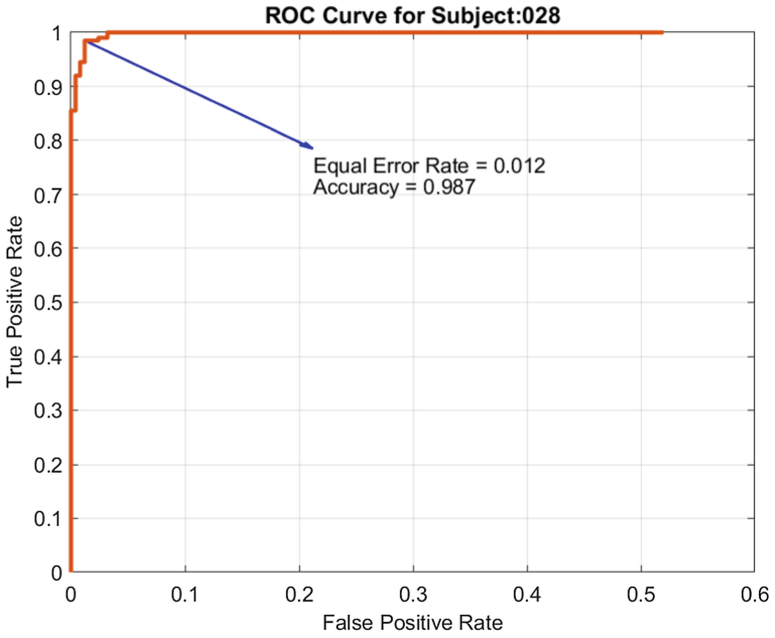

Fig. 4. An example Receiver Operating Characteristics (ROC) curve of a subject which visualizes the performance of the neural network model. The curve demonstrates the trade-off between the true positive rate (hit rate) and the false positive rate (false-alarm rate). The performance can be calculated with proximity to the top-left corner of the graph visually.

False Positive Rate (FPR) also called False Acceptance Rate (FAR) and False Negative Rate (FNR) also called False Rejection Rate (FRR) are used for calculating

Equal Error Rate [29]. A Receiver Operating Characteristics (ROC) curve example in Fig. 4 depicts the performance of the model with FPR and TPR graph.

Equal Error Rate is seen in a clear way in Fig. 5. Basically, ERR is cross point of the FAR and FRR curves when they are plotted to the same graph with similarity threshold in x-axis and error rate in y-axis. Figures 4 and 5 show the performance of model on same subject from different perspectives. Figure 5 also shows accuracy of the model for the current subject. Mean Accuracy and Mean Equal Error Rate represents the mean values of sum of accuracies and sum of equal error rates up to current subject. In this manner, the graph of the last subject will show whole model's average accuracy and average equal error rate in the end.

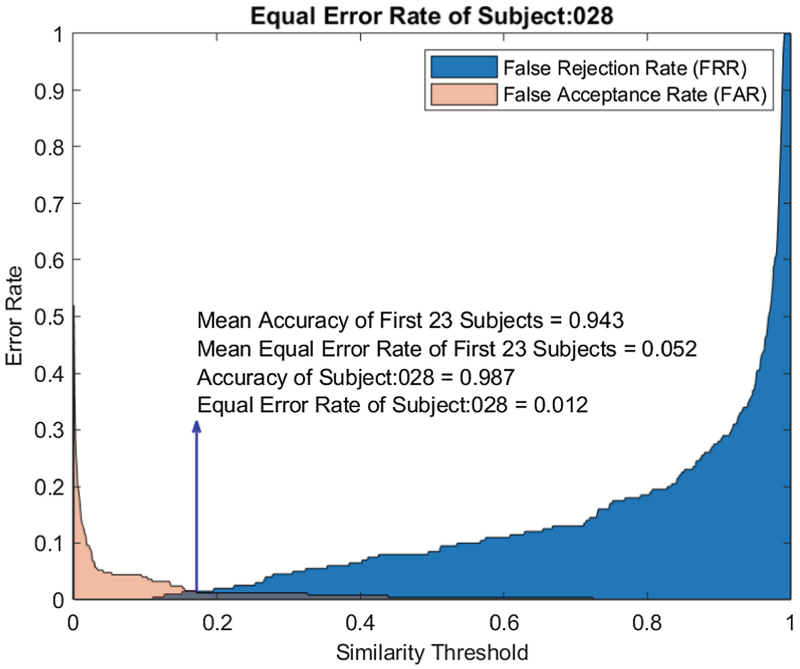

Fig. 5. An example of False Acceptance Rate (FAR) and False Rejection Rate (FRR) versus Similarity Threshold. The equal error rate (EER) is shown as cross point between False Acceptance Rate (FAR) and False Rejection Rate (FRR).

5.2 Training and Stopping Criteria

We trained the three neural network models for each subject for 10 times and plotted ROC Curve graph, Similarity Threshold vs. Error Rate and Mean Squared Error (MSE) vs Epochs graphs. Figure 6 shows MSE vs number of epochs graph of all subjects.

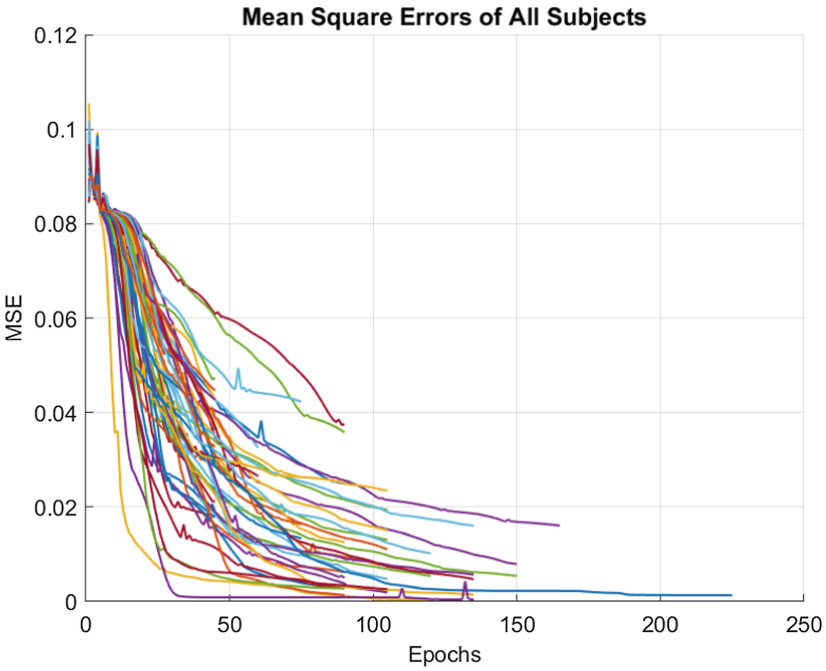

Fig. 6. Mean Square Error (MSE) vs. Epochs graph shows the decline of loss function with respect to increasing epochs

The neural network training convergence is decided with respect to the validation timing features. At each epoch, the cross point between FAR and FRR is found and using the cross points' threshold value accuracy of the model is calculated. For each 15-epochs set, the average of accuracy for the current 15-epochs is compared with that of the previous 15-epochs, and training is resumed if average is increased and stopped if average is decreased. Here, we aimed to prevent neural network model from overfitting.

6 Results

The used evaluation approach was explained in the previous section. There are 51 subjects in CMU Keystroke Dynamics Benchmark Dataset, however, enumeration of them is not in sequential order. This sequencing starts with 2 and ends with 57 having missing sequences in the range [25]. To keep the relation same, we used the same subject enumeration. In Fig. 7, all ROC Curves of these subjects are plotted together to visualize the success of the model easily.

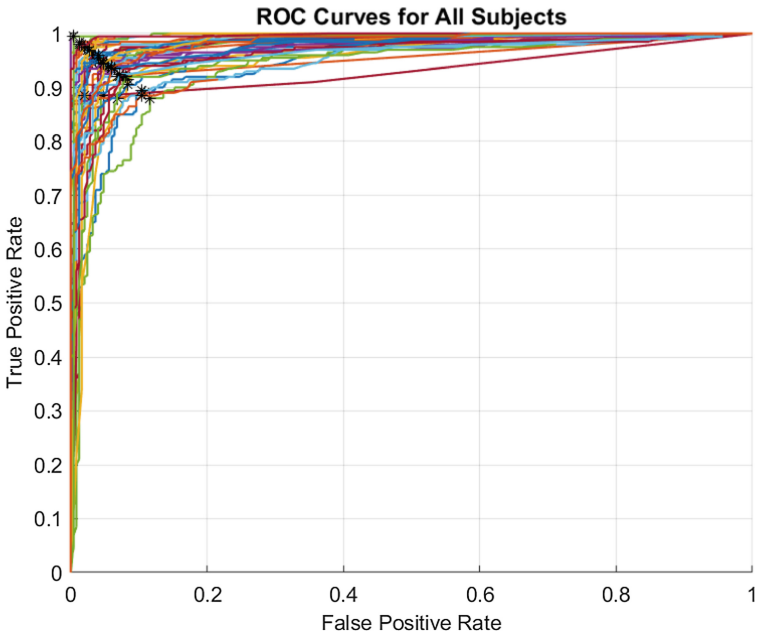

Fig. 7. ROC curves of all users after ending training.

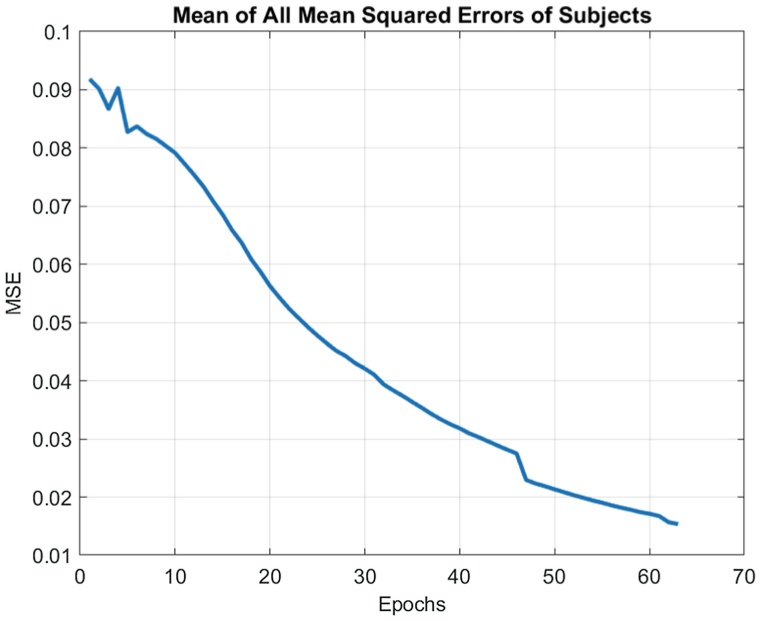

Fig. 8. Average of the MSE levels vs. the epoch number

The star marked points in Fig. 7 denote TPR and FPR on EER. The accuracy of model for a subject can be found by using (6) on EER point [30]. Then average accuracy is found from all these values. Starting from the first subject to the last subject we obtained the average EER and accuracy by incremental averaging method. The results are quite impressive compared to other techniques seen in Table 3, the best results are achieved with 20-30-20 neural network configuration. The neural network having 3 hidden layers produces slightly better results than those having more than 3 hidden layers. 20-30-20 model managed average Equal Error Rate of 0.049 with the average identification accuracy of 94.7%. Also, the average number of epochs needed for the convergence, considering all the models of users, is 70 as seen in Fig. 8.

Table 3 shows comparisons of models which uses CMU Keystroke Dynamics Benchmark Dataset. Our model is positioned in the third place with respect to Average EER, and positioned in the first place with respect to user identification accuracy, which has not been considered in some research reports.

Table 3. Model/Algorithm comparisons

Model/algorithm	Average EER	Average accuracy
Deep Secure	**0.030**	93.59% [33]
Deep Belief Nets (DBN) [31]	0.035	65.60% [33]
Our Model	**0.049**	**94.7%**
Median Vector Proximity [32]	0.080	–
Manhattan-Mahalanobis (No Outlier) [23]	0.084	–
Manhattan-Mahalanobis (Outlier) [23]	0.087	–
Manhattan (scaled) [25]	0.0962	81.20% [33]
Nearest Neighbor (Mahalanobis) [25]	0.0996	–
Outlier Count (z-score) [25]	0.1022	–
SVM (one-class) [25]	0.1025	66.40% [33]
Mahalanobis [25]	0.1101	–
Manhattan (Filter) [25]	0.1360	–
Neural Network (Auto-associated) [25]	0.1614	–
Euclidean [25]	0.1706	–
Fuzzy Logic [25]	0.2213	–
K Means [25]	0.3722	–
Neural Network (Standard) [25]	0.8283	–

7 Conclusions

The studies about the keystroke dynamics and improvements don't claim that keystroke dynamics has the lowest EER, as well as, it is not the most trustworthy mechanism for the user authentication. As a matter of fact, there are retina methods that have much low EERs than keystroke dynamics like 0.01 or even 0 [37]. However, the studies show that keystroke dynamics can rival these methods even with respect to EER although the performance is lower when compared. Above all, there are mainspring motives behind

the preferability of keystroke dynamics. First, it doesn't require any additional expensive cumbersome hardware setup, and maintenance. Also, keystroke dynamics doesn't claim that it is the primary authentication mechanism, rather, by nature of the method, it is a supportive mechanism used with universally accepted username/password authentication. In addition, it is proven that imitating a user using the keystroke dynamics is barely possible. Combining whole above, keystroke dynamics is still an under-research area. On the one hand the advantages of keystroke dynamics make it a plausible option, on the other hand, improvements helps keystroke dynamics compete with other methods.

In this work, we have introduced a successful feedforward neural network scheme with resilient backpropagation based user identification approach. Our model was used for CMU Keystroke Dynamics dataset and the goals were identification and recognition with static text type. The results and the comparisons have shown that the proposed method is able to yield promising values for identifying subjects from keystroke information. We plan extending our models to other long text provided datasets to check if the model is also successful in other datasets.

References

1. Kim, H.-J.: Biometrics, is it a viable proposition for identity authentication and access control? Comput. Secur. **14**, 205–214 (1995)
2. Zhong, Y., Deng, Y.: A survey on keystroke dynamics biometrics: approaches, advances, and evaluations. Gate to Computer Science and Research Recent Advances in User Authentication Using Keystroke Dynamics Biometrics, pp. 1–22 (2015)
3. Dealey, W., Dvorak, A., Merrick, N., Ford, G.: Typewriting behavior (1936)
4. Leggett, J., Williams, G.: Verifying identity via keystroke characteristics. Int. J. Man Mach. Stud. **28**, 67–76 (1988)
5. Goicoechea-Telleria, I., Sanchez-Reillo, R., Liu-Jimenez, J., Blanco-Gonzalo, R.: Attack potential evaluation in desktop and smartphone fingerprint sensors: can they be attacked by anyone? Wirel. Commun. Mob. Comput. **2018**, 1–16 (2018)
6. Ramachandra, R., Busch, C.: Presentation attack detection methods for face recognition systems. ACM Comput. Surv. **50**, 1–37 (2017)
7. Garofalo, G., Rimmer, V., Hamme, T., Preuveneers, D., Joosen, W.: Fishy faces: crafting adversarial images to poison face authentication (2018)
8. Albakri, G., Alghowinem, S.: The effectiveness of depth data in liveness face authentication using 3D sensor cameras. Sensors **19**, 1928 (2019)
9. Zhou, Z., Tang, D., Wang, X., Han, W., Xiangyu, L., Zhang, K.: Invisible mask: practical attacks on face recognition with infrared (2018)
10. Meng, T.C., Gupta, P., Gao, D.: I can be you: questioning the use of keystroke dynamics as biometrics. In: Proceedings of the 20th Network and Distributed System Security Symposium (2013)
11. Epp, C., Lippold, M., Mandryk, R.L.: Identifying emotional states using keystroke dynamics. In: Proceedings of the 2011 Annual Conference on Human Factors in Computing Systems - CHI (2011)
12. Forsen, G., Nelson, M., Staron Jr, R.: Personal attributes authentication techniques. Technical report RADC-TR-77-333, Rome Air Development Center (1977)

13. Spillane, R.: Keyboard apparatus for personal identification. IBM Tech. Disclosure Bull. **17** (3346), 3346 (1975)
14. Gaines, R., Lisowski, W., Press, S., Shapiro, N.: Authentication by keystroke timing: some preliminary results. Technical Report Rand Rep. R-2560-NSF, RAND Corporation (1980)
15. Monrose, F., Rubin, A.D.: Keystroke dynamics as a biometric for authentication. Future Gener. Comput. Syst. **16**, 351–359 (2000)
16. Alsultan, A., Warwick, K.: Keystroke dynamics authentication: a survey of free-text methods. Int. J. Comput. Sci. Issues **10**, 1–10 (2013)
17. Banerjee, S.P., Woodard, D.: Biometric authentication and identification using keystroke dynamics: a survey. J. Pattern Recogn. Res. **7**, 116–139 (2012)
18. Bhatt, S., Santhanam, T.: Keystroke dynamics for biometric authentication—a survey. In: 2013 International Conference on Pattern Recognition, Informatics and Mobile Engineering (2013)
19. Crawford, H.: Keystroke dynamics: characteristics and opportunities. In: 2010 Eighth International Conference on Privacy, Security and Trust (2010)
20. Karnan, M., Akila, M., Krishnaraj, N.: Biometric personal authentication using keystroke dynamics: a review. Appl. Soft Comput. **11**, 1565–1573 (2011)
21. Shanmugapriya, D., Padmavathi, G.: A survey of biometric keystroke dynamics: approaches, security and challenges. Int. J. Comput. Sci. Inform. Secur. **5**, 115–119 (2009)
22. Teh, P.S., Teoh, A.B.J., Yue, S.: A survey of keystroke dynamics biometrics. Sci. World J. **2013**, 1–24 (2013)
23. Zhong, Y., Deng, Y., Jain, A.K.: Keystroke dynamics for user authentication. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (2012)
24. Bleha, S., Slivinsky, C., Hussien, B.: Computer-access security systems using keystroke dynamics. IEEE Trans. Pattern Anal. Mach. Intell. **12**, 1217–1222 (1990)
25. Killourhy, K.S., Maxion, R.A.: Comparing anomaly-detection algorithms for keystroke dynamics. In: 2009 IEEE/IFIP International Conference on Dependable Systems and Networks (2009)
26. Yu, E., Cho, S.: GA-SVM wrapper approach for feature subset selection in keystroke dynamics identity verification. In: Proceedings of the International Joint Conference on Neural Networks (2003)
27. Souza, B., Brito, N., Neves, W., Silva, K., Lima, R., Silva, S.D.: Comparison between backpropagation and RPROP algorithms applied to fault classification in transmission lines. In: 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541) (2004)
28. Prasad, N., Singh, R., Lal, S.P.: Comparison of back propagation and resilient propagation algorithm for spam classification. In: 2013 Fifth International Conference on Computational Intelligence, Modelling and Simulation (2013)
29. Swets, J.A., Pickett, R.M.: Evaluation of Diagnostic Systems: Methods from Signal Detection Theory. Academic Press, New York (1982)
30. Fawcett, T.: An introduction to ROC analysis. Pattern Recogn. Lett. **27**, 861–874 (2006)
31. Deng, Y., Zhong, Y.: Keystroke dynamics user authentication based on gaussian mixture model and deep belief nets. ISRN Signal Process. **2013**, 1–7 (2013)
32. Al-Jarrah, M.: An anomaly detector for keystroke dynamics based on medians vector proximity. J. Emerg. Trends Comput. Inform. Sci. **3**, 988–993 (2012)
33. Maheshwary, S., Ganguly, S., Pudi, V.: Deep secure: a fast and simple neural network based approach for user authentication and identification via keystroke dynamics (2017)

34. Muliono, Y., Ham, H., Darmawan, D.: Keystroke dynamic classification using machine learning for password authorization. Proc. Comput. Sci. **135**, 564–569 (2018)
35. Xiaofeng, L., Shengfei, Z., Shengwei, Y.: Continuous authentication by free-text keystroke based on CNN plus RNN. Proc. Comput. Sci. **147**, 314–318 (2019)
36. Baynath, P., Soyjaudah, K.M.S., Khan, M.H.-M.: Machine learning algorithm on keystroke dynamics pattern. In: 2018 IEEE Conference on Systems, Process and Control (ICSPC) (2018)
37. Chihaoui, T., Jlassi, H., Kachouri, R., Hamrouni, K., Akil, M.: Personal verification system based on retina and SURF descriptors. In: 2016 13th International Multi-Conference on Systems, Signals and Devices (SSD) (2016)