User Classification in Keystroke Dynamics Using Autoencoder And Support Vector Classifier(SVC)

*

1st Dr. Rahul Desai
Information Technology
Army Institute Of Technology
Pune, India
rahuldesai@aitpune.edu.in

2nd Deepanshu Kumar Jha
Computers
Army Institute Of Technology
Pune, India
deepanshujha_16704@aitpune.edu.in

3rd Rekita Supyal
Information Technology
Army Institute Of Technology
Pune, India
rekitasupyal3@gmail.com

4th Kritika Suman
Information Technology
Army Institute Of Technology
Pune, India
kritikasuman_16430@aitpune.edu.in

5th Shweta Dhanajirao Patil
Information Technology
Army Institute Of Technology
Pune, India
kshweta_16486@aitpune.edu.in

***Abstract*—The interest for present day instruments and techniques to limit access to applications and administrations which contain sensitive information is expanding exponentially every year. Customary techniques, for example, passwords ,PINs,or tokens ignore the difficulties exhibited in light of the fact that they can be stolen or lost easily , which risk the framework security.Stolen passwords have the potential to make extreme harm to organizations and people of the same, prompting the prerequisite that the security framework must have the option to recognize and forestall fake login. Biometrics dependent on identifying the individual or how the individual acts, present a critical security progression to meet these new challenges.Biometrics, characterized as the physical attributes and conduct attributes that make every one of us remarkable, are a characteristic decision for personality confirmation. Biometric qualities become the most ideal and perfect contender for verification since they can't be stolen, lost or mimicked . Since biometrics techniques like unique finger impression , iris scanner requires outer and expensive durable goods , Keystroke biometrics is an effective and financial strategy anybody can use to give security dependent on biometrics. Keystroke biometrics is the investigation of the composing conduct so as to distinguish the typist, utilizing highlights extricated during composing. The highlights generally utilized in keystroke biometrics are straight blends of the timestamps of the keystrokes.**

I. Introduction

These days anybody can disguise to be another person just by utilizing the substantial user's userid and password. For this situation the password can't validate its legitimate user. Many electronic authentication frameworks have been proposed to protect business exchanges and to verify data. User ID's and passwords, IP address sifting, message digest authentication, and so forth are the famous ones.But these are inclined to abuse .There exists progressing research into distinguishing the peculiarity of a user by using user's connection with a PC as a type of authentication. The most believing strategy has been Keystroke biometrics which alludes to the constant examples or rhythms an individual shows while composing on a keyboard input gadget. Contrasted with other biometric mappings, keystroke has the essential preferences that:

1.No outside equipment like scanner or identifier is required. All that is needed is a keyboard.

2.The "pattern" of users is a truly dependable measurement.

3.It can undoubtedly be concatenated with the existing authentication frameworks.

4.It can also be used to distinguish between human composing or scripted programs, for example, malware [1].

The keystroke authentication approach has been separated into two. A large portion of the current methodologies center around "static" confirmation, where a user types explicit pre-enlisted string, e.g., a password during a login procedure, and afterward their keystroke features are broke down for authentication reason. The subsequent one is called as "free-content" elements which doesn't have a pre-decided solid. It adjusts to the composing design . For increasingly secure applications, free-content ought to be utilized to constantly verify a user. Timing data for keystroke biometrics are made by recording the time when each key is pressed and the time when it is discharged. From the planning data, one can extract various features, for example, latencies. The features that are normally utilized are hold time, flight time and digraph [2], which are consequently alluded to as the regular features. Hold time is defined as the time elapsed when the key is pressed and it is released. Flight time is slightly different than Hold time as flight time deals with two different key and it is defined as the time gap when a key is released and other key is pressed.

The digraph is characterized as the time gap between press-

ing of two resulting keys. This paper looks at Autoencoder with SVM and PCA as an oddity detection strategy with other different models like ,logistic regression,Random Forest Classifier, Decision Tree Classifier etc.

II. Literature Survey

In the developing market of biometrics, keystroke dynamics biometrics is very much affordable and productive biometric as it does not require any extra hardware like other biometric systems. Comparing the works of various researchers is tough as there is lack of standards for data collection and benchmarking.

Most of previous approaches are based on static verification, in this a user types specific string which is pre-decided, and then their keystroke features are extracted and examined for login purposes [3]. Mainly timing values are used as a feature in keystroke dynamics

Extracted timing features:

1. Key Hold (KD): The time elapsed between the same key pressed and the same key released.

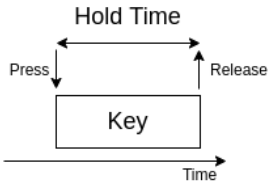

Fig. 1.

2. Down-Down Time: The time in between two consecutive presses of a password.

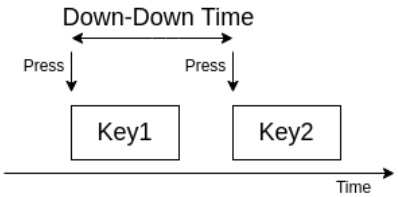

Fig. 2.

3. Up-Up Time: The time between release of 2 successive keys in a password.

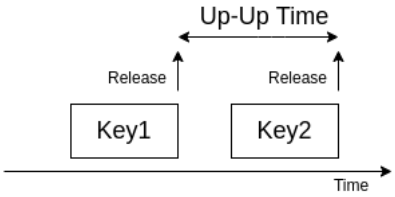

Fig. 3.

4. Up-Down Time: The time gap between the current key released and the next key of the password pressed.

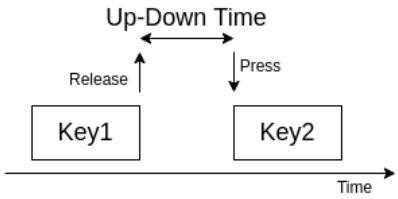

Fig. 4.

5. Down-Up Time:The time gap between the current key pressed and the next key of the password released.

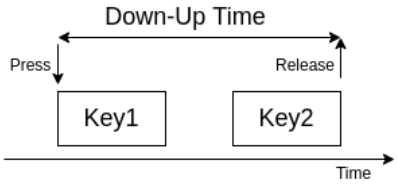

Fig. 5.

Once the features from the password have been extracted and templates for the original user created, classification of users into imposter or original user is known by calculating the similarity between the typist features and original user features and by checking how much of the features resembles the original user characteristics. In order to categorize the typist, simple behavioural patterns are extracted from the data collected and some of the complicated pattern recognition techniques are used. A combination of methods have also been used in some cases.

A. Statistical Algorithms

Some of the least complex statistical strategy comprises of processing the mean of the features and their standard deviations in the template. These would then be able to be utilized for distance estimates, for example, Manhattan distance, Absolute weighted distance, Euclidean distance, etc. There were two researcher named Joyce and Gupta utilizing the absolute distance for verification purpose. Using the absolute distance only the False Acknowledgement Rate(FAR) of 0.25% and False Rejection Rate (FRR) of 16.35% was accomplished. As of late, Guven and Sogukpinar [4] have utilized vector examination to group and categorize clients with a 95% . The keystrokes depends on how the subject(user) uses it so the features are of non-linear nature. If we use linear statistical approach they may not give the up to mark outcome. As we know statistical approach needs training so due to lack of training the linear approach does not yield great outcome.

B. Neural Networks

Neural Network or ANN is very much versatile in nature as they are used as non linear data modeling tool. The ANN is inspired by the working principle of biological neurons. The most important method called backpropagation is used to

allocate optimal loads to the neurons and is used in supervised learning.

Obaidat and Macchiarolo introduced an approach to group between character times utilizing an artificial neural system. During the examination stage, three distinctive neural system models were tried - backpropagation , sum-of-products and half and half sum-of-products. From tests, half breed sum-of-products was found to perform better than other architectures and accomplished an ID pace of 97.8%.

Yong et al. [5] suggested to use weightless neural networks for ordering clients. The scaling of data was done and then they divided it into non - linear and linear intervals. After experimentation they found that the non-linear intervals were generating favourable outcome over the linear intervals. Neural networks can deal with numerous parameters. But,they can be slow during the preparation as well as in the application stage. As Neural Network is like a black box which means we can not know which features are selected or how much of the features are significant. This becomes an issue as results in keystroke verification are needed continuously.

C. Recognizing Pattern and Machine learning based algorithms

Identifying Patterns is the act of utilizing patterns or protests and ordering them into various classes dependent on specific algorithms and similarities in data points [6]. Various machine learning algorithms for example, the nearest neighbor alg and grouping to substantially more mind boggling algorithms for example, data mining , Fishers linear discriminant (FLD),Bayes classifier,support vector machine (SVM) and chart hypothesis.

Yu and Cho [7] improved the performance of keystroke identification by utilizing a 3 stage way . An error rate of 0.81% was accomplished with The SVM novelty detector. A strategy was proposed by Giot et al. [8] to distinguish PC users and rate of identification came out to be 95%. by utilizing a support vector machine (SVM). SVM is a supervised learning calculation which shows assured outcomes for both verification and identification .

One of the greatest advantage of utilizing such algorithms is that they furnish a certainty esteem related with the choice made. Probabilistic learning algorithms can likewise decrease the issue of mistake engendering by disregarding yields with low certainty esteems. Also, unaided learning systems can distinguish patterns in the data naturally.

D. Search Heuristics Hybrid algorithms

Keystroke features selection utilizing a system which is combination of other algorithms , support vector machines (SVM) and stochastic optimization algorithms, for example, Azevedo et al. [9] created Genetic algorithm (GA) and particle swarm optimization (PSO) , these are features selection and features extraction algorithms .

For features selection using developmental algorithms like GA , the SVM classifier gave a minimum error of 5.183% , FAR of 0.434% and FRR of 4.752%. With individual and global acceleration of 1.5 in PSO, the minimum error was 2.213% with a FAR of 0.41% and FRR of 2.071%.

The benefit of utilizing genetic algorithms is that they can without much of a stretch handle huge databases. It additionally gives multiple solutions and can deal with multi-dimensional, non-continuous, non-differential, and even non-parametrical issues.

III. Model Architecture

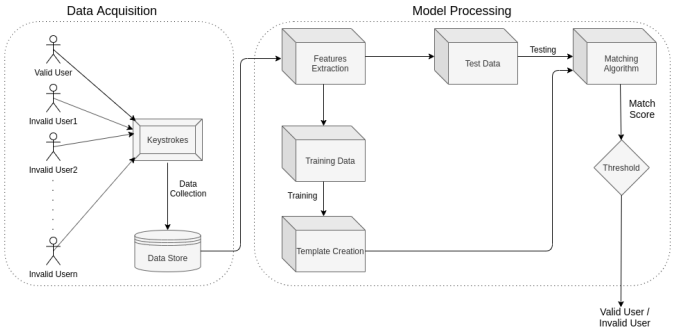

Fig. 6.

A. Data Acquisition

Our dataset generation is unique for each individual user. Each original user have his own dataset, which will contain his password written many times by different users including himself(valid user) as well as other users who are not valid(imposters).This will enable our model to differentiate the typing pattern of original user from others.

Dataset consists of 2 features flight time and hold time.If length of password is n , then dimensions of the dataset will be 2n-1.We can use capital letters and all types of special characters. For our research, the valid password of our valid user is “Reki$1997”.Our dataset therefore consists of 17 features with their timing information.

These 17 features consists of 9 dwell times of password length and 8 flight time between cosecutive pairs of letters in the password. It also consist of 250 rows which have the valid password written by different individuals randomly including valid and invalid users.

B. Model Processing

Auto Encoders and classifiers can be easily implemented so that they are suitable for anomaly detection and for this purpose the model processing surrounding is created. We extract features from the dataset by dividing the data into two major parts known as training and testing in 80 - 20 ratio and a small validation part is taken from the training part for validating model performance. Selection of hyper-parameters for different models was done by several trials and errors and evaluation of their performance on a small development system then coming to a conclusion.

1.Autoencoders : Auto Encoders are used as generative

modeling of data as they compress the data into lower dimension in such a way that they can precisely regenarate the same data using the compressed data only.
Auto Encoders composes of two parts known as Encoder and decoder. The layer which is used to compress the data is called the encoder and the layer that decompresses the data to generate the original input is called decoder. As the model is forced to use the most significant part of original input which is required to regenerate the input precisely by removing the redundant information. A Simple Autoencoder is show in the diagram below :

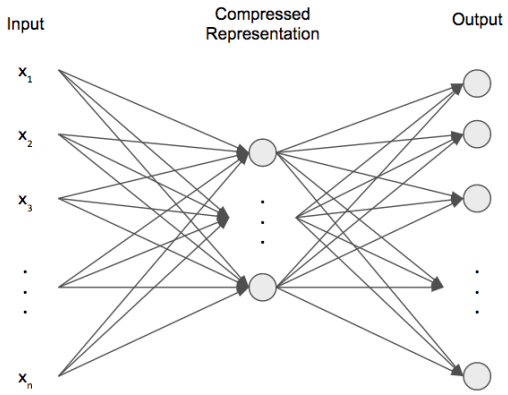

Fig. 7. A Simple Autoencoder

The Auto Encoder has a structure which is very much similar to the Multi Layered Perceptron as MLP also contains one input layer, hidden layers and an output layer.

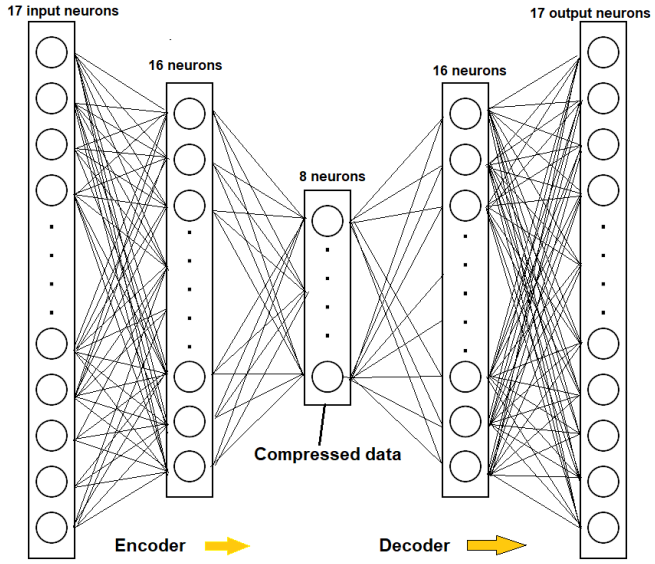

Fig. 8. Our Architecture Of Autoencoder

- Features are extracted from input layer , They are then encoded into a representation of the input , which the Autoencoder learns during training. Classifier is then prepared using the encoded sample, either utilizing just the encoded representation or it is combined with different features, for example, hold time, flight time , digraph, trigraph etc.
- Autoencoders comprises of two encoder layers with 17 input neurons in the first layer and 16 neurons in the second layer . The compressed data consists of 8 neurons .
- This architecture is specific for a password of length 9, Password can be different but its length should be strictly 9.
- For passwords of different lengths , this architecture will be different with different number of layers and different number of neurons in it.
  This implies , with every original user this architecture and dataset will differ.
- Diverse optimizers like Adam, Stochastic Gradient Descent , were tried so as to find an appropriate strategy for training an Autoencoder.
- Adadelta and RMSprop, however we arrived at the conclusion that Adam is best for our architecture . Rectified Linear Unit(ReLU) was used as the activation function and it helped in decreasing the reconstruction error.
- Loss is determined by optimising the mean squared error of the reconstructed input. A few trials were done utilizing the hyperparameters featured in Table 1 and mean values were determined to represent the final result.

Parameters	Values
Activation Function	ReLU
Optimizer	Adam Optimizer
Epochs	150
Learning Rate	0.001
Decay Rate	0.001 / 50
Loss	MSE (Mean Squared Error)

TABLE I
FINAL NETWORK PARAMETERS FOR AUTOENCODER

Tuning Parameters	Values
Kernel	RBF
Gamma	0.05

TABLE II
TUNING PARAMETERS FOR SVC

2.Support Vector Classifier(SVC) : We have tried different tuning parameters and came to the conclusion shown in Table 2 for best result.

C. Results And Discussions

Initially we had 60 data entries in our dataset , so we applied various models by tuning different hyperparameters of the various models and compared there training and testing accuracies.The comparision is shown in Table 3.

After applying such models we increased our dataset upto 250 entries. Several experiments were performed in this dataset.

- In the first try , out of 250 , 1 – 240 entries were given for training wiith number of epochs as 250 and the remaining 241 – 250 were given for testing .This resulted in overfitting of the data and 36% testing accuracy.
- In the next try , the same dataset is given with the same training – testing split but number of epochs was reduced to 150.Here the testing accuracy turn out to be 63%.
- Then , we again reduced the epochs to 100 , here the accuracy was 54%.
- Then we increased the epocs to 200, here the accuracy was 36%. So we found that 150 epochs was optimum for training.
- We used 1 – 230 for training and 231 – 250 for testing with 200 epochs.With gamma as auto the accuracy was 71% and with gamma as 0.05 the accuracy was 34%
- With gamma as 0.1 , the same data and 150 epochs accuracy was 71%.
- Then we do the 80 – 20 split. 1 – 200 for training and 201 – 250 for testing.With 150 epochs the accuracy came out to be 82% with gamma as 0.1 , 74% with gamma as auto and 84% with gamma as 0.05.
- With the same data split and 175 epochs accuracy was 47% with gamma as 0.1 and 74% with gamma as auto.
- On training the abouve data with 100 epochs , accuracy was 68% with gamma as auto and 70% with gamma as 0.005.
-

All of the above variaions are done in the model using Autoencoder,Principal Component Analysis (PCA) , and Support Vector Classifier (SVC).

- Finally, with the 80 – 20 split , and 150 epochs the model with Autoencoder and SVC is evaluated which resulted in 98% training accuracy and 100% testing accuracy With gamma as 0.1.

So we took the final variation and applied it on different models , the testing and training accuracies of various models with the increased data is shown in Table 4.

D. Conclusion

The benefit of Auto Encoders, is the negligible feature engineering that should be performed, is additionally highlighted along with quite less training time. Keystroke dynamics limits the effect on client's experience, requiring less feature engineering implies less time being spent on information processing , this is one of the advantages of keystroke dynamics. Keystroke conduction can also decrease the total time taken to examine the risk of authentication. The keystroke cannot be the only criteria for authentication. The system can be made robust if some bio-metric traits like mouse movements, use of caps lock or shift keys for capital letter, how the typing pattern changes if the user is injured or not well, etc.

Combination of keystroke authentication with One time Password(One time password), Two-Factor Authentication , Or Security Questions will increase the security of the system and will overcome the problem of the original user being injured due to some accident. There might be a possibility to increase the system performance by using different sampling techniques like time varying sampling method e.g. across time data can be filtered. In subsequent work, hyper parameters can be examined more closely , the effect of various hyper parameters on features selection and accuracy can be explored for Auto Encoders. The research can also be extended if additional data is extracted like the mouse pointer pattern and whenever possible touchscreen pattern on touchscreen devices so that the relation between these patterns and keystroke can be investigated.

Models	Training Accuracies	Testing Accuracies
Logistic Regression	86.00 %	50.00 %
Decision Tree Classifier	100.00 %	40.00 %
Random Forest Classifier	100.00 %	60.00 %
Support Vector Classifier(SVC)	94.00 %	40.00 %
Support Vector Classifier(SVC) + Principal Component Analysis(PCA)	84.00 %	70.00 %
Auto Encoder + Support Vector Classifier(SVC)	84.00 %	50.00 %
Auto Encoder + Principal Component Analysis(PCA) + Support Vector Classifier(SVC)	88.00 %	70.00 %

TABLE III
TITLE

Models	Training Accuracies	Testing Accuracies
Logistic Rgression	91.00 %	100.00 %
Decision Tree Classifier	98.50 %	96.07 %
Random Forest Classifier	99.50 %	100.00 %
Support Vector Classifier(SVC)	100.00 %	98.04 %
Support Vector Classifier(SVC) + Principal Component Analysis(PCA)	72.00 %	47.06 %
Auto Encoder + Support Vector Classifier(SVC)	100.00 %	98.04 %
Auto Encoder + Principal Component Analysis(PCA) + Support Vector Classifier(SVC)	91.00 %	100.00 %

TABLE IV
TITLE

REFERENCES

[1] R. Baloch, "An introduction to keyloggers, rats and malware," 2011.
[2] S. P. Banerjee and D. L. Woodard, "Biometric authentication and identification using keystroke dynamics: A survey," *Journal of Pattern Recognition Research*, vol. 7, no. 1, pp. 116–139, 2012.
[3] J. Leggett and G. Williams, "Verifying identity via keystroke characterstics," *International Journal of Man-Machine Studies*, vol. 28, no. 1, pp. 67–76, 1988.
[4] A. Guven and I. Sogukpinar, "Understanding users' keystroke patterns for computer access security," *Computers & Security*, vol. 22, no. 8, pp. 695–706, 2003.
[5] S. Yong, W. K. Lai, and G. Goghill, "Weightless neural networks for typing biometrics authentication," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, pp. 284–293, Springer, 2004.

[6] S. Theodoridis, A. Pikrakis, K. Koutroumbas, and D. Cavouras, *Introduction to pattern recognition: a matlab approach*. Academic Press, 2010.
[7] E. Yu and S. Cho, "Keystroke dynamics identity verification—its problems and practical solutions," *Computers & Security*, vol. 23, no. 5, pp. 428–440, 2004.
[8] R. Giot, M. El-Abed, and C. Rosenberger, "Greyc keystroke: a benchmark for keystroke dynamics biometric systems," in *2009 IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems*, pp. 1–6, IEEE, 2009.
[9] G. L. Azevedo, G. D. Cavalcanti, and E. C. Carvalho Filho, "Hybrid solution for the feature selection in personal identification problems through keystroke dynamics," in *2007 International Joint Conference on Neural Networks*, pp. 1947–1952, IEEE, 2007.