Project Presentation

Group 7

Sumith Varaganti
Sirish Gambhira
Ganni Koushik
Siva Datta B
P. Divyagnan

Outline

The Problem Statement

Introduction

Data Collection

Data pre-processing

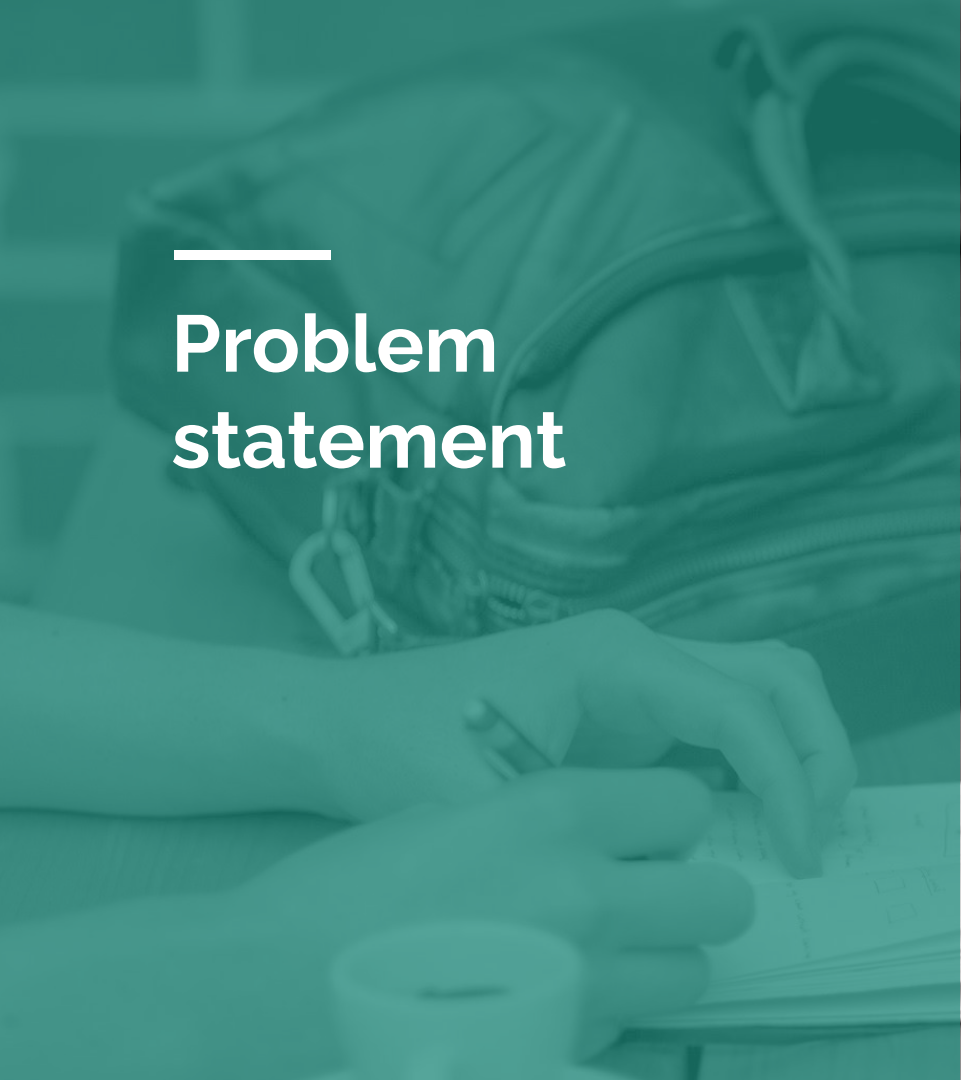
Feature Extraction

Results

Extra Analysis

Discussion

References



Problem statement

Authenticate the user by training and testing keyboard dynamics by the neutral, happy and sad mood data using neural network. Use five-fold validation to report results.

Introduction

The ever-increasing dependency on the computer system has made it an integral part of our daily life. Despite their importance, computer systems today are protected with a primitive security technique of text matching, namely, username and password .

These passwords are vulnerable to various types of attacks, there is a chance of data being stolen or lost in different ways.

In order to avoid the shortcomings of the existing password-based system, biometric-based authentication can be used.

It is a special authentication technique which identifies a person based upon his/her physiological (like face, fingerprint, iris, etc.) or behavioural characteristics (like voice, signature, keystroke, mouse dynamics, etc.)

KeyStroke Dynamics

Keystroke dynamics is a behavioral measurement and it aims to identify users based on the typing of the individuals or attributes such as duration of a keystroke or key hold time, latency of keystrokes (inter-keystroke times)

The keyboard dynamics based authentication is a practical and natural option as the sensor is available by default along with the computer system. So the keystroke dynamics based authentication system can be easily integrated with the existing hardware.

With the use of ML models we can authenticate the user based upon his/her interaction with the keyboard providing an additional security layer.

Data Collection

Mood analysis keyboard database from 10 users is collected for this experiment over a span of 6 weeks.

1. Each user types a random statement and the data is grouped according to his mood response.
2. We combined happy, sad, neutral data from each user.

One of the major problems is that keystroke dynamics runs into is that a person's typing varies substantially during a day and between different days, and may be affected by any number of external factors. To try to decrease this error the data is collected over a long period of time and different times of the day.

Sample Data

KeyDown	e	19:10:20:17:38:37:824
KeyUp	e	19:10:20:17:38:37:906
KeyDown	s	19:10:20:17:38:38:035
KeyUp	s	19:10:20:17:38:38:182
KeyDown	k	19:10:20:17:38:38:330
KeyUp	k	19:10:20:17:38:38:417
KeyDown	i	19:10:20:17:38:38:506
KeyDown	m	19:10:20:17:38:38:599
KeyUp	i	19:10:20:17:38:38:617
KeyUp	m	19:10:20:17:38:38:727
KeyDown	o	19:10:20:17:38:38:762
KeyDown	e	19:10:20:17:38:38:876
KeyUp	o	19:10:20:17:38:38:897
KeyUp	e	19:10:20:17:38:39:002
KeyDown	s	19:10:20:17:38:39:097
KeyDown		19:10:20:17:38:39:207
KeyUp	s	19:10:20:17:38:39:234
KeyUp		19:10:20:17:38:39:330
KeyDown	h	19:10:20:17:38:39:713
KeyUp	h	19:10:20:17:38:39:792
KeyDown	a	19:10:20:17:38:39:911
KeyUp	a	19:10:20:17:38:40:030
KeyDown	v	19:10:20:17:38:40:180
KeyDown	e	19:10:20:17:38:40:261
KeyUp	v	19:10:20:17:38:40:298
KeyUp	e	19:10:20:17:38:40:450
KeyDown		19:10:20:17:38:40:473
KeyUp		19:10:20:17:38:40:540
KeyDown	h	19:10:20:17:38:40:770
KeyDown	u	19:10:20:17:38:40:831
KeyUp	h	19:10:20:17:38:40:865
KeyUp	u	19:10:20:17:38:40:965
KeyDown	n	19:10:20:17:38:40:986
KeyDown	d	19:10:20:17:38:41:079

Data Pre-Processing

While typing a string, two features, namely, keystroke latency (time interval between two consecutive keystrokes) and keystroke hold time (the time for which particular key is pressed) are captured.

Only alphabets are considered to capture the above features.

25 files are taken from each user. So a total of 250 files are used for training and testing.

We used 20 files of each user for training and 5 files for testing.

Feature Extraction

We followed two approaches in order to extract features from our data. In approach 1, we noticed our input vector is sparse, hence we also used approach 2 during our analysis.

Approach 1:

Feature vector consists of average hold time of each alphabet and average latency time for all alphabet pairs . So the size of input feature vector is 26+26*26 = 702

Approach 2:

Feature vector consists of average hold time of all alphabet and average latency time for most 20 frequent alphabet pairs . So the size of input feature vector is 26 + 20 = 46

The most common two-character sequences are:

Sequence	Frequency (per 10,000 chars)
th	330
he	302
an	181
in	179
er	169
nd	146
re	133
ed	126
es	115
ou	115
to	115
ha	114
en	111
ea	110
st	109
nt	106
on	106
at	104
hi	97
as	95
it	93
ng	92
is	86
or	84
et	83
of	80
ti	76

Sequence	Frequency (per 10,000 chars)
ar	75
te	75
se	74
me	68
sa	67
ne	66
wa	66
ve	65
le	64
no	60
ta	59
al	57
de	57
ot	57
so	57
dt	56
ll	56
tt	56
el	55
ro	55
ad	52
di	50
ew	50
ra	50
ri	50
sh	50

Frequency of Character Pairs in English Language per 10,000 characters of text according to a research conducted by Department of Mathematics, Statistics and Computer Science from the University of Illinois, Chicago.

Training

After collecting the features training is done using a 3-layer neural network. In this method, a simple feedforward network is used. We used pytorch library to implement our Artificial Neural Network.

Approach 1:

The size of the input nodes is the same as the dimensionality of the input feature vector 702, and the number of hidden layer neurons 100, followed by a hidden layer with 40 nodes and 2 output nodes.

We observed that Adam optimizer was producing better results. Cross Entropy function is used to calculate the loss.

Approach 2:

The size of the input nodes is 46, and the number of hidden layer neurons 100, followed by a hidden layer with 40 nodes and 2 output nodes.

Results

Cross Validation:

We divided our data into 5 groups. During each fold, we trained on 4 groups and tested on the last group. We analysed the mean and standard deviations of training and validation accuracy among these folds.

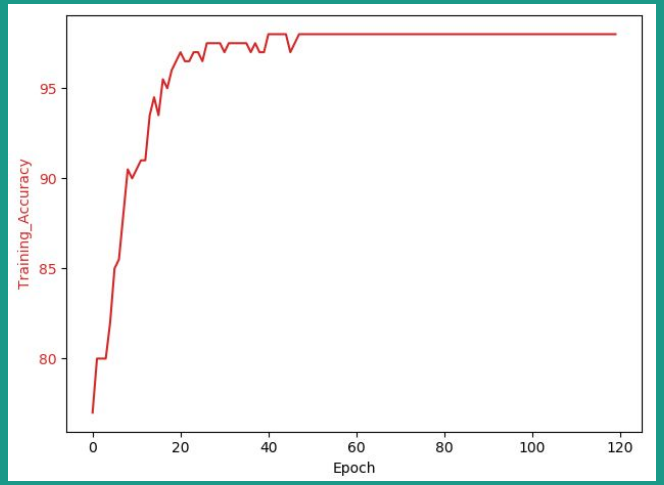
Further Analysis

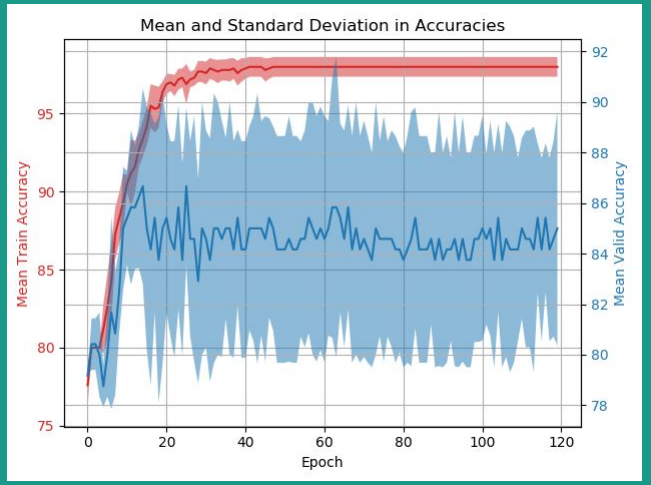
Further, we want to observe the effect of authentic and imposter users on the accuracy of our neural network. Hence, we randomly changed our authentic and imposters during each iteration and observed the training and validation accuracies for 5 iterations.

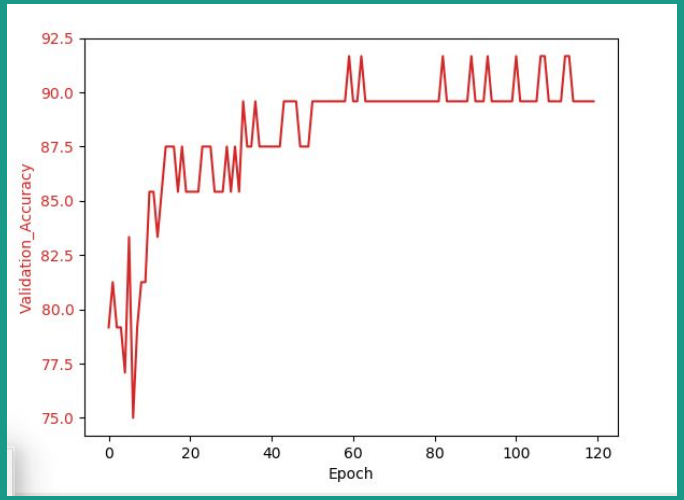
We also the effect of input feature length during these results.

5 Fold

1	Iteration: 0. Train Accuracy: 78.5000. Validation Accuracy: 79.1667 Iteration: 20. Train Accuracy: 98.0000. Validation Accuracy: 81.2500 Iteration: 40. Train Accuracy: 99.0000. Validation Accuracy: 77.0833 Iteration: 60. Train Accuracy: 99.0000. Validation Accuracy: 77.0833 Iteration: 80. Train Accuracy: 99.0000. Validation Accuracy: 77.0833 Iteration: 100. Train Accuracy: 99.0000. Validation Accuracy: 79.1667	1
2	Iteration: 0. Train Accuracy: 75.0000. Validation Accuracy: 79.1667 Iteration: 20. Train Accuracy: 96.0000. Validation Accuracy: 89.5833 Iteration: 40. Train Accuracy: 98.0000. Validation Accuracy: 87.5000 Iteration: 60. Train Accuracy: 98.0000. Validation Accuracy: 89.5833 Iteration: 80. Train Accuracy: 98.0000. Validation Accuracy: 85.4167 Iteration: 100. Train Accuracy: 98.0000. Validation Accuracy: 87.5000	2
3	Iteration: 0. Train Accuracy: 79.5000. Validation Accuracy: 79.1667 Iteration: 20. Train Accuracy: 97.0000. Validation Accuracy: 81.2500 Iteration: 40. Train Accuracy: 97.0000. Validation Accuracy: 81.2500 Iteration: 60. Train Accuracy: 97.0000. Validation Accuracy: 81.2500 Iteration: 80. Train Accuracy: 97.0000. Validation Accuracy: 81.2500 Iteration: 100. Train Accuracy: 97.0000. Validation Accuracy: 81.2500	3
4	Iteration: 0. Train Accuracy: 77.0000. Validation Accuracy: 79.1667 Iteration: 20. Train Accuracy: 97.0000. Validation Accuracy: 85.4167 Iteration: 40. Train Accuracy: 98.0000. Validation Accuracy: 87.5000 Iteration: 60. Train Accuracy: 98.0000. Validation Accuracy: 89.5833 Iteration: 80. Train Accuracy: 98.0000. Validation Accuracy: 89.5833 Iteration: 100. Train Accuracy: 98.0000. Validation Accuracy: 91.6667	4
5	Iteration: 0. Train Accuracy: 78.0000. Validation Accuracy: 79.1667 Iteration: 20. Train Accuracy: 96.5000. Validation Accuracy: 89.5833 Iteration: 40. Train Accuracy: 97.5000. Validation Accuracy: 87.5000 Iteration: 60. Train Accuracy: 98.0000. Validation Accuracy: 85.4167 Iteration: 80. Train Accuracy: 98.0000. Validation Accuracy: 85.4167 Iteration: 100. Train Accuracy: 98.0000. Validation Accuracy: 85.4167	5







The shaded region corresponds to standard deviation.

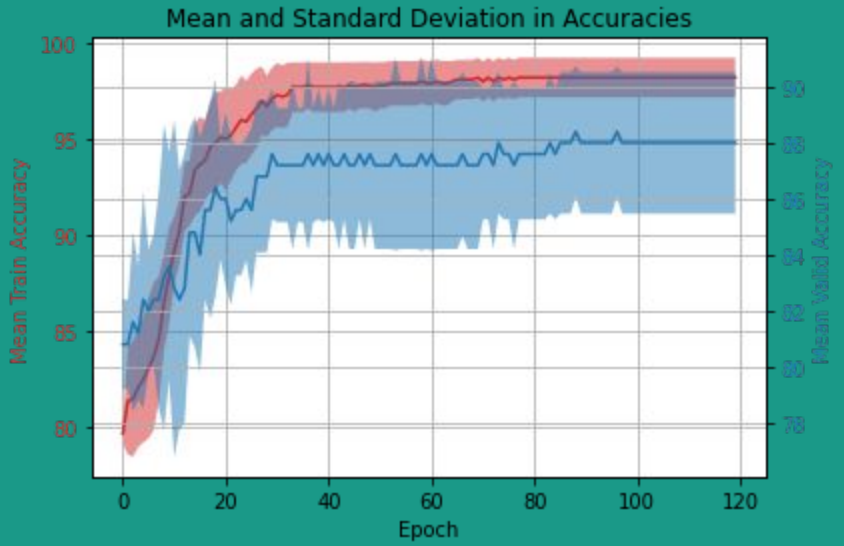
The mean training accuracy achieved is around 92 % whereas the mean validation accuracy achieved is around 85 %. The mean validation accuracy is less than the mean training accuracy as expected. The above plot shows that there is a lot of deviation in the validation accuracy and less deviation in the training accuracy.

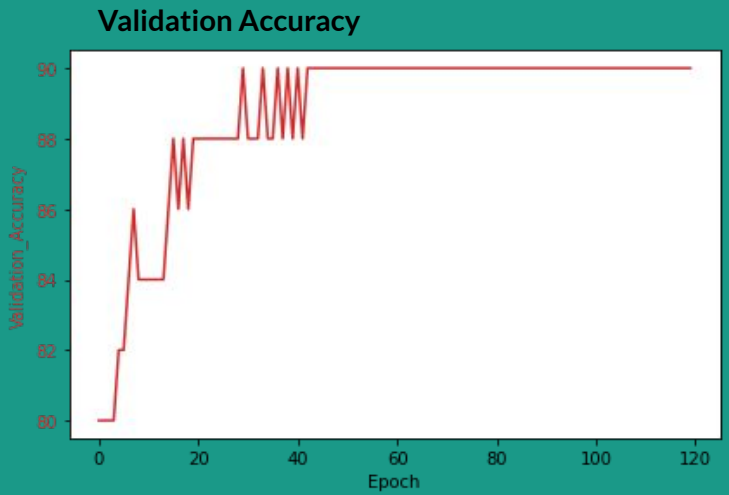
Further Analysis

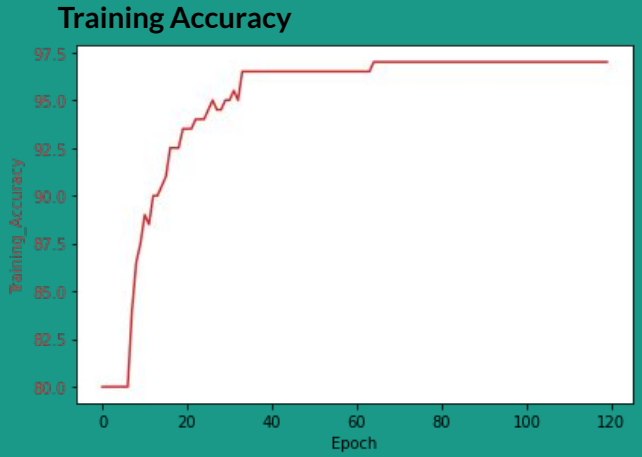
Results

Approach 1

1	Iteration: 0. Train Accuracy: 79.0000. Validation Accuracy: 80.0000 Iteration: 20. Train Accuracy: 96.0000. Validation Accuracy: 88.0000 Iteration: 40. Train Accuracy: 97.5000. Validation Accuracy: 90.0000 Iteration: 60. Train Accuracy: 97.5000. Validation Accuracy: 90.0000 Iteration: 80. Train Accuracy: 97.5000. Validation Accuracy: 90.0000 Iteration: 100. Train Accuracy: 97.5000. Validation Accuracy: 90.0000	Authentic Users: [3 6] Imposters: [0 1 2 4 5 7 8 9]
2	Iteration: 0. Train Accuracy: 79.5000. Validation Accuracy: 80.0000 Iteration: 20. Train Accuracy: 94.5000. Validation Accuracy: 88.0000 Iteration: 40. Train Accuracy: 98.0000. Validation Accuracy: 88.0000 Iteration: 60. Train Accuracy: 98.0000. Validation Accuracy: 88.0000 Iteration: 80. Train Accuracy: 98.0000. Validation Accuracy: 88.0000 Iteration: 100. Train Accuracy: 98.0000. Validation Accuracy: 90.0000	Authentic Users: [1 2] Imposters: [0 3 4 5 6 7 8 9]
3	Iteration: 0. Train Accuracy: 79.5000. Validation Accuracy: 80.0000 Iteration: 20. Train Accuracy: 99.5000. Validation Accuracy: 86.0000 Iteration: 40. Train Accuracy: 100.0000. Validation Accuracy: 86.0000 Iteration: 60. Train Accuracy: 100.0000. Validation Accuracy: 86.0000 Iteration: 80. Train Accuracy: 100.0000. Validation Accuracy: 86.0000 Iteration: 100. Train Accuracy: 100.0000. Validation Accuracy: 86.0000	Authentic Users: [4 7] Imposters: [0 1 2 3 5 6 8 9]
4	Iteration: 0. Train Accuracy: 80.5000. Validation Accuracy: 84.0000 Iteration: 20. Train Accuracy: 91.5000. Validation Accuracy: 88.0000 Iteration: 40. Train Accuracy: 96.5000. Validation Accuracy: 90.0000 Iteration: 60. Train Accuracy: 97.5000. Validation Accuracy: 92.0000 Iteration: 80. Train Accuracy: 98.5000. Validation Accuracy: 90.0000 Iteration: 100. Train Accuracy: 98.5000. Validation Accuracy: 90.0000	Authentic Users: [5 9] Imposters: [0 1 2 3 4 6 7 8]
5	Iteration: 0. Train Accuracy: 80.0000. Validation Accuracy: 80.0000 Iteration: 20. Train Accuracy: 93.5000. Validation Accuracy: 80.0000 Iteration: 40. Train Accuracy: 96.5000. Validation Accuracy: 84.0000 Iteration: 60. Train Accuracy: 96.5000. Validation Accuracy: 82.0000 Iteration: 80. Train Accuracy: 97.0000. Validation Accuracy: 84.0000	Authentic Users: [0 8] Imposters: [1 2 3 4 5 6 7 9]





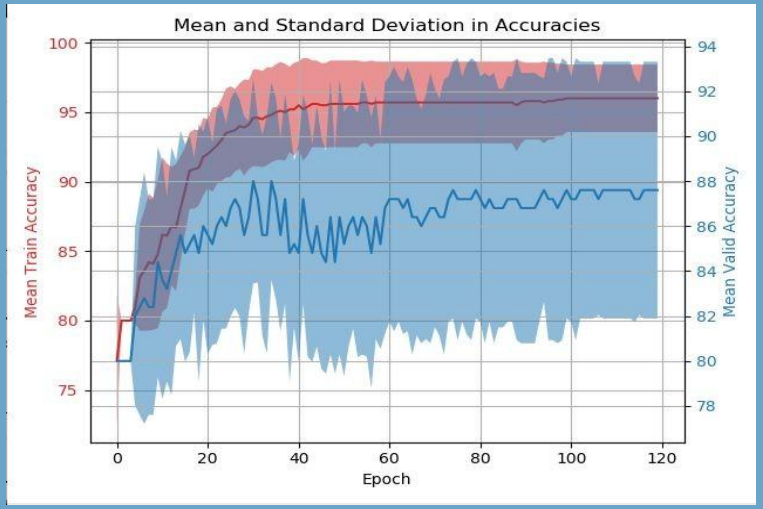


The mean training accuracy achieved is around 98 % whereas the mean validation accuracy achieved is around 88 %. The mean validation accuracy is less than the mean training accuracy as expected. The above plot shows that there is a lot of deviation in the validation accuracy and less deviation in the training accuracy.

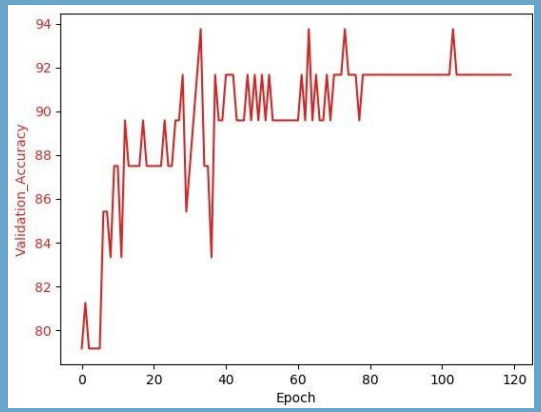
Approach 2

Results

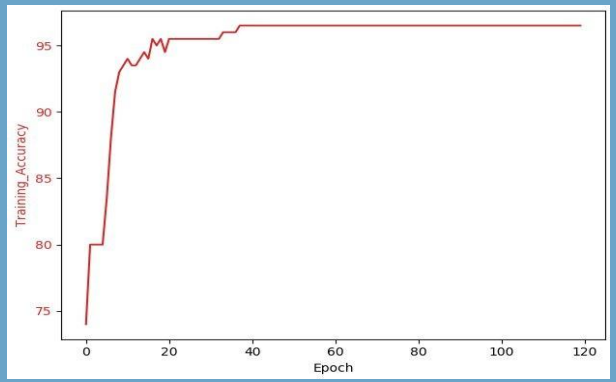
1	Iteration: 0. Train Accuracy: 80.0000. Validation Accuracy: 80.0000 Iteration: 20. Train Accuracy: 94.5000. Validation Accuracy: 82.0000 Iteration: 40. Train Accuracy: 98.5000. Validation Accuracy: 84.0000 Iteration: 60. Train Accuracy: 98.5000. Validation Accuracy: 90.0000 Iteration: 80. Train Accuracy: 98.5000. Validation Accuracy: 90.0000 Iteration: 100. Train Accuracy: 98.5000. Validation Accuracy: 90.0000	Authentic Users : [5 7] Imposters : [0 1 2 3 4 6 8 9]
2	Iteration: 0. Train Accuracy: 78.5000. Validation Accuracy: 80.0000 Iteration: 20. Train Accuracy: 92.5000. Validation Accuracy: 96.0000 Iteration: 40. Train Accuracy: 94.0000. Validation Accuracy: 94.0000 Iteration: 60. Train Accuracy: 94.5000. Validation Accuracy: 96.0000 Iteration: 80. Train Accuracy: 94.5000. Validation Accuracy: 96.0000 Iteration: 100. Train Accuracy: 94.5000. Validation Accuracy: 96.0000	Authentic Users : [8 9] Imposters : [0 1 2 3 4 5 6 7]
3	Iteration: 0. Train Accuracy: 79.5000. Validation Accuracy: 80.0000 Iteration: 20. Train Accuracy: 91.5000. Validation Accuracy: 82.0000 Iteration: 40. Train Accuracy: 98.0000. Validation Accuracy: 82.0000 Iteration: 60. Train Accuracy: 98.0000. Validation Accuracy: 84.0000 Iteration: 80. Train Accuracy: 98.0000. Validation Accuracy: 84.0000 Iteration: 100. Train Accuracy: 98.0000. Validation Accuracy: 84.0000	Authentic Users : [0 6] Imposters : [1 2 3 4 5 7 8 9]
4	Iteration: 0. Train Accuracy: 80.0000. Validation Accuracy: 80.0000 Iteration: 20. Train Accuracy: 94.0000. Validation Accuracy: 84.0000 Iteration: 40. Train Accuracy: 97.0000. Validation Accuracy: 84.0000 Iteration: 60. Train Accuracy: 97.0000. Validation Accuracy: 86.0000 Iteration: 80. Train Accuracy: 97.0000. Validation Accuracy: 86.0000 Iteration: 100. Train Accuracy: 97.0000. Validation Accuracy: 86.0000	Authentic Users : [3 4] Imposters : [0 1 2 5 6 7 8 9]
5	Iteration: 0. Train Accuracy: 68.0000. Validation Accuracy: 80.0000 Iteration: 20. Train Accuracy: 87.5000. Validation Accuracy: 84.0000 Iteration: 40. Train Accuracy: 90.0000. Validation Accuracy: 80.0000 Iteration: 60. Train Accuracy: 90.5000. Validation Accuracy: 80.0000 Iteration: 80. Train Accuracy: 90.5000. Validation Accuracy: 80.0000 Iteration: 100. Train Accuracy: 92.0000. Validation Accuracy: 80.0000	Authentic Users : [1 2] Imposters : [0 3 4 5 6 7 8 9]



Validation Accuracy



Training Accuracy



The mean training accuracy achieved is around 97 % whereas the mean validation accuracy achieved is around 92 %. The mean validation accuracy is less than the mean training accuracy as expected. The above plot shows that there is a lot of deviation in the validation accuracy and less deviation in the training accuracy.

Discussion

- For the subjects with insufficient data, we generated new data by using sliding window method, where we took a sliding window and generated a new sample by taking the average of points in the window.

- We observed that using 702 features produced results with less fluctuations when compared to results with 46 features. The possible explanation for behavior is that the model is able to capture user dynamics better with increased parameters.

- The mean training and validation accuracies for both the cases is the same.
- The spikes in validation accuracies can be explained due to:
    - The test-set is small in size only 5 examples
    - Outliers in the dataset - We performed moving average to generate new features for subjects with insufficient data, this can be a possible reason

..Contd

In the reference paper, the neural network consisted of (2d + 1) nodes in the hidden layer, where d is the number of input features. When the input vector is of size 702, we thought not to increase the parameters further and hence took 100 nodes in the hidden layer. For the second case, when input is of size 46, we didn't modify hidden layer structure to observe the differences in both cases.

We also observe overfitting in our method, which can be prevented by implementing early stopping criteria. Hence, we wrote another code, where we implemented early stopping criteria.

Questions?

References

1. Rajat Kumar Das, Sudipta Mukhopadhyay & Puranjoy Bhattacharya (2014)User Authentication Based on Keystroke Dynamics, IETE Journal of Research, 60:3, 229-239, DOI: 10.1080/03772063.2014.914686
2. "Frequency of Character Pairs in English Language Text." http://homepages.math.uic.edu/~leon/mcs425-s08/handouts/char_freq2.pdf. Accessed 6 Nov. 2020.