KeyStroke Biometrics with RNN(LSTM)

Joel Poah Zhi Kang  
Diploma in Applied Artificial Intelligence and Analytics  
School of Computing   
Singapore Polytechnic

Joel.21@ichat.sp.edu.sg

*Abstract – Cybercrimes have been on the rise in the recent decade people can easily hack into people’s account after gaining access to the user’s password. Black Hat hackers are using various methods to retrieve the password of an account. More sophisticated authentication should therefore be used, one authentication method that authentications based on behavioral patterns instead of a string of characters that everyone can mimic. The objective of the paper is to leverage on LSTM to authenticate a user based on user’s keystroke behavior.*

Keywords—Deep learning, Artificial Intelligence, Behavioral biometrics, Long Short Term Memory (LSTM) (key words)

# Introduction

cybercrime — which is predicted to inflict damages totalling $6 trillion USD globally in 2021 — would be the world’s third-largest economy after the U.S. and China.by Calif, S [1] Trillions of dollars are being lost to cyber-crimes yearly which strongly suggest the need for a robust security system. In recent years, behavioural authentication caught the eyes of security experts as they are harder for hackers to bypass. In this article, we will discuss the basics of keystroke dynamics and how it can be used to authenticate users.

Keystroke biometric is a technique to verify an individual's identity by analysing his or her typing patterns. This technology has several advantages over the current security features that are implemented such as password access, One Time Passcode (OTP). Firstly, Keystroke biometrics are easy to implement for the user since they do not even need to remember any extra information and interact with the keyboard as they normally would. Secondly, individual users can be constantly authenticated instead of a one-time password which the hackers can retrieve once and access the system for a certain period.

A keystroke biometric dataset consists of a set of input sequences (keystrokes) that are associated with a specific individual and a corresponding set of output sequences (texts). This data can be collected either manually or through an automated system by either capturing the user's actions or installing a keyboard capture driver on the device. Once the dataset is collected, it can be processed using machine learning algorithms to identify suitable sequences for use in keystroke authentication.

# Related Works

The idea of keystroke biometrics have been around for decades and several engineers like Kevin S. Killourhy and Roy A. Maxion [2] have collected and experimented various machine learning models on the data. Certain models or algorithms include Support Vector Machines (SVM) , KMeans Clustering and Dense Neural networks. However, LSTM was not tried to tackle this problem and since it is a time series problem which also requires memory, we will look at how LSTM can help with Keystroke biometrics.

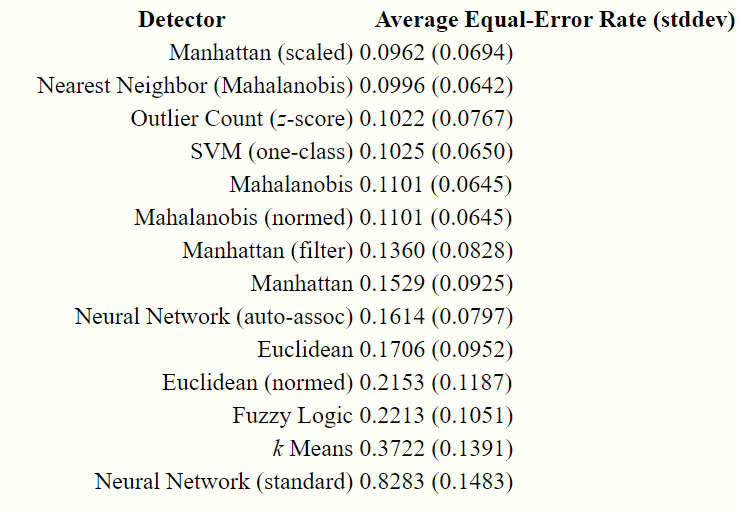


Fig. 1. List of Machine learning Equal Error Rate (EER)

As mentioned by C.,Eric and F.,Joshua EER also known as Crossover Error Rate(CER) is when the False Accept Rate(FAR) and the False Reject Rate (FRR) meet when the sensitivity is lowered .[3]

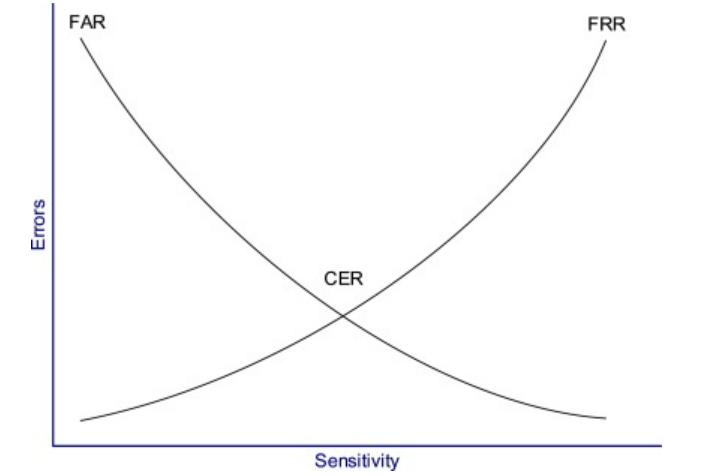


Fig. 2. Graph of CER where FAR meets FRR [3]

FAR is wrongly accepting a person and FRR is wrongly rejecting the legitimate user. This gave a deeper insight into how I can evaluate the model later.



# Methodology

In this paper, the following Python libraries will be used:

**Pandas** - Built on top of NumPy, it is a useful data science tool that can perform data wrangling, display numerical summaries and other data cleaning tasks [3].

**Seaborn** - A visualization library that is built on Matplotlib. It can generate many types of statistical graphs and works well with Pandas data structures [4].

**sklearn** - An extremely useful Python library for Machine Learning. This library contains various Machine Learning algorithms and statistical modeling tools [5].

**TensorFlow** – A Deep learning framework which is often seen as the most popular deep learning framework, which is very powerful & easy-to-use and has excellent community support.[6]

## Data

Kevin S. Killourhy and Roy A. Maxion has kindly shared the data they have collected in a controlled environment back in 2007 where 51 typists are presented with a password to type and if there were errors, the software will prompt the typist to retype until 50 correctly typed password in each session. There was a total of 8 sessions which is 400 data points per typist. The password typed is (.tie5Roanl).

## Exploratory Data Analysis

Exploring the data is important to draw useful insights that can help improve the model or better feature engineer it. I first used pandas to import the csv file that was downloaded from the paper published by Kevin S. Killourhy and Roy A. Maxion. The time metric for the keystroke attributes are in seconds.

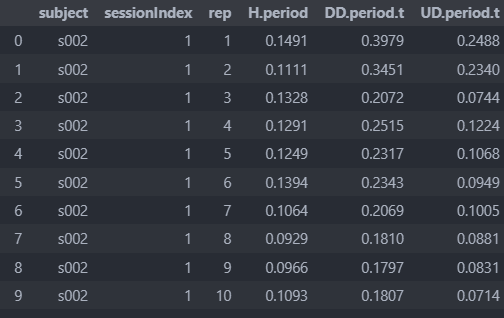


Fig. 3 Head of Pandas Dataframe of Keystroke Dynamics Dataset

There were missing subjects as not all the subjects who registered for the experiment but did turn up on that day. Subject column is in varchar initially. Each typist typed the password 400 times taking a break between 8 sessions.

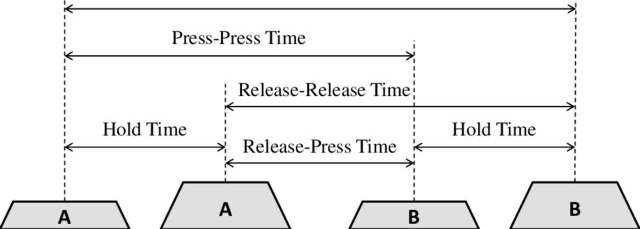


Fig. 4 Representation of Keystroke features/columns [6]

H.{key} represents the duration of which the {key} was held onto.

DD.{key1}.{key2} is the duration between pressing {key1} and then pressing {key2}

UD.{key1}.{key2} is the duration between releasing {key1} and then pressing {key2}.note that UD.{key1}{key2} can be negative if the typist do not release {key1} before hitting {key2}

There is also Release-Release or Press-Press time, but it probably will be that important to process that column since there is it can feature engineered easily by adding Release-Press time with Hold Time

### Data visualisation

There were so many columns when coming to decide which key to showcase the distribution of a typist, but I have decided that it is most appropriate and accurate to analyze the middle character of the password that was typed which is the ‘R’ since it probably can distinguish the typists the most because it is capitalized and also affected by the way the first few characters were typed. I will be analysing the columns ‘DD.five.Shift.r’ , ‘UD.five.Shift.r’ and ‘H.Shift.r’ out of the columns shown below.

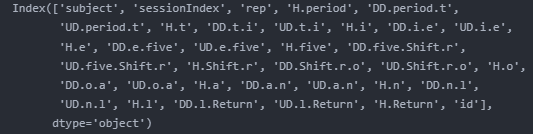


Fig. 6. All columns found in dataset

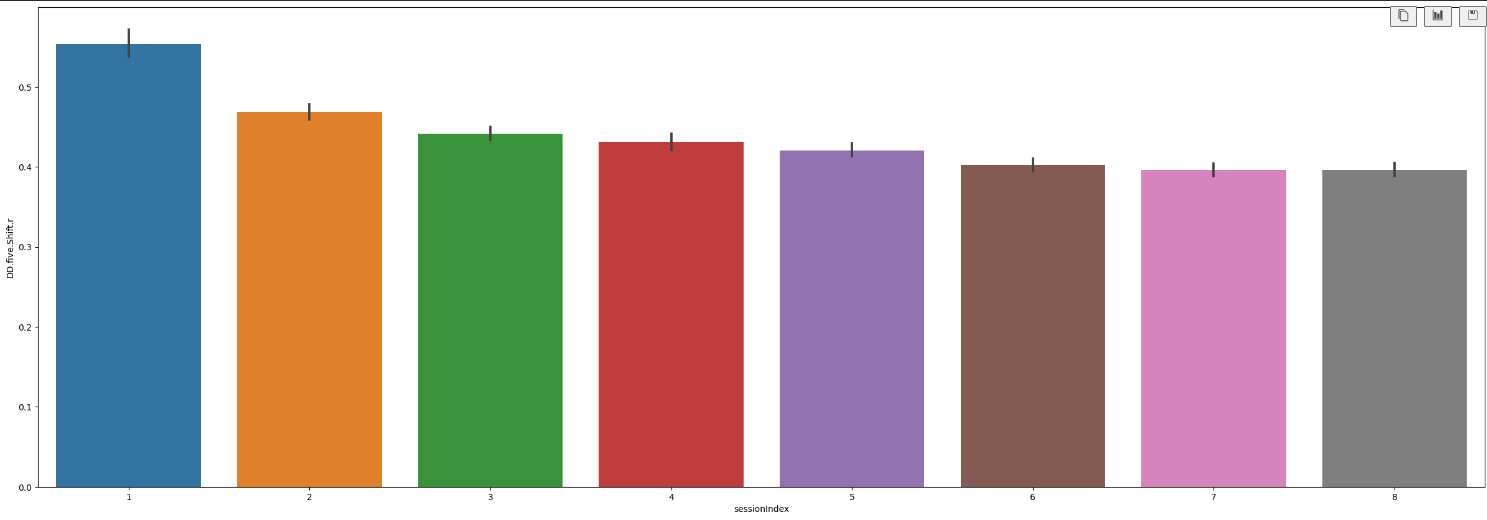


Fig. 7. Bar plot of average time taken for key down 5 to key down R

We see that as the session increases the typists got faster in typing the password which probably shows that generally typists can learn to type faster every day. However, I also realized that by the 6th session the average typists speed stops fluctuating which is good for the model to learn about the way the person types.

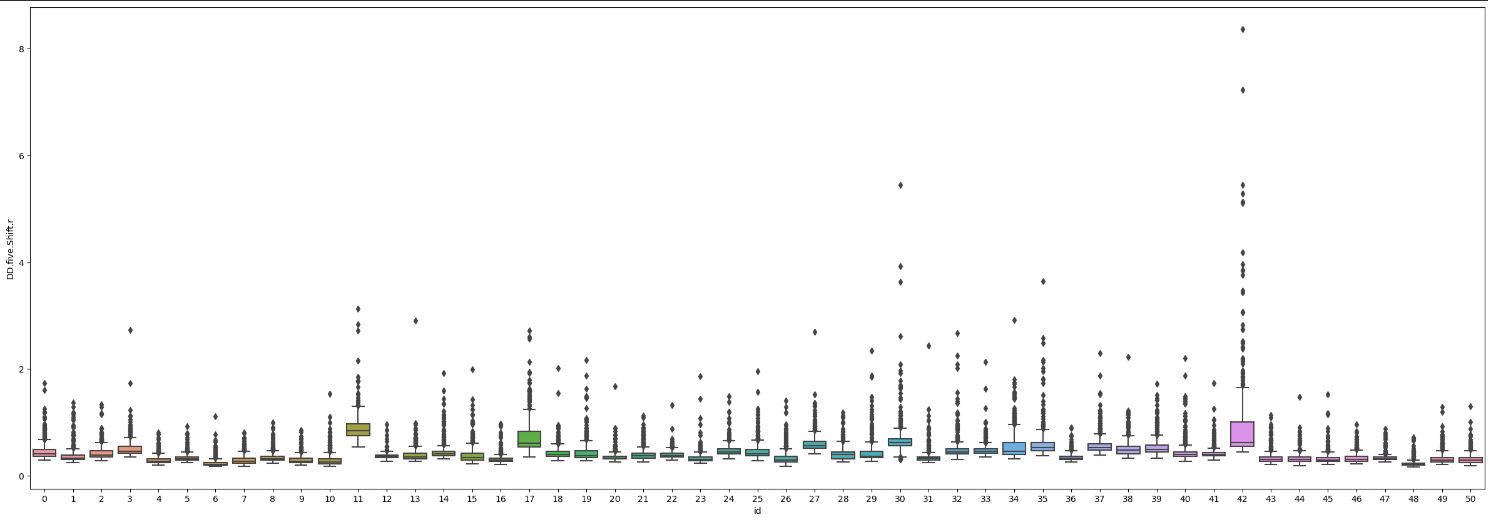


Fig. 8. Boxplot of typing ‘DD. five.Shift.r’

It makes perfect sense that typists get faster as they type the same password multiple times. However, if the spread of the typing style is too wide, it may be hard to authenticate the real user if there are large swings in the way the typist types.

### Simple Clustering Analysis

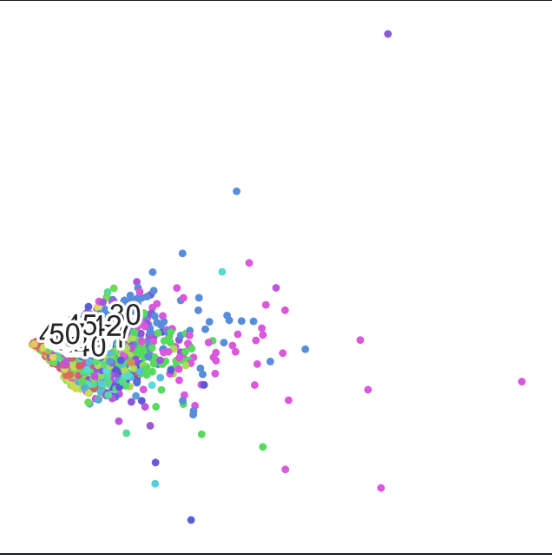


Fig. 9. Clustering with PCA

PCA is the fundamental building block of multivariate data analysis. (Sartorius ,2020) It helps by summarizing the important part of the data and expressing that information as a set of summary called principal components. This method is also called dimensionality reduction. However just by looking at the columns with the highest variance does not help cluster well. I then proceeded to using TSNE.

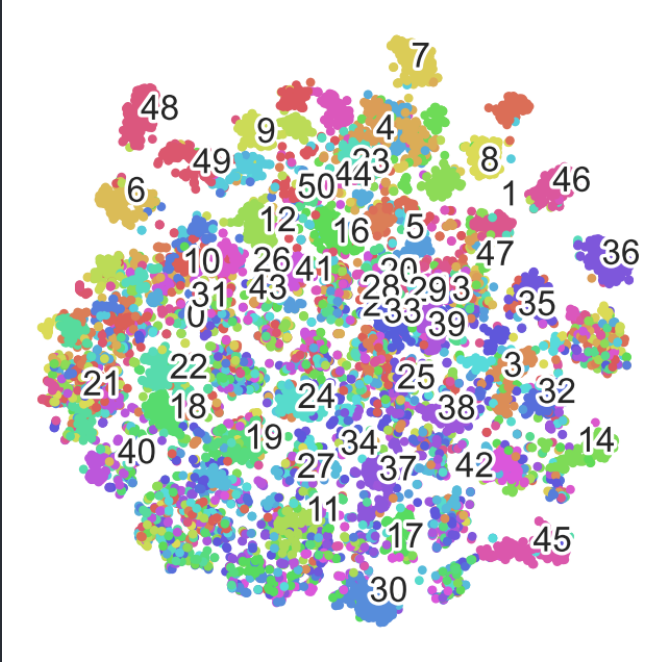


Fig. 10. TSNE clustering

TSNE clustering was carried out on the raw data without much wrangling to see if its possible to segment and cluster all 51 typists. However, although there are some typists that are quite separated from the cluster, there is still quite a number who types similarly to one another with overlapping typing speed. Therefore, this shows a need for a model that can not just leverage on the numerical value of the column but also data that is in sequence as that’s what typing is.

## Feature Engineering / Preproccessing

### Data Wrangling

I created a new column called ‘id’ which is a label encoded column of ‘subject’ this can help the model better understand and for evaluation as well. I then dropped the ‘subject’, ‘sessionIndex’ and ‘rep’ columns. Data is then split to X and y which is then further split into train, validation and test using scikit-learn train-test-split.

### Feature Scaling

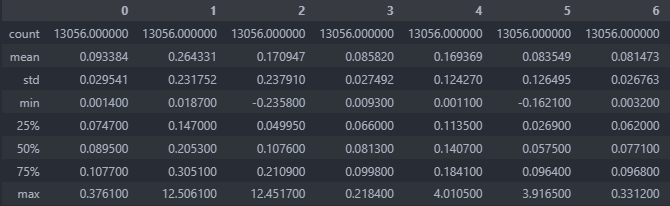


Fig. 11. Data frame without Min Max Scaling

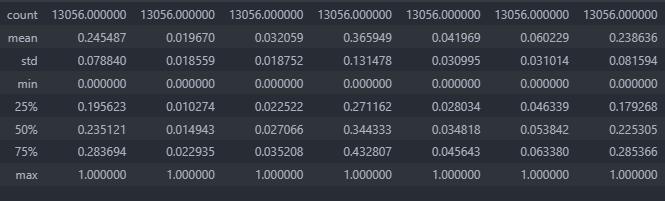


Fig. 12. Data frame with Min Max Scaling

I Imported MinMaxScaler from sklearn.preprocessing and scaled the train and test dataset. Min max scaling helps scale the range of the speed in the range from 0 to 1 for all typing columns

# Recurrent Neural Network(LSTM)

## Neural Network

Neural networks are a subset of machine learning, a combination of modern computer science and cognitive psychology. Data processing and analysis are based on the same processes that neurons in the brain process information. (J, Jeremy,2017) Your neurons require a certain amount of activation energy to fire and send information out of the neuron to axons, synapses, other neurons, etc. Data is fed into fully connected neural networks and matrix multiplication is carried out between the weights and the input. The weights are then improved through back propagation which reduces the validation loss of the function.

## Recurrent Neural Network

Recurrent neural work (RNN) is gaining more popularity due to its ability to retain sequential information. RNNs are called recurrent because they recurrently perform the same task for each element found in the sequence of the data with the output dependent on previous computations.

We can think of RNNs as model with memory (Denny ,2015)

## How LSTM works

LSTM stands for Long Short-Term Memory network used in the field of deep learning. It is a variant of RNNs which is extremely useful when trying to store and analyze data continuously. Therefore I feel that LSTMs are the best solution for Keystroke Biometrics. Furthermore, in previous years the models trained on the dataset are models that are fixed for the given string but LSTM can draw relations between various keys.

# Results and discussions

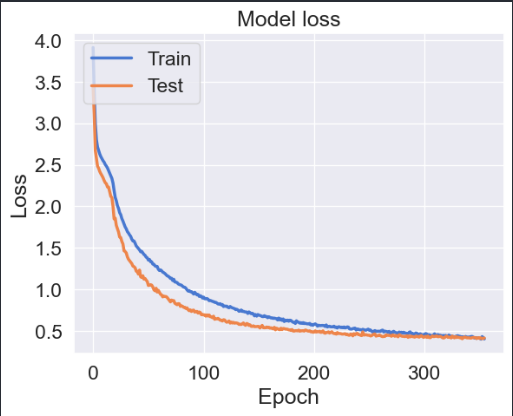
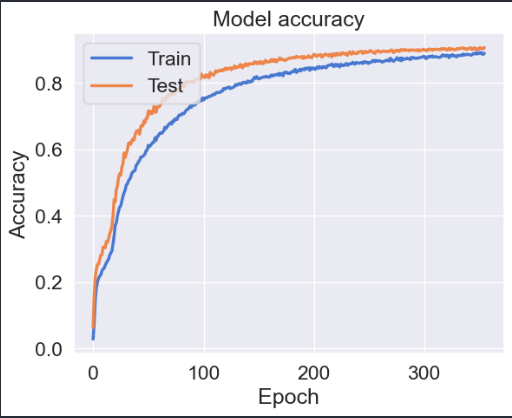


Fig. 13. Learning curve of LSTM model without scaling

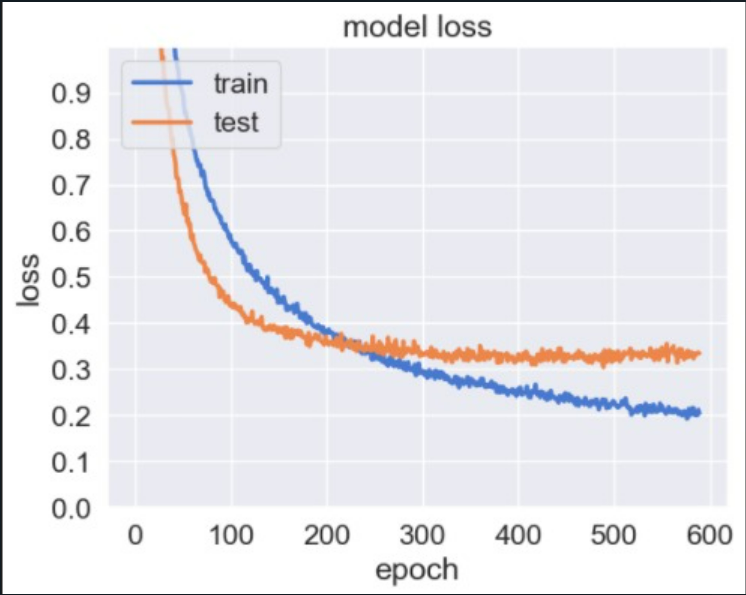
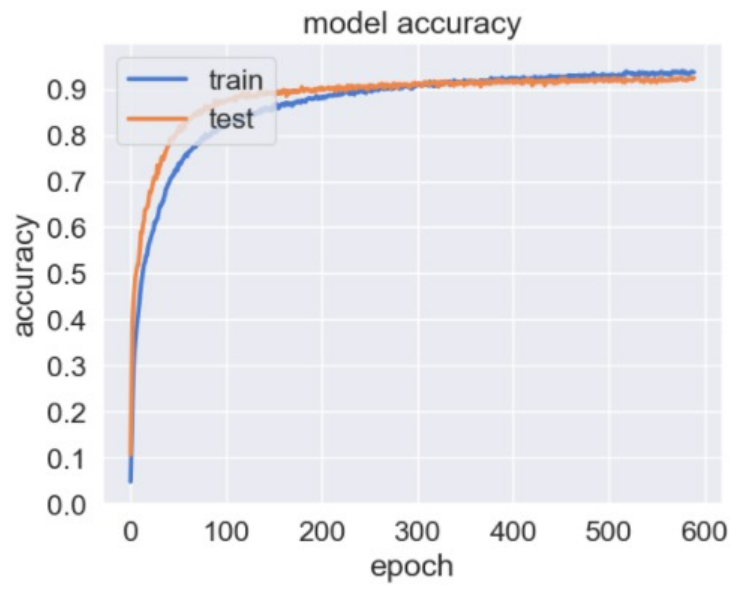


Fig.14. 92% test accuracy with scaling

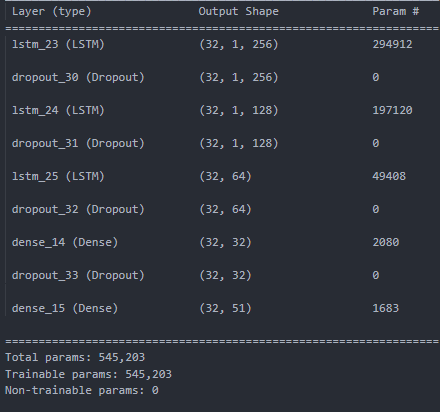


Fig. 15. Keras LSTM model summary

# Conclusion

The model was able to predict and identify the correct typist with a test accuracy of 92% with a loss of 0.3636 test loss. Overall, I am confident that with more data

##### Acknowledgment

I want to specially give credits to an awesome group of young machine learning engineers; Xavier tung, Lim Jun jie , David Sorkin and Pratik who inspired me to investigate new ways of authenticating and securing the digital space of our future.

##### References

1. S.,Calif *Cybercrime To Cost The World $10.5 Trillion Annually By 2025 [Available online]* [*https://cybersecurityventures.com/hackerpocalypse-cybercrime-report-2016/*](https://cybersecurityventures.com/hackerpocalypse-cybercrime-report-2016/) *[Accessed 23rd November 2022]*
2. K.,Kevin , M.,Roy,Accompaniment to “Comparing Anomaly-Detection Algorithms for Keystroke Dynamics” [ Available online] <https://www.cs.cmu.edu/~keystroke/#sec2> [Accessed 23rd November 2022]
3. ActiveState. (n.d.). *What Is Pandas in Python? Everything You Need to Know*. [online] Available at: https:/[/w](http://www.activestate.com/resources/quick-reads/what-is-pandas-in-)w[w.activestate.com/resources/quick-reads/what-is-pandas-in-](http://www.activestate.com/resources/quick-reads/what-is-pandas-in-) python-everything-you-need-to-know/#
4. Pydata.org. (2012). *An introduction to seaborn — seaborn 0.9.0 documentation*. [online] Available at: https://seaborn.pydata.org/introduction.html.
5. scikit-learn (2019). *scikit-learn: machine learning in Python*. [online] Scikit-learn.org. Available at: https://scikit-learn.org/stable/.
6. G., Orhan *Top 5 Deep Learning Frameworks to Watch in 2021 and Why TensorFlow [ Available online]* [*https://towardsdatascience.com/top-5-deep-learning-frameworks-to-watch-in-2021-and-why-tensorflow-98d8d6667351*](https://towardsdatascience.com/top-5-deep-learning-frameworks-to-watch-in-2021-and-why-tensorflow-98d8d6667351) *[Accessed 24th November 2022]*