International Journal of Psychosocial Rehabilitation, Vol. 24, Issue 05, 2020
ISSN: 1475-7192

Impact of Loss Function Using M-LSTM Classifier for Sequence Data

K. Sahityabhilash and S. Prayla Shyry

Abstract--- *The increasing dependence on information systems has opened a lot of possibilities in solving real life problems and led to the increase of threat to privacy, integrity and authentication. Even though a lot of key based authentication systems are in use biometrics provide a better performance, apart from physiological biometrics like iris, thumb impression etc., For verifying a person, a behavioural biometric technique called Keystroke dynamics can be used. Biometric based user authentication is a sequence classification task. This study provides a comparison of different loss functions and their performance on keystroke dynamics data. This work uses Long Short-Term Memory (LSTM) representing Neural Network and we have taken 5 different loss functions for the study.*

Keywords--- *Long Short-Term Memory (LSTM), Loss Function, Cross-Entropy, Hinge, Normalisation, Biometric, Key Stroke Dynamics (KSD).*

I. INTRODUCTION

Passwords are the most common mode of securing a user privacy or account. Even though a weak mechanism they are majority here. The key problem with passwords is the access can be granted to the persons who knows password even though he/she is not the actual owner of that account. This problem extends to streaming services where password sharing leads to loss in billions of revenues every year. Biometrics are proven solution to overcome the authentication problem as they provide more natural and unique identification to the user. Biometrics can be broadly classified into two types physiological and behavioural. Physiological are more static than the later which gives a more humane and natural identification of the user. Keystroke dynamics is a type of behavioural biometrics which vectorises the keypress durations.

Every individual has their own rhythm of writing, and the analysis of this rhythm is named as Key Stroke Dynamics. Authenticated system combined with KSD relies on the individuality of the users' writing patterns. The KSD by integrating with different authentication systems because of it doesn't need any further hardware, desires very less effort for associate degree implementation. This provides transparent and continuous authentication. In distinction, the very fact that someone could change their behaviour over time could cause an impaired performance by making a major challenge within the space.

II. RELATED WORK

Josef Malmström and Hannes Lindström trained the embedding network and prediction model on 90% of the

K. Sahityabhilash, Student, Department of Computer Science and Engineering, Sathyabama Institute of Science and Technology. E-mail: sahithyabhilash@gmail.com
S. Prayla Shyry, Associate Professor, Department of Computer Science and Engineering, Sathyabama Institute of Science and Technology. E-mail: praylashyry@gmail.com

DOI: 10.37200/IJPR/V24I5/PR202059
Received: 01 Mar 2020 | Revised: 24 Mar 2020 | Accepted: 02 Apr 2020

typing samples from 30 different random users, validated on 5% and tested on the remaining 5% can show that their one-shot approach achieves a 10.14% FAR (False Acceptance Rate) and a 15.26% FRR (False Rejection Rate) on samples from test data [5].

Vishnu Shankar and Karan Singh used the BOA (Butterfly Optimization Algorithm) to optimise the extracted features and selected the best. Using the SoftMax Regression (DAE-SR), Deep Auto Encoder and Hybrid deep learning technique the user will be identified and labelled with the selected features. DAE-SR achieved maximum accuracy of 0.970 % and 0.950 % on sitting and walking state [6].

Pawel Kobojek and Khalid Saeed used multiple LSTM and GRU networks with 2,3 LSTM layers trained with benchmark datasets and also artificial generated data and proved to be ineffective and best accuracy is achieved with GRU 3 cells at 70.7% and 0.389 zero-miss rate[10].

Kinga Enyedi and Roland Kunkli used Beizer Curves to propose a method to visualize the keystroke data they only used duration-based metrics and visualized for better understanding [11].

Laura Emmanuella and team used Brazilian hand-based data and GREYC, with 60 and 43 attributes and 7555 and 231 instances, respectively. Compared the performance of genetic algorithm approach with KNN, SVM and Naive classifiers. Their results achieved best accuracy with SVM at 90% [12].

III. DATASET

Benchmark dataset, because it was depicted earlier, in terms of samples per user and user count it contains additional info. Thus, it's plenty of reliable as it involves the algorithmic involvement. Samples within the dataset contain extra info than simply dwell-time. However, as the exclusive dwell-time is being recorded in the author's dataset, initial study was solely performed along with this metric.

The above flow represents the structure of the whole model flow which include the data collection, pre-processing with z-score, model construction and classification.

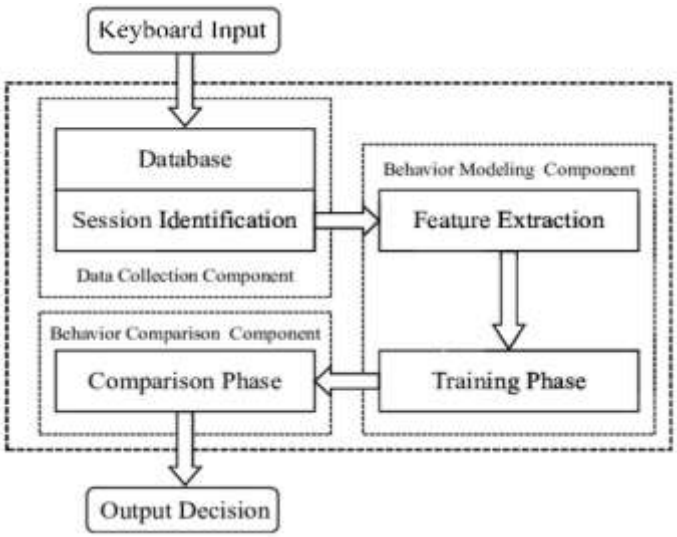


Fig. 1

DOI: 10.37200/IJPR/V24I5/PR202059
Received: 01 Mar 2020 | Revised: 24 Mar 2020 | Accepted: 02 Apr 2020

IV. PRE-PROCESSING

1. Z-score Standardisation

In this method, the standardisation is going to be done using the standard and mean deviation from the mean (SD) achieved for every information. The Z-Score standardisation is assigned by,

$$D\ v - \mu D \quad (1)$$

$$ZScore\_D\ Att = \sigma\ D \quad (2)$$

where, σD and μD represents the SD and Mean of the information values in corresponding attribute. n is all vary of knowledge. The SD and mean were determined by the addition of all the knowledge values in the current attribute.

$$\mu D = \text{No. of data values } P \quad (3)$$

$$(\text{Every individual knowledge value in the attribute} - \mu D)\ 2\sigma D = n.0 \quad (4)$$

The outliers within the dataset are removed with equation two. The analysis of planned algorithms allows us to introduce the subsequent terms, which is an important issue to contemplate.

A true positive is the respective outcome when the model predicts the positive category properly. Similarly, a real negative is the respective outcome when the model predicts the negative category properly.

A false positive is the respective outcome when the model incorrectly predicts the positive category. And a false negative is the respective outcome when the model incorrectly predicts the negative category.

- EER (Equal Error Rate) – threshold value while FPR and miss rate1 – TPR are equal.

V. LSTM

The core idea of LSTM is the cell state, and its varied gates. The cell state act as a transport main road that transfers relative info all the way down the sequence chain. We can use it because the "memory" of the network. The cell state in theory, will carry relevant info throughout the process of the sequence. Thus, even info from the sooner time steps will build its sequence to later time steps, reducing the results of memory. Because the cell state goes on its journey, info gets accessorial or removed to the cell state via gates. The gates are totally different neural networks that decide that info is allowed on the cell state. The gates will learn what info has relevancy to stay or forget throughout coaching. On paper, RNN algorithms will use data at long random sequences, however in application, they're restricted to solely a few steps.

VI. LOSS/COST FUNCTIONS

At its core, a loss perform is implausibly simple, it's a technique of evaluating how good your algorithmic program models your dataset. It quantifies the difference between actual value and predicted value. As you modify your algorithmic program to undertake and improve your model, your cost function can tell you if you're moving in the right direction.

For this work we used 5 loss functions to understand their impact on the sequence data. Those are namely Mean Squared Error (MSE), Categorical Cross Entropy, Binary Cross Entropy, Hinge Loss and Log Cosh.

DOI: 10.37200/IJPR/V24I5/PR202059
Received: 01 Mar 2020 | Revised: 24 Mar 2020 | Accepted: 02 Apr 2020

1. Mean Squared Error

Mathematically, it's the well-liked cost function beneath the reasoning framework of most chance, if the distribution of the target variable is Gaussian. It's the loss operate to be evaluated initially and solely modified, if you have got a decent reason. Mean square error is calculated as the average of the square variations between the expected and actual outputs. This results forever positive, despite the sign of the expected and actual prices with an ideal value as zero. The squaring means larger mistakes end in a lot of error than smaller mistakes, which means that the model is rebuked for creating larger mistakes.

2. Categorical Cross-Entropy

CCE is also known as SoftMax. Its loss a combination of cross-entropy and SoftMax activation in multi class classification we use this to derive probability of n-classes in each sequence. In the Multi-Class classification, the subjects are one-hot encoded. Thus, the positive class term is kept in the process.

$$CE = -log\left(\frac{e^{s_p}}{\sum_j^C e^{s_j}}\right)$$

3. Binary or Sigmoid Cross-Entropy

BCE is an ensemble of Cross-Entropy and a Sigmoid activation loss, result of it sets up a binary classification downside between C =2 categories for each category in C, as explained previously. In contrast to SoftMax loss it's individuality for every vector element (class), which means that the loss computed for each output vector element isn't suffering from alternative element values. That's why it is used for multi-label classification, where the insight of a part happiness to an exact category mustn't influence the choice for an additional category. It's known as Binary Cross-Entropy Loss therefore with support of this Loss, the formulation of Cross Entropy Loss for binary issues is usually used.

4. Hinge Loss

An equivalent to cross-entropy for binary classification problems is the Hinge Loss, it is intended for binary classification where the target labels are in the range {-1, 1}, developed mainly to use with Support Vector Machine (SVM) models.

5. Log Cosh

$$L(y, y^p) = \sum_{i=1}^{n} \log(\cosh(y_i^p - y_i))$$

"Log-cosh is the logarithm of the hyperbolic cosine of the prediction error." (Grover, 2019).

it is approximately equal to *half of square of x* for low x values and to *abs(x) - log (2)* for high x values. This means that 'logcosh' works mostly like the MSE, but mostly does not get affected by the occasional wrong estimations. [9]

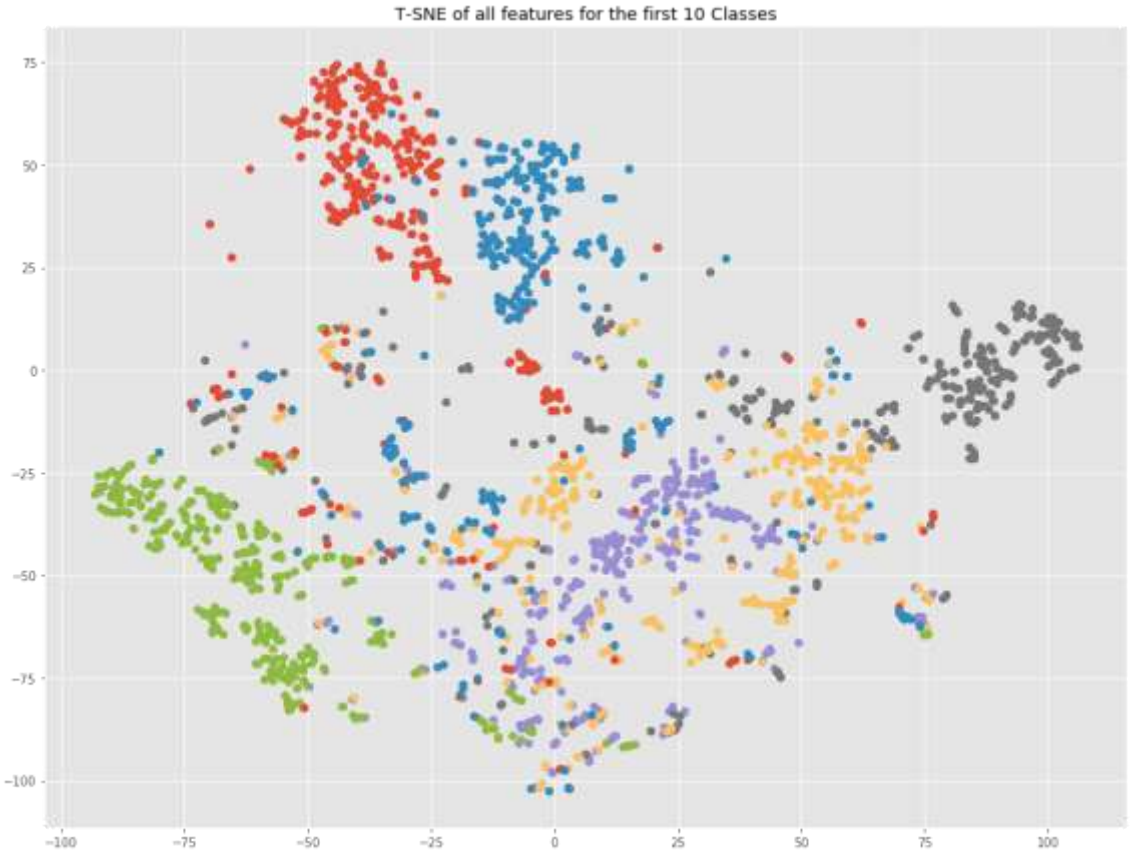
DOI: 10.37200/IJPR/V24I5/PR202059
Received: 01 Mar 2020 | Revised: 24 Mar 2020 | Accepted: 02 Apr 2020

VII. IMPLEMENTATION

Properly implementing LSTM or GRU on the given dataset, firstly requires grasping the output and input form of those networks. For building the models we tend to Keras library and, hence shapes are mentioned from it. For the input, the successive models in Keras need the 3D array. The size of the batch is delineating by primary dimension and refers to the samples' quantity to feed the neural network at a time [15]. As the variety of time, ticks in each of the samples, the quantity of time-steps is given by the second dimension [15]. The quantity of units is the dimension which stands for the quantity of options. This describes each of the time steps [15].

This work uses a core idea of splitting the users into two classes rather than multiple classes thus making it into a binary classification (one vs all), The pre-processing is done using Z-score normalisation thus making the attributes simpler for the model to process.

Visualizing the high dimensional data can be done by using the tool called t-SNE. For minimizing the Kullback-Leibler divergence between the joint probabilities of the high dimensional data and the low dimensional embedding, the similarities from the data points are converted to joint probabilities. t-SNE has a cost function which is not convex. This shows that with different initializations and we are able to get different results.\



T-SNE Plot

Thus, optimal learning rate is deduced to be 0.28 and there isn't much improvement after 20 epochs thus those hyperparameters were finalized

DOI: 10.37200/IJPR/V24I5/PR202059
Received: 01 Mar 2020 | Revised: 24 Mar 2020 | Accepted: 02 Apr 2020

VIII. RESULTS AND DISCUSSION

If we use n-class single label categorification one-hot encoded vector and our expected class label are similar. that the square Error if we concentrate on the anticipated category vector can continuously be zero or a pair of. If we have got a hundred observations and twenty of them square measure wrong, MSE = 0.4 and we may be ready to deduce their square measure twenty errors however that categories were expected as what's going to not be accessible. So, you are doing not get a confusion matrix and the worth of this MSE info is sort of low, thus making the MSE a poor performer even it gives a better loss rate.

Thus, the loss values suggest that Log cosh and MSE having a better cost buts suitable for sequence prediction as accuracy shows the other side of the coin as they are not able to produce a better accuracy on test data.

FUNCTION	*LOSS*
logcosh	1.16866
Hinge	13.9077
Categorial cross entropy	8.8225
Binary cross entropy	7.67853
Mean square error	2.35322

Whereas the both variants of cross-entropy gave a better accuracy than the others and can better fit the sequence classification tasks as our core objective is to make a binary classification Binary cross entropy and log cosh will be better fit whereas the Hinge loss under performed in all the scenarios.

LOSS TYPE	*ACCURACY*
logcosh	85.55%
Hinge	72.54%
Categorial cross entropy	83.72%
Binary cross entropy	84.50%
Mean square error	75.22%

Representation of loss and epochs is shown in below graphs with Fig. 2, 3, 4 representing the loss with 20 epochs for Hinge, log cosh and Mean squared error. Fig 5 represents the key down timings of sample users 1 to 8.

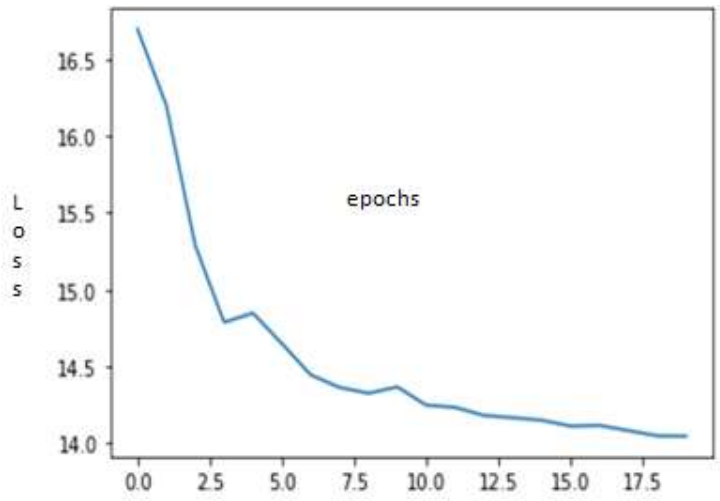


Fig. 2

DOI: 10.37200/IJPR/V24I5/PR202059
Received: 01 Mar 2020 | Revised: 24 Mar 2020 | Accepted: 02 Apr 2020

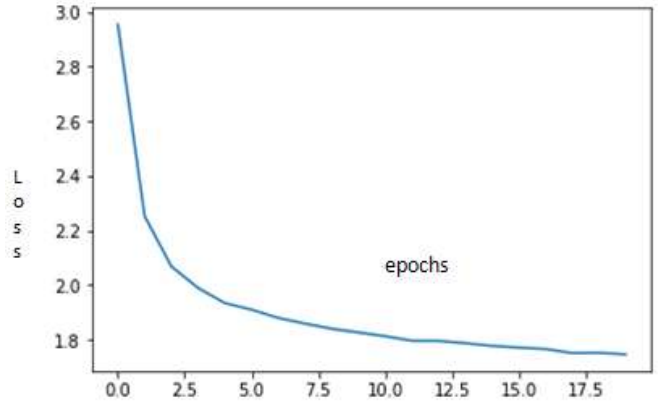


Fig. 3

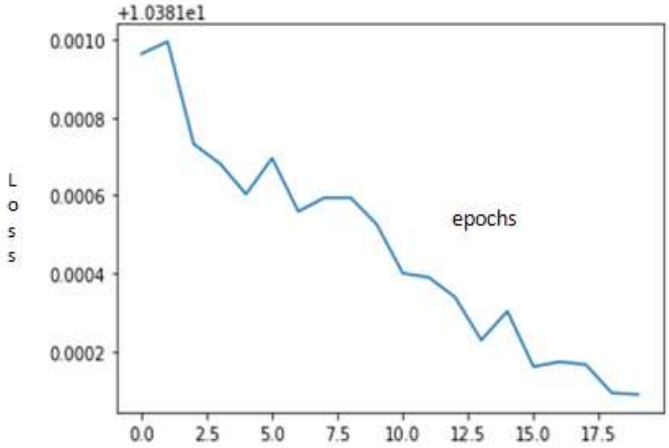


Fig. 4

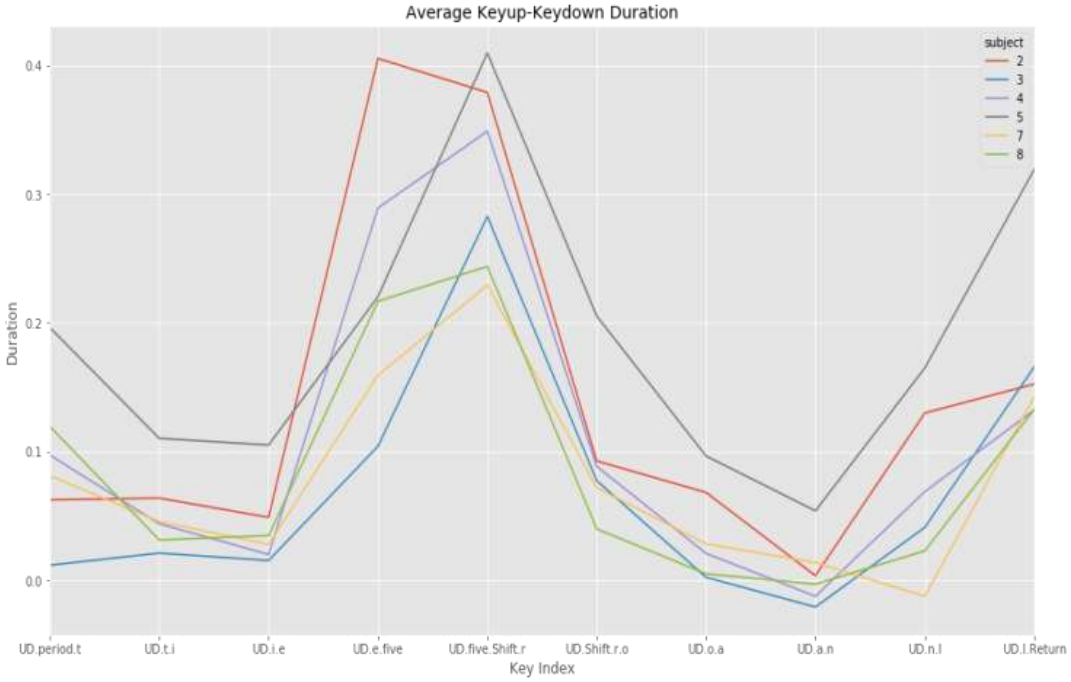


Fig. 5

DOI: 10.37200/IJPR/V24I5/PR202059
Received: 01 Mar 2020 | Revised: 24 Mar 2020 | Accepted: 02 Apr 2020

FUTURE WORK

This work can be further improved by fine tuning the architecture and hyperparameters and also using more mathematical loss functions. With a larger data these functions performance can be evaluated with much depth.

REFERENCES

[1] S. Hochreiter and J. Schmidhuber. (1997). "Long Short-term Memory." *Neural computation.* 9. 1735-80.
[2] Chen, Tianqi. Introduction to Boosted Trees. 2014. url:https://homes.cs.washington.edu/%7B~%7Dtqchen/pdf/BoostedTree.pdf (visited on 02/02/2018).
[3] Chen, Tianqi and Guestrin, Carlos. "XGBoost: A Scalable Tree Boosting System". In:CoRR abs/1603.0 (2016).
[4] DMLC, Distributed Machine Learning Community. XGBoost Documents. 2016. url:http://xgboost.readthedocs.io/en/latest/ (visited on 04/26/2018).
[5] Malmström, Josef, and Hannes Lindström. "Typing Biometrics for User Authentication-a One-shot Approach."
[6] Shankar, V., & Singh, K. (2019). An Intelligent Scheme for Continuous Authentication of Smartphone Using Deep Auto Encoder and Softmax Regression Model Easy for User Brain. *IEEE Access*, *7*, 48645-48654.
[7] Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." *Journal of machine learning research* 12. Oct (2011): 2825-2830.
[8] Buja, Andreas, Werner Stuetzle, and Yi Shen. "Loss functions for binary class probability estimation and classification: Structure and applications." *Working draft, November* 3 (2005).
[9] Deep Learning with Python, 2017 ISBN: 978-1-4842-2765-7 Nikhil Ketka.
[10] Kobojek, Paweł & Saeed, Khalid. (2016). Application of recurrent neural networks for user verification based on keystroke dynamics. 2016. 80-90.
[11] Enyedi, Kinga & Kunkli, Roland. (2019). Type Vis: Visualization of Keystrokes and Typing Patterns based on Time Series Analysis. 346-353. 10.5220/0007584103460353.
[12] Lima do Nascimento, Tuany Mariah, et al. "An investigation of genetic algorithm-based feature selection techniques applied to keystroke dynamics biometrics." (2019): 1-4.

DOI: 10.37200/IJPR/V24I5/PR202059
Received: 01 Mar 2020 | Revised: 24 Mar 2020 | Accepted: 02 Apr 2020