keystroke dynamics using advance deep learning model

Deepanshu Sharma[1,2], *

[1]School of Chemistry, The University of Michigan
[2]Physics Department, The University of Wisconsin
[3]Biological Sciences Department, National forensics sciences university

Abstract

Introduction

In Present times, generating a new password is not a difficult task. On the other hand, remembering these passwords has become all the easier because we tend to relate these passwords with significant events in our lives that are already stored in our memory, therefore they are easier to recollect when needed. Hence it is quite feasible to crack these pass- words quickly. Nowadays, advancements in technol- ogy and high computational power in systems have led to 8-digit passwords being cracked within hours or even minutes. Back in time, passwords were used for providing security, today the tables have turned and we need to give security to passwords. Pass- word authentication systems authenticate passwords alone, which is why there is a rapid breach of data. This gives rise to the need for other authentication systems like fingerprint, retina scan, and voice rec- ognizers, which are user-specific and hence identify authentic users from the rest. However, these sys- tems require additional hardware. When systems are developed using some methodologies like keystroke dynamics, it works as another layer of security be- yond password authentication. This methodology is based on the concept of two styles of typing text. One is free text typing and the other is fixed text typing. Free text typing is one where text is not fixed and the user enjoys the freedom of typing text as per their convenience. This is generally used in writing letters and e-mails. On the other hand, fixed text typing involves typing a predefined text like a password. If an incorrect password is entered, it will generate an error, so password typing is fixed text typing. In fixed text typing, we can recognize unique characteristics like typing rhythm in different users. Authentication of users can then be done by analyzing some parameters like key press time, key hold time, key release time and key transition time. keystroke dynamics are in consecutive patterns in fixed text typing so the data set of keystroke dynam- ics are in a sequential manner in which the data is like one key press time spam then another key press time and different people have their own pressure variation and latencies in key pressing style. in our data set having 51 users in which each user types the same password 400 times, the total data set is about 20400. having this data all users type this password ".tie5Roanl" This password is considered a strong password in this password having all the features of a strong password like having a special character ".", all small letter and capital letter characters with numeric characters, and length of the password is more than 8 character this is an indication of creating this password data set is the authentic data set. The data set has 31 features of keystroke dynamics, based on key press, key hold, key release, and key tran- sition. From this feature, we get new features like pressure variation, latencies, etc. of different users. for this data set we applied different classifier ma- chine learning model like "random forest", decision tree, boosting , svm, Naive bayes and many more. also specifically designed time-series data processing, classification, and prediction advance deep learning model like lstm, bidirectionlstm, GRU for classify the user by their keystroke dynamics pattern. The rest of the article is organized as follows:

Literature Review

Viktor Medvedev have develop a data fusion technique of different data set of keystroke on that ex- tract unique character and complex behaviour of key stroke in the different data and combine with deep learning methodology in which they face and work- out different challenges like different dataset have distinct passwords and typing style and distinct fea- ture which have to analyze in their model. They apply interpolation-based data fusion techniques to regulate and and normalize the data to uniform length of the total dataset of 54000 password record

*Corresponding author: jane@smith.com

Received: October 20, 2023, **Published:** December 14, 2023

of fixed text datasets(the Carnegie Mellon University dataset, the KeyRecs dataset, and the GREYC-NISLAB dataset) and implement neural network with a triplet loss function and archiving the best error rate of 0.13281. Eng. Shimaa S. Zeid, Prof. Dr. Raafat A. ElKamar, Dr. Shimaa I. Hassan investigate about keystroke dynamics in both type fixed text and free text typing. using different data set. in fixed text typing with different degree of password strength 1. weak password, 2. average password, 3. strong password and in free text typing there is no constraint for user typing on keyboard. Classify the the user employing different machine learning technique:- RandomForest (RF), Support Vector Machines (SVM), BayesNet (BN), and K-Nearest Neighbors (KNN), with the highest accuracy in the fixed text typing is 98.8% using RF for classifier and in free text typing the highest accuracy in the fixed text typing is 87.58% using random forest for classifier. Yohan Mulionoa, Hanry Hamb, Dion Darmawanb work on classify the user on the basic of their key stroke dynamics in this using machine learning model SVM with three differnt kernals Linear, RBF and Polynomial and creating deep learning model with Several optimizer and get the best result out of this form svm model it get the 71.15% accuracy and from the deep learning model Nadam optimizer give the best result of 92.60. Hussien AbdelRaouf. Samia Allaoua Chelloug , Ammar Muthanna , Noura Semary, Khalid Amin purposed Convolutional Neural Network-Based Keystroke Dynamics for Boosting in this study in this study they use data agumentation techniques to generate new columns using the data set to improve features and quantile transformation is used preprocessing the data for preparing the data to better capture by model and it uses a CNN with ensemble base technique employing in the main technique for Positive outcome. archiving the having 99.95% accuracy and 0.65% error rate . Asia Othman Aljahdali,Fursan Thabit,Hanan Aldissi,Wafaa Nagro they are analyzing the typing rythms of user in standard "qwery" keyboard in free text typing in the neutral state not having a specific emotions and using 2 different data set having different feature and analyze separately in the data set implies many algorithms and getting the best result in DBN model and for testing they take E-Assessment web application based and the result of testing is 95% accuracy and 5% error rate. Itay Hazan , Oded Margalit , Lior Rokach they resaerch about the real world problem in key stroke dynamics in which key stroke dynamics give the good result single user classification but in multi user shear same credentials like shared bank accounts. in this issue they take three different data sets and employing k-mean classifier in multi user they get 89% accuracy.

Methodologies

Modern cybersecurity is becoming stronger with regard to insider threats against it, such as stealing admin credentials either through monitoring or password extraction. Such people damage the systems while covering their trails by mimicking a real user. To defeat such threats, advanced intrusion prevention systems that work through deep neural networks check out the typing behavior patterns of the users. Static authentication captures unique keystroke dynamics when users enter passwords and use hold time, release-to-press time, and other timestamps to analyze individual typing characteristics for better security.

Data Set

the Carnegie Mellon University (CMU) keystroke dynamics dataset Description Biometric authentication data, 51 subjects, entering the password ".tie5Roanl,", 400 authentication attempts spread over 8 sessions, 50 repetitions per session, separated by at least 24-hour intervals in order to capture the change of daily typing patterns.

The dataset extracts 31 temporal features spread over three major categories: Dwell Time (DT), Down-Down Time (DD), and Up-Down Time (UD). These three categories represent different aspects of keystroke dynamics, which collectively form a particular typing signature:

DT: Dwell Time, accounting for 11 features, is the time between depression and release events for each key and is defined as DT = t(↑) - t(↓). This metric reflects the patterns of key holding and the characteristics of finger pressure as a result of individual motor control behaviors.

DD measures down-down time, providing 10 features that measure intervals between key presses in the following order: DD = t(↓) - t(↓). This latency measure captures typing rhythm and patterns related to keystroke sequences while indicating the user's comfort with the text and generally smooth typing.

Up-Down Time (UD) adds 10 features capturing the time interval between the release of a key and the subsequent pressing of another key, defined as UD = t(↓) - t(↑). The transitional features between keystrokes reflect how efficiently the user is moving his or her fingers.

the experimental protocol under controlled conditions, standardized equipment, and similar testing environments, this data collection enables analysis of intra-user variability, inter-user discriminability, and biometric stability over time. The structured approach in dataset collection about keystroke dynamics is helpful for research studies in behavioral biometrics, development of authentication systems, and human-computer interaction studies.

preprocessing

We standardized our keystroke dynamics feature vectors using the `StandardScaler` in our preprocessing pipeline because keystroke timing measurements are quite often not normally distributed and may include outliers due to differences in typing patterns, latency in the network, or infrequent keystrokes received too late. The `StandardScaler` centers each feature by subtracting the mean and scales it by dividing by the standard deviation so that all features have a standard normal distribution with a mean of 0 and a standard deviation of 1. In particular, for a feature x_i, the scaling operation is given by

$$x_i^{scaled} = \frac{x_i - \mu}{\sigma}$$

where $\overline{\mu}$ denotes the mean and $\overline{\sigma}$ the standard deviation of the feature within the whole dataset. The transformation removes noise in the learning process and all feature vectors are scaled onto a similar scale for easier comparison.

On the keystroke dynamics dataset, we apply the standardization formula for the features. We then reshape the data into the shape required by inputting the deep learning model, particularly for the temporal features which are dwell times (DT), down-down times (DD) and up-down times (UD). The reshaped data preserves the relative ordering of values and ensures a good input for models like LSTM or GRU. The reshaping operation neither alters the standardized values nor reorganizes the features into the required shape.

This preprocessing step stabilizes training and enhances the learning capabilities of a model based on keystroke timing patterns by making the distributions of the features standard and reducing the effects of outliers.

Proposed Model Architecture

This section presents a deep learning model that is specifically tailored for the analysis of keystroke dynamics to biometric authentication. The Residual Bi-GRU is here adapted for the extraction of particular typing patterns of any user by adding residual connections and normalization layers to the model to address the temporal dependencies within keystroke dynamics data. The model thus can classify the users effectively by their keystroke sequences by analyzing the particular timing as well as rhythm patterns of each unique user.

For the input sequence of keystroke features, $X = \{x_1, x_2, \ldots, x_T\}$, in this work each is $x_t \in \mathbb{R}^{31}$ representing 31 temporal features at time step ; the model takes this sequence through a number of Bi-GRU layers, residual connections, and a dense layer for final classification.

Key Architecture Details

- **Input Layer**: The input layer accepts a sequence of 31 temporal features abstracted from keystroke data. These include key press timings, dwell times, and inter-key delays
- **Bi-GRU Layers**:The time dependencies in keystroke sequences can be captured by stacking two layers of Bi-GRUs. Each Bi-GRU layer has 256 units and ReLU activation is applied for better temporal feature learning. We can denote the Bi-GRU layer at time step t as:

  $$h_t^{(l)} = Bi - GRU^{(l)}(h_t^{(l-1)}, h_{t-1}^{(l)})$$

  where l represents the layer index. Each layer is followed by layer normalization for stability and generalization.
- **Residual Connections**: To enable efficient gradient flow and capture finer details of temporal dependencies, residual connections link the Bi-GRU layers, preserving information across layers.
- **Dense Layer**: After the Bi-GRU layers, a fully connected layer processes the final sequence representation for classification. The output of the dense layer is:

  $$y = \tanh(Wx + b)$$

  followed by a softmax activation for class probabilities:

  $$p(y_i|x) = \frac{e^{z_i}}{\sum_{j=1}^{51} e^{z_j}}$$

  where z_i is the logit for class i.
- **Optimization and Loss Function**: The model uses the Adam optimizer with categorical cross-entropy as the loss function:

  $$\mathcal{L} = -\sum_{i=1}^{51} y_i \log(\hat{y}_i)$$

  where y_i is the true label, and $\hat{y}_i$ is the predicted probability for class i.

Model Summary

The model comprises 304,051 trainable parameters in total. Table 1 provides a breakdown of each layer's configuration and parameters.

Table 1: *Model Architecture Summary*

Layer (type)	Output Shape	Param #	Description
input_layer (InputLayer)	(None, 1, 31)	0	Input layer
Bi-GRU_1 (Bi-GRU)	(None, 1, 256)	123,648	Temporal feature extraction
Dropout_1 (Dropout)	(None, 1, 256)	0	Regularization layer
LayerNorm_1 (LayerNorm)	(None, 1, 256)	512	Normalization layer
Bi-GRU_2 (Bi-GRU)	(None, 1, 128)	123,648	Temporal feature extraction
Dense_1 (Dense)	(None, 1, 128)	32,896	Fully connected layer
Dropout_2 (Dropout)	(None, 1, 128)	0	Regularization layer
Reshape (Reshape)	(None, 1, 128)	0	Reshape layer
Add (Add)	(None, 1, 128)	0	Element-wise addition
Flatten (Flatten)	(None, 128)	0	Flatten layer
Dense_2 (Dense)	(None, 128)	16,512	Fully connected layer
BatchNorm_1 (BatchNormalization)	(None, 128)	512	Normalization layer
Dropout_3 (Dropout)	(None, 128)	0	Regularization layer
Dense_3 (Dense)	(None, 51)	6,579	Classification layer

This architecture captures the short-term as well as long-term dependencies in the typing behavior for improving the model's discriminative power for users based on keystroke dynamics. The Bi-GRU layers coupled with residual connections and dropout layers together aids in making the model robust for generalization purposes, thereby fitting well for biometric authentication applications.

Results

Humanize AI We used CMU keystroke dynamics dataset that has temporal features from 51 users entering a common password over multiple sessions, so all in all, we have 20,400 samples. We used several machine learning and deep learning models to evaluate performance across common metrics: precision, recall, and F1-score. Here is the confusion matrix and classification report for our best model, which achieved an accuracy of 94%.

Confusion Matrix

The model's confusion matrix indicates strong predictive power, with most users accurately classified and minimal misclassifications, demonstrating the model's efficacy in differentiating typing patterns.

Classification Report

Overall, the model performed robustly to identify unique individual keystroke patterns with high precision and recall across classes, indicating that our temporal features along with our preprocessing and model architecture successfully capture individual user characteristics.

Model	Precision	Recall	F1-score	A
LSTM	0.88	0.87	0.87	
GRU	0.85	0.84	0.84	
Bidirectional LSTM	0.90	0.89	0.89	
Proposed Model (Bi-GRU)	**0.94**	**0.94**	**0.94**	

Table 2: *Performance comparison of different models on keystroke dynamics dataset*

Conclusion

The proposed Residual Bi-GRU model outperforms traditional machine learning models and standard LSTM models in getting a 94% classification accuracy of keystroke dynamics. Our results emphasize the fact that capturing both the short-term as well as the long-term dependencies within keystroke sequences is important, where the architecture of Bi-GRU with residual connections excels. Validating the generative capability of the model for all the individual classes shows high precision and recall scores for all classes.

This study thus brings to the limelight that advanced architectures of deep learning happen to be apt for keystroke dynamics and can develop promising solutions for secure, biometric-based authentication.

Discussion

A keystroke dynamics-based classification model can be very promising for strengthening advance cybersecurity. Notably, it can enhance the biometric-based user authentication. The proposed Residual Bi-GRU model can classify a person efficiently through typing behavior. Forming a supplementary security layer against varied cyber attacks may be possible. Since this approach is based on keystroke dynamics, which are characteristics very difficult to reproduce by an attacker, it makes it more resistant against unauthorized access if passwords or other static credentials get compromised.

Applications and Problem-Solving Potential

Keystroke dynamics-based systems help solve several critical problems associated with security through authentication. The traditional password-based systems are vulnerable to brute-force attacks, credential stuffing, and phishing attacks. It is proposed, however, that our model could add an extra layer of authentication through behavioral characteristics, giving better security against the above-mentioned attacks without needing more hardware or more invasive methods.

Keystroke dynamics can thus be applied in applications where it would be impractical or too costly to install hardware biometrics, such as finger scanning or facial recognition systems - as a demonstration of an enterprise-scale application or in scenarios such as telecommuting.

This technology may be used for continuous monitoring of user behavior so that anomalous typing patterns may be detected that indicate account takeover or insider threats. Continuous verification ensures only authorized users can maintain access throughout the session; in effect, this adds a dynamic security measure that helps solidify the overall security posture of an organization.

Future research in keystroke dynamics is thus envisioned as adaptive models in learning changing user typing patterns, context-aware dynamics adapting to the variation of devices and environments, and innovative forms of data augmentation for better generalization. This will significantly enhance security as a result of integrating keystroke dynamics with multi-factor authentication (MFA) mechanisms, thereby ensuring robust and flexible authentication systems.

Keystroke dynamics have purposes other than traditional security. In financial services and e-commerce, it can aid in fraud prevention systems by monitoring transactions to recognize suspicious activities, thus helping detect a takeover of accounts. For educational purposes, it authenticates students before any online test is conducted but with minimal risk of impersonation. Keystroke dynamics can also be applied to digital wellbeing by keeping an eye on the patterns of typing or detecting signs of stress or fatigue so that systems may adapt for better productivity and mental health. These applications highlight the many possibilities of keystroke dynamics application, not only in security but also as it pertains to user experience and wellbeing, hence it is a versatile tool crossing multiple sectors.

Future Scope

Investigate incorporation of other state-of-the-art architectures, such as based on Transformer models or attention, to improve performance further, especially for larger and more diverse datasets. Temporal mechanisms may also allow the model to attend to appropriate subsets of the keystroke sequences for improving classification accuracy further. This potential for the new model should also be explored in dealing with multiple types of biometric data, thereby opening up avenues to greater usage in multiple modalities of biometric authentication.