Static Keystroke Dynamic Authentication (SKDA) Model to Authenticate User during Password Change

1st Nataasha Raul
Sardar Patel Institute of Technology
Mumbai, India
nataasharaul@spit.ac.in

2nd Radha Shankarmani
Sardar Patel Institute of Technology,
Mumbai, India
radha_shankarmani@spit.ac.in

3rd Padmaja Joshi
Centre for Development of Advanced Computing
Mumbai, India
padmaja@cdac.in

2023 3rd International Conference on Smart Data Intelligence (ICSMDI) | 978-1-6654-6487-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICSMDI57622.2023.00054

***Abstract*—Keystroke dynamics is considered as a supporting factor of authentication. Especially in the static keystroke dynamics, the user is identified by using the timing featured, which is captured while the user enters the login ID and password. To achieve this, the user profile needs to be created with timing features. However, in a scenario like a change of password where nearly no keystroke timing data is available, non-conventional features may be helpful. This article focuses on using non-conventional features such as NumLock key, Shift key, CapsLock key, etc for identifying users during a change of password. The paper also details how to capture the non-conventional features in static keystroke dynamics and build a model that can be used in the change of password.**

***Index Terms*—Keystroke, Non-conventional, Authentication, Static, Password**

I. Introduction

Password based authentication is the most frequently used authentication method for providing access to personal computers, online services, internet access, etc. As password based authentication has many limitations many times to strengthen the authentication multi-factor authentication is used [1]. Which results in user grievance. To over come this concern, use of support factor authentication is recommended. One such support factor of authentication is keystroke dynamics which is based on behavioral characteristics of a user, in this case the typing behavior [2] [3]. Thus, along with the correctness of the password, the keystroke analysis is used to verify if the user is the same as he claims. This behavioral biometric technique is considered to be unique per user.

There are two types of keystroke dynamics: static and continuous. In the static keystroke dynamics, the typing rhythm of a user is captured only at login time whereas in continuous keystroke dynamics keystrokes are captured continuously when the user is using his/her personal computer. For an identity management system which is checking the authenticity of the user only during the login process, static keystroke dynamics technique is used, as keystrokes cannot be captured by the system continuously.

A lot of research has taken place in the use of static keystroke dynamics using timing parameters referred to as conventional features for authenticating users. The use of timing features requires capturing timing of every key press and release for the given characters. It means that when the password is decided for understanding the timings of the user it needs to be captured atleast a few times until then it can not be used as a support factor as well. Use of various keys such as Caps Lock key, Shift key and Backspace Key like information is captured in continuous keystroke dynamics to improve the performance of the decision. As these do not capture timing such features are called as Non-Conventional features. In our earlier work, we have experimented and demonstrated that non-conventional features when combined with timing features can become a better supportive factor for user authentication and system performance is improved in static keystroke dynamics.

Since in static keystroke dynamics, a model is built for the user through training of timing features as well, whenever the user changes the password, the training cycle is again required to build the model for the user for a new password. It means that we will not be able to make the user of this supporting factor until the system gets trained. In this paper we propose that the use of non-conventional features in static keystroke dynamics can be used during change of password. It will help to predict the user during the training cycle as the model based on the non-conventional features for the user will be available. This paper explores and discusses this possibility and proposes a model that uses non-conventional features as a support factor when the user wants to change the password until the model gets created through timing features for the new password.

The paper is organized as follows: Section II of the paper gives the literature study and research done in the area of keystroke dynamics, non-conventional features and the existing dataset. Section 3 describes the Keystroke Dynamics System and its features, also explains the the password change scenario and the challenges in using only the timing features. Section IV gives details about the keystroke dynamics non-conventional features, how to capture them, challenges in using them in change of password scenario, need of policy, describes the policies that are used for framing passwords and text to capture non-conventional features in static keystroke dynamics along with the test cases. Section V explains the

978-1-6654-6487-1/23/$31.00 ©2023 IEEE
DOI 10.1109/ICSMDI57622.2023.00054

proposed approach to authenticate users during change of password through the use of static keystroke dynamics with non-conventional as well as timing features, along with experimentation which was carried out on the proposed approach to validate the model. Section VI focuses on the experimental results of the research work. Section VII concludes the paper with concluding remarks.

II. Literature Survey

The typing characteristics of a user are captured through feature extraction in keystroke dynamics, which is further used for authentication and verification purposes [4] [5]. The keyboard specifies, aside from the text itself, the exact timing of each key press and release. These are captured as the timing features for keystroke dynamics. The features are derived from the simple data captured during typing and used as feedback for the classification algorithm [6]. The collection of user authentication features and classification algorithms differs from researcher to researcher [7] [8].

Digraph refers to a two-character sequence where characters includes punctuation marks, numerals, letters and spaces. Digraphs may not be able to identify the users independently. For diverse purposes like finding error rate while typing, digraphs are combined with other features such as tri-graphs and n-graphs. Some such examples are discussed ahead. For identifying the types of errors that a user makes when typing, D. R. Gentner et. al. [9] used trigraphs and digraphs. When a user types on a keyboard, Roth et. al. [10] incorporated digraph elements to the keystroke sound. When a key is pressed, one of their fundamental assumptions is that each key will generate a little varied acoustic signal depending on the user. They were motivated to learn a virtual alphabet by clustering sample keyboard noises. The score was then calculated using the digraph latencies inside the pairs of virtual letters. The keystroke features such as digraphs, tri-graphs and n-graphs depends only on the word context [11]. To overcome these dependencies Dowland and Furnell [12] used digraphs, tri-graphs and n-graphs features, as well as the keyword latencies AutoID, Left character, Right character, Latency, and Timestamp. They achieved the most promising results when using digraph latencies.

A. Non-conventional Features

Typically, latency time (distance between keystrokes characterizing the typing speed) and dwell time (amount of time a key is kept down, characterizing a typing style) are used as features of keystroke dynamic [4] [13]. As these features are related to the time, these are called as timing features. Its observed that in static keystroke dynamics timing features are prominently used but the information about which keys are used in keystroke is not been tried. Use of keys such as the Shift Key, Caps Lock Key, Number Key, and the use of the Left or Right Shift Key may provide additional information about the users key uses which can be useful in user authentication. These are called as non-conventional features in continuous keystroke dynamics. User recognition and classification is a necessary steps in keystroke dynamics to authenticate the user using machine learning algorithms. The user's typing pattern is only learned when data on the user profile is trained. When the classification is completed, the data is checked/tested for its correctness [14].

B. Data Set

Most researchers have developed and used their data-set for training and testing, and for comparison with similar algorithms [15] [16]. All of the available data sets contain information only about timing features such as KeyPress, KeyRelease, Hold time and Flight/latency time.

The focus of this research work is to use non-conventional features in static keystroke dynamics, for user authentication and when the user wants to change the password. Data-set for non-conventional features (such as the use of Caps Lock key, Shift key (left or right), and Number key (top/right part of the keyboard) are not available. Also, it was observed from the study that the work done using non-conventional features in the static keystroke dynamics approach is very less as compared to the continuous keystroke dynamics approach [16]. Hence, there was a need to build our data-set to carry out the research work, as we aimed to incorporate non-conventional features at the time of authentication and at the time when the user wants to change the password, in the static keystroke dynamic approach.

In this paper, we propose an approach of using non-conventional features change of password scenario as very little data is available for the changed password. The policy designed for this use case is discussed in the coming sections of this paper.

III. Keystroke Dynamics System and Features

The study of a number of key events and the intervals between them is known as the keystroke dynamics. In order to capture timing features of keystroke, KeyPress and KeyRelease are two most important extracted features. Based on these two captured features, other timing features are derived.

- Hold time is a distance between KeyPress and KeyRelease timing [17].
- UP-Up time is the distance between two consecutive KeyReleases [18].
- Up-Down time is the distance between a KeyRelease and the next KeyPress [18].
- Down- Down is the time between two consecutive KeyPresses [17].
- Down-Up time The time difference between a KeyPress and the next KeyRelease is known as down-up time [17].

In a keystroke dynamics system, users' keystroke features are first captured to create user profile for him/her. The keystroke dynamic system then analyses users' typing patterns and creates user profile. From the user's typing rhythm different timing features are collected. These features are used to form a training set that can be used as a reference during the

authentication phase. During the authentication/testing phase, the typing rhythm of the given input sample from the user will be compared with the reference template of the training set to verify the user. Timing based features are useful in detecting authenticity of the user based on its profile created. The technique cannot be used until the profile of the user is created.

A. Change of password scenario

The system described in the earlier section in useful while verifying the password. But when the user changes the password, the model derived from the training of different password may not work until a new model is developed for the new password. The trained timing features will be helpful only in verifying the user through their typing behaviour of specific words. Entering login and password being Static keystroke dynamics additional typing information cannot be captured and their is a dependency on the model developed for the specific password.

In this paper we are discussing the use of non-conventional features in static dynamics and the change of password is a use case where time based model is not useful. Hence, the change of password scenario is considered and discussed. The possibility of two users having the same typing speed is possible, thus timing features alone will not be enough to verify the benign user.

B. Challenges of using timing features

As can be observed in Fig. 1 two users typing the password ”India@2018” have almost the same timing distance in some of the timing features but have a different patterns when pressing the Numbers key and pressing & releasing the Shift key. Thus, finding non-traditional feature usage patterns, with the timing features to identify the user, may prove to be helpful. As can be seen in Fig. 2, usage of non-conventional features by two different people can be very different. Thus, if the non-conventional characteristic typing pattern is recorded, it is expected to be more helpful in distinguishing users especially in change of password scenario where the timing information is not available. Hence the challenge is of using non-conventional features to identify the user is explored.

IV. Non-Conventional Features

In the case of keystroke dynamics, capturing the usage pattern of Shift Keys, Caps Lock Key, Num Key and Number Keys can be considered as non-conventional features.

Non-conventional features are easier to capture in continuous keystroke dynamics as the system gets a good amount of time and opportunities to capture the user behavior in using the keys. However, in static keystroke dynamics, where the system has access to the user's typing information only during login, capturing non-conventional features is a challenging task.

Possibility of using non-conventional features in static keystroke dynamics was discussed in our earlier research work,

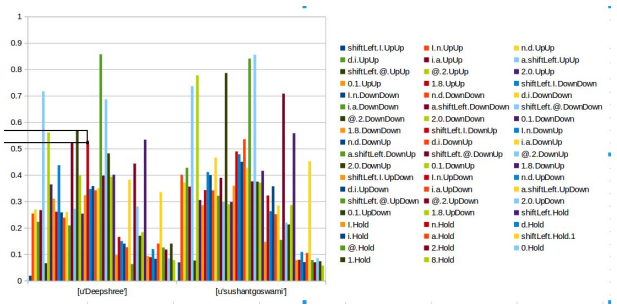

Fig. 1. Conventional features timing plot of two people inputting the identical phrase at various times.

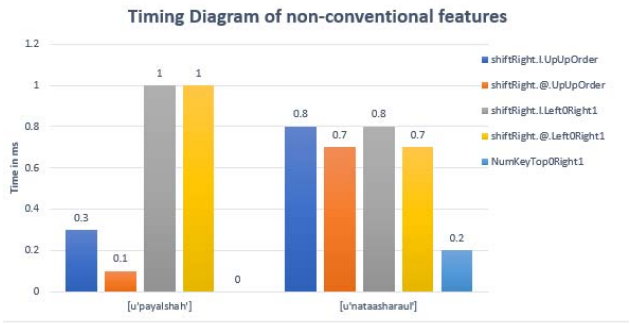

Fig. 2. Non-conventional features timing plot of two people inputting the identical phrase at various time.

which was experimented on a smaller user base, to identify the user more effectively [19]. The non-conventional features used in our research are elaborated ahead for easy reference and to apply it to the use case of Change Password.

A. Factors of Non-conventional features

- ***Usage of (Left/Right) Shift Key***: As can be seen, everyone distinctively uses the keyboard, especially when typing capital letters or unusual characters like punctuation marks. While typing capital letters, the user has a choice to user CapsLock or Left Shift Key / Right Shift Key. While typing punctuation marks as well user has distinct linking to use Left ShiftKey or Right ShiftKey. In our observation, users many times have certain habits of using these specific keys. To capture the user behavior of using these specific keys, which are considered as non-conventional features, the usage at specific circumstances by the user is recorded regularly. This feature will track how often a person uses the specific shift keys.
- ***Usage of Caps Lock Key***: This records the user's habits related to use of CapsLock while typing capital characters [20].
- ***Usage of (Top/Right) Number Key***: Most keyboards have number keys at two locations, at the right side of the keyboard and above the letters. While typing numbers

users demonstrate favoritism in using the number key sets. People habitual to use the number keys on the top panel are observed to use the same more often. The same applies to the people using the right panel. Few are observed to use them with convenience. It means that the usage of number keys may help identify the typing pattern of the user and hence is considered as one of the non-conventional factors to be captured for typing.

- ***Releasing the first key before releasing the second key***: This is another interesting habit. When the user presses two keys simultaneously as is done using non-conventional factor - ShiftKey as discussed earlier, (s)he has to release keys in some sequence or simultaneously. This feature focuses on key-release pattern of the user. Though the time of release is considered here, it's not the timing feature. This is relevant if the person holds down the Shift key as shown in Fig. 3. For instance, this happens when a person hits Shift and afterward presses some other key and then releases Shift before releasing the other key [20].
- ***Releasing the second key before releasing the first key***- This is relevant if the person holds down the Shift key as shown in Fig. 4. For instance, this happens when a person hits the Shift key and afterward presses some other key and releases the other key before releasing the Shift key [20].

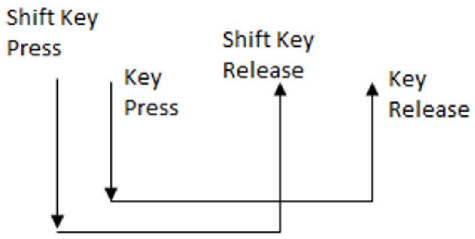

Fig. 3. Releasing the first key before releasing the second key

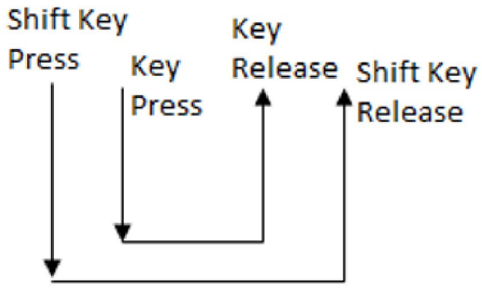

Fig. 4. Releasing the second key before releasing the first key

To investigate whether non-conventional features can be used as a support factor to differentiate users, typing behaviour of the 30 users was studied, by capturing the typing rhythm of each user while typing "India@2018" 15 times. It has been confirmed that two different users who enter the same password may have different non-conventional factor based input pattern as represented in Fig. 1.

B. Need of Policy to be framed

As discussed in Section II, the work of static keystroke dynamics mainly focuses on timing parameters for authenticating the user. As stated in [19], people use the keyboard in a specific style. The insight of the use of various keys such as Caps Lock, Shift, or a particular set of number keys as keystroke feature can lead to additional. These features are called non-conventional features because they are not time-captured but connected to the type of key press and release [19]. It is easier to capture such non-conventional features in continuous keystroke dynamics but is very challenging in static keystroke dynamics due to constraints on the length of username and password and hence the opportunity received to capture the use of these non-conventional features.

To gather more information about the user's non-conventional feature usage, we have framed policies and designed statements covering these policies. These statements are displayed to the user randomly along with the password. So the use of non-conventional features by the user can be gathered for additional information.

C. Policy for framing password and text

For security reasons, there is a requirement that the length of the password should be at least eight characters and it must contain at least one uppercase character, a special character(! @,#,$,%,&, etc) and numbers. But for this experimentation we have put an additional requirement that is the length should not be more than "10" characters. To capture the non-conventional features from the desired requirement of password setting, below five policies were designed.

- **Policy 1 - The uppercase character and the special case character together (Any order)**: For pressing the uppercase character the user will either use a Shift key or a CapsLock key and for pressing the special case character the user has to use the Shift key. This policy tries to identify which key the user will prefer CapsLock or a Shift key for pressing uppercase character. Also in the case of Shift key which one left/right Shift key the user is habitual to use. Thus, this policy helps in capturing the use of CapsLock or right/left Shift key by the user.
- **Policy 2 - Several uppercase characters together**: If there are multiple consecutive letters capital, the user has a choice of keeping the shift key pressed by one hand and typing the characters by the other or using a CAPS lock to type the uppercase characters. The choice is usually based on habits. Thus, this policy focuses on the use of CapsLock by the user.
- **Policy 3 - Alternate upper and lower case characters**: This policy brings out the use of the Shift/Caps key by

justification=centering,margin=0.5cm

TABLE I
TEST CASES FOR FRAMING THE TEXT USING THE POLICY

Test Cases	Examples	Users showing the same pattern behaviour
Case 1: Keeping the uppercase character and the special case character together (Any order)	@W, N@, H@, &S@, @P	86%
Case 2: Several uppercase characters together	CAST, SERVICES, MADE, BEN	86%
Case 3: Alternate upper and lower case characters	MaDe, bAnKiNg	92%
Case 4: Consecutive Numbers	4357, 009	93%
Case 5: Half upper-case Half lower-case	BENchmark	93%

the user. Some users may either use only shift(left/right) keys to type the uppercase letter. While some may prefer Caps Keys to type the uppercase letter.

- **Policy 4 - Consecutive Numbers**: This policy brings out the use of Num Lock keys or the Top vertical numerical panel of the keyboard. The behavior of the user to type the numbers using Num Lock keys or top panel can be captured.
- **Policy 5 - Half uppercase Half lowercase**: This policy brings out the use of the Caps key by the user. Some users may either use only Caps keys to type the uppercase letter. While some may prefer Shift (left/right) Keys to type the uppercase letter.

Using the above mentioned policies, the following three texts were framed which will help to capture various combinations of non-conventional features that users will be using while forming their passwords. Note that all three statements cover all the five policies mentioned above.

1) Plumb MaDe @Way CAST 4357
2) Make N@tions bAnKiNg Agent SERVICES 009
3) Express!ons MADE BENchmark H@mS@m @Pale

D. Test Use Cases Designed

To study the typing behavior of the user, data was collected from 50 users, where users were asked to type the given three sentences one after another 5 times. Examples and percentages of users showing the same typing behavior while choosing their passwords irrespective of any text typed are summarized in Table I.

It was observed that the typing behavior of the users in using the keys such as Shift Key(left/right), CapsLock Key and Number key(Top panel/right-side panel of the keyboard) remains almost the same while typing. The results are shown in the Table I. This indicates that the policies are well defined and can help capture non-conventional features for the users typing a password.

The evaluation results and the performance matrix of the data collected from the users while typing the text is shown in Table II.

TABLE II
PERFORMANCE MATRIX FOR TEXT TYPED BY THE USERS

Parameters	Test Case 1	Test Case 2	Test Case 3	Test Case 4	Test Case 5
TP(True Positive)	65	68	69	68	62
FP(False Positive)	7	8	6	14	9
TN (True Negative)	21	20	22	14	19
FN(False Negative)	5	2	1	2	2
FAR(False Acceptance Rate)	0.25	0.29	0.21	0.5	0.32
FRR(Fasle Rejection Rate)	0.07	0.03	0.01	0.03	0.03
EER(Equal Error Rate)	0.16	0.16	0.11	0.27	0.18

It was observed from the results that the user's non-conventional behavior remains unchanged with respect to the use of the Shift/Caps keys even when they change their password. Also, the users tend to use the same shift (left/right) key and numbers key(Top/Right section) while typing the text irrespective of the text given, let it be password or a given statement or normal typing. So, taking into account the observation from the result, the proposed Static Keystroke Dynamics Authentication(SKDA) Model was designed.

V. PROPOSED STATIC KEYSTROKE DYNAMICS AUTHENTICATION(SKDA) MODEL

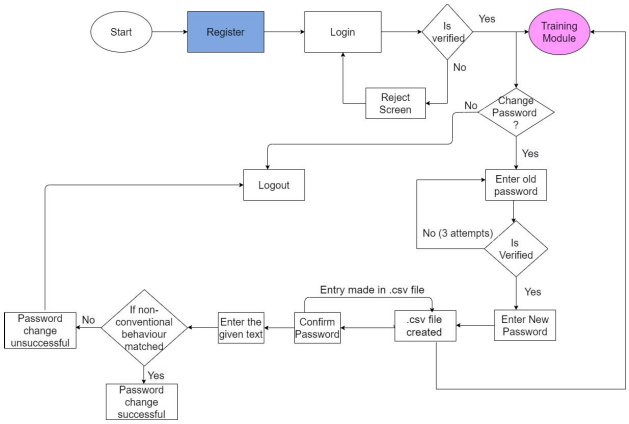

Fig. 5. SKDA Model to authenticate the user

In the Static Keystroke Dynamics Authentication(SKDA) Model, every user's key usage and typing timing data is captured. Timing features such as hold time and flight/latency time are calculated for every user from the captured timing features. Non-conventional features such as usage of Number Key (either the top section of the keyboard or right section of the keyboard used), use of Shift Key (either right side or left side), and CapsLock Key are then stored and an individual pattern is evolved from the captured data.

As mentioned in the Section III-B, when the user changes the password, the required timing features are not available

and hence it becomes difficult to comment on the authenticity of the user based on keyboard usage. The proposed non-conventional features discussed in section IV may come to the rescue till the timing-based model gets built.

The proposed SKDA Model to authenticate the user at the time of change-of-password using non-conventional features is shown in Fig. 5. In this proposed model, the user has to first register himself/herself to the system with the default password. At the time of registration, the key press and release time of the keys typed by the user are captured. After successful registration, the user has to log in once, if the username and password are matched the user password typing keystroke timing data is sent to training. If the verification fails then the user is rejected and asked to log in again.

A. Change Password Use Case

User's non-conventional keystroke keystroke profile is already created as discussed in the earlier sections. Now when the user enters new password, the user will be asked to enter any randomly generated text(which is framed using the policy as discussed in Section IV-C). After entering the text, if the non-conventional features are matched with the new password non-conventional features to authenticate the user. This newly created file of the new password is now sent to the training phase for capturing timing features for the new password.

B. Data Collection Module

For proper functioning of the model, the user needs to enter the username and the default password three times, which the user can change after successful training. The keystroke information of the keys typed for a password by the user is captured. These three entries are deemed authentic. If the first three entries are not similar to ideal, then it will impact the entire dataset. Entries are stored after the third entry only after the consistency check is completed. The consistency check is done using the k-nearest neighbor algorithm to ensure that outliers don't reach the data. In consistency check, a genuine sample of training is considered only if the nearest k^{th} distance from the upcoming training dataset is less than the average distance value. After the fifth entry, the data is sent to the training module (refer Section V-C).

C. Training Module

Every user's data file is provided as an input to the training model. The user data entries obtained at the time of registration and login after consistency check completion as described in Section V-B are considered for training, where n/4 fake entries are added to n data entries for each user. as shown in Fig. 7.

D. Experimentation

The proposed SKDA model was implemented using Python programming language with the Django framework and Javascript, jQuery, and Bootstrap web technologies for front end designing. Each user registered himself/herself using the

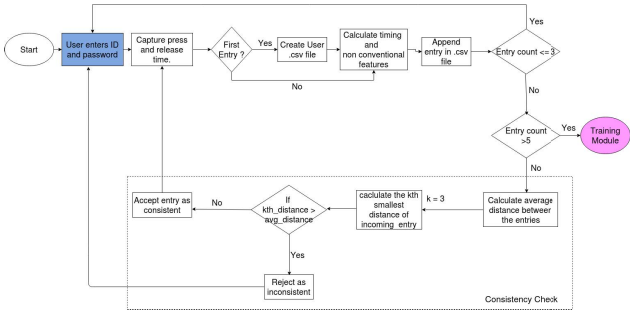

Fig. 6. Data Collection Module

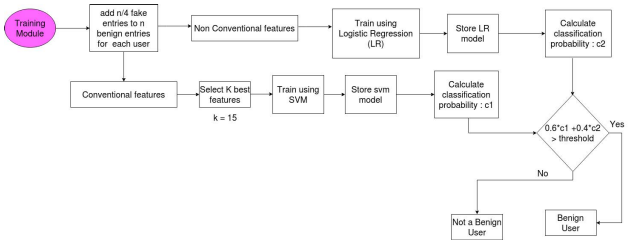

Fig. 7. Training Module

default password "India@2018". After the completion of the training phase, the user can change the password if required as described in Section V.

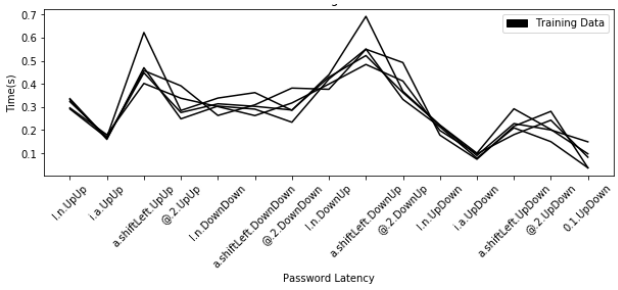

Fig. 8. Genuine user trained dataset of 15 contributing features

It was also observed from Fig. 9 and 10 that a change in the typing pattern was seen when the user uses non-conventional features. The proposed model was tested on 155 users where each user's typing rhythm for the default password and while changing the password was tested, the results of which are discussed in Section VI.

VI. Results

The proposed SKDA model was evaluated using performance metrics False Acceptance Rate (FAR) and False Rejection Rate (FRR). Our main focus was to use non-conventional features in Static Keystroke Dynamics as discussed in section IV and to observe its impact on detecting genuine and fake users while credential checks during login and also when the

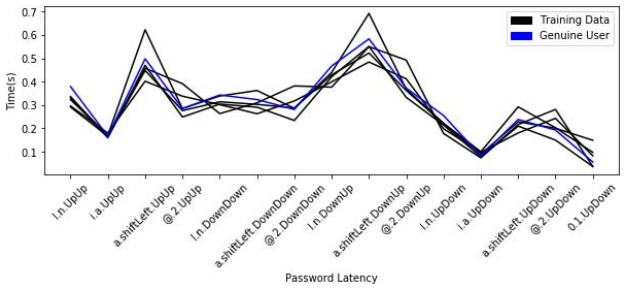

Fig. 9. Login Attempt of Genuine user variations along with its trained dataset of 15 contributing features

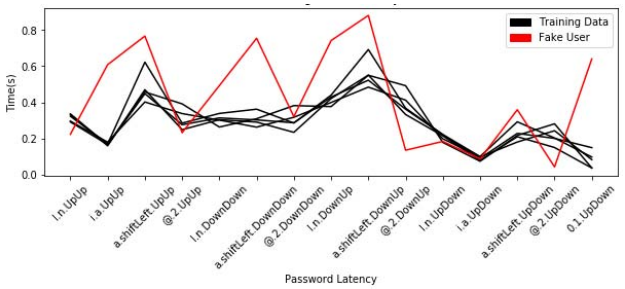

Fig. 10. Login Attempt of fake user variations along with the genuine user trained dataset

user wants to change the login credential. The proposed model was also evaluated on the default password typing rhythm and change password typing rhythm of the user, the result of which is shown in Table III.

TABLE III
DEFAULT AND CHANGED PASSWORD NON-CONVENTIONAL FEATURES PERFORMANCE METRICS

Performance Parameters	Default Password Non-conventional features	Change Password Non-conventional features
FAR	0.22	0.21
FRR	0.05	0.04
ERR	0.13	0.12

When compared our results with other studies as shown in Table IV, the no. of characters used by the SKDA model is very short (10-15 characters). Our SKDA model considered both timing and non-conventional features on a very less number of characters (10-15) which is generally considered to be the length of the password i.e static keystroke dynamics on the very short text. We were successful in achieving FAR of about 0.02% and FRR of about 0.3% as shown in Table IV.

VII. CONCLUSION AND FUTURE SCOPE

In static keystroke dynamics, the user is authenticated based on the model generated from the timing features captured while entering the login and password. However, during the change of password, no data related to key press and release

TABLE IV
RESULTS OF SKDA MODEL

Study	No. of Participants	No. of Characters used	Features	FAR	FRR
Arwa Alsultan et al. [21]	30	1000	Non-conventional features both	0.0011	0.28
Kathryn Hempstalk et al. [22]	10	10800-30000	Non-conventional features	0.113	0.331
Blaine Ayotte et al. [23]	-	-	Timing Features	0.029	0.490
Ahmed A. et al. [24]	53	11000	Timing features	0.0152	4.82
Saira Zahid et al. [25]	25	12500	Timing features	0.292	0.308
A. Alsultan et al. [20]	25	7200	Timing and Non-conventional features both	0.009	0.215
SKDA Model	155	10-15	Timing and Non-conventional features both	0.02	0.3

for the new password is available. This results in, the model available for static keystroke dynamics for the new password of the user. In this paper the use of non-conventional features for change of password is proposed and demonstrated. The paper also discusses the policies which can be used to capture the non-conventional features such as usage of CapsLock key, NumLock key, Shift keys and key release pattern during use of shift keys. The SKDA model uses non-conventional features with timing features for user authentication. The notion was tested through an experiment which was carried out and was tested on 155 participants, and have achieved FAR of about 0.02% and FRR of about 0.3%.

Future work includes working on various types of keyboard and identifying keystroke dynamics with respect to different devices.

REFERENCES

[1] R. Joyce and G. Gupta, "Identity authentication based on keystroke latencies," in *Communications of the ACM, 33(2), 168-176*, 1990.
[2] S. Mondal and P. Bours, "Continuous authentication in a real world settings," in *Advances in Pattern Recognition, Eighth International Conference, pp. 1-6*, 2015.
[3] F. Monrose and A. D. Rubin, "Keystroke dynamics as a biometric for authentication," in *Future Generation computer systems, 16(4), 351-359, Elsevier Science*, 2000.
[4] R. Abinaya and A. Sigappi, "Biometric identification of a genuine user/imposter from keystroke dynamics dataset," in *International Journal of ChemTech Research, Vol.11 No.08, pp 147-160*, 2018.
[5] A. Andrey, Vyazigin, Y. Nadezhda, Tupikina, and V. E. Sypin, "Software tool for determining of the keystroke dynamics parameters of personal computer user," in *International Conference on Micro/Nanotechnologies and electron devices EDM,978-1-7281-1753-9/19/$31.00, IEEE*, 2019.
[6] P. H. Pisani and A. C. Lorena, "A systematic review on keystroke dynamics," in *Journal of the Brazilian Computer Society*, 2013.
[7] M. L. Ali, J. V. Monaco, C. C. Tappert, and M. Qiu, "Keystroke biometric systems for user authentication," in *Journal of Signal Processing Systems,DOI: 10.1007/s11265-016-1114-9, Springer*, 2016.

[8] K. Shekhawat and D. P. Bhatt, "Recent advances and applications of keystroke dynamics," in *International Conference on Computational Intelligence and Knowledge Economy (ICCIKE),978-1-7281-3778-0/19/$31.00, IEEE*, 2019.
[9] D. R. Gentner *et al.*, "A glossary of terms including a classification of typing errors," in *in Cognitive aspects of skilled typewriting, ed: Springer, pp. 39-43*, 1983.
[10] J. Roth, X. Liu, A. Ross, and D. Metaxas, "Investigating the discriminative power of keystroke sound," in *IEEE Transactions on Information Forensics and Security, vol. 10, pp. 333-345*, 2015.
[11] Y. Zhong and Y. Deng, "A survey on keystroke dynamics biometrics: approaches, advances, and evaluations," in *Recent Advances in User Authentication Using Keystroke Dynamics Biometrics. Science Gate Publishing, pp. 1-22*, 2015.
[12] P. S. Dowland and S. M. Fumell, "A long-term trial of keystroke profiling using digraph, trigraph and keyword latencies," in *Security and Protection in Information Processing Systems, ed: Springer, pp. 275-289*, 2004.
[13] P. Lozhnikov, E. Buraya, A. Sulavko, and A. Eremenko, "Methods of generating key sequences based on keystroke dynamics," in *Dynamics of Systems, Mechanisms and Machines (Dynamics) IEEE, DOI: 10.1109/Dynamics.2016.7819038*, 2016.
[14] A. Alsultan and K. Warwick, "User-friendly free-text keystroke dynamics authentication for practical applications," in *International Conference on Systems, Man, and Cybernetics (IEEE), DOI: 10.1109/SMC.2013.793*, 2013.
[15] K. S. Killourhy and R. A. Maxion, "Comparing anomaly-detection algorithms for keystroke dynamics," in *Proc. IEEE/IFIP International Conference on Dependable Systems Networks (DSN), 125–134*, 2009.
[16] R. Nataasha, R. Shankarmani, and P. Joshi, "A comprehensive review of keystroke dynamics-based authentication mechanism," in *Proceedings of Advances in Intelligent Systems and Computing, vol 1059. Springer, DOI: https://doi.org/10.1007/978-981-15-0324-5_ 13*, 2019.
[17] P. S. Teh, A. B. J. Teoh, C. Tee, and T. S. Ong, "A multiple layer fusion approach on keystroke dynamics," in *Pattern Analysis and Applications, 14:23–36, DOI 10.1007/s10044-009-0167-9*, 2011.
[18] A. A. Ahmed and I. Traore, "Biometric recognition based on free-text keystroke dynamics," in *IEEE TRANSACTIONS ON CYBERNETICS, VOL. 44, NO. 4, APRIL*, 2014.
[19] R. Nataasha, R. Shankarmani, and P. Joshi, "Non-conventional factors for keystroke dynamics as a support factor for authenticating user," in *International Journal of Innovative Technology and Exploring Engineering (IJITEE), ISSN: 2278-3075, Volume-9 Issue-4, DOI:10.35940/ijitee.D1194.029420*, 2020.
[20] A. Alsultan, K. Warwick, and H. Wei, "Improving the performance of free-text keystroke dynamics authentication by fusion," in *Applied Soft Computing*, 2017.
[21] A. Arwa, W. Kevin, and W. Hong, "Non-conventional keystroke dynamics for user authentication," in *Pattern Recognition Letters, vol. 89, pp. 53-59*, 2017.
[22] I. H. W. K. Hempstalk, E. Frank, "One-class classification by combining density and class probability estimation," in *The European Conference on Machine and Learning and Principles and Practice of Knowledge Discovery in Database, 505–519*, 2005.
[23] D. H. S. S. Blaine Ayotte, Mahesh K. Banavar, "Fast and accurate continuous user authentication by fusion of instance-based, free-text keystroke dynamics," in *International Conference of the Biometrics Special Interest Group (BIOSIG),978-1-7281-2677-7, IEEE*, 2019.
[24] I. T. A.A. Ahmed, "Biometric recognition based on free-text keystroke dynamics," in *IEEE Trans. Cybern. 44 458–472*, 2014.
[25] S. K. M. F. S. Zahid, M. Shahzad, "Keystroke-based user identification on smart phones," in *The 12th International Symposium on Recent Advances in Intrusion Detection (RAID '09) 224–243*, 2009.