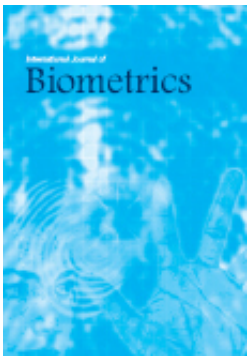


International Journal of Biometrics

ISSN online: 1755-831X - ISSN print: 1755-8301
https://www.inderscience.com/ijbm

Feature ranking for effective continuous user authentication using keystroke and mouse dynamics with the cat recurrent neural model

Princy Ann Thomas, Preetha Mathew Keerikkattil

DOI: 10.1504/IJBM.2024.10064403

Article History:

Received:	17 October 2023
Last revised:	15 January 2024
Accepted:	07 March 2024
Published online:	30 April 2025

Feature ranking for effective continuous user authentication using keystroke and mouse dynamics with the cat recurrent neural model

Princy Ann Thomas*

Department of CSE,
Government Engineering College,
Thrissur, Kerala, India
Email: princy@gecidukki.ac.in
*Corresponding author

Preetha Mathew Keerikkattil

Department of CSE,
Cochin University College of Engineering,
Kuttanad, CUSAT, Kerala, India
Email: preetha.mathew.k@gmail.com

Abstract: Behavioural biometric modalities such as keystroke and mouse dynamics are ideal for continuous user authentication due to their non-intrusive quality. The success of the authentication framework is largely determined by the discriminative power of the features used. It is critical to be able to select the necessary discriminative features for optimal authentication performance. In this research, we implement multiple ranking algorithms on features derived from temporal information of keystroke and mouse dynamics to distinguish their discriminative capacity. The ranked features are then employed for continuous authentication using the cat recurrent neural model (CRNM) to optimise the search space and authenticate users. The experimental results given in this work propose a strategy for developing commercially deployable continuous authentication systems with broad applicability. Experiments are carried out with filter, wrapper, and embedded feature ranking approaches, and authentication outcomes are compared with the CRNM framework. The findings indicate that discrimination is manifested in uncommon rather than normal user conduct. Furthermore, it is discovered that applying feature ranking reduces authentication time from 198 seconds to 138 seconds and improves accuracy from 98.25% to 99.21%.

Keywords: ranking; temporal features; keystroke dynamics; mouse dynamics; cat swarm optimisation; recurrent neural model.

Reference to this paper should be made as follows: Thomas, P.A. and Keerikkattil, P.M. (2025) 'Feature ranking for effective continuous user authentication using keystroke and mouse dynamics with the cat recurrent neural model', *Int. J. Biometrics*, Vol. 17, No. 3, pp.227–251.

Biographical notes: Princy Ann Thomas has obtained her PhD from Cochin University College of Engineering Kuttanadu, Kerala, India and MTech in Cyber Security with the first rank from Mahatma Gandhi University, Kottayam, Kerala, India. Currently, she is an Assistant Professor at Government

Engineering College Thrissur Kerala and joined government service in September 2000. Her research interests include cyber security, machine learning, and data mining.

Preetha Mathew Keerikkattil has received her PhD from the Indian Institute of Technology, Madras. Her areas of interest are cryptography and network security. She has around 30 years of teaching experience and is currently a Professor in Computer Science and Engineering, Cochin University College of Engineering Kuttanadu, Kerala.

1 Introduction

There has been an increase in the knowledge and use of security measures for the prevention of various types of data breaches in recent years. Sensitive data of any kind, whether personal, financial, or business-related, must be well protected to deter criminal activity (Ding et al., 2021). User authentication, most commonly in the form of login id and password verification, is a mechanism widely used to assure security. This technique has various disadvantages (Zimmermann and Gerber, 2020), one of which is that it only supports static authentication; once verified, the system assumes that the entire session will be conducted by the verified entity. Continuous user authentication enables continuous monitoring, making it more efficient in preventing crimes (Stylios et al., 2021).

Continuous user authentication procedures authenticate the identification of the logged-in entity on an ongoing basis. This enables the detection and prevention of illegal activities such as hacking, Malware, bots, and others (Wang et al., 2021). Continuous authentication can be performed using either intrusive or non-intrusive approaches. Non-intrusive approaches are advantageous when it is necessary to detect and prevent illicit activity without interfering with users' usual operations (Chang et al., 2022). Keystroke and mouse dynamics are significant players in the field of continuous non-intrusive user authentication, with the added bonus of not requiring any specific hardware to be installed. Voice recognition, facial recognition, gait recognition, lip movement, hand gesture, and other non-intrusive user identification methods are also available (Earl et al., 2021).

Soft biometric features for authentication are provided by both keyboard and mouse movements. Keystroke dynamics simulates a user's behaviour when typing on a keyboard, whereas mouse dynamics simulates a user's conduct when using a mouse or touchpad. The data collected for keystroke dynamics (Karnan et al., 2011) consists of keys logged with timing information that provides timing information on key hold, key delay, monograms, di-grams, and n-grams. Other data is based on key press pressure, finger area on key and incorrect keystrokes (Salman and Hameed, 2018). Mouse movement information, including mouse clicks and mouse action information, is among the data recorded for mouse dynamics. Furthermore, mouse dynamics such as movement direction, speed, and trajectories have been utilised.

Despite the fact that there have been numerous studies on continuous user authentication utilising keystroke and mouse dynamics (Shen et al., 2013, 2014, 2016; Belman and Phoha, 2020), there have been essentially no studies on feature ranking, or scoring in order to increase the effectiveness of authentication methods. Because each

user authentication system is essentially a prediction model, the curse of dimensionality must be addressed (Aremu et al., 2020). Dimensionality reduction lowers computing costs and enhances model performance. It is accomplished through either feature selection or feature extraction. Feature selection (Zebari et al., 2020) produces a subset of useful features, whereas feature extraction (Li et al., 2017) attempts to map the original data onto a lower dimension feature space. Both are useful in developing more generalised models. When the raw data does not directly contribute to the input feature set for a certain algorithm, feature extraction is chosen. When the original meaning of the characteristics must be preserved, feature selection is preferable.

Feature extraction is classified as linear or nonlinear algorithms, whilst feature selection is classified as supervision-based or strategy-based approaches, each of which is further classified as supervised, unsupervised, wrapper, filter, embedded, or hybrid methods.

In this research, we suggest performing feature ranking on keystroke and mouse dynamics datasets to investigate the impact on continuous user authentication performance. The motivation for this work stems from our observation that, while there are numerous works in the field of keystroke and mouse dynamics for user authentication, there is little work that addresses dimensionality reduction and ranking of discriminative features for user authentication. The findings of this study point to a path for wide commercial deployment in fields such as online learning and testing, age-based authentication, insider threats, and so on.

The major contributions of this work are

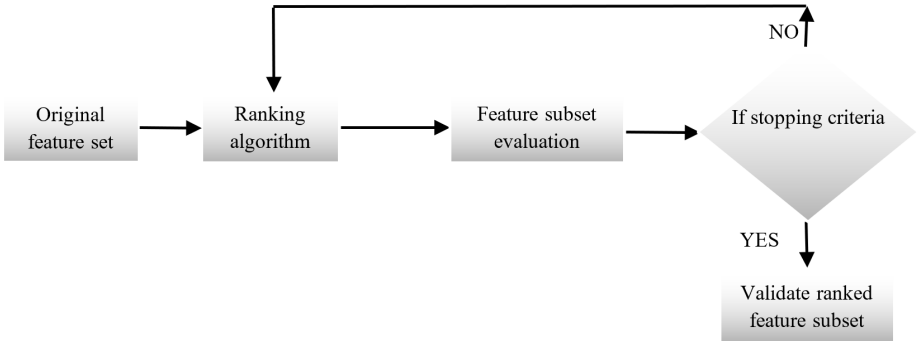
- A set of general guidelines to select discriminative ranked features to decrease both time for data collection and preparation and cost of computation for continuous user authentication.
- Results of feature ranking by filter, wrapper and embedded based methods for both keystroke and mouse data.
- Authentication effectiveness of ranking features utilising the cat recurrent neural model (CRNM) model with enhanced authentication time and accuracy.

It is not the goal of this work to design a new ranking technique but to use existing methods in conjunction with the CRNM framework to enhance performance. The rest of this paper is organised as follows. Section 2 consists of related work followed by background information and experimental set up in Sections 3 and 4 ending with results, discussions and conclusion in Sections 5 and 6.

2 Related work

In this section, we limit our survey to feature ranking. Figure 1 depicts the general approach to feature ranking used. The input dataset containing the original features is initially fed into the ranking algorithm, which produces a subset of the original features based on the selection criteria employed in the specific algorithm. The ranked feature subset is then subjected to several evaluation processes to see whether it is representative of the discriminative features in the original dataset. This method is iterated until a certain stop requirement is met. The feature ranking procedure produces a validated feature subset that is representative of the discriminative features in the original dataset.

Figure 1 Feature ranking process



Typically, features are classified as relevant, weakly relevant, irrelevant, or redundant. The goal of feature ranking is to develop a feature subset with the greatest possible relevance and the least amount of repetition. This set with reduced redundancy is known as a Markov blanket (Ling et al., 2019). Furthermore, stable feature selection is advantageous so that subsets do not need to be produced when the training samples are updated (Utkarsh and Dhanalakshmi, 2022). We now review the existing research on feature selection and feature ranking in keystroke or mouse dynamics.

There are several studies on feature selection. Filter, wrapper, embedded, and hybrid approaches are the most common types of feature selection algorithms. Jović et al. (2015) and Chandrashekar and Sahin (2014) investigate these categories and discover that wrapper approaches outperform filter methods because they evaluate feature subsets in relation to learning models. In many instances, hybrid methods outperform both filter and wrapper approaches because they integrate the benefits of both. Several studies on feature selection for machine learning applications have been conducted. Wang et al. (2016), Cai et al. (2018), and Khalid et al. (2014) investigate different feature selection techniques and find that the primary obstacles in feature selection include imbalanced classes, stability, dynamic feature space, parameter selection, and computation time.

Feature selection is examined from both the feature and data perspectives (Thakkar and Lohiya, 2023; Dai et al., 2023). The advantages and disadvantages of particular methods in specific application fields are also touched upon. Miao and Niu (2016) conduct the same research in an unsupervised learning context. According to the comparison study, adaptive hyper parameter setup is a challenge that influences the performance of feature selection algorithms. Sutha and Tamilselvi (2015), on the other hand, investigate feature selection for data mining and conclude that hybrid methods produce the best results. Other approaches include feature selection (Alsahaf et al., 2022) based on fast scalable tree boosting to reduce redundancy, and various nature inspired algorithms (Baynath and Khan, 2023; Abu Khurma et al., 2022) for feature selection have lately gained favour.

There have been few studies on feature selection and ranking for user authentication using keystroke and mouse dynamics. Instead of handling feature selection and ranking independently, hybrid approaches or wrapper methods are most commonly used. In the case of keystroke dynamics, we are dealing with fixed text input or free text input features. All of the present work on feature selection and ranking for keystroke dynamics

is focused on fixed text. Although not attempted, free text is more relevant for ongoing user authentication. In the case of mouse dynamics

For ranking, El Zein and Kalakech (2018) present filter-based strategies that use Chi square, information gain, and gain ratio. The data show that keystroke hold time ranks first, followed by up-down timing. Kim et al. (2020) compare authentication with and without feature selection, finding that filter-based feature selection improves authentication by 21.8%. Despite the fact that a ranking scheme is described, no ranks of specific attributes are provided.

Several works on wrapper-based feature selection have been published; all of them combine optimisation techniques such as particle swarm optimisation, genetic algorithms, greedy algorithms, artificial bee colony algorithms, and so on with machine learning algorithms. Muthuramalingam et al. (2018) employ the firefly algorithm to improve feature selection, although the selected feature subset and rank are not provided. Shanmugapriya and Ganapathi (2011) combine ant colony optimisation with an extreme learning machine to achieve better outcomes than previous optimisation approaches. Once again, no selected subgroups or ranks are provided. Darabseh and Namin (2015), on the other hand, combine breadth first search with k nearest neighbour to achieve 62.25% dimensionality reduction and 92.85% accuracy, however the feature subset and rank are not given.

Finally, two hybrid methods are presented: Hameed et al. (2014) principal component analysis (PCA) with neural network (NN) to authenticate user with significant increase in performance as shown by FRR 24% and FAR 6% and Mohamed and Moftah (2018) binary bat optimisation, where the former does not present feature ranking but the latter confirms previous research with hold time having highest rank followed by up-down keystroke timing.

In the case of mouse dynamics feature ranking, all the work that we could identify was dependent on free mouse movement. Shen et al. (2009, 2011) identify and rank feature subsets based on interaction and physiological features using hybrid approaches. Single click time intervals provided the highest accuracy of 97.83% in their testing. They then add user behaviour variability over time to the mix, addressing the problem with a mixture of PCA and ISOMAP. Yamauchi (2013) employ random forest (RF) to automate feature selection that aids in detecting a user's anxiety state using mouse trajectory data.

As can be observed, there is a dearth of information in the field of feature ranking for continuous user authentication using the dynamics of both keystroke and mouse. Any ranking is based on static data rather than dynamic data, which is essential for continuous user verification. Because of this, our approach is unprecedented and novel in addressing the aforementioned gap in the literature. The results of our tests are highly encouraging, and they will help other researchers make informed decisions when deciding on keystroke and mouse dynamics pre-processing procedures for continuous user authentication.

3 Background and methodology

In this part, we provide some background information on keystroke and mouse dynamics, as well as references to the datasets used in this work. We also include brief descriptions of the feature ranking techniques used in the trials, as well as parameter values for our experiments. All experiments use the 10 folds cross validation for testing. The ratio of

training set to testing set is 7:3 to decrease the possibility of statistical errors in the performance calculations. In addition, different sets of experiments were conducted taking a different user as the authentic user at a time.

3.1 Keystroke dynamics

Keystroke dynamics reflect the user's behavioural traits when typing on a digital device's keyboard. The raw data that is often obtained is time information. Other known to be used properties (Abdrabou et al., 2022) are key press prediction, touch area, and a few more. Timing information is solely used in this case because no special hardware is required to acquire timing information. When a person types on the keyboard, the <k, k-act, t_k> triplet is captured, where k is the key, k-act characterises the key action as either key press or key release, and t_k represents the action's time information. We compute the additional features from the raw data.

Press-press (pp), press-release (pr), release-release (rr), and release-press (rp) are the basic properties that can be computed from the timing information of the keys. The latency between two monogram key presses is given by pp_m, the hold time of a monogram is given by pr_m, the release time between successive monograms is given by rr_m, and the time between the release of a monogram and the press of the next monogram is given by rp_m. Similarly, pp_d, pr_d, rr_d, rp_d represent di-gram timing information, and pp_w, pr_w, rr_w, rp_w represent word timing information. The raw data can be used to calculate n-gram information also.

Figure 2 Keystroke dynamics feature set

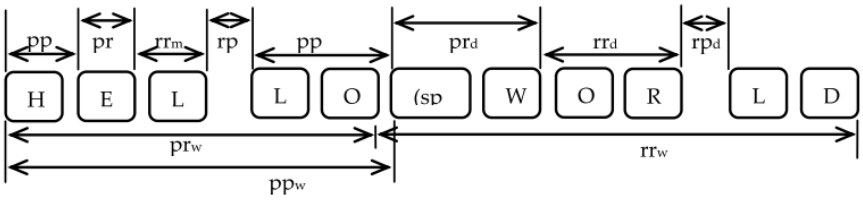


Figure 2 shows an example of the feature set obtained by entering the words 'Hello World,' as well as one of each of the aforementioned data that can be obtained from the typed alphabets. Because we are ranking features primarily for continuous user authentication, the raw data stream is divided into 3 minute periods and calculates features accessible from the 3 minute window to determine a user's validity. As a result, depending on what the user types in the specific window, each window may employ a different set of input features in the pre-processing and authentication phases.

The hypothesis was that the highest scores will correspond to the most infrequently used English language monograms, di-grams, trigrams, and words as described originally by Samuel Morse when delivering codes in the invention of Morse code for telegraph communication transfer. This was supported by the experimental results, and it was also discovered that aggregates of characteristics had a high discriminative value.

3.2 Mouse dynamics

Ahmed and Traore (2007) were the first to incorporate mouse dynamics; the feature set they employed consisted of four mouse actions: move-move (mm), drag and drop (dd), point and click (pc), and no movement or silent (s). <m-act, dist, t_m, d_{mov}> are captured at each instance, where m-act is the mouse action, dist is the distance travelled in pixels, t_m is the timing information, and d_{mov} is the direction of movement. The direction was shown in numbers ranging from 1 to 8, such that 0 to 45 degrees from a point was given one, 46 to 90 degrees was given two, and so on to represent the direction of movement around a point up to 360 degrees covering the region around a point. Average speed in terms of distance travelled and movement direction, as well as average speed per mouse action, were calculated.

This core feature set has been kept, although others (Sayed et al., 2013) such as horizontal velocity, vertical velocity, and so on have been added. In this work, we leverage the three-minute window presented earlier to model user activity in a continuous authentication scenario. The feature set used in this work is explained in the next section.

3.3 Datasets used

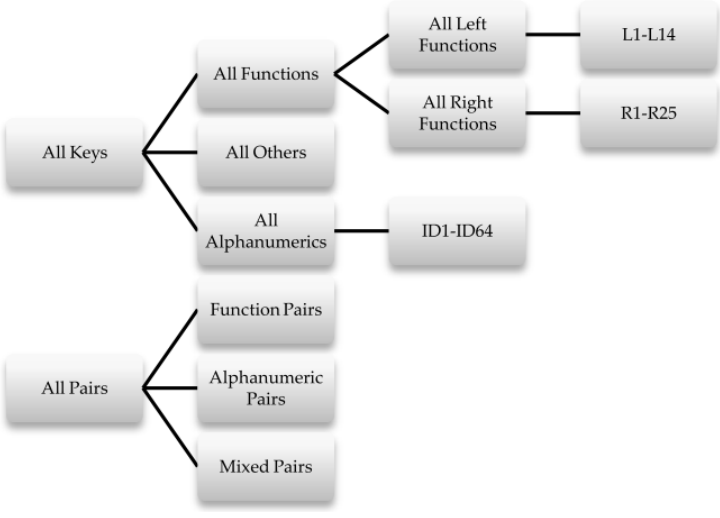
Three datasets are used for this research. The first is the Mendeley dataset (Wesolowski et al., 2020), which contains the free text typing details of 150 individuals, each with 100 sessions. The second dataset is the Balabit mouse dynamics challenge dataset (Fülöp et al., 2016), which contains mouse data from ten users with seven sessions each. The dataset separates each user's training and testing sets and contains labels to indicate legal and unauthorised sessions. The third dataset is bespoke gathered by utilising a simple java programme that runs in the background and collects all typing and mouse movement data after a user begins a session. We have currently collected keystroke and mouse dynamics data from 15 people over the course of 5 half-hour sessions on different days.

The Mendeley dataset (Wesołowski et al., 2016) has 113 attributes, as seen in Figure 3. The key layout is based on a traditional QWERTY keyboard. Esc and F1 are the left function keys, while F7, Tab, Caps lock, Left shift, Left ctrl, Windows, and Left alt labelled L1 to L14 are the right function keys. Scroll lock, pause, insert, delete, home, end, page up, page down, numlock, backspace, enter, right shift, right control, context, right alt, R1 to R25 arrows (up, down, left, right). Alphanumeric keys include all capital and small letter alphabets, as well as the numbers on the keyboard labelled ID1 to ID64. ID63 and ID64 are not associated with any keys. The all others function is assigned to all special keys. Furthermore, digraph information for pairs of consecutive function keys, pairs of consecutive alphanumeric keys, and pairs of function plus alphanumeric mixed keys is provided, along with their total timing.

Each user session's mouse information in the Balabit mouse challenge dataset comprises six attributes: server timestamp, client timestamp, mouse button pushed, mouse action, and mouse cursor coordinates. Session information is also provided to distinguish between authorised and unauthorised sessions. We begin by categorising the data and then take 3 minute data windows to compute the timing information for the 27 mouse actions given in Table 1. Move, point, click, drag, and drop are the five basic mouse actions. The characteristics are the time information of a combination of the user's conceivable mouse actions.

A custom dataset that we created is also used. It includes both keystroke and mouse information from users. Keystroke characteristics include key action code, key action, keyboard area, key ASCII code, keyboard scan code, key and key action time.

Figure 3 Mendeley free text dataset attributes



Mouse actions, mouse action codes, mouse pointer coordinates <x, y>, and mouse action time are all captured. This data is used to calculate all keystroke and mouse dynamics aspects. This dataset was specifically gathered in order to obtain a more accurate depiction of the feature ranking tests performed here. It is not used in the section on feature extraction. We separated the keyboard and mouse data for easier comprehension and application.

Table 1 Mouse features computed from the timing information

move-move (m-m)	point-move (p-m)	double click-move-point (d-m-p)
move-single click (m-s)	move-single click-move (m-s-m)	move-point-single click (m-p-s)
move-double click (m-d)	move-double click-move (m-d-m)	move-point-double click (m-p-d)
single click-move (s-m)	single click-move-single click (s-m-s)	average-move (avg-m)
double click-move (d-m)	double click-move-single click (d-m-s)	average-point click (avg-pc)
point-single click (p-s)	single click-move-double click (s-m-d)	average-single click (avg-sc)
point-double click (p-d)	double click-move-double click (d-m-d)	average-double click (avg-dc)
move-point (m-p)	single click-move-point (s-m-p)	average-click (avg-c)
drag-drop (dr-dp)	point-drag-drop (p-dr-dp)	average-drag drop (avg-drdp)

3.4 Ranking algorithms used

Feature ranking removes features that are unimportant to the performance of a learning model. Filter-based, wrapper-based, and embedded feature selection algorithms are the three types. Unlike feature extraction, after using the feature selection process, we acquire a subset of the original feature set, allowing us to rank or score the features depending on the feature selection criteria. Depending on the criteria employed, different algorithms produce slightly different ranking outcomes. We present the feature ranking methods used in this section.

The findings will be highly valuable for researchers working on continuous user authentication using keystroke and mouse dynamics, leading to the design of commercially feasible non-intrusive continuous authentication systems. To display the ranking of keystroke and mouse features, keystroke and mouse features were rated independently. The top 50 keystroke features and the top 13 mouse features are then used to test the hypothesis of performance enhancement with our CRNM model.

3.4.1 Filter based feature ranking

Filter-based approaches are statistical measures that are used to determine the relationship between input features and classes or between features.

- *Variance threshold method:* The greater the variance in a feature's values, the more probable it is to give more information to the development of a solid learning model. As a result, we eliminate the attributes with the lowest variance. Both the keystroke dataset and the mouse dataset contain a number of keys or mouse actions that will not be used throughout the three minutes of learning time. So we simply delete features with zero variance, leaving just those that reflect actual key and mouse usage by a user in a given session. This strategy is employed in all of the ranking methods since we want to rank features based on variance as well as decrease dimensionality. Variance of each feature is computed using (1) where n is the number of instances in the dataset.

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2 \tag{1}$$

- *Pearson correlation:* It is a bivariate correlation coefficient. We delete the features that have the highest association with each other and keep only one of the set. For feature selection, we keep all variables with correlation levels less than 0.8. To determine which attributes are least associated, we repeat the process by adjusting the threshold from 0.9 to 0.0. The greatest rank is given to features that have the least association with other features.

- *Information gain:* The best features are chosen by information gain based on mutual information between the feature and the target class. The features with the largest reliance on the class have the highest scores. The information gain of each feature and target class is calculated by (2) where D is the dataset and f is a feature variable in the dataset, $H(D)$ is the information entropy of the dataset computed by (3)where p_i is the probability and $H(D|f)$ is conditional entropy of D given variable f given by equation (4).

$$IG(D, f) = H(D) - H(D \mid f) \tag{2}$$

$$H(D) = -\sum_i p_i \log_2 (p_i) \tag{3}$$

$$H(D \mid f) = \sum_i P(f = v) H(D \mid f = v) \tag{4}$$

- *Chi-square test:* Chi-square tests the independence of two variables by using the expected and observed values of the variables. Chi value of each feature is calculated by (5) where O is the observed value, E is the expected value and n is the number features in the dataset D. The feature with the highest Chi value is given the highest rank.

$$\aleph_{df}^2 = \sum_{i=1}^{n} \frac{(O_i E_i)^2}{E_i} \tag{5}$$

- *Relief:* Relief algorithm estimates the quality of a feature based on closeness to different instances based on dependencies between attributes. For each attribute A of the instance update weight using (6). From m instances select random instance R_i and find nearest hit H and nearest miss M repeating the process for all instances. Sort attributes based on W in descending order for ranking.

$$W[A] = W[A] - diff(A, R_i, H)/m + diff(A, R_i, M)/m \tag{6}$$

3.4.2 Wrapper based feature ranking

Wrapper methods are used to determine the best combination of features for a classification model. Forward selection, backward elimination, and bi-directional selection with the k closest neighbour algorithm (knn) is used here. Cross validation of 10 fold is applied to all algorithms.

- *Forward selection:* Forward selection begins with an empty feature set and adds the best feature to the feature set after each iteration. Here we use the k-nearest neighbour algorithm for the sequential feature selection process. Highest rank is given to the best feature.
- *Backward elimination:* It begins with the complete feature set and eliminates the worst feature from the feature set after each iteration. The k-nearest neighbour algorithm is used for the sequential feature selection process. Lowest rank is given to the worst feature.
- *Bidirectional feature selection: It* is a combination of both forward selection and backward elimination. In each step one feature is added to the final set and another is eliminated.
- *Recursive feature elimination:* RFE method removes a small subset of the worst features based on dependencies and collinearity. Here the estimator used is linear regression. Using linear regression, find the subset of the worst feature based on the target class then eliminate them from the feature set. Repeat this till all features are eliminated. Lowest rank is given to the first feature subset and highest to the last to be eliminated.

3.4.3 Embedded feature ranking

Embedded methods work with learning models to find the best feature set to achieve optimum results.

- *Lasso regression:* The LASSO method decreases the regression coefficients to regularise parameters of the model. Only those features with non-zero coefficient estimate value are selected to be used in the model. Lasso error is given in (7), where Y is the target variable, X are the feature variables, β is the vector representing the variable coefficients and λ controls the penalty term. Eliminate features with coefficient estimates 0 and sort the remaining features in descending order of coefficient estimate for ranking.

$$L = \arg\min_{\beta} \left(\|Y - \beta * X\|^2 + \lambda * \|\beta\|_1 \right) \tag{7}$$

- *Ridge regression:* Similar to lasso regression, ridge regression uses a penalty to estimate coefficients. Ridge regression does not reduce to zero but simply minimises as much as possible. It is L2 regularisation and correlated variables have the same coefficient estimates. Feature selection is done by eliminating negative coefficient estimates. Ridge error is given in (8), where Y is the target variable, X are the feature variables, β is the vector representing the variable coefficients and λ controls the penalty term.

$$L = \arg\min_{\beta} \left(\|-\beta * X\|^2 + \lambda * \|\beta\|_2^2 \right) \tag{8}$$

- *Elastic net:* Elastic net combines the regularisation of both lasso and ridge. It does not easily eliminate the high collinearity coefficients making the model prediction, not too dependent on any particular variable. Elastic net error is given in (9), where y is the target variable, X is the feature variables β is the vector representing the variable coefficients and λ controls the penalty term. $\| \beta \|_1$ is the lasso penalty function and $\| \beta \|^2$ is the ridge penalty function.

$$\beta \equiv \arg\min_{\beta} \left(\|y - X\beta\|^2 + \lambda_2 \|\beta\|^2 + \lambda_1 \|\beta\|_1 \right) \tag{9}$$

- *Random forest:* RF algorithm naturally ranks features based on purity of the node. Gini index is used for this purpose and features with least impurity occur at the start of the tree. Feature selection is attained by pruning nodes above a certain threshold of impurity.

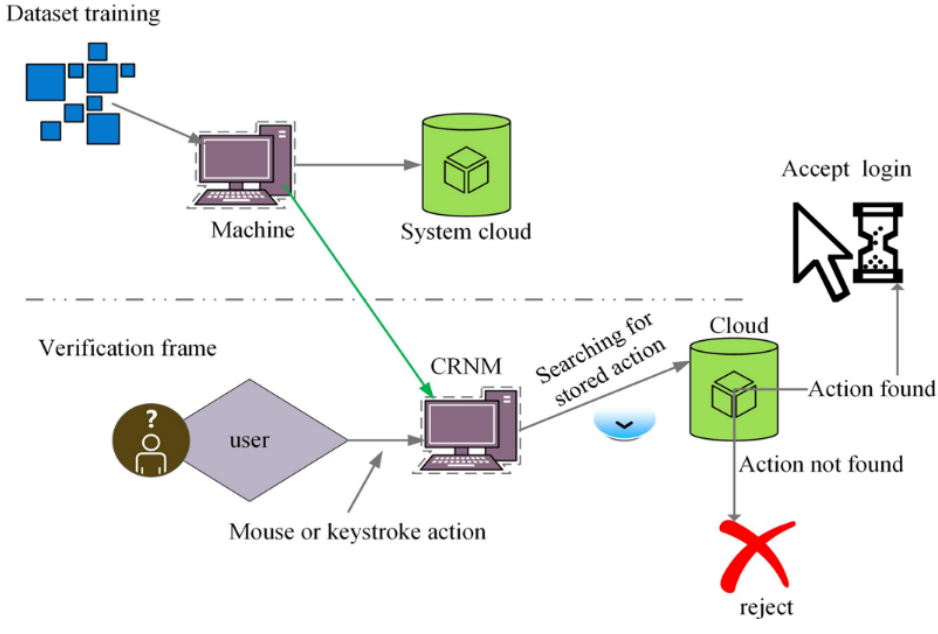
3.4.4 Cat recurrent neural model

We have introduced the CRNM framework in 2021 (Thomas and Mathew, 2022). It employs cat swarm optimisation techniques to find optimal patterns in a limited search space. The suggested framework is run through an optimised deep learning system. To boost detection accuracy, the cat's fitness function is developed in a recurrent dense layer. Figure 4 depicts the process of the suggested architecture and Figure 5 gives the algorithm to authenticate a user. The values r_x, a_x are the mouse data and the keystroke data input in the current session window of the user after feature ranking. The fitness

function is then calculated and based on the previously learned values the user is authenticated.

The trained mouse and keystroke information is initially stored in the cloud. The proposed CRNM identified the proper user during the verification procedure. Furthermore, the developed technique searches the stored action of the dataset in the cloud, and if the user action matches the stored data, the system will log in; if it is not matched, the system will refuse the user. Furthermore, the created framework constantly checks the system.

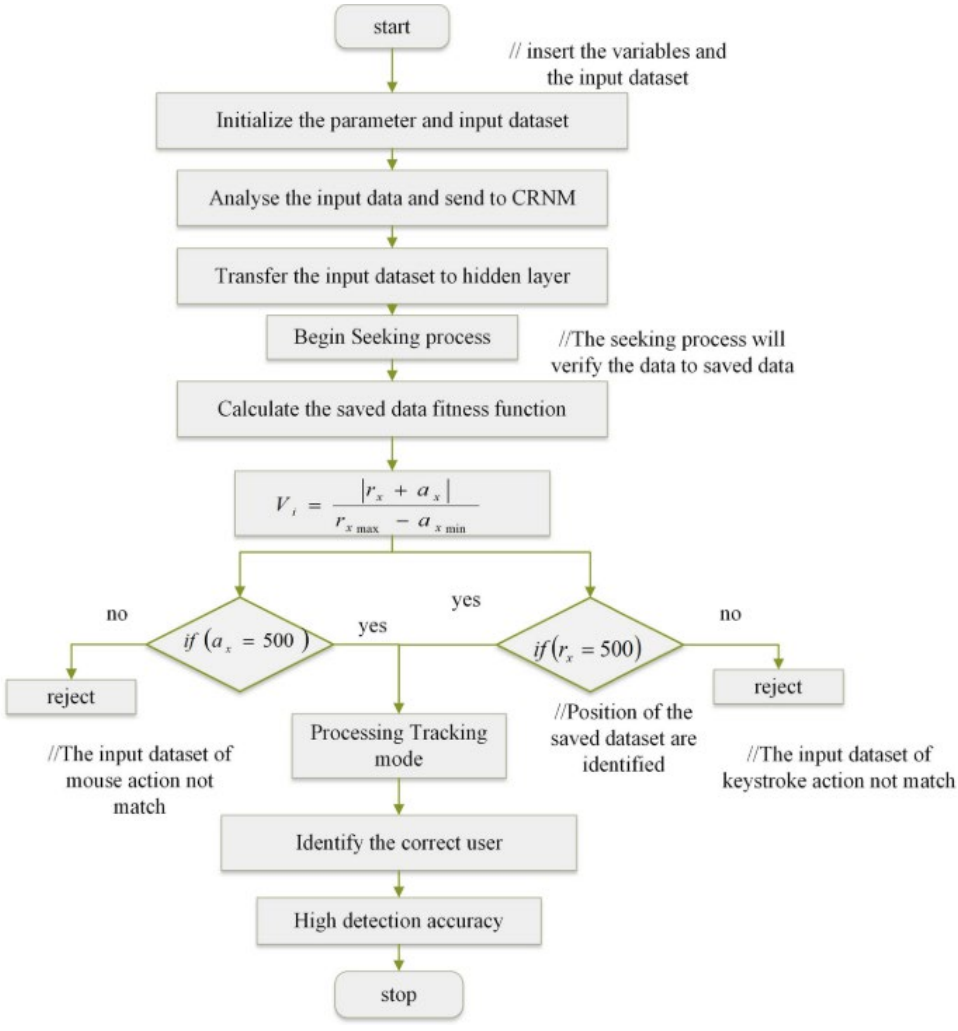
Figure 4 CRNM framework (see online version for colours)



4 Experimental setup

The tests are carried out using the datasets given in Section 3, and features are sorted and presented for each user. Before doing the feature ranking, all features with zero variance were excluded. The number of cross validations is ten, and thresholds or parameters are established for each procedure as stated in Section 3. The overall results are shown by comparing the results for various users, but the individual results are based on the conduct of a single legitimate user. In the case of keystroke and mouse dynamics, the labelled dataset contains 100 occurrences of a genuine user and 50 instances of an impostor user derived by data augmentation when needed.

Figure 5 CRNM authentication (see online version for colours)



5 Results

In this part, we show how each algorithm selects and ranks features. Then, using our CRNM model, we describe the outcomes of employing the ranked features for continuous user authentication. A sample keystroke and mouse output for each approach, as well as feature ranking are shown.

Filter-based, wrapper-based, and embedding approaches have been developed for feature ranking through selection.

5.1 Filter based ranking results

We use the variance threshold, Pearson correlation, information gain, Chi square test, and relief algorithm for filter-based ranking.

- *Variance threshold:* In the case of the variance threshold technique, we calculate the variance of each column and eliminate those with no variance. The threshold in this case is 0, but it can be changed.

Figure 6 Keystroke feature ranking using variance (see online version for colours)

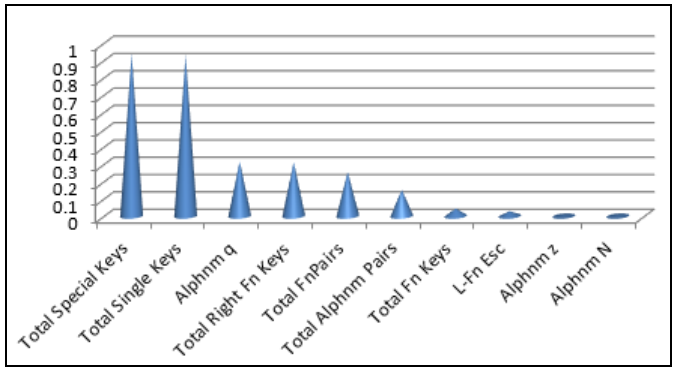
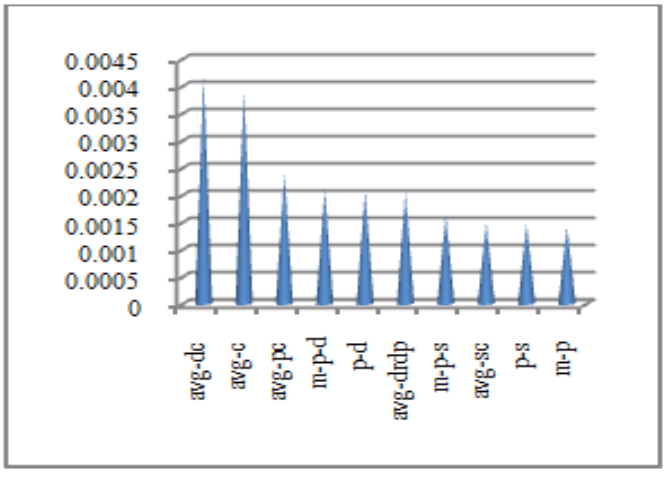


Figure 7 Mouse feature ranking using variance (see online version for colours)



The results for each user varies depending on the keys or actions utilised. As demonstrated in Figure 6, keystroke scoring is unique to each user. Figure 7 depicts mouse scoring. In general, we find that aggregate keystroke and mouse values, as well as unusual keystroke and mouse information, are the most discriminating.

- *Pearson correlation:* In the case of the Pearson correlation approach, we begin by removing any features with zero variance. Then we apply Pearson correlation to feature pairs, keeping those with the lowest correlation and discarding those with a high correlation. To achieve ranking, we gradually lower the threshold from 0.9. The threshold for receiving the top ten best features varies each user. For User 1, at threshold >=0.9, features deleted are (13, specifically {'Alphnm 3','Alphnm M', 'Alphnm q', 'L-Fn CapsLck', 'L-Fn Wnd', 'R-Fn BckSp', 'R-Fn F8', 'R-Fn F9', 'R-Fn Home', 'R-Fn SrcLk', 'Total Alphnm Pairs', 'Total Right Fn Keys', 'Total Special Keys'}). Figure 8, shows the correlation matrix and feature ranks of user 1. Similarly, for User 1, mouse features deleted at threshold>=0.9 are (7, 'avg-c', 'd-m', 'dr-dp', 'm-p', 'm-p-d', 'm-s-m', 'p-d') ranking shown by Figure 9.
- *Information gain:* The best features are chosen by Information Gain based on mutual information between the feature and the target class. The features with the largest reliance on the class have the highest scores. The top 10 ranking for keystroke features were 'Total Alphnm Keys', 'L-Fn F2', 'L-Fn F3', 'L-Fn F7', 'L-Fn CapsLck','L-Fn LShft', 'L-Fn LCtrl', 'L-Fn Wnd', 'R-Fn F8','Total Alphnm Pairs'. Similar ranking for mouse features, with the top 10 being 'avg-pc', 'avg-m','m-s', 'dr-dp','m-m','s-m', 'd-m', 'p-s','m-p', 'p-m'.

Figure 8 Correlation matrix for keystroke features at threshold ≥ 0.6 and ranking of first 10 keystroke features using Pearson correlation (see online version for colours)

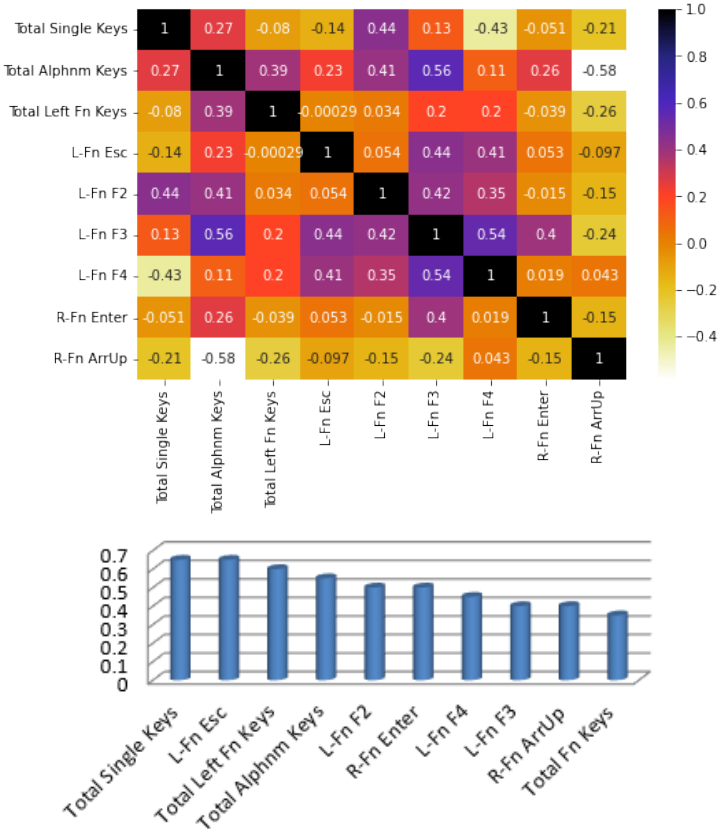
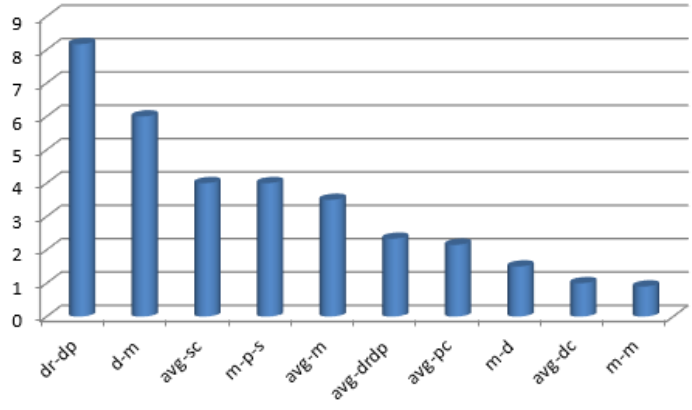


Figure 9 Mouse feature ranking using Pearson correlation (see online version for colours)



- *Chi square test:* In the case of Chi square test the ranking is based on Chi values. Keystroke feature ranking is shown in Figure 10 and Mouse feature ranking in Figure 11. Independent variables have lower Chi values.

Figure 10 Keystroke feature ranking based on chi values (see online version for colours)

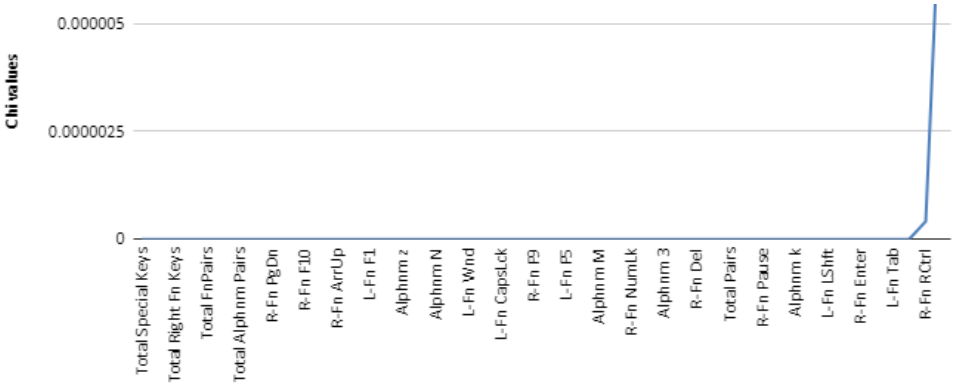
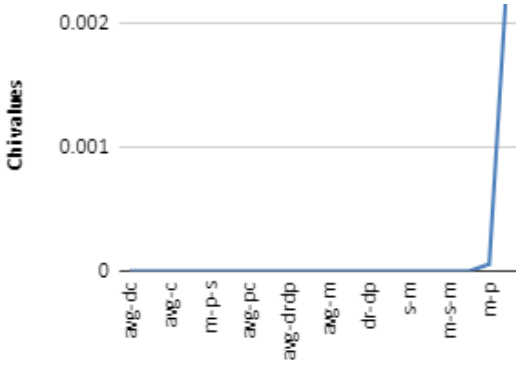


Figure 11 Mouse feature ranking based on chi values (see online version for colours)



- *Relief:* Relief assesses quality of attribute for the learning task. Higher the quality of the input feature set better the performance of the learning algorithms. Figure 12 shows the ranking of keystroke features and Figure 13 shows the mouse feature ranking.

Figure 12 Keystroke feature ranking with relief (see online version for colours)

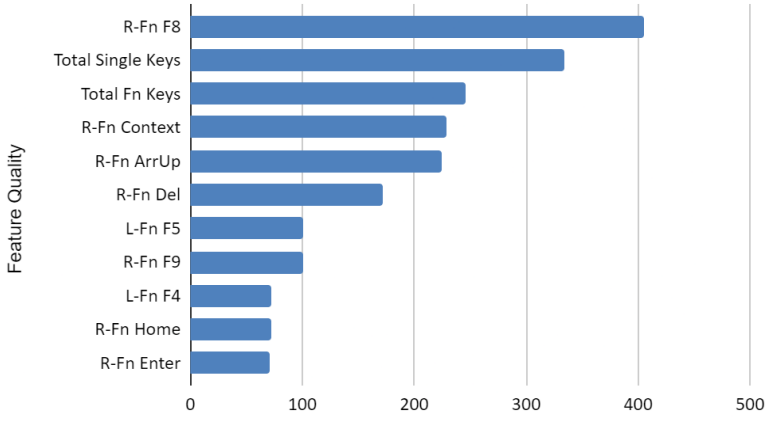
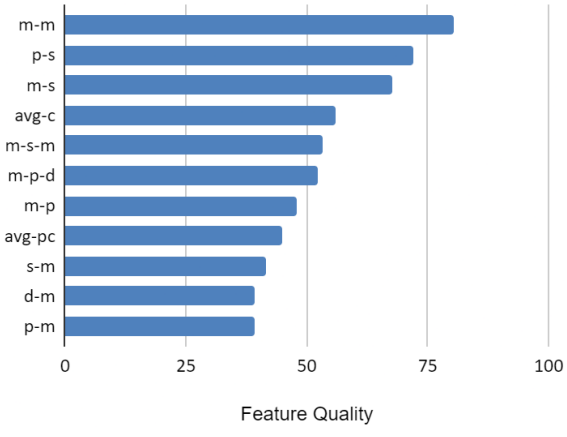


Figure 13 Mouse feature ranking with relief (see online version for colours)



5.2 Wrapper based ranking results

The k-nn algorithm is utilised in the wrapper methods for forward selection, backward elimination, and bi-directional selection. The results for keystroke and mouse feature selection using the k-nn learning method are shown in Table 2.

- *Recursive Feature Elimination (RFE):* When RFE is used, recursive elimination occurs. As indicated in Table 3, the selected features are represented by 1 for each user. The chosen feature set best discriminates against a specific user. The combination varies depending on the user. The ranks of the matching features in the

dataset are presented below, and many features have rank 1 (which is the specified feature set).

Table 2 Ranking of features using k-nn algorithm

Keystroke feature ranking	Forward selection	Total Alphnm Keys, Total Single Keys, Total Fn Keys, Total Left Fn Keys, L-Fn F1, L-Fn F2, L-Fn F3, Total Right Fn Keys, L-Fn F4, L-Fn F5, L-Fn F6, L-Fn F7, Total Special Keys, L-Fn Esc, L-Fn Tab, L-Fn CapsLck, L-Fn LShft, L-Fn LCtrl, L-Fn Wnd, L-Fn LAlt, R-Fn F8, R-Fn F10, R-Fn F12, R-Fn F9, R-Fn PgUp, R-Fn SrcLk, R-Fn PgDn, R-Fn Enter, R-Fn RCtrl, R-Fn Context, R-Fn ArrUp, R-Fn Pause, Alphnm 3, Alphnm N, Alphnm q, Alphnm z, Alphnm k, Total Pairs, Total FnPairs, Total Alphnm Pairs, Total Mixed Pairs, R-Fn BckSp, R-Fn NumLk, Alphnm M, R-Fn F11, R-Fn PrtSc, R-Fn Insert, R-Fn Del, R-Fn Home
	Backward selection	Total Alphnm Keys, Total Single Keys, Total Special Keys, Total Right Fn Keys,L-Fn Esc, Total Left Fn Keys, L-Fn F1, L-Fn F2, L-Fn F3, L-Fn F4, L-Fn F5, L-Fn F6, L-Fn F7, L-Fn Tab, L-Fn CapsLck, L-Fn LShft, L-Fn LCtrl, L-Fn Wnd, L-Fn LAlt, R-Fn F8, R-Fn F9, R-Fn F10, R-Fn F11, R-Fn F12, R-Fn PrtSc, R-Fn SrcLk, R-Fn Pause, R-Fn Insert, R-Fn Del, R-Fn Home, R-Fn End, R-Fn PgUp, R-Fn PgDn, R-Fn NumLk, R-Fn BckSp, R-Fn Enter, R-Fn RCtrl, R-Fn Context, R-Fn ArrUp, Alphnm 3, Alphnm M, Alphnm N, Total Fn Keys, Alphnm q, Alphnm z, Alphnm , Total Pairs, Total FnPairs, Total Alphnm Pairs, Total Mixed Pairs
Mouse feature ranking	Forward selection	avg-pc, avg-drdp, avg-dc, avg-sc, avg-c, avg-m, m-s, m-d, dr-dp, m-m, s-m, d-m, p-s, p-d, m-p, p-m, m-s-m, m-d-m, m-p-s, m-p-d
	Backward selection	avg-pc, avg-drdp, avg-dc, avg-sc, avg-c, avg-m, m-s, m-d, dr-dp, m-m, s-m, d-m, p-s, p-d, m-p, p-m, m-s-m, m-d-m, m-p-s, m-p-d

Table 3 Ranking of features using RFE algorithm

Keystroke feature ranking	Num Features selected: 10
	Selected Features: [False False False True False False False False True False False False False False False False True False False False False True False False False False False False True True False True True False False True False False False True False False False False False False False False False False]
	Feature Ranking: [36 39 33 1 6 14 17 12 1 18 30 24 31 7 35 25 3 1 8 11 22 5 1 21 26 19 32 28 34 1 1 9 1 1 23 2 1 13 41 40 1 10 29 15 16 20 27 38 37 4]
	Features with rank 1 are : Total Alphnm Keys, L-Fn F2, L-Fn LCtrl, R-Fn F10, R-Fn Del, R-Fn Home, R-Fn PgUp, R-Fn PgDn, R-Fn Enter, Alphnm 3
Mouse feature ranking	Num Features: 4
	Selected Features: [False False False False False False False False False False False False True False True True True False False False]
	Feature Ranking: [15 17 12 11 10 4 3 6 9 7 14 5 1 2 1 1 1 16 8 13]
	Features with rank 1 are: p-s, m-p, p-m, m-s-m

5.3 Embedded based ranking results

In embedded methods, implemented are the Lasso regression, Ridge regression and Elastic Net regression methods. RF method is implemented because it is a method that naturally accommodates feature selection and ranking.

- *Lasso Regression:* The lasso regression selection based ranking is once again done by arranging the features in descending order of coefficient estimate and eliminates zero coefficient features. Table 4 shows the feature ranking with lasso regression.

Table 4 Ranking of features based on lasso regression

Keystroke feature ranking	Alphnm z, Total FnPairs
Mouse feature ranking	p-m, m-s-m, m-s, d-m, avg-c, avg-m, avg-sc, avg-pc, avg-dc

- *Ridge Regression:* Table 5 shows the feature ranking with ridge regression. Here the lower the coefficient estimate, lower the relevance of the feature.

Table 5 Ranking of features using ridge regression

Keystroke feature ranking	R-Fn PgDn, R-Fn F9, R-Fn Enter, L-Fn Lalt, R-Fn F10, R-Fn Del, Total Left Fn Keys, R-Fn F12, Alphnm q, Alphnm 3, L-Fn F6, L-Fn F1, L-Fn F4, R-Fn Pause, Total Pairs, L-Fn F5, R-Fn NumLk, Total Special Keys, R-Fn Home, L-Fn F7, R-Fn F11, Total FnPairs, Total Fn Keys, Total Single Keys, R-Fn ArrUp, Total Alphnm Pairs, R-Fn Insert, R-Fn Context, Alphnm z, L-Fn Esc, L-Fn Wnd, L-Fn Tab, Alphnm k, L-Fn CapsLck, L-Fn F3, R-Fn PrtSc, R-Fn SrcLk, L-Fn LShft, R-Fn BckSp, R-Fn RCtrl, Total Right Fn Keys, R-Fn End, R-Fn PgUp, L-Fn F2, Total Mixed Pairs, Alphnm N, Alphnm M, Total Alphnm Keys, R-Fn F8, L-Fn LCtrl
Mouse feature ranking	p-m, m-s-m, m-s, d-m, dr-dp, avg-m, avg-sc, m-m, avg-c, avg-pc, avg-drdp, s-m, avg-dc, m-d-m, m-p-d, m-d, m-p-s, p-s, m-p, p-d

- *Elastic net regression:* Table 6 shows the feature ranking with elastic net regression. Here also the lower the coefficient estimates, lower the relevance of the feature.

Table 6 Ranking of features using elastic net regression

Keystroke ranking	Alphnm z, R-Fn PgDn, Total FnPairs, Total Fn Keys, Total Right Fn Keys, L-Fn F2, L-Fn F3, L-Fn F4, L-Fn F5, L-Fn F6, L-Fn Tab, L-Fn CapsLck, L-Fn LShft, L-Fn LCtrl, L-Fn Wnd, L-Fn Lalt, R-Fn F8, R-Fn F9, R-Fn F10, R-Fn F11, R-Fn F12, R-Fn PrtSc, R-Fn SrcLk, R-Fn Pause, R-Fn Insert, R-Fn Del, R-Fn Home, R-Fn End, R-Fn PgUp, R-Fn NumLk, R-Fn BckSp, R-Fn Enter, R-Fn RCtrl, R-Fn Context, R-Fn ArrUp, Alphnm 3, Alphnm N, Alphnm M, Alphnm q, Alphnm k, Total Pairs, Total Alphnm Pairs, Total Mixed Pairs, Total Special Keys, Total Single Keys, L-Fn F7, L-Fn F1, L-Fn Esc, Total Left Fn Keys, Total Alphnm Keys
Mouse ranking	p-m, m-s-m, m-s, d-m, avg-m, dr-dp, avg-pc, avg-c, avg-sc, m-d-m, m-p-d, avg-dc, m-m, s-m, avg-drdp, m-d, m-p-s, p-s, p-d, m-p

- *Random Forest:* RF model is a natural feature selection and ranking method in machine learning. Only relevant features are output; others are eliminated automatically. Only the TotalAlphnm Keys is relevant for this user in keystroke features and the mouse features are ranked as {p-m, m-s, avg-m, d-m, dr-dp, avg-pc, avg-sc}.

5.4 Authentication results using CRNM framework and ranked features

The effectiveness of feature selection with the CRNM model has been presented in Table 7. Here are presented the results, keeping the first 50 keystroke features and first 13 mouse features. In the case of feature extraction the first 25 components of PCA and LDA are retained. Results indicate that elastic-net regression is best for feature selection

when used with this model. Elastic net regression is a model which works well with multicollinearity, reduces overfit and selects relevant features even when the number of features are high and number of observations are low. The decrease in authentication time is much more significant than the improvement in other performance metrics which leads to the conclusion that this work is ground breaking and leads to the shift from the current stalemate in the area of continuous user authentication using keystroke and mouse dynamics.

Table 7 Performance comparison with ranked features in CRNM model

	Precision (%)	*Recall (%)*	*F-measure (%)*	*FAR*	*FRR*	*Error rate*	*Authentication time (s)*	*Accuracy (%)*
No feature selection	98.5	96.5	98	0.01	1.5	0.1	198	98.25
PCA	83.2	82.1	82	0.5	2.3	0.32	199	89.23
LDA	85.7	84.2	83.1	0.47	2.1	0.28	196	89.78
Variance threshold	83	80.8	82.3	0.52	2.32	0.36	202	77.35
Pearson correlation	84.8	82.6	82.6	0.44	1.87	0.3	213	78.23
Information gain	85.2	84.1	82.2	0.48	1.98	0.29	210	77.26
Chi-square test	86.6	80.2	85.3	0.44	2.2	0.32	200	82.31
Relief	87.3	83.9	86.7	0.16	1.9	0.24	194	83.45
Forward selection kNN	90.2	84.8	88.3	0.12	1.21	0.23	185	86.72
Backward elimination kNN	89.7	79.8	87.7	0.21	2.13	0.33	188	85.77
Bidirectional selection kNN	89.8	79.8	87.5	0.22	2.2	0.29	186	86.21
RFE	93.4	90.5	92.6	0.087	2.023	0.21	190	93.25
Lasso regression	97.2	98.2	95.1	0.066	2.302	0.187	167	96.27
Ridge regression	98.6	97.4	97.2	0.11	2.52	0.212	136	98.76
Elastic net regression	98.8	98.3	96.4	0.042	1.483	0.097	138	99.21
Random forest	96.7	95.2	95	0.141	2.207	0.219	178	97.46

5.5 Discussion

The experiments conducted and observations made lead to the following significant general observations

- the features that are most significant for discrimination are the unusual patterns in an authentication session window
- recurrently selected features are aggregates and uncommon keystroke and mouse actions
- commonly used n-grams in the English language and common mouse movements can be safely eliminated at pre-processing stage
- based on the results seen, researchers can make an informed decision on what kind of features to collect and analyse in the case of continuous user authentication using keystroke and mouse information
- researchers can vary authentication time and session window based on the results presented for further study
- depending on the application, researchers may use these results to partially eliminate feature selection with the help of the ranking presented here.

Some of the specific observations are

- there are some users whose behaviour do not fall in the general category, in these cases features need to be eliminated specifically with respect to the authentic user
- even in these cases, unusual patterns and aggregates are more significant than usual patterns
- the authentication window and size may also vary slightly based on the specific behaviour of the authentic user.

Observations on authentication with CRNM after feature ranking

- Experiments show that there is a significant improvement in authentication time after feature ranking from 198 seconds to 138 seconds. This is unprecedented as literature consistently indicates authentication time to be around 3 minutes. Overall improvements of performance parameters are also observed.

6 Conclusions and future scope

Several experiments are conducted on the keystroke and mouse dataset. The observations made are significant because it will help researchers choose a feature selection method for their work and the ranking can assist in data capture, feature elimination and performance enhancement. The key takeaways are that explicit feature ranking can be helpful in better continuous authentication using keystroke and mouse information and that inorder to get better performance the quality of data is as important and the method of authentication used. This is an unprecedented work to the best of our knowledge and shifts the current trend toward ubiquitous continuous user authentication. As can be seen, elastic net regression worked best with the CRNM model dislodging the authentication time from around 3 minutes as can be seen consistently in the literature. Most research does not mention the authentication time, but whenever given it is always 3 minutes or above. The experiments also suggest that embedded methods are more useful than other methods in this scenario.

This is a first attempt at ranking and using the ranking for enhanced performance with keystroke and mouse dynamics. The literature focus is on implementation of different algorithms without much importance being given to data collection and preprocessing. The experimental results show that focus should be on both. Throughout the experiments the data session window is kept to 3 minutes which is in line with existing literature.

Some of the limitations to this work is that we use public datasets with partially preprocessed data for our experiments. The effect on time when keystroke and mouse information is captured real time and then preprocessed is not studied here. In addition the session window is always kept to 3 minutes, variations in this time may produce different results. Periodic retraining to address authentication for change in behaviour of long term users are also not explicitly addressed.

In future, each of the parameters including the session window size can be varied to check its effect on performance and other feature selection and extraction methods may also be studied. Comparisons need to be done with respect to authentication time along with other standard performance measures. More work is needed in the area to lift the current stalemate and take significant steps towards widespread application of commercially deployable continuous user authentication systems.

References

Abdrabou, Y., Schütte, J., Shams, A., Pfeuffer, K., Buschek, D., Khamis, M. and Alt, F. (2022) ''Your eyes tell you have used this password before': identifying password reuse from gaze and keystroke dynamics', *CHI Conference on Human Factors in Computing Systems (CHI '22). Association for Computing Machinery*, New York, NY, USA, Article 400, pp.1–16, https://doi.org/10.1145/3491102.3517531.

Abu Khurma, R., Aljarah, I., Sharieh, A., Abd Elaziz, M., Damaševičius, R. and Krilavičius, T. (2022) 'A review of the modification strategies of the nature inspired algorithms for feature selection problem', *Mathematics*, Vol. 10, p.464, https://doi.org/10.3390/math10030464.

Ahmed, A.A.E. and Traore, I. (2007) 'A new biometric technology based on mouse dynamics', in *IEEE Transactions on Dependable and Secure Computing*, Vol. 4, No. 3, pp.165–179, July–September, doi: 10.1109/TDSC.2007.70207.

Alsahaf, A., Petkov, N., Shenoy, V. and Azzopardi, G. (2022) 'A framework for feature selection through boosting', *Expert Systems with Applications*, Vol. 187, p.115895, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2021.115895.

Aremu, O.O., Hyland-Wood, D. and McAree, P.R. (2020) 'A machine learning approach to circumventing the curse of dimensionality in discontinuous time series machine data', *Reliability Engineering & System Safety*, Vol. 195, p.106706, ISSN 0951-8320, https://doi.org/10.1016/j.ress.2019.106706.

Baynath, P. and Khan, M.H-M. (2023) 'Application of revised firefly algorithm and grey wolf optimisation on keystroke dynamics', *International Journal of Biometrics*, Vol. 15, Nos. 3–4, pp.480–504.

Belman, A.K. and Phoha, V.V. (2020) 'Discriminative power of typing features on desktops, tablets, and phones for user identification', *ACM Trans. Priv. Secur.*, Vol. 23, No. 1, pp.4:1–4:36.

Cai, J., Luo, J., Wang, S. and Yang, S. (2018) 'Feature selection in machine learning: a new perspective', *Neurocomputing*, Vol. 300, pp.70–79, ISSN 0925-2312, https://doi.org /10.1016/j.neucom.2017.11.077.

Chandrashekar, G. and Sahin, F. (2014) 'A survey on feature selection methods', *Computers & Electrical Engineering*, Vol. 40, No. 1, pp.16–28, ISSN 0045-7906, https://doi.org/10.1016/j.compeleceng.2013.11.024.

Chang, Z., Meng, Y., Liu, W., Zhu, H. and Wang, L. (2022) 'WiCapose: multi-modal fusion based transparent authentication in mobile environments', *Journal of Information Security and Applications*, Vol. 66, p.103130, ISSN 2214-2126, https://doi.org/10.1016/j.jisa.2022.103130.

Dai, X. et al. (2023) 'KD-net: continuous-keystroke-dynamics-based human identification from RGB-D image sequences', *Sensors*, Vol. 23, No. 20, p.8370.

Darabseh, A. and Namin, A.S. (2015) 'Effective user authentications using keystroke dynamics based on feature selections', *IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, pp.307–312, doi:10.1109/icmla.2015.90.

Ding, Y., Wu, Z., Tan, Z. and Jiang, X. (2021) 'Research and application of security baseline in business information system', *Procedia Computer Science*, Vol. 183, pp.630–635, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2021.02.107.

Earl, S., Campbell, J. and Buckley, O. (2021) 'Identifying soft biometric features from a combination of keystroke and mouse dynamics', in Zallio, M., Raymundo Ibañez, C. and Hernandez, J.H. (Eds.): *Advances in Human Factors in Robots, Unmanned Systems and Cybersecurity, AHFE 2021, Lecture Notes in Networks and Systems*, Vol. 268, Springer, Cham., https://doi.org/10.1007/978-3-030-79997-7_23.

El Zein, D. and Kalakech, A. (2018) *IEEE 2018 International Arab Conference on Information Technology (ACIT)*, Werdanye, Lebanon, 28–30 November 2018, pp.1–6, doi:10.1109/ACIT.2018.8672706.

Fülöp, Á., Kovács, L., Kurics, T. and Windhager-Pokol, E. (2016) *Balabit Mouse Dynamics Challenge Data Set* [online] https://github.com/balabit/Mouse-Dynamics-Challenge (accessed June 2022).

Hameed, S., Al-Bahadili, R. and Khidhir, M. (2014) 'Keystroke dynamics authentication based on principal component analysis and neural network', *International Journal of Scientific & Engineering Research*, June, Vol. 5, No. 6, p.830, ISSN 2229-5518.

Jović, A., Brkić, K. and Bogunović, N. (2015) 'A review of feature selection methods with applications', *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp.1200–1205, DOI: 10.1109/MIPRO.2015.7160458.

Karnan, M., Akila, M. and Krishnaraj, N. (2011) 'Biometric personal authentication using keystroke dynamics: a review', *Applied Soft Computing*, Vol. 11, No. 2, pp.1565–1573, ISSN 1568-4946, https://doi.org/10.1016/j.asoc.2010.08.003.

Khalid, S., Khalil, T. and Nasreen, S. (2014) 'A survey of feature selection and feature extraction techniques in machine learning', *2014 Science and Information Conference*, pp.372–378, doi: 10.1109/SAI.2014.6918213.

Kim, D.I., Lee, S. and Shin, J.S. (2020) 'A new feature scoring method in keystroke dynamics based user authentications', *IEEE Access*, pp.1–1, DOI: 10.1109/ACCESS.2020.2968918.

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R.P., Tang, J. and Liu, H. (2017) 'Feature selection: a data perspective', *ACM Comput. Surv.*, November 2018, Vol. 50, No. 6, Article 94, 45pp., DOI: https://doi.org/10.1145/3136625.

Ling, Z., Yu, K., Wang, H., Liu, L., Ding, W. and Wu, X. (2019) 'BAMB: a balanced markov blanket discovery approach to feature selection', *ACM Trans. Intell. Syst. Technol.*, September, Vol. 10, No. 5, Article 52, 25pp., https://doi.org/10.1145/3335676.

Miao, J. and Niu, L. (2016) 'A survey on feature selection', *Procedia Computer Science*, Vol. 91, pp.919–926, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2016.07.111.

Mohamed, T.M. and Moftah, H.M. (2018) 'Simultaneous ranking and selection of keystroke dynamics features through a novel multi-objective binary bat algorithm', *Future Computing and Informatics Journal*, Vol. 3, No. 1, pp.29–40, ISSN 2314-7288, https://doi.org/10.1016/j.fcij.2017.11.005.

Muthuramalingam, A., Gnanamanickam, J. and Muhammad, R. (2018) *Optimum Feature Selection Using Firefly Algorithm for Keystroke Dynamics*, DOI: 10.1007/978-3-319-76348-4_39.

Salman, O.A. and Hameed, S.M. (2018) 'Using mouse dynamics for continuous user authentication', *Advances in Intelligent Systems and Computing*, pp.776–787, doi:10.1007 /978-3-030-02686-8_58.

Sayed, B., Traoré, I., Woungang, I. and Obaidat, M.S. (2013) 'Biometric authentication using mouse gesture dynamics', in *IEEE Systems Journal*, June, Vol. 7, No. 2, pp.262–274, doi: 10.1109/JSYST.2012.2221932.

Shanmugapriya, D. and Ganapathi, P. (2011) 'An efficient feature selection technique for user authentication using keystroke dynamics', *IJCSNS International Journal of Computer Science and Network Security*, October, Vol. 11, No. 10.

Shen, C., Cai, Z., Guan, X. and Cai, J. (2011) 'A hypo-optimum feature selection strategy for mouse dynamics in continuous identity authentication and monitoring', *Proceedings 2010 IEEE International Conference on Information Theory and Information Security, ICITIS 2010*, pp.349–353, DOI: 10.1109/ICITIS.2010.5689603.

Shen, C., Cai, Z., Guan, X. and Roy, A. (2014) 'Maxion: performance evaluation of anomaly-detection algorithms for mouse dynamics', *Comput. Secur.*, September, Vol. 45, pp.156–171.

Shen, C., Cai, Z., Guan, X., Du, Y. and Roy, A. (2013) 'Maxion: user authentication through mouse dynamics', *IEEE Trans. Inf. Forensics Secur.*, Vol. 8, No. 1, pp.16–30.

Shen, C., Cai, Z., Guan, X., Sha, H. and Du, J. (2009) 'Feature analysis of mouse dynamics in identity authentication and monitoring', *Communications, 2009 ICC '09 IEEE International Conference on*, pp.1–5, DOI: 10.1109/ICC.2009.5199032.

Shen, C., Cai, Z., Liu, X., Guan, X. and Roy, A. (2016) 'Maxion: mouseidentity: modeling mouse-interaction behavior for a user verification system', *IEEE Trans. Hum. Mach. Syst.*, Vol. 46, No. 5, pp.734–748.

Stylios, I., Kokolakis, S., Thanou, O. and Chatzis, S. (2021) 'Behavioral biometrics & continuous user authentication on mobile devices: a survey', *Information Fusion*, Vol. 66, pp.76–99, ISSN 1566-2535, https://doi.org/10.1016/j.inffus.2020.08.021.

Sutha, K. and Tamilselvi, J.J. (2015) 'A review of feature selection algorithms for data mining techniques', *International Journal on Computer Science and Engineering (IJCSE)*, June, Vol. 7, No. 6.

Thakkar, A. and Lohiya, R. (2023) 'Fusion of statistical importance for feature selection in deep neural network-based intrusion detection system', *Information Fusion*, February, Vol. 90, pp.353–363.

Thomas, P.A. and Mathew, K.P. (2022) 'An efficient optimized mouse and keystroke dynamics framework for continuous non-intrusive user authentication', *Wireless Pers. Commun.*, Vol. 124, pp.401–422, https://doi.org/10.1007/s11277-021-09363-6.

Utkarsh, M.K. and Dhanalakshmi, R. (2022) 'Stability of feature selection algorithm: a review', *Journal of King Saud University – Computer and Information Sciences*, Vol. 34, No. 4, pp.1060–1073, ISSN 1319-1578, https://doi.org/10.1016/j.jksuci.2019.06.012.

Wang, S., Tang, J. and Liu, H. (2016) *Feature Selection*, DOI: 10.1007/978-1-4899-7502-7_101-1.

Wang, X., Yan, Z., Zhang, R. and Zhang, P. (2021) 'Attacks and defenses in user authentication systems: a survey', *Journal of Network and Computer Applications*, Vol. 188, p.103080, ISSN 1084-8045, https://doi.org/10.1016/j.jnca.2021.103080.

Wesolowski, T., Porwik, P. and Doroz, R. (2020) 'Keystroke dynamics database', *Mendeley Data*, Vol. 1, DOI: 10.17632/vfgn7dd2z4.1.

Wesołowski, T.E., Porwik, P. and Doroz, R. (2016) 'Electronic health record security based on ensemble classification of keystroke dynamics', *Applied Artificial Intelligence*, Vol. 30, No. 6, pp.521–540, DOI: 10.1080/08839514.2016.1193715.

Yamauchi, T. (2013) 'Mouse trajectories and state anxiety: feature selection with random forest', *Proceedings – 2013 Humaine Association Conference on Affective Computing and Intelligent Interaction, ACII 2013*, pp.399–404, DOI: 10.1109/ACII.2013.72.

Zebari, R., Abdulazeez, A., Zeebaree, D., Zebari, D. and Saeed, J. (2020) 'A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction', *Journal of Applied Science and Technology Trends*, 15 May, Vol. 1, No. 2, pp.56–70.

Zimmermann, V. and Gerber, N. (2020) 'The password is dead, long live the password – a laboratory study on user perceptions of authentication schemes', *International Journal of Human-Computer Studies*, Vol. 133, pp.26–44, ISSN 1071-5819, https://doi.org/10.1016/j.ijhcs.2019.08.006.