Automated Script Recognition Based on Human-Computer Interaction Features of Mouse Movement Trajectory

Chenyang Wu

{u202211875@hust.edu.com}

School of Cyber Science and Engineering, Huazhong University of Science and Technology, Hubei, China

Abstract. As human-computer interaction scenarios become increasingly diverse, human-computer recognition technology has become crucial. While ensuring that human-machine recognition technology cannot be cracked, user experience should also be taken into account.This paper extracts the kinematic features and human-computer interaction features of mouse trajectories.Having done 10 cross-validations on a training sample containing 3000 mouse trajectories, it was then inputted as the optimal parameters and the training sample into an Extreme Gradient Boosting (XGBoost) binary classification model to be trained. In the last model, human behavior recognition rate is 97.974 percent and machine behavior recognition rate is 98.012 percent, and the F1 score of had been obtained to be 98.04784. This paper employs machine learning to recommend a new strategy of human-machine recognition.

Keywords: Human-computer interaction, Mouse trajectory, Human-computer recognition, Machine learning.

1 Introduction

On the one hand, online platforms are usually based on common human-machine verification technology that has serious limitations and is easy to overcome using image recognition and user behavior simulation technologies [1-3]. On one side, though, dynamic human-machine verification may consist of complicated processes, and these complicated interrelationships may lead to a poor user experience. Thus, it is of significant practical importance to increase the limits of static verification and inhibit them with the help of discovering a new human-machine verification technology that will guarantee a good user experience.

As implicit data when the human interacts with the computer, the movement of the mouse is an immediate indicator of the intentions of the operational processes and the behavior specific to the user, as well as the human-computer differences [3-5]. When man communicates with computers the human brain needs time to digest information, which is reflected in delays, pauses and information contemplation [6, 7]. Such distinctly human actions are very dissimilar to the

ICIAAI 2026, May 22-24, Singapore, Singapore

DOI 10.4108/eai.22-5-2026.2365097

direct-to-the-point actions of automated scripts. Moreover, the perception of human information is slow and humans cannot fully control their muscles; these are all differences in the interaction of human and automated scripts.

In this paper, the researcher offers a new method of human-computer verification based on the data of the mouse path, which overcomes the drawbacks of the traditional verification that is usually based on a single point and eliminates the nature of interaction and its experience. It also suggests a safer human-computer checking plan without escalating the costs of the user in his or her operation.

2 Methods

2.1 Overall process

This study first preprocesses the original dataset, modifying it to DMatrix format (the XGBoost model's proprietary data structure). After initially confirming the relevant parameters of the XGBoost model, relevant features are extracted from the dataset, and a total of ten cross-validations are performed. The results of these ten cross-validations are then combined to complete full training, resulting in the final trained XGBoost model. This model is then used to predict on the test set, filtering automated script behavior samples and outputting positive sample prediction results, which are saved in text format. Furthermore, during the discrimination process, the importance scores of different features are also output in text format. The specific process is shown in Figure 1.

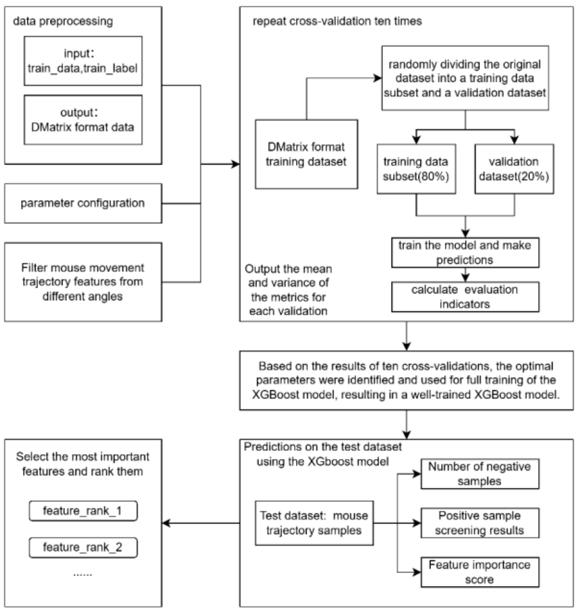

Figure 1. Overall Machine Learning Flowchart (Picture credit: Original)

2.2 Data preprocessing

In this study, the data preprocessing workflow was as follows: loading the dataset, performing data cleaning, implementing feature engineering, splitting the dataset into training and test sets, and converting the processed data into DMatrix format. This conversion improved training efficiency and is suitable for large-scale data.

2.3 Feature extraction

Mouse motion trajectory is a continuous dynamic flow of user behavior during perception, decision-making, and execution, containing rich individual behavioral characteristics. This study analyzed the mouse motion process dataset and extracted 20 kinematic features [8, 9]. Some of these features, their definitions, and their discriminative significance are shown in Table 1.

Table 1. Kinematic characteristics of common mouse trajectories.

Feature	Definition	Discrimination meaning
Displacement length	Euclidean distance between adjacent trajectory points	The basic measurement value, the amplitude of a single mouse movement, is a prerequisite variable for other characteristic values.
Time interval	Time difference between adjacent points	Basic measurement value: the time difference between two mouse movement operations
instantaneous speed	displacement change per unit time	The speed of mouse movement during operation. Excessive instantaneous speed, corresponding to jerky mouse movements, is often machine-generated behavior.
acceleration	rate of change of velocity	The acceleration of mouse movement during operation. Humans often exhibit irregular acceleration values when controlling a mouse by manipulating their hand muscles.
Total operation time	Time difference between the beginning and end of the trajectory	The total time from moving the mouse to clicking it. An excessively short operation time is often characteristic of machine-simulated behavior driven by a pursuit of efficiency.

Humans cannot maintain absolute control over their muscles. Introducing these basic kinematic features can filter out overly regular, straight, and fast mouse trajectories. These kinematic features will be imitated during the continuous iteration of automated scripts. Therefore, 15 human-computer interaction features were re-extracted from the perspective of human-computer interaction. Some of these features, their definitions, and their discriminative significance are shown in Table 2.

Table 2. Characteristics of Common Human-Computer Interaction Behaviors.

Feature	Definition	Discrimination meaning
Longest pause duration	Calculate the maximum time difference between all adjacent points in the mouse trajectory.	Human operation involves long pauses due to thinking and hesitation, while automated scripts have no subjective decision-making and therefore have almost no such pauses.

Percentage of time spent on pause operations	Calculate the proportion of pauses to the total number of operation steps.	Human operations involve a high percentage of pauses, while automated scripts execute according to fixed logic and have a low percentage of pauses.
Percentage of fine-tuning operations	The proportion of tiny displacements to the total number of displacements reflects the precision and stability of the operation.	Human operation may generate numerous small displacements due to tension and target adjustments, while automated script path planning is precise, resulting in a lower proportion of small displacements.
Path planning efficiency	Calculate the ratio of the actual travel path length to the straight-line distance from the starting point to the ending point.	Human-operated paths may contain redundancies (such as backtracking and fine-tuning), while automated script paths approach 100% efficiency.

Compared to kinematic features, human-computer interaction features better reflect the hesitation and thought processes humans exhibit when manipulating a mouse, as humans cannot pursue absolute efficiency like machines [6, 7]. Combining these features yields Feature Combination 1 (containing 20 kinematic features) and Feature Combination 2 (containing 20 kinematic features and 15 human-computer interaction features).

2.4 Cross-validation

Ten cross- validations were done in the training set. In every cross-validation, an attribute of randomization was created: the random state which was used to divide the training set. The set was randomly selected into a training subset of data (2400 data points, containing 80 percent of the original data) and a validation subset of data (600 data points, containing 20 percent of the original data). The XGBoost model was then trained on the training subset and the validation set was used to test the model. Successful measurements at the conclusion of every cross-validation cycle included the corresponding metrics, namely, recall and precision. Depending on the overall outcomes of the ten cross-validations, the best parameters were determined and set up. Lastly, a complete training cycle was undertaken and the model that was obtained was applied to the test set to make predictions.

2.5 Extreme Gradient Boosting binary classification model

The current research uses the Open-source Extreme Gradient Boosting (XGBoost) binary classification model as the main modeling tool to accomplish a binary classification problem [10]. XGBoost is an ensemble algorithm that is used on the gradient boosting framework. It repeatedly uses numerous decision trees and balances and combines their outputs, therefore, minimizing the bias and variance of the model. It shows high nonlinear modelling and it also generalizes well. This model adopts the logistic regression objective function which is customarily applied in binary classification of a sample with the aim of retrieving the probability value of a sample falling into the positive category. The objective function is also optimized in a continuous manner during training and multiple iterations are performed so that the model can have good generalization capabilities whilst being able to fit the training data.

3 Results

3.1 Performance Analysis of Model Based on Feature Combination 2

Table 3 shows the specific performance metrics of the XGBoost model trained based on feature combination 2. The model performed excellently in 10 cross-validations, with a mean recall of 0.97974 and a variance of only 0.0001, indicating an extremely low false negative rate and stable results, making it suitable for scenarios sensitive to false negatives. The mean precision was 0.98102 with a variance of 0.00003, indicating a low false positive rate, highly reliable prediction results, and strong stability. The mean custom-weighted F-score reached 98.04784 with a variance of only 0.09707, reflecting the model's excellent overall performance and good generalization ability. The number of negative samples was 16, which is within a reasonable range, indicating that the data processing is logical.

Table 3. Analysis of Experimental Results for Feature Combination 2.

Index	Numerical value	Indicator definition	Conclusion
recall mean	0.97974	Mean recall rate across 10-fold cross-validation	The false negative rate is extremely low, making it suitable for scenarios where false negatives are critical.
recall var	0.0001	Recall variance using 10-fold cross-validation	The recall rate shows no fluctuations, indicating extremely strong model stability.
precision mean	0.98102	Mean accuracy of 10-fold cross-validation	The error rate is extremely low, and the predictions are highly reliable.
precision var	0.00003	Variance of accuracy across 10-fold cross-validation	The accuracy remains stable, indicating extremely strong model stability.
score mean	98.04784	Custom weighted F-score mean	This indicates that the model has good overall performance.
score var	0.09707	Custom weighted F-score variance	Low variance, strong model generalization ability.
negative number	16	The number of negative samples in data processing.	The values are within the normal range and are logical.

Overall, the model exhibits high accuracy, strong stability, and good robustness, making it suitable for practical scenarios with high requirements for accuracy and stability.

3.2 Comparative analysis of the performance of the two models

In this study, two XGBoost models were trained based on feature combination one and feature combination two, respectively. The performance comparison of the two models is shown in Table 4. The results show that after introducing human-computer interaction features, the overall performance of the model is significantly improved: the score F increases from 97.19 to 98.05, and both recall R and precision P improve, which means a reduction in false negatives and false negative rates. More notably, the variance of the score F in cross-validation decreases from 0.198 to 0.097, indicating a significant improvement in model stability and generalization ability.

Table 4. Performance Comparison Analysis of Models Trained by Feature Combination 1 and Feature Combination 2.

Index	Feature Combination 1	Feature combination 2	Change
F1 overall score	97.19244	98.04784	Accuracy improved.
Recall rate R	0.96842	0.97974	The rate of missed calls/errors has decreased.
Precision R	0.97439	0.98102	The error rate has decreased.
The variance of the F1 score in cross-validation.	0.19829	0.09707	Improved model stability
Number of negative samples	17	16	Reasonable fluctuations

The newly introduced human-computer interaction features effectively supplement the behavioral context information, improving the model's recognition accuracy and robustness from multiple dimensions. This indicates that the fusion of human-computer interaction features and kinematic features is more conducive to judgment in complex scenarios.

3.3 Feature Importance Ranking

In the XGBoost model trained based on feature combination 2, different features have different importance scores, as shown in Table 5. Feature importance analysis shows that the feature "maximum pause time" ranks first with a score of 90.0, significantly higher than other features. This indicates that machine-simulated behavior often lacks the randomness of natural human pauses in its operation rhythm, tending to pause continuously or regularly. This difference becomes the core criterion for judgment.

Table 5. Ranking of the Importance of Features in Feature Combination 2.

Ranking	Feature name	Feature type	Feature importance score
1	max_pause_time	Human-computer interaction features	90.0
2	xt_angles_max	Kinematic characteristics	70.0
3	get_target	Kinematic characteristics	58.0
4	x_min	Kinematic characteristics	51.0
5	avg_speed	Human-computer interaction features	48.0
6	first_data_x	Kinematic characteristics	44.0
7	first_data_y	Kinematic characteristics	43.0
8	first_delt_t	Kinematic characteristics	40.0
9	dxspeed_mean	Kinematic characteristics	38.0
10	pause_ratio	Human-computer interaction features	35.0

In summary, kinematic features occupy seven of the ten most important characteristics, indicating that analyzing and identifying trajectories through basic kinematics remains the core approach. The model's overall discriminative ability has been significantly improved after the introduction of human-computer interaction features, especially features such as "percentage of pauses" and "maximum pause time," which reflect the significant differences between machine-simulated behavior and human manipulation behavior in terms of overall time distribution and speed control.

4 Conclusion

This paper begins with the features of human-computer interactions. It does not need the conventional explicit model of human-computer verification but relies on implicit data generated by users (mouse motion tracks) to uncover human activities such as hesitation in decision-making and corrections in using the computer and features these as characteristics. These characteristics portray the uncertainty, irregularity, and irrationality of human behavior and are thus applicable as a new condition of perceiving the machine-simulated behavior.

The experimental findings indicate that when human-computer interaction properties are added to the data model, the model's capacity to differentiate human and machine behavior is significantly improved, and is more stable under different test conditions, which proves the efficiency of the model optimization in relation to the human-computer interaction process.

Further development of the model will incorporate more work on the generalization capability and efficiency. In this experiment 10 fold cross-validation dataset was employed to train the model but the model should be improved in terms of generalization ability. Also, the feature selection and optimization will be done continuously to enhance the ease of the model in real time.

References

[1] Xu, Z., Kang, Y., Cao, Y., et al.: Man-machine verification of mouse trajectory based on the random forest model. Frontiers of Information Technology & Electronic Engineering. pp. 925-929 (2019)

[2] Lu, X., Feng, Z., Xia, J.: An efficient man-machine recognition method based on mouse trajectory feature de-redundancy. Proceedings of the 37th Annual Computer Security Applications Conference. pp. 365-374 (2021)

[3] Xu, D., Lu, X.: Human-machine recognition based on mouse behavior modeling. Second International Symposium on Computer Applications and Information Systems (ISCAIS 2023). SPIE. pp. 228-232 (2023)

[4] Maldonado, M., Dunbar, E., Chemla, E.: Mouse tracking as a window into decision making. Behavior Research Methods. pp. 1085-1101 (2019)

[5] Pan, X., et al.: Password extraction via reconstructed wireless mouse trajectory. IEEE Transactions on Dependable and Secure Computing. pp. 461-473 (2016)

[6] Reeder, R. W., Maxion, R. A.: User interface defect detection by hesitation analysis. International Conference on Dependable Systems and Networks (DSN'06). pp. 61-72 (2006)

[7] Vodlan, T., Tkalčič, M., Košir, A.: The impact of hesitation, a social signal, on a user's quality of experience in multimedia content retrieval. Multimedia Tools and Applications. pp. 6871-6896 (2015)

[8] Kieslich, P. J., Schoemann, M., Grage, T., et al.: Design factors in mouse-tracking: What makes a difference? Behavior Research Methods. pp. 317-341 (2020)
[9] Wulff, D. U., Haslbeck, J. M. B., Kieslich, P. J., et al.: Mouse-tracking: Detecting types in movement trajectories. A handbook of process tracing methods. Routledge. pp. 131-145 (2019)
[10] Chen, T.: XGBoost: A scalable tree boosting system. Cornell University. (2016)