A Study of the Consistency in Keystroke Dynamics

Chao Shen[1], Roy A. Maxion[2], and Zhongmin Cai[1]

[1] MOE KLINNS Lab, Xi'an Jiaotong University, Xi'an, Shaanxi, China
[2] Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, USA
{cshen,zmcai}@sei.xjtu.edu.cn, maxion@cs.cmu.edu

Abstract. Keystroke dynamics is the process of identifying individual users on the basis of their distinctive typing rhythms. Most current approaches implicitly assume that individual typing behavior has high consistency as well as high discriminatory ability, both of which underpin the power of the technique. However, no earlier work has been done to quantify or measure the consistency of typing behavior. This study aims to investigate the consistency of users' typing behavior in keystroke dynamics. We obtain a keystroke benchmark dataset, propose a consistency measurement model, develop an evaluation methodology, and conduct three studies. We first quantify the consistency of users' behavior in repeatedly typing a password, observing that a typical user's typing behavior would become consistent over time, and changes in her typing would diminish. We then measure the consistency of keystroke timing features, finding that the combination of all features has the best consistency and smallest fluctuation. We finally examine the effect of consistency on keystroke-biometric systems, observing that the authentication performance gets better as the user's typing behavior becomes more consistent.

Keywords: Keystroke dynamics, Evaluation and benchmarking, Consistency.

1 Introduction

Keystroke dynamics is the analysis of individual typing behavior for use as a biometric identifier. Current research largely uses the timing latencies, which are extracted from typing behavior between key-down and key-up events, to discriminate legitimate users from impostors. Its prerequisite is the high discriminability and consistency of users' typing behavior.

It has been established that users' typing behavior is a form of perceptual-motor skill acquisition, and the gradual improvement of a repeated activity [1]. Most computer users have the experience of being required to use a new password or type a certain paragraph repeatedly. At the beginning, the typing behavior for the new text (either a password or a paragraph) appears to be clumsy, but by time it becomes easy and quick, and to the end it would become consistent and fluent [15].

In the field of keystroke biometrics, the typing behavior is used to discriminate users, so the influence of gradual skill acquisition may be an issue. However, most current approaches implicitly assume the timing latencies have high consistency as well

Z. Sun et al. (Eds.): CCBR 2013, LNCS 8232, pp. 409–416, 2013.

as high discriminatory power. Moreover, due to lack of consistency measure and benchmark dataset, no earlier studies have been done to investigate the consistency of users' typing behavior.

2 Background and Related Work

Keystroke dynamics is the procedure of measuring and assessing users' typing behavior. Since Forsen *et al.* [3] first investigated in 1977 whether the way of users typing their names could be used to distinguish a legitimate user from an impostor, several usages for keystroke dynamics have been proposed. There are really two usages of interest for this biometric: static analysis (e.g., verification at login time) and continuous analysis (e.g., verification throughout the use of a computer). Most static analysis approaches use fixed-text models [4], [5], [6], [7], [8], in which they use the same static piece of text (e.g., password), to identify users. In these approaches, the length of the required text varies between different studies, and usually the use of a long text [5], [6] could lead to a better performance. Recent work has explored free-text models for continuous verification [9], [10], in which users' keystroke activities are monitored and analyzed during their routine computing activities, and have shown good potential if observing sufficient period of data.

In terms of consistency considerations, there are only few efforts of implicitly exploring consistency in keystroke dynamics, by investigating the effect of enrollment sample size on detection results or the use of updating strategies for performance improvement. Bartmann *et al.* [13] examined the authentication results at different amount of enrollment data, and they observed that the results get better with more amounts of enrolment data. Kang *et al.* [14] showed that the authentication results get improved when they continually retrain the classifier with recent typing data.

3 Problem and Approach

In this work, our goal is to investigate the consistency of users' typing behavior in keystroke dynamics, in part to assess the effect of the consistency on keystroke-biometric systems. To achieve this goal, we obtain a benchmark dataset, propose a consistency measurement model, develop evaluation methodology, and conduct three studies. Specifically, we lay out a set of three questions to guide our investigation:

1. How is the consistency of a user's behavior in typing a password?
2. How is the consistency of the keystroke timing features?
3. Does the consistency affect the accuracy of keystroke-biometric system?

Each question concerns a different facet of the consistency of users' typing behavior in keystroke dynamics. We conduct three studies to answer these questions. In Sections 4–6, we describe the three studies in more detail.

4 Study1: Consistency and Tying a Password

In this study, we measure and quantify the consistency of a user's behavior and its change in repeatedly typing a password. The purpose of Study 1 is to answer the question: *how is the consistency of a user's behavior in typing a password?*

4.1 Study 1: Method

4.1.1 Collecting Data

We used an existing dataset that has been published and shared in our previous study [12]. The data were obtained when 51 users typed the same 10-character password 400 times each. The 400 passwords were typed in 8 sessions of 50 passwords each, with the sessions all occurring on different days. The password was **.tie5Roanl**.

The reasons for the dataset enabling our study of the consistency of keystroke dynamics are that (a) the data set is generated by 51 users, (b) the password is novel, and (c) it is typed many times by each user.

4.1.2 Extracting Features

We extracted keystroke timing features from key-down and key-up events. Usually three types of features are used: (1) the latency between keydown events in a digram (keydown-keydown time); (2) the latency between keyup and keydown events in a digram (keyup-keydown time), and (3) the length of time that a key is pressed (hold time). For each repetition of the password, we extracted 31 timing features: 10 keydown-keydown times, 10 keyup-keydown times, and 11 hold times.

4.1.3 Proposing a Consistency Measurement Model

We developed a simple and effective consistency measure, based on Gini Mean Difference (GMD) [2]. Since direct use of timing features could not accurately reflect the overall picture of a user's typing behavior, here we employed distance-based metric to compute the consistency. We used cosine distance to calculate the distance between each feature sample and a reference sample, and then used this distance to represent the feature sample. We chose cosine distance instead of commonly used Euclidean distance due to its generalized applicability of measuring the similarity of two samples with result between 0 and 1. We next calculated the mean of the absolute difference between all possible pairs of the distances as the consistency measurement. We defined the consistency measurement of the typing behavior for user *k* as:

$$GMD_{cons}(k) = \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{n} \left| DM_i - DM_j \right|$$

where the DM_i is the distance measure associated with timing features, and n is the number of feature samples of that user. This measurement is a real value which is zero if the data are identical, and increases as the data become more diverse.

4.1.4 Measuring the Consistency

For a given user, we first computed the distance measurement between each of her feature samples and a reference sample. The procedure is as follow:

Step 1: Generate the reference sample using a one vector due to its simplicity.
Step 2: Compute the pairwise distance between each feature sample and the reference sample by using cosine distance.

Then given the distance metrics from a reference sample, we could easily obtain the consistency measurements across difference sessions. The procedure is as follow:
Step 1: Compute the consistency measurement of all distance samples in one session from a user by using the proposed consistency model.
Step 2: Repeat *Step 1* for all reaming sessions of that user.
Step 3: Repeat the above procedure and calculate the average consistency measurements for all 51 users in each of 8 sessions.

4.2 Study 1: Results

Figure 1 shows a plot of the average consistency measurements and the standard deviations of users' typing behavior in different sessions over all 51 users. The figure reveals that the consistency measurement improves greatly within first three sessions, but after the fourth session, only small fluctuations with error range are apparent. These results suggest a typical user would become consistent when repeatedly typing a password, and the changes in her typing diminish after a number of repetitions.

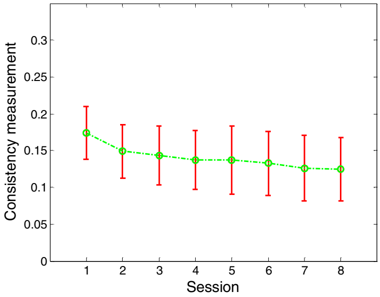

Fig. 1. Consistency measurement over different sessions. Error bars represent the standard deviation in the consistency measurements of all 51 users.

The above results show a user would provide better consistency on her typing with more repetitions in an average manner. But in many evaluations, an individual user may also provide poor consistency in her typing behavior. This poor consistency may have an effect on keystroke timing features, as will be explored in Study 2.

5 Study 2: Consistency and Keystroke Timing Features

When users' typing behavior becomes consistent, the keystroke timing features must change. Here we conduct a study to answer the question: *how is the consistency of the keystroke timing features?*

5.1 Study 2: Method

We generated all combinations of keystroke timing features, and used the consistency measurement model to evaluate the consistency of each feature combination.

5.1.1 Deriving Feature Combinations

Current researchers typically used three kinds of time interval as keystroke features: keydown-keydown time (KDD), keyup-keydown time (KUD), and hold time (KH). Here we derived seven combinations of the individual timing features. They are: (1) three individual keystroke features (KDD, KUD, and KH); (2) three two-component feature combination (KDD and KUD, KDD and KH, KUD and KH); (3) one three-component feature combination.

5.1.2 Measuring Consistency of Each Feature Combination

We employed the consistency measurement model to calculate the consistency on each of seven feature combinations in different data sessions. We first computed the consistency measurement of each feature combination in one session overall all users. Then we calculated the metric for all other sessions.

5.2 Study 2: Results

Figure 2 depicts the consistency measurements of different feature combinations over different data sessions.

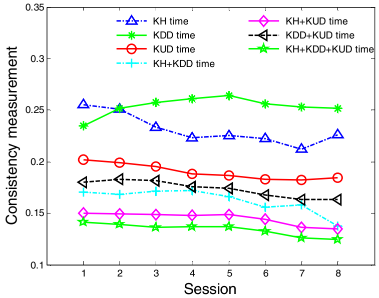

Fig. 2. Consistency measurements of seven feature combinations across different data sessions. This figure shows the change and comparison of consistency for three individual features, three two-component feature combinations, and one three-component feature combination.

The results from Figure 2 show that nearly the consistency of all the feature combinations (with the exception of KDD feature) gets better when the users type more repetitions of the password. Specifically, the combination of all three individual features holds best consistency and smallest fluctuation. Moreover, we observe that the consistency of the KUD time and its change are much better and more stable than other two individual features. These results confirm the earlier logic [4], [5] that using multiple features would lead to better overall performance, and are consistent with the previous results [5] which showed the discriminability of each feature combination.

The above results show the consistency of keystroke timing features will improve with more repetitions. But if a biometric system is trained with inconsistent feature samples, it may lead to a poor performance. Thus the consistency may have an influence on the accuracy of keystroke-biometric systems, as will be explored in Study 3.

6 Study 3: Consistency and Biometric-System Accuracy

Having shown that the keystroke timing features get more consistent with more repetitions of password, it is natural to investigate whether the consistency manifests in the error rates of keystroke-biometric systems. The purpose of Study 3 is to answer the question: *does the consistency affect the accuracy of keystroke-biometric systems?*

6.1 Study 3: Method

We first develop a keystroke-authentication system. We then examine the authentication accuracy at different levels of consistency.

6.1.1 Keystroke-Authentication System

We implemented the keystroke-authentication system which was proposed by Araujo *et al.* [11]. The reason is that this approach had the top performance in a field of approaches evaluated in our previous work [12]. The system was divided into two phase: an enrollment phase and an authentication phase. In the enrollment phase, a training dataset composed by several repetitions of the password from a legitimate user is used to build a profile of the user. Then, the mean vector and mean absolute deviation of each feature are calculated. In the authentication phase, a test sample is presented to the system and compared with the profile. The system produces a classification score indicating whether the test sample is similar to the profile or different from the profile.

6.1.2 Training and Testing Procedure

We started by designating one of 51 users as the legitimate user, and the rest as impostors. We trained and tested the authentication system as follows:

Step 1: we run the enrollment phase of the system on the feature vectors from one session's password data (50 repetitions) typed by the legitimate user.
Step 2: we run the authentication phase of the system on the feature vectors from another session's password data (50 repetitions) typed by the legitimate user, to test its ability to classify the legitimate user.
Step 3: the evaluation procedures of above two steps (*Step 1* and *Step 2*) can be performed by any two of the eight sessions' data. To examine the effect of consistency on performance, we first used Session 1 as the training session and Session 2 as the testing session for the legitimate user. We then used Session 2 and session 3 as the training and testing sessions respectively. We continued in this way until Session 7 and Session 8 were used as the training and testing sessions for the legitimate user.
Step 4: we run the authentication phase of the system on the feature vectors from the first session's password data typed by each of the 50 impostors, to test its ability to classify the impostors.

This process was then repeated, designating each of the other users as the legitimate user in turn.

6.1.3 Calculating Authentication Performance

To convert the classification scores of legitimate users and impostors into aggregate measures of classifier performance, we computed false-acceptance rate (FAR) and false-rejection rate (FRR). We brought FAR and FRR together to generate an ROC curve, and we also set the threshold for the classification scores to make the FAR equal with FRR, for presenting the equal-error rate (EER).

6.2 Study 3: Results

Figure 3 show the ROC curves for different training and testing data sessions from legitimate user, which represents different levels of consistency.

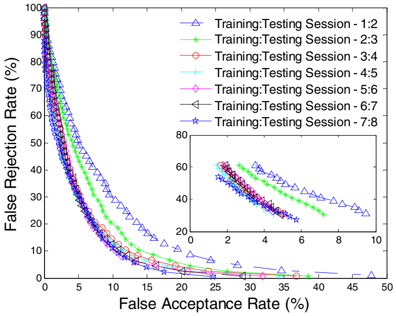

Fig. 3. ROC curves for different training and testing data sessions from legitimate user. The inside panel details the points of ROC curves at the left corner.

We can obtain that the EER of the evaluation, which uses the training and testing sessions of Session 1 and 2 from the legitimate user, is 14.27%, and the corresponding ROC curve is worse than other 6 evaluations. When using the training and testing sessions from the legitimate user with more repetitions (like Session 3 and 4), the EER reduces to 10.29% and the corresponding ROC curve obviously gets improved. Along with the results (obtained in Study 1 and Study 2) that users' typing behavior would become more consistent with more repetitions of the password, we could draw a conclusion that the authentication accuracy gets better as the user's typing becomes more consistent with more repetitions of the password.

7 Conclusion and Future Work

The goal of this work is to investigate the consistency of users' typing behavior in keystroke dynamics, in part to assess the effect of the consistency on keystroke-biometric systems. Experimental results show: (1) a typical user's typing behavior would become consistent over time and the changes in her typing would diminish; (2)

the use of all features has the best consistency and smallest fluctuation; (3) the authentication accuracy gets better as users' typing behavior becomes more consistent.

Acknowledgements. The research is supported by NFSC (61175039，61221063), 863 High Tech Development Plan (2007AA01Z464, 2012AA011003), Research Fund for Doctoral Pro-gram of Higher Education of China (20090201120032), International Research Collaboration Project of Shaanxi Province (2013KW11) and Fundamental Research Funds for Central Universities (2012jdhz08). This work was also partially supported by the US National Science Foundation, grant number CNS-0716677.

References

1. Rosenbaum, D.A., Carlson, R.A., Gilmore, R.O.: Acquisition of Intellectual and Perceptual-Motor Skills. Annual Review of Psychology 52, 453–470 (2001)
2. Yitzhaki, S.: Gini's Mean Difference: a Superior Measure of Variability for Non-Normal Distributions. METRON-Int'l J. Statistics 285–316 (2003)
3. Forsen, G., Nelson, M., Staron, R.: Personal Attributes Authentication Techniques., Tech. Report RADC-TR-77-1033, Griffis Air Force Base (1977)
4. Bergadano, F., Gunetti, D., Picardi, C.: Identity Verification through Dynamic Keystroke Analysis. Intell. Data Anal. 7(5), 469–496 (2003)
5. Bergadano, F., Gunetti, D., Picardi, C.: User Authentication through Keystroke Dynamics. ACM Trans. Inf. Syst. Secur. 5(4), 367–397 (2002)
6. Gaines, R., Lisowski, W., Press, S., Shapiro, N.: Authentication by Keystroke Timing: Some Preliminary Results (1980)
7. Joyce, R., Gupta, G.: Identity Authentication based on Keystroke Latencies. Commun. ACM 33(2), 168–176 (1990)
8. Monrose, F., Rubin, A.D.: Keystroke Dynamics as a Biometric for Authentication. Future Gener. Comput. Syst. 16(4), 351–359 (2000)
9. Dowland, P., Furnell, S.A.: A Long-term Trial of Keystroke Profiling using Digraph, Trigraph, and Keyword Latencies. In: Deswarte, Y., Cuppens, F., Jajodia, S., Wang, L. (eds.) Security and Protection in Information Processing Systems. IFIP International Federation for Information Processing, vol. 147, pp. 275–289. Springer, Heidelberg (2004)
10. Gunetti, D., Picardi, C.: Keystroke Analysis of Free Text. ACM Trans. Inf. Syst. Secur. 8(3), 312–347 (2005)
11. Araujo, L.C.F., Sucupira, L.H.R., Lizarraga, M.G., Ling, L.L., Yabu-Uti, J.B.T.: User Authentication through Typing Biometrics Features. IEEE Trans. Signal Process. 53(2), 851–855 (2005)
12. Killourhy, K.S., Maxion, R.A.: Comparing Anomaly Detectors for Keystroke Dynamics. In: Proc. Annual Int'l Conf. Dependable Systems and Networks, pp. 125–134 (2009)
13. Bartmann, D., Bakdi, I., Achatz, M.: On the Design of an Authentication System Based on Keystroke Dynamics Using a Predefined Input Text. Int'l J. Infor. Secur. Priva. 1(2), 1–12 (2007)
14. Kang, P., Hwang, S.-s., Cho, S.: Continual Retraining of Keystroke Dynamics based Authenticator. In: Lee, S.-W., Li, S.Z. (eds.) ICB 2007. LNCS, vol. 4642, pp. 1203–1211. Springer, Heidelberg (2007)
15. Cialdini, R.B.: Influence: Science and practice. Needham Heights (2001)