Keystroke Analysis of Different Languages: A Case Study

Daniele Gunetti, Claudia Picardi, and Giancarlo Ruffo

Department of Informatics, University of Torino,
corso Svizzera 185, 10149 Torino, Italy
{gunetti, picardi, ruffo}@di.unito.it

Abstract. Typing rhythms are one of the rawest form of data stemming from the interaction between humans and computers. When properly analyzed, they may allow to ascertain personal identity. In this paper we provide experimental evidence that the typing dynamics of free text can be used for user identification and authentication even when typing samples are written in different languages. As a consequence, we argue that keystroke analysis can be useful even when people may use different languages, in those areas where ascertaining personal identity is important or crucial, such as within Computer Security.

1 Introduction to Keystroke Analysis

Keystroke Analysis is the biometric area concerned with the problem of ascertaining users' identity through the way they type on a computer keyboard [1]. As such, it is essentially a form of Pattern Recognition, as it involves *representation* of input data measures, *extraction* of characteristic features and *classification* or *identification* of patterns data so as to decide to which pattern class these data belong [9].

In the case of typing rhythms, input data is usually represented by a sequence of typed keys, together with appropriate timing information so that it is possible to compute the elapsed time between the release of the first key and the depression of the second (the so-called *digraph latency*) and the amount of time each key is held down (the *keystroke duration*). The extraction of such features turns a sequence of keystrokes into a *typing sample*. Appropriate algorithms are then used to classify a typing sample among a set of pattern classes, each one containing information about the typing habits of an individual. Pattern classes are often called *profiles* or *models*, and they are built using earlier typing information gathered from the involved individuals.

Within computer science, a biometric such as keystroke dynamics is particularly appealing, since it can be sampled without the aid of special tools, just the keyboard of the computer where the biometric analysis has to be performed. Keystroke analysis is however a difficult task, for several reasons: (1) keystrokes, unlike other biometric features, convey an unstructured and very small amount of information. Keystroke duration and digraph latency are in fact a pretty shallow kind of information. (2) Keystroke dynamics are a *behavioral* biometric, like voiceprints and handwritten signatures. As such, they are intrinsically unstable, and show a certain degree of variability even without any evident reason. After all, it is pretty difficult to control the number of milliseconds we hold

A.F. Famili et al. (Eds.): IDA 2005, LNCS 3646, pp. 133–144, 2005.

down a key when typing. (3) The variability of typing rhythms may be magnified by the fact that, of course, during the normal use of a computer, different texts are entered, possibly in different languages.

To deal with the instability of typing dynamics, most experiments within keystroke analysis have been limited to samples produced from a unique pre-defined text (e.g. [13,12,5,6,17,3]) or from a small number of different, but still pre-defined texts, (e.g. [14,10,4]). and we refer to [3] and [4] for a thorough descriptions of the various methods found in the literature. However, a large part of the interest in keystroke analysis lies in the possibility to use what stems from the normal use of a computer: the typing rhythms of free text. For example, Intrusion Detection techniques would benefit from such ability, as we discuss at the end of the paper. Unfortunately, when analyzing the typing dynamics of free text the variability of keystroke dynamics is akin to get worse, since the timings of a sequence of keystrokes may be influenced in different ways by the keystrokes occurring before and after the one currently issued. This is even more true if different languages are involved.

Analysis of "true" free text is attempted in [16], where the authors test different methods based on the Euclidean distance and on the mean typing speed and standard deviation of digraphs to measure similarities and differences among typing samples of 31 users, reaching a 23% of correct classification of the typing samples.

In [8] four users are monitored for some weeks during their normal activity on computers, so that thousands of digraphs latencies can be collected. Authors use both statistical analysis and different data mining algorithms on the users' data sets, and are able to reach an almost 60% of correct classification. Authors' approach is improved in [7], both in the outcomes and in the number of users (35) involved, collecting over three months of continuous monitoring more than 5 millions keystrokes.

In [4] we showed experimentally that, on the average, typing samples of different texts provided by the same individual are more similar than typing samples of the same text provided by different individuals. Thus, it was shown that keystroke analysis of free text, though more difficult than keystroke analysis of fixed text, can still be achieved.

In this paper we perform a further step, and show that it is possible to identify a user through the way he types on a keyboard, even when the user is entering free text in a language different from the one used to form his profile. Such ability is important since, for example, more and more people writing text with a computer may use their own language when communicating with others understanding the same language, but use English as the "Lingua Franca" to communicate with the rest of the world.

As we showed in [11], typing dynamics may provide meaningful information to improve the accuracy of an Intrusion Detection System, and may help to limit the number of false alarms. Thus, being able to deal with typing dynamics regardless of the language in use provides a double advantage. On the one hand, a legal user is free of entering text in the language she prefers, without particular risks of raising more alarms: the ability of the system to acknowledge her as the legal owner of the account under observation will not be affected by the use of a different language. On the other hand, intruders would not find any benefit by trying to disguise themselves using a language different from the one normally used by the intruded user: the system will not be fooled by the typing rhythms of a language different from the one of the user's profile.

As far as we know, this is the first work showing that keystroke analysis can be used to ascertain personal identity even when different languages are involved.

2 Computing the *Distance* Between Two Typing Samples

We will use the combination of two measures to evaluate the similarities and differences between the typing rhythms "recorded" in two samples we want to compare. We introduced the first measure, d_1, in [3]. The second measure, d_2, is described here for the first time. The only timing information we use in our experiments is the time elapsed between the depression of the first key and the depression of the second key of each digraph. We call such interval the $duration$ of the digraph. If the typed text is sufficiently long, the same digraph may occur more than once. In such case, we report the digraph only once, and we use the mean of the duration of its occurrences.

Given any two typing samples **S1** and **S2**, each one turned into digraphs and sorted with respect to duration of such digraphs, we define the distance between **S1** and **S2**, d_1(**S1**,**S2**), as the sum of the absolute values of the distances of each digraph of **S2** w.r.t. the position of the same digraph in **S1**. When computing d_1(**S1**,**S2**), digraphs that are not shared between the two samples are simply removed. It is clear that, from the definition of d_1, we may compute the distance between any two typing samples, provided they have some digraphs in common, even if written in different languages. As an example, in the left part of the Table 1 we report typing samples **E1** and **E2** obtained typing, respectively the texts *mathematics* and *sympathetic*. Only digraphs shared between **E1** and **E2** are actually shown. Numbers beside digraphs are their typing speed in milliseconds. The right part of the table illustrates pictorially the computation of the distance between **E1** and **E2**. From the figure it is easy to see that: d_1(**E1**,**E2**) = 3+0+0+1+4 = 8.

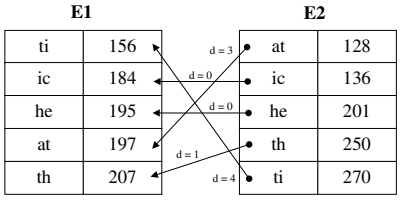
Given any two typing samples, the maximum distance they may have is when the shared digraphs, sorted by their typing speed, appear in reverse order in one sample w.r.t. the other sample. Hence, if two samples share N digraphs, the maximum distance they can have is given by: $N^2/2$ (if N is even); $(N^2-1)/2$ (if N is odd).

The above value can be used as a normalization factor of the distance between two typing samples sharing N digraphs, dividing their distance by the value of the maximum distance they may have. In this way it is possible to compare the distances of pairs of samples sharing a different number of digraphs: the normalized distance d_1(**S1**,**S2**) between any two samples **S1** and **S2** is a real number between 0 and 1. Measure d_1 returns 0 when the digraphs shared by the two samples are exactly in the same order w.r.t. their duration, and returns 1 when the digraphs appear in reverse order (d_1(**S1**,**S2**) is also set to 1 if S1 and S2 do not share any digraph). In our example, **E1** and **E2** share 5 digraphs. Thus, their normalized distance is $8/[(5^2-1)/2] = 0.66666$. From now on, in the paper we will always use the normalized version of d_1.

Distance d_1 performed very well to identify users through their typing rhythms on fixed text, and we refer to [3] for a thorough description of the measure and its properties. Readers may have noticed that d_1 completely overlooks any absolute value of the timings associated to the samples. Only the relative positions (which is a consequence of the typing speed) of the digraphs in the two samples are taken into consideration.

Table 1. Computation of the distance for typing samples **E1** and **E2**

E1		**E2**
156	**ti**	270
184	**ic**	136
195	**he**	201
197	**at**	128
207	**th**	250



However, even the actual typing speed at which digraphs are entered can be useful to discriminate between different individuals. For example, users A and B may both type the word *on* more slowly than the word *of*, but if the average typing speed of the two words are, for user A, say: *on* = 127 millisec.; *of* = 115 millisec.; and for B: *on* = 239 millisec.; *of* = 231 millisec., than A and B can hardly be the same individual.

To take care of such situations, we introduce a second distance measure, d_2, based on the actual typing speeds of digraphs. We could just consider the average typing speed of samples entered by the user, but since we want to combine this new distance with d_1, we prefer to use a measure that considers the average typing speed of single digraphs, and that is normalized in the interval [0..1]. We define d_2(**S1**,**S2**) be the number of digraphs shared by **S1** and **S2** whose typing speeds do not differ for more than 30%,[1] divided by the total number of digraphs shared by **S1** and **S2**. For example, in the case of samples **E1** and **E2**, it is easy to check that d_2(**E1**,**E2**) = 2/5 = 0.4.

Finally, to combine together distances d_1 and d_2 we simply define d(**S1**,**S2**), the distance between any two samples **S1** and **S2** that will be used in all the experiments described in this paper, as: d(**S1**,**S2**) = d_1(**S1**,**S2**) + d_2(**S1**,**S2**).

3 Experiments in User Identification and User Authentication

To perform the experiments described in this paper, we asked 31 volunteers to provide two typing samples written in Italian and two typing samples written in English. All the people participating to the experiments are native speakers of Italian, and, though with varying typing skills, all of them are well used to type on normal computer keyboards. Moreover, all volunteers are more or less used to write in English, since they are colleagues and PhD students.

People provided the samples from their computer, through an HTML form with a text area of 780 characters to be filled by the users and submitted to the collecting server. A client side Javascript was used to record the time (in milliseconds) when a key was depressed, together with the ascii value of the key.

[1] In order to chose this "30% rule", at the same time trying to limit overfitting, we did the following. When the first five volunteers of our experiments had provided their samples, we performed the identification task described in Section 3, in order to test different percentages: 10%, 20%, 30% and 40%. The best outcomes were reached using a 30% rule, and thus this value is used in all the experiments of this paper. It is of course possible that better outcomes could be reached for some other values (say, 15% or 33%), but we did not bother to find such particular values, that would hardly perform in a similar way on a different set of users.

Volunteers were instructed to enter the samples in the most natural way, more or less as if they were writing an e-mail to someone. They were completely free to choose what to write, and the only limitations were of not typing the same word or phrase repeatedly in order to fill the form, and not to enter the same text in two different samples. People were free to make typos, and to correct them or not, using the backspace key or the mouse, as preferred. People were free to pause in every moment when producing a sample, for whatever reason and as long as they wanted. No sample provided by the volunteers was rejected, for any reason.

In our approach, a user's profile is simply made of a set of typing samples provided by that user. Hence, suppose we are given a set of users' profiles and a new typing sample from one of the users, so that we want to identify who actually provided the sample. If the measure d defined in Section 2 works well, we may expect the computed distance between two samples of the same user to be smaller than the distance between two samples coming from different users. As a consequence, we may expect the mean distance of a new sample X from (the samples in) the profile of user U to be smaller if X has been provided by U than if X has been entered by someone else.

Hence, suppose we have three users A, B and C, with, say, 3 typing samples each one in their profiles (so that, for example, A's profile contains typing samples A_1, A_2 and A_3). A new typing sample X has been provided by one of the users, and we have to decide who entered the sample. We may compute the mean distance (md for short) of X from each user's profile as the mean of the distances of X from each sample in the profile:

md(A,X) = (d(A_1,X) + d(A_2,X) + d(A_3,X))/3;
md(B,X) = (d(B_1,X) + d(B_2,X) + d(B_3,X))/3;
md(C,X) = (d(C_1,X) + d(C_2,X) + d(C_3,X))/3.

Then, we decide that X belongs to the user with the smallest mean distance among the three. This rule has been tested using all possible combinations of Italian and English samples in the profiles of the 31 volunteers, while one of the remaining samples is the one that must be identified. The outcomes of this experiment are reported in the "Identif. errors" columns of Table 2. Outcomes are grouped w.r.t. the number of samples in users' profiles, and are detailed w.r.t. the actual composition of the profiles. Right below each group we report the whole outcomes obtained for the corresponding group. Within brackets we indicate the numerical values that provide the corresponding percentages. For example, suppose there are 3 samples in users' profiles, two Italian samples and one English sample. In this case the system can be tested using the other English sample, for a total of 62 attempted classifications (since both English samples play, in turn, the role of testing sample). In this case all samples are correctly classified, with an identification error of 0.0%. When profiles contain one Italian sample and two English samples, the system makes 2 errors out of 62 attempts, for an identification error of 3.23%. On the whole, when there are 3 samples in users' profile, the system can be tested with 124 samples, and shows an error of 1.61%.

From the outcomes we see that the accuracy of the system increases with the number of samples in users' profiles. When profiles are made of just on sample, almost one out of three testing samples are not correctly classified, with an identification error

of 29.57%. But such value quickly shrinks to 6.18% when users' profiles contain 2 samples, and to 1.61% with 3 samples in the profiles.

Quite obviously, when profiles contain exactly one sample in a given language, testing samples are more easily classified correctly if they are written in the same language. We detail more in depth this in the left part of the table. For example, when profiles contain only one Italian sample, we have 12 identification errors out of 62 attempts when trying to classify the other Italian sample, but 49 errors out of 124 attempts when trying to classify the two English samples.

When users' profiles contain two Italian samples, testing samples are all written in English, but less than one out of 15 are not correctly classified, for an identification error of 6.45%. The identification error is larger when users' profiles contain two English samples, and the Italian ones must be classified. Presumably, this is due to the fact that when users are writing in a language different from their own, their particular typing traits tend to remain more hidden. By putting together outcomes of these two identification tasks, we get 12 identification errors out of 124 attempts, that is, less than 10% of mistakes when attempting to identify a typing sample written in a language different from the one used for the two typing samples in the profiles. We get the best outcomes when profiles contains samples written in both languages. In this case it is easier to correctly identify the testing samples, regardless of the language used to write them.

Table 2. Results in user identification and authentication for different compositions of profiles

samples in profiles	Identif. errors
1 Italian sample	(*Ita.*) 19.35% (12/62)
	(*Eng.*) 39.51% (49/124)
- - -	- - -
1 English sample	(*Ita.*) 32.26% (40/124)
	(*Eng.*) 14.52% (9/62)
1 sample	29.57%

samples in profiles	Identif. errors	k = 0.9		k = 0.8	
		IPR	FAR	IPR	FAR
2 Italian samples	6.45% (4/62)	2.07% (77/3720)	12.9% (8/62)	1.24% (46/3720)	16.13% (10/62)
2 English samples	12.9% (8/62)	2.07% (77/3720)	14.51% (9/62)	1.24% (46/3720)	17.74% (11/62)
1 Ita. + 1 Eng. sample	4.44% (11/248)	2.17% (323/14880)	5.65% (14/248)	1.44% (214/14880)	8.47% (21/248)
2 samples	6.18%	2.14%	8.33%	1.37%	11.29%
2 Ita.+1 Eng. samples	0.0% (0/62)	1.98% (147/7440)	0.0% (0/62)	1.07% (80/7440)	0.0% (0/62)
1 Ita.+2 Eng. samples	3.23% 2/62	2.02% (150/7440)	4.83% (3/62)	1.09% (81/7440)	6.45% (4/62)
3 samples	1.61%	1.99%	2.42%	1.08%	3.23%

The identification rule just described can be used to authenticate users simply by marking the samples with an identity: a new sample X claimed to come from user A is authenticated as belonging to A if md(A,X) is the smallest among all known users. Now, the system can be evaluated w.r.t. two kinds of mistakes it can make: 1) the *Impostor Pass Rate (IPR)*, which is the percentage of cases in which a sample X from an unknown

individual is erroneously attributed to one of the users of the system; 2) the *False Alarm Rate (FAR)*, which is the percentage of cases in which a sample belonging to some user is not identified correctly.

From the "Identif. errors" column of Table 2 it is easy to see that our system shows, e.g., an average FAR of 1.61% when users have in their profiles three samples: 2 samples out of 124 authentication attempts produce false alarms. But what about the IPR? If there are 31 users in the system, it is simply (100/31)% = 3.23%. In fact, an impostor unknown to the system, pretending to be a legal user U, has a chance out of 31 that the sample she provides is closer to U's profile than to any other profile known to the system. We may improve such *basic performance* by observing the following. Suppose again that we have 3 users A, B and C, with 3 samples in their profiles and a new sample X to be classified, so that we compute: md(A,X)=0.419025; md(B,X)=0.420123; md(C,X)=0.423223. As a consequence, X is classified as belonging to user A. However, suppose that the mean of the distances of the samples forming the model of A (denoted by m(A)) is:

$d(A_1,A_2)$ = 0.312378; $d(A_1,A_3)$ = 0.304381; $d(A_2,A_3)$ = 0.326024.
m(A) = (0.312378 + 0.304381 + 0.326024)/3 = 0.314261.

Then, we may expect another sample of A to have a mean distance from the model of A similar to m(A), which is not the case for X in the example above. Even if X is closer to A than to any other user's profile in the system, it should be rejected.

To deal with such situations, we restate the classification rule as follow: a new sample X claimed to belong to user A is classified as belonging to A if and only if:

1. md(A,X) is the smallest w.r.t. any other user B and
2. md(A,X) is *sufficiently* closer to m(A) than to any other md(B,X) computed by the system. Formally: md(A,X) $< m$(A) + $|k(md$(B,X) - m(A))| for any user B, and for some k such that $0 < k \leq 1$.

If a user A meeting the above rules does not exist, X is rejected. Clearly, different values for k provide different trade-offs between IPR and FAR. Smaller values of k will allow to reject more samples from impostors, but could cause more false alarms. For k = 1, we fall back to the plain classification rule.

The IPR and FAR columns of Table 2 reports the outcomes of the experiments in user authentication for two different values for k. Again, in brackets are the numerical values from which we computed the corresponding percentage. For example, when profiles contain two samples, the system can be tested 22320 times for attacks from impostors: the profile of each user, in turn, is removed from the system,[2] and the Italian and English samples of that (now unknown) individual are used to attack all users in the systems.[3] Hopefully, the system should reject the attacking samples. Moreover,

[2] Otherwise, the attacking sample will be very likely attributed to the attacking user.

[3] Thus, we have (31 attacking users)·(4 attacking samples)·(30 attacked users)·(6 different pair of samples in a user's profile) = 22320 impostors' attacks.

the system is tested 372 times with legal samples claimed to belong to the users who actually provided them.[4]

The outcomes clearly show the effect of the authentication rule in use. For $k = 0.8$ and three samples in users' profiles, the system shows an IPR of 1.08%, that is, about one third of the IPR of the basic classification rule with 31 legal users. The cost is in the worsening of the ability to identify legal users, since the FAR = 3.23%, is now twice that of the basic classification method. Note also that, from the FAR columns we see that English samples appear easier to authenticate correctly using Italian samples in the profiles than vice versa. A result that we already noted in the experiments on identification. On the contrary, the corresponding IPRs do not change in both cases.

4 Discussion and Applications

Beside the outcomes of the previous section, an additional evidence of the fact that personal identity can be ascertained through the analysis of typing rhythms even when different languages are involved can be obtained by considering the mean distances (md for short in the table) reported in the last but one row of Table 3 for the samples gathered in our experiments.[5]

Table 3. Mean distances between different groups of samples

md between the Ita. samples provided by the same individual	md between the Eng. samples provided by the same individual	md between Ita. and Eng. samples provided by the same individual	md between any two Ita. samples provided by different individuals	md between any two Eng. samples provided by different individuals	md between any Ita. and Eng. samples provided by different individuals
md=1.11131 (31) [141]	md=1.12666 (31) [150]	md=1.15948 (124) [123]	md=1.36223 (1860) [140]	md=1.37821 (1860) [139]	md=1.38149 (3720) [122]

From the values in the table we see that typing samples of different text and language provided by the same individual (column 3) are, on the average, more similar than typing samples of different text but same language provided by different individuals (columns 4 and 5). Of course, even samples of different text and languages, coming from different individuals, have a larger distance between each other (column 6). Quite obviously, typing samples provided by the same individual in a certain language (columns 1 and 2), are more similar than typing samples provided by the same individual in different languages (column 3). But the mean of column 3 is only about

[4] In fact, we have (31 users)·(6 different pair of samples in a user's profile)·(2 testing samples) = 372 legal connections' attempts.

[5] Again, within round brackets we report the number of distances between samples used to compute the corresponding mean distance. For example, 62 English samples from different individuals allows to compute in (62·61)/2 - 31 = 1860 distances, where 31 is the number of comparisons between the two English samples provided by each volunteer.

4.24% greater than the mean value of column 1. On the contrary, the mean distance of typing samples written in the same language by different individuals (e.g., column 4) is about 16% greater that the mean distance between typing samples provided by the same individual in different languages (column 3). Thus, keystroke analysis involving different languages, though more difficult than when samples are all written in the same language, can still be achieved.

We also note that it is the combination of distances d_1 and d_2 that provides the good outcomes illustrated in the previous section. For example, when d_1 is used alone in the experiments in user identification, we get an identification error of 9.67% with 3 samples in users' profiles, and an error of 15.67% with 2 samples in users' profiles. When d_2 is used alone, the identification error is, respectively, 16.32% and 25.27%. The outcomes in user authentication worsen similarly when using only d_1 or d_2.

The accuracy of our method is related to the number of digraphs shared by the samples under comparison, as we showed in [3]. Samples written in different languages can be compared only if the two languages share some legal digraphs (That is, digraphs that occur in words belonging to the language). Within square brackets in Table 3 we report the average number of digraphs shared between any two samples of the corresponding columns. Samples of different languages (columns 3 and 6) share an average number of digraphs smaller than samples written in the same language. Note that English samples from the same user share a greater number of digraphs than Italian samples from the same user, probably because people tend to use a more restricted set of words when using a language different from their own. For a given length of the samples, the more similar the two languages, the larger the number of digraphs shared by the samples on the average, and the more accurate the distance between them returned by the distance measure used in this paper. Clearly, our method stops being useful when the languages involved (or just the samples under comparison) share a very small number of legal digraphs.

The outcomes of our experiments are among the best found in the literature about keystroke analysis of both free and fixed text, but one may wonder which is their statistical significance. A large amount of research on this issue, explicitly related to biometrics, is available, and we refer to [20] for a comprehensive treatment of the subject (or see [4] for a review of different available techniques). However, J. L. Wayman, Director of the U.S. National Biometric Test Center notes in [20] *our inability to predict even approximately how many tests will be required to have 'statistical confidence' in our results. We currently have no way of accurately estimating how large a test will be necessary to adequately characterize any biometric device in any application, even if error rates are known in advance.* In practice, the number of individuals and samples collected to test a system are not determined by pre-defined confidence intervals, but by the amount of time, budget and resources available [19]. Once test data has been collected and used on the system, it is then possible to estimate the uncertainty of the observed error rates with different methods, but such estimates will have to be taken with a grain of salt, due to the many sources of variability that affect biometric features [15]. We agree with the above view: especially in the case of an unstable biometric such as keystroke dynamics, the only way to evaluate a system is to test it in real conditions, with as many individuals as possible. The number of parameters that may influence

keystroke rhythms is so high that any statistical evaluation of the system outcomes will very likely be of limited use.

We conclude this section by proposing possible applications of keystroke analysis of free text.

Intrusion detection. The generation of false alarms is an endemic problem within intrusion detection [2]. In principle, keystroke analysis can be used to notice possible anomalies in the typing dynamics of individuals connected to the system, that may be intruders. However, the inaccuracy of the analysis may itself be the source of false alarms or undetected intrusions. On the contrary, if keystroke analysis is used conjunction with other techniques, it may be useful to mitigate the problem of false alarms, by providing an additional evidence of identity, as we showed in [11]. A scenario where keystroke analysis can be useful even used alone is when it is performed off-line, on accounts monitored in the recent past, to look for possible anomalies that could be simply reported to the system administrator. At the very least, the legal user of the account could be suggested (possibly by an automatic procedure) to change his/her password. In such case, even a relatively high FAR of, say, 2% or 3% would not be a serious problem: false alarms will simply make users changing their passwords a bit more frequently than usual.

Intrusions are often successful because no monitoring procedure is active, and because different form of intrusions are used. Hence, it is important to "attack the attackers" with different and complementary techniques, in order to improve the chances to detect them reliably and quickly. Experiments in this paper show that keystroke analysis can be a valid aid to intrusion detection even when individuals under analysis are using different languages.

User identification over the Internet. The ability to identify users through their typing habits can be used to achieve some form of User and Usage Modeling, in order to be able to offer personalized graphical interfaces, services and advertising to users on their return on a Web site visited previously [18]. Keystroke analysis would in particular be of great help to identify returning users of web sites that provide mailing lists, forums, chat lines and newsgroups access. The use of such services produces a large amount of typed text, whose typing rhythms can be stored and used to identify people on their return to the site, especially when no form of registration is required to visit the site and use its services. User identification over the Internet through the analysis of typing rhythms would find an interesting application also within the investigation of illegal activities that use the web (e.g., newsgroups and anonymous mailing services) to exchange information. For example, the analysis of the typing rhythms coming from different anonymous accounts and web connections could be useful to restrict and direct investigations on a subset of the individuals under observation.

It is worth to note that the above use of keystroke analysis may raise some concern about user's privacy. As a consequence, users should at the very least be informed that some form of monitoring is going on. One may observe that if a typing sample is stored only in term of the digraphs it is made, it would in general be pretty difficult to recover the original text. However, various kind of digit sequences entered, such as phone numbers, numerical passwords and pins, could be easy to recover, thus undermining users' privacy.

5 Conclusion

In this paper we have shown that keystroke analysis of free text can be a useful tool for user identification and authentication even when the typing dynamics stem from the use of different languages. As far as we know, such a situation has never been investigated before in the literature. Our outcomes have been obtained without any particular form of overfitting or tailoring of the system on the given data set, and our technique does not rely on the classical training-testing approach that may require the system to be tuned anew when a different set of users' profiles is involved. We used in our experiments typing samples relatively long, but we believe that, at the current state of the art, keystroke analysis of free text cannot be performed with very short samples: timing analysis on such texts does not provide a sufficient amount of information to discriminate accurately among legal users. On the contrary, if relatively long sample texts are accepted, keystroke analysis can become a valid tool to ascertain personal identity.

The ability to deal with typing samples of different texts and languages improves the possibility of making computers safer and more able to fit personal needs and preferences. We believe keystroke analysis can be a practical tool to help implementing better systems able to ascertain personal identity, and our study represents a contribution to this aim.

Acknowledgements: We want to thank all the volunteers in our Department who contributed to our research.

References

1. J. Ashbourn. *Biometrics: Advanced Identity Verification. The Complete Guide*. Springer, London, GB, 2000.
2. S. Axelsson. The Base-rate Fallacy and the Difficulty of Intrusion Detection. *ACM Transactions on Information and System Security*, 3(3):186–205, 2000.
3. F. Bergadano, D. Gunetti and C. Picardi. User authentication through keystroke dynamics. *ACM Trans. on Information and System Security (ACM TISSEC)*, 5(4):1–31, 2002.
4. F. Bergadano, D. Gunetti and C. Picardi. Identity Verification through Dynamic Keystroke Analysis. *Journal of Intelligent Data Analysis*, 7(5), 2003.
5. S. Bleha, C. Slivinsky, and B. Hussein. Computer-access security systems using keystroke dynamics. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-12(12):1217–1222, 1990.
6. M. E. Brown and S. J. Rogers. User identification via keystroke characteristics of typed names using neural networks. *Int. J. of Man-Machine Studies*, 39, pages:999–1014. 1993.
7. P. Dowland, and S. Furnell. A Long-term Trial of Keystroke Profiling using Digraph, Trigraph and Keyword Latencies. In *Proc. of of IFIP/SEC 2004 - 19th Int. Conf. on Information Security*, Toulouse, France. Kluwer, 2004.
8. P. Dowland, S. Furnell and M. Papadaki. Keystroke Analysis as a Method of Advanced User Authentication and Response. In *Proc. of of IFIP/SEC 2002 - 17th Int. Conf. on Information Security*, Cairo, Egypt. Kluwer, 2002.
9. R. O. Duda, P. E. Hart and D. G. Stork. *Pattern Classification*. John Wiley and Sons, 2000.
10. S. Furnell, J. Morrissey, P. Sanders, and C. Stockel. Applications of keystroke analysis for improved login security and continuous user authentication. In *Proc. of the Information and System Security Conf.*, pages 283–294. 1996.

11. D. Gunetti and G. Ruffo. Intrusion Detection through Behavioural Data. In *Proc. of the Third Symp. on Intelligent Data Analysis*, LNCS 1642, Springer-Verlag, 1999.
12. R. Joyce and G. Gupta. newblock User authorization based on keystroke latencies. *Comm. of the ACM*, 33(2):168–176, 1990.
13. J. Leggett and G. Williams. Verifying identity via keystroke characteristics. *Int. J. of Man-Machine Studies*, 28(1):67–76, 1988.
14. J. Leggett, Gl. Williams and M. Usnick. Dynamic identity verification via keystroke characteristics. *Int. J. of Man-Machine Studies*, 35:859–870, 1991.
15. A. J. Mansfield and J. L. Wayman. Best Practices in Testing and Reporting Performances of Biometric Devices. Deliverable of the Biometric Working Group of the CESG Gov. Communication Headquarters of the United Kingdom. National Physical Laboratory, Report CMCS 14/02. *Teddington, United Kingdom*, 2002. Report available at www.cesg.gov.uk/technology/biometrics/media/Best%20Practice.pdf
16. F. Monrose and A. Rubin. Authentication via keystroke dynamics. In *Proc. of the 4th ACM Computer and Communications Security Conf.*, 1997. ACM Press.
17. M. S. Obaidat and B. Sadoun. A simulation evaluation study of neural network techniques to computer user identification. *Information Sciences*, 102:239–258, 1997.
18. M. Perkowitz and O. Etzioni. Adaptive Web Sites: Conceptual Framework and Case Study. *Artificial Intelligence*, 118(1,2):245–275, 2000.
19. J. L. Wayman. Fundamentals of Biometric Authentication Technologies. In *Proc. of CardTech/SecurTech Conference*, 1999.
Available at www.engr.sjsu.edu/biometrics/nbtccw.pdf
20. J. L. Wayman (Editor). National Biometric Test Center: Collected Works 1997-2000. (Biometric Consortium of the U.S. Government interest group on biometric authentication) *San Jose State University, CA*. 2000.
Report available at www.engr.sjsu.edu/biometrics/nbtccw.pdf