World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:2, 2008

Learning User Keystroke Patterns for Authentication

Ying Zhao

Abstract**—** Keystroke authentication is a new access control system to identify legitimate users via their typing behavior. In this paper, machine learning techniques are adapted for keystroke authentication. Seven learning methods are used to build models to differentiate user keystroke patterns. The selected classification methods are Decision Tree, Naive Bayesian, Instance Based Learning, Decision Table, One Rule, Random Tree and K-star. Among these methods, three of them are studied in more details. The results show that machine learning is a feasible alternative for keystroke authentication. Compared to the conventional Nearest Neighbour method in the recent research, learning methods especially Decision Tree can be more accurate. In addition, the experiment results reveal that 3-Grams is more accurate than 2-Grams and 4-Grams for feature extraction. Also, combination of attributes tend to result higher accuracy.

Keywords**—** Keystroke Authentication, Pattern recognition, Machine Learning, Instance-based Learning, Bayesian, Decision Tree.

I. Introduction

GENERALLY speaking, an authentication process refers to three stages: access request, information extraction and authenticating. For instance, the conventional authentication system functions based purely on textual user name and password. It identifies the claimed identity by comparing to prefixed information which is stored as a valid user identity in database. Textual based authentication system remains dominant technique currently. However, it has shown to be a fairly weak security mechanism which has a high risk of information leak. It is reported that the successful impost rate can up to 25% through exhaustive search due to the choice habit [17]. Furthermore, the information can be lost in many ways. For example, it might be forgotten after a long time idle or it might be stolen. In contrast, authentication systems based on biometric characteristics currently become a very active research area. Biometric authentication is believed to be a new mechanisms with better scalability.

The term "Biometrics" is used to the emerging field of technology devoted to identification of individuals using biological features. Biometrics includes two categories, which are "physiological biometric" and "behavioral biometrics" [4]. Physiological biometrics such as fingerprints, iris scanning and face recognition include features which are stable and identical for individuals. However, behavioral biometrics are more flexible. For example, voice tune and typing pattern are typical behavioral biometrics, which varies even for the same individual. A biometric authentication system is essentially a pattern recognition system.

In this paper, we focus on keystroke authentication, a type of behavioral biometrics authentication. It is based on the hypothesis that, individuals type in a characteristic way on a keyboard. Although user keystroke pattern is not unique nature of a person, it is a sufficient characteristic to distinguish users. Physiological systems usually require advanced equipments such as specialized camera for facial geometry, scanner for fingerprint and infrared camera for facial thermogram. In contrast, a keystroke authentication system does not require any special devices.

A range of research has been done in this area, which has shown the feasibility of using keystroke biometrics for the authentication [6], [19], [20], [18]. It is reported that users' typing pattern is quite recognizable and relative repeatable. In addition, it is more difficult to be simulated intentionally. Although, an impostor might get the exactly textual identity information, such as "user name" and "password", the keystroke authentication system would fail the access attempts. Keystroke authentication systems provide a higher level of security protection. In addition, keystroke dynamics features can be used in conjunction with other mechanisms, such as generating a "harden password" [23], [17], [22].

Similarity measure is a challenging issue due to the highly dynamic and diverse nature of user input patterns. The conventional Nearest Neighbour distance measure might not be the best choice. Instead, machine learning is a good candidate to tackle this problem, because learning methods can usually construct models which are more adaptive than Nearest Neighbour.

Machine learning is a technique to automatically improve algorithms by extracting information from existing data, also known as experience [16]. It has been used in a wide range of areas such as image recognition, speech recognition and time series predication. These tasks are generally quite difficult in terms of data complexity. The successful applications of machine learning in these areas demonstrate its capability which makes us think it could also have good potential in keystroke authentication.

In this paper, we will explore keystroke authentication in a way similar to [5]. However learning methods are used and the data processing method is modified. The rest of this paper is organized in the following structure: section 2 describes the methodology including the data collection, data processing and learning methods; section 3 presents the experiments and results; section 4 and section 5 are the discussion of the issues of keystroke authentication and the conclusion respectively.

II. Methodology

A. Data Collection

Data collection is the first step and a critical step of our experiments. Due to the human ethics issue, the data used by other researchers are not available for sharing. To collect the data, several participants from different background were involved.

The text we used in this study is a part of an article called "Pumas at Large" from "New Concept English", an English

Ying Zhao is with the School of Computer Science and Information Technology, RMIT University of Australia, GPO Box 2476V, Melbourne 3001, Victoria, Australia (e-mail: yizhao@cs.rmit.edu.au).

textbook(see the footnote for the full text)[1]. There are two reasons to choose this text in our experiments. First, some of the users invited to participate our experiments are not native English speakers, thus such kind of easy reading text does not require a big vocabulary from these users and the chance of mis-spelling can be reduced. Second, the words in this text are quite diverse in length, which varies from single-character to 15-character. This kind of combination makes the text an suitable choice of authentication text.

The length of authentication text is an important factor. Presumably it should be long enough to contain sufficient information for differentiating user patterns. However excessively long text could be impractical to type and bring extra difficulties for later keystroke pattern matching. The text chosen in this study contains 664 characters including spaces and punctuation marks. It is about the same length with the text used in [5]. It is noticeable that this text only contains lower case characters. It is due to that we tried to avoid using unprintable keys at this stage. So keys such as "Caps Lock", "Shift" and "Ctrl" are not used.

The data collection program developed for this study captures three categories of information from users' typing behavior. They are Scan Code, System Time Stamp and Key Hold Time. A *scan code* is the key pressed by the user. A *system time stamp* records the beginning time of a key pressing. A *key hold time* shows how long a key has been pressed. Key hold time can also be called the duration of a keystroke. These time information is measured in system millisecond and stored as raw data used for later process. Users are allowed to have typing errors.

B. Data Processing

The purpose of such a transformation is to extract the system usable information from the raw collected keystroke data.

1) N-Grams: N-Grams refers to a combination of N keystrokes. Taking 3-Grams as an example, if the give text is *"pumas are "*[2] then seven 3-Grams can be generated from the text. They are *"pum"*, *"uma"*, *"mas"*, *"as "*, *"s a"*, *" ar"*, *"are"* and *"re "*. The reason of introducing N-Grams is to provide more varieties of user keystroke behaviors compared to single-key typing patterns. The elapse time, keystroke durations and latency of N-Grams can be computed form the raw data for further information extraction. In this study, 2-Gram, 3-Gram and 4-Gram are used and compared.

The captured raw data of a user pattern contains the scan code, the time stamp and the duration of each character typed by the user. For example, the raw data of the above text *"pumas are "* typed by a user is shown as below (Scan code/Time stamp/Duration):

```
80/0/156(p), 85/156/79(u),
77/344/62(m), 65/438/78(a),
83/641/125(s), 32/875/94(\),
65/1125/125(a), 82/1281/79(r),
69/1360/78(e), 32/1516/78(\ )
```

Based on these data, three kinds of information, the elapse time, keystroke duration and latency of each appeared N-Gram then can be calculated. A elapse time of a N-Gram is time from the first key being press to the $(N+1)$th key being pressed. It is actually the whole duration of that N-Gram. A keystroke duration is the sum of durations of N characters in the N-Gram. A latency is the sum of latency of each character in the N-Gram. The latency of a single character indicates how long from the key being released to the next key being pressed.

2) Typographical Error: Typographical errors are hard to avoid especially when the authentication text is lengthy. Research has been done and found out that typographical error rate was more than 24% [22]. A commonly used method in the literature is to remove the samples which contain typographical errors [6], [20]. However, it results a significant false alarm rate in test, which means the access from a genuine user would be denied if the user typed the authentication text but with some typing errors.

In contrast, the occurrence of typographical errors are allowed in our approach. The data collection program detects such errors by comparing the user input with the authentication text. However a user can correct the errors during the typing. In our opinion, allowing typing errors is more practical because it reduces the chance of re-typing. Furthermore, rejecting patterns with typing errors would result inadequate data for the experiments.

3) Information Extraction: Keystroke is of behavioral biometrics, which is relatively unstable compared to other physical biometrics, such as fingerprint, iris scan and face geometry. In the literature, most researchers analyze the time sequences measurement directly using an assumed kernel Gaussian distribution [6], [14], [7], [3]. However, the intrinsic variability of users' typing behavior remains as a problem in this research area. Using three keystroke patterns to illustrate this issue, the elapse time of the first four 2-Grams are shown below:

```
Data one: 1:273.5, 2:281, 3:453.5, 4:320
Data two: 1:257.5, 2:343, 3:484.5, 4:344
Data three: 1:289, 2:367,3:391, 4:515
```

It can be seen that even for the 2-Grams of same text from a same user, the elapse time varies quite noticeably. Although the elapse time is inconsistent caused by inconsistent typing speed, the order of these times are of less change. When the elapse times are sorted, the order of the elapse time are now much more regular. After sorting the elapse time listed above, the orders are shown in below. Instead of elapse time, a 2-Gram is represented by its index in the original order. For example the elapse time 320 in "Data one" is indexed as 4 and placed at the third position after sorting.

```
Data one: 1, 2, 4, 3
Data two: 1, 2, 4, 3
```

[1] *pumas are large, cat like animals that are found in america. when reports came into london zoo that a wild puma had been spotted forty five miles south of london, they were not taken seriously. however, as the evidence began to accumulate, experts from the zoo felt obliged to investigate, for the descriptions given by people who claimed to have seen the puma were extraordinarily similar. the hunt for the puma began in a small village where a woman picking blackberries saw 'a large cat' only five yards away from her. it immediately ran away when she saw it, and experts confirmed that a puma would not attack a human being unless it is cornered.*

[2] *There is a space key at the end after letter "e".*

```
Data three: 1, 2, 3, 4
```

The sorted indices is more reliable than the absolute time measure. By sorting and converting to indices, the variability of user patterns is significantly reduced. However there is still one issue, which is the high dimensionality of the processed patterns. In our experiments, there were 662 3-Grams generated. After removing the duplicates there are still 445 3-Grams. Such a high dimensionality would cause difficulties in the learning. So information need to be extracted to reduce the dimensionality. Such a process is also known as feature extraction.

The information extracted are the distances of a pattern to other patterns. Several distance functions have been introduced [9], [1], [15]. In this study p-norm distance is used, which can be expressed like:

P-norm Distance

$$D_{ij} = \sqrt[p]{\sum_{i=1}^{n}(y_i - x_i)^p}$$

Choosing p-norm distance is due to its low computational complexity and relatively good performance reported in the literature. It is desirable that a keystroke authentication system can response user input in real-time. So the distance measure should not be too expensive to compute. In p-norm distance, if the value of p is 1 then it is Manhattan distance, called as 1-norm distance as well. If p equals 2 then it is Euclidean distance, or 2-norm distance.

One distance measure is a single value. Therefore the dimensionality of pattern data is significantly reduced. Both Manhattan distance and Euclidean distance were computed for elapse time and duration of a pattern. So a processed user pattern contains only four values, Manhattan distance for elapse time, Manhattan distance for duration, Euclidean distance for elapse time and Euclidean distance for duration. In the context of learning, a pattern can be viewed as a vector of four dimensions.

C. Learning Patterns

Machine learning methods are used to differentiate user patterns in this paper. Generally speaking, machine learning has two broad fields, supervised learning and unsupervised learning. In supervised learning, each sample of data is provided with the expected output. By analyzing sufficient amount of sample data, some kind of knowledge can be generalized and can be applied to produce output for unseen samples. Supervised methods requires human intervention to label the samples before learning. In contrast of supervised learning, such kind of human assistance is not required in unsupervised learning methods. It is less suitable for the task of differentiating keystroke patterns. In our study, only supervised learning methods are use.

There are three well known categories in classification methods, statistical learning, rule based learning and tree based learning. In choosing learning methods, couple of methods are selected from each category. Instance based learning, Naive Bayesian Classifier and K-star are typical statistical methods. OneR and Decision Table are chosen to represent rule based methods. C4.5 and Random Tree are tree based. A brief description of each method is given below.

1) Instance Based Learning: Instance based learning measures the distance between a new pattern with its surrounding patterns. The class of the new pattern is It is suitable for data which have complex boundaries between the different classes [2]. It is worth mentioning that the number of surrounding patterns chosen for classification is an adjustable parameter in instance based learning. If this number is set as one, then it becomes a standard Nearest Neighbour method which has been used in other recent keystroke authentication research.

2) K-star: K-star can be considered as a variation of instance based learning which uses an entropic distance measure [10]. To compute the distance between two samples, the concept of "complexity of transforming from one sample into another sample" is introduced. A k-star distance is then defined by summing over all possible transformation paths between two distances. This approach can be applied on real numbers as well as symbolic data.

3) Bayesian Classifiers: Bayesian classifiers are based on Bayes probability theorem [13], [11]. There are several variations of Bayesian classifiers. Among them, Naive Bayes classifier is a highly practical learner. Further, it has been frequently reported as a competitive algorithm for real world applications. So it is used in our study as well. Naive Bayesian classifier uses an independent assumption which refers to that the attribute values a_1, $a_2 \ldots a_n$ of a sample are independent.

$$P\left(a_1, a_2 \ldots a_n | v_j\right) = \Pi_i \; P\left(a_i | v_j\right)$$

Derived from Bayesian theorem with mentioned assumption, we can write Naive Bayesian Classifier in this way:

$$V_n = argmax_{v_j \in V} \; P\left(v_j\right) \Pi_i \; P\left(a_i | v_j\right)$$

Here, V_n denotes the result that achieved by Naive Bayesian Classifier. The key issue of using Bayesian classifier is to get the probabilities of and $P\left(a_i | v_j\right)$. However, it is very difficult to estimate the probabilities of $P\left(a_i | v_j\right)$ from limited data set.

Associate to the given Naive Bayesian formula above, Gaussian distribution is used to compute $P\left(a_i | v_j\right)$:

$$p\left(a_i | v_j\right) = g\left(x, \mu, \sigma\right)$$

$$g\left(x, \mu, \sigma\right) = \frac{1}{\sqrt{2\pi}\sigma} \; e^{\frac{(x-\mu)^2}{2\sigma^2}}$$

4) OneR: OneR is a simple rule based learning method. It only uses one attribute to build the classifier. In some extend it is similar to the Nearest Neighbour method. However it does not measure distance but building rules. The rule with the highest accuracy on training data is selected as the classifiers. OneR can be used to determine whether there is a dominant attribute associated to the classes [8].

5) Decision Table: Decision Table can be considered as the extension of the basic OneR idea to several attributes. A generated decision table contains two parts: a schema and a body that consists a set of features and labeled instances. When a new patten is given, a decision table classifier searches for exact matches using only the feature in the schema [12].

TABLE I
COMPARISON OF LEARNING METHODS (AVERAGE OF 5,000 SPLITS, 3-GRAMS)

Learning Methods	Training	Test Accuracy
C4.5 Decision Tree	95.6%	93.3%
Naive Bayesian	93.3%	90.8%
K-star	100%	85.6%
Decision table	95.6%	81.1%
Random Tree	100%	77.8%
OneR	91.3%	75.2%
IB KNN		
($k = 8$)	90.2%	87.4%
($k = 7$)	91.1%	89.4%
($k = 5$)	93.3%	91.1%
($k = 1$)	100%	81.5%

6) C4.5: C4.5 is a well known tree based learning method. It generates a decision tree by analyzing the information gain and ratio of attributes. Based on such a measurement, an attribute with higher information gain is selected as the top of the tree. The same process is recursively used to generate the branches of the tree. When a new pattern is applied, it goes through the tree until a leaf node of tree is reached. The label of the leaf node is then the class of this given new pattern [21].

III. EXPERIMENTS AND RESULTS

A. Training and Test

In general there are two steps, training and test. Training is a process to learn a model. To evaluate the performance of a generated model, some unseen data should be introduced to test that model. This process is known as test.

During the training, the learning method tries to map the attributes with the assigned class as accurate as possible. In other words, it seeks the correlation between patterns and their users. The performance of such a mapping can be measured during training, which is the training accuracy. The performance of such a mapping done by the generated classifier on unseen data is the test accuracy. In our experiments, the accuracies are measured as the percentage of keystroke patterns which are correctly mapped to the user they actually belong to.

To measure training and test accuracy, the processed data are split into two parts, training data and test data. The ways of generating these two sets of data can be leave-one-out, cross validation and random split. The last one was used in the experiments due to the limited data set. 66% of user patterns are randomly selected from the processed data to form training data. The rest of data are left for test.

However a single split might not be able to truly reflect the performance of a learning method in training and test, because the split could be not random enough to make training data and test data representative. For example, a model learned from a training data could be untransferable to test data or a model with poor performance on training could luckily have high accuracy in test. Therefore each learning process was repeated 5000 times in our experiments. The training accuracy and test accuracy are the averages of the 5000 runs. By doing that, the randomness of split is ensured. The accuracies should be more reliable than that from a single split.

1) Comparing Classifiers: To compare the learning algorithms mentioned in Section II-C, each of these methods were applied on the processed user keystroke patterns. For instance-based learning, several k values were chosen which were 1, 5, 7 and 8. As mentioned before, with $k = 1$ this method is virtually Nearest Neighbour methods. So this set of experiments compared learning methods with the conventional method used in keystroke authentication.

The training and test accuracies obtained by these methods on 3-Grams data are shown in Table I. Except IBK, the employed methods are sorted against their test accuracies. The test accuracy achieved by Nearest Neighbour was 81.5%. Most of the learning methods achieved better performance than that. The highest accuracy, 93.3%, was obtained by C4.5 decision tree. Within instance based learning, using multiple surrounding neighours as reference points seems better than using a single nearest neighbour. The accuracies of $k = 5, 7, 8$ were all higher that of $k = 1$. However it does not mean that more neighbours would result higher accuracy. The highest accuracy of IBK was actually achieved by $k = 5$, not 7 or 8.

2) Comparing Combinations of Attributes: The data used in the previous experiments contain four attributes, Manhattan distance of elapse time of 3-Grams, Manhattan distance of durations of 3-Grams, Euclidean distances of elapse time and durations. It would be desirable to know that whether all these attributes are needed. Less attributes mean less computational cost for processing raw keystroke patterns.

The accuracy of OneR shown in Table I was 75.2%. Such a low accuracy of OneR indicates that there was no dominant attribute strongly associated to the classes. So one rule was not enough to achieve high accuracy. Therefore one attribute is almost certainly not a good choice. In the following experiments, four kinds of combinations of two attributes were selected to perform learning to investigate whether two attribute could contribute equivalent accuracies as four attributes did.

The experiment results are shown in Table II. The leftmost column lists the test accuracies of using four attributes obtained by different methods. The column titled with "M_e+E_e" lists the test accuracies of using Manhattan distance of elapse time and Euclidean distance of elapse time as two attributes. The other columns list the test accuracies of two distance of durations, Manhattan distances of elapse time and duration, Euclidean distances of elapse time and duration. Not all the methods listed in Table I were used in this set of experiment. The two most accurate ones, C4.5 and Naive Bayesian, were chosen as well as IBK. It can be seen from the results that the highest accuracy was achieved by using four attributes. It is true for all methods used here. The worst results were from the two distances of elapse times. It is possibly due to the elapse time is less stable, so the two distances of elapse time were not good enough to contribute an accurate classification.

3) Comparing N-Grams: The experiments been done so far are based on information extracted from 3-Grams. However 3-Gram might not be the optimal choice of N-Gram. To verify that, 2-Gram, 3-Gram and 4-Gram were compared. N values greater than 4 were not used because the possible Grams would be too many to be manageable. For each N of Gram, four attributes were generated which were the two distances of

TABLE II
COMPARISON OF THE NUMBER OF ATTRIBUTES

Algorithm	4-Attributes.	$M_e + E_e$	$M_d + E_d$	$M_e + M_d$	$E_e + E_d$
Naive Bayesian	90.8%	39.6%	72.9%	81.1%	84.6%
J48 Decision Tree	93.3%	43.8%	70.8%	82.2%	87.9%
IB KNN					
$(k = 7)$	89.4%	39.6%	77.3%	82.9%	83.3%
$(k = 5)$	91.1%	45.8%	77.1%	85.7%	86.9%
$(k = 1)$	81.5%	43.8%	62.5%	77.7%	80.1%

TABLE III
COMPARISON OF N-GRAMS

Algorithm	2-Grams	3-Grams	4-Grams
Naive Bayesian	82.8%	90.8%	75.6%
J48 Decision Tree	83.7%	93.3%	80.3%
IB KNN			
$(k = 7)$	79.8%	89.4%	75.6%
$(k = 5)$	84.2%	91.1%	82.1%
$(k = 1)$	73.1%	81.5%	71.1%

elapse time and durations.

The results of comparison are shown in Table III. Similar to Table II, only C4.5, Naive Bayesian and IBKs were used in the experiments. Only test accuracies of each learning process are listed. The table shows that the data extracted from 2-Grams achieved higher accuracy than that of 4-Grams, and the highest accuracy was obtained by using data extracted from 3-Grams. This kind of results is consistent with all learning methods.

IV. DISCUSSION

The experiment outcomes demonstrate that most of the learning methods outperformed Nearest Neighbour method (IB$k = 1$). It holds true in all our experiments whether it is for comparing methods(Table I), comparing combinations of attributes(Table II) or comparing N-Grams (Table III). The results suggest that machine learning approach can be a good substitute of conventional Nearest Neighour in keystroke authentication.

The rule based learning, such as Decision Tree and OneR, might not be the good choices. Instead, C4.5 decision tree, Naive Bayesian and instance based learning performs better. The highest test accuracy in all experiments was 93.3%, achieved by C4.5. However that does not mean that it is the best method for learning keystroke patterns. In some experiments Naive Bayesian outperformed C4.5. For example, Naive Bayesain was more accurate in "$M_d + E_d$" column in Table II (72.9% vs. 70.8%). In some experiments IB$K = 5$ seemed better than C4.5, such as column "2-Grams" in Table III (84.2% vs. 83.7%). In further work, especailly when more data are involved, all these three methods are worth trying and tuning to achieve best performance.

Much prior researches has investigated how to improve the performance of keystroke authentication system. In their approaches the time information were directly used. This approach is unstable due to the intrinsic variability of users' typing behavior. The time measurement is too sensitive. It could vary dramatically even for a same user, as illustrated in Section II-B.3. Using this kind of data could be difficult for learning methods. So data transformation was employed as described in Section 2. The experiment results show that these data transformation processes are suitable. By using them, the variability and the dimensionality of keystroke patterns were reduced. These processes make machine learning methods more feasible on our keystroke authentication tasks.

Accuracy is not the only way to measure the successfulness of a keystroke authentication method. Precision and Recall can also be In the experiment of using C4.5 decision tree to classify four attributes generated from 3-Grams, the test accuracy was 93.3%. The average precision over all users was 94.2%, that the average receall of that was 92.5%. Such results are considered reasonable. The percision was higher than the recall, which means that the chance of accepting a wrong user is lower.

There are a few issues of future development are discussed with the observation from the experiments:

- Our investigation of learning keystroke patterns produced positive outcomes. However there is still a large space for improvement before commercialization. Due to the high expense of collecting data, our current data set is limited. Although random split was used in experiments to overcome this issue, more data are certainly highly desirable.
- Besides N-Grams other information extraction methods would be investigated, such as using a word as a unit set or using a particular pattern as a unit. Instead of measuring elapse time or durations, characteristics of a user keystroke pattern could be extracted from histogram information or frequency information.
- Currently a set of typing conditions is required in data collection to increase the stability of data. In a real world application, these conditions are hard to meet. So further investigation is needed to explore how to handle unstable typing behavior. A threshold or a filter might be required for data processing to enhance the stability of user keystrokes.
- In our work the occurrence of typing error is allowed, but not used. If the patterns of errors are consistent, then they can assist decision making in recognizing users.

V. CONCLUSION

In this paper, we present a methodology of learning keystroke patterns for user authentication. It includes converting raw data into N-Grams, calculating elapse time, duration and latency, extracting information by applying Manhattan

distance and Euclidean distance and learning classifiers. The results indicate that learning user keystroke patterns for authentication is a feasible approach. Compared to the conventional Nearest Neighbour method, learning methods especially C4.5 decision tree achieved much better results, 95.6% of training accuracy and 93.3% of testing accuracy.

The results also show that decision tree, Naive Bayesian and instance based methods are more suitable for learning keystroke patterns. Furthermore data processing is a necessary procedure which is to generate data suitable for learning process by increasing the stability and reducing the dimensionality of user keystroke patterns. The experiment outcomes suggest that 3-Grams is a better way of converting raw data than 2-Grams and 4-Grams. The outcomes also demonstrate that the combination of four attributes can result a better performance than the combinations of two attributes.

Overall, our study of keystroke authentication is at its early stage. This investigation shows machine learning as a promising direction. There are many issues needed to be explored in our future work. However the current results already confirm the feasibility of learning approach and provide evidence of the capabilities of machine learning in keystroke pattern recognition.

ACKNOWLEDGMENT

A thank you to the participants of our experiments who contributed their valuable time to provide data for this research. Further, thanks to and WEKA learning environment, which helped us to realize our approaches.

REFERENCES

[1] C. C. Aggarwal. Re-designing distance functions and distance-based applications for high dimensional data. *SIGMOD Rec.*, 30(1):13–18, 2001.

[2] S. Aha and D. Kibler. Instance based learning algorithms. *Machine Learning*, 6:37–66, 1991.

[3] L. C. Arajo, L. H. S. Jr., M. G. Lizrraga, L. L. Ling, and J. B. Yabuuti. User authentication through typing biometrics features. In *First International Conference on Biometric Authentication*, volume 3072, pages 694–700. Springer, 2004.

[4] Ashbourn and D. M. Julian. *Biometrics: Advanced Identity Verification: The Complete Guide*. Springer-Verlag UK, 2000.

[5] F. Bergadano, D. Gunetti, and C. Picardi. User authentication through keystroke dynamics. *ACM Trans. Inf. Syst. Secur.*, 5(4):367–397, 2002.

[6] S. Bleha, C. Slivinsky, and B. Hussien. Computer-access security systems using keystroke dynamics. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 12, pages 1217–1222, 1990.

[7] S. Haider, A. Abbas, and A. Zaidi. A multi-technique approach for user identification through keystroke dynamics. In *IEEE International Conference on Systems, Man and Cybernetics*, volume 2, pages 1336–1341, 2000.

[8] R. C. Holte. Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11:63–91, 1993.

[9] D. P. Huttenlocher and K. Kedem. Computing the minimum hausdorff distance for point sets under translation. In *Proceedings of the sixth annual symposium on Computational geometry*, pages 340–349. ACM Press, 1990.

[10] C. G. John and T. E. Leonard. K*: An instance-based learner using an entropic distance measure. In *Proceedings of the 12th International Conference on Machine Learning*, pages 108–114. Morgan Kaufmann, 1995.

[11] G. H. John and P. Langley. Estimating continuous distributions in bayesian classifiers. In *Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345. Morgan Kaufmann Publisher, 1995.

[12] R. Kohavi. The power of decision tables. In *Proceedings of the 8th European Conference on Machine Learning*, volume 912 of LNAI, pages 174–189. Springer, 1995.

[13] P. Langley and S. Sage. Tractable average-case analysis of naive bayesian classifiers. In *Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 220–228. Morgan Kaufmann Publisher, 1999.

[14] D.-T. Lin. Computer-access authentication with neural network based keystroke identity verification. In *IEEE Transaction on Neural Networks*, volume 1, pages 174–178, 1997.

[15] U. V. Luxburg and O. Bousquet. Distance–based classification with lipschitz functions. *J. Mach. Learn. Res.*, 5:669–695, 2004.

[16] T. M. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.

[17] F. Monrose, M. K. Reiter, and S. Wetzel. Password hardening based on keystroke dynamics. In *Proceedings of the 6th ACM conference on Computer and communications security*, volume 6, pages 73–82. ACM Press, 1999.

[18] F. Monrose and A. Rubin. Authentication via keystroke dynamics. In *Proceedings of the 4th ACM conference on Computer and communications security*, volume 2, pages 48–56. ACM Press, 1997.

[19] M. S. Obaidat. A verification methodology for computer systems users. In *Proceedings of the 1995 ACM symposium on Applied computing*, pages 258–262. ACM Press, 1995.

[20] M. S. Obaidat and B. Sadoun. Verification of computer users using keystroke dynamics. In *IEEE Transaction on Systems, Man and Cybernetics, part B*, volume 27, pages 261–269, 1997.

[21] R. Quinlan. *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann, 1993.

[22] J. Robinson, V. Liang, J. Chambers, and C. MacKenzie. Computer user verification using login string keystroke dynamics. In *IEEE Transaction on Systems, Man and Cybernetics, Part A*, volume 28, pages 236–241, 1998.

[23] D. R. W.G and E. J.H.P. Enhanced password authentication through fuzzy logic. In *IEEE Expert*, volume 12, pages 38–45, 1997.