Keystroke Authentication using Ant Colony Optimization Algorithm

Seham Bamatraf
Information System Department
Faculty of Computers and Information
Cairo University- Egypt

Mohammed Bamatraf
Computer Science Department
Faculty of Science
Hadhramout University- Yemen

Abstract—Keystroke dynamic provides a confidence rate for user authentication throughout the work session. In this paper we propose a novel approach to enhance the accuracy rate for user authentication. The approach is based on combining both a sequence alignment and ant colony algorithm. The ACO algorithm is adapted to comprehend the keystroke dynamic variables. The proposed algorithm is evaluated against the traditional classifiers models to measure the accuracy and precision rates of identification.

Keywords—Keystroke dynamic; security; sequence alignment; Ant Colony Optimization

I. INTRODUCTION

The importance of security in information systems is self-evident. With the development of networks and increase in the computers power, enhance the security has become very interesting. However, the traditional user name and password authentication method reflects several shortcomings such as the ability to recognize the password by hackers if it is easy otherwise if the password is more complex the user should to be aware of his password. Although the commonly used biometric authentication technologies, such as finger print recognition and iris recognition have overcome the traditional username and password, many problems were emerged. For example, cost, implementation complexities, and decrease the degree of user’s acceptance. However, keystroke dynamic is one of the biometric authentications but provides several advantages that overcome the problems of general biometric technologies. Keystroke dynamics provides higher security with less cost to authenticate users based on their typing behavior on the keyboard or keypad. The behavior is measured using dwell time (time duration to press the key) and flight time (time duration between key up and key down). Keystroke dynamics methods proved a useful tool for user authentication[1]. The method maintains a profile for each user with his features and applies data mining classification techniques to identify users based on their profiles. Different classification techniques have been utilized such as neural network [2], pattern recognition [3], and recently ACO algorithms [4].

Ant Colony Optimization (ACO) algorithms have been derived from the living nature of ants in their colony. In a colony of social insects, such as ants, bees, wasps and termites, each insect usually performs its own tasks independently from the other members of the colony. However, these insects act together without any kind of supervisor or centralized controller to solve the important survival-related problems. ACO [5] [6] is mainly oriented for solving the combinational optimization problems such as travelling salesman person (TSP), vehicle routing, and telecommunication networks. The main purpose is to find good paths through graphs.

Recently, many works have been proposed to exploit ACO in machine learning and bioinformatics areas. ACO can provide an optimization solution in many problems such as learning classification rule. The researchers focused to inspire rules based on the various behaviors of ants. Additionally, ACO is exploited in sequence alignment problems [7]. The sequences are aligned based on ACO algorithm to improve time and quality of alignment process.

In this paper we aim to enhance the efficiency of accuracy of our previous proposed approach [8] for user authentication by using ACO algorithm. However, the previous work achieved high accuracy but suffered from the local-minima trap which is raised from using global sequence alignment. In contrast, ACO was employed in the literature [9] to find global solutions based on its diversity in chosen solutions. For this reasons, in this work we enhance the performance of our previous NM&W keystroke dynamic by exploiting the ACO benefits. In summary, our new proposed approach is divided into several stages starting by converting the keystroke features (dwell time and flight times) of users into sequences. Next we arrange the sequences similar to DNA sequences to find the matches degree of sequences using a customized similarity matrix rather than BLOSUM [10]. However, the matrix is developed from the traditional BLOSUM in parallel with keyboard characters. The final stage of our proposed approach applies the ACO strategy to perform classification and identification users.

The remainder of this paper is organized as follows: Section 2 presents a background about some related techniques and metrics in keystroke and the related works of ACO. Section 3 presents the proposed method. Section 4 presents the experimental and results, and Section 5 concludes the paper.
II. REVIEW AND BACKGROUND

A. Classification Technique

As said previously, keystroke dynamics method maintains a profile for each user with his features and applies data mining classification techniques to identify users based on their profiles. Recent researches create a user profile (template) using a combination of matrices and made analysis statistics to compute the degree of disorder of keystroke latency and duration [11]. Other approach [12] uses a variation of typing sequences. Additionally, a classification technique is also applied to predict the class of users. In essence, classification is considered an important task in data mining approaches and utilized in several commercial and industry applications [13]. In general, the classification task involves with discovering knowledge to predict the class of unknown instances objects in the search space problem. Essentially, the task uses a data set which is divided into training set used to build classification model and test set used to predict classes of unknown instances. The classification model is built by using a classification algorithm which is applied on the training set where their classes are defined previously. For each training instance, the algorithm analysis the relationship between predictor attributes and class. Finally, the algorithm finishes the analysis by discovering a classification model which is applied later to the instances in the test set. The discovered classification model is evaluated according to the predictive accuracy of classes. In the essence, the accuracy of prediction is measured as follows. The predicted class of an instance of test data is validated by comparing by the actual class of the instance. As a result, the predictive accuracy is commonly computed by dividing the number of instances which are correctly classified by the total number of text set instances.

The classification of users was performed based on their templates that were extracted during the acquisition data stage. Several classifications methods were presented include statistical methods to compute the mean and the standard deviation for templates. Early, Joyce and Gupta [14] presented an authentication method based on absolute distance. This method generated error metrics approximately 25% for FAR and 16.36% for FRR. Additionally, Guven and Sogukpinar [15] used vector analysis as a statistics method to authenticate users. Their method achieved about 95% accuracy to authenticate users.

Other proposed methods for classification used neural network. Obaidat and Macchiarolo [16] proved that the hybrid sum-of-products method achieved an efficient identification rate with approximately 97.8% accuracy. Another approach [2] suggested to use weightless neural network for classification. Later, Saevanee and Bhattarakosol [17] used probabilistic neural network and achieved about 99% for identification rate. Although the accuracy is very high but in neural network it is so difficult to choose the important features in classification. This makes a real problem in continuous keystroke where results are executed online.

The third trend of classification used in keystroke is pattern recognition. This means to identify objects and assign to different categories based on certain algorithms such as the nearest neighbor algorithm. Furthermore, several clustering methods were used like: Bayes classifier, SVM, and graph theory. For example, Giot et al. [18] proposed to use SVM for authenticates users and achieve about 95% identification rate. Recently, Balagani [19] used two classifications: k-nearest and ridge-LR and achieved 87.06% and 93.47% identification rate respectively. Learning methods provide a confidence value to determine a decision and very suitable to identify patterns online.

Finally, the last classification methods contain search heuristic to find an optimum solution. It is preferred in the case of large database. It can also provide multiple solutions. ACO and bioinformatics are examples of such approaches using genetic algorithm. Revett et al. [20] exploited the bioinformatics approach where the authors used multiple global sequence alignment [21] as a classification method to identify and authenticate system users based on their keystroke attributes (used three attributes: di-graphs, dwell times, and tri-graphs). The attributes were discretised into amino (ID/Password). The authentication sequence was then compared with the sequences in the database to get a score using a simple global alignment algorithm. The bioinformatics approach achieves about 0.1% for both FAR and FRR error metrics.

B. Sequence Alignment

This section we briefly give an overview of sequence alignment algorithms which will be used later in our proposed system.

Protein in molecule biology is the main component of RNA which composed DNA. Proteins have complex molecules and play an important role to support life activities: starting from respiration ending with thinking. The schema representation of protein structure involves of a sequence of amino acids connected with each other by a chemical substance. There are about 20 naturally amino acids depending on their specific combination. There are specific genes in all organisms which responsible the main performed functions in the organism. Founding the similarity between genes in the organisms has been the main task in the bioinformatics community. Such task refers to the pattern-matching or searching in the computer science view. The main goal is to find the similarity degree between two strings of genes in DNA.

In the essence of bioinformatics community, many works have been proposed to use sequence alignment algorithms to solve the pattern matching problem. A sequence alignment algorithm aims to compute an alignment score to measure the degree of convergence between input sequences. For example, when two sequences A and B are matched by aligned vertically: when a matched occurs at given position, 1 is added; otherwise 1 is subtracted from the whole computing score. Additionally, in the case of global alignment, gaps are introduced when the two sequences matched except some positions. It also serves as a tool to ensure the two strings (sequence) has the
same length. These gaps will significantly reduce the overall alignment score.

![Sequence Alignment Example](image1)

**Fig. 1 Sequence Alignment Example**

Figure 1 shows an example of two different alignments for two sequences `ABBTFGMG` and `ABTFTGKG`. In the alignment on the left-hand, there are 7 matches, 1 mismatch, and 1 gap. On the other hand, the second alignment on the right-hand, there are 5 matches, 3 mismatches and 1 gap. Of course the first alignment is better since it has higher score than other. However, many works have been proposed to determine the best possible alignment. Needleman and Wunch algorithm [21] (NM&W) is an efficient algorithm to determine the best score. The basic idea starts by the optimal alignment of the smallest possible subsequences (i.e. sequences X and Y have no alignment to each other). Next, an iteration process is started to determine the optimal score by proceeding in both sequences one position at a time. Whenever the iteration process ends, a traceback is started.

The traceback process keep tracks the optimal score for each possible subsequence in a matrix. The traditional pseudo code for NM&W algorithm is shown in Figure 2.

![Pseudo Code of Needleman-Wunch Alg.](image2)

**Fig. 2 Pseudo Code of Needleman-Wunch Alg.**

The iteration process ends when the return trip, the ant leaves pheromone trail on the ground which indicates the quality and quantity of the food and helps the other ants later to find the source of the food. In summary, the model reflects the ants’ foraging behavior as follows: initially all the ants place in nodes, then each ant moves from source node to food node. Finally, all ants conduct their return trip and reinforce their chosen path as outlined above. The pseudo code of ACO algorithm is shown in Figure 3. The basic idea is to simulate artificial ants to generate new solutions to the given problem. The first step initializes the pheromone. In the second step, ants incrementally build a solution to the optimization problem. The ant evaluates the candidate solution to decide how much deposit of pheromone in the update stage. The update pheromone modifies the pheromone on trails. The modification is either increasing the pheromone if the path is a good solution which will be used in the optimal solution. Otherwise, the pheromone is decreased to allow the exploration of new areas in the search space. The last step is the daemon procedure. This step is daemon action which is a centralized action performed to avoid lies in local optimization. One example of daemon actions is to observe the path that is found by each ant to decide which path is better. Consequently, a set of ants have been selected to perform the deposit additional pheromone on the connection.

![Pseudo Code of Metaheuristic ACO](image3)

**Fig. 3 Pseudo Code of Metaheuristic ACO**

Formally, the designed model [24] is defined as a graph $G$ of network, $G = \langle V, E \rangle$ where $V$ represents the nodes, and $E$ represents the edges connected the nodes. Two types of nodes distinguished in $V$ include $V_n$ for nest of the ants and $V_d$ for food source node. Edges also are distinguished into two types of links: $e_n$ and $e_d$ connected between nodes $V_n$ and $V_d$. Here $e_n$ holds the shortest path and $e_d$ holds the longest path between $V_n$ and $V_d$.

Real ants deposit pheromone on the tracks that walk out. The design of chemical pheromone trail is calculated as follow:
Let $\tau_i$ is an artificial pheromone value for two links $e_i$, $i = 1, 2$. By using the probability:

$$p_1 = \frac{\tau_1}{\tau_1 + \tau_2}, i = 1, 2$$

(1)

The ant is able to reach to the food source $v_2$ to choose one of two paths ($e_1, e_2$).

If $\tau_1 > \tau_2$ then the probability of $e_1$ is higher,

Else the probability of $e_2$ is higher.

For returning back to the nest node $V_n$, each ant will cross the same path with the change of artificial pheromone value $\tau_i$ as follow:

$$\tau_i = \tau_i + \frac{\rho}{l_i}$$

(2)

Where $\rho$ represents a constant value in the model and $l_i$ denotes to the length of the path. The added amount of artificial pheromone is inversely proportional to the length of the path. That is, the shorter of the paths chosen, the higher the amount of added pheromone.

On the other hand, the pheromone trails evaporate over time. This pheromone evaporation is simulated in the artificial model as follows:

$$\tau_i = (1 - \rho) \times \tau_i, i = 1, 2$$

(3)

The parameter $\rho \in (0, 1]$ is a parameter that regulates the pheromone evaporation.

### III. THE PROPOSED AUTHENTICATION MODEL

In a previous work we addressed the problem of keystroke data representation and classification that improve the accuracy identification of users. In this section we describe these steps [8] in detail to make the paper self-contained. Next, we discuss the ACO algorithm employment to enhance the performance.

The main data set used for simulation is the set user keystroke dynamics described below. To make sense of this data it is first fuzzified using the membership function given in equation (1). This phase termed as the data representation phase, which is the first phase of the model. Later it is applied to the proposed model described in the subsection below.

#### A. Data Representation Phase

Features are collected during the login process. These features include the duration of a keystroke, key hold time, and latency. Whenever the data is collected a clean process is started to eliminate any outlier data by taking the median and standard deviation of the data. Next, a fuzzy-logic is used where the data values are represented as a sequence of characters based on the range of speed. For this purpose, we use a membership function and described in Figure 4.

The membership function assigns the range times into fuzzy sets (the times in the range 2-3 msec are part of a set of set very fast) and can be represented by character $A$. and so on.

![Fig. 4 Pseudo Code of Membership Function](image)

In the case of large ranges such as [2-8] msec, the range is assigned into 6 fuzzy sets as can be evidently computed from Equation (4). Each character is measured based on the gap that exists between a pair of characters. For example, the “A” has a minor difference from letter “B” compared with far letter such as “D”. Such consideration make an important role when computing the penalty and the score using NM&W algorithm as will be described in next section.

At the end of this phase, we get a set of resulted sequences which are then used as unique patterns for each user (reference signature).

#### B. Computing Sequence Similarity

In this phase, NM&W algorithm is employed to compute the similarity degree between users. Before using NM&W algorithm, we create a customized matrix to compute similarities rather the traditional BLOSUM matrix. The new matrix consists of the arrangement alphabetic letters for rows and columns. Each intersection is assigned by a value represents the similarity degree between the intersected pair letters. Several measures can be considered such as the alphabet arrangement, distance between letters in the keyboard, and so on. Here we use the alphabet arrangement for example the similarity degree between letters A and A takes the largest similarity degree, and the similarity degree between letters A and B will be less than (A, A), however the degree of similarity for letters that are far then their similarity will be less.

Consider the two strings keystroke fuzzy sequences to be globally aligned are:

$q_1, ..., sequence_1, i = 0..n$

$q_j, ..., sequence_2, j = 0..m$

Each sequence consists of a set of alphabet characters. Three steps are performed to compute the score using classical NM&W algorithm with slight modification includes initialization, scoring, and traceback. The initialization step involves creating an $n$ by $m$ scoring matrix and initialized by multiplying each cell by the value of gap penalty. The scoring step compute the cells based on three
cells up, left, or diagonally. Finally, traceback step is conducted to find the optimal sequence alignment.

C. Proposed System for Keystroke using ACO

The proposed model is innovative rather inventive. The model employs the ACO algorithm as the core element in the proposed model. Unfortunately, ACO algorithm in its original construct can’t handle all types of data. The proposed model reconstructed the ACO algorithm to handle the fuzzified keystroke sequences from one side; it also introduced a customized selection and pheromone update based on the NM&W alignment algorithm as shown in Figure 5.

As shown in Figure 5 the core structure is the ACO. Nodes represent user trials, and the routes are decided on the generated score by every ant, the pheromone value is updated accordingly. Firstly, nodes are initially set to random unlabeled values (sequences/vectors), from the training set. Then a set of ants represents user trials set are used to train the system and select best routes for the user.

Ants usually cooperate of communicate by pheromones. Traditional ACO algorithm uses the probability measure to check the level similarity to select a specific path formula equation (1).

Here, the decision is made based on the similarity score between the ants and nodes using the set of steps of sequence alignment of NM&W algorithm which is described in previous section. The pheromone is also updated similarly using equation 2 in Section II.

D. Training and Rule Extraction

For the graph $G = < V, E >$ where $V$ is consists of two sets of nodes namely $V_a$ and $V_f$. $V_a$ represents the set of ants, $r$ is the number of exemplars or trials for the target user. $V_f$ is the destination or food node set, where $r$ is a selected number estimated based on the number of target rules that could represent the class; the minimum set of nodes with high entropy and can represent the class are selected to be the classification rule set for the class.

$E$ is the set of edges connects $V_a$’s, say $e_1$, and $e_2$ is an edge followed to reach a target $V_f$. The set of edges $E$ receives the pheromone from the ants while ant movement. Thus, the pheromone values are modeled as follows.

For a pheromone value $\tau_i$ for every pair of $V_a$’s this value indicates the strength of the target pheromone on the target path.

As shown in Figure 5 the core structure is the ACO. Nodes represent user trials, and the routes are decided on the generated score by every ant, the pheromone value is updated accordingly. Firstly, nodes are initially set to random unlabeled values (sequences/vectors), from the training set. Then a set of ants represents user trials set are used to train the system and select best routes for the user.

Ants usually cooperate of communicate by pheromones. Traditional ACO algorithm uses the probability measure to check the level similarity to select a specific path formula equation (1).

Here, the decision is made based on the similarity score between the ants and nodes using the set of steps of sequence alignment of NM&W algorithm which is described in previous section. The pheromone is also updated similarly using equation 2 in Section II.

The pseudo code for adaptation of ACO algorithm in keystroke dynamic is shown in Figure 6. Here no return trip is conducted from the ants, which will definitely safe time.

Finally, the optimal set of solutions represents the set of rules can be used to identify the user. This process is iteratively repeated for every user, coming up with set of rules to be tested and evaluated as discussed in the experimental results section.

IV. THE EXPERIMENTS AND RESULTS

In the section we investigate the impact of using ACO to enhance user authentication in our purposed model. For this purpose we will conduct a set of experiments to measure the accuracy and precision of authentication against the stat-of-the-art classification methods.

A. Settings

The experiments used the benchmark data set which are presented by Killourhy and Maxion [25]. The data set was composed of timing information that were collected from 51 users (typists) each of which types a specific
accuracy = \frac{T_{Pos} + T_{Neg}}{T_{Pos} + T_{Neg} + F_{Pos} + F_{Neg}} \times 100  
(5)

precision = \frac{T_{Pos}}{T_{Pos} + F_{Pos}} \times 100  
(6)

Whereas TPos represents the number of users who are truly accepted, TNeg reflects the number of illegal users who are not accepted. Meanwhile, FPos is the number of illegal users who are accepted. FNeg indicates the number of legal users who are not accepted.

All experiments are conducted on a PC running windows 7 with RAM 3 GB and CPU 2.2GHz. ACO algorithm is downloaded from [26] and we modified it to adapt our proposed keystroke dynamics method using visual studio c#.

B. Results

The accuracy and precision of our proposed system with ACO-based and our other traditional classification methods are listed in Table I. From Table 1, it can be seen that the accuracy rate of our both proposed models with and without ACO are more effective than other classifications methods. The precision rate also has the same noticeable effect. This is due to several reasons. First, the traditional classification methods did not work well with the nominal data which using in our experiments. However, the performance may different when using the real data. Second reason also concern with the traditional classification methods which are based on the whole attributes of the users. This leads to reduce the chance to find the similarity between users.

Table I Classifiers Results using Weka Tool

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>BayesNet</td>
<td>50.5</td>
<td>50.5</td>
</tr>
<tr>
<td>NaiveBayes</td>
<td>50.8</td>
<td>50.7</td>
</tr>
<tr>
<td>Kstar</td>
<td>46.8</td>
<td>47.2</td>
</tr>
<tr>
<td>Id3</td>
<td>35.1</td>
<td>35.2</td>
</tr>
<tr>
<td>J48</td>
<td>39.7</td>
<td>40.12</td>
</tr>
</tbody>
</table>

Other reasons concern with our proposed systems where a heuristic search is used to find more optimum solutions and avoid the local-minima trap that is raised by using NM&W algorithm in our previous method. This explains the superiority of our proposed system with ACO above the NM&W alone.

1. Conclusion

In this paper, we explain using of ACO in user authentication based on keystroke dynamic. The proposed system consists two phases: training phase and testing phase. During the training phase, we construct a classifier model by customizing a graph for each user and generating sequence nodes to compute score using NM&W algorithm. In the testing phase, users are authenticated based on the classification rules constructed in the training phase. The proposed system improved performance by 41 % and 32% for precision and accuracy compared to the traditional classification models and about 1% and 2% compared to our previous classification method without using ACO.

References


