Journal of Engineering and Technology Research Vol.1 (5), pp.072-080, August, 2009
Available online at http:// www.academicjournals.org/JETR

Full Length Research

Feature subset selection in keystroke dynamics using ant colony optimization

Marcus Karnan[1]*, M. Akila[2] and A. Kalamani[1]

[1]Tamilnadu College of Engineering, Coimbatore, India.
[2]Vivekanandha College of Engineering for Women, Tiruchengode, India.

Accepted 23 June, 2009

The need to secure sensitive data and computer systems from intruders, while allowing ease of access for authenticated users is one of the main problems in computer security. Traditionally, passwords have been the usual method for controlling access to computer systems but this approach has many inherent flaws. Keystroke Dynamics is a relatively new method of biometric identification and provides a comparatively inexpensive and low profile method of hardening the normal login and password process. This paper presents the feature subset selection in Keystroke Dynamics for identity verification, and it reports the results of experimenting Ant Colony Optimization technique on keystroke duration, latency and digraph for feature subset selection. Here, the Ant Colony Optimization is used to reduce the redundant feature values and minimize the search space. Optimum feature subset is obtained using keystroke duration values when compared with the other two feature values.

Key words: Feature extraction, feature subset selection, mean and standard deviation, ant colony optimization algorithm, keystroke dynamics.

INTRODUCTION

Access to computer systems is usually controlled by user accounts with usernames and passwords. Such scheme has little security (Hu et al., 2008; Pavaday and Soyjaudah, 2007) if the information falls to wrong hands. Key cards or biometric systems, (Adrian et al., 2006; Gláucya et al., 2007; Anil et al., 2003; Duane et al., 2007), for example fingerprints (Lin and Anil, 1998) is being used nowadays to improve the security. Biometric methods measure biological and physiological characteristics to uniquely identify individuals. The main drawback of most biometric methods is that they are expensive to implement, because most of them require specialized hardware to strengthen security. On the other hand keystroke dynamics (Fabian and Aviel, 2000; Jarmo, 2003) consist of many advantages like (i) It can be used without an additional hardware (ii) Hardening the existing security and (iii) Remote access.

Keystroke analysis (Christopher et al. (2008) is of two kinds; Static and Dynamic. Static keystroke analysis essentially means that the analysis is performed on typing samples produced using the same predetermined text for all the individuals under observation. Dynamic keystroke analysis implies a continuous or periodic monitoring of issued keystrokes and is intended to be performed during a log-in session, after the authentication phase has passed.

One area where the use of a static approach to keystroke dynamics may be particularly interesting is in restricting source level access to the master server hosting a Kerberos (Gabriel et al., 2007) key database. Any user accessing the server is prompted to type a few words or a pass phrase in conjunction with his/her username and password. Access is granted if his/her typing pattern matches within a reasonable threshold of the claimed identity. This safeguard is effective as there is usually no remote access allowed to the server, and the only entry point is via console login. Alternatively, dynamic or continuous monitoring of the interaction of users while accessing highly restricted documents or executing tasks in environments where the user must be alert at all times (for example air traffic control), is a ideal scenario for the application of a keystroke authentication system. Keystroke dynamics may be used to detect uncharacteristic typing rhythm (brought on by drowsiness, fatigue etc.) in the user and notify third parties. Keystroke dynamics include several different measurements (Pin et al., 2007;

*Corresponding author. E-mail: karnanme@yahoo.com.

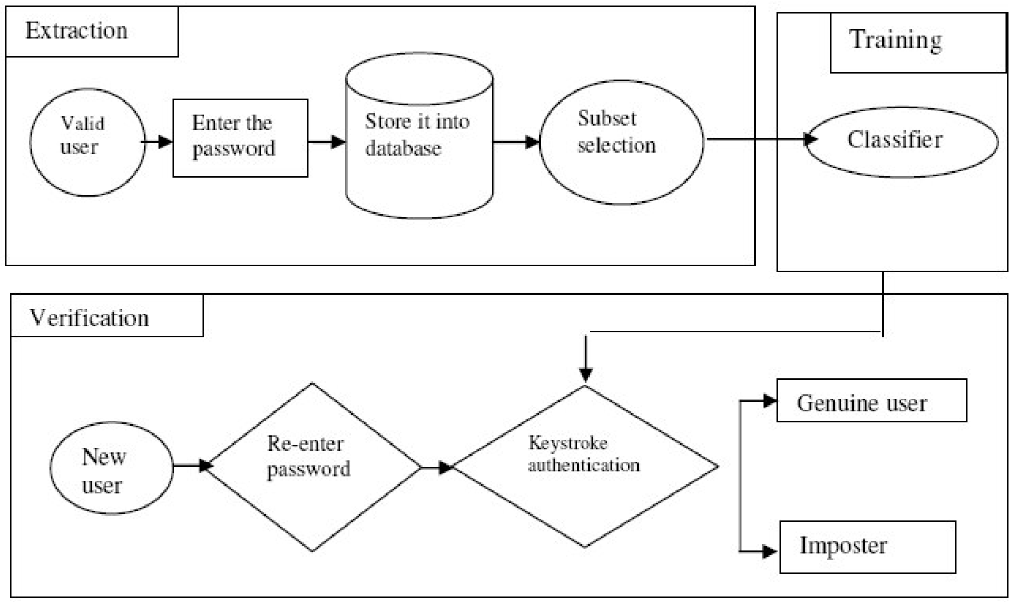

Figure 1. Keystroke dynamics analysis framework.

Shepherd, 1995)) such as:

1. Duration of a keystroke or key hold time.
2. Latency of keystrokes or inter-keystroke times.
3. Typing error.
4. Force keystrokes etc

There are two phases namely extraction phase and verification phase as in Figure 1.

During the feature extraction phase (Gláucya et al., 2007; Christopher et al., 2008; Gaines et al., 1980; Young and Hammon, 1989) user keystroke features from one's name or password are captured, processed and stored in a reference file as prototypes for future use by system in subsequent authentication operations. During the verification phase (Bleha and Obaidat, 1991; Daw-Tung, 1997) user keystroke features are captured, processed in order to render an authentication decision based on the outcome of a classification process of the newly presented feature to the pre-stored prototypes (reference templates) (Sylvain et al., 2005; Yu and Cho, 2004). It would be necessary for the user to type his/her name or password a number of times in order for the system to be able to extract the relevant features that uniquely represent the user. However, the task of typing one's name or password over and over is both tiring and tedious in the feature extraction phase, which could lead users to alter their normal typing pattern. Thus, most systems based on biometrics are required to work with a summarized set of information from which to extract knowledge. In order to reduce this problem, we could eliminate some features of the original dataset, selecting only the best ones in terms of class cohesion.

Feature subset selection (Yang and Honavar, 1998; John et al., 1994) is applied to high dimensional data prior to classification. Feature subset selection is essentially an optimization problem, which involves searching the space of possible features to identify one that is optimum or near-optimal with respect to certain performance measures, since the aim is to obtain any subset that minimizes a particular measure (classification error, for instance) (Shiv et al., 2007; Surendra and Huan, 2006). In order to reduce the complexity and to increase the performance of the classifier, the redundant and irrelevant features are reduced from the original feature set. Many feature subset selection (Karnan et al., 2006) approaches is proposed in the previous studies.

Related work

Major feature subset selection methods are detailed in this section. Yu and Cho (2004, 2003) propose a GA-SVM based wrapper approach for feature subset selection in which Genetic Algorithm (GA) is employed to implement a randomized search and Support Vector Machines (SVM), an excellent novelty detector with fast learning speed, is employed as a base learner. The degree of diversity and quality are guaranteed, and thus they gave result in an improved model performance and stability. Ki-seok and Sungzoon (2006) propose one step approach similar to that of Genetic Feature Selection (GEFS), yet with a more direct diversity term in the fitness function and SVM as base classifier and similar to that of Yu and Cho (2003), yet with a diversity term and no more post processing step. In particular, so called "uniqueness" term is used in a fitness function, measuring how unique each classifier is from others in terms of the features used. To adapt SVM authors, use Gaussian kernel. GA is used to filter the data and to carry out a selection of characteristics. It reports an average FAR of 15.78% with minimum FAR of 5.3% and maximum FAR of 20.38% for raw data with noise.

Gabriel et al. (2007a, b) designed a hybrid system based on SVM and Stochastic Optimization Techniques. Standard GA and Particle Swarm Optimization (PSO) variation was used which produced a good result for the tasks of feature selection with a FAR of 0.81% and IPR of 0.76%. Glaucya et al. (2007) used weighted probability measure by selecting N features of the features vector with the minors of standard deviation, eliminating the features less significant. They obtained optimum result using 90% of the features with 3.83% FRR and 0% FAR.

In Section 2, the feature extraction is discussed. Section 3 explains the method of feature subset selection. Section 4 discusses the results of the experiments and in Section 5 conclusion is presented.

FEATURE EXTRACTION

To capture a keystroke, it would be necessary for users to type their password a number of times. The system would set about capturing these features using three methods regarding the time (in milliseconds) that a particular user maintains the key pressed (Duration time), the time elapsed between releasing one key and pressing the next (latency time) and the combination of the above is called Digraph. The data was collected from 27 participants with different passwords using application software in .net framework. Each participant was asked to type his/her password 100 times. The mean and standard deviation values were measured as shown in Table 1.

The raw data file recorded by the application, measures the duration, latency and digraph timing information in milliseconds (ms) for the entered password. During the creation of the raw data file, the mean (μ) and standard deviation (σ) (Fabian et al., 1999; Francesco et al., 2002; Magalhaes et al., 2005) of each feature (i) of the pattern set (x) are calculated for N samples in agreement with the following equations:

$$\text{Mean } (\mu_i) = (1/N)\ \Sigma\ \mathbf{x}\ [i], \text{ where } i=1 \text{ to } N. \quad (1)$$

$$\text{Standard deviation } (\sigma_i) = (1/N\text{-}1)\ \Sigma\ |\mathbf{x}[i]\text{-}\mu\ [i]|, \quad (2)$$

Where i=1 to N.

For instance, for the password “ANT” the timing information for duration is (205, 250, and 235) ms. Figure 2 shows the measurement of duration, latency and digraph of keystrokes of the letter “AN” of the password “ANT”.

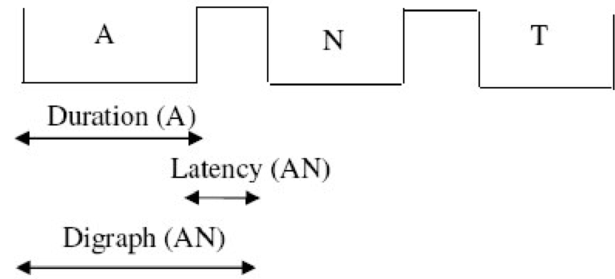

Figure 2. Measurement of duration, latency and digraph.

FEATURE SUBSET SELECTION

In the feature extraction phase, several samples are typed by user for n number of times. During the verification phase, it takes more time to verify all the n number of features. To reduce the time complexity we are using feature subset selection methods. In feature subset selection (Yu and Cho, 2004, 2003) method, we extract the optimized features from the n number of features. It is essentially an optimization problem, which involves searching the space of possible features to identify one that is optimum. Various ways to perform feature subset selection has been studied earlier. Here, we propose Ant Colony Optimization to select the feature subset.

Ant colony optimization (ACO)

Ant algorithms (David et al., 2007; Haibin et al., 2007; Thangavel et al., 2006; Dorigo and Gambardella, 1997) was first proposed by Dorigo and colleagues (Gabriel et al., 2007) as a multi-agent approach to difficult combinatorial optimization problems such as the Traveling Salesman Problem (TSP) and the Quadratic Assignment Problem (QAP). There are currently various activities in the scientific community to extend and apply anti-based algorithms to many different discrete optimization problems.

The ACO heuristic (Haibin et al., 2007; Dorigo et al., 1991; Youmei and Zongben, 2003) has been inspired by the observation on real ant colony’s foraging behavior, and that ants can often find the shortest path between food source and their nest. Ant individuals transmit information through the volatile chemical substances which ants leave in its passing path known as the “pheromone” and then reach the purpose of finding the best way to search food sources. An ant encountering a previously laid trail can detect the dense of pheromone trail. It decides with high probability to follow a shortest path, and reinforce that trail with its own pheromone. The large amount of pheromone is on the particular path, the large probability is that an ant selects that path and the paths pheromone trail will become denser. At last, the ant colony collectively marks the shortest path, which has the largest pheromone amount. Such simple indirect communication way among ants embodies actually a kind of collective learning mechanism. Figure 3 explain the working

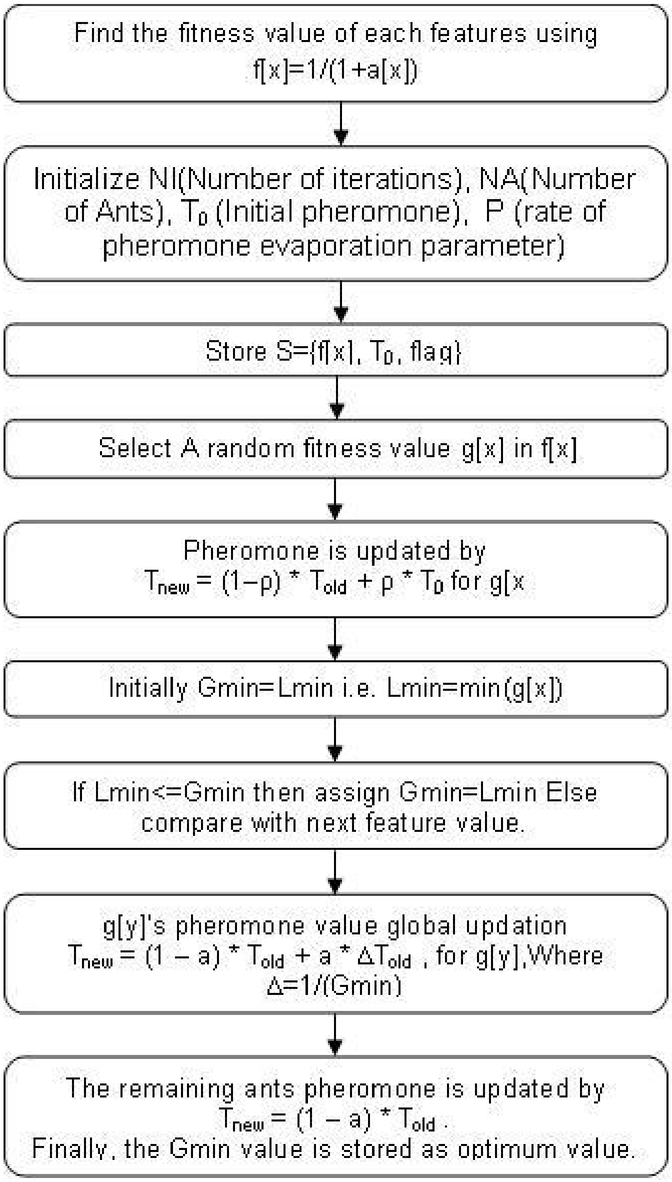

Figure 3. Ant colony optimization for subset selection.

procedure of ant colony algorithm in keystroke dynamics. The Ant Colony Optimization algorithm is as follows:

ACO algorithm:

Step 1. Get the feature value a[x] from duration/latency/ digraph of keystrokes.
Step 2. Calculate the fitness function f[x] by the following equation for every a[x].

F[x] =1/ (1+a[x])

Step 3. Initialize the following:

NI = 5 (Number of iterations)
NA = 2 (Number of Ants)
T_0 = 0.001 (Initial pheromone value for every a[x])
ρ = 0.9 (rate of pheromone evaporation parameter for every a[x])
Step 4. Store the fitness function values in S, where S = {F[x], T_0, flag} where flag column mentions whether the feature is selected by the ant or not.
Step 5. The following is repeated for NI times:

1) A random feature value g[x] in a[x] is selected for each ant with the criteria that the particular feature value should not have been selected previously.
2) Selected feature value's, pheromone value is updated by the following:

$T_{new} = (1-\rho) \times T_{old} + \rho \times T_0$ for g[x]

Where; T_{new} and T_{old} are the new and old pheromone value of the feature value.

3) Obtain Lmin = min (g[x]) where Lmin is the Local minimum.
4) Check if Lmin ≤ Gmin then assign Gmin = Lmin. Else no change in Gmin value where Gmin is the Global minimum.
5) Select the best feature g[y], whose solution is equal to the Local minimum value at the end of the last iteration.
6) The selected g[y]'s pheromone value is globally updated by the following equation:

$T_{new} = (1-\alpha) \times T_{old} + \alpha \times \Delta T_{old}$, for g[y],

Where; α is a rate of pheromone evaporation parameter, Δ=1/ (Gmin).

The remaining ants and their pheromone is updated as:

$T_{new} = (1-\alpha) \times T_{old}$

Where; α is a rate of pheromone evaporation parameter.

7) Finally, the Gmin value is stored as optimum value.

RESULTS AND DISCUSSION

In this section the results obtained from experiments is presented. Mean and standard deviation were measured for duration, latency and digraph for each sample. Ant colony algorithm is used for selecting the optimum feature for each participant and the selected features are considered for future classification. For instance, for the password "ANT" the calculated values using duration, latency and digraph as features and the Table 1 – 9 Considering the Duration (D), Latency (L) and Digraph (Di) values from user's keystroke profile: D [('A', 240),('N', 215), ('T', 235)], L [('AN', 275), ('NT', 235)] and Di [('A',515), ('N', 470), ('T', 235)]. The mean and result after applying ACO algorithm is displayed from standard deviation of Duration, Latency and Digraph are computed

Table 1. Before feature subset selection using duration as feature and calculation of mean and standard deviation.

Mean	Standard deviation	Fitness value		Local minimum		Pheromone update		Global minimum	
		Mean	STD	Mean	STD	Mean	STD	Mean	STD
230	12.9	0.00432	0.76394	0.00414	0.04670	0.00100	0.00100	0.00414	0.046707
240	20.4	0.00414	0.04670	0.00414	0.03286	0.00759	0.00759	0.00414	0.032862
170	29.4	0.00584	0.03286	0.00523	0.03286	0.00759	0.00759	0.00414	0.032862
190	8.16	0.00523	0.10917	0.00438	0.03096	0.00759	0.00759	0.00414	0.030960
227	31.3	0.00438	0.03096	0.00369	0.03041	0.00759	0.00759	0.00369	0.030414
270	31.8	0.00369	0.03041	0.00369	0.02144	0.00759	0.00759	0.00369	0.021441
195	45.6	0.00510	0.02144	0.00398	0.02144	0.00759	0.00759	0.00369	0.021441
250	16.3	0.00398	0.05773	0.00398	0.05773	0.00759	0.00759	0.00369	0.021441
220	4.08	0.00452	0.19681	0.00452	0.12391	0.00759	0.00759	0.00369	0.021441
200	7.07	0.00497	0.12391	0.00432	0.07639	0.00759	0.00759	0.00369	0.021441

Table 2. After feature subset selection using duration - mean.

Mean	Standard deviation	Fitness value		Local minimum		Pheromone update		Global minimum	
		Mean	STD	Mean	STD	Mean	STD	Mean	STD
270	31.8	0.00369	0.03041	0.00369	0.02144	0.00759	0.00759	0.00369	0.021441
250	16.3	0.00398	0.05773	0.00398	0.05773	0.00759	0.00759	0.00369	0.021441
240	20.4	0.00414	0.04670	0.00414	0.03286	0.00759	0.00759	0.00414	0.032862
230	12.9	0.00432	0.76394	0.00414	0.04670	0.00100	0.00100	0.00414	0.046707
220	4.08	0.00452	0.19681	0.00452	0.12391	0.00759	0.00759	0.00369	0.021441

Table 3. After feature subset selection using duration - standard deviation.

Mean	Standard deviation	Fitness value		Local minimum		Pheromone update		Global minimum	
		Mean	STD	Mean	STD	Mean	STD	Mean	STD
195	45.6	0.00510	0.02144	0.00398	0.02144	0.00759	0.00759	0.00369	0.021441
270	31.8	0.00369	0.03041	0.00369	0.02144	0.00759	0.00759	0.00369	0.021441
227	31.3	0.00438	0.03096	0.00369	0.03041	0.00759	0.00759	0.00369	0.030414
170	29.4	0.00584	0.03286	0.00523	0.03286	0.00759	0.00759	0.00414	0.032862
240	20.4	0.00414	0.04670	0.00414	0.03286	0.00759	0.00759	0.00414	0.032862

Table 4. Before feature subset selection using latency as feature and calculation of mean and standard deviation.

Mean	Standard deviation	Fitness value		Local minimum		Pheromone update		Global minimum	
		Mean	STD	Mean	STD	Mean	STD	Mean	STD
250	30	0.00398	0.03225	0.00398	0.03225	0.00100	0.00100	0.00398	0.03225
230	5	0.00433	0.16666	0.00432	0.09091	0.00759	0.00759	0.00398	0.03225
200	10	0.00497	0.09091	0.00362	0.01176	0.00759	0.00759	0.00369	0.01176
275	84.03	0.00362	0.01176	0.00362	0.01176	0.00759	0.00759	0.00362	0.01176
220	20	0.00452	0.04762	0.00452	0.04761	0.00759	0.00759	0.00362	0.01176
195	15	0.00510	0.06250	0.00473	0.06250	0.00759	0.00759	0.00362	0.01176
210	10	0.00473	0.09091	0.00452	0.09091	0.00759	0.00759	0.00362	0.01176
220	10	0.00452	0.09091	0.00369	0.02439	0.00759	0.00759	0.00362	0.01176
270	40	0.00369	0.02439	0.00369	0.02439	0.00759	0.00759	0.00362	0.01176

Table 5. After feature subset selection using latency – mean.

Mean	Standard deviation	Fitness value		Local minimum		Pheromone update		Global minimum	
		Mean	STD	Mean	STD	Mean	STD	Mean	STD
275	84.03	0.00362	0.01176	0.00362	0.01176	0.00759	0.00759	0.00362	0.01176
270	40	0.00369	0.02439	0.00369	0.02439	0.00759	0.00759	0.00362	0.01176
250	30	0.00398	0.03225	0.00398	0.03225	0.00100	0.00100	0.00398	0.03225
230	5	0.00433	0.16666	0.00432	0.09091	0.00759	0.00759	0.00398	0.03225
220	10	0.00452	0.09091	0.00369	0.02439	0.00759	0.00759	0.00362	0.01176

Table 6. After feature subset selection using latency - standard deviation.

Mean	Standard deviation	Fitness value		Local minimum		Pheromone update		Global minimum	
		Mean	STD	Mean	STD	Mean	STD	Mean	STD
275	84.03	0.00362	0.01176	0.00362	0.01176	0.00759	0.00759	0.00362	0.01176
270	40	0.00369	0.02439	0.00369	0.02439	0.00759	0.00759	0.00362	0.01176
250	30	0.00398	0.03225	0.00398	0.03225	0.00100	0.00100	0.00398	0.03225
220	20	0.00452	0.04762	0.00452	0.04761	0.00759	0.00759	0.00362	0.01176
220	10	0.00452	0.09091	0.00369	0.02439	0.00759	0.00759	0.00362	0.01176

Table 7. Before feature subset selection using digraph as feature and calculation of mean and standard deviation.

Mean	Standard deviation	Fitness value		Local minimum		Pheromone update		Global minimum	
		Mean	STD	Mean	STD	Mean	STD	Mean	STD
240	12.2	0.00831	0.07552	0.00831	0.07552	0.00100	0.00100	0.00831	0.0755
235	5.47	0.00847	0.15456	0.00847	0.04708	0.00090	0.00759	0.00831	0.04708
185	20.2	0.01082	0.04708	0.00885	0.04708	0.00090	0.00759	0.00831	0.04708
232.5	17.7	0.00885	0.05336	0.00885	0.04739	0.00090	0.00759	0.00831	0.04708
223.5	20.1	0.00891	0.04739	0.00879	0.04739	0.00090	0.00759	0.00831	0.04708
232.5	12.6	0.00879	0.07331	0.00879	0.07331	0.00090	0.00759	0.00831	0.04708
202.5	7.74	0.00984	0.11441	0.00860	0.07127	0.00090	0.00759	0.00831	0.04708
235	13.0	0.00850	0.07127	0.00821	0.75458	0.00090	0.00759	0.00821	0.04708
235	17.3	0.00821	0.05458	0.00497	0.75458	0.00090	0.00759	0.00497	0.04708
200	10.488	0.00497	0.08704	0.00497	0.07552	0.00090	0.00759	0.00497	0.04708

Table 8. After feature subset selection using digraph - mean.

Mean	Standard deviation	Fitness value		Local minimum		Pheromone update		Global minimum	
		Mean	STD	Mean	STD	Mean	STD	Mean	STD
200	10.488	0.00497	0.08704	0.00497	0.07552	0.00090	0.00759	0.00497	0.04708
235	17.3	0.00821	0.05458	0.00497	0.75458	0.00090	0.00759	0.00497	0.04708
240	12.2	0.00831	0.07552	0.00831	0.07552	0.00100	0.00100	0.00831	0.0755
235	5.47	0.00847	0.15456	0.00847	0.04708	0.00090	0.00759	0.00831	0.04708
235	13.0	0.00850	0.07127	0.00821	0.75458	0.00090	0.00759	0.00821	0.04708

as shown in Table 1 - 9. Computation of Duration for each letter of the password "ANT" and calculation of mean, fitness value, local minimum, pheromone update and global minimum is as follows: For instance if the

Table 9. After feature subset selection using digraph - standard deviation.

Mean	Standard deviation	Fitness value		Local minimum		Pheromone update		Global minimum	
		Mean	STD	Mean	STD	Mean	STD	Mean	STD
185	20.2	0.01082	0.04708	0.00885	0.04708	0.00090	0.00759	0.00831	0.04708
240	12.2	0.00831	0.07552	0.00831	0.07552	0.00100	0.00100	0.00831	0.0755
223.5	20.1	0.00891	0.04739	0.00879	0.04739	0.00090	0.00759	0.00831	0.04708
235	17.3	0.00821	0.05458	0.00497	0.75458	0.00090	0.00759	0.00497	0.04708
232.5	17.7	0.00885	0.05336	0.00885	0.04739	0.00090	0.00759	0.00831	0.04708

Duration = {A-240, N-215, T-235} ms.

Calculation of fitness value for duration is as follows:

Mean (μ_i) = (1/N) Σ x [i],

Where; i=1 to N = 240 + 215 + 235 / 3 = 230 = a[x]

Fitness value f[x] =1 / 1 + a[x] =1 / 1+ 230= 0.00432

Standard deviation (σ_i) = (1/N-1) Σ |x[i]-μ [i]|,

Where; i=1 to N =$\sqrt{}(230-240)^2+(230-215)^2+(230-235)^2/3$
(σ_i) = 12.9 = a[x]

Fitness value f[x] = 1/1+12.9 = 0.76394

Calculation of local minimum for duration

Initially the fitness value a[x] is directly assigned to Local Minimum (Lmin) for the first value (that is. a [1]). Then the next fitness value a[x] (that is. a [2]) is compared with the previous value already calculated. The minimum is found in them and is replaced with the Local minimum value.

For mean value a [1] = 0.00432
Assign a [1] = Lmin = 0.00432
After finding the next value a [2] = 0.00414
Check whether a [1] less then or equal to the value a [2].
If the condition is true,
Assign Lmin = a [1].
Otherwise, Lmin = a [2]
Here, in this example, if (0.00432 ≤ 0.00414)
Lmin = 0.00414
Similarly, for standard deviation,
A [1] = 0.76394 and Assign a [1] = Lmin = 0.76394
The second value a [2] = 0.04670
Check the condition, a [1] less then or equal to the value a [2].
Here, in this sample, if (0.76394 < 0.04670)
Lmin = 0.04670.

Calculation of local pheromone update for duration

T_{new} = (1 - α) x T_{old} + α x T_0, where T_{new} = new pheromone rate, T_{old} = old pheromone rate, T_0 = Initial pheromone value. Initially, T_{old} = 0.001, T_0 = 0.001.

For mean, first Local pheromone is updated as:

T_{new} = (1 - 0.9) x 0.001+ 0.9x0.001= 0.1x0.001+ 0.0009 = 0.00100.

Note: T_{old} value change due to the previous T_{new} value that is. T_{old} = 0.00100.

For standard deviation, Local pheromone is updated as

T_{new} = (1 - 0.9) x 0.001+ 0.9x0.001 = 0.1x0.001+0.0009 = 0.00100.

Calculation of global minimum for duration

Global minimum (Gmin) is assigned Lmin value initially (that is Gmin <= Lmin). Next the value in the Gmin is compared with Lmin, to find the minimum amongst them.

For mean, Lmin = 0.00414.
Initially, Gmin = Lmin that is. Gmin = 0.00414
For next feature value, the following condition should be satisfied for Gmin. that is. (Gmin ≤ Lmin).
Next, Lmin consist of next minimum value.
According to the sample next minimum value is the same value. This value is compared with Gmin, if the value is less than or equal to the Lmin. Therefore, Gmin = 0.00414.

For Standard deviation, Lmin = 0.04670;
Initially, Gmin = Lmin and now Gmin = 0.04670
Next Lmin = 0.03286
Compare with Gmin values, that is, (Gmin ≤ Lmin). .
Finally Gmin = 0.03286

Calculation of global pheromone update for duration

The selected Gmin, pheromone value is updated as follows:

T_{new} = (1 - α) x T_{old} + α x ΔT_{old}
α = rate of pheromone evaporation parameter, Δ =1/ (Gmin)

For mean, Global pheromone is updated as;

T_{new} = (1 - 0.9) x 0.001 + 0.9 x (1/0.00362) x 0.001 = (.0001) + (0.9 x 276.24 x 0.001) = 0.248716

For standard deviation, Global pheromone is updated as:

T_{new} = (1 – 0.9) x 0.001 + 0.9 x (1/0.01176) x 0.001 = (.0001) + (0.9 x 85.03401x0.001)
= 0.0766306

Calculation of pheromone value update for remaining ants

The remaining Ants pheromone value is updated as follows:

$T_{new} = (1 - \alpha) \times T_{old}$

For mean, Global pheromone is updated as:

T_{new} = (1 - 0.9) x 0.001=.1 x .001=.0001

For standard deviation, Global pheromone is updated as:

T_{new} = (1-0.9) x 0.001=.1 x .001=.0001.

Similarly the values for the latency and digraph are calculated as above.

The Table 1 - 9 was achieved by comparing the Ant Colony Optimization final subset values of duration, latency and digraph. We can see that the best result is presented from the ant colony duration value for the sample composed of 11 individuals.

Conclusion

In this paper, Mean and Standard Deviation is used to extract the features from the Keystroke duration, latency and digraph. From the extracted features, the optimized features are selected using the Ant Colony Optimization Algorithm to reduce the searching space. Better performance among these extraction methods is achieved with the Duration feature values. The experiments conducted in this work attempt to illustrate that ACO can be used in feature selection problems in keystroke dynamics.

REFERENCES

Adrian K, Pawel K, Piotr K (2006). Modern access control based on eye movement analysis and keystroke dynamics, Proceedings of the International Multiconference on Computer Science and Information Technology pp. 477-483

Anil J, Ling H, Sharath P (2003). Biometrics Identification, Signal Processing, Commun. ACM 83(12): 2539 – 2557.

Bleha SA, Obaidat MS (1991). Dimensionality reduction and feature-extraction applications in identifying computer users, IEEE Trans. Syst. Man Cybern. 21(2) 452 – 456.

Christopher S, Leberknight, George RW, Michael LR (2008). An Investigation into the efficacy of Keystroke Analysis for Perimeter Defense and Facility Access",: Proceedings of the IEEE Conference on Technologies for Homeland Security pp. 345-350.

David M, Manu DB, Raf H, Jan V, Monique S, Bart B (2007). Classification with Ant Colony Optimization 11: 651-665.

Daw-Tung L (1997). Computer-Access Authentication with Neural Network Based Keystroke Identity Verification, Proceedings of the IEEE International Conference on Neural Networks 1(9-12): 174 – 178.

Dorigo M, Gambardella LM (1997). Ant colonies for the traveling salesman problem Biosystems 43(2): 73-81.

Dorigo M, Maniezzo V, Colorni A (1991). Positive feed back as a search strategy, Technical Report Politecnico di Milano, 58426, Italy.

Duane B, Chris M, Brad W, Kim S (2007). Biometrics Overview, National Science and Technology Council Sub-Committee on Biometrics.

Fabian M, Aviel DR (2000). Keystroke dynamics as a biometric for authentication, Future Generation Computer Systems 16(4): 351 – 359.

Fabian M, Michael K, Reiter, Susanne W (1999). Password Hardening Based on Keystroke Dynamics, Proceedings of the 6th ACM conference on Computer and communications security, Publisher - Springer Berlin / Heidelberg, ISSN 1615-5262 (Print) 1615-5270 (Online) Kent Ridge Digital Labs, Singapore pp. 73- 82.

Francesco B, Daniele G, Claudia P (2002). User authentication through Keystroke Dynamics, ACM Transaction of information and system security, ISSN:1094-9224. 5(4): 367-397

Gabriel LFB, Azevedo, George DC, Cavalcanti, Carvalho F (2007). Hybrid Solution for the Feature Selection in Personal Identification Problems through Keystroke Dynamics, Proceedings of the International Joint Conference on Neural Networks pp. 1947 – 1952.

Gabriel LFB, Azevedo, George DC, Cavalcanti ECB, Carvalho F (2007). An Approach to Feature Extraction for Keystroke Dynamics Systems based on PSO and Feature Weighting, IEEE Congress on Evolutionary Computation Digital Object Identifier 10.1109/ CEC.2007. 4424936. 25(28): 3577 – 3584

Gaines R, Lisowski W, Press S, Shpiro N (1980). Authentication by keystroke timing: Some preliminary results, Rand Report R-256-NSF. Rand Corporation, Grand No. MCS76-00720.

Gláucya C, Boechat, Jeneffer C, Ferreira, Edson CB, Carvalho F (2007). Authentication Personal, Proceedings of the International Conference on Intelligent and Advanced Systems, ISBN: 978-1-4244-1355-3. pp. 254-256

Haibin D, Member, Xiufen Y (2007). Hybrid Ant Colony Optimization Using Memetic Algorithm for Traveling Salesman Problem, IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning, ISBN: 1-4244-0706-0. pp. 92-95.

Hu J, Gingrich D, Sentosa A (2008). A k-Nearest neighbor approach for user authentication through biometric keystroke dynamics, Proceedings of the IEEE International Conference on Communications pp. 1556 – 1560.

Jarmo I (2003). Keystroke Dynamics, http://www.it.lut.fi/kurssit/03-04/010970000/seminars/Ilonen.pdf

John GH, Kohavi R, Pfleger K (1994). Irrelevant features and the subset selection problem, W. W. Cohen and H. Hirsh (eds), Machine Learning: Proceedings of the Eleventh International Conference, Morgan Kaufmann, San Francisco pp. 121-129.

Karnan M, Thangavel K, Sivakumar R, Geetha K (2006). Ant Colony Optimization for Feature selection and Classification of Micro-calcifications in Digital Mammograms, IEEE International Conference on Advanced Computing and Communications pp. 298 – 303.

Ki-seok S, Sungzoon C (2006). "GA SVM Wrapper Ensemble for Keystroke Dynamics Authentication", Proceedings of the International Conference on Biometrics, Hong Kong, CHINE, ISBN: 978-3-540-31111-9. 3832: 654-660.

Lin H, Anil J (1998). Integrating Faces and Fingerprints for Personal Identification, IEEE Trans Pattern Anal. Mach. Intell. 20(12): 1295 – 1307.

Magalhaes, Paulo S, Henrique DD (2005). An improved Statistical Keystroke Dynamics Algorithm, proceedings of the IADIS- Virtual Multi-Conference on Computer Science, Lisboa ISBN: 972-8939-00-0.

Pavaday N, Soyjaudah KMS (2007). Investigating performance of neural networks in authentication using keystroke dynamics, AFRICON pp. 1 – 8.

Pin ST, Andrew BJT, Thian SO, Han FN (2007). Statistical Fusion Approach on Keystroke Dynamics, Proceedings of the Third Interna-

tional IEEE Conference on Signal-Image Technologies and Internet-Based System, Shanghai, ISBN: 978-0-7695-3122-9. pp. 918-923.
Shepherd SJ (1995). Continuous Authentication by Analysis of key-stroke typing characteristics, European convention on security and detection pp. 111 – 114.
Shiv SKN, Raj BS, Ravinder S (2007). Improved Authentication Mechanism using Keystroke Analysis, Proceedings of the International Conference on Information and Communication Technology 7(9): 258-261.
Surendra K, Singhi, Huan L (2006). Feature Subset Selection Bias for Classification Learning, Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh ISBN: 1-59593-383-2. pp. 849-856.
Sylvain H, Jean-Y ves R, Hubert C (2005). Fusion of methods for keystroke Dynamics Authentication, Fourth IEEE Workshop on Automatic Identification Advanced Technologies pp. 224 – 229.
Thangavel K, Karnan M, Sivakumar R, Kaja MA (2006). Ant Colony system for segmentation and Classification of Microcalcifications in Mammograms Proceedings of the International Conference on Advanced Computing and Communications pp. 298 – 303.
Yang J, Honavar V (1998). Feature subset selection using a Genetic algorithm. IEEE Intell. Syst. Appl. 13(2): 44 – 49.
Youmei L, Zongben X (2003). An Ant Colony Optimization Heuristic for solving Maximum Independent Set Problems, Proceedings of the IEEE Conference on Fifth International Computational Intelligence and Multimedia Applications pp. 206 – 211.
Young JR, Hammon RW (1989). Method and apparatus for verifying an individual’s identity, US Patent 6862610, Patent No. 4 805 222, U.S. Patent and Trade- mark Office.
Yu E, Cho S (2003). GA-SVM Wrapper Approach for Feature Subset Selection in Keystroke Dynamics Identity Verification, Pro-ceedings of the International Joint Conference on Neural Networks, ISBN: 0-7803-7898-9. 3: 2253-2257.
Yu E, Cho S (2004). Keystroke dynamics identity verification – its problems and practical solutions. Comput. Security 23: 428–440.