Treinamento de Redes Neurais com Algoritmos Imunológicos em Dinâmica da Digitação

Paulo Henrique Pisani† e Ana Carolina Lorena§
†Universidade Federal do ABC (UFABC), Santo André, SP
§Universidade Federal de São Paulo (UNIFESP), São José dos Campos, SP
email: paulo.pisani@ufabc.edu.br, aclorena@unifesp.br

***Resumo*—O ritmo de desenvolvimento da tecnologia é notável e trouxe diversos avanços como, por exemplo, a identidade digital. Entretanto, a identidade digital potencializou o roubo de identidades devido à maior exposição dos dados. Diante deste cenário, sistemas de detecção de intrusões que analisam o comportamento do usuário mostram-se como uma alternativa promissora para combater este problema. Estes sistemas criam um modelo de comportamento do usuário e, posteriormente, eventos observados que desviem deste modelo são classificados como intrusões em potencial. A dinâmica da digitação, discutida neste artigo, é uma das características que podem ser analisadas para definição do modelo do usuário. Neste trabalho, é feita uma comparação entre o tradicional algoritmo de treinamento de redes neurais *backpropagation* e duas abordagens evolutivas baseadas em algoritmos imunológicos, atuando no reconhecimento de usuários por dinâmica da digitação.**

***Palavras-chave*—Dinâmica da digitação, redes neurais, algoritmos imunológicos.**

I. Introdução

ATUALMENTE, com o surgimento da identidade digital, o roubo de identidades foi potencializado, principalmente devido à maior exposição dos dados. Com o intuito de mitigar este problema, surgiu o conceito de detectar intrusões baseando-se no comportamento do usuário, o que é conhecido como *user profiling* [1]. Desta forma, qualquer evento observado que desvie do perfil normal do usuário é considerado uma potencial intrusão, evitando assim que intrusos utilizem o sistema. A definição do perfil pode levar em consideração uma série de aspectos, contudo, a proposta deste trabalho é focar no estudo da *dinâmica da digitação* [2], que envolve a análise do ritmo de digitação do usuário e é classificada como uma tecnologia biométrica.

Entretanto, os dados referentes ao ritmo de digitação possuem ruídos e, consequentemente, a tarefa de reconhecimento de padrões neste cenário torna-se complexa. Dentre as ferramentas disponíveis em Inteligência Computacional para solucionar esse problema, os algoritmos imunológicos merecem ser destacados devido ao sucesso observado em diversas aplicações [3].

Este trabalho tem o objetivo de avaliar duas formas de aplicação do algoritmo imunológico CLONALG [4] com o *backpropagation* (BP) para treinamento de redes neurais no reconhecimento de padrões em *dinâmica da digitação*, de forma similar ao apresentado em [5]. Ao final, é realizado um comparativo de desempenho entre o *backpropagation* e as duas abordagens com algoritmos imunológicos. O restante do trabalho está organizado da seguinte forma: na Seção II, são apresentados trabalhos relacionados em *dinâmica da digitação*; na Seção III, são introduzidos conceitos dos algoritmos utilizados; na Seção IV, são mostrados detalhes do experimento conduzido neste trabalho, com a especificação de configurações e valores dos parâmetros adotados; na Seção V, são discutidos os resultados obtidos pelo experimento; e, na Seção VI, são apresentadas as conclusões.

Artigo recebido em 8 de fevereiro de 2012. Artigo aceito em 16 de agosto de 2012.

II. Dinâmica da Digitação

A partir do monitoramento das entradas fornecidas pelo teclado, é possível analisar o modo como os usuários digitam. Com isso, modelos que representam o ritmo de digitação normal do usuário são definidos. Posteriormente, estes modelos são utilizados para o seu reconhecimento [2].

Dentre as tecnologias biométricas, a *dinâmica da digitação* destaca-se devido a alguns fatores:

- O uso desta tecnologia não requer a aquisição de *hardware* adicional, como ocorre no caso de outras tecnologias (e.g. íris, impressão digital) [6];
- O nível de transparência é maior devido ao fato de não ser preciso executar ações especificamente para o reconhecimento, ou seja, o reconhecimento por dinâmica da digitação pode ser utilizado sem afetar o fluxo de atividades do usuário. Diferentemente de outras tecnologias, como a impressão digital, que requer a utilização de um dispositivo de leitura específico. Este fator contribui para o aumento da aceitação da *dinâmica da digitação* pelos usuários [7].

Em *dinâmica da digitação*, podem ser estudados dois processos distintos: a extração de características e a análise das características extraídas. No primeiro processo, diversas características podem ser extraídas. Contudo, normalmente são extraídas apenas duas [8], chamadas de *dwell time* e *flight time*. *Dwell time* é a diferença de tempo entre o instante que a tecla é pressionada e o instante que ela é solta. *Flight time* é a diferença de tempo entre o instante que uma tecla é solta e a próxima é pressionada. A Figura 1 mostra um exemplo de captura de dados de digitação e apresenta as duas características de forma gráfica. O segundo processo, que envolve a análise das características extraídas, pode ser realizado por

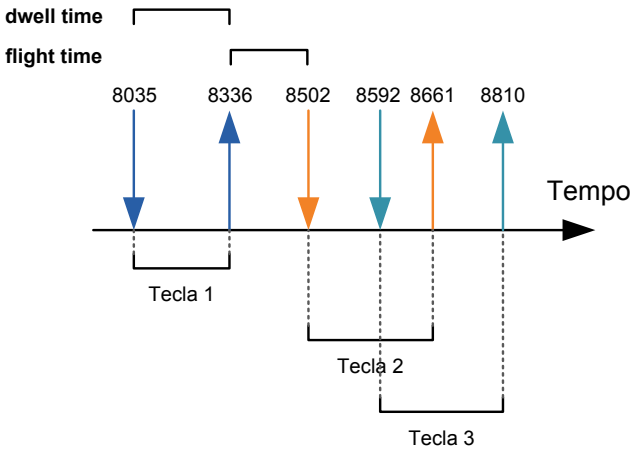

Figura 1. Exemplo de captura de dados de digitação (adaptado de [9]).

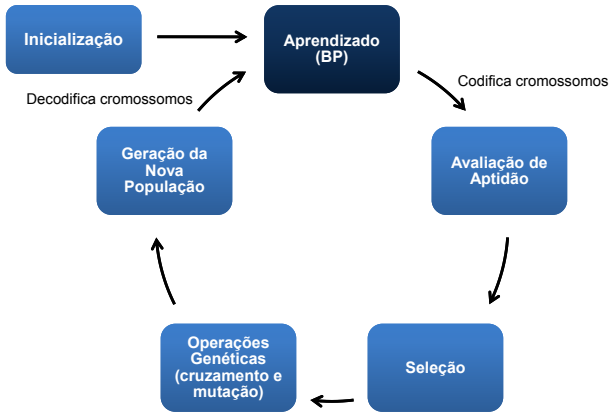

Figura 2. Treinamento baseado na Teoria de Lamarck (adaptado de [5]).

diversos algoritmos, como redes neurais artificiais e *support vector machines* (SVMs).

A *dinâmica da digitação* é considerada uma tecnologia de biometria comportamental e, devido a isso, o seu desempenho é normalmente avaliado de acordo com as seguintes taxas:

- **FAR e FRR**: A FAR (*False Acceptance Rate* - Taxa de Falsa Aceitação) mede o percentual de vezes que um intruso é aceito erroneamente como sendo legítimo e a FRR (*False Rejection Rate* - Taxa de Falsa Rejeição) mede o percentual de vezes que o usuário legítmo é rejeitado indevidamente [7]. Normalmente, quando o valor de uma das taxas aumenta, o valor da outra diminui e vice-versa.
- **EER**: A EER (*Equal Error Rate*) corresponde ao valor quando a FAR e a FRR são iguais [10]. O valor da ERR correspoderia, portanto, a um balanço entre as taxas, quando ambas assumem os mesmos valores.

Estudos com *dinâmica da digitação* começaram a ser realizados há algumas décadas. Um dos primeiros trabalhos publicados foi em 1980 [11]. Neste trabalho, sete pessoas participaram dos testes e foi obtida uma FAR de 0% e uma FRR de 4%. Um série de pesquisas foram realizadas após essa, conforme mostra a Tabela I, que apresenta classificadores utilizados em trabalhos anteriores na área de *dinâmica da digitação*.

III. Redes Neurais Evolutivas com Algoritmos Imunológicos

Esta seção apresenta conceitos dos algoritmos utilizados para o treinamento de redes neurais: redes neurais evolutivas, algoritmos imunológicos e treinamento com algoritmos imunológicos.

A. Redes Neurais Artificiais Evolutivas

Diversas ferramentas para reconhecimento de padrões foram utilizadas em trabalhos anteriores [10], como SVMs e redes neurais artificiais. Conforme observado em experimentos anteriores [2], redes neurais atingiram bom desempenho para reconhecimento de usuários por *dinâmica da digitação*.

Uma rede neural pode ser definida como um conjunto de unidades de processamento interligadas (neurônios) que atuam paralelamente para armazenar conhecimento e torná-lo disponível para o uso [24]. Dentre os diversos modelos de redes neurais artificiais propostos na literatura, este trabalho utiliza um *perceptron multicamada*. Entretanto, algoritmos de treinamento comumente utilizados, como o *backpropagation*, são sensíveis às condições iniciais que, neste caso, correspondem aos valores dos pesos definidos antes de iniciar o treinamento [4]. Normalmente, estes pesos são definidos de forma aleatória dentro de uma faixa de valores.

Uma das formas de solucionar este problema é com a utilização de algoritmos evolutivos (e.g. algoritmos genéticos) no treinamento de redes neurais, resultando em um treinamento híbrido [25]. Desta forma, são combinadas as características de busca em vários pontos do espaço (algoritmo evolutivo) com as da busca local (*backpropagation*), gerando as chamadas redes neurais artificiais evolutivas. Isto é possível, pois a determinação dos pesos de uma rede neural pode ser entendida como um problema de otimização que tem o objetivo de minimizar o erro da rede.

Neste trabalho, são discutidas duas formas de treinamento híbrido: uma baseada na teoria de *Darwin* [26] e uma baseada na teoria de *Lamarck* [27]. A primeira consiste em executar um algoritmo evolutivo para definição dos pesos iniciais da rede neural. Após isso, o algoritmo de busca local é executado. Enquanto isso, a segunda forma de treinamento adiciona uma etapa de aprendizado no ciclo do algoritmo evolutivo, conforme mostra a Figura 2.

B. Algoritmos Imunológicos

Dentre os diversos algoritmos de otimização que podem ser utilizados em redes neurais evolutivas, este trabalho utiliza uma abordagem com a aplicação de algoritmos imunológicos [5]. Os algoritmos imunológicos são algoritmos inspirados no sistema imunológico biológico e aplicados na solução de diversas classes de problemas, como, por exemplo: reconhecimento de padrões, aprendizado de máquina, vida artificial,

Tabela I
DESEMPENHO OBTIDO POR DIVERSOS CLASSIFICADORES EM DINÂMICA DA DIGITAÇÃO (ADAPTADO DE [12]).

Algoritmo de classificação	**Quantidade de usuários**	**EER**	**FAR** (falsa aceitação)	**FRR** (falsa rejeição)
Algoritmo de Gunetti e Picardi [13], aplicado em [14]	205	13%		
SVM [15]	100	6,95%		
Nearest neighbour [16]	51	9,96%		
Hidden Markov Model [17]	20	3,6%		
Algoritmo de Bleha (com equalização) [18], aplicado em [19]	47	6,2%		
Distância de Manhattan [20]	51	7,1%		
Random Forests [21]	53		1%	14%
Baseado em árvore com Distância Euclidiana [22]	12		0%	3,47%
Medida R [13]	205		0,005%	5 %
4-layer AAMLP [23]	21		0%	0,25 %

busca e otimização. Há uma série de algoritmos imunológicos diferentes, cada um inspirado em uma parte do sistema imunológico biológico, sendo que características como adaptação, aprendizado e memória são enfatizadas [4].

Estes algoritmos podem ser agrupados em cinco classes principais [4]: seleção clonal, seleção negativa, modelos da medula óssea, redes imunológicas discretas e contínuas. Este trabalho foca nos algoritmos de seleção clonal que, entre outras capacidades, atuam na solução de problemas do tipo busca e otimização, como é o caso do treinameno de redes neurais. O CLONALG [4] é um destes algoritmos de seleção clonal, que possui versões para problemas de reconhecimento de padrões e otimização. Na Figura 3, é apresentada a versão para problemas de busca e otimização, que é mais adequada para ser aplicada com o conceito de redes neurais evolutivas.

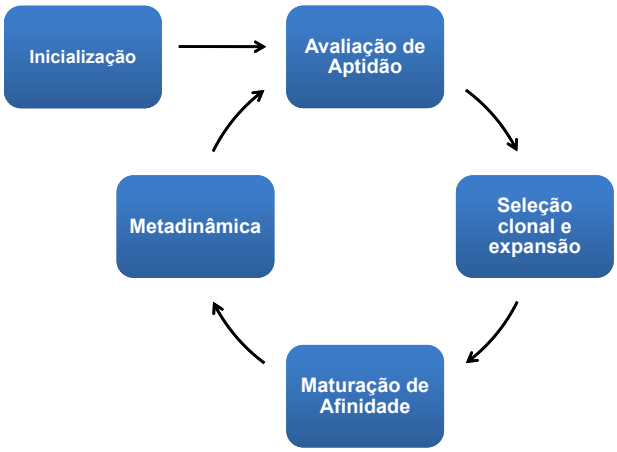

Figura 3. Ciclo - CLONALG.

Na inicialização do CLONALG, é gerado um conjunto de soluções, chamado de população de *anticorpos*. Após isso, cada membro da população é avaliado para determinar sua aptidão para solucionar o problema em questão. Esta população é então modificada pelo ciclo apresentado na Figura 3 até que um critério de parada seja atendido, como um limite de ciclos de execução (gerações). Na *seleção clonal e expansão*, n *anticorpos* com maior aptidão são clonados de maneira proporcional ao seu valor de aptidão. Após isso, na *maturação de afinidade*, os *anticorpos* clonados passam por um processo de mutação inversamente proporcional ao seu valor de aptidão. A *metadinâmica* consiste em substituir os m *anticorpos* com menor aptidão por novos gerados aleatoriamente.

C. Treinamento com Algoritmos Imunológicos

Dentre os diversos algoritmos imunológicos existentes, este trabalho utiliza o CLONALG (versão para busca e otimização). Com este algoritmo, são estudadas duas abordagens. A primeira, denominada de CLONALG+BP, consiste em utilizar o CLONALG para gerar um conjunto inicial de pesos para posterior aplicação do *backpropagation*. A segunda abordagem, denominada aqui de LMK-CLONALG, é uma modificação da evolução *Lamarckiana* para suportar o CLONALG (Figura 4). Na evolução *Lamarckiana*, os cromossomos são modificados por aprendizado, além de sofrerem alterações pelos operadores genéticos de *cruzamento* e *mutação*. No caso do CLONALG, são utilizados *anticorpos* ao invés de *cromossomos* e somente o operador de *mutação* é usado (etapa maturação de afinidade). A Figura 4 mostra a inclusão da etapa de aprendizado no ciclo do CLONALG para criar o LMK-CLONALG. Este algoritmo é similar ao apresentado em [28], mas possui algumas diferenças, como a *recombinação clonal*.

IV. EXPERIMENTO

Esta seção apresenta detalhes do experimento realizado neste trabalho em termos de base de dados, características extraídas e configuração dos algoritmos.

A. Base de dados e Características extraídas

A base de dados de digitação utilizada neste trabalho foi a mesma de [9]. Nesta base, 10 usuários participaram da coleta de amostras, sendo que todos utilizavam computadores diariamente. Cada usuário forneceu 10 amostras para as expressões testadas, em que as 5 primeiras amostras foram usadas no conjunto de treinamento dos algoritmos e as demais no conjunto de testes. Neste trabalho, foi considerada a expressão fixa "control userpasswords2" para a avaliação de desempenho dos algoritmos, que estava presente na base de dados utilizada.

As amostras possuem os instantes em que cada tecla foi pressionada e solta. Com estes dados, foram extraídas as características *dwell time* e *flight time*, descritas na Seção II.

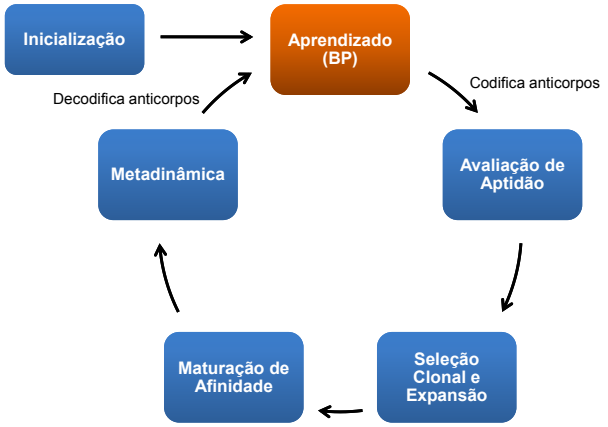

Figura 4. Ciclo - LMK-CLONALG (adaptado de [5]).

B. Configuração dos algoritmos

Os algoritmos comparados neste trabalho são utilizados para treinamento de redes neurais, em particular, o *perceptron multicamada* [4]. Portanto, o algoritmo de classificação é a rede neural em todos os casos. A rede utilizada possui 3 camadas de neurônios: 2 neurônios na primeira camada intermediária, 3 neurônios na segunda camada intermediária e 1 neurônio na camada de saída, conforme representado na Figura 5.

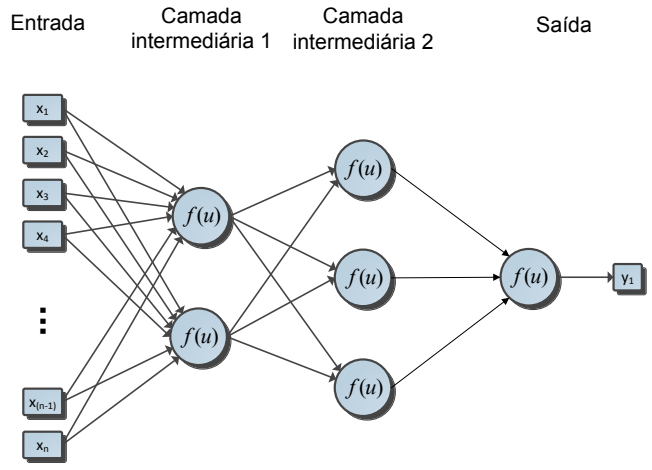

Figura 5. Estrutura da rede neural (adaptado de [9]). Cada neurônio é representado por um círculo, em que $f(u)$ é a função de ativação sigmóide.

Durante o treinamento, as amostras que pertencem ao usuário legítimo assumem o valor de saída 1 enquanto que as demais amostras assumem o valor 0. Na fase de testes, os pesos treinados são carregados na rede neural e as características extraídas são apresentas à rede. Se a saída for maior que um determinado valor de corte, o usuário é classificado como legítimo, caso contrário, como um intruso. Neste trabalho, o valor de corte utilizado foi 0,7.

Para este experimento, cada um dos três algoritmos assumiu os valores de parâmetros apresentados nas tabelas II, III e IV. Como os três algoritmos utilizam o *backpropagation*, os valores dos parâmetros deste foram mantidos iguais em praticamente todas as abordagens. A exceção foi no caso do LMK-CLONALG, em que o parâmetro *Épocas* foi diminuido para 50, pois esta abordagem executa o *backpropagation* diversas vezes e o valor 1000, utilizado nos demais algoritmos, exigiria um tempo de treinamento muito elevado. Outro aspecto a ser destacado é que tanto CLONALG+BP quanto LMK-CLONALG utilizam o algoritmo CLONALG. Nestas abordagens, os valores dos parâmetros referentes ao CLONALG foram mantidos os mesmos para facilitar a comparação. Nas três abordagens, é preciso definir um conjunto aleatório de pesos iniciais. Estes pesos foram definidos na faixa [-0,5;0,5], conforme recomendado em [29].

Tabela II
PARÂMETROS: *Backpropagation* SIMPLES.

Parâmetro	Valor
Épocas	1000
Taxa de aprendizado	0,75
Alfa momentum	0,15

Tabela III
PARÂMETROS: CLONALG+BP.

Parâmetro	Valor
Gerações	50
População	20
Codificação do anticorpo	Por valor: sequências dos valores dos pesos
Metadinâmica	5%
Avaliação de aptidão	Inverso do erro global da rede
Épocas	1000
Taxa de aprendizado	0,75
Alfa momentum	0,15

Tabela IV
PARÂMETROS: LMK-CLONALG.

Parâmetro	Valor
Gerações	50
População	20
Codificação do anticorpo	Por valor: sequências dos valores dos pesos
Metadinâmica	5%
Avaliação de aptidão	Inverso do erro global da rede
Épocas	50
Taxa de aprendizado	0,75
Alfa momentum	0,15

V. RESULTADOS

Para os testes dos algoritmos avaliados neste trabalho foi utilizada a aplicação implementada em [9] como base, que

utilizou a linguagem de desenvolvimento *Delphi*. Sobre esta aplicação, foi adicionado o suporte às abordagens com algoritmos imunológicos. Os testes foram conduzidos em um computador com *Intel Core 2* 2,40Ghz.

Como a *dinâmica da digitação* é considerada um tipo de biometria comportamental, a avaliação de desempenho utilizou medidas comumente adotadas em biometria: taxa de falsa aceitação (FAR) e taxa de falsa rejeição (FRR). A Figura 6 apresenta os valores obtidos para as duas taxas pelos três algoritmos. Devido à característica estocástica dos algoritmos aplicados, os resultados apresentados aqui são a média de cinco testes para cada caso. Esta abordagem foi adotada também em [5] para lidar com algoritmos que possuem esta característica.

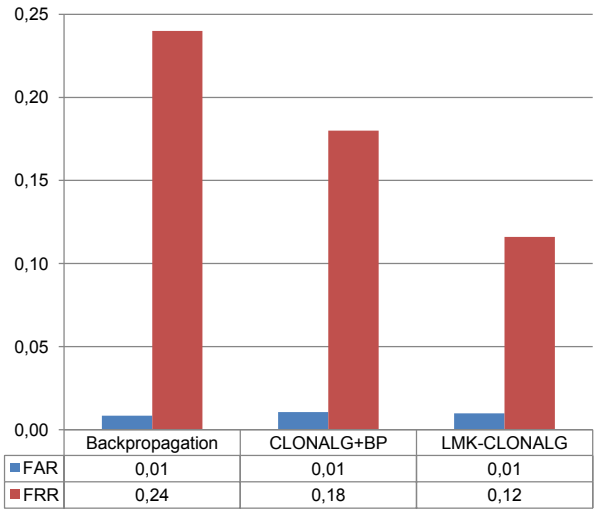

Figura 6. Desempenho obtido: FAR e FRR.

Além disso, na Figura 7 é mostrado o tempo gasto para executar o treinamento por cada algoritmo.

Conforme observado nos testes, as abordagens baseadas em algoritmos imunológicos obtiveram melhor desempenho em termos de FRR. Isso pode ser explicado pela capacidade de explorar diversas partes do espaço de busca, diminuindo a suscetibilidade às condições iniciais do *backpropagation*. Entretanto, o consumo de recursos computacionais da combinação de algoritmos foi maior, principalmente no caso do LMK-CLONALG, o que pode ser constatado pelo maior tempo para treinamento. Outro fator importante é que a FAR atingiu valores menores que a FRR em todos os testes. Isso pode ser explicado pela maior quantidade de exemplos negativos utilizados durante o treinamento da rede neural artificial, pois, para cada usuário, havia 5 exemplos positivos e 45 negativos (9 x 5).

VI. Conclusão

O surgimento da identidade digital trouxe uma série de avanços, mas também contribuiu para o aumento do roubo de identidades. Neste artigo foi apresentada uma abordagem para combater o roubo de identidades baseada em *dinâmica*

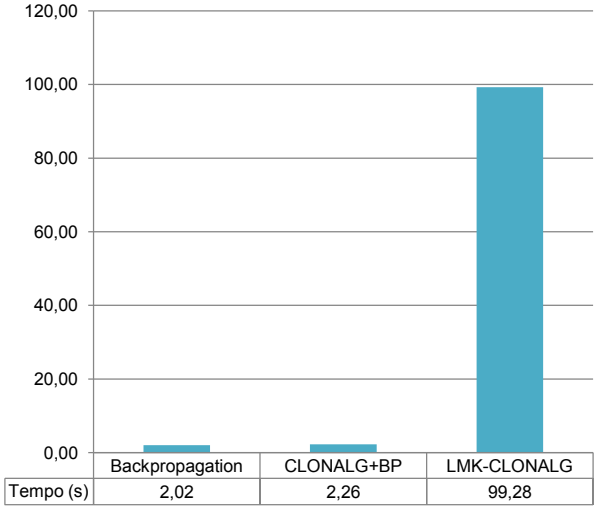

Figura 7. Desempenho obtido: Tempo de execução.

da digitação. Foram analisadas duas formas para combinar algoritmos imunológicos com o *backpropagation* e, após isso, foi efetuada uma comparação entre elas. Conforme observado nos testes, a taxa de acerto obtida pelas abordagens com algoritmos imunológicos foi superior à do *backpropagation* tradicional.

Entretando, é importante destacar que os resultados obtidos aqui são dependentes da base de dados e dos parâmetros adotados nos algoritmos. Uma investigação mais aprofundada em outras bases de dados e com valores de parâmetros diferentes pode ser efetuada em trabalhos futuros.

Agradecimentos

À UFABC, à CAPES, à FAPESP e ao CNPq, pelo apoio financeiro.

Referências

[1] L. Wang and X. Geng, *Behavioral Biometrics for Human Identification*, ser. Medical Information Science Reference. IGI Global, 2009.

[2] M. Karnan, M. Akila, and N. Krishnaraj, "Biometric personal authentication using keystroke dynamics: A review," *Applied Soft Computing*, vol. 11, pp. 1565–1573, 2011.

[3] J. Timmis, A. Hone, T. Stibor, and E. Clark, "Theoretical advances in artificial immune systems," *Theoretical Computer Science*, vol. 403, pp. 11–32, 2008.

[4] L. N. de Castro, *Fundamentals of Natural Computing*. Chapman & Hall/CRC, 2006.

[5] P. Pisani and A. Lorena, "Evolutionary neural networks applied to keystroke dynamics: Genetic and immune based," in *Evolutionary Computation (CEC), 2012 IEEE Congress on*, june 2012, pp. 2965–2972.

[6] D. Hosseinzadeh and S. Krishnan, "Gaussian mixture modeling of keystroke patterns for biometric applications," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 38, no. 6, pp. 816–826, 2008.

[7] A. Peacock, X. Ke, and M. Wilkerson, "Typing patterns: a key to user identification," *Security Privacy, IEEE*, vol. 2, no. 5, pp. 40–47, 2004.

[8] R. Moskovitch, C. Feher, A. Messerman, N. Kirschnick, T. Mustafic, A. Camtepe, B. Lohlein, U. Heister, S. Moller, L. Rokach, and Y. Elovici, "Identity theft, computers and behavioral biometrics," in *Intelligence and Security Informatics, 2009. ISI '09. IEEE International Conference on*. IEEE, 2009, pp. 155–160.

[9] P. H. Pisani and S. do Lago Pereira, "Lamarckian evolution of neural networks applied to keystroke dynamics," in *ICEC 2010 - Proceedings of the International Conference on Evolutionary Computation, [part of the International Joint Conference on Computational Intelligence IJCCI 2010], Valencia, Spain, October 24 - 26, 2010*, J. Filipe and J. Kacprzyk, Eds. SciTePress, 2010, pp. 358–364.

[10] H. Crawford, "Keystroke dynamics: Characteristics and opportunities," in *Privacy Security and Trust (PST), 2010 Eighth Annual International Conference on*, 2010, pp. 205–212.

[11] R. Gaines, W. Lisowski, S. Press, and N. Shapiro, "Authentication by keystroke timing: some preliminary results, technical report," Rand Corporation, Tech. Rep., 1980.

[12] P. H. Pisani and A. C. Lorena, "Detecção de intrusões com dinâmica da digitação: uma revisão sistemática," Universidade Federal do ABC, Santo André, Brasil, Technical Report 06/2011, dezembro 2011.

[13] D. Gunetti and C. Picardi, "Keystroke analysis of free text," *ACM Trans. Inf. Syst. Secur.*, vol. 8, pp. 312–347, 2005.

[14] J. Montalvao, C. Almeida, and E. Freire, "Equalization of keystroke timing histograms for improved identification performance," in *Telecommunications Symposium, 2006 International*. IEEE, 2006, pp. 560–565.

[15] R. Giot, M. El-Abed, and C. Rosenberger, "Keystroke dynamics with low constraints SVM based passphrase enrollment," in *Biometrics: Theory, Applications, and Systems, 2009. BTAS 2009. IEEE 3rd International Conference on*. IEEE, 2009, pp. 1–6.

[16] K. Killourhy and R. Maxion, "The effect of clock resolution on keystroke dynamics," in *Recent Advances in Intrusion Detection*, ser. Lecture Notes in Computer Science, R. Lippmann, E. Kirda, and A. Trachtenberg, Eds. Springer Berlin / Heidelberg, 2008, vol. 5230, pp. 331–350.

[17] R. Rodrigues, G. Yared, C. do N. Costa, J. Yabu-Uti, F. Violaro, and L. Ling, "Biometric access control through numerical keyboards based on keystroke dynamics," in *Advances in Biometrics*, ser. Lecture Notes in Computer Science, D. Zhang and A. Jain, Eds. Springer Berlin / Heidelberg, 2005, vol. 3832, pp. 640–646.

[18] S. Bleha, C. Slivinsky, and B. Hussien, "Computer-access security systems using keystroke dynamics," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 12, no. 12, pp. 1217–1222, 1990.

[19] J. R. M. Filho and E. O. Freire, "On the equalization of keystroke timing histograms," *Pattern Recognition Letters*, vol. 27, no. 13, pp. 1440–1446, 2006.

[20] K. Killourhy and R. Maxion, "Why did my detector do that?! predicting keystroke-dynamics error rates," in *Recent Advances in Intrusion Detection*, ser. Lecture Notes in Computer Science, S. Jha, R. Sommer, and C. Kreibich, Eds. Springer Berlin / Heidelberg, 2010, vol. 6307, pp. 256–276.

[21] N. Bartlow and B. Cukic, "Evaluating the reliability of credential hardening through keystroke dynamics," in *Software Reliability Engineering, 2006. ISSRE '06. 17th International Symposium on*. IEEE, 2006, pp. 117–126.

[22] W. Chang, "Reliable keystroke biometric system based on a small number of keystroke samples," in *Emerging Trends in Information and Communication Security*, ser. Lecture Notes in Computer Science, G. Muller, Ed. Springer Berlin / Heidelberg, 2006, vol. 3995, pp. 312–320.

[23] E. Yu and S. Cho, "Novelty detection approach for keystroke dynamics identity verification," in *Intelligent Data Engineering and Automated Learning*, ser. Lecture Notes in Computer Science, J. Liu, Y.-m. Cheung, and H. Yin, Eds. Springer Berlin / Heidelberg, 2003, vol. 2690, pp. 1016–1023.

[24] S. Haykin, *Redes Neurais: Princípios e Prática, 2a Edição*. Bookman, 1999.

[25] K. W. C. Ku, M. W. Mak, and W. C. Siu, *Approaches to combining local and evolutionary search for training neural networks: a review and some new results*. Springer-Verlag New York, Inc., 2003, pp. 615–641.

[26] X. Yao, "Evolving artificial neural networks," *Proceedings of the IEEE*, vol. 87, no. 9, pp. 1423–1447, sep 1999.

[27] P. A. C. Valdivieso, M. G. Arenas, F. J. G. Castellano, J. J. M. Guervos, A. Prieto, V. M. Rivas, and G. Romero, "Lamarckian evolution and the baldwin effect in evolutionary neural networks," *CoRR*, 2006.

[28] J. Yang, M. Gong, L. Jiao, and L. Zhang, "Improved clonal selection algorithm based on lamarckian local search technique," in *Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on*, 2008, pp. 535–541.

[29] J. F. Kolen, J. B. Pollack, J. F. Kolen, and J. B. Pollack, "Back propagation is sensitive to initial conditions," in *Complex Systems*. Morgan Kaufmann, 1990, pp. 860–867.