



Exacta

ISSN: 1678-5428

exacta@uninove.br

Universidade Nove de Julho

Brasil

Belan, Peterson Adriano; Petinari Nery, Edilaine; Alves de Araújo, Sidnei  
Software para auxílio à pré-alfabetização infantil baseado em reconhecimento inteligente de  
caracteres manuscritos  
Exacta, vol. 4, núm. 1, janeiro-junho, 2006, pp. 87-93  
Universidade Nove de Julho  
São Paulo, Brasil

Disponível em: <http://www.redalyc.org/articulo.oa?id=81040108>

- ▶ Como citar este artigo
- ▶ Número completo
- ▶ Mais artigos
- ▶ Home da revista no Redalyc

redalyc.org

Sistema de Informação Científica

Rede de Revistas Científicas da América Latina, Caribe , Espanha e Portugal  
Projeto acadêmico sem fins lucrativos desenvolvido no âmbito da iniciativa Acesso Aberto

# Software para auxílio à pré-alfabetização infantil baseado em reconhecimento inteligente de caracteres manuscritos

Peterson Adriano Belan

Graduado em Ciência da Computação – Uninove.  
peterson@gruponef.com.br, São Paulo – SP [Brasil]

Edilaine Petinari Nery

Graduada em Ciência da Computação – Uninove.  
edilaine@gruponef.com.br, São Paulo – SP [Brasil]

Sidnei Alves de Araújo

Mestre em Ciência da Computação – Mackenzie;  
Doutorando em Ciência da Computação – Poli-USP;  
Professor na graduação [Ciência da Computação] – Uninove.  
saraugo@uninove.br, São Paulo – SP [Brasil]

Neste trabalho, propusemos o desenvolvimento de um programa (*software*) para auxiliar na pré-alfabetização infantil, com base no reconhecimento de caracteres manuscritos, usando uma rede neural artificial (RNA) supervisionada. Previamente, crianças com idade entre 5 e 6 anos utilizaram o *software* composto de um conjunto de letras do alfabeto escritas por outras com mesma idade, por meio do módulo neural que compõe seu núcleo. Após a fase de treinamento, o *software*, composto também de uma interface multimídia devidamente caracterizada para interagir com a criança, ficou pronto para ser utilizado.

**Palavras-chave:** Caracteres manuscritos.  
Inteligência artificial. Redes neurais. *Software* educacional.



## 1 Introdução

Com as inúmeras mudanças pelas quais a educação vem passando, observa-se a necessidade de inovações no ambiente escolar, entre as quais a informatização. Nesse contexto, a tecnologia tem a contribuir para a ampliação e melhoria dos materiais pedagógicos.

Recentes pesquisas demonstram que computadores, por si mesmos, não resolvem os problemas da educação, mas, se bem utilizados, podem tornar-se um recurso importante no meio (LIGUORI, 1997). O uso da informática no ensino deve promover a criação de ambientes educacionais de tal forma que o computador seja utilizado como ferramenta que permita a interação entre o aluno, o professor e o conteúdo a ser ensinado. Vale ressaltar que essa relação com as novas tecnologias precisa ser contextualizado e ter objetivos bem definidos (GUEDES; GUEDES, 2004).

A inteligência artificial (IA) vem contribuindo para o desenvolvimento de técnicas empregadas na solução de problemas educacionais. A IA é um campo da computação que se está expandindo rapidamente, dada a sua aplicabilidade nas mais diversas áreas. Os sistemas ditos “inteligentes” devem apresentar três características envolvendo conhecimento: a capacidade de armazená-lo, a de aplicá-lo na solução de um problema, e a de adquiri-lo por meio da experiência. São várias as técnicas empregadas para criação desses sistemas, entre as quais estão as redes neurais artificiais (RNAs) usadas neste trabalho.

O objetivo é apresentar o desenvolvimento de um *software* que, voltado para o ensino infantil, auxilie a criança no período de pré-alfabetização. Esse *software* exibe na tela palavras incompletas, acompanhadas de ilustrações para facilitar o raciocínio, e, ao mesmo tempo, solicita que a criança preencha as lacunas com letras que devem ser escritas, com o auxílio de uma

caneta óptica. Ao final, um *feedback* é fornecido. Com isso, pretende-se incentivar a criança a usar o computador não apenas como ferramenta de entretenimento, mas também como complemento no seu aprendizado.

## 2 Softwares educacionais

Nos últimos anos, a tecnologia vem alterando a estrutura da educação. Entre as alterações estão os *softwares* educacionais, cujo crescimento tem contribuído para a melhor utilização dos recursos eletrônicos disponíveis. Nesse contexto, o computador surge como um dos recursos auxiliadores, e não como o meio principal do trabalho educativo (GUEDES; GUEDES, 2004).

Vale ressaltar que o *software* deve fazer mais que instruir diretamente, que ensinar as crianças a contar e a decorar os nomes das capitais dos estados. Ele permite estratégias pedagógicas que podem ser usadas para algum objetivo de ensino, qualquer que seja a sua natureza ou finalidade. Isso quer dizer que até mesmo processadores de textos, gerenciadores de banco de dados, planilhas, geradores gráficos e outros programas servem para viabilizar fins educativos. Em se tratando de *softwares* educacionais, no entanto, um fator muito importante deve ser considerado: a interface que, diferentemente dos *softwares* comerciais, tem de receber atenção particular durante seu desenvolvimento, principalmente do ponto de vista do aproveitamento, ou seja, requer-se uma interface amigável, de fácil acesso e entendimento. Para isso, são necessárias entrevistas, avaliações e análises detalhadas das necessidades dos usuários, que podem indicar os pontos para o sucesso da aplicação como um todo.

Outro problema encontrado nesses *softwares* é a falta de suporte pedagógico, visto que muitos analistas não têm prestado atenção para essa carac-

terística (OLIVEIRA; CRUZ; EZEQUIEL, 2004). Para o sucesso dos programas educacionais, faz-se necessário também submetê-los à aprovação de professores e outros profissionais da educação, que devem ser envolvidos nos projetos, do início ao fim, avaliando e auxiliando a modelagem do sistema. A ausência dessa interação é um obstáculo freqüente ao desenvolvimento de *softwares* dessa natureza (FERNANDES; RAABE; BENITTI, 2004).

### 3 RNAs aplicadas no reconhecimento de caracteres

As técnicas de RNAs consistem em imitar a estrutura e o funcionamento do cérebro humano (HAYKIN, 1999). Tem-se como grande objetivo a modelagem de um “cérebro digital”, para atribuir às máquinas o poder de raciocinar, por meio de conhecimentos adquiridos do ambiente externo (aprendizado) (ROCHA FERNANDES, 2003). Assim, com função similar à de nosso cérebro, uma RNA é formada pelo agrupamento de elementos (neurônios) interligados entre si, responsáveis pelo processamento das informações.

As RNAs têm sido bastante utilizadas nas mais diversas áreas, como matemática, estatística, física, computação e engenharia, sendo geralmente empregadas em reconhecimento de padrões, percepção, controle motor, jogos etc. Existem vários modelos de RNA, cada um mais adequado a um determinado tipo de problema. Neste trabalho, abordou-se um tipo supervisionado conhecido como *multi-layer perceptron* (MLP), treinado com um algoritmo de retropropagação de erro, denominado *backpropagation*.

#### 3.1 Multi-layer perceptrons

O MLP é caracterizado por possuir uma ou mais camadas ocultas, além das camadas de en-

trada e saída, que possibilitam à rede mapear, com eficiência, padrões de entrada com estruturas similares para saídas diferentes. Para isso, as camadas ocultas funcionam como detectores de características, permitindo que a rede crie sua própria representação do problema (RICH; KNIGHT, 1994), enquanto a camada de saída tem a função de receber os estímulos da última camada oculta e construir o padrão que será a resposta.

O treinamento do MLP é composto por duas fases. Na primeira, quando um padrão é apresentado à rede pela camada de entrada, o sinal de ativação é propagado até a camada de saída, produzindo uma resposta. Na segunda, a saída obtida é comparada com a desejada e um sinal de erro é produzido. Esse sinal é então retropropagado, desde a camada de saída até a de entrada e os pesos sinápticos vão sendo ajustados para que a resposta da rede se aproxime da desejada.

#### 3.2 Algoritmo *backpropagation*

O algoritmo *backpropagation* é um dos mais utilizados para treinar as redes MLP atualmente. Como qualquer algoritmo de RNA, seu objetivo é o ajuste dos pesos sinápticos da rede. O treinamento de uma RNA baseado nele utiliza, geralmente, um grande esforço computacional, com várias iterações, para que se obtenha a resposta desejada. Esse treinamento pode ser acelerado usando a técnica de aumentar linearmente a taxa de aprendizado, que normalmente é um número muito próximo de zero. Se essa alteração não for calculada com exatidão, poderá causar instabilidades na rede (RUSSELL; NORVIG, 2004).

#### 3.3 Reconhecimento de caracteres

A utilização do reconhecimento de caracteres tem sido estudada ao longo do tempo e pode ter muitos objetivos. Inicialmente, era possível reconhecer apenas caracteres impressos e de fontes predefinidas (SILVA, 2002; GONZALEZ;



WINTZ, 1987). Com as novas tecnologias, tornou-se praticável o reconhecimento de caracteres manuscritos com bons resultados. Essa tarefa, conhecida como reconhecimento inteligente de caracteres (em inglês, *intelligent character recognition* [ICR]), é um dos objetos de estudo deste trabalho. Em alguns tipos de reconhecimento óptico de caracteres (em inglês, *optical character recognition* [OCR]), trabalha-se com formatos predefinidos e os caracteres são comparados somente pela sua similaridade. Já no ICR, há uma variação de amostras muito grande, pois dificilmente existem dois indivíduos com as letras idênticas. São várias as técnicas empregadas na tarefa de reconhecimento de caracteres manuscritos. Neste trabalho, vale salientar, utilizamos uma RNA do tipo MLP treinada com o algoritmo *backpropagation*. Assim, a RNA compõe o “núcleo” do *software* proposto.

### 3.4 Metodologia de desenvolvimento do software

Na implementação do *software* em foco, empregamos a linguagem C++ para o desenvolvimento do algoritmo *backpropagation* e a *visual basic* (VB) para a execução da interface gráfica responsável pela comunicação com o usuário. O sistema está dividido em três módulos básicos: um responsável pela interface, outro pelo treinamento e o terceiro pelo reconhecimento de caracteres. As Ilustrações 1 e 2 mostram, respectivamente, um esquema em blocos dos módulos do *software* e sua interface em pleno funcionamento.

Como o *software* requer treinamento para ser utilizado, coletamos 780 amostras de caracteres de 30 crianças. O conjunto de treinamento foi composto por 650 letras, e o de testes, pelas 130 restantes. Tanto no treinamento quanto na sessão de testes, utilizamos a mesma quantidade de cada letra do alfabeto.



Ilustração 1: Estrutura do sistema proposto

Fonte: Os autores.



Ilustração 2: Interface do software

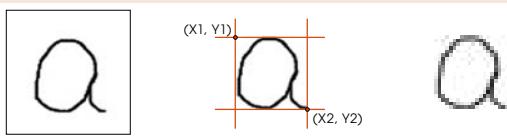
Fonte: Os autores.

A interface do *software* (Ilustração 2), possui recursos visuais de fácil compreensão, com imagens e cores que representam o objetivo a ser alcançado, e conta com o recurso multimídia de emitir um som correspondente à letra que está sendo solicitada para a criança escrever. Para tornar o *software* bastante simples, apenas alguns objetos estão localizados na interface, como área para a entrada da letra e três botões de ação: 1) para executar a rotina de reconhecimento; 2) para limpar a área de escrita; 3) para sair da aplicação. Há ainda uma área em que é apresentada a figura correspondente à questão que deve ser resolvida e, por fim, uma imagem e um som sinalizando se a criança acertou ou não a tarefa proposta.

### 3.5 Pré-processamento

Para compor o conjunto de treinamento da RNA, as imagens dos caracteres passam por um pré-processamento, dividido em duas etapas: captação e escalonamento.

A área reservada para a entrada dos caracteres manuscritos tem o tamanho predefinido de 200 x 200 *pixels*. Após a aquisição do caractere, é feito o escalonamento e a imagem (binária) resultante passa ter 20 x 20 *pixels*, o que compõe a entrada da RNA. Esse procedimento é demonstrado na Ilustração 3. Vale ressaltar que esse processo deve ser repetido, durante o uso do *software*, cada vez que a criança escreve uma letra e solicita que o programa avalie.



3.1: Caractere em seu formato original (200 x 200 pixels)

3.2: Caractere sem as partes inutilizadas da área de captura (122 x 122 pixels)

3.3: Caractere no tamanho final (20 x 20 pixels)

### Ilustração 3: Exemplo do processo de escalonamento

Fonte: Os autores.

### 3.6 Características da RNA proposta

A RNA proposta (Ilustração 4) possui duas camadas ocultas. Sua estrutura é formada por 400 neurônios de entrada (de acordo com a quantidade de *pixels* da imagem pré-processada do caractere), 26 na primeira camada oculta, 78 na segunda e 26 de saída (cada um para representar uma letra do alfabeto).

## 4 Resultados experimentais

A fim de averiguar o desempenho do *software*, realizamos o treinamento da RNA (núcleo da aplicação) com o conjunto de 650 imagens (25 de cada letra) e, depois, o testamos com as 130 imagens restantes (5 de cada letra). Os resultados estão sintetizados na Tabela 1. Convém lembrar que os caracteres usados na fase de testes não faziam parte

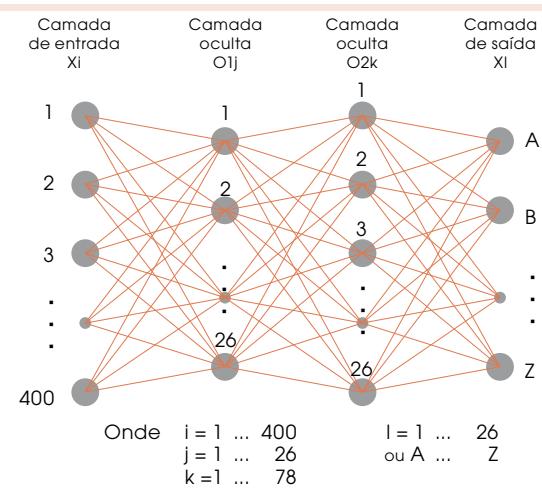


Ilustração 4: Representação da RNA proposta

Fonte: Os autores.

Tabela 1: Resultados gerais obtidos com os testes realizados

Letra	Acertos	Erros	Margem de erro
A	5	0	0%
B	4	1	20%
C	5	0	0%
D	5	0	0%
E	5	0	0%
F	5	0	0%
G	5	0	0%
H	4	1	20%
I	5	0	0%
J	5	0	0%
K	5	0	0%
L	5	0	0%
M	5	0	0%
N	5	0	0%
O	5	0	0%
P	4	1	20%
Q	5	0	0%
R	4	1	20%
S	5	0	0%
T	5	0	0%
U	4	1	20%
V	4	1	20%
X	5	0	0%
W	4	1	20%
Y	4	1	20%
Z	5	0	0%
Totais	122	8	6,15%

Fonte: Os autores.



do conjunto de treinamento e foram escritos por crianças de 5 e 6 anos, exclusivamente para essa finalidade.

Analisando os dados da Tabela 1, percebe-se que houve dificuldades no reconhecimento de certas letras como “b”, “h”, “p”, “r”, “u”, “v”, “w” e “y”. Na Ilustração 5, apresentamos alguns caracteres utilizados na fase de testes da rede, os quais nos fornecem esses indícios.

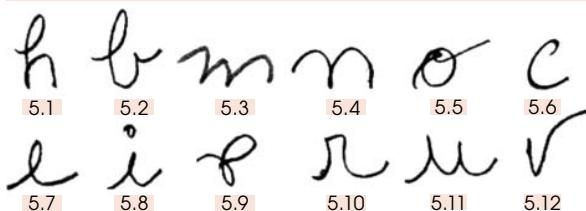


Ilustração 5: Caracteres utilizados na fase de testes

Fonte: Os autores.

Observamos que uma das principais dificuldades da RNA foi extrair as características intrínsecas das letras, uma vez que alguns caracteres distintos possuem partes muito semelhantes, o que confunde a rede. Podemos citar como exemplo as letras “h” e “b”, que têm traços similares, como pode ser visto nos itens 5.1 e 5.2 da Ilustração 5. Esse mesmo problema é percebido por meio dos itens 5.7 e 5.8, nos quais o que diferencia as letras “e” e “i” é, praticamente, a maior abertura na parte superior do “e”. Existem letras, no entanto, tais como “o” e “c”, que têm características bastante particulares, como se pode observar nos itens 5.5 e 5.6.

Os resultados nos mostraram que a parametrização da rede é de extrema importância, visto que não há referência que forneça os parâmetros necessários para a resolução de um determinado problema. Assim, todos os padrões que definem a estrutura de uma RNA devem ser obtidos por meio de muitos testes.

A Tabela 1 permite observar os bons resultados obtidos, devido ao considerável êxito na

parametrização da rede e também à forma como foi realizado o pré-processamento das imagens de entrada.

Merecem destaque também as tentativas para tornar a interface o mais simples possível, visando ao nosso público-alvo. Os bons resultados obtidos devem-se também a pesquisas realizadas em trabalhos relacionados a *softwares* educacionais.

Não obstante, as ferramentas utilizadas no desenvolvimento do *software* mostraram-se adequadas a cada propósito, em função das características específicas dessas linguagens.

## 5 Considerações finais

Neste trabalho, desenvolvemos um *software* de auxílio à pré-alfabetização infantil baseado no reconhecimento de caracteres manuscritos por uma RNA supervisionada. Após os testes realizados, acreditamos nos resultados satisfatórios de sua aplicação, pois demonstramos que, embora se utilizem recursos de IA, a simplicidade e a possibilidade de uso da interface podem tornar a utilização do computador ainda mais atrativa.

Tendo em vista o porcentual de erro obtido (6,15% em média), julgamos que o *software* apresentou um bom desempenho. Podemos afirmar também que a configuração do módulo neural foi um fator de extrema importância para o seu desempenho, devido a algumas dificuldades descritas na tarefa de reconhecimento dos caracteres. Fica o desafio de, a partir desse modelo, criar um *software* mais completo, que reconheça palavras ou até mesmo frases, para que a qualidade do auxílio ao aprendizado da criança seja ainda melhor. Além disso, uma interface inteligente, adaptável ao aluno e que proporcione sua plena utilização, será da mesma forma uma proposta importante e muito bem-vinda.

## Software for helping childish pre-alphabetization based on intelligent recognition of manuscript characters

In this work, we propose the development of a software to help the childish pre-alphabetization, based on the recognition of manuscript characters, using the supervised artificial neural network. Previously, children of 5 and 6 years old ran the software composed of a set of letters of the alphabet written by other children in the same age, by means of the neural module that compounds its nucleus. After the training phase, the software, also composed of a multimedia interface duly characterized to interact with the child, was considered ready to use.

**Key words:** Artificial intelligence. Educational software. Manuscript characters. Neural networks.

## Referências

FERNANDES, L. S.; RAABE A. L. A.; BENITTI, F. B. V. Interface de software educacional: desafios de design gráfico. In: CONGRESSO BRASILEIRO DE COMPUTAÇÃO, 4., 2004, Itajaí. *Anais...* Itajaí: CBCOMP, p. 254-258, 2004.

GONZALEZ, R. C.; WINTZ, P. *Digital image processing*. 2. ed. Boston: Addison-Wesley, 1987.

GUEDES, J. R.; GUEDES, C. L. Produção de software educativo através de um projeto interdisciplinar. In: CONGRESSO BRASILEIRO DE COMPUTAÇÃO, 4., 2004, Itajaí. *Anais...* Itajaí: CBCOMP, p. 223-228, 2004.

HAYKIN, S. *Neural networks: a comprehensive foundation*. 2. ed. Nova Jersey: Prentice Hall, 1999.

LIGUORI, L. M. As novas tecnologias da informação e da comunicação no campo dos velhos problemas e desafios educacionais. In: LITWIN, E. (Org.).

*Tecnologia educacional: política, histórias e propostas*. 1. ed. Porto Alegre: Artes Médicas, 1997. p. 78-97.

OLIVEIRA, A. A. F. O.; CRUZ, D. T.; EZEQUIEL, J. P. Interface homem-computador para desenvolvimento de software educativo. In: CONGRESSO BRASILEIRO DE COMPUTAÇÃO, 4., 2004, Itajaí. *Anais...* Itajaí: CBCOMP, p. 219-222, 2004.

RICH, E.; KNIGHT, K. *Inteligência artificial*. 1. ed. São Paulo: Makron Books, 1994.

ROCHA FERNANDES, A. M. *Inteligência artificial: noções gerais*. 1. ed. Florianópolis: Visual Books, 2003.

RUSSELL, S.; NORVIG, P. *Inteligência artificial*. 2. ed. Rio de Janeiro: Campus, 2004.

SILVA, E. *Reconhecimento inteligente de caracteres manuscritos*. 2002 (Mestrado em Sistemas e Computação)–Instituto Militar de Engenharia, Rio de Janeiro, 2002.

recebido em: 18 jan. 2006 / aprovado em: 18 abr. 2006

### Para referenciar este texto

BELAN, P. A.; NERY, E. P.; ARAÚJO, S. A. de. Software para auxílio à pré-alfabetização infantil baseado em reconhecimento inteligente de caracteres manuscritos. *Exacta*, São Paulo, v. 4, n. 1, p. 87-93, jan./jun. 2006.