Uso de redes neurais artificiais para reconhecer personagens com imagens de quadros de animação

Adson M. da S. Esteves1, Alisson S. Henrique1, Augusto C. Pluschkat1

1Ciência da Computação – Universidade do Vale do Itajaí (UNIVALI)  
CEP 88302-202 – Itajaí – SC – Brazil

{shinadson,ali.steffens}@gmail.com, acpluschkat@hotmail.com

**Resumo.** Este artigo tem por objetivo o desenvolvimento de um sistema para avaliar a necessidade de ajuda de um usuário da IDE Portugol Studio enquanto o mesmo programa, utilizando a lógica difusa e formulando regras para defuzzificar variáveis que contam os erros, avisos, número de linhas, de variáveis e funções que o usuário pode fazer no código. Após realizar testes com diferentes códigos com diferentes problemas e complexidades, o sistema foi validado. Ainda há melhoras a serem feitas como uma pesquisa melhor sobre as variáveis utilizadas e seus limites e detalhes de tempo de análise, porém o uso da lógica difusa foi no Portugol Studio foi aceita dentro dos parâmetros estabelecidos.

# 1. Introdução

O.

# 2. Fundamentação Teórica

< - - Neste capítulo serão discutidos os principais temas de interesse deste artigo, sendo eles: o cenário atual da informática na educação, linguagens de programação para iniciantes e logica difusa. - - >

## 2.1 Redes Neurais Artificiais

Usando neurônios criados artificialmente, a ideia de redes neurais é simular o processo de pensamento do um ser humano, ao fazer com que cada neurônio artificial se comunique uns com os outros enviando sinais, como se fossem sinapses de um cérebro humano normal. Esse sistema foi feito para a identificação de padrões, quando não se sabe exatamente o problema ocorrente, resolvendo-o a partir de uma generalização dos dados com que é treinado.

Geralmente as redes neurais feitas, são multicamadas, onde cada camada de neurônios recebe entradas de uma camada anterior a ela, sendo assim, o sinal original entra pela primeira camada, passando por cada neurônio dela e liberando sinais para camadas inferiores, até se ter uma saída na última camada.

Foi primeiramente proposta e realizada em 1943 quando Warren McCulloch do o Instituto Tecnológico de Massachusetts e Walter Pitts da Universidade de Illinois criaram um modelo de redes neurais com algoritmos e matemática a partir de eletrônicos, chamado de Lógica de limite (Threshold logic), simulando um comportamento de neurônios. A ideia trazia que a saída de um neurônio era uma função da soma de diversas entradas [MACCULLOCH e PITTS, 1943].

Em 1958 Frank Rosenblatt crio o Perceptron, um algoritmo de redes neurais que utilizava apenas uma camada de neurônios[ROSENBLATT, 1985], porém após pesquisas de Marvin Minsky e Seymour Paper em 1969, as pesquisas no campo de redes neurais ficaram estagnadas, pois de acordo com o artigo deles haviam 2 problemas com o processamento de redes neurais, perceptrons básicos não conseguiam processar XOR e computadores da época não tinham poder de processamento suficiente para executar algoritmos de redes neurais.

Assim, Redes Neurais voltou em 1975, com o algoritmo de Backpropagation após Paul Werbos apresentar a proposta para redes neurais artificiais, aonde a saída final da rede é comparada com o resultado desejado e retreinada se necessária até que fique com um erro ou diferença ao valor desejado mínimo.



# 3. Metodologia

Foi definido o uso de RNA no reconhecimento de imagens, para isso, foi selecionado o jogo desenvolvido pela Nintendo, Pokémon Diamante. Dele seriam extraídos os 2 quadros de animação que cada personagem pokémon possui dentro do jogo e definido que a rede neural treinaria com o primeiro quadro de 145 pokémons, e a partir da entrada do segundo quadro de animação de um pokémon na rede, a saída deveria ser o pokémon pertencente ao quadro usado de entrada.

# 4. Desenvolvimento

Para a implementação da rede neural artificial, foi utilizada a biblioteca em Javascript Neataptic, o qual, permite a criação de redes neurais como também a visualização delas para um melhor entendimento, além da possibilidade de usar algoritmos genéticos.

## 4.1 Arquitetura da Rede Neural

O .

## 4.2 Regras de Treinamento

P.

# 5. Testes e Resultados

Fora.

# 6. Conclusões

A.

# Referencias

BEAUBOUEF, Theresa; MASON, John (2005). Why the high attrition rate for computer science students: some thoughts and observations. In: ACM SIGCSE Bulletin, USA, v. 37, n. 2, p. 103-106, jun. 2005.

CASTRO, T. H. C. (2002) et al. “Arquitetura SAAP - Sistema de Apoio à Aprendizagem de Programação”. XXII Congresso da Sociedade Brasileira de Computação, volume 5 - VIII Workshop de Informática na Escola, Florianópolis.

KINNUNEN, P. and MALMI, L. (2006) “Why students drop out CS1 course?” In Proceedings of the Second international Workshop on Computing Education Research (Canterbury, United Kingdom).ICER '06.ACM, New York, NY, 97-108.

LAHTINEN, E.; ALA-MUTKA, K.; JÄRVINEN, H.-M (2005). A study of the difficulties of novice programmers. Annual Sigcse Conference On Innovation And Technology In Computer Science Education, v. 37, n. 3, p. 14–18, Caparica, Portugal.

LOFTI A. Zedah. (1964). Fuzzy Sets. Information and Control. http://www.sciencedirect.com/science/article/pii/S001999586590241X , November.

LOFTI A. Zedah. (1964). A computational approach to fuzzy quantifiers in natural languages. Computers & Mathematics with Applications. http://www.sciencedirect.com/science/article/pii/0898122183900135, November.

MANNES, Paula (2013). Integração do Portugol Core com o Bipide. http://siaibib01.univali.br/pdf/Paula%20Mannes.pdf, November.

NORVIG, Peter; RUSSELL, Stuart Jonathan (2004). Inteligência artificial. editora Campus.

NOSCHANG, Luiz Fernando (2012). Adaptação Do Portugol Core Para Permitir A Integração Com Outras Ferramentas. Universidade do Vale do Itajaí.

NOSCHANG, Luiz Fernando; PELZ, Fillipi; JESUS, Elieser A. de Jesus; RAABE, André L. A.(2014). Uma IDE para Iniciantes em Programação. In: Workshop de Informática na Escola, No prelo.

OLIVEIRA, B. C. d. S.; WANG, M., GIBBONS, J. (2008). The visitor pattern as a reusable, generic, type-safe componente OOPSLA ’08 Proceedings of the 23rd ACM SIGPLAN conference on Object-oriented programming system languages and applications.

PELZ, Fillipi.(2011). Correção Automática de Algorítmos no Ensino Introdutório de Programação. http://lite.acad.univali.br/portugol/resources/pdfs/tra02.pdf, June.

PELZ, Fillipi (2014). Um Gerador de Dicas para Guiar Novatos na Aprendizagem de Programação. http://lite.acad.univali.br/portugol/resources/pdfs/dis01.pdf, February.

RAABE, André L. A.; SILVA, Júlia M. Carvalho da. Um ambiente para atendimento as dificuldades de aprendizagem de algoritmos. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 25., São Leopoldo, Rio Grande do Sul. Anais... RS:SBC, 2005.

SCHEFFLER, P. (2008). Teaching Algorithmics – Theory and Pratice. In: Proc. 2nd Intern. Sc. Conf. “Informatics in the Scientific Knowledge.” Varna, pp. 259-269.

SHEARD (2009), Judy et al. Analysis of research into the teaching and learning of programming. In: INTERNATIONAL COMPUTING EDUCATION RESEARCH WORKSHOP, 5., 2009, Berkeley, USA. Proceedings… USA: ACM, 2009. p. 93-104.