Classificação de páginas da Internet utilizando redes neurais artificiais.

Genilto Dallo

Departamento de Ciência da Computação (DECOMP)
Universidade Estadual do Centro-Oeste (UNICENTRO) Guarapuava, PR – Brasil

geniltodallo@gmail.com

Resumo. Este artigo aborda a classificação de páginas da Internet pelo seu conteúdo utilizando uma Rede Neural Kohonen. Desenvolveu-se um sistema em Java para extrair o conteúdo das páginas, e a partir do seu conteúdo é feita um análise e classificação com a RNA.
Palavras-chave: Redes Neurais Artificiais, RNA, classificação de páginas.

Abstract. This paper reports the classification of web pages by their content using a Kohonen Neural Network. A system was developed in Java to extract the contents of the pages, and from its content is made an analysis and classification with the RNA.
Keywords: Artificial Neural Networks, Kohonen, RNA, web pages classification.

1.Introdução

Este trabalho é motivado pelo crescimento constante do conteúdo na Internet, que tem como consequência a existência de uma grande quantidade de informações, dificulta a tarefa de recuperação e classificação desses dados.

Desde sua criação, a World Wide Web (WWW) apresenta taxas de crescimento espantosas. Isso se deve ao fato da sua alta acessibilidade e escalabilidade, que propiciam um ambiente muito favorável para o compartilhamento de informações entre usuários. Segundo um levantamento realizado em setembro de 2007, por um provedor de serviços norte-americano chamado Netcraft1,

estimasse que existam mais de 135,1 milhões de sítios na Web. Há também uma projeção que indica que a taxa de crescimento atual da Internet levará à existência de aproximadamente duzentos milhões de sítios no ano de 2010.[1]

A utilização de RNAs para classificação de conteúdo pode ser utilizada em várias áreas, como ferramentas de busca, sistemas de gerenciamento de acesso a Internet(Proxy e Firewall), Anti-vírus, bloqueio de SPAM em emails, recuperação de dados, data-mining, detecção de intrusos, etc.

Este artigo irá abordar a implementação de um software utilizando a RNA Kohonen para a classificação de páginas da Internet pelo seu conteúdo e uma breve explanação sobre redes neurais e redes neurais Kohonen.

2.Redes Neurais Artificiais (RNA)

As RNAs constituem uma das várias linhas de pesquisa no campo da Inteligência Artificial e têm por objetivo investigar a possibilidade da simulação de comportamentos inteligentes através de modelos baseados na estrutura e funcionamento do cérebro humano. Estes modelos são construídos a partir de técnicas computacionais e podem ser implementadas em hardware ou software. O estudo das RNAs é um dos ramos da Inteligência Artificial (IA) que mais se desenvolve, atraindo pesquisadores de diversas áreas do conhecimento [2].

A RNA possui a característica de ser uma memória do tipo associativa, ou seja, é capaz de recuperar o conhecimento armazenado a partir de partes da informação. Isso significa que dado um padrão aprendido, ao se apresentar entradas incompletas em relação ao padrão, a característica associativa permite a inferência do restante da informação [1].

O aprendizado de uma RNA pode ser:

Aprendizado Supervisionado: Neste tipo, a rede neural recebe um conjunto de entradas padronizadas e seus correspondentes padrões de saída, onde ocorrem ajustes nos pesos sinápticos até que o erro entre os padrões de saída gerados pela rede tenha um valor desejado.

Aprendizado não Supervisionado: neste tipo, a rede neural trabalha os dados de forma a determinar algumas propriedades dos conjuntos de dados. A partir destas propriedades é que o aprendizado é constituído.

Híbrido: neste tipo, ocorre a utilização dos dois tipos supervisionado e não-supervisionado, oferecendo a rede neural uma maior abrangência [4].

2.1 Equivalência de Computabilidade

A Tese de Church-Turing diz que todo problema computável pode ser resolvido por máquina de Turing. Se as redes neurais são ou não equivalentes a uma máquina de Turing (TM) e em

consequência são capazes de resolver qualquer problema computável e apenas eles, tem despertado grande interesse recentemente. Visto a luz dos trabalhos publicados por Arbib [3], pode-se dizer que em termos de computabilidade TM's e neurocomputadores são equivalentes. Isso quer dizer que um neurocomputador não sabe resolver nenhum problema que não pudesse ser resolvido com uma TM e vice versa. Esta afirmação pode ser descrita mais precisamente por dois teoremas [4]. Todo problema que pode ser resolvido por um TM poderá ser resolvido, por uma RNA munida de convenientes dispositivos de entrada e saída. Com efeito, usando neurônios artificiais (e dos mais simples, aqueles que possuem apenas saídas binárias) á possível construir os circuitos lógicos ‘e’, ‘ou’ e ‘não’ além de circuitos biestáveis. Pode-se tirar várias conclusões, dentre as quais os teoremas e o corolário que, existem redes neurais que não podem ser implementadas em TM. Consequentemente existem problemas que podem ser resolvidos por neurocomputadores que não podem ser resolvidos pela Máquina de Turing [4].

2.2 Redes Neurais KOHONEN

O algoritmo de Kohonen foi desenvolvido por Teuvo Kohonen em 1982, sendo considerado relativamente simples e com a capacidade de organizar dimensionalmente dados complexos em agrupamentos, de acordo com suas relações. Este método solicita apenas os parâmetros de entrada, mostrando-se ideal para problemas onde os padrões são desconhecidos ou indeterminados [5].

Este algoritmo é considerado um mapa auto-organizável (SOM), capaz de diminuir a dimensão de um grupo de dados, conseguindo manter a representação real com relação as propriedades relevantes dos vetores de entrada, tendo-se como resultado um conjunto das características do espaço de entrada [6].

Além disso, possui a propriedade de transformar um mapa multidimensional em bidimensional, adicionando os elementos ao novo mapa de tal forma que os objetos similares sejam posicionados próximos uns dos outros [6].

Apresenta duas importantes características: utiliza aproximação dos pontos similares onde os mesmos são processados separadamente e permite ao mapa obter centros em um plano bidimensional disponibilizando uma visualização facilmente compreensível [7].

Este algoritmo utiliza o método de aprendizagem por competição (*competitive learning*), considerado o mais comum nas RNA auto-organizáveis, permitindo que aconteça o aprendizado dividindo-se os padrões de entrada dos dados em conjuntos inseparáveis. Este método avalia os neurônios de saída da rede de maneira que ocorra uma competição entre eles, tendo-se como resultado o neurônio que possui maior ativação. A rede neural de Kohonen é composta por duas camadas: a de entrada e de Kohonen. Cada nó da camada de entrada tem a função de distribuir os valores padrões para a de Kohonen, que é um conjunto de nodos organizados de forma tabular. O

vetor de entrada possui seus elementos conectados com cada nó da camada Kohonen por meio de ligações, as quais são responsáveis por manterem atualizados os valores durante o processo de treinamento da RNA [5]. A figura 1 mostra de maneira simplificada a estrutura de uma rede Kohonen.

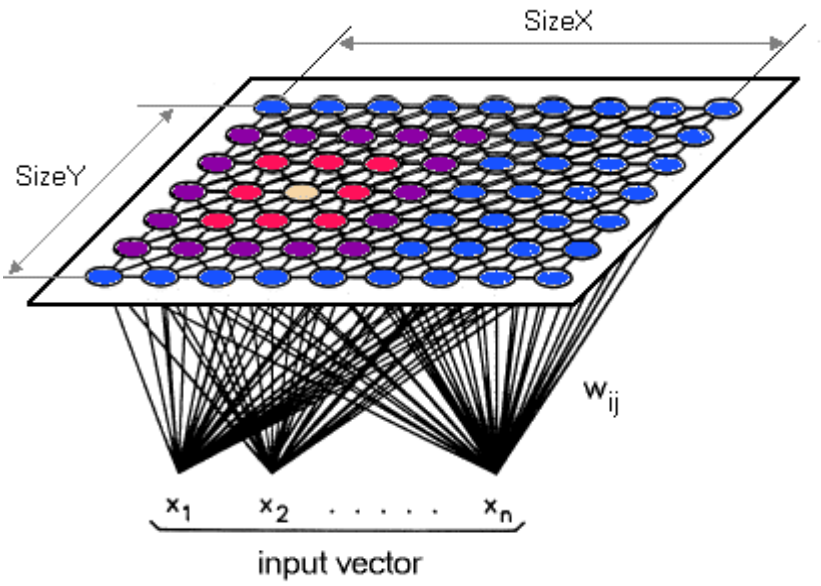

Figura 1. Representação da uma rede Kohonen [8].

As informações (e as abstrações) "aprendidas" por uma rede de Kohonen podem ser exploradas após o treinamento da rede e utilizadas das mais variadas formas. Algumas áreas em que a RNA Kohonen pode ser aplicada são: Classificação bibliográfica, sistema de busca em imagens, diagnósticos médicos, interpretação de atividades sísmicas, compressão de dados e reconhecimento de voz.

6. O Problema

Estre trabalho busca classificar páginas da internet pelo seu conteúdo, há uma grande dificuldade hoje no reconhecimento e classificação. Por exemplo, quando o conteúdo por palavras é utilizado em um servidor proxy, um acesso precisa ser feito a um endereço, e este endereço possui uma palavra que esta bloqueada, consequentemente o acesso a esta página será bloqueado. Nem sempre a página que possui apenas uma palavra faz parte de um conteúdo impróprio. Utilizando RNA podemos relacionar várias palavras de uma página e classificá-la de uma forma mais eficaz. Sites de busca podem ter um resultado mais eficaz, realizando uma busca contextualizada a um assunto de interesse.

6.1 Reconhecimento de padrões

Este trabalho caracteriza-se como um reconhecedor de padrões. O reconhecimento de padrões envolve três níveis de processamento: filtragem da entrada, extração de características e classificação [9]. A filtragem da entrada de dados tem o objetivo de eliminar dados desnecessários ou distorcidos fazendo com que a entrada apresente apenas dados relevantes para o reconhecimento do objeto em análise. A extração de características consiste da análise dos dados de entrada a fim de extrair e derivar informações úteis para o processo de reconhecimento.

O estágio final do reconhecimento de padrões é a classificação, onde através da análise das características da entrada de dados o objeto em análise é declarado como pertencente a uma determinada categoria.[9]

Quando busca-se realizar o reconhecimento de padrões em modelos estáticos, em especial a fase de classificação que é bastante onerosa, esses modelos são eficientes apenas quando suposições de limite são satisfeitas. A eficiência de modelos estáticos depende de um grande conjunto de suposições ou condições sobre as quais o modelo é construído. Para que o modelo seja empregado com sucesso, é necessário que os usuários possuam um bom conhecimento sobre as propriedade dos dados analisados e das capacidades do modelo [10]. As redes neurais são uma alternativa promissora para vários métodos de classificação convencionais. Elas possuem vantagens como ser adaptativas em função dos seus dados, ou seja, são capazes de se ajustar a si próprias sem a necessidade de qualquer especificação explícita. As redes neurais também são modelos não lineares capazes de modelar com flexibilidade as complexas relações do modelo do mundo real.

7 Implementação

O sistema foi implementado em linguagem Java, utilizando a IDE Netbeans. Testes de performance foram praticados em alguns hardwares diferentes, o algoritmo desenvolvido obteve uma ótima performance. O número máximo de neurônios foi de 600.000.
O desempenho do aplicativo depende das condições da Internet de onde ele esta sendo utilizado.

Para representação dos pesos foi utilizada um espaço vetorial de dimensão 3, que corresponde às ligações dos neurônios com as entradas, são nestas ligações onde ficam armazenadas os pesos. Um vetor representa a entrada, cada posição do vetor possui um índice (i), e cada entrada possui uma ligação e um peso com cada neurônio de índice (j , k) de um espaço vetorial de dimensão 2, formando assim um espaço vetorial de dimensão 3, conforme mostra a Figura 2.

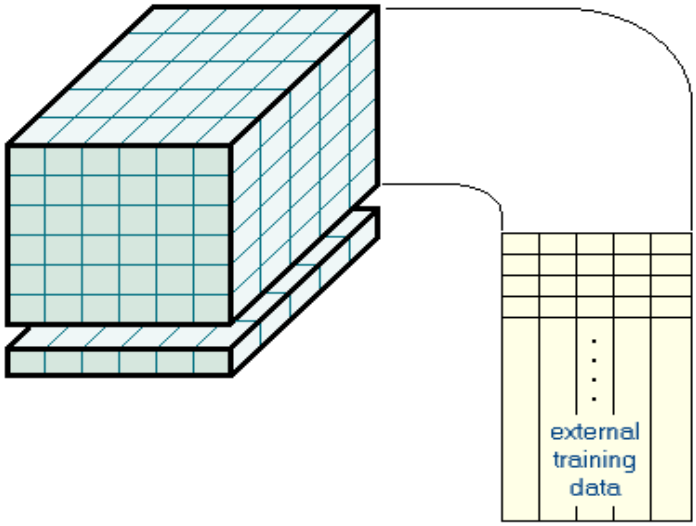

Figura2. Representação do conhecimento armazenado(pesos) [8].

O sistema permite a entrada de um endereço(url), a partir deste endereço será feita uma extração de seu conteúdo (palavras).

Após a extração das palavras de um site, um filtro processa somente as palavras que é necessário classificar, essas palavras ficam armazenadas em um arquivo de texto que podem ser facilmente adicionadas ou removidas. Vários sites foram analisados, e as palavras comumente encontradas em sites da mesma classe foram adicionadas a este arquivo.

Cada palavra é representada de forma numérica em uma variável de ponto flutuante. A Tabela 1 exemplifica algumas palavras e suas respectivas representações.

Palavra	**Representação numérica**
Assine	0.125827
Automobilismo	0.422155
Futebol	0.160995
Tecnologia	0.921023
Tempo	0.722192
Mulheres	0.991217
Indicadores	0.321127
Graduação	0.276014
Palavra não existente	0

Tabela1, representação numérica das palavras.

7.2.1 Pesos iniciais

Inicialização dos pesos é feita aleatoriamente com valores em double de 0 até 1.

7.2.2 Cálculo do neurônio vencedor

O cálculo dos pesos é feito através da distância euclidiana de cada entrada até cada neurônio conforme mostra a Fórmula 1.

$$Dist = \sqrt{\sum_{i=0}^{i=n} (V_i - W_i)^2} \quad 1$$

Onde W é o valor do peso do neurônio e V o valor da entrada.

O neurônio vencedor é aquele que possui a menor distância euclidiana.

7.2.3 Ajuste dos pesos (Aprendizado)

O Ajuste dos pesos (Aprendizado) é efetuado para o neurônio vencedor e também os seus vizinhos. No sistema implementado são considerados os 8 vizinhos de um espaço vetorial de dimensão 2. A fórmula 2 foi utilizada para o ajuste dos pesos.

$$W(t+1) = W(t) + L(t)(V(t) - W(t)) \quad 2$$

Onde w é o valor do peso do neurônio t, L é a taxa de aprendizado, V é o valor de entrada.

7.2.4 Parâmetros da rede

A rede foi desenvolvida para ter parâmetros flexíveis, quantidade de neurônios, tamanho da entrada, taxa de aprendizado e épocas. O Algoritmo desenvolvido pode ser utilizado em várias aplicações seguindo as características aqui apresentadas, como valores de entrada numérica.

7.2.5 Armazenamento dos pesos.

O armazenamento dos pesos pode ser feito por um botão através do aplicativo, os dados da matriz que representa os pesos é salvo então em um arquivo.

A recuperação destes dados poderá ser feita posteriormente através de uma funcionalidade implementada na interface gráfica (botão carregar pesos).

7.2.7 Interface Gráfica do sistema

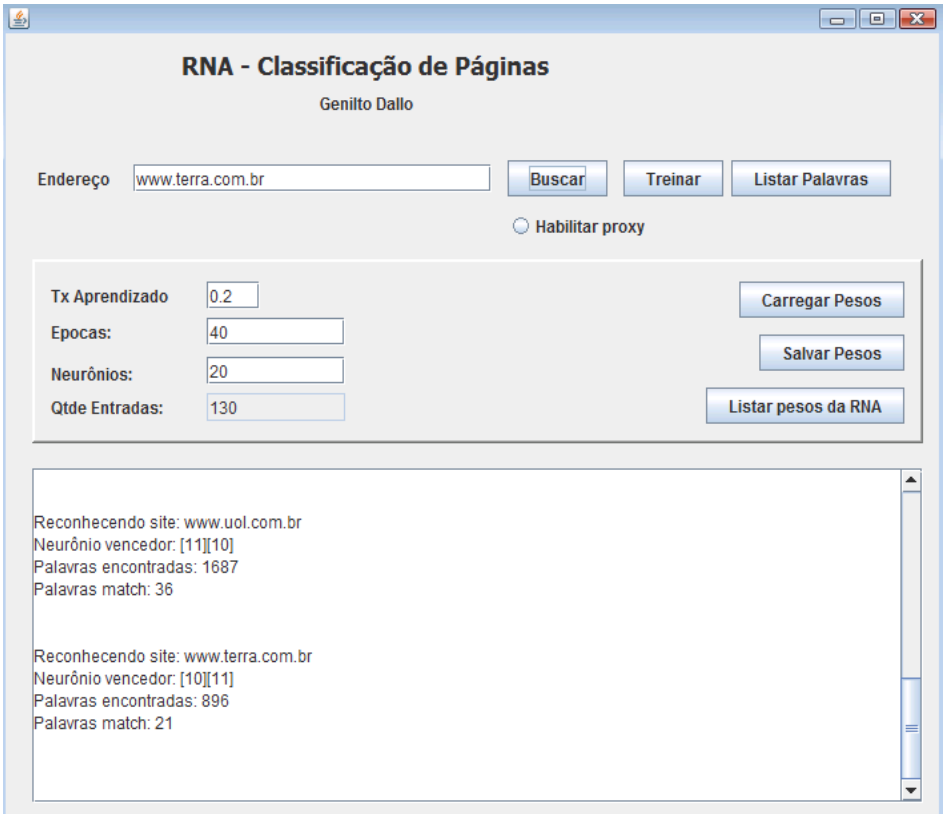

Figura 3. Interface gráfica do sistema.

8.Resultados

Os resultados obtidos foram satisfatórios para o problema proposto. A rede desenvolvida conseguiu obter os dados e classificá-lo através da RNA apesar de algumas restrições.

Algumas páginas utilizam tecnologias em que o sistema não conseguiu obter palavras necessárias para poder fazer a classificação. Somente páginas que utilizam a linguagem HTML e que possuem texto podem ser classificadas. Através da tabela 2, podemos ver alguns resultados depois de alguns treinamentos.

Endereço	Quantidade de Palavras	Neurônio vencedor
Www.uol.com.br	1653	[3][3]
Www.bol.com.br	1129	[3][2]
Www.terra.com.br	939	[3][2]
Www.yahoo.com.br	113	[3][2]
Www.unipar.br	355	[4][18]
Www.unicentro.br	516	[4][13]
Www.caixa.gov.br		[17][8]
Www.hsbc.com.br		[17][7]
Www.santander.com.br		[17][8]

Tabela 2. Resultados obtidos em testes com o sistema.

A quantidade de neurônios de entrada foi de 130, sendo que 130 palavras foram utilizadas.A quantidade de neurônios na rede foi de 40.000. Taxa de aprendizagem de 0.4 e 40 épocas.

A partir dos resultados obtidos, podemos associar um determinado número de neurônios como se fossem uma classe.

9. Conclusão

As aplicações de reconhecimento de padrões aproveitam a capacidade de aprendizado e a grande capacidade de processamento das redes neurais a fim de obter identificações de padrões dentro de categorias previamente estabelecidas mais rapidamente. Além disso, quando uma rede neural possui o treinamento adequado, ela consegue tolerar e contornar algumas diferenças nos dados analisados de cada site, oferecendo um reconhecimento de padrões eficiente.

As redes neurais são uma excelente técnica para a classificação por padrões, e podem ser utilizadas nas mais diversas situações, basta compreender o contexto do problema e aplicá-la com as configurações que mais se adaptam a ele.

Devido a inconsistência dos dados obtidos, como por exemplo alguns sites de mesma classe possuírem palavras diferenciadas em seu conteúdo, o reconhecimento pela RNA foi dificultado. Pretende-se utilizar outros padrões de entrada, para melhorar a sua classificação.

Para demonstrar a flexibilidade do algoritmo apresentado, um webservice foi desenvolvido para fazer a classificação através de um site de busca. Os pesos também podem ser treinados, utiliza o mesmo arquivo de armazenamento de pesos do aplicativo. Este site pode ser acessado pelo endereço: http://www.wimep.com.br/xus/

9. Referencia bibliográfica

[1] MONTEIRO, P.P.. (2007) Filtragem de páginas Web baseada em redes neurais artificiais de Hopfield, ftp://docentes.puc-campinas.edu.br/pub/professores/ceatec/juan/TCC/PedroMonteiro/Pedro%20Monteiro-TCC-Monografia.pdf, agosto 2009.

[2] MARIN, A..(2003) Um Mecanismo para Filtragem de Paginas da Web baseado no modelo de Rede Neural Artificial de Hopfield, 91 f. Dissertação (Mestrado em Informática). Centro de Ciências Exatas, Ambientais e de Tecnologias. Pontifícia Universidade Católica de Campinas, Campinas.

[3] ARBIB, M. A. (1964) Brains, Machines and Mathematics. McGraw-Hill.

[4]BARRETO, J. M. (2002) Introdução às Redes Neurais, http://www.inf.ufsc.br/~barreto/tutoriais/Survey.pdf, agosto 2009.

[5] JOEY R.. (1997) Object-oriented neural networks in C++, London: Academic Press.

[6] MANCINI et al. (2006) Aplicação de redes neurais artificiais no auxílio ao diagnóstico de crianças respiradoras bucais e nasais. Anais do X Congresso Brasileiro de Informática em Saúde, Florianópolis.

[7] Pavel B. (2002) "Survey of clustering data mining techniques", http://www.ee.ucr.edu/~barth/EE242/clustering_survey.pdf, agosto 2009.

[8] SOM Tutorial, http://www.ai-junkie.com/ann/som/som5.html , agosto 2009.

[9] JESAN, J.P.. (2005) "The neural approach to pattern recognition", http://www.acm.org/ubiquity/views/v5i7_jesan.html, agosto 2009.

[10] ZHANG, G.P..(2000) "Neural networks for classification: A survey". IEEETSMC: IEEE Transactions on Systems, Man, and Cybernetics..

[11] LIEBSTEIN, L. H. (2002) "Data Mining – Teoria e Prática" , http://www.inf.ufrgs.br/~clesio/cmp151/cmp15120021/artigo_lourdes.pdf, Outubro 2009.