

Página de buscas utilizando Inteligência Artificial

Genilto Dallo

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

E-mail: geniltodallo@gmail.com

Resumo. *Este artigo aborda a utilização de técnicas de Inteligência Artificial para uma página de buscas. Desenvolveu-se três agentes e uma Rede Neural Artificial(RNA). O Agente WebSeeker busca e armazena endereços, o WebSpider busca e armazena conteúdo, o Agente de Busca que disponibiliza as informações e a RNA classifica as páginas.*

Palavras-chave: *Agentes, Sistemas Multi-Agentes, Sites de busca, WebSpider, RNA, Redes Neurais, Kohonen.*

Abstract. *This paper reports the use of Artificial Intelligence techniques to a search engine. Developed three agents and one Neural Artificial Network(RNA). LinkSeeker agent, searches and store links, WebSpider agent search and store content, the Search Agent provides the informations and the RNA ranks pages.*

Keywords: *Agents, Multi-Agent systems, search engines, WebSpider, Neural Artificial Networks.*

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, o que dificulta a tarefa de recuperação e classificação desses dados.

A necessidade atual deste tipo de ferramenta é enorme, já que estamos na era da informação, ferramentas que facilitam o acesso a informação são vitais em âmbito geral. Uma ótima aplicação para utilização de técnicas inteligentes é a indexação, armazenamento e classificação para uma busca posterior, ajudando a organizar o conteúdo da Internet.

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 aborda a implementação de uma página de busca utilizando técnicas de Inteligência Artificial, desenvolveu-se um SMA com três agentes, o WebSpider para busca e armazenamento do conteúdo, o LinkSeeker para busca e armazenamento de endereços, e o Agente de Busca, que disponibiliza o conteúdo e faz a busca no banco de dados, e uma Rede Neural Kohonen para a classificação as páginas.

2.Agentes Inteligentes

Segundo Russel e Norvig [1], um Agente é tudo o que pode ser considerado capaz de perceber seu ambiente por meio de sensores e agir sobre este ambiente por intermédio de atuadores. Este conceito é ilustrado na figura 1.

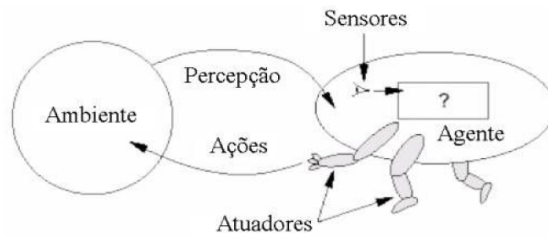


Figura 1. Arquitetura básica de um agente [1].

Um agente pode ser visto como uma entidade que pode perceber um ambiente e baseado nessa percepção, interagir sobre ele [1]. O exemplo mais comum de agente é o agente robótico. Ele percebe o ambiente através de algum dispositivo de entrada, como uma câmera de vídeo, por exemplo. E, com base no que foi analisado através da câmera, ele é capaz de exercer alguma ação sobre o mundo a seu redor.

De acordo com [2], um sistema de agentes é semelhante a uma sociedade de seres, que resolvem problemas e trocam informações. Este sistema é um sistema aberto, onde qualquer agente pode entrar ou deixar a sociedade. O termo é utilizado para denotar hardware ou software que possui as seguintes propriedades:

- a) Autonomia - o agente deve poder funcionar sem intervenção humana, baseando suas ações em seu conhecimento armazenado sobre o ambiente;
- b) Reatividade - os agentes percebem o ambiente e respondem a mudanças que ali ocorrem;
- c) Proatividade - os agentes são capazes de tomar iniciativas, exibindo comportamentos dirigidos ao objetivo;
- d) Mobilidade - habilidade do agente poder se mover ao redor da rede;
- e) Flexibilidade - aceita a intervenção de outros agentes;
- f) Aprendizagem - aprende com as informações oriundas do ambiente e com os outros agentes;
- g) Habilidade social - diz respeito à capacidade dos agentes de interagir com outros agentes através de algum tipo de linguagem de comunicação.

2.1 Sistemas Multi-Agentes (SMAs)

Segundo [3], nos SMAs, os agentes são a parte central que devem cooperar e trocar conhecimento para obter a solução de problemas antes desconhecidos, ou seja, o projetista não volta sua atenção para um problema específico, mas para um domínio específico, onde o processo de coordenação das ações dos agentes ocorre em tempo de execução. A forma menos distribuída do raciocínio distribuído é aquela em que um único agente [4]:

- a) Decompõe o objetivo em sub-objetivos;
- b) Atribui as submetas aos vários outros agentes.

Este tipo de raciocínio em geral é chamado de planejamento multi-agente. A primeira etapa, decomposição do problema, é essencialmente a mesma de sistemas de planejamento com um único agente. O ideal é que a decomposição resulte em um conjunto de subproblemas mutuamente independentes. Uma vez produzida a decomposição, o subproblema precisa ser alocado aos agentes disponíveis para execução. Neste ponto, o planejamento distribuído difere do planejamento com um único agente nos seguintes aspectos importantes [1]:

- a) Se os agentes escravos não forem todos idênticos, o agente-mestre precisará ter acesso aos modelos dos recursos dos vários escravos. Esses modelos permitem a alocação de tarefas aos agentes com melhores condições para executá-las;
- b) Mesmo que todos os agentes escravos sejam idênticos, o mestre precisa fazer um balanço da carga para assegurar que o objetivo global seja concluído o mais rápido possível.

Uma vez distribuídas as tarefas, é necessário uma sincronização entre os agentes, caso elas não sejam completamente independentes. No planejamento com um único agente, as dependências em geral são trabalhadas no momento da criação do plano. Em um sistema com vários agentes, nem sempre é possível fazer isso, já que um esquema estático estaria fadado ao fracasso, caso os vários

agentes demorassem um tempo imprevisível para executar suas tarefas [5].

Para dar suporte à ideia de múltiplos agentes interagindo para solucionar problemas necessita-se do estabelecimento de um meio para que estes possam se comunicar. Os agentes podem criar seus próprios planos, compostos por ações de solução de problemas e de ações de comunicação, às vezes, endereçadas a um agente específico ou a uma estrutura de memória compartilhada, onde certamente será lida por todos os outros [3].

3.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 [6].

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 [7].

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 [8].

3.1 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 [9].

O 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 [10].

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 [10].

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 [11].

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 saída. Cada nó da camada de entrada tem a função de distribuir os valores padrões para a de saída, que é um conjunto de nós organizados de forma tabular. O vetor de entrada possui seus elementos conectados com cada nó da camada de saída por meio de ligações, as quais são responsáveis por manter atualizados os valores durante o processo de treinamento da RNA

[9]. A figura 2 mostra de maneira simplificada a estrutura de uma rede Kohonen.

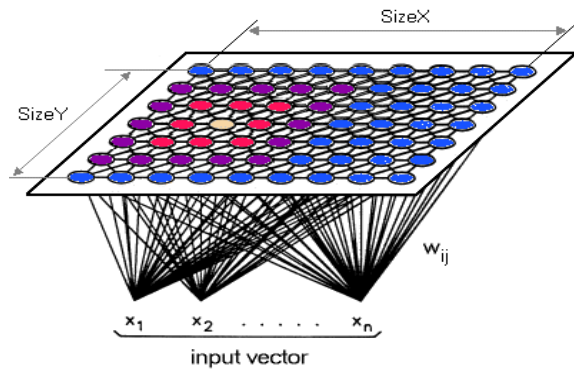


Figura 2. Representação da uma rede Kohonen [12].

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 [12].

4. O Problema

O crescimento constante do conteúdo na Internet, que tem como consequência a existência de uma enorme quantidade de informações, dificulta a tarefa de recuperação dados.

Desde sua criação, a WEB (*World Wide Web*) 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, estima-se que existam mais de 135,1 milhões de páginas na WEB. Há também uma projeção que indica que a taxa de crescimento atual da Internet levará à existência de aproximadamente duzentos milhões no ano de 2010 [7].

As ferramentas de busca na Internet tem como objetivo principal realizar a coleta e armazenamento das informações encontradas, disponibilizando para seus usuários este conteúdo de forma organizada, oferecendo diferentes maneiras de pesquisa em sua base de dados.

4.1 Ferramentas de busca

As Ferramentas de Busca podem ser definidas como programas ou páginas especializadas em localizar e/ou descrever informações na Internet, ou seja, é um instrumento de pesquisa na WEB através de palavras-chave ou categorias, orientadas por texto. Estas são também conhecidas como Mecanismo de Busca, *Search Engines*, Sites de busca, etc [13].

Hoje quase todas as pessoas que precisam da Internet para trabalhar ou pesquisar usam as Ferramentas de Busca. Esse tipo de serviço começou há apenas alguns anos atrás, em 1990, quando surgiu a primeira Ferramenta de Busca, chamada Archie [13].

Quando teve início a WEB, foi também criada a primeira Ferramenta de Busca, em 1993, por Matthew Gray, e ela foi chamada de "Wandex". E o Aliweb, que apareceu no mesmo ano e ainda funciona [13].

Em 1994 o WebCrawler foi criado, e foi a primeira Ferramenta de Busca de "texto cheio", que deixava as pessoas procurarem por qualquer palavra em qualquer página da WEB. O Lycos, em 1994 e tornou-se a Ferramenta de Busca mais popular da década. Muitas surgiram depois disso, como o Infoseek, o Inktomi, o Northern Light, o AltaVista e o Excite. Elas competiram com populares, como o Yahoo!, que surgiu em 1994, que fornecia uma descrição com cada endereço. Mais tarde, em 1998, a Ferramenta de Busca que se tornaria a mais popular atualmente foi lançada: o Google [13].

O Google teve sua ascensão em torno de 2001, porque ele possui um sistema de *Ranking*, que diz se uma página é mais popular do que outra. Ele utiliza vários critérios para determinar a relevância dos assuntos dos sites. Em 2003, o Yahoo! usava o Google para seus resultados de pesquisa, mas em 2004 ele criou sua própria Ferramenta de Busca. O site do gênero mais recente é o Bing, feito pela Microsoft e a Yahoo em 2009, que juntamente com o Google, dominam o mercado hoje.

5. Implementação dos Agentes

O sistema desenvolvido possui três agentes. Dois desenvolvidos em Java, o LinkSeeker que faz a procura por endereços(URL) e o WebSpider que acessa estes endereços e armazena o conteúdo. O sistema WEB foi desenvolvido em PHP, que faz a busca neste mesmo banco de dados, este agente denominado agente de busca.

Segundo [14], os agentes que fazem buscas autônomas na Internet também são chamados de aranhas(*spiders*), agentes viajantes(*wanderers*), rastejadores(*crawlers*), ou vermes(*worms*). E define estes como sendo programas que o computador hospedeiro da ferramenta de busca lança regularmente na Internet na tentativa de obter dados sobre o maior número possível de documentos para integrá-los, posteriormente, à sua base de dados. Os agentes WebSpider e LinkSeeker age, desta forma, acessando o conteúdo na Internet de forma autônoma e grava os dados obtidos no banco de dados.

5.1 O Agente WebSpider

A função deste agente é, a partir do endereço(URL) de uma página, extrair o seu conteúdo e armazenar em um banco de dados. E, enquanto houver endereços no banco de dados repetir o procedimento. Se este agente receber uma mensagem do Agente de Buscas, ele armazena este pedido e após realizar a tarefa atual, ele faz uma busca na Internet por essa informação. Este agente é considerado autônomo, pois age por conta própria.

O agente possui as seguintes características: percepção, conhecimento, decisão e ação. A percepção contemplará as observações do agente durante a interação com a Internet, recebendo o conteúdo destas páginas. O conhecimento do agente é representado pelas informações que possuirá sobre as páginas da Internet, o qual é atualizado durante cada interação, e armazenado no banco de dados. A decisão se dá pelo endereço de um determinado site, com regras dizendo se ele pode ou não acessar aquele site. A ação se dá a partir do endereço do site, acessando o conteúdo do mesmo. O Sensor é um endereço no qual inicia-se uma pesquisa por um site, e durante a execução o próprio agente passa como parâmetro pra ele mesmo outros endereços para fazer a pesquisa.

5.2 O Agente LinkSeeker

A função deste agente é semelhante ao WebSpider, a partir do endereço(URL) de uma página, extrair outros endereços por ela referenciados e armazenar estes endereços em um banco de dados. E, enquanto houver endereços no banco de dados repetir o procedimento. Este agente é considerado autônomo, pois age por conta própria.

O agente possui as seguintes características: percepção, conhecimento, decisão e ação. A percepção contemplará as observações do agente durante a interação com a Internet, recebendo os endereços destas páginas. O conhecimento do agente é representado pelas informações que possuirá sobre as páginas da Internet, o qual é atualizado durante cada interação, e armazenado no banco de dados. A decisão se dá pelo endereço de um determinado site, com regras dizendo se ele pode ou não acessar aquele site. A ação se dá a partir do endereço do site, acessando o conteúdo do mesmo. O Sensor é um endereço no qual inicia-se uma pesquisa por um site, e durante a execução o próprio agente passa como parâmetro pra ele mesmo outros endereços para fazer a pesquisa.

5.3 O Agente de busca

A função deste agente é, a partir palavras ou frase, fazer uma busca no banco de dados e mostrar o resultado para o usuário. É um agente reativo, pois aguarda uma ação para uma reação do usuário.

O agente possui as seguintes características: percepção, conhecimento, decisão e ação. A percepção se dá com a interação com o usuário, recebendo uma entrada, esta é uma palavra ou uma frase que o usuário deseja buscar na base de conhecimento. O conhecimento do agente é representado pelas informações que possuirá sobre as páginas da Internet, que é armazenado no banco de dados. A decisão se dá a partir do conteúdo enviado pelo usuário. A ação é a busca que será efetuada no banco de dados e disponibilizada para o usuário.

Os sensores deste agente são o formulário de entrada e o botão de busca, o formulário para o preenchimento de uma palavra ou frase, e o botão que é para executar o comando de busca.

Fazem parte do ambiente do SMA, o servidor no qual eles estão hospedados, o servidor WEB, que disponibiliza o site de busca, a Internet onde é realizada a pesquisa e o banco de dados, o qual é utilizado para o armazenamento do conhecimento. Considerando que trata-se de um SMA, o sistemas trocam informações e interagem entre o agente WebSpider e o busca.

Os agentes compartilham o conhecimento armazenado com os dados do banco de dados MySQL. A comunicação é feita através da troca de mensagens entre o agente LinkSeeker e o Agente de Busca.

O sistema de busca desenvolvido está disponível *online*¹. A figura 3 mostra a interface do sistema, que possui um campo para preenchimento do texto ou palavra que se deseja buscar, e o botão busca que executará a ação no sistema.



Figura 3. Interface do Agente de Busca

6. Implementação da RNA

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 utilizado 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 4.

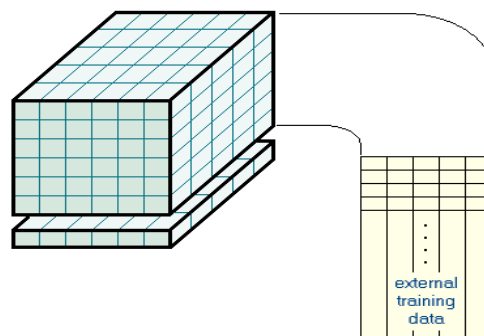


Figura4. Representação do conhecimento armazenado(pesos) [12].

¹ Sistema de busca disponível em: <http://www.voicomp.com.br/xus/>

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 banco de dados. 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.

A rede foi treinada com vários conjuntos de palavras mais relevantes para cada classe de página. O resultado(neurônio vencedor) dirá a qual classe a página pertence. Para a utilização neste trabalho, foram criadas 10 classes de páginas que podem ser observadas na tabela 2.

Classe:	Neurônio Vencedor
Portal	[3][3]
Universidades	[4][18]
Universidades	[4][13]
Bancos	[17][8]
Jogos	[7][19]
Esporte	[10][13]
Adulto	[1][2]
Tecnologia	[6][9]
Cinema	[19][15]
Lazer	[4][7]
Economia	[8][12]

Inicialização dos pesos é feita aleatoriamente com valores em *double* de 0 até 1. 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. E no sistema representará como uma classificação para uma determinada página. 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.

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

2

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

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, e com valores de entrada numérica.

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. Resultados

O sistema funcionou como o planejado. O agente WebSpider e o LinkSeeker percorreram de maneira autônoma a Internet, pesquisando vários sites e armazenou corretamente no banco de dados. O agente de busca fez a busca no banco de dados retornando a pesquisa do usuário de forma correta. A RNA mostrou-se adequada para o problema, classificando na maioria dos casos corretamente as páginas. Na figura 5 pode-se ter uma visão geral da arquitetura do sistema.

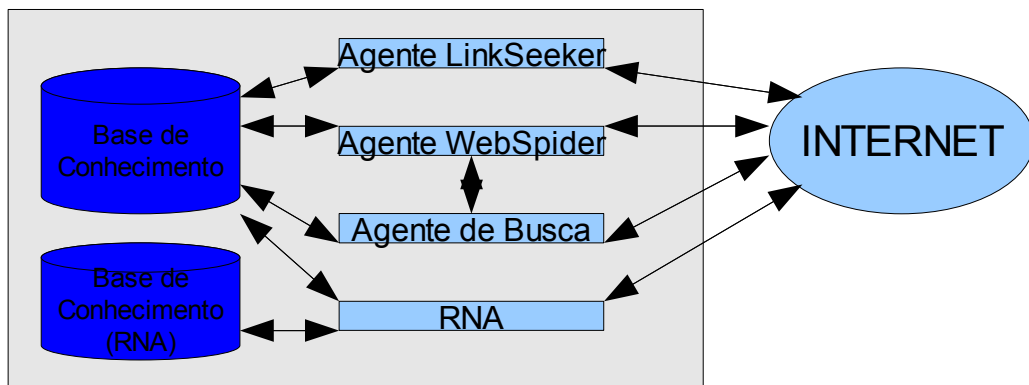


Figura 5 . Arquitetura do sistema desenvolvido.

A pesquisa pode ser feita desde uma única palavra até uma frase. O sistema mostra os resultados encontrados com paginação, indicando o número total de páginas encontradas e sua classificação que foi dada pela RNA. Na figura 6 é mostrado o resultado de uma busca no sistema.

Para a RNA a seguinte configuração foi satisfatória, quantidade de neurônios de entrada foi de 20, sendo que 20 palavras de cada classe foram utilizadas. A quantidade de neurônios na rede foi de 40.000. Taxa de aprendizagem de 0.4 e 40 épocas.



Resultados: 12 para: Esportes

[Terra - Notícias, vídeos, esportes, economia, diversão, música, moda, fotolog, blog, chat](#)

Terra - Notícias, vídeos, esportes, economia, diversão, música, moda, fotolog, blog, chat Ir ao topo da página Ir ao menu Ir ao conteúdo Ir para a busca Ir ao rodapé Terra Todos os canais Busca Radiona WebRadiono TerraBusca Email Horóscopo Tempo Assinaturas e serviços Notícias Esport
<http://www.terra.com.br/portal/#>

[Terra - Notícias, vídeos, esportes, economia, diversão, música, moda, fotolog, blog, chat](#)

Terra - Notícias, vídeos, esportes, economia, diversão, música, moda, fotolog, blog, chat Ir ao topo da página Ir ao menu Ir ao conteúdo Ir para a busca Ir ao rodapé Terra Todos os canais Busca Radiona WebRadiono TerraBusca Email Horóscopo Tempo Assinaturas e serviços Notícias Esport
<http://www.terra.com.br/portal/#menu>

[Terra - Notícias, vídeos, esportes, economia, diversão, música, moda, fotolog, blog, chat](#)

Terra - Notícias, vídeos, esportes, economia, diversão, música, moda, fotolog, blog, chat Ir ao topo da página Ir ao menu Ir ao conteúdo Ir para a busca Ir ao rodapé Terra Todos os canais Busca Radiona WebRadiono TerraBusca Email

Figura 6. Resultados de uma busca

8. Conclusão

Este artigo apresentou um sistema utilizando técnicas de Inteligência Artificial para uma página de busca. Como proposto anteriormente, foi utilizado e melhorado para se adaptar a este problema os resultados de trabalhos anteriores desenvolvidos para a matéria de Inteligência Artificial.

Pode-se mostrar com este trabalho, que técnicas de Inteligência Artificial podem ser utilizadas em várias áreas, e para uma página de buscas são várias as técnicas que podem ser aplicadas para melhorar o resultado das buscas.

O sistema é inteligente porque consegue armazenar conhecimento, tem comunicação entre os agentes, toma ações autônomas e reage com a ação de um usuário. Possui aprendizado através da rede neural que através do treinamento é capaz de reconhecer a classe de uma página.

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. A vantagem de utilizar RNA é de que essa técnica é uma das melhores maneiras de reconhecimento de padrões.

A utilização de agentes é fundamental em uma página de busca, as vantagens são que o sistema pode atuar autonomamente, procurando novas páginas e armazenando elas no banco de dados de forma automática durante as 24 horas do dia. Sem eles as informações teriam que ser buscadas manualmente.

Como trabalhos futuros pretende-se criar novas técnicas de classificação de páginas e de ranking.

9. Referência bibliográfica

[1] RUSSEL, Stuart T.; NORVIG, Peter, (1995) *Artificial Intelligence: A Modern Approach*, Prentice Hall.

[2] ANDRADE, Adja F. (1999) *Proposta Metodológica para Criação de Roteiros em Ambientes Virtuais para Aplicação Educacional*, Dissertação de Mestrado, CPCC-UFSC, Florianópolis.

[3] FARACO, Rafael Avila.(1998) *Uma Arquitetura de Agentes Para Negociação Dentro do*

Domínio do Comércio Eletrônico. Dissertação de Mestrado, PPGE, UFSC, Florianópolis.

[4] RICH, Elaine; KNIGHT, Kevin.; (1994) *Inteligência Artificial*. Makron Books. São Paulo.

[5] WEISS, G. (1999) *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. MIT Press.

[6] MARIN, A. (2003) *Um Mecanismo para Filtragem de Páginas 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.

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

[8] BARRETO, J. M. (2002) *Introdução às Redes Neurais*, disponível em <http://www.inf.ufsc.br/~barreto/tutoriais/Survey.pdf>, acesso em agosto 2009.

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

[10] 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.

[11] PAVEL, B. (2002) *Survey of clustering data mining techniques*, disponível em http://www.ee.ucr.edu/~barth/EE242/clustering_survey.pdf, acesso em agosto 2009.

[12] *SOM Tutorial*, disponível em <http://www.ai-junkie.com/ann/som/som5.html> , acesso em agosto 2009.

[13] CECILIA, M. *Ferramentas de busca*, disponível em: <http://marcela.cecilia.zip.net/>, acesso em novembro de 2009

[14] CENDON, Beatriz. *Ferramentas de busca na WEB*, disponível em: <http://www.scielo.br/pdf/ci/v30n1/a06v30n1.pdf> , acesso em novembro de 2009.