Asociación Argentina
de Mecánica Computacional

Mecánica Computacional Vol XXIX, págs. 9075-9090 (artículo completo)
Eduardo Dvorkin, Marcela Goldschmit, Mario Storti (Eds.)
Buenos Aires, Argentina, 15-18 Noviembre 2010

# USO REDES NEURAIS MULTILAYER PERCEPTRON (MLP) EM SISTEMA DE BLOQUEIO DE WEBSITES BASEADO EM CONTEÚDO

**Emmanuel T. F. Affonso[a], Alisson M. Silva[a,b,c], Michel P. Silva[a], Thiago M. D. Rodrigues[b,c] and Gray F. Moita[c]**

[a]*Departamento de Computação, Centro Universitário de Formiga, Avenida Doutor Arnaldo de Senna, 328, 35570-000, Formiga, MG, Brasil, nelskt@gmail.com, michel.silva@gmail.com, http://www.uniformg.edu.br*

[b]*Departamento de Ciências Exatas, IFMG - Instituto Federal de Minas Gerais - Campus Bambuí, Faz. Varginha - Rodovia Bambuí/Medeiros - Km 05 - 38900-000 - Bambuí - MG - Brasil, alisson.marques@ifmg.edu.br, thiago.magela@ifmg.edu.br, http://www.cefetbambui.edu.br*

[c]*LSI - Laboratório de Sistemas Inteligentes, CEFET-MG - Centro Federal de Educação Tecnológica de Minas Gerais, Av. Amazonas, 7675, Nova Gameleira, 30.510-000, Belo Horizonte, MG, Brasil, alisson@lsi.cefetmg.br, gray@lsi.cefetmg.br, pema@lsi.cefetmg.br, http://www.lsi.cefetmg.br*

**Palavras-chave:** Sites, Redes Neurais Artificiais, Classificação, MLP.

**Resumo.** Este trabalho propõe a implementação de um sistema de classificação de *websites* como próprio ou impróprio, com base no seu conteúdo. Neste sistema os *websites* passam por um processo de preparação, no qual seu código fonte é padronizado e uniformizado, transformando as informações complexas presentes em cada *website* em informações mais simples. Isto permite uma redução no custo computacional da classificação e uma melhora no desempenho. Métodos estatísticos foram utilizados para selecionar as características mais relevantes de cada categoria. A partir desta seleção de características foram gerados vetores binários de entrada para o classificador. Como classificador empregou-se as Redes Neurais Artificiais *MultiLayer Perceptron* (MLP), cuja capacidade de generalização e adaptabilidade são características importantes para o problema em questão. Diferentes maneiras de preparação, métodos de seleção de características e configurações de rede foram experimentadas. Os resultados obtidos foram satisfatórios, com uma taxa de acerto superior a 95%.

## 1 INTRODUÇÃO

A internet é uma poderosa ferramenta e atualmente é a mídia que mais cresce em todo o mundo. Esse crescimento se deve aos seguinte fatores: acessível de qualquer lugar do mundo; alta interatividade e fluxo de informações; acesso fácil e direto ao conteúdo disponível; permite interação direta entre empresas e clientes; disponível 24 horas por dia, 7 dias por semana.

Segundo VeriSign (2009) o número de domínios registrados na internet até o primeiro trimestre de 2009 é de 183 milhões, o que representa um aumento de três por cento em relação ao quarto trimestre de 2008 e doze por cento em relação ao primeiro trimestre de 2008. No Brasil (domínio .br), de acordo com o RegistroBR (2009), foram registrados 1.798.253 domínios até 26 de agosto de 2009. O gráfico da Figura 1 apresenta o número de usuários de internet por continente e, o apresentado na Figura 2 ilustra a quantidade de usuários por país e mostra o Brasil com o maior número de usuários na América Latina.

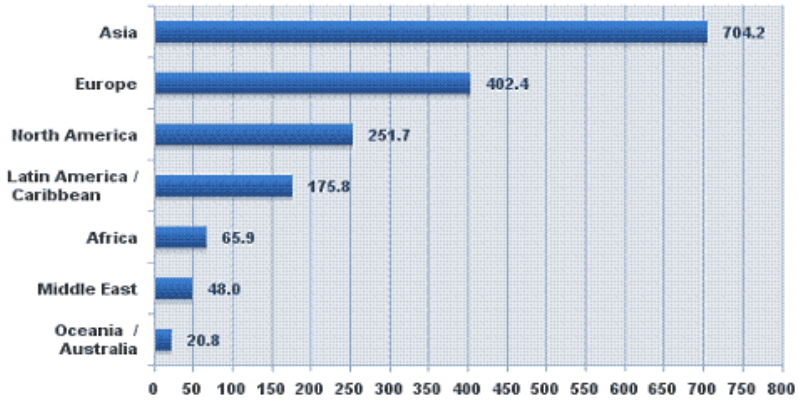


Figure 1: Usuários de internet no mundo.

Fonte - Stats (2009)

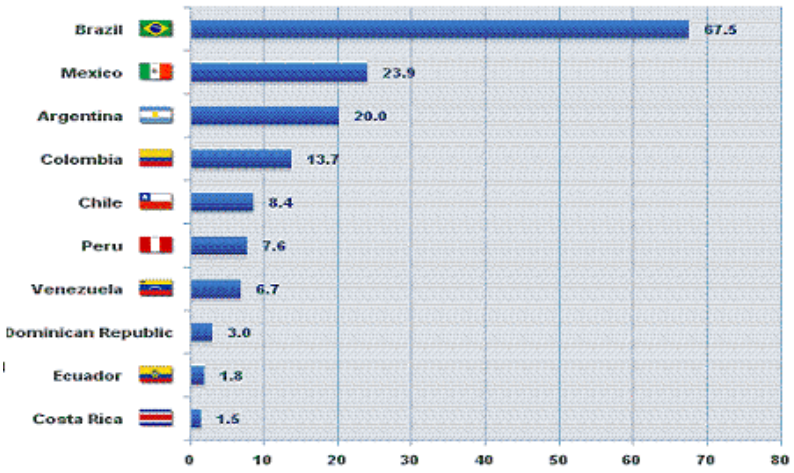


Figure 2: Usuários de internet na America Latina.

Fonte - Stats (2009)

Com o crescimento da internet as empresas passam cada vez mais a utilizá-la para suas atividades. Os negócios efetuados por meio da internet tem um crescimento superior ao crescimento do mercado e, cada vez mais, as pessoas optam por realizar suas compras pela internet. Segundo a Ebit (2009), a facilidade de comprar pela internet encanta cada vez mais pessoas no Brasil, o que pode ser comprovado pelo crescimento de 27% no faturamento do comércio eletrônico no primeiro semestre de 2009. Em 2009, 15,2 milhões de pessoas tiveram pelo menos uma experiência de compra pela internet e em meados de 2008 esse número era de 11,5 milhões de consumidores. Esse crescimento dá margem à expectativa de ultrapassar os R$ 10 bilhões em negócios virtuais até o final de 2010 no Brasil (Ebit, 2009).

> Pesquisa realizada pela *Price Waterhouse Coopers* afirma que o investimento mundial em entretenimento e mídia atingirá US$ 1,6 trilhão em 2013, com um ritmo anual de crescimento relativamente moderado, de 2,7%. O relatório *Global Entertainment and Media Outlook* divulgou que a migração para o entretenimento digital se acelerará à medida que as empresas buscam por mais eficiência em publicidade e distribuição, em meio à crise, e os consumidores procuram por mais controle sobre o conteúdo e mais valor (Focus, 2009).

O comércio eletrônico movimentou R$ 2,3 bilhões no Brasil no primeiro trimestre de 2009, obtendo um crescimento superior a 25% em relação ao mesmo período de 2008. No ano de 2008, o setor movimentou R$ 8,2 bilhões, com um crescimento 32% em relação a 2007 (Focus, 2009). O crescimento da utilização da internet para o comércio, entreterimento, cultura e outras atividades relevantes, traz consigo também a utilização deste meio de comunicação para a proliferação de *softwares* maliciosos, violência, apologia às drogas, pornografia, racismo, entre outras atividades ilícitas.

Pesquisa publicada pela SaferNet (2008) informa que 53% dos jovens internautas brasileiros já tiveram contato com conteúdos agressivos e que consideravam impróprios para sua idade. Além disso, quase 30% desses jovens afirmaram já ter encontrado presencialmente, ao menos uma vez, amigos que conheceram no mundo virtual. No ambiente empresarial, os funcionários vêm utilizando a internet para fins pessoais, com isso, reduzindo sua produtividade e, consequentemente, gerando prejuízos para as empresas. De acordo com estudo realizado pela Union (2009), a maioria dos funcionários que tem acesso a internet, passam, em média, 3 horas por dia em *sites* de relacionamento, lendo *e-mails* pessoais, fazendo pesquisa de produtos ou conversando com amigos (através de aplicativos de mensagens instantâneas ou *sites* de relacionamento). Com base nestas informações, pode-se concluir que mais de 35% do tempo de trabalho destes funcionários são dispendidos em atividades não relacionadas as atividades da empresa. Outro problema ocasionado pelo mal uso das possibilidades da internet no ambiente de trabalho (mas não somente) são os riscos a segurança devido a contaminação por vírus e demais *softwares* maliciosos. A prática de crimes virtuais como pornografia, pedofilia e racismo utilizando os recursos computacionais das empresas também é um fator preocupante.

Controlar o uso dos recursos computacionais dentro das empresas e o tempo dispendido pelos funcionários no uso inadequado destes recursos é um grande desafio para as empresas. Muitas optam por investir na educação e treinamento dos funcionários afim de reduzir esses prejuízos, outras empresas por bloquear e controlar o acesso a esses recursos.

Diversas ferramentas estão disponíveis para controlar e bloquear o acesso a internet, inclu-

sive algumas são *opensource*[1]. Entre essas ferramentas pode-se citar o *Squid*[2], *SquidGuard*[3], *DansGuardian*[4].

Diante deste cenário e visando apresentar uma alternativa para classificar *websites* sem fazer uso de *URL*, palavras chaves ou expressões regulares, este trabalho propõe um sistema de classificação de *websites* em livres ou bloqueados com base no seu conteúdo. No processo de classificação, características relevantes das duas classes de *websites* devem ser identificadas. Essas informações, disponibilizadas previamente, podem servir para, de algum modo, ensinar um sistema a fazer a classificação. As redes neurais artificiais mostram-se bastante adequadas para abordar o problema pela sua capacidade de ser ensinada empregando treinamento por meio de exemplos (Meireles et al., 2003).

## 2 REDES NEURAIS ARTIFICIAIS

Entender o funcionamento do cérebro humano e construir uma máquina que possa reproduzir suas habilidades, ainda que parcialmente, tem sido o sonho de gerações de pesquisadores. O cérebro processa as informações de forma totalmente diferente de um computador digital convencional. O cérebro é um computador altamente complexo, não-linear e paralelo, com a capacidade de organizar seus constituintes estruturais, os neurônios, de forma a realizar o processamento mais rapidamente que o mais rápido computador digital hoje existente (Haykin, 2001).

As RNA são modelos matemáticos que se assemelham às estruturas neurais biológicas e que têm capacidade computacional adquirida por meio de aprendizagem e generalização (Braga et al., 2000). Esses modelos almejam semelhança com o sistema nervoso dos seres vivos e a com sua capacidade de processar informações. Trata-se de uma metáfora da maneira como o cérebro humano processa as informações utilizadas em computação. A rede neural, vista como uma máquina adaptativa, é definida por Haykin (2001) como:

> um processador maciçamente paralelamente distribuído constituído de unidades de processamento simples, que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para uso. Ela se assemelha ao cérebro em dois aspectos:
>
> 1. O conhecimento é adquirido pela rede a partir de seu ambiente através de um processo de aprendizagem.
> 2. Forças de conexão entre neurônios, conhecidas como pesos sinápticos, são utilizadas para armazenar o conhecimento adquirido.

Para Zurada (1992), as redes neurais são sistemas celulares físicos com a capacidade de adquirir, armazenar e utilizar conhecimento experimental. Esse conhecimento está embutido na rede sob a forma de estados estáveis, que podem ser relembrados, em resposta à apresentação de estímulos. A RNA tem capacidade adquirida por meio de aprendizado e generalização. O aprendizado se dá por meio exemplos e a generalização é a capacidade de dar respostas coerentes para dados não apresentados na etapa de aprendizado. As RNA's são amplamente empregadas em tarefas de aproximações de funções, previsão de séries temporais, classificação e reconhecimento de padrões.

[1] *Opensource* conhecidos como *software* livre, deve ser gratuito e seu código fonte aberto
[2] http://www.squid-cache.org/
[3] http://www.squidguard.org/
[4] http://dansguardian.org/

A arquitetura da rede é definida pela forma na qual esses neurônios estão organizados e interconectados, ou seja, o número de camadas, o número de neurônios por camada, tipos de conexão entre os neurônios e a topologia da rede (Silva, 2009; Haykin, 2001). Existem diversos modelos para implementação de uma estrutura de rede neural artificial, como a SOM (*Self-organizing map*), RBF (*Radius Basis Function*), LMS (*Least Mean Square*), MPL (*Multi-Layer Perceptron*).

Na seção seguinte é apresentado o modelo MLP. Informações complementares sobre as arquiteturas de redes neurais artificiais podem ser encontradas em Haykin (2001); Braga et al. (2000); Zurada (1992).

### 2.1 MLP

As redes de múltiplas camadas distinguem-se das redes de camada simples pelo número de camadas intermediárias, aquelas entre a camada de entrada e a de saída. Essa arquitetura possui uma ou mais camadas ocultas, que são compostas por neurônios computacionais, também chamados de neurônios ocultos. Segundo Haykin (2001), a função dos neurônios ocultos é intervir entre a camada de entrada externa e a saída da rede de maneira útil. Adicionando-se uma ou mais camadas ocultas, tornamos a rede capaz de extrair estatísticas de ordem elevada. A habilidade dos neurônios ocultos extraírem estatísticas de ordem elevada é particularmente valiosa quando o tamanho da camada de entrada é grande. De acordo com Braga et al. (2000), atualmente, são os modelos neurais mais utilizados e conhecidos. Na Figura 3 encontra-se uma ilustração do modelo MLP. Cada camada do modelo MLP tem uma função específica:

- **Camada de entrada:** É uma camada não-computacional, nela não há processamento, responsável pelo recepção e propagação das informações de entrada para camada seguinte.
- **Camadas ocultas ou intermediárias:** Existem uma ou mais camadas ocultas, compostas por nós. São camadas computacionais, efetuam processamento, nelas são transmitido as informações por meio das conexões entre as unidades de entrada e saída. Essas conexões guardam os pesos que serão multiplicados pelas entradas, garantindo o conhecimento da rede (Silva, 2009).
- **Camada de saída:** Camada composta por neurônios computacionais, recebem as informações das camadas ocultas fornacendo a resposta.

O algoritmo de treinamento mais utilizado em modelos MLP é o *Backprogation*, que se baseia na aprendizagem por correção de erros. O algoritmo de *Backprogation* é um tipo de aprendizado supervisionado, quando o valor de saída é gerado o erro é calculado e seus valores são retro-propagados para entrada, os pesos são ajustados e os valores são novamente calculados. De acordo com Ferrari et al. (2006) o algoritmo de *Backprogation* funciona da seguinte maneira:

- Primeiramente apresenta-se um padrão à camada de entrada da rede.
- Esse padrão é processado camada por camada até que a camada de saída forneça a resposta processada.
- A resposta é comparada com a resposta desejada e se estiver errada, o erro é calculado.
- Os valores são retropropagados da camada de saída para a camada de entrada e conforme isso acontece, os pesos são ajustados e o processamento é feito novamente, até que se obtenha a resposta desejada.

## 3 CLASSIFICAÇÃO

A classificação é uma técnica utilizada para atribuir automaticamente um conjunto de textos a uma ou mais categorias predefinidas. A aplicação mais comum é na indexação de textos, sistemas de *data mining*, categorização de mensagens, notícias, resumos e arquivos de publicações periódicas. Rizzi et al. (2000) define a classificação de texto como uma técnica usada, principalmente, para descoberta do conhecimento, cujo objetivo é classificar documentos em relação a um conjunto de categorias predefinidas. É uma técnica para atribuir automaticamente um documento textual a um ou mais conjuntos.

O processo de classificação é menos complexo quando executado por seres humanos, devido à relativa facilidade em inferir conceitos a partir das palavras contidas nos documentos. No entanto, quando o número de documentos é grande o processo, apesar de simples, pode se tornar bastante demorado. Nos sistemas computacionais, o processo de classificação envolve técnicas para extrair as informações mais relevantes de cada categoria e utilizar estas informações para ensinar o sistema a classificar corretamente os documentos. O processo aplicado na classificação de *websites* com base no seu conteúdo, pode ser dividido em cinco etapas:

1. **Conjunto de dados e categorias:** esta etapa consiste em selecionar o conjunto de dados que será utilizado no processo de treinamento e teste do sistema e também as categorias presentes no conjunto;

2. **Preparação:** preparação ou pré-processamento é o processo de uniformização das informações presentes no conjunto de dados em que cada documento é analisado com o objetivo de remover as informações irrelevantes como acentuação, caracteres especiais, figuras, entre outros;

3. **Seleção das características:** visa selecionar, através de métodos estatísticos, as palavras mais relevantes, isto é, as que melhor representam as classes definidas;

4. **Vetor de características:** nessa etapa, as palavras selecionadas na etapa anterior são indexadas e utilizadas para compor o vetor de entrada para o agente classificador.

5. **Classificador:** a etapa final do processo é a utilização do vetor de características para realizar o processo de classificação.

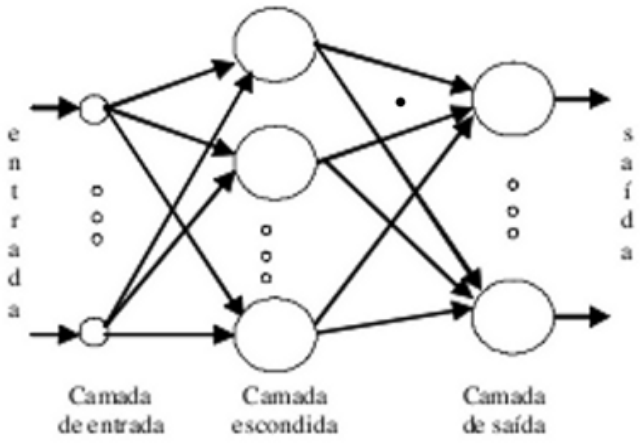


Figure 3: Modelo MLP .

A Figura 4 ilustra as etapas do processo que classificação que serão detalhadas nas próximas seções.

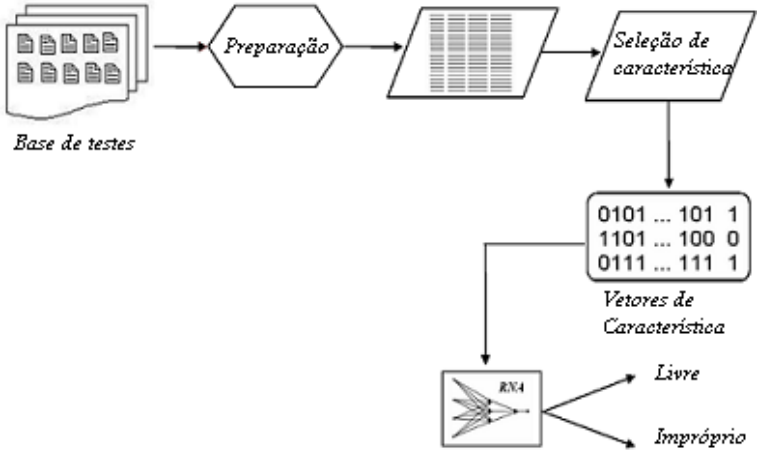


Figure 4: Diagrama das etapas do processo de classificação .

### 3.1 Conjunto de Dados

Segundo Silva (2009), um bom conjunto de dados de categorias diversas é de grande importância para o processo de classificação. Para a criação do conjunto de dados foi desenvolvido uma aplicação em Java. A aplicação recebe como parâmetro de entrada um arquivo texto contendo um conjunto de URL ou somente uma URL digitada diretamente na aplicação. A Figura 5 apresenta a tela inicial do sistema onde o usuário pode definir o arquivo contendo a lista de *sites* a serem copiados ou a URL de um *site* específico. O diretório local onde os *sites* serão armazenados também pode ser escolhido na tela principal da aplicação. A lista de *sites* foi obtida por meio dos *logs* de um servidor de internet.

A aplicação trabalha da seguinte forma: o *software* verifica na internet se existe um domínio para URL informada, se houver é feito o *download* do código fonte do *site*, e logo após é criado um arquivo com o mesmo nome da URL e todo o código fonte é descarregado no arquivo no diretório informado, caso contrário é gerado um erro. A Figura 6 ilustra o diagrama de funcionamento da aplicação. A base de *sites* é divida em 2 sub-conjuntos:

- **Livres:** *Sites* de categorias diversas, com conteúdo adequado, como sites de notícias, blog, pesquisas, etc;
- **Impróprios:** *Sites* de conteúdo impróprio, como pornôs, jogos, drogas, dentre outros.

### 3.2 Preparação dos sites

Esta etapa é responsável por uniformizar o conteúdo dos sites que se encontram na base de dados. A uniformização consiste em padronizar todo o conteúdo de um site[5], reduzindo a complexidade das informações presente no código fonte. Este processo permite um melhor desempenho na classificação.

[5]Neste trabalho quando nós referimos ao conteúdo dos sites, estamos falando sobre o seu código fonte.

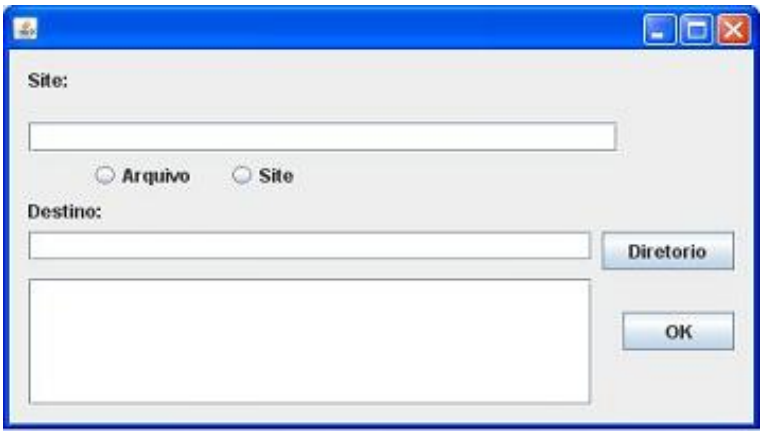


Figure 5: Software responsável pela coleta de sites .

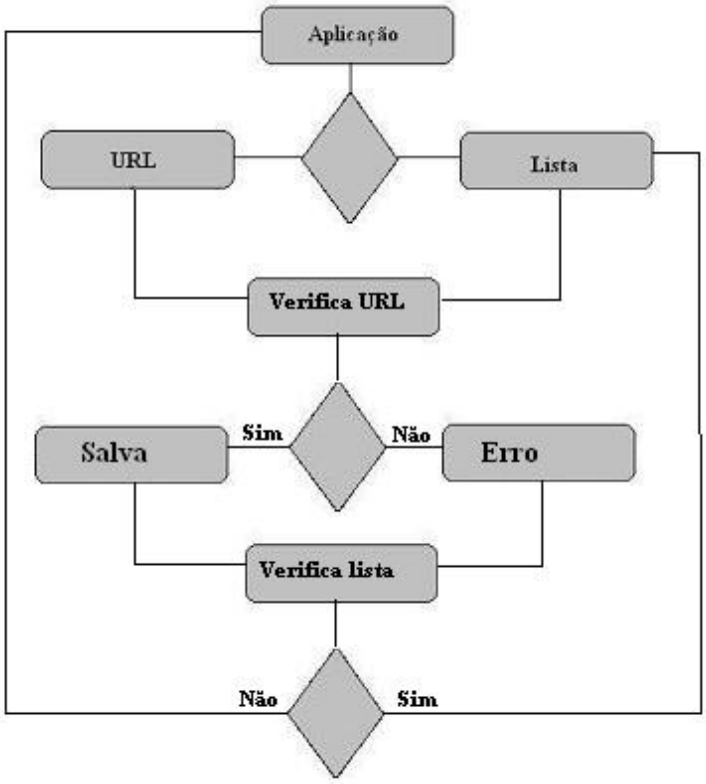


Figure 6: Diagrama da aplicação responsável pela coleta de sites .

De acordo com Assis (2006) é o pré-processamento que facilitará a seleção de características, permitindo eliminar informações irrelevantes e consequentemente facilitar o treinamento da rede, tornando a classificação mais rápida e eficiente. Para realizar a preparação dos dados foi desenvolvido uma aplicação baseada nas soluções propostas por Assis (2006) e Silva (2009). Neste trabalho o processo de uniformização foi dividido em três etapas: Processamento das *tags* HTML; Tokenização; Padronização.

As etapas do processo de uniformização serão detalhadas nas próximas seções. Para execução da preparação dos sites são empregados os seguintes diretórios:

- **Imprório:** diretório com a base de dados contendo os sites com conteúdo impróprio. Neste diretório os sites ainda não passaram pelo processo de uniformização.
- **Livre:** diretório com a base de dados contendo os sites com conteúdo regular. Neste diretório os sites ainda não passaram pelo processo de uniformização.
- **UniformFree:** diretório onde serão armazenados os sites com conteúdo regular, após o processo de uniformização.
- **UniformBlock:** diretório onde serão armazenados os sites com conteúdo impróprio, após o processo de uniformização.

### 3.2.1 Processamento do conteúdo HTML

O processamento HTML ocorre de três maneiras distintas, de acordo com o aproveitamento ou não das *tags* e conteúdo HTML. As três maneiras são apresentadas a seguir:

1. Neste todo o conteúdo é aproveitado, são removidos somente os delimitadores das *tags*. Exemplo:
   - **Conteúdo original:** <body leftmargin="0" topmargin="0" bgcolor="#ffffff" >
   - **Conteúdo uniformizado:** body leftmargin="0" topmargin="0" bgcolor="#ffffff" ;
2. Neste as *tags* são parcialmente aproveitadas, será removido tudo que não trás informações adicionais, o conteúdo da *tag* é analisado e somente informações relacionadas a cor (bgcolor="red" ou color="red") são aproveitaras. Exemplo:
   - **Conteúdo original:** <body leftmargin="0" topmargin="0" bgcolor="#ffffff" >
   - **Conteúdo uniformizado:** bgcolor="#ffffff" ;
3. Neste processo toda a *tag* é removida, aproveitando somente o conteúdo do site. Exemplo:
   - **Conteúdo original:** <h1>Redes Neurais<h1>
   - **Conteúdo uniformizado:** Redes Neurais ;

### 3.2.2 Tokenização

Tokenizar é decompor todo o código fonte do site em cada termo que o compõe, separando o código fonte em palavras. O delimitadores utilizados para decompor o código fonte foram ponto-e-vírgula, ponto, espaço, tabulação, vírgula e os sinais (+, -, =, < e >).

Neste projeto o processo de tokenização foi dividido em duas etapas. Em primeiro momento as mensagens foram separadas possuindo delimitadores como espaço e tabulação, para possibilitar a padronização dos *links* e *e-mails*. Após esta etapa os *e-mails* e *links* são padronizados. Logo em seguida o processo de tokenização deste projeto discorre da seguinte maneira:

- É verificado todas as pontuações como vírgula, ponto, ponto-e-vírgula. Toda pontuação encontrada será removida e a mensagem dividida. Exemplo:
  - **Conteúdo original:** "meu.projeto,redes;neurais"
  - **Conteúdo uniformizado:** "meu" "projeto" "redes" "neurais"

### 3.2.3 Padronização

A etapa de padronização consiste em uniformizar todas as referências à *links* e *e-mails* encontrados e também uniformizar todas as acentuações, juntamente com cedilhas e caracteres maiúsculos. Informações como *links* e *e-mails* seriam irrelevantes sem passarem por uma uniformização, a chance de encontrar referências a um mesmo endereço seriam pequenas. A seguir discorre a forma que são uniformizados os *links*, *e-mails* e palavras:

- **E-mails:** Verifica-se a palavra é algum *e-mail* válido e padroniza para /EMAIL/. Exemplo:
  - **Conteúdo original:** "emmanuel@uniformg.edu.br"
  - **Conteúdo uniformizado:** /EMAIL/
- **Links:** Todos os *links* encontrados serão substituídos por /LINK/. Exemplo:
  - **Conteúdo original:** "http://www.WEBSITE.com.br/"
  - **Conteúdo uniformizado:** /LINK/.
- **Palavras:** Acentuações, cedilhas serão removidas e caracteres maiúsculos padronizados em minúsculo. Exemplo:
  - **Conteúdo original:** "aDminIstRação"
  - **Conteúdo uniformizado:** "administracao".

## 3.3 Seleção de características

A seleção de característica consiste em um método que possibilita a extração das informações mais relevantes de um conjunto de dados. Os métodos de seleção de características visam remover características não informativas das classes e construir um conjunto de termos que facilite a identificação da categoria a qual ele pertence. Um dos grandes problemas para selecionar as características mais importantes é o grande volume de informação, e em um site muitas dessas características são irrelevantes. Neste projeto é empregado o método de seleção

de características de maneira a melhorar a eficácia da classificação e reduzir a complexidade computacional através da redução do número de termos.

Diferentes algoritmos podem ser empregados para realizar essa tarefa, dentre eles: Ganho de Informação ($IG$) (Wang et al. (2006); Sakkis et al. (2003)); Informação Mútua ($MI$) (Carpinteiro et al. (2006); Assis (2006); Ozgur et al. (2004)); $\chi_2$ *statistic* ($QUI$): (Assis (2006); Meyer and Whateley (2004); Yang and Pedersen (1997)); Distribuição por Frequência ($DF$) (Carpinteiro et al. (2006); Assis (2006); Wang et al. (2006); Drucker et al. (1999); Yang and Pedersen (1997)). Neste trabalho será utilizado o método DF por ter baixo custo computacional e pelos resultados obtidos nos trabalhos de Assis (2006) e Silva (2009).

### 3.3.1 Distribuição de freqüência (DF)

A distribuição de frequência possui um calculo simples, é definida pelo número de ocorrência de cada palavra extraída do conjunto de dados contidos na base de testes. A DF é uma das técnicas mais simples para redução da dimensionalidade, possui uma complexidade computacional aproximadamente linear, o que possibilita seu uso em grandes conjuntos de dados a um custo computacional relativamente pequeno (Silva, 2009). Dada equação 1:

$$DF = \frac{[x \in livres, improprios]}{T}, \tag{1}$$

onde, $N$ é o numero de ocorrência da palavra $x$ nas classes de $livres$ e $improprios$, e $T$ o número total de palavra dentro das classes. As palavras com valores DF mais altos são utilizadas para representar as categorias pré-definidas.

## 3.4 Vetor de características

Os vetores de características são os valores de entrada para a rede neural artificial, eles são criados a partir das $n$ características mais relevantes de acordo com o método de seleção de característica empregado, onde $n$ é o tamanho de cada vetor característico, o tamanho do vetor é gerado por meio da aplicação. Segundo Silva (2009) três das diferentes técnicas de compor os vetores são: Frequência do Termo, Peso Normal e Peso Binário.

Para este projeto foi escolhido o método de peso binário. O qual apresenta 1 para a palavra que aparece ao menos uma vez no site e 0 para a que não aparece. Nos testes realizados no projeto de Silva (2009), este método apresentou os melhores resultados na classificação, menor consumo de tempo e recursos computacionais durante o processamento. Os vetores são gerados empregando os três métodos de processamento HTML descritos, e a aplicação permite a criação de vetores com 15 a 100 características. Nesses conjuntos de vetores cada linha representa um site e cada célula ou coluna dessas linhas é uma característica, sendo representadas pelo valor um (1) se existe aquela característica no site e zero (0) se não existe. A última célula indica se o site é *livre* (1) ou *impróprio* (0). A Figura 7 apresenta um exemplo de vetor com 15 características.

# 4 EXPERIMENTOS

A rede neural foi implementada com 50 ou 100 elementos na camada de entrada e 5, 25 ou 50 na camada oculta, a camada de saída foi composta por um único neurônio. Cada uma destas configurações de rede foi executada 10 vezes para as três formas de uniformização dos dados. Os resultados apresentados ilustram a média dos resultados obtidos em cada configuração de rede. Como medida de desempenho foi utilizado:

- FPR: taxa de falso positivo, site livre classificado como impróprio;
- FNR: taxa de falso negativo, site impróprio classificado como livre;
- TPR: taxa de verdadeiro positivo: site livre classificado como livre;
- ACC: percentual de sites classificados corretamente.

Nos experimentos com as redes MLP foi empregada a técnica de validação cruzada. Segundo Haykin (2001), na validação cruzada o conjunto de dados é dividido aleatoriamente em um conjunto de treinamento, teste e validação. O conjunto de treinamento é utilizado no processo de aprendizagem da rede. O de teste serve para interromper a aprendizagem, quando o erro médio quadrático para os dados de teste começa a aumentar de forma contínua. Os dados de validação são utilizados para verificar a capacidade de generalização da rede. O conjunto de dados foi distribuído da seguinte forma: 60% para treinamento, 20% para teste e 20% para validação Demuth et al. (2008).

## 5 RESULTADOS OBTIDOS

Os resultados dos experimentos com 50 e 100 características são apresentados nas próximas seções. Na primeira seção é apresentado os resultados obtidos empregando os vetores de 50 características, na seção seguinte, os vetores de 100 características.

### 5.1 Resultados com 50 características

A Tabela 1 apresenta os resultados alcançados utilizando o método de processamento HTML que faz uso parcial das *tags*, aproveitando somente parte do seu conteúdo HTML. A Tabela 2 ilustra os resultados obtidos com vetores criados com o aproveitamento total das *tags* HTML. Já a Tabela 3 faz uso somente do conteúdo HTML, ignorando totalmente as *tags*. A Tabela 4 apresenta os melhores resultados cruzando todas as linhas e colunas das tabelas geradas com os vetores de 50 características.

Analisando os resultados obtidos empregando as três formas de processamento HTML e vetores com 50 características, os melhores resultados foram atingidos utilizando o processo de exclusão total das *tags* HTML, exceto no percentual de sites impróprios classificados como livres empregando 50 camadas ocultas na arquitetura de rede usada.

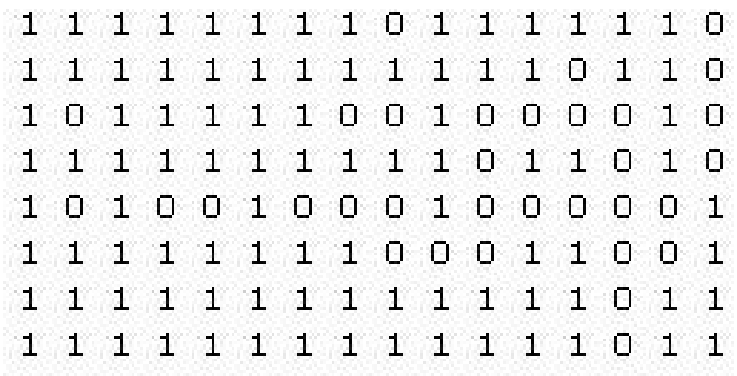


Figure 7: Vetores de características - Imagem gerada com todas as *tags*.

Table 1: Vetores gerados com o processamento parcial das *tags* HTML.

| Camada Oculta | FPR | FNR | TPR | ACC |
|---|---|---|---|---|
| 5 | 13,58% | 19,95% | 80,05% | 83,39% |
| 25 | 13,16% | 22,02% | 77,98% | 82,76% |
| 50 | 13,99% | 21,49% | 78,51% | 82,55% |

Table 2: Vetores gerados com o aproveitamento total das *tags* HTML.

| Camada Oculta | FPR | FNR | TPR | ACC |
|---|---|---|---|---|
| 5 | 17,15% | 37,46% | 62,56% | 76,52% |
| 25 | 12,72% | 45,99% | 54,01% | 76,53% |
| 50 | 18,93% | 37,69% | 66,31% | 76,81% |

Table 3: Vetores gerados com a exclusão total das *tags* HTML.

| Camada Oculta | FPR | FNR | TPR | ACC |
|---|---|---|---|---|
| 5 | 7,96% | 19,56% | 80,44% | 88,83% |
| 25 | 7,32% | 20,68% | 79,32% | 88,83% |
| 50 | 5,09% | 24,70% | 75,30% | 88,42% |

Table 4: Tabela com os melhores resultados para vetores com 50 entradas.

| Camada Oculta | FPR | FNR | TPR | ACC |
|---|---|---|---|---|
| 5 | 7,96% | 19,56% | 80,44% | 88,83% |
| 25 | 7,32% | 20,68% | 79,32% | 88,83% |
| 50 | 5,09% | 21,49% | 75,30% | 88,42% |

## 5.2 Resultados com 100 características

A Tabela 5 ilustra os resultados alcançados utilizando o método de processamento HTML que faz uso parcial das *tags*, aproveitando somente parte do seu conteúdo HTML. A Tabela 6 mostra os resultados obtidos com vetores criados com o aproveitamento total das *tags* HTML. Já a Tabela 7 faz uso somente do conteúdo HTML, ignorando totalmente as *tags*. A Tabela 8 apresenta os melhores resultados cruzando todas as linhas e colunas das tabelas geradas com os vetores de 100 características.

Table 5: Vetores gerados com o processamento parcial das *tags* HTML.

| Camada Oculta | FPR | FNR | TPR | ACC |
|---|---|---|---|---|
| 5 | 16,56% | 31,94% | 78,06% | 82,40% |
| 25 | 13,12% | 30,11% | 69,89% | 81,43% |
| 50 | 15,62% | 27,05% | 72,95% | 81,44% |

Table 6: Vetores gerados com o aproveitamento total das *tags* HTML.

| Camada Oculta | FPR | FNR | TPR | ACC |
|---|---|---|---|---|
| 5 | 12,20% | 43,68% | 66,32% | 80,00% |
| 25 | 13,95% | 24,49% | 75,51% | 82,11% |
| 50 | 8,72% | 41,33% | 58,67% | 80,44% |

Table 7: Vetores gerados com a exclusão total das *tags* HTML.

| Camada Oculta | FPR | FNR | TPR | ACC |
|---|---|---|---|---|
| 5 | 3,65% | 19,90% | 80,10% | 90,84% |
| 25 | 6,70% | 23,47% | 76,53% | 90,15% |
| 50 | 5,48% | 15,31% | 84,69% | 89,91% |

Table 8: Tabela com os melhores resultados para vetores com 100 entradas.

| Camada Oculta | FPR | FNR | TPR | ACC |
|---|---|---|---|---|
| 5 | 3,65% | 19,90% | 80,10% | 90,84% |
| 25 | 6,70% | 23,47% | 76,53% | 90,15% |
| 50 | 5,48% | 15,31% | 84,69% | 89,91% |

Analisando os resultados obtidos empregando as três formas de processamento HTML e vetores com 100 características, todos os resultados utilizando o processo de exclusão total das *tags* HTML foram melhores que nos outros métodos.

## 6 ANALISE DOS RESULTADOS

Observando os resultados, nota-se que foram atingidos resultados satisfatório. A RNA conseguiu dar respostas mais coerentes classificando vetores com 50 e 100 características. A Tabela 9 apresenta os melhores resultados:

Table 9: Tabela com os melhores resultados de todos os testes.

| Camada Oculta | FPR | FNR | TPR | ACC |
|---|---|---|---|---|
| 5 | 3,65% | 19,17% | 80,83% | 90,84% |
| 25 | 6,70% | 14,98% | 85,02% | 90,15% |
| 50 | 5,09% | 15,31% | 84,69% | 89,91% |

## 7 CONSIDERAÇÕES FINAIS

Foi proposto neste trabalho o desenvolvimento de um aplicativo classificador de sites, tendo como impróprios sites como pôrnos, violência, drogas, jogos, dentre outros, e livres os sites

como de notícias, blogs, esportes, infantis. A ferramenta proposta para este projeto faz uso de redes neurais artificiais e visa provar sua eficiência na classificação de *websites*.

Para que tais resultados fossem alcançados com sucesso, foram estudados diferentes técnicas de uso de redes neurais artificiais, como empregá-las, formas de manipulação, algoritmos de treinamentos, funções de ativação. Na revisão bibliográfica foram efetuados diferentes comparações entre aplicações que utilizam RNA, quais aplicações são mais semelhantes a aplicação proposta, os resultados obtidos e a eficiência da rede neural artificial, após todo esse processo de estudos, RNA que fazem uso da arquitetura MLP com algoritmo de treinamento *Backpropagation* e método de seleção de característica DF aparentou atender com maior eficiência a necessidade deste projeto. Com as técnicas de desenvolvimento da rede neural artificial já selecionada, foram estudas formas de preparações das informações, para este fim foi desenvolvido três formas de preparação do conteúdo HTML, para testar qual melhor atenderia as necessidades do projeto e mostrar diferentes resultados.

Dadas as palavras (característica) que apresentam maior relevância, cada site foi transformado em um vetor, para cada característica que for encontrada no site sua posição no vetor é preenchida com um (1) e para as que não forem encontradas no site sua posição no vetor será preenchida com zero (0), a última posição do vetor indica se o site é impróprio ou livres, (0) para impróprios e (1) para livres. Após todas essas etapas de preparação foi realizado o treinamento da rede neural artificial, foram testadas diferentes arquiteturas MLP, variando seu número de camadas ocultas. O número de entradas também foram alterados na busca de uma melhor relação custo/benefício. Para futuras melhorias do sistema poderia ser utilizadas outras técnicas de seleção de característica, outras arquiteturas de redes neurais artificiais, vetores característicos com outros tamanhos e como também ampliação de suas funcionalidades.

## REFERENCES


Assis J.M.C. *Detecção de E-mails Spam Utilizando Redes Neurais Artificiais*. Master's Thesis, Universidade Federal de Itajubá - Programa de Pós-Graduação em Engenharia Elétrica, 2006.

Braga A., de Carvalho A., and Ludermir T. *Redes Neurais ddd Artificiais: Teoria e aplicações*. LTC, 1 edition, 2000.

Carpinteiro O.A.S., Lima I., J. M. C. Assis A.C.Z.S., Moreira E.M., and Pinheiro C.A.M. A neural model in anti-spam systems. In *Artificial Neural Networks - ICANN 2006, 16th International Conference*, volume 4132 of *Lecture Notes in Computer Science*, pages 847–855. Springer, Athens, Greece, 2006. ISBN 3-540-38871-0.

Demuth H., Beale M., and Hagan M. *Neural Network Toolbox 6*. The MathWorks, Natic, MA, USA, 2008.

Drucker H., Wu D., and Vapnik V.N. Support vector machines for spam categorization. *IEEE Transactions on Neural Networks*, 10(5):1048 – 1054, 1999.

Ebit. Faturamento cresce 27 por cento no 1ž semestre. disponível em: http://www.ebitempresa.com.br. 2009.

Ferrari H.V., Hoto R., Maculan N., de Oliveira C., and de O. Camargo Brunetto M.A. Uma comparação entre redes neurais wavelet, lms, mlp e rbf para classificação de dpoc. *Foz 2006 Congresso de Matemática e suas Aplicações*, 2006.

Focus. E-commerce para pequenas e médias empresas. disponível em: http://www.focusconsultoria.com.br/blog/noticias/e-commerce-para-pequenas-e-medias-empresas. 2009.

Haykin S. *Redes Neurais*. Bookman, Porto Alegre, 2ł edition, 2001.

Meireles M.R.G., Almeida P.E.M., and Simões M.G. A comprehensive review for indus-

trial applicability of artificial neural networks. *IEEE Transactions on Industrial Electronic*, 50(3):585–601, 2003.

Meyer T.A. and Whateley B. Spambayes: Effective open-source, bayesian based, e-mail classification systems. In *Proceedings of the First Conference on Email and Anti-Spam*. CEAS, Mountain View, CA, USA, 2004.

Ozgur L., Gungor T., and Gurgen F. Adaptive anti-spam filtering for agglutinative languages: a special case for turkish. *Pattern Recognition Letters*, 25(16):1819 – 1831, 2004. ISSN 0167-8655. doi:http://dx.doi.org/10.1016/j.patrec.2004.07.004.

RegistroBR. Domínios registrados por domínio de primeiro nível. disponível em: http://www.registro.br/. 2009.

Rizzi C.B., Wives L.K., de Oliveira J.P.M., and Engel P.M. Fazendo uso da categorização de textos em atividades empresariais. In *Proceedings of the International Symposium on Knowledge Management/Document Management*. PUC-PR, Curitiba, PR, Brasil, 2000.

SaferNet. Pesquisa revela perigos nos acessos com internet. disponível em: http://www.safernet.org.br/site/noticias/pesquisa-revela-perigos-nos-acessos-com-internet. 2008.

Sakkis G., Androutsopoulos I., Paliouras G., Karkaletsis V., Spyropoulos C.D., and Stamatopoulos P. A memory-based approach to anti-spam filtering for mailing lists. *Information Retrieval*, 6(1):49 – 73, 2003. ISSN 1386-4564.

Silva A.M. *Utilização de Redes Neurais Artificiais para Classificação de Spam*. Master's Thesis, CEFET-MG - Centro Federal de Educação Tecnológica de Minas Gerais - Programa de Pós-Graduação em Modelagem Matemática e Computacional, 2009.

Stats I.W. Estimated internet users are 1668870408 people. disponível em: www.internetworldstats.com/stats.html. 2009.

Union S. Abusos feitos no uso dos recursos das empresas. disponível em: http://www.smartunion.com.br. 2009.

VeriSign. Internet domain names surpass 180 million in first quarter of 2009. disponível em: https://press.verisign.com/easyir/customrel.doeasyirid=afc0ff0db5c560d3version=liveprid =506988releasejsp=custom97. 2009.

Wang B., Jones G.J.F., and Pan W. Using online linear classifiers to filter spam emails. *Pattern Analysis and Applications (PAA)*, 9(4):339–351, 2006. ISSN 1433-7541. doi:http://dx.doi.org/10.1007/s10044-006-0045-7.

Yang Y. and Pedersen J.O. A comparative study on feature selection in text categorization. In *ICML '97: Proceedings of the Fourteenth International Conference on Machine Learning*, pages 412–420. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997. ISBN 1-55860-486-3.

Zurada J.M. *Introduction to Artificial Neural Systems*. PWS Publising Company, Boston, 1992.