ESCOLA POLITÉCNICA
DE PERNAMBUCO

Departamento de
Sistemas
Computacionais

Recuperação Inteligente de Informação: Construção de interface de navegação sobre mapa de documentos

Trabalho de Conclusão de Curso

Engenharia da Computação

Ubiracy dos Santos Rego Junior
Orientador: Prof. Renato Fernandes Corrêa

Recife, dezembro de 2006

UNIVERSIDADE
DE PERNAMBUCO

ESCOLA POLITÉCNICA
DE PERNAMBUCO

Departamento de Sistemas Computacionais

Recuperação Inteligente de Informação: Construção de interface de navegação sobre mapa de documentos

Trabalho de Conclusão de Curso

Engenharia da Computação

Este Projeto é apresentado como requisito parcial para obtenção do diploma de Bacharel em Engenharia da Computação pela Escola Politécnica de Pernambuco – Universidade de Pernambuco.

Ubiracy dos Santos Rego Junior
Orientador: Prof. Renato Fernandes Corrêa

Recife, dezembro de 2006

UNIVERSIDADE
DE PERNAMBUCO

Ubiracy dos Santos Rego Junior

Recuperação Inteligente de Informação: Construção de interface de navegação sobre mapa de documentos

Resumo

Com a grande quantidade de documentos digitais disponíveis, formular uma busca para retornar informações relevantes algumas vezes torna-se frustrante por diversas razões: a desorganização dos documentos na ordem em que são mostrados; a grande quantidade de informações não relevantes para a busca; e as diversas maneiras em que a informação pode ser representada. Portanto, um método automático de organização de documentos é preciso, agrupando os documentos de acordo com a similaridade textual. Esta monografia tem, como objetivo principal, apresentar um programa para construir uma interface para navegação interativa sobre mapas de documentos. Mapas de documentos são construídos usando mapas auto-organizáveis, que é um tipo de rede neural com grande potencial na classificação de documentos texto. O mapa é uma estrutura em que os documentos são representados em um plano bidimensional, mostrando suas relações de similaridade de acordo com os conteúdos textuais. A interface gerada pelo programa facilita a visualização de coleções de documentos e encontrar de documentos correlacionados.

Abstract

With the great amount of available digital documents, to formulate a search for good information results becomes frustrating sometimes for many reasons: the disorganization of documents in the order they are shown; the retrieved great amount of irrelevant information for the search; and the many ways in which information can be represented. Therefore, an automatic documents organization method is needed, grouping documents according to its textual similarity. This work has, as main objective, present a program to build an interface for interactive navigation on document maps. Documents maps are constructed using self-organizing maps, which is a sort of neural network with great potential on classifying text documents. The map is a structure in which documents are represented in a bidimensional plan, showing its similarity relations according to textual contents. The interface generated by the program facilitates the visualization of document collections and to find correlated documents.

Sumário

Índice de Figuras

Índice de Tabelas

Tabela de Símbolos e Siglas

SOM – self-organizing map (mapa auto-organizável)
HTML – hiper text mark-up language (Linguagem de Marcação de Hiper Texto)
SSOM – scaleable self-organizing map (mapa auto-organizável escalável)
GHSOM – growing hierarchical self-Organizing map (mapa auto organizável de crescimento hierárquico)
BMU – best match unit (melhor unidade vencedora)

Agradecimentos

Acabo de concluir uma das grandes etapas da minha vida, e agradeço a todos que colaboraram direta e indiretamente para a realização deste trabalho:

Aos meus pais, Maria da Conceição Félix e Ubiracy dos Santos, por todo apoio e condições que me ofereceram para que hoje pudesse realizar um sonho da minha vida.

Às minhas irmãs, que estiveram sempre presentes em todos os momentos da minha vida.

À minha namorada, Ana Paula, pelo carinho e compreensão nos momentos de tensão e alegrias durante a realização deste estudo.

Aos meus amigos que sempre me incentivaram a lutar por meus objetivos.

A todos os meus amigos da Procenge, que contribuíram com bons conhecimentos para minha vida profissional.

Ao professor Renato Fernandes, pelo aprendizado e pela paciência na caminhada deste trabalho.

Aos meus amigos da Escola Politécnica, pelo companheirismo e amizade. Aos integrantes do grupo de estudo: André Camara, André Henrique, Assis, Carlos Eduardo, Daniel, Eduardo Zoby, que sempre estiveram compartilhando conhecimentos nesta vida acadêmica.

Principalmente a Deus, que me presenteou com a amizade de todas essas pessoas que me ajudaram nessa grande realização da minha vida.

Capítulo 1

Introdução

Os Sistemas de Recuperação de Informação objetivam a realização das tarefas de indexação, busca e classificação de documentos (expressos na forma textual), a fim de satisfazer a necessidade de informação do indivíduo [16], geralmente expressa através de consultas. A necessidade de informação pode ser entendida como a busca de respostas para determinadas questões a serem resolvidas, a recuperação de documentos que tratam sobre determinado assunto ou ainda o relacionamento entre assuntos.

Hoje em dia, a localização de documentos através de engenhos de busca, geralmente, é feita com a utilização de buscas por palavras chaves ou expressões contidas nos documentos [11]. O sucesso em encontrar documentos relevantes depende do casamento dos termos fornecidos pelo usuário numa consulta, com os utilizados como índices na indexação da base de dados de documentos.

Com o crescimento das coleções de documentos digitais, os sistemas de recuperação de informação que localizam documentos utilizando buscas por palavras chaves e expressões simples têm se tornado cada vez menos efetivos. Este insucesso está relacionado aos seguintes motivos: a dificuldade do usuário em expressar o que ele realmente procura através de uma consulta; a forma desorganizada em que os documentos resultantes da busca são mostrados; o número excessivo de documentos retornados.

Com a vasta quantidade e variedade de documentos disponíveis, formular uma consulta efetiva para uma busca é uma tarefa difícil, e examinar uma lista resultante de uma pesquisa onde os itens são muitos e estão ordenados de forma claramente não significativa pode ser tediosa. Assim, tornam-se necessários métodos que sejam capazes de realizar uma organização automática dos documentos em conjuntos, evidenciando o relacionamento entre os conteúdos desses documentos, e as relações de proximidade entre os conjuntos de documentos de forma visual. Esta organização facilitará a navegação e a pesquisa sobre a coleção de documentos. Vários trabalhos na literatura de redes neurais têm sugerido o uso de mapa auto-organizável (do inglês *Self-Organizing Map* - SOM) [12] para a construção de sistemas de recuperação de informação baseados em mapa de documentos [4-5] [11] [13] [16-18] [21], pois é possível organizar um agrupamento de documentos de entrada em uma camada de saída visual, geralmente em duas dimensões.

Os mapas auto-organizáveis têm sido utilizados na criação dos mapas semânticos de documentos, mas apesar da grande aplicabilidade, não acontece uma grande exploração de metodologias de construção de mapas mais significativos e é muito difícil encontrar um programa

disponível que organize os documentos de acordo com a proximidade textual e facilite a navegação sobre um mapa de documentos.

1.1 Objetivos

Este estudo tem como objetivo gerar uma ferramenta de navegação interativa sobre mapa de documentos, além de aprender mais sobre as áreas de recuperação de informação, e de redes neurais artificiais.

Essa ferramenta será utilizada para facilitar a visualização de documentos que apresentam similaridades em relação a seus contextos, explicitando relações semânticas presentes na coleção de documentos. O sistema proposto é baseado em redes neurais artificiais, em particular, mapas auto-organizáveis. O modelo de rede neural utilizado pode ser encontrado em várias aplicações, pelo motivo de sua capacidade de organização topológica da base de dados treinada. Este trabalho vem contribuir para o incentivo a novas pesquisas sobre este tema.

1.2 Estrutura do Trabalho

No capítulo 2 são apresentados os conceitos necessários para entendimento dos Sistemas de Recuperação de Informação utilizando mapas de documentos. Neste capítulo também é descrito um breve histórico de alguns projetos presentes na literatura, focando as metodologias na construção de sistemas de recuperação de documentos texto usando mapa auto-organizável.

No capítulo 3 é apresentado de modo breve o funcionamento de um modelo de rede neural artificial conhecido como mapa auto-organizável.

No Capítulo 4 são descritas algumas técnicas de pré-processamento para representação dos dados a serem utilizados no experimento, como também a base de dados utilizada no trabalho, os processos de inicialização, treinamento e visualização do mapa de documentos, e como foi implementada a interface proposta no trabalho.

No capítulo 5 é apresentada a interface obtida a partir do mapa de documentos gerado.

No capítulo 6 são apresentadas as conclusões obtidas com a realização deste trabalho e também as contribuições e propostas para trabalhos futuros.

Capítulo 2

Sistemas de Recuperação de Informação Baseados em Mapas de Documentos

Na construção de um sistema de recuperação de informação baseado em mapa de documentos estão envolvidas as seguintes etapas: representação dos documentos, construção do mapa e construção da interface com usuário. Estas etapas são mostradas na Figura 1:

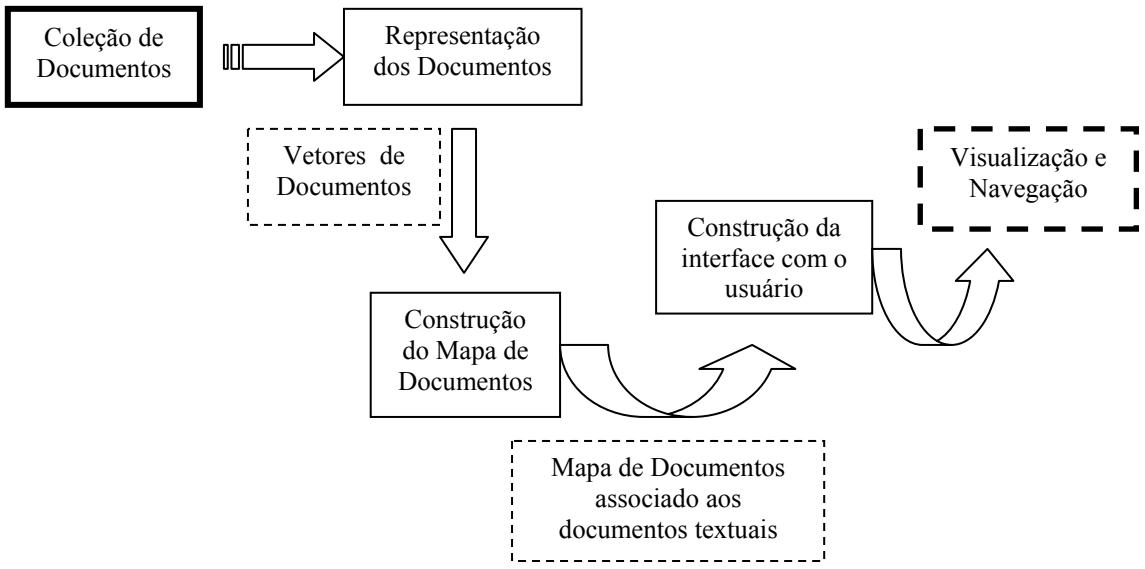


Figura 1. Diagrama de etapas na construção de um sistema de informação.

No tópico seguinte são descritos os passos envolvidos na construção de um sistema de recuperação de informação.

2.1 Representação dos Documentos

Em virtude da grande quantidade de documentos existentes atualmente, é necessário utilizar alguma forma abstrata para representação destes documentos para depois construir o mapa. Para que as redes neurais auto-organizáveis possam ser utilizadas na construção de mapas de documentos, é preciso que estes documentos sejam representados em vetores de características numéricas.A representação mais utilizada na literatura é o modelo de espaço vetorial que pode ser de dois tipos: através de vetores binários que indicam a presença ou ausência de palavras ou termos nos documentos; e histograma ponderado de palavras. Neste trabalho foi adotado o modelo de espaço vetorial [2] para representação de documentos textuais. Este modelo foi escolhido pela simplicidade de implementação e por atender bem aos propósitos ilustrativos deste trabalho.

Neste modelo, cada documento é representado por vetor onde cada componente do vetor significa a freqüência em que cada palavra ocorre no documento, tratando-se do histograma de palavras. Os histogramas denotam o conjunto de palavras utilizadas num documento em termos aparecimento de cada uma. A Tabela 1 exemplifica a representação de um documento por vetor, onde a primeira coluna representa a posição da palavra no vetor, a segunda coluna representa o número de ocorrências da palavra no documento e a terceira coluna descreve as palavras que representam o documento:

Tabela 1. Representação de um documento por vetor.

Índice no vetor	Freqüência	Termos no documento
1	4	campeonato
2	5	times
3	2	justiça
4	1	desorganização
5	3	anulação
6	1	prejudicados
7	2	jogadores

A grande desvantagem de se utilizar o modelo de espaço vetorial para representação dos documentos refere-se à grande dimensão do vetor histograma, ou seja, o grande número de palavras utilizadas para representar os documentos. O tamanho do vocabulário deve ser reduzido para que a dimensionalidade não se estenda muito. Para reduzir o tamanho do vocabulário podem-se utilizar métodos de seleção manual ou automática do conjunto de palavras mais adequadas seguindo algum critério [23]. Porém, encontrar um conjunto de palavras que represente bem as características essenciais de um documento é um problema complicado.

2.2 Fase de Construção do Mapa de Documentos

Esta fase corresponde à formação de agrupamentos de vetores de documentos. Este processo compreende o treinamento de uma rede neural SOM com um conjunto de vetores documentos obtidos na fase de representação de documentos. Para o treinamento das redes auto-organizáveis são necessários vários parâmetros, por exemplo, é preciso especificar: as dimensões do mapa,

números de épocas de treinamento, raio inicial e final de vizinhança, taxa de aprendizagem, além de vetores modelos iniciais.

O algoritmo SOM é utilizado quando se tem um grande conjunto de dados. A complexidade computacional é linearmente proporcional ao número de vetores de entrada. No entanto, a complexidade é quadraticamente proporcional ao número de unidades no mapa. O número de unidades contidas no mapa definem a precisão e a generalização dos mapas auto-organizáveis. O mapa SOM pode ser entendido como um mapeamento que conserva a topologia do espaço de entrada em uma grade bidimensional de unidades.

2.3 Visualização e Navegação pelo Mapa

Existem diversos modos para visualização do mapa SOM a fim de melhorar a compreensão dos padrões processados. Um dos modos mais utilizados é a representação através de um gráfico bidimensional onde se encontram os vetores de documentos mapeados. Para navegação, este gráfico pode ser mostrado ao usuário como uma interface conectada a várias páginas HTML permitindo a exploração dos documentos contidos em cada agrupamento. Podem ser incorporadas à imagem, palavras selecionadas automaticamente que caracterizam os documentos em cada região no mapa, a esse processo dá-se o nome de rotulação. As palavras identificadoras das categorias dos documentos ficam distribuídas sobre o mapa e funcionam como pontos de referência durante a navegação e servem também como sumário da coleção de documentos mapeados nas unidades. A visualização do mapa de documentos será o foco deste trabalho.

2.4 Estado da Arte

Esta seção mostra alguns dos projetos presentes na literatura referentes à construção de sistemas de recuperação de documentos texto usando redes neurais auto-organizáveis. Inicialmente os trabalhos utilizaram conjuntos de documentos pequenos, posteriormente os artigos passaram a focar a construção de mapas com grandes coleções de documentos e nas propostas de metodologias utilizadas, buscando obter mapas de melhor qualidade.

A primeira experiência na utilização de redes neurais para a área de recuperação de informação foi descrita em [18], o mapa de documentos obtido é mostrado na Figura 2.
Neste artigo foram utilizados redes SOM para classificação de 140 documentos, os quais foram identificados por 25 termos de indexação extraídos dos respectivos documentos. Estes termos de indexação foram obtidos através da remoção de palavras irrelevantes , as palavras restantes foram reduzidas a seus radicais, depois o processo foi de retirar os termos com mais freqüência e os que ocorriam em até três vezes. Os números no mapa indicam o número de documentos mapeados nas unidades correspondentes. Estes números revelam a distribuição dos documentos no mapa.

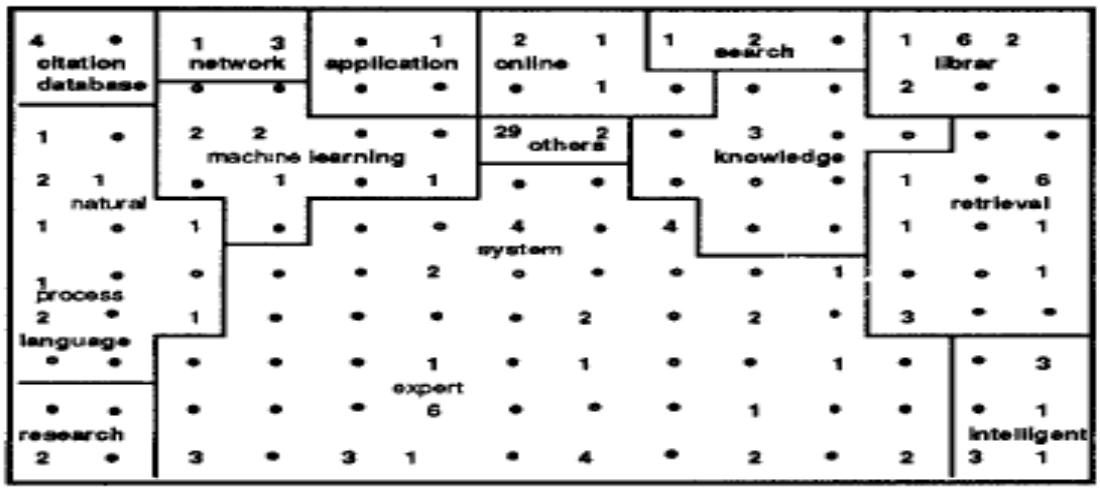


Fig. 3 . A self-organizing semantic map of AI literature. 140 documents from LISA database are used as input to produce the map. The areas on the map are automatically generated, their relative positions, neighbors, and sizes are determined by the input data. The numbers on the map represent the number of documents mapped to each node.

Figura 2. Mapa obtido de [18]

Em [21] é apresentado um sistema escalável de categorização de documentos e classificação textual baseado no algoritmo SOM, o chamado SSOM (*Scaleable Self-organizing Map)*. Neste trabalho, o algoritmo é utilizado para obter uma taxonomia hierárquica de conjuntos de documentos como também categorias descobertas neles. Os documentos são representados por vetores binários e depois do treinamento da rede as regiões no mapa representam as categorias presentes. Um rótulo é apresentado a uma unidade atribuindo o termo que tem maior valor no respectivo vetor modelo. A esse termo dá se o nome de termo vencedor. As regiões vizinhas que tiverem o mesmo termo vencedor são agrupadas para obter regiões maiores e o termo vencedor representará a categoria para toda a região. Os documentos que pertencem a uma mesma categoria são usados recursivamente para a construção de mapas menores que correspondem a um nível inferior na hierarquia de categorias. O SSOM foi aplicado nas coleções utilizadas em [3] [20] onde variou-se a dimensionalidade no intervalo de 25 a 400 termos, e também foi aplicado na coleção COMPENDEX. O principal alvo destes experimentos era medir a escalabilidade do algoritmo SSOM. A coleção COMPENDEX contém 247.721 resumos de artigos de campos relacionados à engenharia elétrica, sistema de informação, ciência da computação, etc. Depois do processo de indexação da coleção COMPENDEX, foram removidos 160.000 termos únicos sem contar com as palavras irrelevantes que estão contidas na *stop word list*, e também foram extraídas palavras que aparecem menos de três vezes. Dos termos restantes, os 10.000 mais freqüentes foram escolhidos como características para representação das categorias. Um mapa de 30 por 30 unidades foi treinado por 4.500.000 ciclos de treinamento, de modo que cada documento foi apresentado 15 vezes. O processo de treinamento durou 15,7 CPU horas numa máquina DEC Alpha 3000/600 200 MHz com 128 MB de RAM.

Outro experimento realizado teve a finalidade de classificar 10.000 *homepages* de entretenimento. O mapa gerado na organização das *homepages* é mostrado na Figura 3. Cada região no mapa representa um tópico relacionado a entretenimento (categoria) e o número de *homepages* categorizadas em cada região. Cada região pode ser clicada para visualizar um mapa mais detalhado com sub-tópicos classificados sob a região. Clicando-se no mapa com detalhes o usuário poderá visualizar as homepages reais.

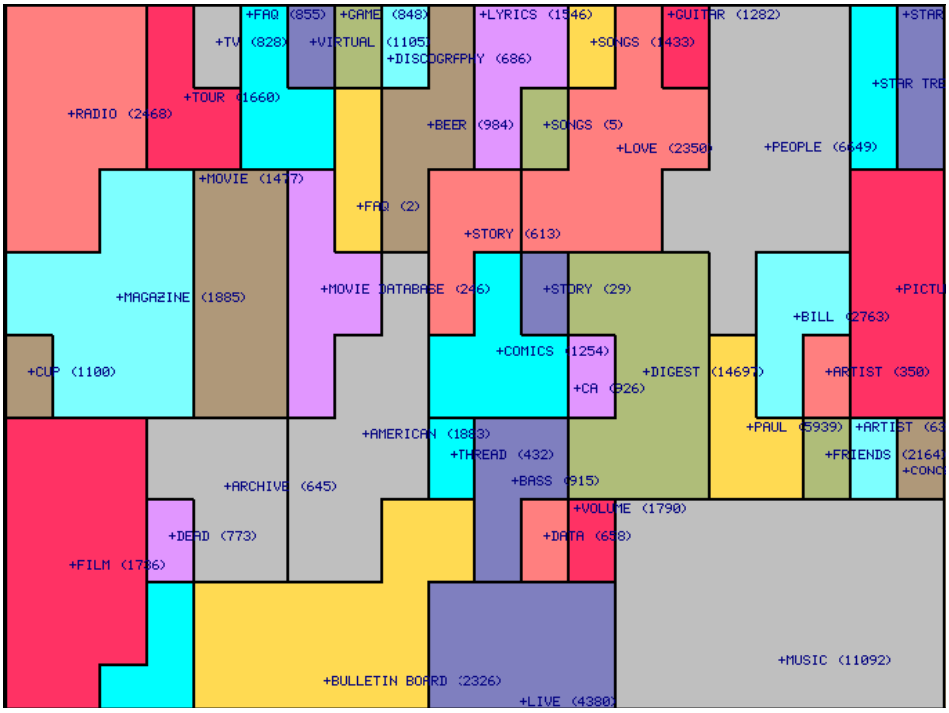
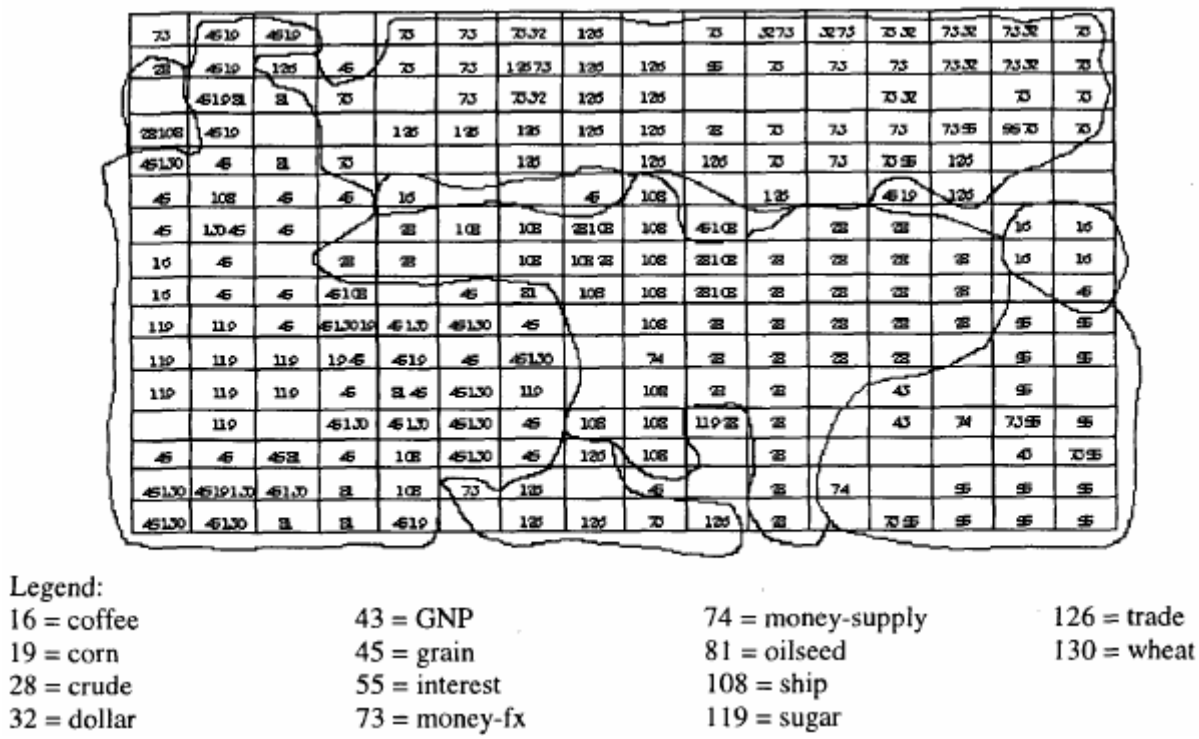


Figura 3. Mapa obtido de [21]

Em [1] é apresentado o projeto chamado *light* SOM, que é uma metodologia baseada em SOM para agrupamentos de grandes quantidades de documentos texto. O sistema herda as técnicas de redução de dimensionalidade do WEBSOM e a hierarquia estruturada do GHSOM, que é um método que permite uma decomposição hierárquica dependendo da distribuição dos dados [6]. a Figura 4 mostra o mapa de documentos obtido no projeto *light* SOM. O sistema procura melhorar na velocidade e na eficiência destes dois modelos, a fim de tornar-se apropriado para coleções grandes de documentos texto.



Legend:

16 = coffee	43 = GNP	74 = money-supply	126 = trade
19 = corn	45 = grain	81 = oilseed	130 = wheat
28 = crude	55 = interest	108 = ship	
32 = dollar	73 = money-fx	119 = sugar	

Figura 4. Mapa de documentos de [1]**.**

O WEBSOM foi um projeto desenvolvido no *Neural Network Research Centre da Helsinki University of Technology* liderado por T. Kohonen, e tinha o objetivo de organizar de forma automática coleções arbitrárias de documentos textos não estruturados no intuito de facilitar a navegação e exploração destas coleções de documentos [8]. O WEBSOM utiliza o algoritmo SOM para organizar automaticamente os documentos em um mapa de unidades. O projeto pode ser visualizado no endereço eletrônico: http://websom.hut.fi/websom/. A Figura 5 representa a interface gráfica bidimensional intuitiva criada para a navegação no mapa SOM. A imagem do mapa é criado a partir de mensagens do *newsgroupcomp.ai.neural-nets* disponível no site do projeto WEBSOM. Para rotular o mapa foi criada uma metodologia de rotulação automática de unidades e regiões [17]. A interface do mapa permite que documentos em interessantes áreas do mapa possam ser vistos. Ao clicar em uma região no mapa permite-se visualizar uma região mais detalhada com tópicos relacionados, que aparecem em áreas próximas no mapa. Às unidades mais similares no mapa são atribuídas cores similares e as palavras chaves distribuídas sobre o mapa servem para fornecer informações sobre os tópicos evidenciados pelos documentos e como pontos de referência durante a navegação.

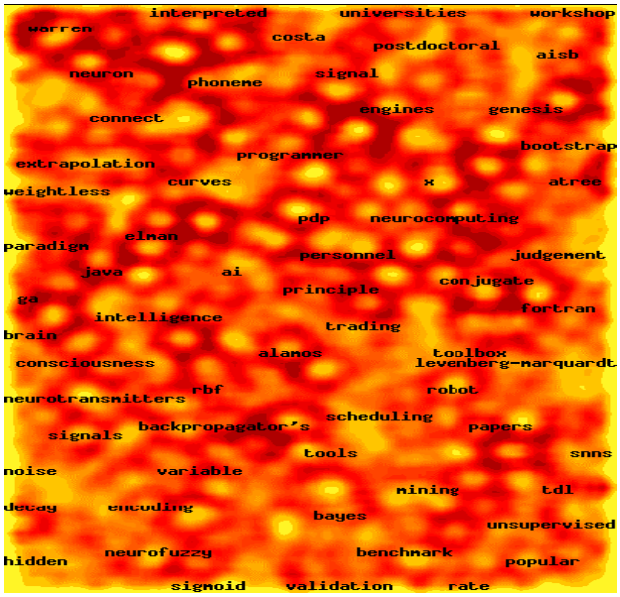


FIGURA 5. Mapa de mensagens do *newsgroup comp.ai.neural-nets* disponível no servidor WEBSOM.

Capítulo 3

Mapas Auto-Organizáveis

Este capítulo tem por objetivo descrever de modo breve o funcionamento de uma espécie de redes neurais conhecida como Mapas Auto-Organizáveis [14] [15].

O Mapa Auto-Organizável é um tipo de rede neural artificial baseada em aprendizado competitivo, onde os neurônios de saída competem entre si para serem ativados, sendo que somente um neurônio fica ativo em determinado momento, dando-se o nome de neurônio vencedor. Esta rede é capaz de mapear um conjunto de padrões de entradas fornecido à rede em um conjunto finito de neurônios organizados em um arranjo geralmente de uma ou duas dimensões.
O desenvolvimento deste tipo de rede neural foi motivado pela maneira em que se encontram arranjados os neurônios do córtex cerebral. Pesquisadores têm demonstrado, por exemplo, que estímulos sensoriais como os produzidos pelo sentido do tato, auditivos e visuais são mapeados em diferentes áreas do córtex cerebral de uma forma ordenada, supondo um mapa computacional, como referência construtiva básica na estrutura de processamento de informação do sistema nervoso [7] [9].

3.1 Estrutura Básica

O mapa auto-organizável é uma rede de duas camadas. A primeira camada é a camada de entrada. Normalmente, a segunda camada é a camada competitiva e é organizada como uma grade bi-dimensional. Todas as interconexões vão da primeira camada para a segunda. As duas camadas ficam inteiramente interconectadas, porque cada padrão de entrada é conectado a todas as unidades da camada competitiva.

Cada neurônio i é representado por um vetor de pesos sinápticos mi = [m_{i1}, ..., m_{iD}] onde i identifica a unidade na camada competitiva.

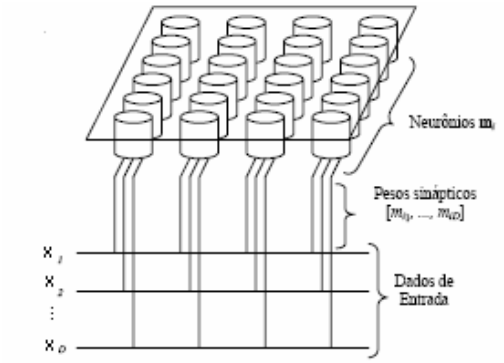


Figura 6. Estrutura do mapa de kohonen.

Uma rede SOM é definida por um conjunto de neurônios i, dispostos em um arranjo que define a vizinhança de cada neurônio, como pode ser visto na Figura 7 para as possibilidades mais usadas.

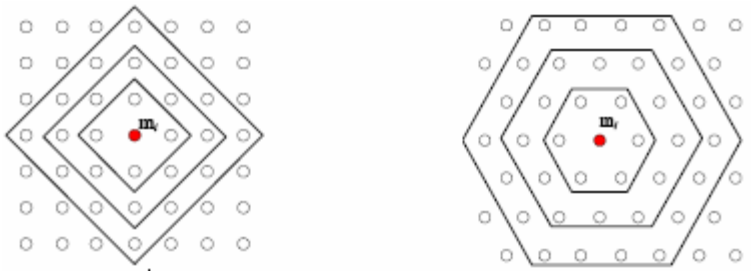


Figura 7. Relação de vizinhança.

Na Figura 7 vê-se no arranjo à esquerda uma vizinhança retangular, enquanto no arranjo à direita tem-se uma vizinhança hexagonal. Se os lados do mapa forem conectados, a forma do mapa tranforma-se num cilindro ou um toróide, vê a Figura 8. Os formatos cilíndrico e toroidal são pouco explorados. O formato padrão utilizado para visualização do mapa produzido é o plano retangular. Se o objetivo é obter uma visualização bidimensional fechada utiliza-se o formato cilíndrico ou se é desejada uma visualização tridimensional é sugerido o formato toroidal.

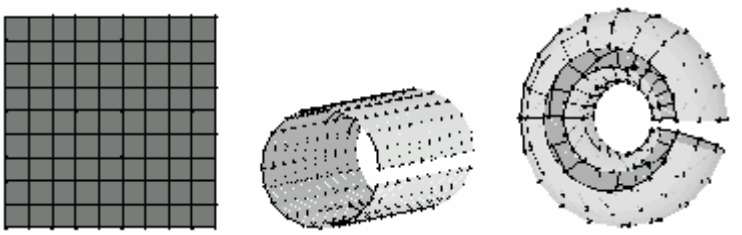


Figura 8. Formas diferentes de mapas: plano retangular à esquerda, cilindro no centro e toróide à direita.

3.2 Funcionamento do Algoritmo SOM

Para o presente trabalho entende-se como uma base de dados, um conjunto de m vetores com n componentes cada. Normalmente, para uso de técnicas de análise de grupos de dados, a quantidade m de vetores da base de dados ou o número n de componentes é elevado.
Os mapas auto-organizáveis possuem os seguintes elementos:

• Base de dados;

• Número fixo de vetores modelo(neurônios);

• Algoritmo de auto-organização;

• Elevado tempo de processamento.

Mapa: Os vetores modelos também conhecidos como pesos sinápticos, são simplesmente vetores com a mesma dimensão dos vetores que representam os dados de entrada. Tipicamente para os valores iniciais dos vetores modelos são estabelecidos valores aleatórios.

Dados: A base de dados é composta pelos vetores de dados a serem analisados e que serão fornecidos ao algoritmo para o aprendizado não supervisionado e para a organização do mapa.

Algoritmo: o algoritmo de auto-organização define a forma com que o mapa irá responder à entrada de dados externos. Os vetores mi(t) são rearranjados de acordo com os dados x(t) fornecidos conforme a Equação 1:

$$\mathbf{mi(t+1) = mi(t) + h(c(x),t) \,.\, (x(t) - mi(t))}$$

Equação 1. Reposicionamento dos vetores modelos.

onde:

- mi(t+1) representa o novo vetor modelo, já reposicionado de acordo com x(t);
- mi(t) é o vetor da iteração anterior, não está reposicionado;
- h(c(x), t) é a função vizinhança e aprendizagem e define a taxa de aprendizagem do mapa, com valores variando de 0 a 1;
- x(t) representa um dos vetores de entrada(padrão da base de dados de entrada).

O algoritmo de auto-organização do mapa opera basicamente em duas etapas:

- Competição – Uma dada amostra de dados x(t) é comparada a cada neurônio mi(t), esta comparação pode ser feita através da verificação da distância Euclidiana entre os vetores ou através de alguma expressão que determine logicamente a similaridade espacial entre x(t) e mi(t). A unidade no mapa mais similar, ou mais próxima, a x(t) é chamada de unidade vencedora.

- Cooperação – nesta etapa, o neurônio vencedor aproxima-se do vetor de dados x(t) conforme a Equação 1. Todos os neurônios vizinhos ao neurônio vencedor têm seus vetores ajustados para

uma aproximação de x(t) de acordo com a Equação 1. O fator h(c(x),t) diminui à medida que a distância de cada neurônio ao neurônio vencedor aumenta. Logo, os neurônios que estão mais próximos ao neurônio vencedor se ajustam mais rapidamente a x(t) e os neurônios que estão mais distantes ao neurônio vencedor sofrem poucas modificações.

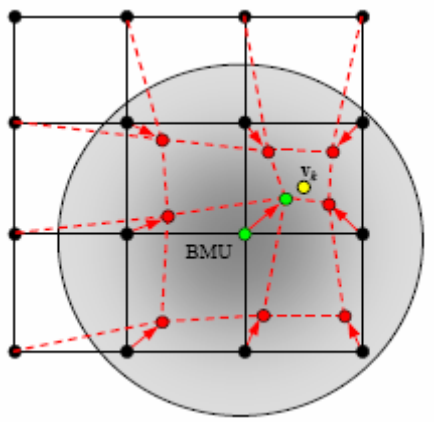


Figura 9. Adaptação dos neurônios próximos ao neurônio vencedor.

A Figura 9 mostra que quanto mais próximo um neurônio encontra-se do BMU, ou seja, quanto menor a distância ||Rc – Ri||, maior é a adaptação aplicada ao neurônio. O neurônio com maior adaptação é o BMU.

A função h(c(x),t) varia de acordo com:
• c(x) – representa a coordenada do neurônio vencedor;

• t – para cada instante t, tem-se um novo x(t) e novas comparações são feitas entre x(t) e os vetores modelos dos neurônios do mapa;

• σ(t) – representa a função que determina o raio da vizinhança, o valor decresce com o tempo;

• ∞(t) – é a taxa de aprendizagem que mostra a rapidez com o mapa se ajusta aos dados de entrada, o valor da taxa decresce com o tempo.

A função vizinhança pode ser expressa como uma função gaussiana, definida na Equação 2:

$$H(c(x),t) = \infty(t) * \exp(-||Rc-Ri|| / 2\ \sigma(t)\ ^2)$$

Equação 2. Função vizinhança.

Onde:
• Rc – representam as coordenadas do neurônio vencedor;
• Ri – representam as coordenadas de um vetor vizinho mi(t).

Quatro versões possíveis são descritas para a função vizinhança: Bubble, Gaussian, Cutgauss, Epanechnikov.

Bubble: Esta versão define uma largura ou um raio do neurônio vencedor, e somente os neurônios que estão no alcance deste raio são ajustados ao padrão de entrada.

Gaussian: É a versão mais usada. Uma importante propriedade desta função é que um neurônio que se encontra mais próximo do padrão de entrada sofrerá mais ajuste do que um neurônio que esteja mais afastado.

Cutgauss: Esta é uma combinação das duas versões acima. Se a distância de um dado neurônio ao neurônio vencedor estiver dentro de um valor dado(raio), o neurônio será atualizado.

Epanechnikov: Uma propriedade dessa versão é que a taxa de aprendizagem diminui mais enquanto a distância de cada neurônio ao padrão de entrada aumenta.

3.3 Visualização de dados com SOM

A ferramenta SOM apresenta grande potencial quando é usada na área de classificação de textos. Esta verificação geralmente é feita pela matriz-U tomando como base a forma em que os neurônios foram rearranjados.

O SOM comporta-se como uma grade contendo neurônios interconectados por conexões elásticas, responsáveis por dobrar, esticar ou comprimir o estado da grade, de forma a representar da melhor maneira possível o conjunto de dados de entrada. A idéia do algoritmo é realizar uma projeção dos dados de entrada na tentativa de preservar ao máximo a topologia original dos dados. Na Figura 10 é mostrada a idéia deste comportamento, onde a imagem à esquerda mostra-se a inicialização dos vetores de pesos da grade de neurônios e na imagem à direita observa-se a grade de neurônios já adaptada ao conjunto de dados de entrada. Na figura à esquerda os pontos com as cores vermelho, verde e azul representam o conjunto de entrada para a grade de neurônios. A superfície gerada mostra a deformação que a grade original sofre na tentativa de representar o conjunto de entrada, onde os neurônios da grade se comprimem em regiões de alta densidade dos dados de entrada(pontos azuis) e se esticam em regiões de baixa densidade(pontos verdes).

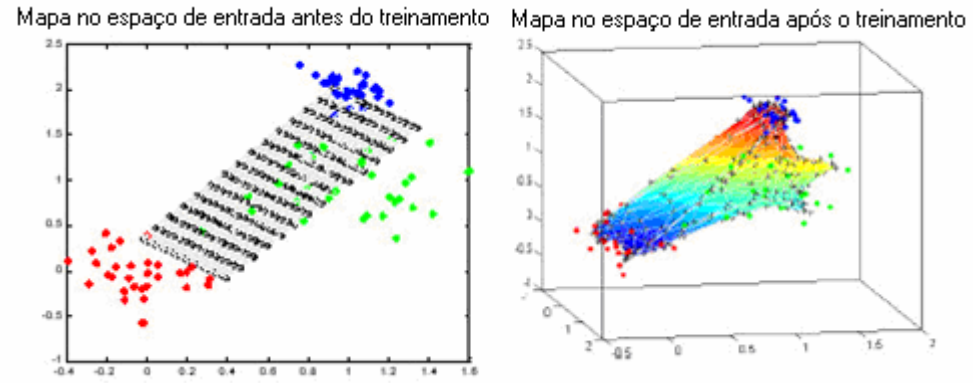


Figura 10. Capacidade de representação dos dados

Capítulo 4

Materiais e Métodos

O presente trabalho utilizou uma ferramenta muito usada para cálculos matemáticos chamada MATLAB, versão 5.2.0.3084, ver detalhes no endereço eletrônico: http://www.mathworks.com.
O *toolbox* escolhido do MATLAB foi o *somtoolbox*, que é um conjunto de rotinas especialmente para a criação e treinamento de mapas auto-organizáveis. Esta biblioteca do MATLAB apropriada para SOM está disponível gratuitamente no endereço: http://www.cis.hut.fi/projects/somtoolbox/download/.

Para se obter um mapa de documentos é preciso antes fazer um pré-processamento na coleção de documentos, na tentativa de remover palavras irrelevantes para categorização do texto e para se ter a informação da quantidade de palavras relevantes contidas nos documentos, porque essas palavras é que determinam a dimensão do vetor que representará o documento.

As seguintes subseções descrevem as fases de preparação e representação dos dados para utilização no experimento, e os processos de inicialização, treinamento e visualização do mapa de documentos. É descrito também as estratégias para a construção da interface de navegação.

4.1 Stopwords

Os documentos possuem palavras insignificantes ("*stopwords*") que não declaram as idéias do conteúdo do documento, e por isso estas palavras podem ser excluídas. São excluídas dos textos dos documentos palavras sem valor semântico ao texto como: os artigos, as preposições, as conjunções, símbolos, números e outros caracteres. É gerada uma lista com as palavras a serem descartadas, chamada de *stoplist.* Geralmente essas palavras aparecem freqüentemente, o que levaria a um erro muito grande na formação dos vetores pois esses teriam grandes valores atribuídos.

4.2 Stemming

A próxima etapa é realizar o ("*stemming*") que é o processo que remove sufixos e flexões para representar as palavras que tem grande valor semântico no texto por seus radicais, com o objetivo de aumentar a freqüência das palavras no vetor, reduzindo as diferenças entre derivações do mesmo termo.

4.3 Representação dos documentos

Os documentos que vão ser manipulados por um organizador automático de textos devem estar representados de maneira adequada. De acordo com [19], os documentos do conjunto de treinamento e de teste são representados conforme um modelo de representação de dados.

Quando se trata de classificação de textos, os documentos são normalmente representados por vetores de termos de indexação, onde para cada termo existente no vetor existe um peso associado [22].

A atribuição de peso aos termos dos vetores é uma maneira de diferenciar os termos mais relevantes dos termos que tem menor importância. Conforme [10], existem dois fatores importantes para a determinação do valor correspondente ao peso de um termo: a freqüência do termo em um documento e a quantidade de aparecimentos de termos na coleção.

4.3.1 Dicionário de características

O Dicionário ou vetor de características refere-se ao conjunto de palavras que representam maior relevância na coleção de documentos. As palavras que têm um grande aparecimento nos documentos estão contidas no dicionário. O agrupamento dessas palavras representará as categorias em que os documentos serão organizados. O dicionário de características pode apresentar uma dimensão muito elevada pelo fato de conter palavras relevantes que têm um alto índice de aparecimento na maioria dos documentos. Para que o tamanho do vetor de características não atinja um valor elevado, pode-se efetuar algumas operações na tentativa de redução do tamanho, como: remover do vetor de características as palavras que aparecem na lista de *stopwords*; definir um mínimo de freqüência que cada palavra deve aparecer para ser incluída no vetor de características; remover as palavras que são variações de outra palavra pela lista resultante de *stemming*.

4.3.2 Modelo vetorial

O modelo de espaço vetorial, ou simplesmente modelo vetorial , representa os documentos como vetores de termos e cada termo possui um valor associado que indica o grau de importância(peso) deste no documento. Em outras palavras, cada documento possui um vetor associado que é constituído por pares de elementos na forma {(palavra_1, peso_1), (palavra_2, peso_2), ..., (palavra_n, peso_n)}.

O peso de um termo em um documento pode ser calculado de várias maneiras. Os pesos são usados para computar a similaridade entre cada documento armazenado, geralmente o cálculo do peso baseia-se no número de ocorrências do termo no documento.

Com o vetor de características é gerada uma matriz onde as linhas representam os documentos e as colunas representam as palavras. Cada elemento da matriz é representado por seu TF-IDF ("*term frequency – inverse term frequency*": freqüência de cada palavra pelo inverso da freqüência do número de documentos em que a palavra aparece) mostrado na Equação 3:

$$w_{if} = \log(tf_{ij}) \, . \, \log(n / df_j)$$

Equação 3. peso de um termo i do documento.

Onde, w_{ij} representa o peso atribuído à palavra i do documento j, o termo tf_{ij} representa a freqüência do termo i no documento j, a variável *n* representa o número de documentos na coleção de documentos, e o termo df_j é a freqüência da palavra j na coleção de documentos.

Os documentos podem ser colocados em um espaço euclidiano de k dimensões (onde k é o número de termos) e a posição do documento em cada dimensão é dada pelo seu peso.

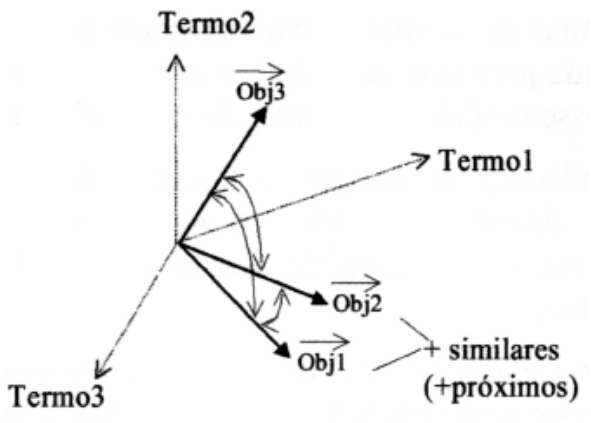


Figura 11. Similaridade dos documentos.

As distâncias entre um documento e outro indicam seu grau de similaridade, ou seja, documentos que possuem os mesmos termos acabam sendo colocados em uma mesma região do espaço e, em teoria, tratam de assuntos similares, como é mostrado na Figura 11.

4.4 Construção do Mapa de Documentos

Os vetores que representam os documentos são gerados a partir do vetor de características, ou seja, do conjunto de palavras que representam as categorias dos documentos. O método usado para criar os vetores de documentos baseia-se em mapear cada palavra do conjunto de categorias no vetor, indicando se a palavra existe no documento ou não. O resultado gerado é uma mapa de documentos, onde estes se encontram agrupados de acordo com a similaridade.

4.4.1 Conjunto de treinamento e conjunto de teste

Os classificadores automáticos de documentos trabalham com uma base de documentos textuais digitais previamente classificados, denominado de conjunto ou coleção. Estes documentos estão divididos em dois grupos, chamados de conjunto de treinamento e conjunto de teste. O conjunto de treinamento é utilizado pelo algoritmo de classificação para determinar as características das categorias dentro da coleção. Estas categorias encontradas são as mesmas para a classificação de novos documentos. O conjunto de teste serve para testar o desempenho do classificador. Os documentos de teste são analisados pelo classificador e são organizados conforme a categoria que

pertencem. A base de documentos utilizada como conjunto de treino e conjunto teste corresponde a um grupo de notícias *Reuters-21578 v1.0* [23] muito utilizado na literatura na área de classificação de texto. O conjunto de treinamento contém 7770 documentos, enquanto o conjunto de teste contém 3019 documentos.

O conjunto de categorias em que os documentos serão classificados contém 90 categorias, a Tabela 2 descreve as categorias, informando o identificador da categoria com sua respectiva descrição:

Tabela 2. Conjunto de categorias.

1	'acq'	31	'hog'	61	'platinum'
2	'alum'	32	'housing'	62	'potato'
3	'barley'	33	'income'	63	'propane'
4	'bop'	34	'instal-debt'	64	'rand'
5	'carcass'	35	'interest'	65	'rape-oil'
6	'castor-oil'	36	'ipi'	66	'rapeseed'
7	'cocoa'	37	'iron-steel'	67	'reserves'
8	'coconut'	38	'jet'	68	'retail'
9	'coconut-oil'	39	'jobs'	69	'rice'
10	'coffee'	40	'l-cattle'	70	'rubber'
11	'copper'	41	'lead'	71	'rye'
12	'copra-cake'	42	'lei'	72	'ship'
13	'corn'	43	'lin-oil'	73	'silver'
14	'cotton'	44	'livestock'	74	'sorghum'
15	'cotton-oil'	45	'lumber'	75	'soy-meal'
16	'cpi'	46	'meal-feed'	76	'soy-oil'
17	'cpu'	47	'money-fx'	77	'soybean'
18	'crude'	48	'money-supply'	78	'strategic-metal'
19	'dfl'	49	'naphtha'	79	'sugar'
20	'dlr'	50	'nat-gas'	80	'sun-meal'
21	'dmk'	51	'nickel'	81	'sun-oil'
22	'earn'	52	'nkr'	82	'sunseed'
23	'fuel'	53	'nzdlr'	83	'tea'
24	'gas'	54	'oat'	84	'tin'
25	'gnp'	55	'oilseed'	85	'trade'
26	'gold'	56	'orange'	86	'veg-oil'
27	'grain'	57	'palladium'	87	'wheat'
28	'groundnut'	58	'palm-oil'	88	'wpi'
29	'groundnut-oil'	59	'palmkernel'	89	'yen'
30	'heat'	60	'pet-chem'	90	'zinc'

A Tabela 3 exemplifica a representação das categorias de cada documento, onde para cada categoria existe a identificação se a categoria está contida no documento. Dado um documento, se

a categoria tiver valor (1) significa que ela pertence ao documento, no caso de ser (0) ela não pertence.

Tabela 3. Exemplo de representação das categorias de cada documento.

	Categoria_1	Categoria_2	...	Categoria_k
Docs	'acq'	'alum'	.	'zinc'
Doc_1	0	1	.	1
Doc_2	1	0	.	1
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
Doc_n	1	1		0

4.4.2 Inicialização e treinamento do mapa

Para a construção do mapa de documentos foi desenvolvida a função `Reuters_SOM` e nela foram definidos alguns parâmetros necessários:

- Dimensões do mapa a ser gerado;
  No trabalho a dimensão do mapa de documentos usada foi 30 x 30.

- Quantidade de épocas de treinamento;
  O número de épocas para a fase de ordenamento foi igual a 10 e para fase de refinamento foi igual a 20.

- O formato do mapa, que pode ser: plano retangular, cilindro ou toróide;

- Função de vizinhança, por exemplo: `cutgauss, gaussian, ep, bubble;`

- A métrica utilizada, que é usada para encontrar o neurônio vencedor (BMU), podendo ser:
  Distância euclidiana ou cosseno.

Algumas funções do *som_toolbox* sofreram otimizações que ajudaram na realização do objetivo deste trabalho. É bom deixar claro que foi utilizada uma rede SOM treinada com os parâmetros apresentados e que os valores dos parâmetros foram estabelecidos segundo heurísticas mostradas na tese de mestrado de Corrêa [4].
A função utilizada para criar e inicializar os vetores modelos do mapa foi a `ssom_randinit` que é uma adaptação da função `som_randinit` do *somtoolbox*.

Na fase de treinamento foi utilizada a função `ssom_batchtrainDP` que é uma adaptação da função `som_batchtrain do` *som_toolbox* que treina o mapa usando distância euclidiana. Depois do treinamento do mapa, foi feito o mapeamento dos documentos, para encontrar o neurônio vencedor para cada documento. A função `getBmus` que é uma adaptação da função

`som_bmus` do *som_toolbox* retorna o número dos neurônios vencedores. A Tabela 4 mostra a relação de alguns documentos mapeados e os seus BMUS. Cada linha na tabela representa o mapeamento de um determinado documento em um neurônio de um total de 900. Por exemplo, o documento de índice 2997 foi mapeado no neurônio 429 e o neurônio de número 98 ganhou o documento de índice 2998. Na fase de construção da interface foi preciso obter o conjunto de documentos mapeados em cada neurônio, para então criar as páginas correspondentes às unidades que contêm documentos.

Tabela 4. Mapeamento do conjunto de teste.

Documento	BMU
2997	429
2998	98
2999	178
3000	310
3001	17
3002	274
3003	864
3004	581
3005	581
3006	354
3007	307
3008	63
3009	258
3010	497
3011	68
3012	4
3013	183
3014	161
3015	896
3016	308
3017	118
3018	25
3019	18

4.4.3 Visualização do mapa de documentos

Existem vários métodos de visualização de mapas auto-organizáveis. No *somtoolbox*, uma ferramenta básica para visualização é a função `som_show`.Ela pode ser usada para mostrar a matriz de distância unificada ou matriz-U e também superfícies planas do SOM.

Na Figura 12 é mostrada uma visualização do mapa de documentos de dimensão 30 x 30 com rótulos referentes às categorias que os documentos do conjunto de treinamento pertencem. Cada neurônio tem a forma hexagonal e tem um vetor modelo associado. Depois do treinamento os neurônios vizinhos apresentam vetores modelos similares.

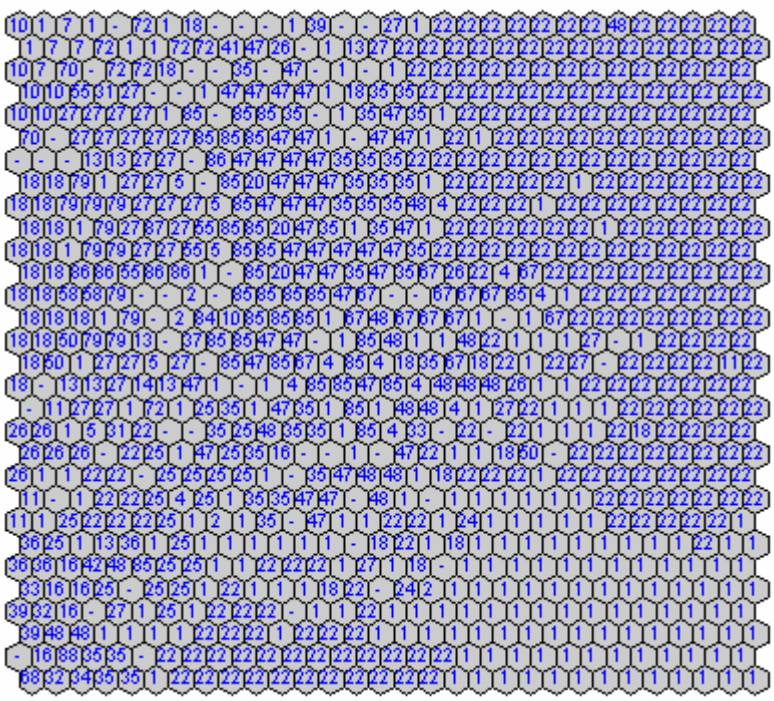


Figura 12. SOM com rótulos das categorias

É mostrada na Figura 13 uma representação da superfície da matriz-U do SOM 30 x 30 de documentos. As elevações representam a separação entre dois agrupamentos e as áreas uniformes representam a existência de um agrupamento. Embora a resposta neural confirme a separação dos documentos em categorias, a matriz-U não mostra com clareza a separação dos agrupamentos de documentos.

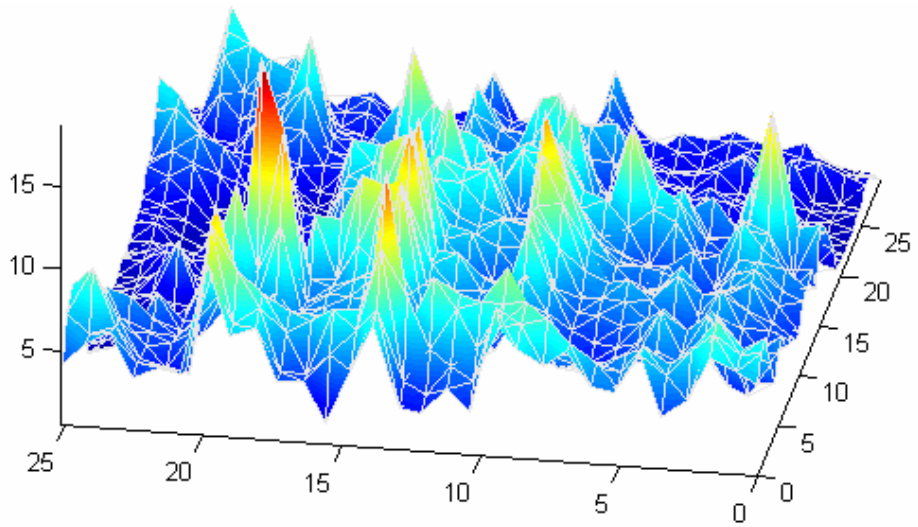


Figura 13. Matriz-U do SOM de documentos

Outro modo de visualização é mostrado na Figura 14, o mapa de documentos está com cores na escala de cinza. Os neurônios que têm vetores modelos similares estão com tons de cinza próximos.

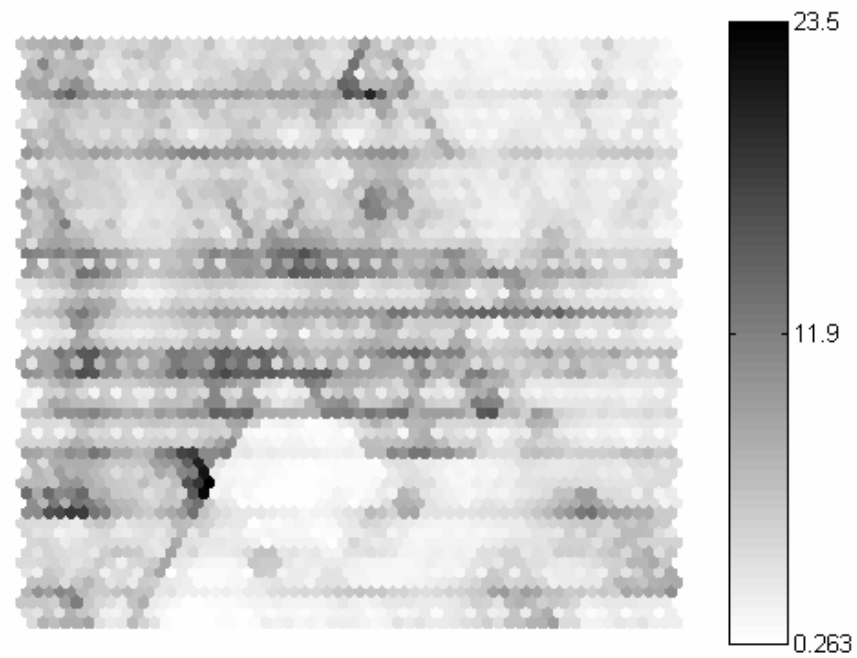


Figura 14. SOM de documentos em escala de cinza

4.4.4 Construção da interface

O sistema proposto deve entender a estrutura de um mapa auto-organizável treinado e gerar sua interface, para isto, serão implementados métodos de representação gráfica e rotulação dos mapas de documentos que permitirão de uma forma automática apresentar as categorias onde os documentos estão contidos. Esta representação gráfica ficará interligada a páginas a serem implementadas com tecnologia *web*, permitindo a visualização e exploração dos documentos dentro de cada região.

A linguagem escolhida para implementação da interface foi HTML. HTML é uma linguagem utilizada para criação de páginas que podem ser lidas em qualquer tipo de computador e transmitidas pela *internet*. A linguagem permite fazer ligações com outros documentos, possibilitando uma navegação virtual. A sigla HTML vem do inglês: *Hiper Text Mark-up Language,* e pode ser traduzida por Linguagem de Marcação de Hiper Texto. É importante lembrar que hipertexto denomina documentos que podem conter todo o tipo de informação:textos, fotos, animações, trechos de sons e vídeos.

Primeiramente foram criadas as páginas correspondentes a cada unidade no mapa. Cada página contém ligações para os documentos ganhos por cada neurônio. Para encontrar os documentos mapeados em cada neurônio foi preciso encontrar para cada neurônio vencedor os

índices dos documentos mapeados. Cada índice representando a ordem do documento foi relacionado com a identificação real do documento.

A função `som_interface` foi desenvolvida para construção da interface. Nesta função encontra-se o código para encontrar os documentos que estarão em cada página referente a cada unidade da interface.

A Tabela 5 mostra a relação entre os índices dos documentos mapeados nos BMUS e as identificações reais dos documentos.

Tabela 5. Identificações reais dos documentos

índice do documento	ID do documento
3	6
10	18
15	27
20	38
100	153
630	1207

O segundo passo é encontrar a categoria predominante em cada unidade da interface. Com os documentos encontrados em cada neurônio, foi gerado um histograma contendo as categorias relacionadas a todos os documentos de cada neurônio. A partir do histograma, encontra-se a categoria majoritária que representará cada unidade da interface.

Tabela 6. Histograma de categorias.

Categoria	Freqüência
10	15
55	1
60	1
70	1
72	1
83	1

A Tabela 6 mostra a relação dos identificadores das categorias encontradas para um determinado neurônio (unidade) com suas respectivas freqüências nos documentos. A categoria com maior freqüência representará a unidade na interface.

Em seguida obteve-se o conjunto com todas as categorias que são visíveis no mapa de documentos, a Figura 15 mostra os identificadores das categorias mapeadas do conjunto de treinamento.

```
Columns 1 through 12

   1     2     4     5     7     8    10    11    13    14    16    18

Columns 13 through 24

  20    22    23    24    25    26    27    31    32    33    34    35

Columns 25 through 36

  36    37    39    44    47    48    50    55    56    58    67    68

Columns 37 through 44

  70    72    79    84    85    86    87    88
```

Figura 15. Conjunto de categorias visíveis no SOM

Tabela 7. Descrição das categorias mapeadas no SOM.
É mostrada a identificação da categoria com o respectivo nome.

1	acq
2	alum
4	bop
5	carcass
7	cocoa
8	coconut
10	coffee
11	copper
13	corn
14	cotton
16	cpi
18	crude
20	dlr
22	earn
23	fuel
24	gas
25	gnp
26	gold
27	grain
31	hog
32	housing
33	income

34	instal-debt
35	interest
36	ipi
37	iron-steel
39	jobs
44	livestock
47	money-fx
48	money-supply
50	nat-gas
55	oilseed
56	orange
58	palm-oil
67	reserves
68	retail
70	rubber
72	ship
79	sugar
84	tin
85	trade
86	veg-oil
87	wheat
88	wpi

Capítulo 5

Resultados

A interface construída manualmente representa um mapa de documentos de dimensão 30 x 30, onde cada célula corresponde a um neurônio contendo uma lista de documentos mapeados. Cada célula está rotulada com a categoria predominante dos documentos contidos.

Esta implementação visual traz alguns benefícios como:

- ✓ Um resumo da coleção explicitando os tópicos presentes nos grupos de documentos;
- ✓ Permite a navegação interativa, pode-se navegar pelo mapa, verificando os documentos mapeados em cada neurônio;
- ✓ Fornece um mecanismo visual que facilita a navegação sobre o mapa de documentos, através de meios de seleção dos documentos mais interessantes.

A Figura 16 mostra a interface representada por uma tabela 30 x 30, correspondendo a 900 células. Cada unidade está rotulada com o identificador da categoria e o nome da unidade.

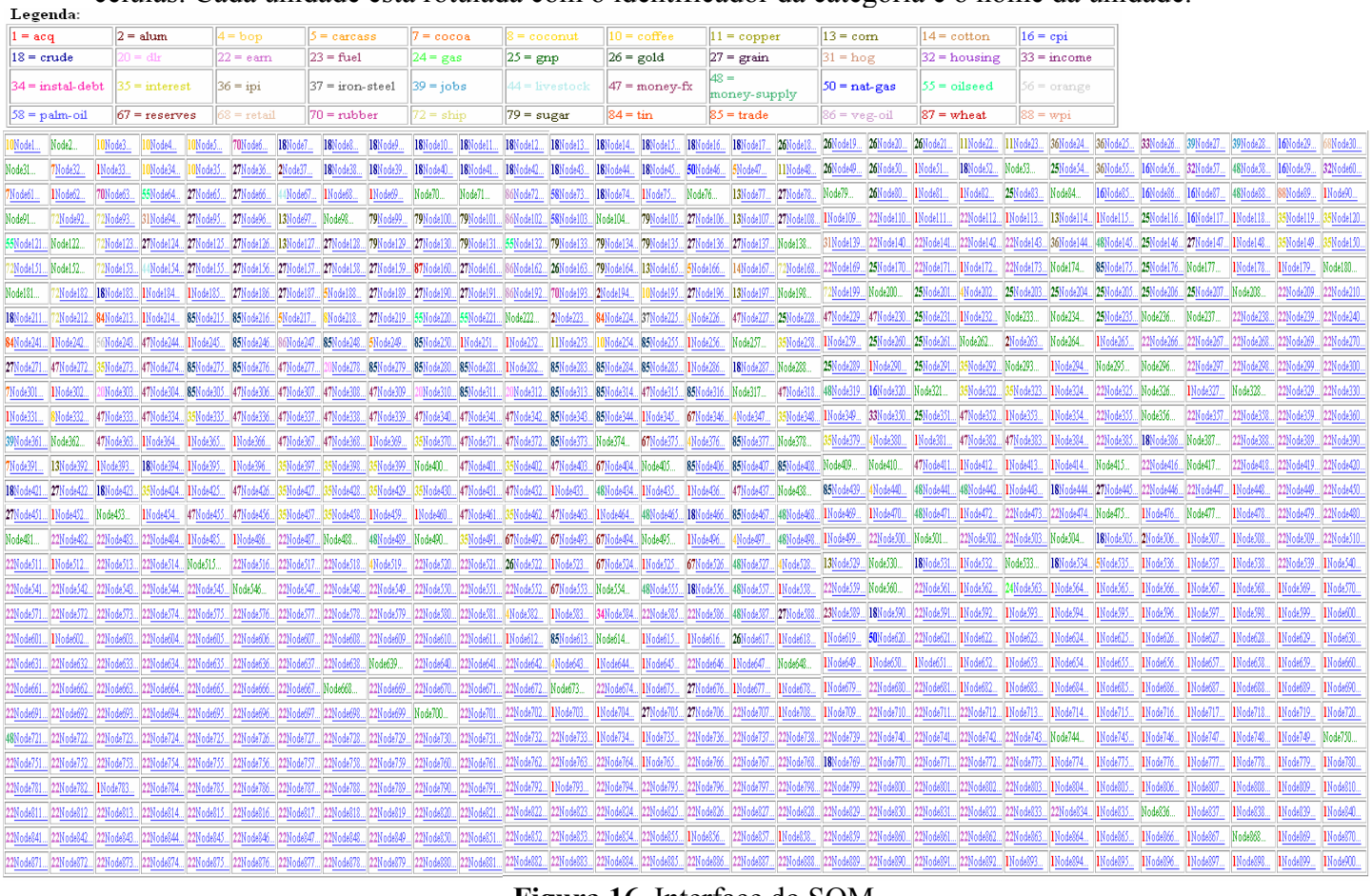


Figura 16. Interface do SOM

A disposição das categorias no mapa de documentos apresenta-se diferente da disposição mostrada na interface HTML, pelo fato de que o algoritmo implementado para exibição das categorias no mapa de documentos difere do modo como foi implementado para interface, mas o resultado exibido na interface está compatível com o resultado apresentado no mapa de documentos. Na Figura 12 que representa o mapa de documentos com as categorias rotuladas encontra-se uma grande concentração da categoria '22' na parte superior à direita do mapa e na Figura 16 representando a interface a concentração da categoria '22' encontra-se na parte inferior à esquerda da interface. Essa diferença na arrumação das categorias não alterou o resultado para efeito de classificação dos documentos. O conjunto de categorias que representam os documentos é o mesmo para as duas figuras.

10Node4...	10Node5...	70Node6...	18Node7...	18Node8...	18Node9...	18Node10...
10Node34...	10Node35...	27Node36...	2Node37...	18Node38...	18Node39...	18Node40...
55Node64...	27Node65...	27Node66...	44Node67...	1Node68...	1Node69...	Node70...
31Node94...	27Node95...	27Node96...	13Node97...	Node98...	79Node99...	79Node100...
27Node124...	27Node125...	27Node126...	13Node127...	27Node128...	79Node129...	27Node130...
44Node154...	27Node155...	27Node156...	27Node157...	27Node158...	27Node159...	87Node160...
1Node184...	1Node185...	27Node186...	27Node187...	5Node188...	27Node189...	27Node190...
1Node214...	85Node215...	85Node216...	5Node217...	8Node218...	27Node219...	55Node220...
47Node244...	1Node245...	85Node246...	86Node247...	85Node248...	5Node249...	85Node250...
47Node274...	85Node275...	85Node276...	47Node277...	20Node278...	85Node279...	85Node280...
47Node304...	85Node305...	47Node306...	47Node307...	47Node308...	47Node309...	20Node310...
47Node334...	35Node335...	47Node336...	47Node337...	47Node338...	47Node339...	47Node340...
1Node364...	1Node365...	1Node366...	47Node367...	47Node368...	1Node369...	35Node370...
18Node394...	1Node395...	1Node396...	35Node397...	35Node398...	35Node399...	Node400...
35Node424...	1Node425...	47Node426...	35Node427...	35Node428...	35Node429...	35Node430...
1Node454...	47Node455...	47Node456...	35Node457...	35Node458...	1Node459...	1Node460...
22Node484...	1Node485...	1Node486...	22Node487...	Node488...	48Node489...	Node490...
22Node514...	Node515...	22Node516...	22Node517...	22Node518...	4Node519...	22Node520...
22Node544...	22Node545...	Node546...	22Node547...	22Node548...	22Node549...	22Node550...

Figura 17. Visualização detalhada da interface para o conjunto de treinamento

É apresentada uma legenda que contém o nome de todas as categorias predominantes nas células da interface. Cada categoria está destacada com uma cor para facilitar a navegação pelo mapa de documentos. A atribuição de cores às categorias ocorreu de forma manual, onde apartir de um de um conjunto pré-definido de cores foi escolhida uma cor para enfatizar uma determinada categoria. A Figura 18 representa a legenda com as categorias obtidas a partir do mapeamento da base de dados.

Legenda:

1 = acq	2 = alum	4 = bop	5 = carcass	7 = cocoa	8 = coconut	10 = coffee	11 = copper	13 = corn	14 = cotton	16 = cpi
18 = crude	20 = dlr	22 = earn	23 = fuel	24 = gas	25 = gnp	26 = gold	27 = grain	31 = hog	32 = housing	33 = income
34 = instal-debt	35 = interest	36 = ipi	37 = iron-steel	39 = jobs	44 = livestock	47 = money-fx	48 = money-supply	50 = nat-gas	55 = oilseed	56 = orange
58 = palm-oil	67 = reserves	68 = retail	70 = rubber	72 = ship	79 = sugar	84 = tin	85 = trade	86 = veg-oil	87 = wheat	88 = wpi

Figura 18. Descrição das categorias

Quando uma célula que contiver documentos for clicada, o browser fornecerá uma página com um conjunto de ligações que representarão os documentos residentes na unidade da interface, indicando o número da unidade e também a quantidade de documentos contida. A Figura 19 mostra uma página com referências para documentos.

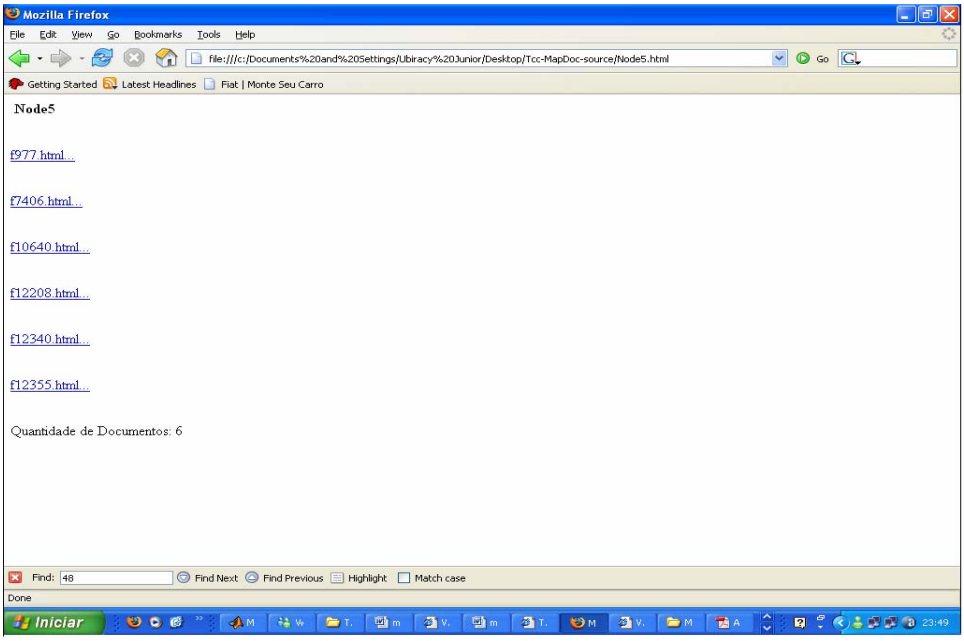


Figura 19. Página contendo referências para documentos

Ao clicar em alguma referência para o documento, é exibida outra página contendo o texto presente no documento. Os documentos contidos na mesma categoria são similares em relação aos contextos. A seguir são mostrados dois documentos localizados numa mesma unidade da interface.

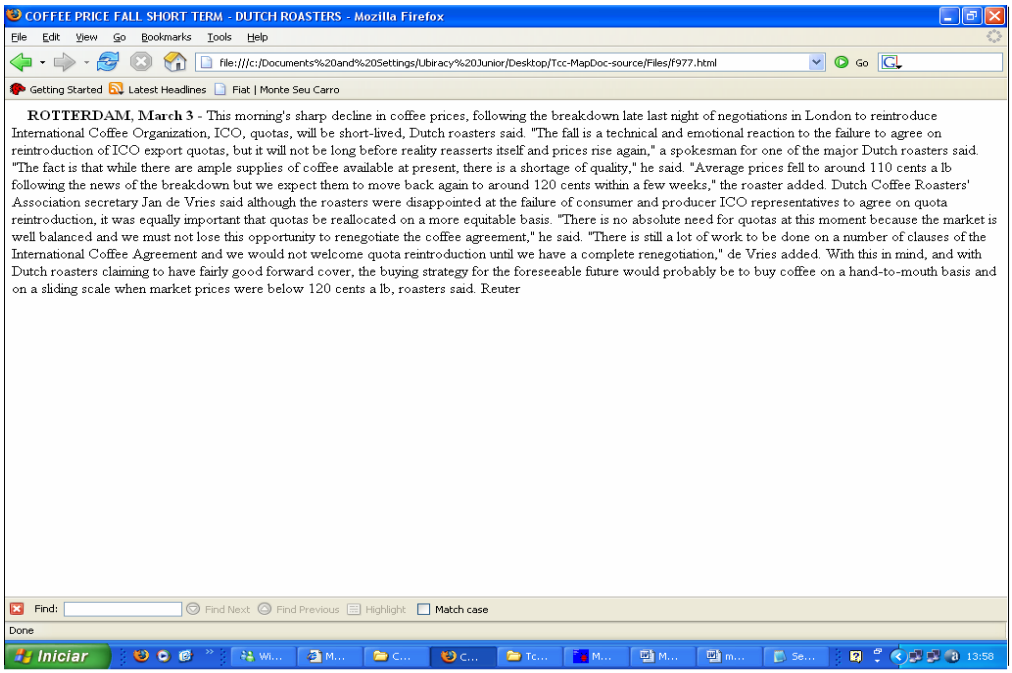


Figura 20. Documento visualizado

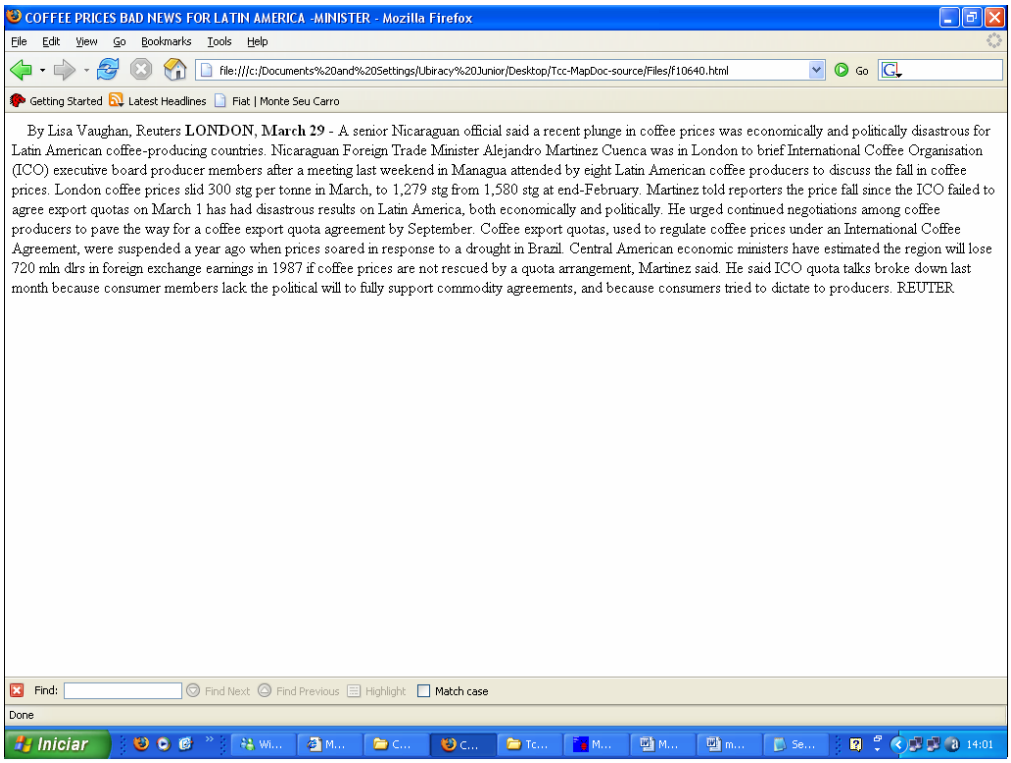


Figura 21. O documento apresenta semelhança textual com o documento exibido na Figura 20.

Observa-se uma grande semelhança entre os textos contidos nos dois documentos exibidos acima. Foram encontradas palavras comuns e relevantes nos documentos, como: *coffe*, *export*, *quotas*, *prices*, *negotiation* e *consumer*, destacando a proximidade textual entre os documentos da mesma unidade.

O conjunto de treinamento com 7770 documentos ao ser mapeado gerou um conjunto com 44 categorias. Das 900 unidades que representam o mapa, 826 contêm documentos.
Mapeando o conjunto de teste que contém 3019 documentos, foi gerada uma interface em que 653 unidades contêm documentos.

É mostrado na Tabela 8 um comparativo entre os dados do conjunto de treinamento e do conjunto de teste mapeados na interface. Estão relacionadas algumas categorias com a respectiva quantidade de documentos encontrada.

Tabela 8. Quantidade de documentos do conjunto de treino e teste para cada categoria.

Categoria	Treino	Teste
1 - acq	1847	940
2 - alum	31	26
4 - bop	65	21
5 - carcass	45	6
7 - cocoa	49	17
8 - coconut	8	3
10 - coffee	99	2
18 - crude	368	1009

As unidades na interface que não contém documentos estão destacadas com a cor verde e não estão ativas para uma ligação com as páginas que contém as referências para os documentos. A seguir na Figura 22 é mostrada a interface gerada para o conjunto de teste, das 900 unidades da interface 247 não estão ativas.

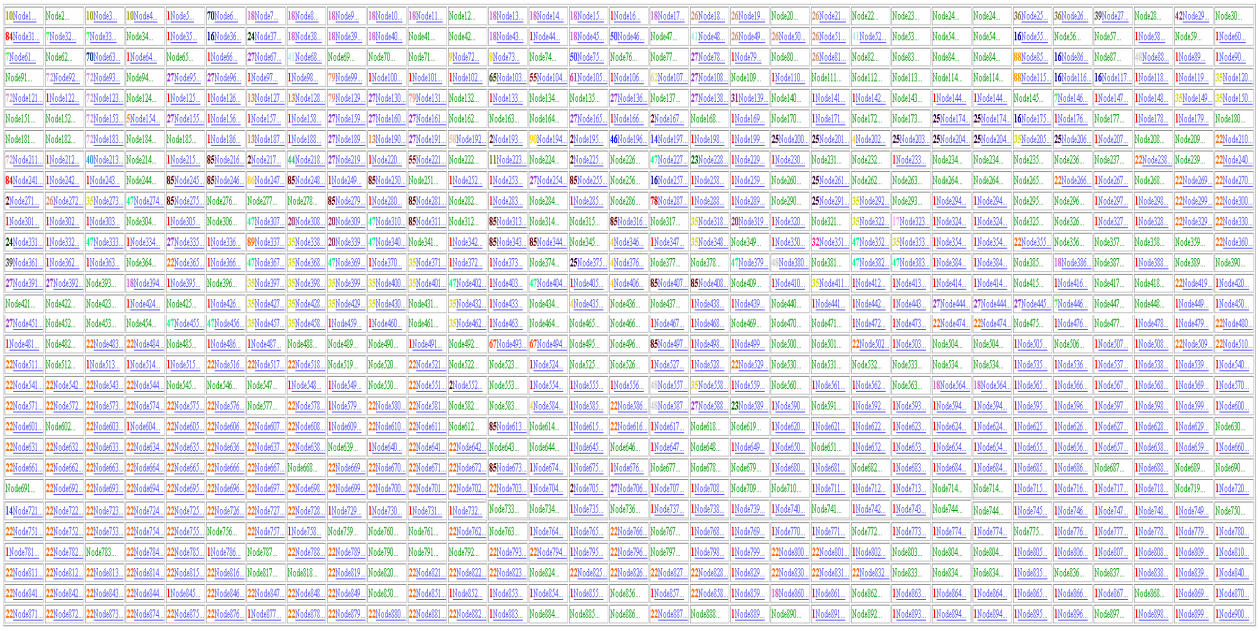


Figura 22. Interface mapeada com o conjunto de teste

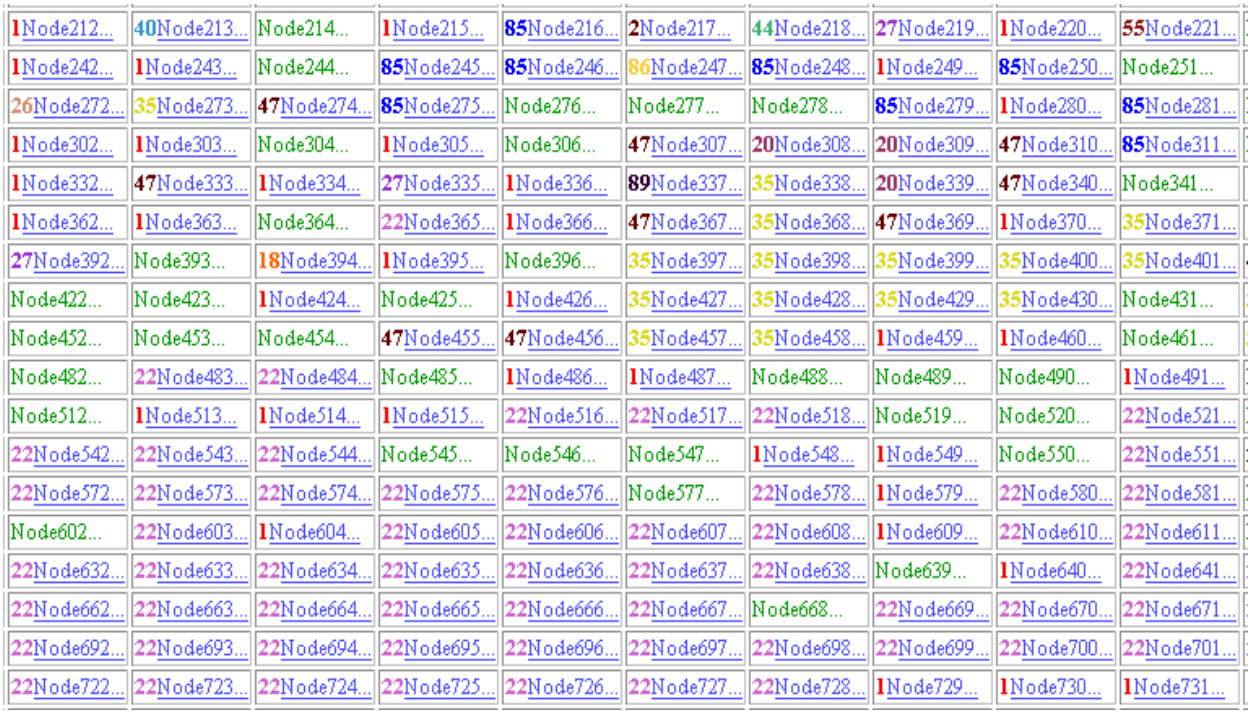


Figura 23. Visão ampliada da interface mapeada com o conjunto de teste

Embora as unidades visualizadas no mapa de documentos apresentem um formato hexagonal, as unidades da interface apresentam formato retangular porque não é possível criar formatos diferentes somente utilizando a linguagem HTML.

Pode ser observado que no mapeamento do conjunto de treino as unidades referentes às categorias ‘ 27 - grain’, ’13 - corn’, ’10 - coffee’, ’85 - trade’, 79 - sugar’ agrupam-se próximas mostrando semelhança textual entre os documentos. Além disso, as categorias ‘72 – ship’ e ’87 - wheat’ se encontrarem próximas realçando a proximidade textual.

Uma das vantagens da ferramenta produzida é a capacidade de organizar documentos que tenham similaridade textual em regiões próximas. Outra vantagem é o atrativo visual relacionado às cores dos rótulos das categorias, que vem facilitar a navegação na busca de informações. Uma desvantagem é quando se deseja obter uma mapa com grande dimensão, resultando numa interface muito ampla. E quanto maior o números de unidades na interface maior é o número de páginas com as referências para os documentos a serem geradas, necessitando de um espaço físico maior para o armazenamento dos arquivos.

Capítulo 6

Conclusões

O trabalho apresentado explora o uso de categorias para a classificação de documentos textuais. Toda a pesquisa realizada para o desenvolvimento deste trabalho de conclusão de curso contribuiu para um enriquecimento de informações na área de sistema de recuperação de informação.

De modo geral, o trabalho mostrou a utilidade e a funcionalidade de uma ferramenta computacional, utilizando SOM para categorização dos textos. É importante deixar claro que este trabalho utilizou uma base de dados reais para avaliação da ferramenta produzida.

A área de sistema de recuperação de informação tem um grande campo de pesquisa para ser explorado, como por exemplo, na busca de melhorar o desempenho no quesito classificação dos documentos, como também na forma de descobrir novos métodos de classificação e otimização dos algoritmos já existentes. Fica claro neste trabalho o uso do método SOM, bem como suas características, fases para o desenvolvimento do mapa de documentos e de ilustrações, na tentativa facilitar o entendimento e mostrar que o método tem capacidade de organizar documentos textuais.

6.1 Contribuições

As principais contribuições deste trabalho foram:

- Apresentação de uma ferramenta que gera uma interface gráfica de navegação sobre mapa de documentos;
- Conhecimento e aplicação do método SOM;
- Aplicação prática da linguagem HTML para a construção da interface do mapa de documentos.

6.2 Trabalhos Futuros

Promove-se como possíveis trabalhos futuros:

- Um estudo mais detalhado de metodologias que utilizam a tecnologia de redes neurais auto-organizáveis para fins de recuperação de informação;
- Otimização de algoritmos de treinamento das ferramentas existentes na área;
- Um estudo mais aprofundado de tecnologias *web* para implementação de interface de mapa de documentos.

Bibliografia

[1] AZCARRAGA, A. and YAP JR., T. SOM-Based Methodology for Building Large Text Archives. 7th Intl Conference on Database Systems for Advanced Applications,DASFAAOl. Hong Kong. April 18-20, pages 66-73. 2001.

[2] BAEZA-YATES, R.; RIBIERO-NETO, B. Modern Information Retrieval. 1. ed. New York: Addison-Wesley, 1998.

[3] CHEN,H., SCHUFFELS, C. and ORWIG, R. Internet categorization and search: a machine learning approach. Journal of Visual Communications and Image Representation, v. 7, n. 1, pp. 88-102, 1996.

[4] CORRÊA, R. F., Categorização de Documentos Utilizando Redes Neurais: Análise Comparativa com Técnicas Não-Conexionistas. . Dissertação de Mestrado. Centro de Informática da UFPE, Recife, 2002.

[5] CORRÊA, R. F., Redes Neurais Auto-Organizáveis aplicadas à Recuperação de Informação em Coleções de Documentos Texto. Plano de Pesquisa de Doutorado. Centro de Informática da UFPE, Recife, 2004.

[6] DITTENBACH, M, MERKL, D. & RAUBER, A.: The growing hierarchical self-organizing map. In Proceedingsof the International Joint Conference on Neural Networks. (IJCNN 2000), Como, Italy 6, (2000) 15-19.

[7] HAYKIN, S. “Neural Networks A Comprehensive Foundation”, Prentice Hall,1994.

[8] HONKELA, T.; KASKI, S.; LAGUS, K.; KOHONEN, T. Exploration of fulltext databases with self-organizing maps. Proceedings of ICNN'96, IEEE International Conference on Artificial Neural Networks, 1996. Piscataway, NJ: IEEE Service Center, 1996. v. I, p. 56-61.

[9] HUBEL, D., WIESEL, T. Receptives fields, binocular interaction and functional architecture in the cat’s visual cortex. J. Physiol., v. 160, p. 106–154, 1962.

[10] JURAFSKY, D. e MARTIN, J. (2000). Speech and Language Processing – An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. – New Jersey: Prentice Hall.

[11] KOHONEN, T.; KASKI, S.; LAGUS, K.; SALOJÃRVI, J.; HONKELA, J.; PAATERO, V.; SAARELA, A. Self Organization of a Massive Document Collection. IEEE Transaction on Neural Networks, v. 11, n. 3, pages 574-585. May 2000.

[12] KOHONEN, T., Self-Organizing Maps.2nd ed. Berlim, Alemanha: Springer, 1997.

[13] KOHONEN, T. Self-organization of very large document collections: State of the art. Proc. ICANN98, the 8th Int. Conf. on Artificial Neural Networks, Londres, 1998. v. 1, p. 65-74.

[14] KOHONEN, T. “Self-Oganized Formation of Topologically Correct Feature Maps”. Biological Cybernetics 43, pg 59-69, 1982.

[15] KOHONEN, T. “Self-Organizing Maps”. Series in Information Sciences, vol. 30, 2nd edition. Springer-Verlag, Heidelberg, 1997.

[16] LAGUS, K., "Text Mining with the SOM". Acta Polytechnica Scandinavica, Mathematics and Computing Series n.º 110. Dr. Tech Thesis, Helsinki University of Technology, Finlândia, 2000.

[17] LAGUS, K., KASKI, S. Keyword selection method for characterizing text document maps.

In Proc. ICANN99, Ninth Int. Conf. Artificial Neural Networks, vol.1, Londres, 1999. pp. 371-376.

[18] LIN, X.; SOERGEL, D.; MARCHIONINI, G. A self-organizing semantic map for information retrieval. Proceedings of the Fourteenth Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, 1991, Chicago, IL. p. 262-269.

[19] MANNING, C. e SCHÜTZE, H. (1999). Foundations of Statistical Natural Language Processing. Cambridge: The MIT Press.

[20] ORWIG, R.; CHEN, H.; NUNAMAKER, J. F. A graphical, self-organizing approach to classifying electronic meeting output. Journal of the American Society for Information Science, v. 48, n. 2, p.157-170, February 1997.

[21] ROUSSINOV, DMITRI & CHEN, HSINCHUN. A Scalable Self-organizing Map Algorithm for Textual Classification: A Neural Network Approach to Thesaurus Generation. Communication and Cognition in Artificial Intelligence Journal (CC-AI), v. 15, n. 1-2, p. 81-111, 1998.

[22] SEBASTIANI, F. A Tutorial on Automated Text Categorization. In Proceedings of ASAI-99, 1st Argentinian Symposium on Artificial Intelligence. 1999.

[23] SEBASTIANI, F. Machine learning in automated text categorization. ACM Computing Surveys (CSUR) Volume 34 , Issue 1, March 2002. p.1-47.