Classificação Hierárquica de Documentos Textuais Digitais usando o Algoritmo *k*NN

Leonardo Cavalheiro Langie, Vera Lúcia Strube de Lima

Programa de Pós-Graduação em Ciência da Computação
Faculdade de Informática
Pontifícia Universidade Católica do Rio Grande do Sul
Av. Ipiranga, 6681 – 90619-900 – Porto Alegre – RS

{llangie, vera}@inf.pucrs.br

Abstract. *This paper describes an ongoing work that focuses the development of a method aiming at textual document classification into subject hierarchies. These hierarchies privilege the document organization allowing a decomposition of the classification problem into subproblems. These problems are treated by specialized classifiers that deal with a small set of information. In this paper, we describe the proposed method and details about its implementation.*

Resumo. *Este artigo descreve um trabalho em andamento para a construção de um método de classificação de documentos textuais digitais em categorias de assuntos organizadas em estruturas hierárquicas. A utilização da hierarquia facilita a organização dos documentos e permite que o problema de classificação seja decomposto em subproblemas específicos. Esses problemas são tratados por classificadores especializados (classificadores locais) que lidam com um conjunto mais reduzido de informações. Neste artigo apresentamos a abordagem utilizada para a classificação hierárquica, bem como detalhes de sua implementação.*

1. Introdução

Com a difusão do uso do computador, é comum a utilização de documentos textuais digitais para armazenamento de informações. Empresas, centros de pesquisas e órgãos governamentais manipulam grandes quantidades de documentos textuais digitais. Esses documentos devem ser organizados e gerenciados para que suas informações possam ser utilizadas. Um dos grandes desafios com o qual nos deparamos está na forma como esses documentos poderão ser organizados de modo a facilitar o acesso a informações relevantes.

Uma forma de organizar documentos se dá por meio do uso de sistemas de classificação, nos quais os documentos são organizados de acordo com o tópico ou assunto de que tratam. Talvez os mais tradicionais sistemas desse tipo sejam aqueles utilizados por bibliotecários na catalogação de livros. Considerando documentos

textuais digitais, temos como grandes exemplos de sistemas de classificação os motores de busca da Internet como o Yahoo![1].

Um dos grandes problemas de tais sistemas reside na abordagem inicial utilizada, que é a classificação manual – processo pelo qual um indivíduo lê um documento, identifica seu assunto e classifica-o em uma ou mais categorias de assuntos previamente existentes. Considerando a quantidade e a rapidez com que novos documentos são produzidos, a classificação manual caracteriza-se por ser um processo lento e oneroso.

Nesse contexto, existe a necessidade de empregar técnicas automáticas ou semi-automáticas, que auxiliem o ser humano no processo de classificação. A Classificação (ou Categorização) Automática de Textos (CT) consiste em atribuir categorias de assuntos pré-definidas a novos documentos, a partir de um conjunto de documentos de treino [Yang e Liu, 1999]. Os documentos de treino são documentos previamente classificados, ou seja, documentos cujas categorias foram identificadas, geralmente, por um processo de classificação manual. Eles são usados pelo classificador para identificar as características das categorias existentes.

Uma forma de melhorar a organização dos documentos textuais é fazer uso da estruturação das categorias de assuntos em hierarquias. Essas estruturas facilitam a organização dos documentos ao permitirem o estabelecimento de relações entre assuntos mais genéricos e mais específicos. Além disso, a própria estrutura de assuntos pode ser utilizada para navegação por entre os documentos, facilitando o gerenciamento e o acesso às informações. Com a utilização de uma estrutura hierárquica de categorias, o processo de classificação pode ser decomposto em subprocessos menores, nos quais a quantidade de variáveis envolvidas é reduzida. Conforme Koller e Sahami, em [Koller, 1997], categorias que se encontram próximas, dentro da estrutura hierárquica, possuem, em geral, mais características em comum do que outras categorias. Por exemplo, o termo "jogador" pode não ser um bom atributo para diferenciar documentos da categoria "Futebol" de documentos da categoria "Vôlei". Porém, este mesmo termo pode ser um bom atributo para diferenciar documentos da categoria "Esporte" dos documentos pertencentes à categoria "Agricultura". A Classificação Hierárquica de Textos (CHT) consiste em atribuir categorias de assuntos, pré-definidas e organizadas em uma estrutura hierárquica, a novos documentos. Assim como a Classificação de Automática de Textos, a CHT, geralmente, utiliza um conjunto de documentos de treino a partir do qual o classificador identifica as características de uma categoria.

Neste artigo, apresentamos um algoritmo para classificação automática de documentos textuais digitais em uma estrutura de categorias organizadas em forma de árvore. A abordagem de classificação utilizada é denominada *top-down level-based*, conforme [Sun e Lim, 2001]. Nessa abordagem, a classificação hierárquica é o resultado da execução de um ou mais classificadores não-hierárquicos em determinados nodos da árvore de categorias. Utilizamos, como classificador não-hierárquico, um algoritmo do tipo *k-Nearest Neighbor* (*k*NN).

Este artigo se encontra organizado em 6 seções. A Seção 2 reporta trabalhos relacionados. Na Seção 3 é apresentado um embasamento conceitual sobre classificação

[1] http://www.yahoo.com/

automática de textos. Na Seção 4 são apresentados o método de classificação hierárquica e detalhes do classificador local. Na Seção 5 é apresentado um experimento realizado. Na Seção 6 tecemos considerações e discutimos trabalhos futuros.

2. Trabalhos Relacionados

Existem, na literatura, diferentes métodos para classificação automática de textos em categorias não estruturadas. Estas categorias são tratadas separadamente, sem uma estrutura que defina seus relacionamentos, e são conhecidas como *flat categories* [Sun e Lim, 2001]. Denominamos os classificadores que trabalham com tais categorias de classificadores não-hierárquicos. A classificação hierárquica de textos é uma área de pesquisa mais recente, sendo que, atualmente, observa-se um grande interesse na classificação de documentos da *Web* [Sun, Lim e Ng, 2002, Frommholz, 2001, Dumais e Chen, 2000]. Nesse tipo de classificação, características mais particulares (típicas de documentos HTML) como *tags* e *links*, são muitas vezes consideradas.

Dumais e Chen [Dumais e Chen, 2000] apresentam um trabalho em classificação hierárquica de documentos da *Web* que emprega a abordagem *top-down level-based* e classificadores não-hierárquicos do tipo *Support Vector Machine* (SVM). A idéia do trabalho dos autores é classificar resultados de buscas (obtidos por motores de busca da *Web*) em uma estrutura hierárquica de categorias. A estrutura utilizada é uma árvore de categorias com exatamente dois níveis, e os resultados de buscas são classificados apenas nas folhas dessa árvore.

Os classificadores não-hierárquicos são criados para cada categoria da árvore, e atribuem um escore ao documento sendo classificado (escore de classificação). Este escore é comparado a um valor limiar para decidir se o documento pertence ou não à categoria. Duas abordagens distintas para classificar os documentos são testadas por Dumais e Chen. Na primeira abordagem, denominada *sequential boolean*, um documento só é testado em uma categoria c_i do segundo nível se ele for classificado como pertencente à categoria pai de c_i. Na segunda abordagem, denominada *multiplicative*, um documento é testado em todas as categorias de ambos os níveis. Consideram-se, então, os escores de classificação do documento para categorias do primeiro e segundo nível. Estes escores são multiplicados e, caso o resultado seja superior a um valor limiar, o documento é classificado como pertencente à categoria do segundo nível.

Os experimentos realizados por Dumais e Chen utilizam uma árvore de categorias com dois níveis. No primeiro nível existem 13 categorias e no segundo nível 150 categorias. Eles utilizam um corpus com 60.102 documentos provenientes do diretório LookSmart[2]. Estes documentos são gerados a partir dos pequenos sumários retornados pelo motor de busca e são representados por vetores binários que indicam a ocorrência dos termos.

Sun e Lim [Sun e Lim, 2001] apresentam um trabalho em classificação hierárquica de textos que também emprega a abordagem *top-down level-based* e classificadores não-hierárquicos do tipo *Support Vector Machine* (SVM). A

[2] http://www.looksmart.com

classificação é realizada com o uso de uma árvore de categorias, permitindo que os documentos sejam classificados tanto nas folhas quanto nos nodos intermediários.

No trabalho de Sun e Lim são utilizados dois tipos de classificadores não-hierárquicos: classificadores locais e classificadores de subárvore. Os classificadores locais são criados e executados para cada categoria da árvore. O objetivo de um classificador local executado em uma categoria c_i é determinar se um documento pertence ou não a c_i. Os classificadores de subárvore são construídos e executados para cada categoria interna da árvore, e são responsáveis por identificar se um documento deve ou não ser enviado para os classificadores de suas subcategorias.

Os experimentos realizados por Sun e Lim utilizam três diferentes árvores de categorias. Cada uma das árvores possui dois níveis. Os documentos são representados por vetores binários e um pré-processamento é realizado para *stemming* e remoção de *stopwords*.

3. Princípios da Classificação Automática de Textos

Nesta seção abordamos alguns conceitos e procedimentos comuns a diferentes métodos de classificação de textos, sejam eles hierárquicos ou não.

3.1. Base de Treino e Base de Teste

Os classificadores automáticos trabalham com um conjunto de documentos textuais digitais previamente classificados, denominado coleção ou corpus. Estes documentos são divididos em dois conjuntos, denominados base de treino e base de teste.

A base de treino é utilizada pelo algoritmo de classificação para identificar as características das categorias da coleção. Estas categorias são as mesmas nas quais novos documentos poderão ser classificados. A base de teste é utilizada para testar o desempenho do classificador. Os documentos de teste são analisados pelo classificador, que determina a(s) categoria(s) à(s) qual(is) o documento pertence. A análise do desempenho é realizada comparando-se o resultado fornecido pelo algoritmo com a classificação original (manualmente realizada) dos documentos.

3.2. Representação dos documentos

Os documentos a serem manipulados por um classificador de textos devem estar representados de forma adequada. Conforme [Manning e Schütze, 1999], os documentos das bases de treino e de teste são representados de acordo com um *modelo de representação de dados*. Em se tratando de classificação de textos, os documentos são geralmente representados por *vetores de termos de indexação* nos quais, para cada termo do documento, existe um peso associado [Sebastiani, 1999].

Atribuir peso aos termos de um documento é uma forma de diferenciar os termos mais relevantes daqueles termos de menor importância. Conforme [Jurafsky e Martin, 2000], existem dois fatores importantes para a determinação do peso de um termo: a freqüência do termo em um documento e a distribuição de termos na coleção. Salton e Buckley [Salton e Buckley, 1988] ressaltam a utilidade de um terceiro fator, que permite representar os documentos por vetores de mesmo comprimento. Este fator é importante, pois documentos pequenos tendem a ser representados por vetores pequenos.

3.3. Seleção de Atributos

Muitos termos não são suficientemente significativos para descrever o assunto de um texto, sendo que alguns termos "carregam" mais significado do que outros [Meadow et al., 2000]. Considerar todos os termos de um documento para a geração de sua representação pode prejudicar o desempenho do classificador e, também, apresentar-se computacionalmente inviável.

A seleção de atributos consiste em eliminar termos que não são representativos, ou então combinar mais de um termo em um único atributo. A seleção também serve para diminuir o número de elementos que compõem os vetores dos documentos.

4. Implementação da Classificação Hierárquica de Textos

O método de classificação hierárquica implementado utiliza uma abordagem baseada na *top-down level-based* e em classificadores não-hierárquicos do tipo *k-Nearest Neighbor* (*k*NN). Em nossa versão da abordagem *top-down level-based*, classificadores *k*NN são criados e executados apenas nas categorias intermediárias da árvore, e são denominados classificadores locais.

A idéia do processo de classificação hierárquica é iniciar a classificação de um documento pela categoria mais genérica, desdobrando-se para as categorias mais específicas sempre que possível. À medida que a especificidade das categorias aumenta, classificadores locais mais específicos são utilizados. Esses classificadores lidam com um vocabulário mais restrito e, possivelmente, um número menor de atributos.

O início do processo de classificação hierárquica se dá com a criação e execução de um classificador local para a raiz da árvore. A raiz da árvore não é uma categoria na qual os documentos podem ser classificados, ela apenas representa o nodo mais genérico, pai das categorias do primeiro nível, conforme apresentado na Figura 1.

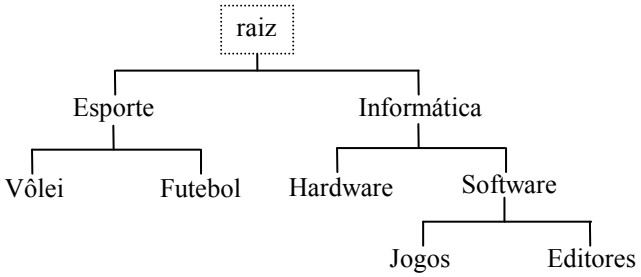

Figura 1. Exemplo de árvore de categorias

Os classificadores locais funcionam como um filtro, decidindo o caminho que o documento sendo classificado percorrerá, na árvore de categorias. A função de um classificador local, executado em uma categoria intermediária c_i, é classificar um documento d com relação às categorias filhas de c_i.

O processo de classificação hierárquica é considerado seletivo, pois permite que determinadas subárvores da árvore de categorias sejam desconsideradas do processo. Isso acontece ao identificar-se que o documento sendo classificado não pertence a uma determinada categoria intermediária da árvore. Nesse caso, considera-se que, se o documento não pertence a uma categoria, então ele também não pertence as suas subcategorias. Por exemplo, considerando a árvore de categorias da Figura 1, e um

documento *d* a ser classificado: caso o resultado do classificador da raiz seja que *d* pertence à categoria "Informática", o processo de classificação continuará apenas para a subárvore da categoria "Informática", desconsiderando a subárvore de "Esporte".

O processo de classificação termina quando um documento é classificado em uma categoria-folha, a partir da qual o processo não pode mais ser desdobrado. Outra condição de parada ocorre quando um classificador local, executado em uma categoria c_i, não consegue classificar um documento nas categorias-filhas de c_i (veja detalhes na subseção 4.1). Nesse caso, o classificador local identificou que o processo não deve ser desdobrado para as categorias mais específicas e, portanto, deve ser finalizado.

4.1. Implementação do Classificador Local

Os classificadores locais constituem-se nos principais elementos do processo de classificação hierárquica implementado. De fato, o resultado da execução de diversos classificadores locais corresponde ao próprio processo de classificação hierárquica.

Um classificador local é uma classe desenvolvida em Java, implementando um algoritmo do tipo *k-Nearest Neighbor* (*k*NN). Esta classe foi desenvolvida adaptando a classe IBk, proveniente do pacote de algoritmos de *machine learning* WEKA[3] descrito em [Witten e Frank, 2000]. O princípio de funcionamento do algoritmo *k*NN é classificar um documento *d* de acordo com os *k* documentos da base de treino mais próximos a *d* (os "vizinhos" de *d*). O classificador implementado permite que o valor de *k* seja definido pelo usuário sendo que, atualmente, utilizamos o valor 7.

Os classificadores locais são criados e executados em categorias intermediárias da árvore, classificando um documento em exatamente uma categoria. Um classificador local, executado em uma categoria intermediária c_i, tem a responsabilidade de classificar um documento *d* com relação às categorias-filhas de c_i. Para isso, ele trabalha com uma base de treino composta por documentos de c_i. Para permitir que os documentos também sejam classificados em categorias intermediárias, um classificador local executado em c_i pode classificar um documento na própria categoria c_i. Isto acontece quando o documento sendo classificado não pertence às categorias-filhas de c_i. Esta característica constitui-se em uma das condições de parada do processo de classificação hierárquica.

Os classificadores locais são ditos específicos, pois lidam com um número reduzido de categorias, o que lhes permite trabalhar com um vocabulário mais restrito e, possivelmente, um número menor de atributos. Um classificador local criado na categoria intermediária c_i utiliza uma base de treino composta por documentos de c_i e de suas subcategorias.

O funcionamento de um classificador local pode ser dividido em três etapas: análise dos documentos, cálculo e seleção de categorias. Considerando um documento *d* a ser classificado, as etapas do classificador funcionam da seguinte maneira: na etapa de análise dos documentos o classificador local busca informações da sua base de treino, gera a representação para *d* e identifica os documentos da base de treino mais próximos a *d* (os documentos "vizinhos"). Na etapa de cálculo, obtém-se, para cada um dos

[3] http://www.cs.waikato.ac.nz/ml/weka/

documentos "vizinhos", um valor de similaridade relativo a *d*. Este valor é calculado usando o coeficiente do co-seno – uma medida que pode ser usada para calcular similaridade entre dois documentos representados por vetores [Manning e Schütze, 1999, Jurafsky e Martin, 2000]. Com base nesses valores, gera-se a tabela de semelhança, que identifica a semelhança entre *d* e cada uma de suas possíveis categorias. A seleção das categorias é a etapa final da execução de um classificador local, sendo responsável por identificar a categoria de *d*, a partir da tabela de semelhança. Atualmente seleciona-se a categoria com maior valor de semelhança. Estão previstas alterações nesta etapa, que possibilitem classificar um documento em mais de uma categoria da tabela de semelhança.

5. Experimento

Nesta seção é descrito um experimento realizado com o algoritmo de classificação hierárquica. Primeiramente são descritas as condições do experimento, a seguir são apresentadas medidas de avaliação utilizadas e, por fim, os resultados obtidos.

5.1. Condições do experimento

Em nosso experimento utilizamos uma coleção de documentos provenientes de um corpus composto por artigos do jornal Folha de São Paulo do ano de 1994[4]. A partir desse corpus foram selecionados e classificados manualmente 701 documentos em 13 categorias, conforme a Figura 2. Em média, um documento da coleção possui 241 palavras e pertence a 1,6 categorias.

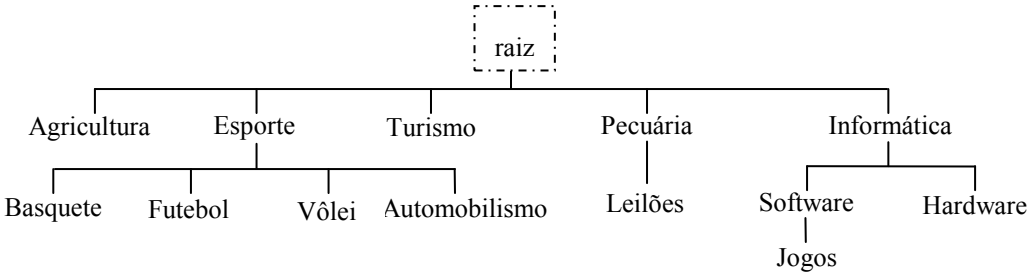

Figura 2. Árvore de categorias usada no experimento

Cada documento foi classificado em uma única subárvore da hierarquia. Assim, não existe um documento classificado ao mesmo tempo nas categorias "Informática", "Hardware" e "Software", pois estas categorias compõem duas subárvores diferentes. Essa abordagem foi empregada pois os classificadores locais ainda não são capazes de classificar um documento em mais de uma categoria da tabela de semelhança (veja subseção 4.1). Deve-se salientar que essa abordagem pode ter influência nos resultados, superestimando o desempenho do classificador, porém não invalida o experimento realizado.

Os documentos são separados em base de treino e teste de forma aleatória, com 415 documentos compondo a base de treino e 286 compondo a base de teste. Cada documento está normalizado, verticalizado e armazenado em um arquivo individual. Os documentos são representados por vetores de termos de indexação, sendo que os termos

[4] Este corpus foi cedido pelo NILC (Núcleo Interinstitucional de Lingüística Computacional), ao grupo de pesquisa em PLN da PUCRS.

são monopalavra. A geração da representação de um documento utiliza o texto integralmente, ignorando apenas as etiquetas de categorias. O peso na composição dos vetores é a combinação de fatores conhecida como TFC [Salton e Buckley, 1988]. Nessa combinação, termos mais importantes recebem peso próximo a 1 e termos menos importantes recebem valores próximos a zero.

Antes de terem suas representações geradas, os documentos passam por uma etapa de *pré-processamento*. Nesta etapa os termos dos documentos são colocados em minúsculas, e caracteres de pontuação e dígitos são eliminados. Além disso, aplica-se uma seleção de atributos com a remoção de *stopwords* (artigos, advérbios, conjunções, numerais, preposições, pronomes, e verbos de ligação). A lista de *stopwords* contém 365 termos. Atributos também são selecionados com a remoção de termos cuja *freqüência do documento* (FD) é inferior a 3. A FD de um termo *t* corresponde ao número de documentos nos quais *t* ocorre pelo menos uma vez. Conforme [Yang e Pederson, 1997], o princípio da seleção de atributos usando FD é que termos raros não são informativos para predizer a categoria de um documento e também não influenciam no desempenho do classificador.

5.2 Medidas de avaliação

Utilizamos quatro medidas de avaliação em nossos experimentos: precisão, *recall*, *micro-average* e *macro-average*. Estas medidas são comumente utilizadas para avaliação de métodos de classificação [Sebastiani, 1999, Yang, 1999, Sun e Lim, 2001].

Conforme [Yang, 1999, Sun e Lim, 2001], a precisão (pr_i) para uma categoria c_i mede o percentual de documentos classificados corretamente em c_i dentre todos os documentos classificados em c_i. *Recall* (re_i) mede o percentual de documentos classificados corretamente em c_i dentre todos os documentos que deveriam ser classificados em c_i. Estas duas medidas fornecem uma avaliação para categorias individuais. Para avaliar o desempenho com relação ao conjunto de todas as categorias utilizam-se as medidas *micro-average* e *macro-average*. Estas medidas fornecem o desempenho médio do classificador, baseado nas medidas de precisão e *recall*. A medida *macro-average* fornece uma média na qual as categorias são tratadas com igual importância, enquanto que a medida *micro-average* fornece uma média na qual os documentos (e não as categorias) são tratados com igual importância [Sun e Lim, 2001]. Conforme [Yang, 1999] é importante fornecer estas duas medidas, visto que a *micro-average* é mais influenciada pelo desempenho do classificador em categorias com muitos documentos, enquanto que a medida *macro-average* é mais influenciada pelo desempenho do classificador em categorias com poucos documentos.

5.3. Resultados

Os resultados do experimento realizado são apresentados na Tabela 1. Foram coletadas medidas de precisão e *recall* para cada categoria. Estas medidas foram utilizadas para calcular as médias *micro-average* e *macro-average* relativas à precisão e ao *recall*. O tempo médio de classificação de um documento foi de 0,45 segundos, executando-se o método de classificação, implementado em Java, em um computador Pentium 4, 1.5GHz, 128MB de memória, com plataforma Microsoft Windows 2000. O algoritmo *k*NN não necessita de treinamento, trabalhando diretamente com os vetores dos documentos de treino e com o vetor do documento a ser classificado. Os vetores dos

documentos de treino podem ser gerados *off-line*. A geração de todos os vetores de treino e seleção de atributos foi realizada em 50 segundos.

O algoritmo teve um bom desempenho na coleção de documentos testada (veja Tabela 1). Porém, não é possível generalizar esses resultados para outras coleções. Deve-se levar em conta, por exemplo, a quantidade de documentos testados e a distribuição de documentos por categorias (veja Tabela 1). Também deve ser levado em conta o fato de que cada documento da coleção foi classificado em uma única subárvore da hierarquia. Isto não necessariamente corresponde ao comportamento dos documentos em outras coleções ou até mesmo na *Web*. Outra consideração que pode ser feita, diz respeito à normalização dos documentos, que resulta na redução no número de atributos e pode influenciar no desempenho do classificador.

Tabela 1: Resultados do experimento e distribuição de documentos por categoria

Categoria	Documentos de treino	Documentos de teste	Precisão	*Recall*
Agricultura	32	22	1,00	0,95
Pecuária	32	22	0,95	0,95
Leilões	9	6	0,67	0, 67
Esporte	141	98	1,00	1,00
Basquete	12	8	0,89	1,00
Automobilismo	20	14	1,00	1,00
Futebol	91	62	0,95	0,98
Vôlei	9	7	0,86	0,86
Informática	132	91	0,99	0,99
Hardware	28	19	0,71	0,79
Software	52	36	0,79	0,75
Jogos	12	9	0,78	0,78
Turismo	78	53	0,96	0,98
Macro-Average	-	-	0,89	0,90
Micro-Average	-	-	0,94	0,95

6. Conclusões e Trabalhos Futuros

O trabalho apresentado neste artigo explora o uso de categorias organizadas em estrutura do tipo árvore na classificação de documentos textuais digitais em português. O processo de classificação é incremental e seletivo, sendo guiado pela organização hierárquica das categorias.

A utilização de uma estrutura hierárquica de categorias tem como objetivo facilitar a organização dos documentos e permitir que o processo de classificação seja decomposto e realizado por classificadores mais específicos. Estes classificadores lidam com um conjunto menor de categorias, um vocabulário mais restrito e, possivelmente, um número menor de atributos.

Embora os experimentos não possam ser comparados diretamente com outros, os resultados coletados são interessantes, encorajando a continuidade das pesquisas e refinamento do método. Como trabalhos futuros destacam-se: permitir que os classificadores locais classifiquem um documento em mais de uma categoria da tabela de semelhança; realizar experimentos para comparar a classificação hierárquica e a classificação não-hierárquica; realizar experimentos em uma coleção formada por um número maior de documentos organizados em uma árvore com maior número de

categorias; realizar experimentos que utilizem medidas de avaliação que considerem a hierarquia de categorias. As medidas utilizadas consideram as categorias de forma independente. Nesse caso, não existe diferença ao se classificar, de forma errada, um documento de "Pecuária" como "Agricultura" ou como "Hardware".

Referências

Dumais, S. e Chen, H. (2000). Hierarchical Classification of Web Content. In *Proceedings of SIGIR-00, ACM International Conference on Research and Development in Information Retrieval*.

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.

Frommholz, I. (2001). Categorizing Web Documents in Hierarchical Catalogues. In *Proceedings of ECIR-01, European Colloquium on Information Retrieval Research*.

Koller, D. e Sahami, M. (1997). Hierarchically classifying documents using very few words. In *Proceedings of ICML-97, 14th International Conference on Machine Learning*.

Manning, C. e Schütze, H. (1999). Foundations of Statistical Natural Language Processing. Cambridge: The MIT Press.

Meadow, C., Boyce, B. e Kraft, D. (2000). Text Information Retrieval Systems. San Diego: Academic Press.

Salton, G. e Buckley, C. (1988). Term-Weighting Approaches in Automatic Text Retrieval. In *Information Processing e Management*, Vol. 24 n. 5, pp. 513-523.

Sebastiani, F. (1999). A Tutorial on Automated Text Categorization. In *Proceedings of ASAI-99, 1st Argentinian Symposium on Artificial Intelligence*.

Sun, A. e Lim, E. (2001). Hierarchical Text Classification and Evaluation. In *Proceedings of the 2001 IEEE International Conference on Data Mining*.

Sun, A., Lim, E. e Ng, W. (2002) Web Classification Using Support Vector Machine. In *Proceedings of WIDM'2002*, *ACM Fourth International Workshop on Web Information and Data Management*.

Witten, I. e Frank, E. (2000). WEKA: Machine Learning Algorithms in Java. In *Data Mining: Pratical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann Publishers.

Yang, Y. e Pedersen, J. (1997). A Comparative Study on Feature Selection on Text Categorization. In *Proceedings of ICML-97, 14th International Conference on Machine Learning*.

Yang, Y. e Liu, X. (1999). A re-examination of text categorization methods. In *Proceedings of SIGIR-99, ACM International Conference on Research and Development in Information Retrieval*.

Yang, Y. (1999). An Evaluation of Statistical Approaches to Text Categorization. In *Journal of Information Retrieval*, Vol 1, n.1/2, pp. 67-88.