Classificação Automática de Documentos usando Subespaços Aleatórios e Conjuntos de Classificadores

Chu Chia Gean *Celso A. A. Kaestner*

Pontifícia Universidade Católica do Paraná (PUCPR)
Programa de Pós-Graduação em Informática Aplicada (PPGIA)
Rua Imaculada Conceição, 1155 – CEP 80.215-901
Curitiba – Paraná – BRASIL
{ccg, kaestner}@ppgia.pucpr.br

Resumo. Atualmente, devido ao volume grande de texto disponível em meios digitais, a classificação automática de documentos se torna uma tarefa importante da área do Tratamento Automatizado de Informações. Neste artigo descreve-se uma nova abordagem para o problema, baseada no modelo vetorial para o tratamento de textos e no uso de técnicas de Reconhecimento de Padrões. Como coleções de textos produzem espaços vetoriais de dimensão bastante elevada, o problema é tratado usando várias técnicas de pré-processamento e um conjunto de classificadores baseados em instâncias – do tipo *k*-vizinhos mais próximos, cada um dos quais dedicado a um subespaço do espaço original. A classificação final é obtida por uma combinação de resultados dos classificadores individuais. Esta abordagem foi aplicada a documentos oriundos das bases de dados TIPSTER e REUTERS, amplamente utilizadas na área. São apresentados os principais resultados obtidos e algumas conclusões e perspectivas do trabalho.

Abstract. Nowadays, due to the large volume of text available in digital media, the automatic document categorization becomes an important modern Information Retrieval task. In this paper we describe a new approach to the problem, based on the classical vector space model for text treatment and on the use of Pattern Recognition techniques. As texts collections produce huge dimensional vector spaces, we attack the problem using several preprocessing techniques, and a set of *k*-Nearest-Neighbors classifiers, each of them dedicated to a sub-space of the original space. The final classification is obtained by a combination of the results of the individual classifiers. We apply our approach to documents extracted from the TIPSTER and REUTERS databases. The obtained results and some conclusions are presented.

Palavras-chave: Recuperação de Informações, Classificação Automática de Documentos, Aprendizagem de Máquina Baseada em Instâncias. Subespaços Aleatórios. Conjuntos de Classificadores.

Keywords: Information Retrieval, Automatic Text Categorization, Instance-Based Machine Learning, Random Subspaces, Multiple classifiers.

Submissão ao "X Congreso Argentino de Ciencias de la Computación"

1- Introdução

A explosão de informações é uma realidade. Um recente estudo de Berkeley [Lyman 03] indicou que em 2002 havia 5 milhões de terabytes de informações novas criadas por impressões, filmes, mídias de armazenamento óticas e magnéticas. Estes números são aproximadamente duas vezes dos dados gerados em 1999, determinando uma taxa de crescimento de cerca de 30% ao ano. A quantidade de informações disponíveis na *Web* sozinha corresponde a 17 vezes o tamanho da Biblioteca do Congresso americano

Por outro lado é muito difícil de usar estas informações. Problemas como a busca e recuperação, a extração automática de informações, a sumarização automática de textos e a classificação automática de textos se tornaram uma área importante de pesquisa em Ciência da Computação. O uso das ferramentas de tratamento automático de informação se tornou essencial ao usuário comum, pois sem elas é praticamente impossível o pleno uso do potencial informativo do *Web* [Zhong 02].

Em particular, a tarefa da classificação de documento é aplicável a um grande número de situações práticas, como distribuição automática de e-mails, organização automática de documentos legados, filtração de textos, etc [Belkin 92], [Dhillon 01], [Sebastiani 02].

Neste artigo propõe-se uma nova abordagem na realização desta tarefa: utilizam-se um conjunto de classificadores baseados em instâncias (*k* vizinhos mais próximos ou *k*-NN), cada um deles aplicado a um subespaço do espaço vetorial original dos documentos. A classificação final é obtida pela combinação dos resultados individuais dos classificadores aplicados em cada subespaço. A abordagem foi testada em documentos obtidos de duas coleções: a coleção TIPSTER [Trec 04] e a coleção REUTERS [Lewis 04], ambas amplamente empregadas na área de Recuperação de Informações.

O restante deste trabalho é organizado da seguinte forma: na seção 2 apresenta-se uma visão geral do modelo vetorial para representação de documentos, e a descrição das etapas de pré-processamento e da tarefa de classificação; na seção 3, detalha-se o formalismo matemático subjacente à proposta; na seção 4 descreve-se a metodologia utilizada nos experimentos e apresentam-se os resultados obtidos; finalmente, na seção 5 apresentam-se algumas conclusões e as perspectivas deste trabalho.

2- O modelo vetorial e classificação de documentos

O propósito principal de um modelo de representação é obter uma descrição semântica adequada de um documento, em uma forma que habilite o cumprimento correto da tarefa alvo, conforme as necessidades do usuário.

Foram propostos vários modelos formais para textos, como o modelo booleano [Wartik 92], o modelo probabilista [vanRijsberger 92] e o modelo vetorial [Salton 97]. Neste trabalho, utiliza-se basicamente o modelo vetorial. Neste modelo, a unidade básica para processar textos é chamada de *termo*, que pode corresponder a uma palavra, um radical ou uma subseqüência da palavra original, conforme indicado a seguir. Em acordo com o modelo vetorial, cada documento é representado como um vetor em um espaço *n*-dimensional onde *n* é o número de termos considerados em toda a coleção. O valor da coordenada correspondente a um termo no vetor que representa um documento é normalmente obtido como uma função da freqüência deste termo no documento e na coleção.

Pré-processamento

Na etapa de pré-processamento os documentos considerados em texto puro, sem qualquer formatação, são tratados para produzir uma representação que seja mais adequada à execução da tarefa desejada [Sparck-Jones 97]. Este procedimento também é chamado de *indexação* [Sebastiani 02], e tipicamente envolve algumas das etapas seguintes:

1) Eliminação das *stopwords*: *stopwords* são termos do texto que não têm significado semântico; sua presença em um documento não agrega informação adicional relevante sobre o assunto ou conteúdo do documento. As *stopwords* normalmente são os termos mais freqüentes na coleção, constituídos de verbos auxiliares, artigos, preposições, etc. como: "*is*", "*or*" "*the*", ou "*in*".

2) Obtenção de Radicais (*stemming*): em uma língua natural, os elementos que designam plurais, conjugações de verbos, ou outras variações são normalmente sintaticamente semelhantes à palavra original, mantendo com esta um *radical* (ou *stem*) comum. A obtenção dos radicais dos elementos do texto é um processo que tem como objetivo obter associações entre os diversos elementos de texto como mesma semântica. Por exemplo, variações como "*working*" e "*works*" são consideradas originárias do mesmo radical "*work*".

3) Representação em *n-grams*: é uma representação alternativa do texto, proposta em [Cavnar 94], onde o tratamento é puramente sintático. A partir de uma palavra com m caracteres, são obtidas todas as subseqüências com n caracteres nela contidas. Por exemplo, a partir da palavra com *house* e considerando $n = 4$, são obtidos os 4-grams seguintes: *_hou*, *hous*, *ouse* e *use_* onde o sublinhado é usado para indicar o começo ou o fim da palavra. A representação que usa *n-grams* normalmente produz um número maior de elementos do que o uso sucessivo da eliminação de *stopwords* e o *stemming*.

A execução do pré-processamento o texto gerado está consideravelmente reduzido em relação ao texto original. Evidentemente os procedimentos 1 e 2 requerem algum conhecimento sobre o idioma no qual o documento está escrito, desde as *stopwords* e os radicais das palavras são idioma-dependentes, enquanto que o procedimento 3 é independente de idioma.

O modelo vetorial

Após a indexação, os documentos podem ser facilmente considerados segundo o modelo vetorial. Neste modelo cada dimensão está associada a um termo da coleção. Como previamente indicado os termos podem corresponder: (1) às palavras originais no texto - quando não há *stemming*; (2) aos radicais obtidos pelo *stemming*; ou (3) às cadeias geradas pela obtenção das *n-grams*.

Formalmente, seja $C = (d_1, d_2, \ldots, d_m)$ um conjunto ordenado de m documentos d_i, com n termos diferentes. Então a representação de um documento i será $d_i = (f_{i1}, f_{i2}, \ldots, f_{in})$, onde f_{ij} é a função de ponderação associada a termo j no documento i. A função de ponderação (peso) f_{ij} mais utilizada em conjunto com o modelo vetorial é conhecida como *tf * idf* [Salton 97], na qual: $f_{ij} = tf_{ij} \ln(\frac{m}{idfj})$, onde tf_{ij} é a freqüência do termo j (tf) no documento i, idf_j é o número de documentos em que o termo j aparece, e m é o tamanho de coleção. Outras medidas, como a freqüência simples (tf_{ij}), também podem ser usadas [Salton 97].

Desta forma, de acordo com o modelo vetorial, um conjunto de documentos pode ser visto como um (enorme) matriz $C_{m \times n}$, onde f_{ij} é o peso do termo de j em documento i, n é o número de condições e m é o número de documentos no conjunto [Berry 99]:

$$C = \begin{bmatrix} f11, f12, \ldots, f1n \\ f21, f22, \ldots, f2n \\ \ldots\ldots\ldots\ldots \\ fm1, fm2, \ldots, fmn \end{bmatrix}$$

Classificação de documentos e o classificador *k*-NN

A partir do modelo vetorial de representação, a tarefa de classificação (ou categorização) de documentos pode ser formalmente definida. A tarefa corresponde ao problema de classificação supervisionada no contexto de Reconhecimento de Padrões [Duda 00]. Considera-se que todo documento da coleção pertence a uma *classe*, tomada sobre um conjunto finito não vazio de classes. Um classificador é um procedimento para determinar uma classe de um determinado documento. Outras tarefas de tratamento de textos também são descritas na literatura, tais como a busca e recuperação de informações [Baeza-Yates 99], a filtragem de documentos [Belkin 92] e a sumarização de documentos [Mani 01].

No caso do uso de algoritmos baseados em aprendizagem de máquina, um conjunto de documentos previamente classificado é conhecido. A utilização de um algoritmo de classificação treinável geralmente ocorre da seguinte forma: um "conjunto de treinamento" formado por pares (*documento, classe*) é fornecido. O algoritmo deve, durante o treinamento, "memorizar" o padrão de cada classe, a ser previsto a partir dos atributos obtidos em cada documento, gerando uma estrutura de decisão. Posteriormente, durante a fase de classificação, a estrutura de decisão gerada deve associar uma classe a um (novo) documento não rotulado.

Um classificador famoso na literatura de Reconhecimento de Padrão é o procedimento de *k*-NN [Duda 00]. Ele é extensamente usado, principalmente por causa da sua simplicidade conceitual e seu erro limitado. De maneira geral um classificador de *k*-NN associa a um determinado documento *d* à classe mais freqüente entre aquelas dos *k* vizinhos mais próximos de *d* na coleção, de acordo com algumas distâncias calculadas no espaço de documentos.

Em tratamento de texto as distâncias empregadas mais comuns (entre dois documentos d_i e d_j) é a distância euclidiana $dist(d_i, d_j) = \left[\sum_{k=1}^{n} (f_{ik} - f_{jk})^2\right]^{1/2}$ e a assim chamada "métrica do cosseno" $cos(d_i, d_j) = \frac{d_i * d_j}{\|d_i\| . \|d_j\|}$, que é amplamente utilizada na área de Recuperação de Informações [Salton 97].

Avaliação do desempenho

A avaliação do desempenho dos sistemas automatizados de classificação é feita em geral pela taxa de correção, isto é, o número de documentos corretamente classificados. Esta correção pode também ser calculada por classe, dependendo dos objetivos da avaliação.

Também se empregam as medidas de *cobertura* e *precisão* [Baeza-Yates 99]. Estas medidas exigem que se conheçam as respostas corretas à tarefa em questão. Considerar-se-á, nas definições a seguir, um documento como *relevante* se pertencer ao conjunto de respostas corretas, e *recuperado* se for obtido como resposta do sistema automatizado de tratamento. Precisão (*precision*) é definida como a razão entre o número de documentos relevantes recuperados pelo sistema automatizado e o número total de documentos recuperados; e cobertura (*recall*) é definida como a razão entre o número de documentos relevantes recuperados e o número total de documentos relevantes na coleção.

3- Subespaços randômicos e classificadores k-NN

Devido à enorme dimensão do espaço de documentos, a proposta deste trabalho é a da sua divisão em subespaços, de forma randômica, e a utilização em cada subespaço de um classificador específico; posteriormente os resultados dos classificadores são combinados para produzir o resultado final.

O procedimento proposto pode ser descrito como segue. Inicialmente algumas colunas da matriz (*documentos* x *termos*) *C* são randomicamente selecionadas. Se 1,2,…,*n* são as colunas de *C*, e se *F* é um subconjunto destas colunas, a *F* está associado um subespaço de dimensão *f*, *proj F (C)* representa a (*m* x *f*) matriz obtida de *C* pela projeção em *F*, e *proj F (d)* é a matriz (*1* x *f*) que corresponde ao documento *d*.

Obviamente um classificador pode ser aplicado facilmente ao subespaço *F*. Nesta proposta emprega-se o classificador *k*-NN, baseado na métrica do co-seno como critério de classificação básica. Por exemplo, dada uma matriz de treinamento *C* e um documento não classificado *d*, para número de vizinhos $k = 1$ tem-se a seguinte regra de classificação:

$Class\ (d) = Class(d_i)$ onde d_i é tal que:

$cos(\ proj_X\ (d_i), proj_X\ (d)\ ) < cos(\ proj_X\ (d_j), proj_X\ (d)\ )$ para todo $1 \leq j \leq m$, $i \neq j$.

O procedimento anterior pode ser repetido *p* vezes, gerando *p* subespaços e respectivamente *p* classificações para *d*. Nas experiências realizadas considerou-se todos os subespaços com a mesma dimensão, isto é, *f* é constante para todos os *p* subespaços.

Finalmente, para se obter uma única classe para *d*, deve-se usar um procedimento de decisão para combinar os resultados individuais.

Neste trabalho foram empregadas duas regras de decisão:

1) Na primeira regra de decisão, emprega-se o princípio de maioria de votos diretamente: a classe assinalada ao documento *d* é a classe mais freqüente retornada pelos *p* classificadores.

2) Na segunda regra de decisão, em primeiro lugar constrói-se um conjunto cujos elementos são os documentos que constituem os mais próximos vizinhos nomeados pelos *p* classificadores; em seguia, determina-se a classe de cada elemento deste conjunto e seleciona-se o resultado mais freqüente. Na realidade, este procedimento considera apenas documentos diferentes para calcular a classe final a ser assinalada, não importando quantas vezes este documento aparece nas *p* classificações retornadas.

O método global pode ser considerado como um derivativo do método de discriminação estocástica, onde vários classificadores estocasticamente criados são combinados em ordem crescente de precisão. Estes procedimentos têm sido aplicados com sucesso em outros domínios [Ho 98].

4- Experimentos realizados e resultados obtidos

Para verificar a aplicabilidade da proposta no problema de classificação de documentos, algumas experiências foram feitas. Foram empregadas duas coleções: (1) a coleção TIPSTER, oriunda da *Text Retrieval Conference* (TREC) [Trec 04], a qual é destinada à avaliação de sistemas automáticos para o tratamento de documentos; e (2) a coleção REUTERS [Lewis 04], que foi construída especificamente para testar sistemas de classificação de textos.

A coleção de TIPSTER é formada por milhares de documentos originais em inglês (em formato XML), com tamanhos que variam de uma ou duas linhas até uma ou duas páginas. Os documentos são agrupados em subconjuntos formados por dez até mil elementos.

A TREC não tem uma tarefa específica de classificação de documentos. Porém, uma das tarefas de TREC era a busca e recuperação de documentos: a partir de uma consulta (ou tópico, na terminologia usada em TREC) proposta por um usuário, o sistema deve devolver uma lista ordenada dos documentos que são pertinentes à questão. Na TREC, para propósitos de avaliação, uma lista de documentos pertinentes foi construída para cada consulta; estas listas foram obtidas pelo julgamento da relevância entre cada documento da coleção e a consulta, preparados por um grupo de especialistas humanos.

Para utilizar esta coleção na tarefa de classificação, considerou-se que as classes são formadas pelos documentos que respondem a mesma questão na tarefa descrita anteriormente. Nas experiências preliminares foram selecionados da coleção TIPSTER 200 documentos que respondem 10 consultas, formando 10 classes com 20 elementos cada.

Na primeira experiência os documentos foram pré-processados usando eliminação de *stopwords*, *stemming* e um filtro que seleciona somente termos que aparecem em pelo menos dois documentos. A lista de *stopwords* foi obtida da Biblioteca BOW – da Universidade Carnegie Mellon, e utilizou-se o algoritmo de Porter para o *stemming* [Porter 97]. Finalmente, aplicou-se um procedimento de filtragem que seleciona apenas os termos que aparecem em pelo menos dois documentos (#docs>1). No total o pré-processamento gerou 3.342 termos, ou seja, uma matriz $C_{200\ x\ 3342}$. Os elementos de C foram calculados usando a freqüência simples, isto é $f_{ij} = tf_{ij}$.

Três quartos da coleção (75% = 150 documentos) foram usados no treinamento, e o restante (25% = 50 documentos) para teste. Foram empregados 30 subespaços randômicos com dimensão 50, isto é, $p = 30$ e $f = 50$ no formalismo previamente descrito. A experiência foi repetida para os valores $f = 100$ e $f = 150$. Foram considerados os casos onde k (número de vizinhos) varia de 1 a 2. Os resultados obtidos são resumidos à Tabela 1, que apresenta a precisão total, ou seja, a porcentagem correta de classificação para todas as classes.

A experiência foi repetida mudando-se apenas a etapa do pré-processamento. Neste caso, utilizou-se a eliminação de *stopwords*, a geração de 4-*grams*, e o mesmo procedimento de filtragem, gerando uma Matriz $C_{200\ x\ 8659}$. Também se empregaram 30 subespaços com dimensões 150, 300 e 450. Os resultados obtidos também são apresentados à Tabela 1.

Tabela 1: Correção na classificação (%), coleção TIPSTER, classes balanceadas

	Pré-processamento: eliminação de *stop-words*, *stemming* e filtro #docs > 1					
	$f = 50$		$f = 100$		$f = 150$	
	1ª regra	2ª regra	1ª regra	2ª regra	1ª regra	2ª regra
$k = 1$	35	64	37	60	52	54
$k = 2$	30	52	34	48	40	40
	Pré-processamento: eliminação de *stop-words*, 4-*grams*, e filtro #docs > 1					
	$f = 150$		$f = 300$		$f = 450$	
	1ª regra	2ª regra	1ª regra	2ª regra	1ª regra	2ª regra
$k = 1$	20	44	49	60	38	40
$k = 2$	26	40	40	34	34	44

Na segunda experiência os testes foram repetidos para classes não balanceadas. Foram selecionados 200 documentos da coleção TIPSTER, com freqüências diferentes para cada classe. Empregou-se exatamente o mesmo pré-processamento na execução da etapa. A matriz obtida tem dimensão (200 x 3369) - com o uso da eliminação de *stopwords*,

stemming e filtro (#docs>1), e dimensão (200 x 8704) - com o uso de eliminação de *stopwords*, 4-*grams* e filtro (#docs>1). Os resultados da precisão total são apresentados à Tabela 2.

Tabela 2: Correção na classificação (%), coleção TIPSTER, classes não-balanceadas

	Pré-processamento: *stop-words*, *stemming* e filtro #docs > 1					
	$f = 50$		$f = 100$		$f = 150$	
	1ª regra	2ª regra	1ª regra	1ª regra	2ª regra	1ª regra
$k = 1$	36	58	50	50	50	54
$k = 2$	46	56	52	38	52	46
	Pré-processamento: *stop-words*, 4-*grams*, e filtro #docs > 1					
	$f = 150$		$f = 300$		$f = 450$	
	1ª regra	2ª regra	1ª regra	1ª regra	2ª regra	1ª regra
$k = 1$	40	52	44	34	54	46
$k = 2$	28	38	32	38	40	34

As experiências seguintes utilizam a coleção REUTERS [Lewis 04]. Esta coleção é formada por 22 grupos de documentos, assinados por *tags* XML, e pertencentes a uma ou várias classes. Desta forma as classes de cada documento são explicitamente indicadas nos próprios documentos. A classificação indicada foi obtida através do julgamento de pertinência preparado por especialistas humanos. No experimento foi empragado o segundo grupo, com 1000 documentos, e novamente um pré-processamento e uma etapa de teste muito semelhantes ao das experiências anteriores. A partição de treinamento e teste foi de 70% e 30% respectivamente.

Na terceira experiência utilizaram-se os seguintes parâmetros: (1) pré-processando por eliminação de *stopwords*, *stemming* e filtro (#docs≥2); (2) $f_{ij} = tf_{ij}$ e $f_{ij} = tf*idf_{ij;}$ (3) $p = 30$; (4) $f = 1000$; (5) $k = 1{,}2$; (6) as regras para a combinação de resultados já explicadas (1ª. e 2ª. regras). A matriz C obtida tem dimensão (1000 x 3633). Consideraram-se duas classificações diferentes na coleção: <Lugar> (<*Place*>), que representam uma classe com 133 elementos diferentes, e <Tópico> (<*Topic*>) que é uma classe com 82 elementos diferentes. São apresentados os resultados das precisões totais obtidas à Tabela 3.

Tabela 3: Correção na classificação (%), coleção REUTERS

	Pré-processamento: *stop-words*, *stemming* e filtro #docs > 2			
	Classe = <*Place*>			
	$f = 1000$, $f_{ij} = tf_{ij}$		$f = 1000$, $f_{ij} = tf*idf_{ij}$	
	1ª regra	2ª regra	1ª regra	2ª regra
$k = 1$	60	60	63	63
$k = 2$	60	60	60	63
	Classe = <*Topic*>			
	$f = 1000$, $f_{ij} = tf*idf_{ij}$			
	1ª regra		2ª regra	
$k = 1$	63		64	
$k = 2$	57		62	

Na experiência seguinte utilizou-se a coleção REUTERS com a categorização <*Topic*>. Foram utilizados os seguintes procedimentos na etapa de pré-processamento: eliminação de *stopwords*, *stemming* e a aplicação de um filtro que seleciona para compor a matriz somente

os termos com freqüências médias: no caso foram considerados aqueles que se aparecem mais de 15 vezes e menos de 287 vezes na coleção. Estes valores foram obtidos a partir do histograma de freqüência dos termos da coleção. A matriz *C* obtida tem dimensão (1000 x 943). Empregaram-se os seguintes parâmetros: (1) $f_{ij} = tf*idf_{ij}$; (2) $p = 30$; (3) $f = 100$, 200, 300, 400 e 500; (4) $k = 1,2$; (5) as mesmas regras de decisão para a combinação dos resultados (1ª. e 2ª. regras). Os resultados obtidos são resumidos à Tabela 4.

Tabela 4: Correção na classificação (%), coleção REUTERS

	Pré-processamento: *stopwords*, *stemming* e filtro de frequência					
	$f = 100$		$f = 300$		$f = 500$	
	1ª regra	2ª regra	1ª regra	2ª regra	1ª regra	2ª regra
$k = 1$	62	60	63	67	65	63
$k = 2$	61	58	60	64	60	62

Enfatiza-se que os resultados apresentados utilizam como elemento de avaliação a precisão total, que considera todas as classes na coleção e não apenas classes selecionadas. Isto pode explicar a magnitude das taxas de precisão obtidas, consideradas baixas em comparação com outros resultados da literatura.

Para melhor avaliar a questão indicada acima, na última experiência calculou-se a precisão para cada classe. O experimento anterior foi repetido com 2000 documentos, e o mesmo conjunto de parâmetros. Os resultados obtidos para classes selecionadas de parâmetros $f_{ij} = tf*idf_{ij}$; $p = 30$, $f = 400$ são apresentados à Tabela 5. Estes resultados são consistentes com vários encontrados na literatura, como será indicado nas conclusões.

Tabela 5: Correção na classificação (%), coleção REUTERS

	$k = 1$		$k = 2$	
<*Topic*>	1ª regra	2ª regra	1ª regra	2ª regra
Acq	86	87	85	87
Crude	96	97	96	97
Grain	99	99	99	99
Interest	98	99	98	99
Money-fx	98	99	98	99
Ship	99	99	99	99
Trade	98	98	98	98

5- Conclusões e trabalho futuros

Este artigo tratou da tarefa da classificação automática de documentos, propondo uma nova abordagem que utiliza subespaços gerados de forma aleatória do subespaço original dos documentos e um conjunto de classificadores do tipo *k*-NN. Esta é uma abordagem nova para o problema, que já foi aplicada com sucesso em outros domínios.

Nas experiências realizadas empregou-se o modelo vetorial como formalismo subjacente à proposta; deste modo a aplicação do método ao problema pode ser feita de forma direta. As avaliações realizadas foram concentradas no cálculo da precisão total de classificação. Os resultados obtidos são encorajadores e mostram a aplicabilidade do método.

Os melhores resultados (58 % a 64 %) nos experimentos com a coleção de TIPSTER – 200 documentos, 10 classes - foram obtidos usando eliminação de *stopwords* e *stemming*, e

utilizando $p = 30$ subespaços com dimensão $f = 50$, usando $k = 1$ vizinho mais próximo na classificação, e a segunda regra para combinação de classificadores.

Para a coleção de REUTERS – 1000 documentos, com 133 e 82 classes, dependendo da categoria, os resultados obtidos com as variações nos parâmetros foram semelhantes, variando de 57 % a 67 %. Os melhores resultados usaram eliminação de *stopwords* e *stemming*, e foram obtidos usando um filtro de freqüência para o pré-processamento, com os seguintes parâmetros: $k = 1$ vizinho mais próximo, $p = 30$ subespaços de dimensão $f = 300$, e a segunda regra de combinação dos classificadores. Apresentou-se também, para uma experiência específica, a precisão por classe, na qual os valores obtidos variaram de 85 % a 99 %. Como se pode observar nos resultados, surpreendentemente, a segunda regra de decisão produziu os melhores resultados.

Para fins de comparação apresentam-se a seguir alguns resultados obtidos para a tarefa de classificação encontrados na literatura da área. Joachims [Joachims 98] obteve m resultados de 57 a 86% para o *breakeven point* (ponto onde precisão = cobertura) em várias experiências. Schapire e Cantor [Schapire 00] obtiveram valores que variam de 85 % a 95 % para a taxa de precisão no algoritmo de AdaBoost, mas só em sub-coleções selecionadas da REUTERS. Nigan et all [Nigan 00] na mesma coleção obtiveram resultados entre 56 % e 66 % para o *breakeven point*, usando um algoritmo adaptável. Tong e Koller [Tong 01] apresentam resultados de 93.4 % de precisão e 54.3 % para o *breakeven point* mas só considerando apenas as 10 classes de maior freqüência na coleção.

Desta forma considera-se que os resultados obtidos são aceitáveis em várias situações práticas, mostrando que os procedimentos de classificação definidos nesta proposta podem ser aplicados eficazmente na tarefa de classificação de textos, no caso do uso de ferramentas automáticas ou semi-automáticas.

Outros experimentos devem ser realizados na continuidade do trabalho, nas seguintes direções:

- Uso de outras medidas para avaliação de desempenho, de forma a permitir a compação direta dos resultados com os obtidos em outras abordagens;
- Aplicação do procedimento em uma coleção maior, para avaliar a escalabilidade;
- Realização de mais testes com diferentes parâmetros, como as dimensões dos subespaços (f), variações no número de classificadores (p), e assim por diante;
- Uso de diferentes técnicas para seleção dos subespaços, usando critérios mais sofisticados, como Análise Semântica Latente [Berry 90], [Zha 98], [Zha 98b];
- Uso de classificadores diferentes em cada subespaço, como árvores de decisão, naïve-Bayes, etc [Deb 01], [Mitchell 97].

6- Referências

[Baeza-Yates 99] Baeza-Yates, R.; Ribeiro-Neto, B. *Modern Information Retrieval*. Addison-Wesley, 1999.

[Belkin 92] Belkin, N.; Croft, W. "Information Filtering and Information Retrieval: Two Sides of the Same Coin". *Communications of the ACM*, Nº 35, pp. 29-38, 1992. .

[Berry 99] Berry, M.; Drmac, Z.; Jessup, E. "Matrices, Vector Spaces, and Information Retrieval", *SIAM Review*, Vol. 41, Nº 2, pp.335-362, 1999.

[Cavnar 94] Cavnar, W. B. "Using An N-Gram-Based Document Representation With a Vector Processing Retrieval Model". In *Proceedings Of TREC-3 (Third Text Retrieval Conference)*. Gaithersburg, Maryland, USA, 1994.

[Deb 01] Deb, K. *Multi-Objective Optimization using Evolutionary Algorithms*, John Wiley & Sons, 2001.

[Dhillon 01] Dhillon, I.; Modha, D. "Concept Decompositions for Large Sparse Text Data using Clustering". *Machine Learning*, 42:1, pp. 143-175, 2001.

[Duda 00] Duda, R.; Hart, P.; Stork, D. *Pattern Classification (2nd. Edition)*, Wiley Interscience, 654 p., 2000.

[Ho 98] Ho, T.K. "The Random Subspace Method for Constructing Decision Forests", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 20, Nº 8, pp. 832-844, 1998.

[Joachims 98] T. Joachims, T. "Text Categorization with Support Vector Machines: Learning with Many Relevant Features". *In Proceedings of the European Conference on Machine Learning* (ECML), Springer, 1998.

[Lewis 04] Lewis, D.D. http://www.daviddlewis.com/resources/testcollections/reuters21578/; accessed on [03/08/2004].

[Lyman 03] Lyman, P. and Varian H.R. (2003). How Much Information. Retrieved from http://www.sims.berkeley.edu/how-much-info-2003 accessed on [01/19/2004].

[Mani 01] Mani, I. *Automatic Summarization*. J.Benjamins Publishing Co. Amsterdam Philadelphia, 2001.

[Mitchell 97] Mitchell, T. *Machine Learning*. McGraw-Hill, 414p., 1997.

[Nigan 00] Nigan, ; McCallum, A.K.; Thrun, S.; Mitchell, T. "Text Classification from Labeled and Unlabeled Documents using EM". *Machine Learning* 39, pp. 103-134, 2000.

[Porter 97] Porter, M.F. "An algorithm for suffix stripping". *Program 14*, 130-137. 1980. Reprinted in: Sparck-Jones, K.; Willet, P. (eds.) *Readings in Information Retrieval*. Morgan Kaufmann, pp. 313-316, 1997.

[Salton 97] Salton, G.; Buckley, C. "Term-weighting approaches in automatic text retrieval". *Information Processing and Management 24,* 513-523. 1988. Reprinted in: Sparck-Jones, K.; Willet, P. (eds.) *Readings in Inf.Retrieval*. Morgan Kaufmann, pp. 323-328, 1997.

[Schapire 00] Schapire, R.E.; Singer, Y. "BoosTexter: A Boosting-based System for Text Categorization". *Machine Learning*, 39, pp.135-168, 2000.

[Sebastiani 02] Sebastiani, F. "Machine learning in automated text categorization". *ACM Computer Surveys*, Vol. 34, Nº1, pp.1-47, 2002.

[Sparck-Jones 97] Sparck-Jones, K.; Willet, P. (Eds.) *Readings in Information Retrieval*. Morgan Kaufmann, 1997.

[Tong 01] Tong, S.; Koller, D. "Support Vector Machine Active Learning with Appplications to Text Classification". In *Journal of Machine Learning Research*, pp.45-66, 2001.

[Trec 04] http://trec.nist.gov/data.html; accessed on [03/08/2004].

[van Rijsbergen 92] van Rijsbergen, C.J. Probabilistic retrieval revisited. *The Computer Journal*, Vol. 35, No. 3, pp. 291-298, 1992.

[Wartik 92] Wartik, S. "Boolean Operations". *In Information Retrieval: Data Structures and Algorithms*. Frakes, W.B.; Baeza-Yates, R. (Eds.), Prentice Hall, pp. 264-292, 1992.

[Zha 98] Zha, H.; Marques, O.; Simon, H. "Large-Scale SVD and Subspace-Based Methods for Information Retrieval". IRREGULAR '98, Berkeley, California, USA, *Lecturer Notes in Computer Science Nº 1457*, Springer Verlag, pp. 29-42, 1998.

[Zha 98b] Zha, H.; Marques, O.; Simon, H. "A Subspace-Based Model for Information Retrieval with Applications in Latent Semantic Indexing". IRREGULAR '98, Berkeley, California, USA, *Lecturer Notes in C. Science Nº 1457*, Springer Verlag, pp.29-42, 1998.

[Zhong 02] Zhong, N.; Liu, J.; Yao, Y. "In Search of the Wisdom *Web*". *IEEE Computer*, Vol. 35, Nº 1, pp. 27-31, 2002.