UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

CENTRO TECNOLÓGICO

PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

MARCOS RODRIGUES SAÚDE

UMA ESTRATÉGIA PARA MODERAÇÃO AUTOMÁTICA DE UM GRANDE CONJUNTO DE COMENTÁRIOS DE USUÁRIOS

VITÓRIA-ES

2014

MARCOS RODRIGUES SAÚDE

UMA ESTRATÉGIA PARA MODERAÇÃO AUTOMÁTICA DE UM GRANDE CONJUNTO DE COMENTÁRIOS DE USUÁRIOS

Dissertação apresentada ao Programa de Pós-Graduação em Informática do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para a obtenção de Título de Mestre em Informática.

Orientador: Elias de Oliveira

Co-orientador: Patrick Marques Ciarelli

VITÓRIA-ES

2014

Dados Internacionais de Catalogação-na-publicação (CIP)
(Biblioteca Setorial Tecnológica,
Universidade Federal do Espírito Santo, ES, Brasil)

Saúde, Marcos Rodrigues, 1975-

S255e Uma estratégia para moderação automática de um grande conjunto de comentários de usuários / Marcos Rodrigues Saúde. – 2014.

88 f. : il.

Orientador: Elias de Oliveira.

Coorientador: Patrick Marques Ciarelli.

Dissertação (Mestrado em Informática) – Universidade Federal do Espírito Santo, Centro Tecnológico.

1. Recuperação da informação. 2. Documentos – Indexação automática. 3. Moderação automática de comentários. 4. Redução de dimensionalidade (Computação). 5. Seleção de características (Computação). I. Oliveira, Elias de. II. Ciarelli, Patrick Marques. III. Universidade Federal do Espírito Santo. Centro Tecnológico. IV. Título.

CDU: 004

MARCOS RODRIGUES SAÚDE

UMA ESTRATÉGIA PARA MODERAÇÃO AUTOMÁTICA DE UM GRANDE CONJUNTO DE COMENTÁRIOS DE USUÁRIOS

Dissertação submetida ao programa de Pós-Graduação em Informática do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para a obtenção do Grau de Mestre em Informática.

Aprovada em 29 de setembro de 2014.

COMISSÃO EXAMINADORA

Prof. Dr. Elias de Oliveira
Universidade Federal do Espírito Santo
Orientador

Prof. Dr. Patrick M. Ciarelli
Universidade Federal do Espírito Santo
Co-orientador

Profa. Dra. Karin Satie Komati
Instituto Federal do Espírito Santo

Profa. Dra. Claudine Badue
Universidade Federal do Espírito Santo

A Deus, por me guardar nas viagens, e pela graça de conseguir concluir este trabalho.

À minha esposa, Queila Nepomuceno, meu presente, por estar sempre ao meu lado.

Às minhas filhas, Talita e Ana Júlia, por darem novo sentido à minha vida.

Ao meu pai, Sinval, pelos inesquecíveis gestos de incentivo aos meus estudos.

À minha mãe, Solange, por sua vida dedicada à família e pelo apoio incondicional.

Agradecimentos

Considerando esta dissertação como resultado de uma caminhada que não começou na UFES, agradecer pode não ser tarefa fácil, nem justa. Para não correr o risco da injustiça, agradeço de antemão a todos que de alguma forma passaram pela minha vida e contribuíram para a construção de quem sou hoje.

E agradeço, particularmente, a algumas pessoas pela contribuição direta na construção deste trabalho:

- À Universidade Federal do Espírito Santo (UFES), por me oportunizar um aperfeiçoamento gratuito e de excelência.
- Ao prof. Elias Oliveira, pela paciência, e por acreditar que eu poderia ir mais longe, sendo o maior incentivador na superação de meus limites.
- Ao prof. Patrick Marques Ciarelli, por estar sempre pronto a me ouvir, por toda a atenção, disponibilidade e apoio na elaboração deste trabalho.
- Ao colega Frederico Pinto de Souza, por todo apoio e consideração, por me ajudar a transpor os momentos difíceis com palavras de ânimo.
- Aos colegas Eduardo Pissinati e Henrique Basoni, pelo companheirismo em todo este tempo, e ajuda na resolução de tarefas de interesse comum.
- Aos professores Giancarlo Guizzardi e Renata Guizzardi, pelo acolhimento, e por me incentivarem a trilhar esta trajetória.
- Aos demais professores do PPGI, que com seus conhecimentos compartilhados em sala de aula, ajudaram a completar o conhecimento necessário para a conclusão deste trabalho.

- À Globo.com, nas pessoas de Marcelo de Medeiros Soares e Silvano Buback, por todo apoio e por fornecerem as bases de dados objetos deste trabalho.

“. . . e levo esse sorriso, porque já chorei demais.

Hoje me sinto mais forte, mais feliz, quem sabe;

só levo a certeza de que muito pouco eu sei

. . . ou nada sei.”

(Almir Sater e Renato Teixeira)

Publicações

Como parte deste trabalho, foram desenvolvidos e publicados os seguintes trabalhos, no período do mestrado, que apresentam, em maior ou menor grau, relação com o tema proposto.

Publicação em revista:

- Saúde, M. R.; Soares, M. M.; Ciarelli, P. M.; Oliveira, E. **Seleção de características aplicada à moderação automática de comentários de usuários**. Revista Eletrônica Científica Inovação e Tecnologia. Universidade Federal Tecnológica do Paraná. Campus Medianeira. No prelo (2014). ISSN 2175-1846.

Publicação de trabalhos em anais de congressos:

- Saúde, M. R.; Soares, M. M.; Basoni, H. G.; Ciarelli, P. M.; Oliveira, E. **A Strategy of Automatic Moderation of a Large Data Set of Users Comments**. XL Conferencia Latinoamericana en Informática (CLEI). Edición 40. Montevideo, Uruguay. 2014.
- Saúde, M. R.; Medeiros, M. S.; Ciarelli, P. M.; Oliveira, E. **Seleção de características aplicada à moderação automática de comentários de usuários**. V Meditec. Universidade Federal Tecnológica do Paraná. Medianeira, PR. 2014.
- Oliveira, E.; Basoni, H. G.; Saúde, M. R.; Ciarelli, P. M. **Combining Clustering and Classification Approaches for Reducing the Effort of Automatic Tweets Classification**. 6th Conference Internacional of Knowledge Discovery and Information Retrieval (KDIR). Rome, Italy. 2014.

Resumo

A expansão das mídias sociais e o advento da Web 2.0 promoveram a participação de pessoas interessadas em expor suas opiniões sobre o que se propõe discutir num ambiente coletivo ou sobre algum fato noticiado pela imprensa. No entanto, em virtude de mecanismos legais que exercem controle sobre material de cunho particularmente ofensivo, com expressões que agridem as personalidades, torna-se de grande interesse a classificação de documentos referentes a comentários inseridos por usuários de sites de notícias, com o intuito de se identificar quais podem ou não ser divulgados no ambiente digital, evitando-se demandas judiciais aos provedores desses ambientes. Este trabalho propõe o uso de técnicas de classificação automática para identificação de comentários cuja divulgação nos veículos de comunicação deva ou não ser permitida, auxiliando o ser humano no trabalho de moderação de comentários. Para tanto, foram exploradas várias técnicas no tratamento dos dados, tais como extração de sufixos das palavras *(stemming)*, redução de dimensionalidade e ponderação de termos. Todas essas técnicas foram estudadas no sentido de modelar um algoritmo capaz de imitar as decisões humanas para liberação ou não do comentário.

Palavras-chave: moderação automática, seleção de características, algoritmo genético

Abstract

The expansion of social media and the advent of Web 2.0 promoted the participation of persons interested in exposing their opinions on what it intends to discuss a collective environment or on any facts reported by the press . However, due to legal mechanisms to exert control over a particularly offensive punch, with expressions that attack personalities material, it becomes of great interest to the classification of documents relating to comments entered by users of news sites, in order to identify which may or may not be disclosed in the digital environment, avoiding judicial providers demands of these environments. This work proposes the use of automatic classification techniques to identifying reviews the disclosure in the media should be allowed or not, aiding humans in the work of comment moderation. For both, various techniques in data processing, such as reducing words to their canonical form, dimensionality reduction and weighting terms were explored. All these techniques have been studied in order to model an algorithm able to mimic human decisions to release or not the comment.

Keywords: automatic moderation, feature selection, genetic algorithms

Lista de Figuras

Lista de Tabelas

Sumário

Capítulo 1

Introdução

O homem é um ser social por natureza, necessitando se comunicar com seus semelhantes e conviver em uma sociedade organizada por regras e hierarquias. A comunicação, portanto, representa uma componente essencial para o exercício da cidadania e fortalecimento da cultura (CASTELLS, 1999).

Os veículos de comunicação hoje existentes estão espalhados por diversos meios, como a televisão, os jornais, a Internet, a rádio e as revistas. Dentre estes veículos de comunicação, a Internet se destaca atualmente como segundo meio mais utilizado por brasileiros (REPÚBLICA, 2014).

O advento da *Web* 2.0 ampliou a forma como a Internet interage com seus usuários, permitindo a produção de um grande volume de textos que denotam exposição de ideias, conceitos e opiniões do público a respeito das informações publicadas. Há, neste contexto, diversos *sites* que dão abertura à participação de seus leitores, permitindo registrar seus comentários, muitas vezes carregados de conteúdo malicioso.

1.1 Problema

A liberdade de expressar ideias também carrega a necessidade da responsabilidade pelo o que se expressa. Há, portanto, no ordenamento jurídico brasileiro, a existência de mecanismos legais que dão proteção sobre material de cunho particularmente ofensivo, ou mesmo que agridam os direitos das personalidades. Por exemplo, o Código Penal Brasileiro (OLIVEIRA; NAVES, 1984) prevê as seguintes sanções legais à prática de crimes de Calúnia (art. 138), Difamação (art. 139) e Injúria (art. 140):

> Art. 138 - Caluniar alguém, imputando-lhe falsamente fato definido como crime:
> Pena - detenção, de seis (seis) meses a 2 (dois) anos, e multa.
>
> Art. 139 - Difamar alguém, imputando-lhe fato ofensivo à sua reputação:
> Pena - detenção, de 3 (três) meses a 1 (um) ano, e multa.
>
> Art. 140 - Injuriar alguém, ofendendo-lhe a dignidade ou o decoro:
> Pena - detenção, de 1 (um) a 6 (seis) meses, ou multa.

Em relação à responsabilização por material divulgado em meios de comunicação, especificamente em ambiente virtual, temos que:

> [...] diante a esses abusos na internet, aquele usuário responsável por dizer, publicar, compartilhar mensagens indevidas, deverá ser responsabilizado pelos danos causados. Percebem-se vários casos que envolvem as redes sociais virtuais, é no sentido de criação de perfis falsos, veiculação de informações e imagens ofensivas. [...] No que tange a responsabilidade do usuário infrator, pode-se observar, diante dos casos apresentados, que este sofrerá a responsabilização pelas informações ilícitas vinculadas no ambiente virtual, e o provedor do site de relacionamento será responsabilizado somente se deixar de excluir ou

> bloquear as imagens ou informações ofensivas, após transcorrido certo prazo desde a notificação feita pela vítima (TRENTIN; TRENTIN, 2012).

Assim sendo, é recomendável que os comentários inseridos por usuários, para publicação nos meios de comunicação, devam ser submetidos a uma análise de seleção para posterior divulgação.

Dado o grande número de notícias constantemente divulgadas e a efetiva participação dos usuários em contribuir com suas ideias, a moderação manual de comentários torna-se uma atividade extremamente árdua para o ser humano, demandando tempo considerável de leitura, interpretação e análise dos comentários enviados.

Este trabalho faz uso de técnicas de classificação automática de documentos para identificação dos comentários, cuja divulgação no veículo de comunicação deva ou não ser permitida, na tentativa de imitar o especialista humano no trabalho de moderação dos comentários.

1.2 Objetivos

O objetivo deste trabalho é apresentar uma estratégia composta por técnicas de tratamento de textos e seleção de características, possibilitando a moderação automática de um grande conjunto de comentários de usuários, de tal forma que sua aplicação obtenha resultados próximos aos alcançados pelo especialista humano.

Para tanto, foram analisadas diversas técnicas de redução de dimensionalidade e seleção de características, sendo comparadas as medidas de acurácia da classificação após o uso destas técnicas, escolhendo-se as técnicas que apresentaram melhores resultados.

As medidas obtidas nos experimentos demonstram que a aplicação da estratégia apresentada neste trabalho melhora os índices de acerto da moderação automática, atingindo em alguns casos altos níveis de aproximação do trabalho realizado pelo especialista humano.

1.3 Metodologia do Trabalho

Inicialmente foi realizada uma revisão de literatura para identificação de estudos recentes que abordaram o tema de moderação automática de documentos ou temas similares.

Com a revisão, foram analisadas algumas técnicas de classificação a serem utilizadas no presente trabalho. Além disso, foram estudadas algumas técnicas de redução de dimensionalidade objetivando aumentar a eficiência dos algoritmos de classificação. Em seguida, foi realizado um levantamento de técnicas de seleção de características para extração de termos mais relevantes a serem aplicadas nas bases de dados, comparando-se os resultados da classificação em cada caso.

Para a realização dos experimentos, foram obtidas duas bases compostas por comentários reais cedidas pelo portal de notícias Globo.com (<http://g1.globo.com/>), sendo uma delas formada por um conjunto de 657405 comentários classificados manualmente pelo especialista humano, em categorias de comentários aprovados ou rejeitados para divulgação.

Os classificadores utilizados para comparação dos resultados foram o K-*Nearest Neighbors* (KNN) (MASAND; LINOFF; WALTZ, 1992), o *Centroid-Based Classifier* (CBC) (HAN; KARYPIS, 2000) e o *Support Vector Machine* (SVM) (VAPNIK; CORTES, 1995).

Foram adotadas métricas de avaliação de resultados de classificação para comparação de resultados antes e após as técnicas de extração de termos utilizadas. Os resultados foram comparados observando-se as medidas obtidas em cada etapa do tratamento de textos e das técnicas de seleção de características utilizadas.

Para o tratamento dos textos foram utilizadas técnicas de retirada de *stopwords*, extração de sufixos de palavras da língua portuguesa (*stemming*), retirada de termos pouco referenciados nos documentos da base, além de uma medida de ponderação de termos.

Para a extração de termos que melhor caracterizam cada categoria da base (seleção de características) foram utilizados o algoritmo SFS (*Sequential Forward Selection*), algorit-

mos genéticos e uma técnica que analisa diversas combinações de retirada de termos mais raros e termos mais comuns da base de dados. A estratégia foi dividida em etapas de pré-processamento, redução de dimensionalidade e seleção de características, sendo observada a evolução das medidas de classificação em cada uma dessas etapas.

1.4 Contribuições

Como principal contribuição deste trabalho para a moderação automática de comentários, destaca-se o uso de algoritmos genéticos buscando extrair os termos que melhor caracterizam os comentários de cada classe Aprovado e Reprovado. Para isso, o algoritmo genético será aplicado com propósito de aumentar a densidade de cada classe, separadamente. Busca-se, assim, aproximar os documentos de uma mesma classe, facilitando a classificação de um novo comentário.

Além disso, um refinamento nos índices da classificação foi obtida aplicando-se uma combinação de extração de termos mais raros e de termos mais comuns, seguida da aplicação do algoritmo genético. A estratégia aplicada à base de dados com 657405 comentários obteve uma taxa de acerto de cerca de 96.78% (*recall*) com uso do classificador KNN, sendo uma estratégia recomendável para casos reais de moderação automática.

1.5 Estrutura do Trabalho

Este capítulo apresentou uma introdução às principais ideias desta dissertação, descrevendo a motivação, definição do problema, objetivos e metodologia aplicada. Além desta introdução, este trabalho está organizado da seguinte forma:

- **Capítulo 2 (Classificação Automática de Documentos)**: são apresentados os principais conceitos que dão base à classificação automática de documentos, destacando o

modelo de representação vetorial de documentos, adotado no presente trabalho, bem como as técnicas de redução de dimensionalidade e de seleção de características a serem utilizadas nos experimentos.

- **Capítulo 3 (Trabalhos Relacionados)**: são mencionados outros trabalhos encontrados na literatura que são relacionados ao tema de moderação automática de documentos ou de conteúdo similar.
- **Capítulo 4 (Aprendizado Supervisionado)**: é explicado o princípio de funcionamento dos algoritmos de classificação adotados neste trabalho, bem como a forma como as bases de comentários devem ser organizadas para validação dos testes de classificação.
- **Capítulo 5 (Experimentos e Resultados)**: são descritos e discutidos os resultados obtidos nos experimentos com uso das técnicas utilizadas na estratégia proposta.
- **Capítulo 6 (Conclusões e Trabalhos Futuros)**: são apresentadas as conclusões do trabalho, suas contribuições e propostas futuras de aprimoramento do trabalho.

Capítulo 2

Classificação Automática de Documentos

A possibilidade de produção de textos em meio digital tem provocado um grande aumento no volume de informações registradas nas mais variadas áreas do conhecimento. Como forma de melhor organizar todo esse volume de dados, torna-se desejável agrupar as informações, facilitando a pesquisa, recuperação e disseminação das ideias. No entanto, interpretar e classificar esses documentos torna-se um grande desafio, uma vez que novos documentos são criados a cada instante em todas as partes do mundo, especialmente em virtude da evolução dos meios de comunicação.

De acordo com (SEBASTIANI, 2002), a classificação automática de documentos consiste em classificar um documento digital em uma determinada categoria, segundo critérios pré-estabelecidos *a priori* pelo especialista humano. Categorizar textos é, portanto, a tarefa de atribuir um valor lógico (verdadeiro ou falso) para cada par $(d_j, c_i) \in D \times C$, sendo D um domínio de documentos (*corpus*) e $C = c_1, \ldots, c_{|C|}$ um conjunto de categorias pré-definidas. Caso o valor (d_j, c_i) seja verdadeiro, o documento d_j será classificado como pertencente à classe c_i, e caso o valor seja falso significa que ele não pertence a esta categoria.

A implementação da classificação automática envolve a escolha de um modelo para a

representação lógica dos documentos. Muitos modelos têm sido propostos ao longo dos anos, sendo um dos mais clássicos o modelo de representação vetorial, adotado neste trabalho por ser largamente utilizado na literatura e por ter sido adotado no trabalho a ser discutido na Seção 3.1, que serviu de *baseline* desta dissertação, facilitando as comparações entre os resultados obtidos.

2.1 Modelo de Espaço Vetorial

O modelo de espaço vetorial constitui uma técnica baseada no modelo de representação de textos para tarefas de mineração conhecido como *bag of words*. Este modelo consiste em uma representação que se importa com a ocorrência das palavras, ignorando a ordem em que estas ocorrem no texto. Cada documento d_j da base de dados é representado em forma de um vetor. Cada dimensão desse vetor representa um termo t_i referenciado no documento, sendo associado a cada termo (dimensão) um valor de peso $w_{i,j}$, obtido a princípio pelo número de vezes que o termo ocorre (*Term Frequency* - tf) (SALTON; WONG; YANG, 1975). A Equação 2.1 apresenta a representação do documento no modelo vetorial:

$$d_j = (w_{1,j}, w_{2,j}, ..., w_{i,j}, ..., w_{n,j}) \tag{2.1}$$

onde n é o total de termos referenciados no documento.

Uma expressão de busca é representada neste modelo da mesma forma como são representados os documentos. A Figura 2.1 mostra a representação da expressão de busca **eBUSCA**$_1$ juntamente com os documentos **DOC**$_1$ e **DOC**$_2$ em um espaço vetorial formado pelos termos **t**$_1$**,** **t**$_2$ **e** **t**$_3$. Esta representação limitou o espaço em apenas três dimensões (três termos) para melhor visualização gráfica.

Em sistemas reais, o número de dimensões (termos) e de vetores (documentos) é muitas vezes bem mais elevado que o demonstrado na Figura 2.1. Por este motivo, o *corpus* pode

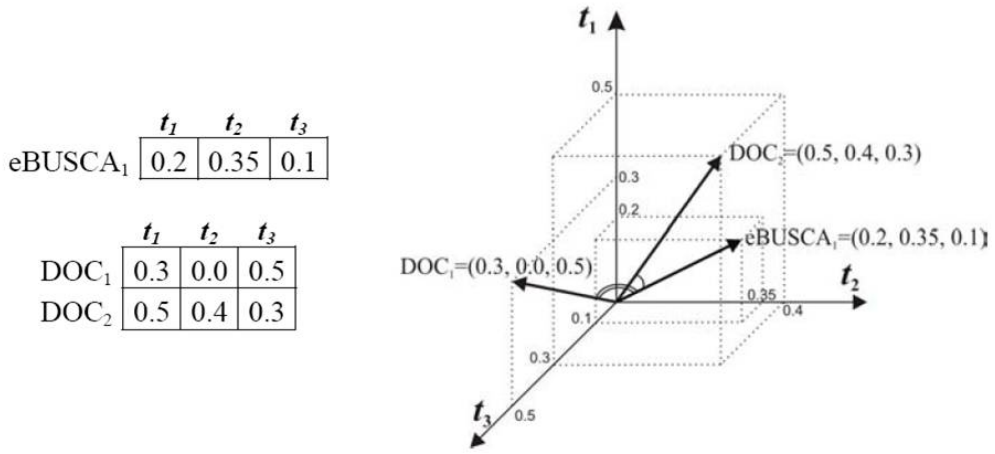

Figura 2.1: Representação de uma expressão de busca em um espaço vetorial (FERNEDA, 2012)

ser representado através de uma matriz onde cada linha representa um documento e cada coluna representa a referência a um termo. A Tabela 2.1 representa a matriz de um *corpus* contendo t documentos e n termos.

	$\mathbf{t}_1$	$\mathbf{t}_2$	$\mathbf{t}_3$	...	$\mathbf{t}_n$
DOC$_1$	$w_{1,1}$	$w_{1,2}$	$w_{1,3}$	...	$w_{1,n}$
DOC$_2$	$w_{2,1}$	$w_{2,2}$	$w_{2,3}$	...	$w_{2,n}$
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
DOC$_t$	$w_{t,1}$	$w_{t,2}$	$w_{t,3}$	...	$w_{t,n}$

Tabela 2.1: Matriz de pesos de um *corpus*

onde $w_{t,n}$ é o peso do t-ésimo termo do n-ésimo documento.

2.1.1 Cálculo de Similaridade

O modelo vetorial permite comparar o grau de similaridade entre conjuntos de documentos. Para este propósito, torna-se necessário o uso de métricas. Este trabalho utiliza o cál-

culo do cosseno do ângulo para medir a similaridade entre pares de documentos (vetores), especificada na Equação 2.2.

$$cos(d_i, d_j) = \frac{d_i.d_j}{||d_i||.||d_j||} = \frac{\sum_{k=1}^{n} w_{k,i}.w_{k,j}}{\sqrt{\sum_{k=1}^{n}(w_{k,i})^2}\sqrt{\sum_{k=1}^{n}(w_{k,j})^2}} \quad (2.2)$$

Sendo o valor $w_{k,i} \geq 0$ e $w_{k,j} \geq 0$, o valor do cos(d_i,d_j) deve assumir um valor entre 0 e 1. Esta medida indica maior similaridade entre dois documentos quanto mais próximo de 1 (um) for o seu valor, enquanto valores mais próximos de 0 (zero) indicam documentos menos similares.

Esta medida foi adotada em virtude de ser largamente utilizada na literatura, e para melhor comparação com os resultados obtidos no trabalho que serviu de *baseline* desta dissertação, que adotou esta métrica de similaridade entre documentos.

2.1.2 Caracterização de uma base de dados

Para a caracterização das bases de dados utilizadas neste trabalho serão adotadas as medidas de caracterização indicadas em (SALTON; WONG; YANG, 1975). A Figura 2.2 ilustra uma representação típica de documentos agrupados num espaço vetorial, onde os documentos são representados como caracteres “x”, os vários grupos de documentos (classes) são contornados por linhas fechadas, e os centroides de cada classe são representados por um ponto. Os centroides estão localizados mais ou menos no centro do seu respectivo grupo.

Seja D_j o domínio de representação dos m_j documentos da base de dados, para uma determinada classe C_j, j = 1,...,$|C|$, onde $|C|$ é o número de classes da base de dados. O peso de cada elemento (termo) t_k do centroide c_j da classe C_j pode ser definido como a média dos pesos dos mesmos elementos (termos) nos respectivos documentos pertencentes à classe C_j (Equação 2.3).

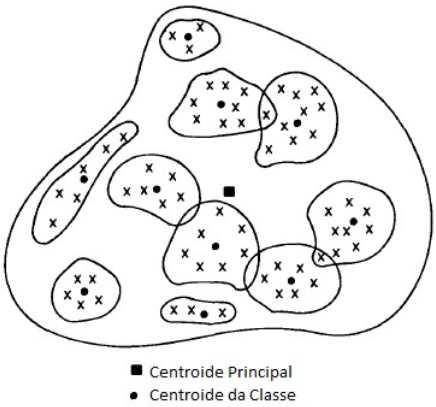

Figura 2.2: Representação espacial de documentos agrupados. Adaptado de (SALTON; WONG; YANG, 1975)

$$t_k = \frac{1}{m_j} \sum_{\substack{i=1 \\ d_i \in D_j}}^{m} d_{i,k} \tag{2.3}$$

De forma análoga, um centroide de uma base de dados pode ser definido considerando-se o conjunto completo de documentos do domínio D, identificado como um pequeno retângulo no centro da Figura 2.2. O centroide principal pode ser obtido de duas formas: através da média de todos documentos ou através da média de todos os centroides. Na primeira abordagem o centroide fica polarizado pela classe com o maior número de documentos, enquanto que na segunda abordagem cada classe tem uma contribuição igualitária na formação do centroide principal. As duas abordagens retornam o mesmo centroide se o número de documentos em cada classe for igual. Neste trabalho, o centroide principal é obtido através da média de pesos dos documentos existentes no domínio considerado.

A média de similaridades dos documentos de cada classe com seus respectivos centroides de classe (MSDC) é um valor de 0 (zero) a 1 (um), indicando que os documentos encontram-se mais agrupados em suas respectivas classes quanto mais próximo de 1 (um) for a medida MSDC.

A média de similaridades entre pares de centroides (MSPC) é um valor entre 0 (zero) e 1 (um), indicando que quanto mais próximo de 0 (zero) for a medida MSPC, mais separadas

estão as classes.

A média de similaridades entre os centroides de cada classe e o centroide principal (MSCCP) é um valor entre 0 (zero) e 1 (um), indicando que quanto mais próximo de 1 (um) for a medida MSCCP, mais próximas estão as médias de similaridades dos documentos de uma classe com a média geral de similaridades das classes.

A medida *ratio* indicada na Equação 2.4 é usada para medir a densidade geral do *corpus*. Quanto mais próxima de 0 (zero) for a medida, melhor a distribuição espacial para a classificação, ou seja, classes bem separadas com seus respectivos documentos bem agrupados.

$$ratio = \frac{MSPC}{MSDC} \tag{2.4}$$

2.2 Tratamento dos Textos

Os textos que compõem a base de dados devem ser submetidos a um pré-processamento para adequação do conjunto de termos que, se não tratados, podem causar ruídos no resultado final da classificação automática.

Técnicas de tratamento dos textos pertencentes a uma base de dados são utilizadas com intuito de identificar os termos que melhor caracterizam os documentos de uma categoria, eliminando-se termos com pouca ou nenhuma relevância. De maneira geral, após a extração de termos pouco relevantes no contexto da base, busca-se selecionar de um conjunto original T um subconjunto T' de termos que possibilite maior eficiência dos classificadores aplicados ao *corpus* (YANG; PEDERSEN, 1997).

Descreveremos nesta seção as técnicas utilizadas neste trabalho para tratamento dos textos.

2.2.1 Extração de *stopwords*

Stopwords são termos considerados não relevantes na indexação dos documentos por possuírem pouco valor semântico. Na maioria das vezes são utilizados na função de conectivos ou termos auxiliares, como pronomes, artigos, preposições, advérbios ou conjunções.

Em (BETTIO et al., 2007) é demonstrado alguns exemplos de termos considerados *stopwords* na língua portuguesa (Figura 2.3).

A	com	dos	já	também	sem	estão
O	não	como	está	só	mesmo	você
Que	uma	mas	seu	pelo	aos	tinha
E	os	foi	sua	pela	ter	foram
Do	no	ao	ou	até	seus	essa
Da	se	ele	ser	isso	quem	num
Em	na	das	quando	ela	nas	nem
Um	por	tem	muito	entre	me	suas
Para	mais	à	há	era	esse	meu
Qual	essas	tu	minhas	nossa	estes	isto
Será	esses	te	teu	nossos	estas	aquilo
Nós	pelas	vocês	tua	nossas	aquele	havia
tenho	este	vos	teus	dela	aquela	seja
Lhe	fosse	lhes	tuas	delas	aqueles	pelos
deles	dele	meus	nosso	esta	aquelas	elas
numa	têm	minha	às	a	de	cujo
A	com	dos	já	também	sem	estão
O	não	como	está	só	mesmo	você

Figura 2.3: Exemplos de *stopwords* (BETTIO et al., 2007)

Como pode ser observado na Figura 2.3, não é possível abstrair conceitos relevantes a partir destas palavras. Portanto, a extração das *stopwords* pretende remover termos que não adicionam informações relevantes e que podem prejudicar o processo de classificação.

2.2.2 Ponderação de termos tf-idf

Inicialmente, os pesos de cada termo num documento podem ser definidos como sendo o número de vezes que cada termo aparece no documento (SALTON; MCGILL, 1983). Essa medida denominada *term frequency* (tf) não faz distinção entre termos que ocorrem em todos os documentos e termos que ocorrem somente em alguns documentos. Termos que ocorrem em todos os documentos terão, provavelmente, pouca relevância em identificar os documentos.

Os pesos dos termos contidos em cada documento da base podem ser adequados com uso de fatores de ponderação, fazendo com que termos que ocorrem em grande quantidade de documentos tenham seu peso diminuído em virtude de sua menor importância na classificação. Uma medida de ponderação mais comumente utilizada é chamada idf (*Inverse Document Frequency*) proposto por (SALTON; YANG, 1973), que é calculada de acordo com o demonstrado na Equação 2.5.

$$idf_i = log\frac{|N|}{|n_i|}, \tag{2.5}$$

onde $|N|$ é o total de documentos do *corpus* e $|n_i|$ é o total de documentos que contém o termo t_i. O novo peso é calculado multiplicando-se este fator aos pesos de cada termo (tf*idf).

2.2.3 *Stemming*

Em Processamento de Linguagem Natural (PLN), o estudo da "normalização de variações linguísticas" busca reduzir os termos a elementos de escrita mais simples. Uma técnica conhecida como *stemming* consiste num processo de redução de formas variantes de uma palavra a uma representação comum (o radical da palavra, ou *stem*). Por exemplo, os termos "apresentadora", "apresentadores" e "apresentador" são essencialmente iguais, porém sem o processo de *stemming* serão tratados como três termos distintos.

Em (ORENGO; HUYCK, 2001) foi desenvolvido um algoritmo para extração de sufixos de palavras da língua portuguesa, denominado RSLP (Redutor de Sufixos da Língua Portuguesa). O algoritmo considera a extração de sufixos de palavras através de 8 (oito) passos, promovendo a retirada da forma plural, feminina, adverbial, aumentativo ou diminutivo, terminações verbais, vogais e remoção de acentos. Uma grande vantagem da utilização desta ferramenta de extração de radicais de termos da língua portuguesa é a utilização de um

dicionário externo e editável, contendo cerca de 32 mil palavras, com regras para a correta extração de sufixos, possibilitando remanejar seu conteúdo ou mesmo aperfeiçoar a extração através das regras de exceção contidas em sua configuração. Na Figura 2.4 é demonstrada a forma como o dicionário externo expressa suas regras.

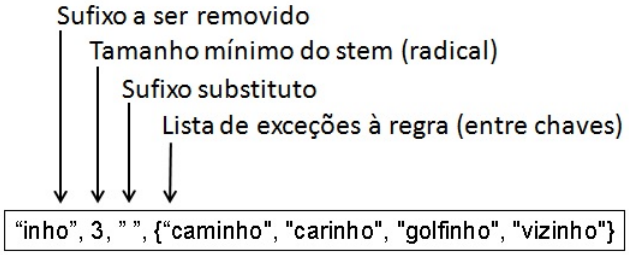

Figura 2.4: Exemplo da forma como a ferramenta RSLP expressa suas regras no dicionário externo editável

Na Tabela 2.2 estão demonstradas as retiradas de sufixos realizadas pelo RSLP nos termos “apresentadora”, “apresentadores” e “apresentador”.

Palavra	**Extração do Sufixo com RSLP**
APRESENTADORA	APRESENTAD
APRESENTADORES	APRESENTAD
APRESENTADOR	APRESENTAD

Tabela 2.2: Exemplo de aplicação da ferramenta RSLP (Redutor de Sufixos da Língua Portuguesa).

2.2.4 Frequência de Documentos (FD)

Termos referenciados numa pequena quantidade de documentos do *corpus*, em geral, são pouco significativos na composição do conjunto de palavras que melhor definem uma categoria. Dessa forma, a retirada de termos raramente mencionados reduz a dimensionalidade do espaço de termos, otimizando o processamento da classificação. O valor FD de um termo t_k é dado pela fórmula expressa na Equação 2.6.

$$FD(t_k) = \frac{|TR_{tk}|}{|Tr|} \tag{2.6}$$

Sendo TR_{tk} a quantidade de documentos que contêm o termo t_k no *corpus* considerado e Tr a quantidade total de documentos (YANG; PEDERSEN, 1997).

2.2.5 Seleção de características

Para (BAEZA-YATES; RIBEIRO-NETO, 2011) um grande espaço de características (ou termos) pode tornar impraticável a classificação de documentos, visto que a classificação de novos documentos consumiria muito tempo. A solução clássica para este problema é reduzir o tamanho do espaço de características, selecionando um sub-conjunto de todos os termos para a representação dos documentos. Este passo é chamado seleção de características (*feature selection*).

2.2.5.1 SFS (*Sequential Forward Selection*)

O algoritmo *Sequential Forward Selection* (SFS) inicia seu funcionamento a partir do conjunto vazio de termos, e vai adicionando sequencialmente à base de treino o termo x^+ cuja presença implica uma elevação na função objetivo (medida de avaliação) $J(Y_k + x^+)$, quando combinada com os termos Y_k já anteriormente selecionados (LADHA; DEEPA, 2011).

A seguir, estão descritos os passos do algoritmo:

1. inicia com o conjunto vazio $Y_0 = \emptyset$
2. seleciona o próximo termo $x^+ = \text{argmax}[J(Y_k + x)]$; $x \not\subset Y_k$
3. atualiza $Y_{k+1} = Y_k + x^+$; $k = k + 1$
4. volta para o passo 2

Ainda segundo (LADHA; DEEPA, 2011), o algoritmo SFS obtém melhor desempenho quando o subconjunto ideal obtém menor número de características (termos).

As desvantagens desse algoritmo são a impossibilidade de remover características já inseridas no subconjunto de dados após a adição de uma nova característica (ou termo), e também o custo computacional de sua execução.

2.2.5.2 Algoritmos Genéticos

Um algoritmo genético pode ser definido como um processo repetitivo que mantém uma população de "indivíduos" representando as possíveis soluções para um determinado problema (MITCHELL, 2002). A cada "geração", os indivíduos da população passam por uma avaliação de sua capacidade em oferecer uma solução satisfatória para o problema. Essa avaliação é feita por uma função de adaptação, também chamada função de *fitness*, que indica uma "nota de avaliação" a cada indivíduo (cromossomo) de uma população, facilitando a identificação da melhor solução indicada pelo algoritmo (seleção do melhor indivíduo).

O operador de cruzamento tem como principal função a de combinar os cromossomos dos pais para gerar os cromossomos dos filhos. Para isso, o algoritmo escolhe aleatoriamente alguns pontos de cruzamento, copiando tudo o que vem antes desse ponto de um dos pais, e copiando também tudo o que vem depois desse ponto do outro pai, produzindo assim uma nova combinação (um novo cromossomo).

Na busca por uma maior diversidade de soluções, uma fase de mutação é necessária para que se garanta a diversidade genética da população, alterando arbitrariamente um ou mais componentes da estrutura escolhida (cromossomo), produzindo assim novos elementos na população. A mutação procura contornar o problema de ótimos locais, possibilitando alterar levemente a direção da busca. A mutação é aplicada aos indivíduos com uma probabilidade dada pela taxa de mutação.

Apesar do algoritmo genético não contemplar todas as possíveis combinações para

se atingir um resultado ótimo, o que consumiria muito tempo de processamento, seu uso justifica-se pela possibilidade de encontrar soluções aceitáveis em casos de problemas de elevado grau de complexidade matemática ou com grande número de soluções possíveis.

Na implementação do algoritmo genético, portanto, devem ser especificadas as seguintes informações: tamanho da população; taxa de mutação; taxa de cruzamento; operador de seleção e número de gerações, além da função de fitness. Estudos realizados em (CATARINA; BACH, 2003) concluíram que o tamanho da população não deve ser muito grande, pois faz o algoritmo trabalhar por um período de tempo maior, por outro lado, não deve ser muito pequeno para não causar diminuição do espaço de busca da solução pretendida. Para aplicação do algoritmo genético de codificação binária (usada neste trabalho), a taxa de mutação deve ser baixa, entre 1 e 5%, evitando que a busca se torne essencialmente aleatória. O número de gerações não deve ser muito alto, o que provocaria substituição da maior parte da população, tornando a execução do algoritmo muito lenta.

Para a implementação do algoritmo genético neste trabalho será utilizada a GAlib (*Genetic Algorithm Library*), uma biblioteca de funções para a linguagem C++, disponível em <ftp://lancet.mit.edu/pub/ga>.

Neste trabalho o algoritmo genético será utilizado com propósito de aumentar a densidade das classes existentes nas bases de dados. Para isso, cada indivíduo (cromossomo) consistirá em uma representação binária, indicando permanência (valor 1) ou retirada (valor 0) de cada termo existente na respectiva base de dados. A cada nova geração, o algoritmo genético submetirá à função de *fitness* (função de avaliação) novas combinações de permanência e retirada de termos, até que se obtenha o conjunto de características que melhoram a densidade das classes, ou seja, que aproximam os documentos dos centroides de suas respectivas classes.

2.2.5.3 Combinação de Retirada de Termos Mais Raros e de Termos Mais Comuns

Uma técnica utilizada para seleção de características busca identificar as possíveis combinações de retirada de termos que poderiam aperfeiçoar os resultados da classificação.

Dessa forma, a experimentação combinatória verifica as diversas combinações de remoção de termos presentes nas extremidades do *ranking* de pesos dos termos da base. Para cada combinação foram removidas uma determinada porcentagem dos termos mais raros (com baixa frequência na base de dados) e uma porcentagem dos termos mais comuns (mais frequentes) contidos na base de dados. A combinação com melhor resultado de classificação é selecionada. Ressalte-se que os percentuais aplicados em cada combinação não podem totalizar ou ultrapassar o valor de 100%, que seria o equivalente a remover todos os termos, o que descaracterizaria a base de dados.

Capítulo 3

Trabalhos Relacionados

No capítulo anterior foram apresentados os conceitos básicos que norteiam a classificação automática de documentos, sendo possível compreender como representar os documentos num espaço multi-dimensional. Foram também abordados os conceitos de modelo vetorial, dando-se noção de como comparar similaridade entre documentos.

Neste capítulo, serão demonstrados recentes trabalhos relacionados à moderação automática ou temas similares que serviram como guia para o desenvolvimento desta dissertação.

3.1 Moderação Automática de Comentários

Há poucas referências a trabalhos diretamente relacionados à moderação automática de comentários. No entanto, encontramos na literatura o trabalho desenvolvido em (BUBACK, 2011), que realizou experimentos com o mesmo *corpus* utilizado neste trabalho, denominado Globo-*Comments*, com 657405 comentários pré-classificados manualmente por especialistas humanos. No trabalho, foram utilizados métodos de extração de atributos *bag of words* e *n-grams* com *n* máximo de 3 (três) para o tratamento dos textos. Os experimentos comparam

os resultados obtidos com os classificadores SVM, *BoostTexter* e *Naive-Bayes*, concluindo que o SVM com *tri-grams* mostrou ser o classificador mais indicado, dentre os utilizados. Apesar disso, o maior *recall* obtido foi de 36,94%. Com este resultado, o autor conclui que a probabilidade de um comentário indesejado ser aprovado é de 63,06%, o que tornaria a filtragem de comentários adotada em seu trabalho não muito eficiente para este *corpus*.

3.2 Filtros Anti-*Spam*

Trabalhos similares ao desenvolvido nesta dissertação tratam de filtros *anti-spam* para classificação de conjuntos de mensagens que devam ou não ser consideradas como de conteúdo malicioso. O trabalho desenvolvido em (SHRIVASTAVA; BINDU, 2014) utilizou algoritmo genético para extração de termos que melhor identificam o conteúdo *spam*, obtendo-se alta eficiência nos resultados alcançados, com acurácia acima de 82%.

Neste mesmo contexto, em (POURHASHEMI; OSAREH; SHADGAR, 2013) foram utilizados métodos de seleção de características em duas etapas. Na primeira etapa foram extraídos *stopwords* e termos pouco referenciados. Na segunda etapa de filtragem foi aplicado o método *chi-square* para extração de termos mais relevantes no *corpus* considerado. Comparações foram feitas com uso dos classificadores DMNB (*Discriminative Multinomial Nayve Bayes*), MNB (*Multinomial Nayve Bayes*), SVM (*Support Vector Model*) e *Random Forest*, concluindo que a combinação de seleção de características e o uso de um classificador apropriado aumentam os índices da classificação, além de melhorar seu desempenho.

Ressaltamos também o trabalho desenvolvido em (BASAVARAJU; PRABHAKAR, 2010), que utilizou técnicas de lematização (*Porter Stemmer*) no tratamento dos textos da base, e comparou técnicas de classificação utilizando os algoritmos de agrupamento (*clustering*) K*-Means* e *BIRCH* (*Balanced Iterative Reducing and Clustering using Hierarchies*) com os classificadores $KNNC$ e NNC (variantes do classificador KNN), concluindo que o uso do

algoritmo *BIRCH* com o classificador $KNNC$ obteve maior acurácia quando utilizada na classificação de *corpus* com grande número de documentos.

3.3 Análise de Sentimentos

De acordo com (PANG; LEE, 2008), mais e mais pessoas estão deixando suas opiniões disponíveis para estranhos terem conhecimento através da *Internet*. Este fenômeno tem despertado interesse de empresas que queiram obter informações a respeito de seus produtos e serviços, de usuários que se interessam em saber informações relevantes sobre produtos que possam vir a consumir, e até mesmo de pessoas que queiram saber o que está sendo falado sobre elas.

A análise de sentimentos objetiva classificar as opiniões das pessoas sobre determinado tema, identificando a existência de polaridade positiva ou negativa nos textos em análise. Há casos também em que a análise torna-se mais ampla, podendo identificar sentimentos como medo, felicidade, angústia, tristeza, etc.

Para fins de revisão literária com tema compatível com o proposto nesta dissertação, foram analisados trabalhos que abordam identificação de apenas duas polaridades de sentimentos (duas categorias), já que a moderação automática busca classificar os comentários em apenas duas categorias (aprovados ou reprovados para divulgação).

Assim sendo, o trabalho desenvolvido em (DUARTE, 2013) compara resultados de medidas de acurácia média, *recall* e *precision* para a classificação de sentimentos, de um total de 300 mil comentários em língua portuguesa, extraídos da rede social *Twitter*, que continham o verbo sentir e suas diferentes conjugações (presente, passado e futuro do modo indicativo). Duas classes foram consideradas na classificação: positiva e negativa. Para extração de características foram utilizados o SentiLex, uma ferramenta para identificação de polaridade positiva ou negativa de textos em português, e a estratégia de negação *Bigrams* encontrado

em (PAK; PAROUBEK, 2010). Os algoritmos usados para classificação foram *Naive Bayes*, *Decision Tree*, SVM e KNN. Os resultados mostraram que, para distinguir comentários positivos e negativos, deve-se utilizar o SentiLex ou uma combinação de SentiLex e negação Bigrams com uso do classificador SVM, que obteve acurácia média em torno de 68,05%.

Capítulo 4

Aprendizado Supervisionado

Há certos problemas que as técnicas convencionais de programação não são capazes de resolver. Classificar um texto, por exemplo, identificando a categoria à qual ele pertence, pode exigir alguma apresentação prévia de elementos (caracteres e textos), que descrevam padrões de identificação individual, para posterior reconhecimento. Com este formato de resolução de problemas, é possível descrever uma metodologia de aprendizado.

Segundo (KODRATOFF; MICHALSKI, 1990), a aprendizagem de máquina é uma subárea da Inteligência Artificial, cujas técnicas procuram projetar e desenvolver algoritmos que identificam padrões presentes em bases de dados fornecidas como entrada.

A pesquisa desenvolvida em (MITCHELL, 1997) propõe a construção de algoritmos de aprendizagem de máquina que possam aprender com a experiência. Dessa forma, para a solução dos problemas, busca-se encontrar uma função que mapeie os dados de entrada aos de saída através de exemplos fornecidos. Esta forma de aprendizagem denomina-se aprendizagem supervisionada. Há também a aprendizagem não-supervisionada, cujos únicos dados de entrada fornecidos são os documentos da coleção, não sendo fornecidos nem mesmo rótulos de classes. Neste caso, a tarefa do classificador é separar os documentos em grupos, ou classes, sendo um procedimento comumente conhecido como *clustering* (agrupamento).

Mitchell (1997) afirma que a aprendizagem de máquina possui grande valor prático para uma variedade de domínios de aplicações, sendo especialmente úteis nos seguintes casos:

- em problemas de Mineração de Dados (*Data Mining*), onde grandes bancos de dados são analisados automaticamente, existe uma busca de regularidades implícitas que possam ser úteis;
- em domínios ainda pouco entendidos onde os humanos não possuem o conhecimento necessário para desenvolver algoritmos efetivos;
- em domínios onde o programa necessita adaptar-se dinamicamente a mudanças; e
- em domínios em que o custo da aquisição ou codificação manual do conhecimento é muito custosa.

Algoritmos de aprendizado são utilizados como forma de se obter o conhecimento através dos exemplos (base de dados) fornecidos.

Neste trabalho, a classificação automática de comentários segue o método de aprendizagem supervisionado, de tal forma que cada comentário, fornecido na base de dados, é pré-classificado na categoria à qual pertence. Através da comparação entre as saídas desejadas e as fornecidas pela classificação automática será possível avaliar o desempenho do modelo de classificação.

4.1 Conjunto de treinamento, validação e teste

De maneira geral, o processo de classificação automática é dividida em três fases distintas: aprendizagem, validação e classificação.

A aprendizagem é realizada com base em um conjunto de treinamento composto por documentos pré-classificados. A validação ocorre quando ajustes são realizados através

de métricas de desempenho, avaliando-se parâmetros das técnicas através da seleção dos valores que apresentarem melhor resultado, com uso da base de treino. Por fim, a base de testes é composta por um grupo de documentos externos à base de treino, sendo submetida às técnicas de classificação, como forma de medir a efetividade destas.

4.1.1 Validação Cruzada (*Cross-Validation*)

O método *Cross-Validation* (Validação Cruzada) tem se tornado um método padrão para garantir a validação estatística de resultados de classificação (SEBASTIANI, 2002). É largamente encontrado na literatura, e destaca-se por sua simplicidade e robustez (MITCHELL, 1997).

Segundo (KOHAVI et al., 1995), este método consiste na construção de k diferentes classificações: ψ_1, ψ_2, ..., ψ_k. Como forma de validar cada classificação, o conjunto de documentos D_t pertencentes à base de dados, com um total de N_t documentos, é dividido em k grupos mutuamente exclusivos, chamados *folds* (partições) de tamanhos: N_{t_1}, N_{t_2}, ..., N_{t_k}. A classificação ψ_i utiliza a i-ésima partição (*fold*), de tamanho N_{t_i}, como base de teste e os documentos restantes de D_t, de tamanho N_t - N_{t_i}, como base de treino.

Cada classificação é avaliada de forma independente, usando alguma métrica de avaliação. A validação cruzada é realizada através do cálculo da média das k medidas de avaliação obtidas em cada classificação. O processo da validação cruzada está ilustrado na Figura 4.1. Nesta ilustração, a base de dados foi dividida em 4 partições, representadas por 4 quadrados.

São realizadas 4 avaliações, numeradas de 1 a 4, correspondendo a um processo de classificação realizado com uma das partições para conjunto de teste (quadrado hachurado) e as demais partições para conjunto de treino. Ao final, é obtida a média dos resultados obtidos nas 4 avaliações.

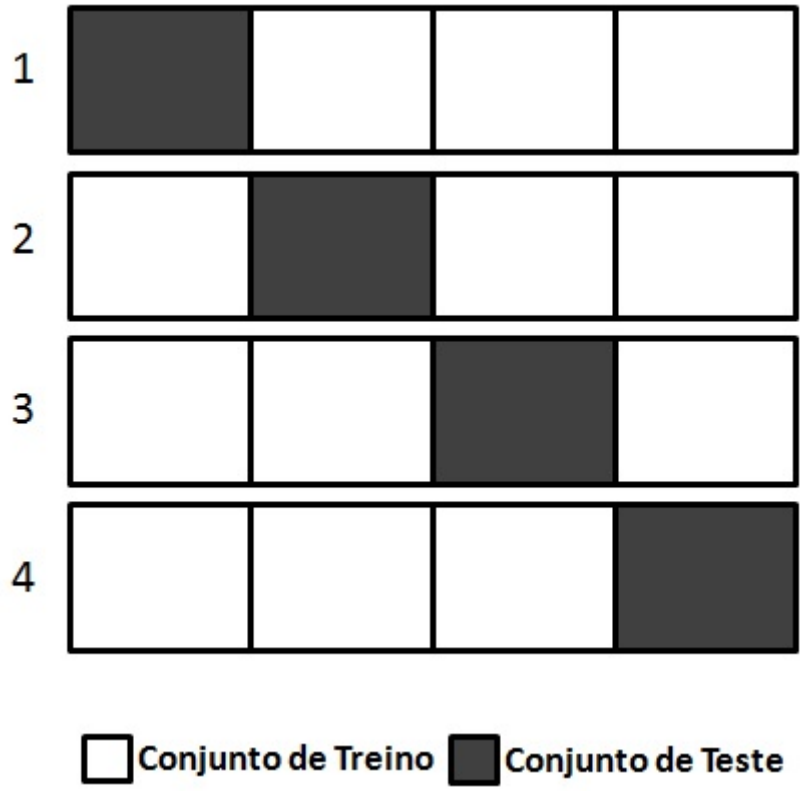

Figura 4.1: Validação cruzada para 4 partições

4.2 Algoritmos de Classificação

Em (CHEN; HAN; YU, 1996) é afirmado que o processo de classificação envolve a construção de um modelo para que sejam aplicados os dados ainda não classificados, visando categorizá-los numa classe pré-definida. Esse processo de classificação é realizado baseado nos padrões reconhecidos no conjunto de dados da base de treinamento.

Os algoritmos de aprendizagem possuem diferentes características. Nesta seção serão descritos os algoritmos de classificação selecionados para compor os experimentos realizados neste trabalho.

4.2.1 O Classificador KNN (K-*Nearest Neighbors*)

O classificador KNN, do inglês K-*Nearest Neighbors* (“K vizinhos mais próximos”), é um método baseado na analogia. O conjunto de treino é formado por vetores n-dimensionais

e cada elemento deste conjunto representa um ponto no espaço n-dimensional. Sendo assim, dado um documento de teste d_t, o método KNN realiza as seguintes atividades para classificá-lo (SHAKHNAROVICH; INDYK; DARRELL, 2006):

- a distância entre o documento d_t e cada um dos documentos de treino é calculada utilizando alguma medida de distância ou similaridade entre documentos, tal como a medida de similaridade cosseno, adotada neste trabalho, e descrita na seção 2.1.1;
- os K documentos de treino mais próximos, ou seja, mais similares do documento d_t são selecionados;
- o documento d_t é classificado em determinada categoria de acordo com algum critério de agrupamento das categorias dos K documentos de treino selecionados na etapa anterior. Em geral, são observadas quais são as classes desses K vizinhos mais próximos, e o documento d_t será classificado como pertencente à classe mais frequente.

Para melhor compreender o processo de classificação do KNN, é ilustrada uma demonstração na Figura 4.2, sendo um espaço vetorial de duas dimensões (duas palavras), três classes, e dois documentos de teste, cuja classificação pretende-se obter. Os documentos de teste serão classificados através do cálculo de similaridade com os 7 vizinhos mais próximos.

Analisando a classe predominante dos 7 vizinhos mais próximos, o documento teste desconhecido X será classificado como pertencente à classe B, enquanto o documento teste desconhecido Y será classificado como pertencente à classe A.

Apesar de ser um método simples e de fácil implementação, o uso do classificador KNN pode ser um processo computacionalmente complexo para casos em que o conjunto de treino é relativamente grande.

A escolha deste classificador justifica-se por ser um método amplamente utilizado em experimentos que envolvem recuperação de informação. Apesar de sua simplicidade, seus

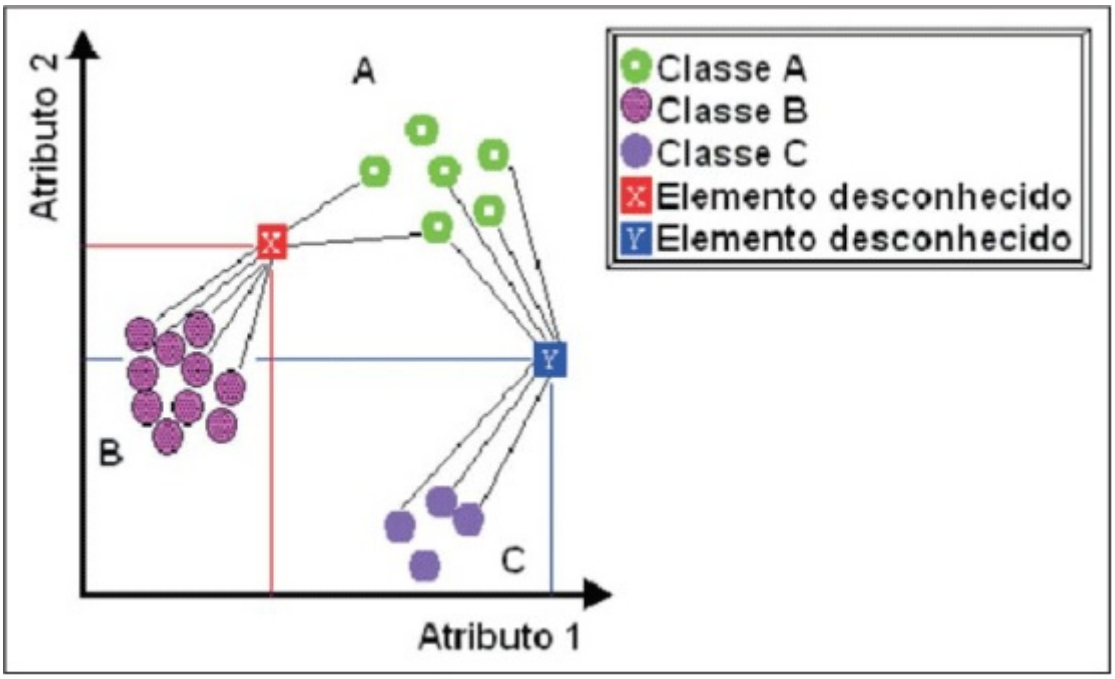

Figura 4.2: Ilustração hipotética da classificação de dois documentos (X e Y) através do classificador KNN em um espaço vetorial de dois termos (duas dimensões), com K = 7.

resultados alcançam bom desempenho em diferentes cenários (YANG; LIU, 1999) (COVER; HART, 1967).

4.2.2 O Classificador CBC (*Centroid Based Classifier*)

O classificador CBC (*Centroid-Based Classifier*) (HAN; KARYPIS, 2000) é baseado na ideia de interpretar a base de treino de cada classe como se fosse apenas uma única amostra. Para cada categoria de documentos de treino C_p, contendo m documentos, é calculado um centroide c_p através da média dos pesos de cada termo k_i pertencente aos documentos da categoria C_p, conforme demonstrado na Equação 4.1.

$$c_p = \frac{1}{m} \sum_{j \in C_p} w_{i,j} \tag{4.1}$$

onde $w_{i,j}$ é o peso do termo k_i da j-ésima amostra da classe C_p.

Após os cálculos dos centroides de cada classe, cada documento da base de teste é classificado de acordo com a maior proximidade ao centroide de uma determinada categoria, sendo atribuída sua classificação a esta categoria.

A escolha deste classificador justifica-se por ser de execução rápida, tanto para treinar quanto para testar os documentos.

4.2.3 O Classificador SVM (*Support Vector Machine*)

O classificador SVM (*Support Vector Machine*) é uma técnica de aprendizado de máquina introduzida pela primeira vez por (VAPNIK; CORTES, 1995), sendo utilizada pela primeira vez em problemas de categorização de documentos por Joachims (1999). De maneira geral, o método constitui uma abordagem geométrica para o problema da classificação. Tomando-se como exemplo um *corpus* com duas classes C_a e C_b, a técnica busca encontrar uma superfície de decisão (hiperplano) que pode ser usada como separador dos elementos das classes. O hiperplano é obtido na fase de aprendizagem, através dos dados da base de treino, e divide o espaço em duas regiões, de tal forma que os documentos da classe C_a estejam em uma região e os documentos da classe C_b estejam na outra região. Num espaço bi-dimensional, o hiperplano é uma linha. Num espaço tri-dimensional esse hiperplano é um plano. Após a obtenção do hiperplano, um novo documento d_j pode ser classificado pela sua posição relativa ao hiperplano (BAEZA-YATES; RIBEIRO-NETO, 2011). Na Figura 4.3 é apresentada uma ilustração de um hiperplano que separa elementos de duas classes.

Seja x um documento da base de treino, e duas classes linearmente separáveis C_a e C_b. Segundo (ABE, 2010), cada documento receberá um rótulo: y = +1 se $x \in C_a$, e y_i = -1 se $x \in C_b$. Na Equação 4.2 é mostrada a fórmula geral da função de decisão linear, onde w é um vetor m-dimensional (pesos), b é o termo independente, m representa a dimensionalidade dos dados.

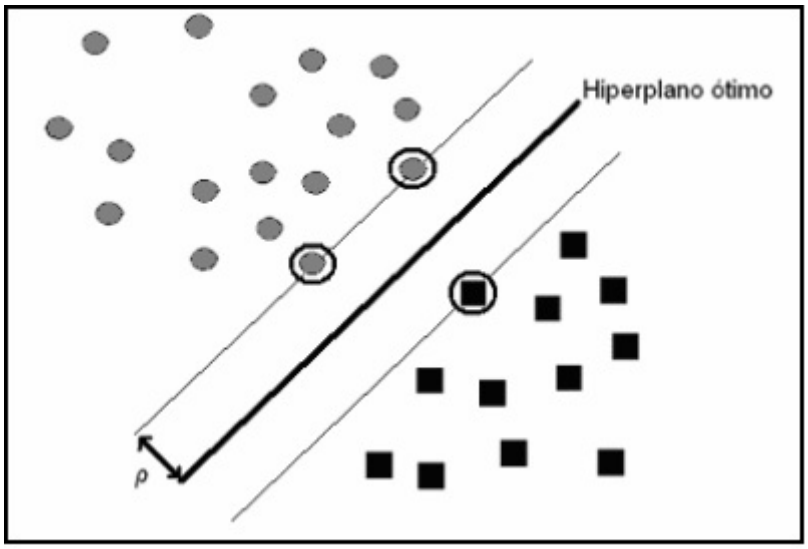

Figura 4.3: O hiperplano ótimo, que separa os documentos com a margem máxima ρ. Adaptado de (ABE, 2010).

$$D(x) = \sum_{i=1}^{m} w_i x + b \tag{4.2}$$

Na Equação 4.3 é representada a função equivalente à demonstrada na Equação 4.2, porém como um produto interno entre dois vetores.

$$D(x) = w^T x + b \tag{4.3}$$

Seja M o número de documentos da base de treino, e w e x os vetores representados por w_i e x_i para i = 1,...,m. Para garantir que os elementos da base de treino sejam linearmente separáveis, seus elementos devem satisfazer às restrições mostradas nas Equações 4.4.

$$\begin{aligned} w^T x + b > 0, x \in C_a(y = +1) \\ w^T x + b < 0, x \in C_b(y = -1) \end{aligned} \tag{4.4}$$

Estas desigualdades podem ser combinadas, obtendo-se a condição exibida na Equação

4.5, para i = 1, 2, . . . , M.

$$y_i(w^T x_i + b) \geq 1 \tag{4.5}$$

Portanto, o hiperplano que forma a superfície de separação entre as duas classes é obtido através da Equação 4.6.

$$D(x) = w^T x + b = c, para - 1 < c < 1 \tag{4.6}$$

Para c = 0, a equação define um hiperplano situado à meia distância entre os dois hiperplanos nos extremos c = +1 e c = -1. A distância entre estes dois hiperplanos extremos é denominada “margem”, e está indicada na Figura 4.3 por ρ. A região entre os dois hiperplanos extremos é chamada “região de generalização”. O hiperplano $D(x)$ = 0, ao maximizar o valor da margem, maximiza a região de generalização, sendo, portanto, considerado um hiperplano ótimo (Figura 4.3).

Basicamente, a tarefa principal de aprendizagem do classificador SVM é encontrar parâmetros w e b que maximizem a região de margem de separação entre as classes (CHERKASSKY; MULIER, 2007).

A teoria de classificação do SVM é principalmente baseada nas seguintes considerações (YONG-FENG; YAN-PING, 2004):

- minimizar o risco estrutural, de modo a torná-lo eficiente;
- maximizar a distância entre classes, encontrando o hiperplano de classificação ótimo, que pode ser garantido por um teorema da estatística;
- utilizar funções *kernel* para trabalhar num espaço de maior dimensão, porém linear.

A escolha deste classificador justifica-se por ser o método adotado em (BUBACK, 2011), facilitando as comparações dos resultados obtidos na classificação. Além disso, em classificação de documentos, este método têm apresentado bom desempenho, sendo um dos

algoritmos com maior precisão obtida empiricamente em diferentes trabalhos na literatura (SEBASTIANI, 2002). O critério de escolha do tipo de *kernel* e seus parâmetros são um ponto fraco do algoritmo, pois ainda são muito baseados em experiências humanas.

4.3 Métricas para avaliação dos resultados

Para a classificação será utilizada a medida de similaridade cosseno, apresentada na seção 2.1.1, nas comparações entre documentos ou entre documentos e centroides das classes, conforme o caso.

As métricas adotadas para a classificação foram as medidas de avaliação *Recall* (Equação 4.7), *Precision* (Equação 4.8) e F1-*measure* (Equação 4.9) (SEBASTIANI, 2002);(YANG; LIU, 1999).

$$Recall(C_p) = \frac{TP(C_p)}{TP(C_p) + FN(C_p)} \tag{4.7}$$

$$Precision(C_p) = \frac{TP(C_p)}{TP(C_p) + FP(C_p)} \tag{4.8}$$

$$F1 - measure(C_p) = \frac{2Precision(C_p)Recall(C_p)}{(Precision(C_p) + Recall(C_p))} \tag{4.9}$$

onde TP (*True Positive*) é a quantidade de documentos atribuídos corretamente à classe C_p pelo classificador automático, FP (*False Positive*) é a quantidade de documentos atribuídos incorretamente à classe C_p pelo classificador automático e FN (*False Negative*) é a quantidade de documentos pertencentes à classe C_p e classificada incorretamente pelo classificador automático como pertencente à outra classe.

A métrica F1-*measure* realiza a média harmônica entre as medidas *Recall* e o *Precision*.

Capítulo 5

Experimentos

Nos capítulos anteriores, foram apresentadas as técnicas a serem utilizadas para o tratamento dos textos contidos nos comentários das bases de dados. Foram também explicadas as técnicas para validação e teste dos classificadores, bem como o funcionamento dos algoritmos de classificação adotados neste trabalho.

Neste capítulo, serão apresentadas as bases de dados objetos de estudo deste trabalho. Primeiro, será apresentada a ferramenta utilizada para execução dos classificadores. Em seguida serão mostrados os resultados obtidos na aplicação dos algoritmos de seleção de características. Ao final, serão apresentadas as conclusões extraídas por meio das análises sobre os dados observados nos experimentos.

5.1 *Bases de Dados*

Para a realização dos testes, as técnicas de seleção de características e de classificação automática serão aplicadas em duas bases de dados distintas. Ambas as bases referem-se a comentários de usuários sobre notícias publicadas no site <http://g1.globo.com/> (Portal de Notícias Globo), cedidos pela empresa Globo Comunicação e Participações S.A.. Todos

os comentários foram pré-classificados nas categorias de Aprovados ou Reprovados para divulgação, através de um trabalho manual realizado por especialistas da referida empresa.

Os comentários pertencem a domínios variados do conhecimento, tais como política, esporte, economia, entre outros. Entretanto, não há uma separação explícita desses domínios nas bases de dados fornecidas.

Em seguida serão detalhadas mais informações sobre a distribuição dos documentos entre as classes de cada base de dados, bem como suas caracterizações.

5.1.1 Base Globo-*Comments* 01

A primeira base, chamada aqui Globo-*Comments* 01, contém 978 comentários. A distribuição de comentários entre as classes da base está demonstrada na Tabela 5.1.

Classe	Número de Documentos
Aprovado	539
Reprovado	439

Tabela 5.1: Número de documentos em cada classe da base Globo-*Comments* 01

Esta base foi submetida a um pré-processamento, que consistiu na retirada dos *stopwords* e na extração de sufixos (*stemming*), conforme técnicas especificadas nas seções 2.2.1 e 2.2.3. Após a redução dos termos aos seus respectivos radicais, a base ficou com um total de 4434 termos. As medidas de caracterização desta base foram processadas, conforme especificado na seção 4.3, e estão demonstradas na Tabela 5.2.

MSDC	MSCCP	MSPC	Razão (MSPC/MSDC)
0,0805	0,9138	0,6701	6,2255

Tabela 5.2: Caracterização da base Globo-*Comments* 01

As informações contidas na Tabela 5.2 demonstram que os comentários de uma mesma categoria encontram-se espacialmente bem separados em virtude do baixo valor médio de

similaridade entre os documentos das classes com seus centroides (medida MSDC). Como o valor de MSCCP é alta, próxima do valor máximo 1, podemos concluir que os centroides das classes estão muito próximos do centroide principal. O valor obtido para MSPC indica que as classes encontram-se relativamente próximas, ocasionando um alto índice da Razão. Dessa forma, no espaço vetorial, os documentos de cada categoria estão bem espalhados, mas existem documentos de categorias diferentes que estão próximos, ocorrendo um alto índice de sobreposição, o que dificulta a classificação de novos comentários. Nesta base serão aplicados experimentos, cujos resultados darão direção para a estratégia a ser montada para classificação automática numa base de maior tamanho, ou seja, a base Globo-*Comments* 02.

5.1.2 Base Globo-*Comments* 02

A base Globo-*Comments* 02 contém 657.405 comentários. A distribuição de comentários entre as classes da base está demonstrada na Tabela 5.3.

Classe	Número de Documentos
Aprovado	573.821
Reprovado	83.584

Tabela 5.3: Número de documentos em cada classe da base Globo-*Comments* 02

Foram, então, processados esses dados removendo-se primeiro os *stopwords*, em seguida todas as palavras foram submetidas à extração de sufixos com uso do RSLP, restando assim um total de 213.795 termos na base.

As medidas que caracterizam esta base de dados estão demonstradas na Tabela 5.4.

MSDC	MSCCP	MSPC	Razão (MSPC/MSDC)
0.0805	0.9833	0.9338	11.5973

Tabela 5.4: Caracterização da base Globo-*Comments* 02

Semelhantemente ao exposto na seção 5.1.1, os valores mostrados na Tabela 5.4 nos permite concluir que os comentários de uma mesma categoria encontram-se espacialmente bem separados em virtude do baixo valor de MSDC. Como valor de MSCCP é alto, muito próximo do máximo, podemos dizer que os centroides das classes estão próximos do centroide principal. O alto valor indicado em MSPC indica que as classes estão sobrepostas, causando uma alta taxa da razão MSPC/MSDC. É possível, portanto, concluir que os comentários de ambas as categorias encontram-se espacialmente misturados, o que dificulta a classificação de novos comentários.

5.2 Ferramentas de classificação

Para a execução dos algoritmos de classificação, utilizando técnicas de validação cruzada (*cross-validation*), foram utilizadas as ferramentas desenvolvidas por Souza (2014). Ao todo, foram desenvolvidas duas ferramentas: uma para pré-processamento da base de dados e a outra para a etapa de classificação propriamente dita. Ambas as ferramentas foram implementadas em linguagem de programação C++.

5.2.1 Pré-processamento

Na etapa de pré-processamento, três informações básicas são necessárias: a classe, o documento e as palavras. A classe é representada pela pasta de arquivos onde os documentos estão armazenados, ou seja, documentos de uma mesma classe devem estar dispostos separadamente em uma mesma pasta de arquivos. Com essa informação, a ferramenta realiza a leitura dos arquivos da base de dados no formato texto original, com o objetivo de extrair as palavras e indexar os documentos. Após a execução desta etapa, o resultado é escrito em cinco arquivos. Três arquivos no formato binário são utilizados para representar a estrutura de índices e os outros dois arquivos, um no formato texto e outro no formato

binário, para armazenar informações complementares, tais como: a lista de palavras encontradas e o número total de documentos. A Figura 5.1 mostra a representação dos arquivos que armazenam as classes, os documentos e os índices.

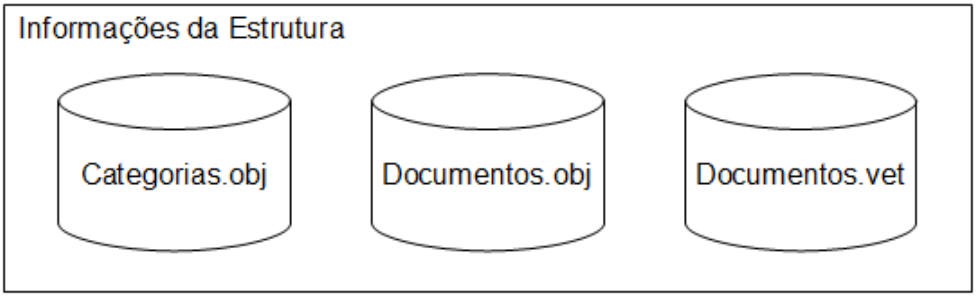

Figura 5.1: Estrutura de arquivos que armazenam os índices (SOUZA, 2014).

O arquivo binário “Documentos.vet” contém o vetor que representa o índice de cada um dos documentos processados. O início e o fim dos índices são marcados por dois ponteiros e cada posição no vetor armazena o identificador único e a frequência da palavra. Palavras com frequência zero (não encontradas) não são armazenadas no vetor. A representação interna do arquivo “Documentos.vet” é mostrada na Figura 5.2.

Figura 5.2: Representação interna do arquivo “Documentos.vet” (SOUZA, 2014).

O arquivo “Documentos.obj” é organizado de maneira similar ao arquivo “Documentos.vet”. A diferença é que, ao invés de palavras, cada posição no arquivo armazena um cabeçalho com dados sobre o documento. Neste cabeçalho, pode ser encontrado o identificador único do documento e um par de ponteiros. Os ponteiros marcam o início e o fim do índice do documento, representado na Figura 5.2.

As informações sobre as classes são armazenadas no arquivo “Categorias.obj”. Para

esse propósito, cada conjunto de documentos lidos na pasta de arquivos gera uma entrada, que irá armazenar os ponteiros que delimitam os documentos pertencentes a cada classe. Esta organização de arquivos é vantajosa, pois permite que dados sejam lidos e transferidos para a memória em tempo linear.

A principal vantagem da aplicação da fase de pré-processamento é obter uma representação composta basicamente por números, ao invés de palavras, simplificando a realização de comparações, ordenações, e outras operações similares.

5.2.2 Sistema de Classificação

Após a etapa de pré-processamento, os índices dos documentos são submetidos a um sistema de classificação. As ferramentas de classificação desenvolvidas em (SOUZA, 2014) foram o KNN e o CBC, já estando implementadas as métricas de avaliação *Recall*, *Precision* e F1-*measure*.

Neste trabalho, implementamos o classificador SVM. Para esta finalidade, foi utilizada uma biblioteca de funções para execução do SVM, denominada libSVM (CHANG; LIN, 2011). A representação UML (*Unified Modeling Language*) das classes que implementam os classificadores, com acréscimo do classificador SVM, está mostrada na Figura 5.3.

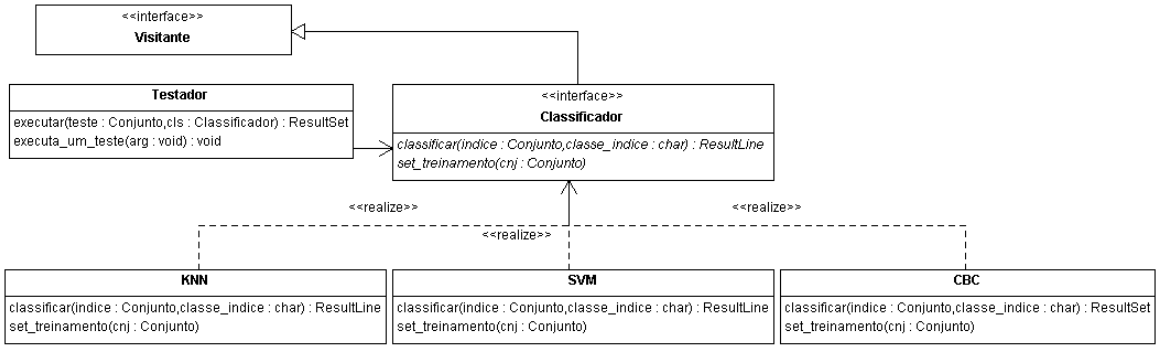

Figura 5.3: Diagrama UML representando os elementos envolvidos na classificação. Adaptado de (SOUZA, 2014).

O “Testador” é iniciado com a parte da base de dados que é utilizada para testes e com

o algoritmo de classificação. Ele é responsável por enviar cada um dos índices da base de testes para classificação. Os componentes do tipo “Classificador” são inicializados com a parte da base de dados que será utilizada para treinamento do algoritmo. Ao receber um índice, eles utilizam as informações dessa base de dados para efetuar a classificação e retornam uma lista ordenada que parte da classe mais relevante para a menos relevante, de acordo com o algoritmo de classificação utilizado. O documento será atribuído à classe mais relevante.

É importante frisar que o sistema de classificação aqui apresentado também permite que os resultados sejam produzidos utilizando métodos diferentes de validação (*holdout* ou k-*fold*).

5.3 Resultados obtidos

Os testes foram conduzidos sobre as bases de dados apresentadas nas Seções 5.1.1 e 5.1.2. Experimentos realizados na base de dados Globo-*Comments* 01 servirão para escolha de uma estratégia a ser aplicada à base de dados Globo-*Comments* 02. Por este motivo, os resultados das classificações com o uso das técnicas de pré-processamento, bem como de redução de dimensionalidade e seleção de características, serão mostrados nesta seção, separadamente, para cada base de dados.

Para a classificação dos comentários foi utilizada a técnica do 10-*fold cross-validation*, onde a base é dividida em 10 (dez) partes e são obtidos 10 resultados diferentes, cada uma sobre uma partição diferente, e as demais partes são usadas para treinamento. Ao final é calculada a média dos índices obtidos, de acordo com o descrito em (KOHAVI et al., 1995).

O classificador KNN será calibrado usando a base de treino como conjunto de dados para obtenção do valor ótimo de K (SEBASTIANI, 2002).

Para utilização do classificador SVM, nos experimentos com as duas bases, foram ado-

tados os mesmos parâmetros de calibração utilizados por (BUBACK, 2011), ou seja, *kernel* linear e parâmetro de margem c = 20.

5.3.1 Experimentos com a Base Globo-*Comments* 01

Os experimentos com a base de dados Globo-*Comments* 01 foram realizados usando os classificadores SVM (*Support Vector Machine*), CBC (*Centroid-based Classification*) e KNN (K-*Nearest Neighbors*).

O pré-processamento da base de dados e as técnicas de seleção de características serão aplicados em etapas para observação e comparação dos resultados de *Recall*, *Precision* e *F1-measure* em cada caso, objetivando adotar a melhor estratégia a ser aplicada posteriormente numa base de dados com maior número de comentários. A sequência de etapas está demonstrada na Figura 5.4.

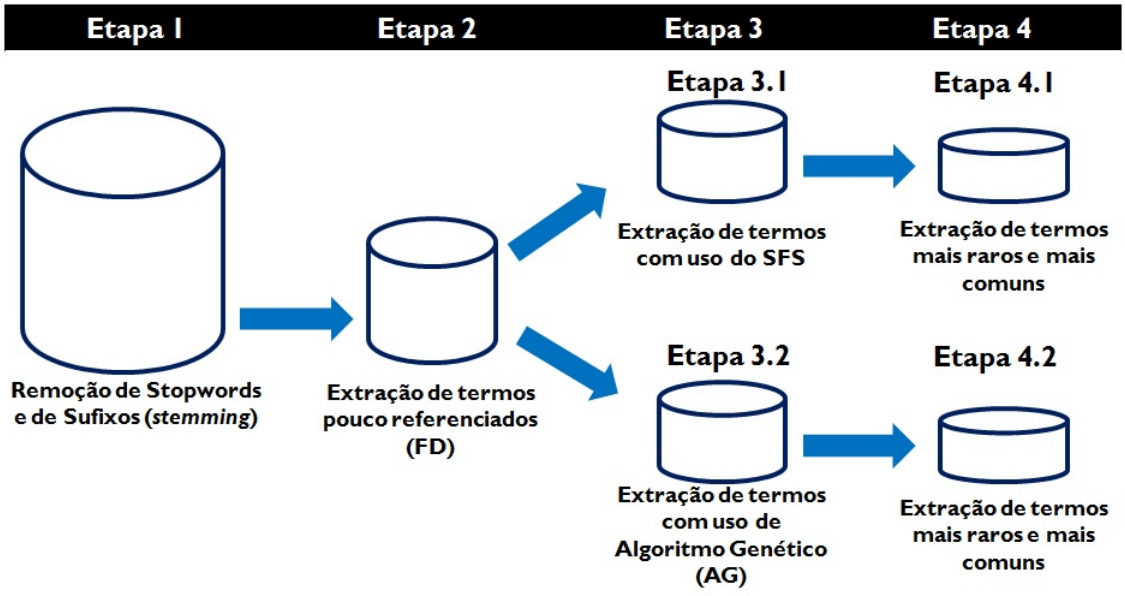

Figura 5.4: Etapas de aplicação das técnicas de redução de dimensionalidade e seleção de características sobre a base de dados Globo-*Comments* 01.

Serão, portanto, aplicadas duas sequências de técnicas. A primeira sequência iniciando na etapa 1 e seguindo pelas etapas 2, 3.1 e 4.1 da Figura 5.4. A segunda sequência iniciando também na etapa 1 e seguindo pelas etapas 2, 3.2 e 4.2 da Figura 5.4. Basicamente, a

diferenças entre as sequências de aplicação de técnicas está na etapa 3, com a utilização do algoritmo de seleção de características SFS na etapa 3.1, e utilização de algoritmos genéticos para seleção de características na etapa 3.2.

A seguir, serão demonstrados o valor de K escolhido para utilização do classificador KNN, bem como os resultados obtidos em cada uma das etapas demonstradas na Figura 5.4 aplicadas à base Globo-*Comments* 01.

5.3.1.1 Calibração do valor de K para o classificador KNN

Para aplicação do classificador KNN, a base de dados foi submetida a um processo de calibração que consistiu em comparar a média da medida F1-*measure* para um conjunto variado de valores de K. Na Figura 5.5 é exibido um gráfico comparativo para a calibração e escolha do valor de K.

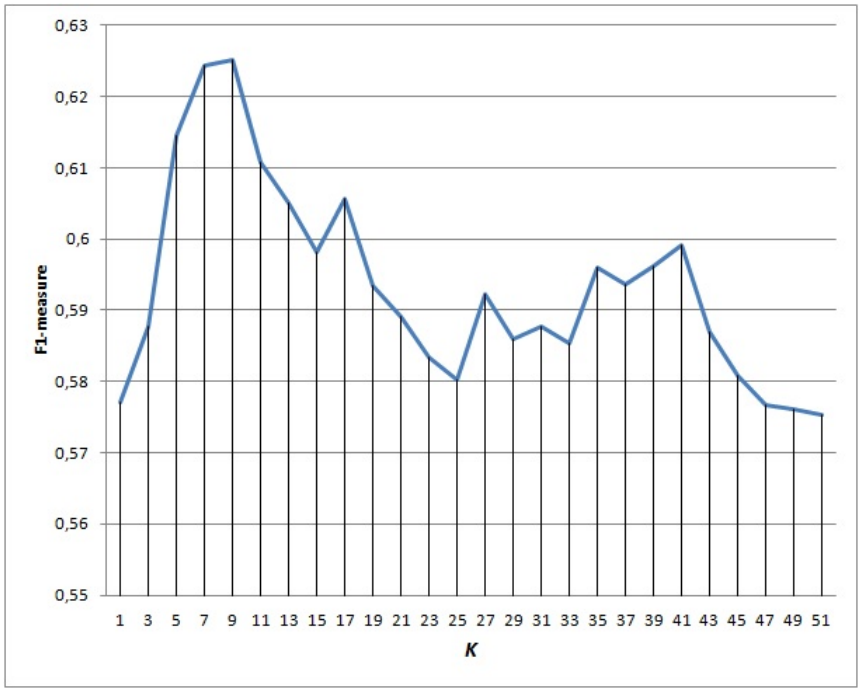

Figura 5.5: Escolha do valor de K para aplicação do classificador KNN na base de dados Globo-*Comments* 01. O valor de K foi selecionado com o objetivo de aumentar o valor do F1-*measure*.

O valor de K que indicou o melhor índice, dentre os testados nesta base de dados, foi de

K = 9, sendo, portanto, este o valor escolhido.

5.3.1.2 Etapa 1 - Remoção de *stopwords* e de sufixos dos termos (*stemming*)

A primeira etapa de classificação (etapa 1) foi executada com a base no estado inicial de pré-processamento, com um total de 4434 termos. Os dados obtidos para classificação com a base de dados neste estado inicial são mostrados na Tabela 5.5.

Classificador	***Recall***	***Precision***	***F1-Measure***
SVM	0.6075	0.6211	0.6000
CBC	**0.6235**	**0.6333**	**0.6284**
KNN	0.6189	0.6316	0.6252

Tabela 5.5: Resultados da classificação (etapa 1) - Extração de *stopwords* e extração de sufixos dos termos com uso do RSLP. Melhor resultado para cada métrica em negrito.

5.3.1.3 Etapa 2 - Remoção de termos pouco referenciados (FD)

A seguir, na etapa 2, aplicou-se a técnica de retirada de termos presentes em menos de 2 (dois) documentos (Frequência de Documentos), reduzindo o número de termos para 1726 termos. Apesar de não trazer melhorias significativas nas taxas de classificação dos três classificadores, a retirada de termos pouco referenciados reduziu o tempo de processamento da classificação em razão da expressiva redução de termos de 4434 para 1726. Os resultados da classificação nesta etapa 2 são mostrados na Tabela 5.6.

5.3.1.4 Etapa 3.1 - Seleção de características com uso do algoritmo SFS

Em busca de tentar melhorar a classificação, foram usadas três estratégias para seleção de características. A estratégia de SFS (etapa 3.1), buscando maximizar a métrica F1-*measure*, foi usada até selecionar 528 termos. Selecionando mais termos, o valor de

Classificador	*Recall*	*Precision*	*F1-Measure*
SVM	**0.6258**	0.6327	0.6221
CBC	0.6250	**0.6352**	**0.6301**
KNN	0.6243	0.6326	0.6284

Tabela 5.6: Resultados da classificação (etapa 2) - Extração de termos pouco referenciados (FD). Melhor resultado para cada métrica em negrito.

F1-*measure* apresentava um decréscimo. Os resultados da classificação nesta etapa estão mostrados na Tabela 5.7.

Classificador	*Recall*	*Precision*	*F1-Measure*
SVM	0.7606	0.7764	0.7684
CBC	**0.8271**	**0.8336**	**0.8303**
KNN	0.7603	0.8025	0.7808

Tabela 5.7: Resultados da classificação (etapa 3.1) - Seleção de características com SFS após etapa 2. Melhor resultado para cada métrica em negrito.

É possível observar uma melhora significativa dos resultados para os três classificadores, especialmente para o classificador CBC. No entanto, é importante ressaltar que a execução do algoritmo SFS consome longo tempo de processamento, uma vez que este algoritmo procura testar praticamente todas as possibilidades de combinações dos termos existentes.

5.3.1.5 Etapa 3.2 - Seleção de características com uso de algoritmos genéticos

Uma outra estratégia utilizada neste trabalho para melhorar a classificação buscou selecionar os termos que produzem aumento na densidade de cada categoria da base, uma vez que a proximidade dos comentários de uma mesma categoria aos seus respectivos centroides diminui o efeito da sobreposição entre os comentários, favorecendo a classificação automática de novos comentários.

A Figura 5.6 ilustra o efeito desta estratégia. Nesta ilustração, são representados dois conjuntos como circunferências, indicativas de duas categorias de uma determinada base de dados. Uma das categorias, contém elementos (comentários) representados por estrelas, e a outra categoria contém elementos (comentários) representados por triângulos. Os centroides de cada categoria estão representados pelos quadrados. Na Figura 5.6.a, os elementos de cada categoria estão mais espalhados em relação aos seus respectivos centroides, havendo assim maior sobreposição entre elementos de diferentes categorias. Já na Figura 5.6.b, os elementos de uma mesma categoria estão mais próximos dos seus respectivos centroides, diminuindo assim o efeito da sobreposição.

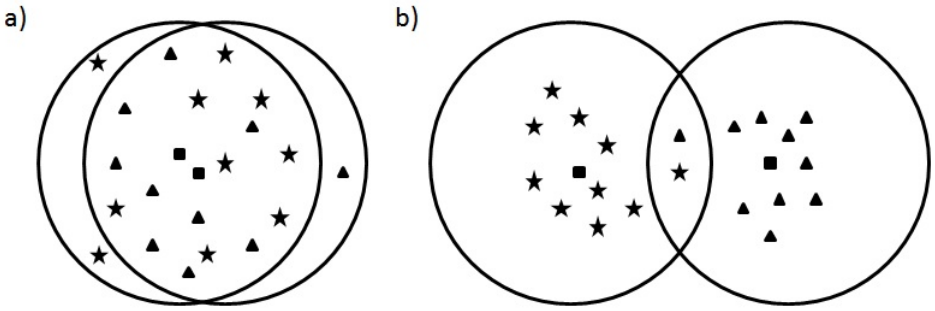

Figura 5.6: Representação gráfica de densidades das categorias de uma base

Testar todas as possibilidades de combinações de termos que produzam aumento na densidade de cada categoria da base de dados é uma tarefa que exige muito tempo de processamento, sendo praticamente inviável sua execução. Portanto, foi utilizada a técnica de Algoritmo Genético para extração deste conjunto de termos que produza o efeito ilustrado na Figura 5.6.b.

Sendo assim, duas abordagens foram adotadas. A primeira abordagem implica em separar os termos apresentados no centroide principal da base de dados. A meta é remover os termos que impedem que as categorias estejam espacialmente mais separadas. Para isso, foi utilizada uma configuração de cromossomo binário para representação do centroide principal, de tal forma que o valor 1 significa a presença de um termo, enquanto o valor 0 significa que o termo deve ser removido. A Equação 5.1 (ORRIOLS-PUIG; MACIA; HO, 2010) foi utilizada para avaliar as diferentes combinações de termos.

$$intra/inter\_class = \frac{\sum_{i=1}^{|N|} intraSimil(d_i)}{\sum_{i=1}^{|N|} interSimil(d_i)} \tag{5.1}$$

onde $|N|$ é o número de documentos na base de dados. Esta medida compara as similaridades dos comentários no interior da classe com as similaridades aos vizinhos mais próximos das outras classes. Para cada comentário d_i, é calculada a similaridade ao seu vizinho mais próximo dentro da classe (interSimil(d_i)) e a similaridade de seu vizinho mais próximo de qualquer outra classe (intraSimil(d_i)).

A função *fitness* do Algoritmo Genético seleciona o menor valor da medida calculada, uma vez que quanto menor a medida, mais próximos os comentários serão agrupados em suas respectivas classes, e mais separadas as classes devem estar umas das outras.

Os parâmetros do Algoritmo Genético adotados nos experimentos foram: tamanho da população igual a 200; taxa de mutação de 5%; e número de gerações igual a 50.

Aplicando-se esta técnica na base de dados, após ser processada até a etapa 2, 1099 termos foram removidos, permanecendo 627 termos na base de dados. Os resultados obtidos neste experimento estão apresentados na Tabela 5.8.

Classificador	***Recall***	***Precision***	***F1-Measure***
SVM	0.6616	0.6764	0.6689
CBC	**0.7086**	0.7108	**0.7097**
KNN	0.6622	**0.7197**	0.6897

Tabela 5.8: Resultados da classificação - Seleção de características com Algoritmo Genético (após etapa 2), tendo como função *fitness* o menor valor da relação obtida na Equação 5.1. Melhor resultado para cada métrica em negrito.

É possível observar que houve uma pequena evolução nos índices de classificação em relação à etapa 2. Esse resultado pode ser explicado pela dificuldade em se encontrar uma combinação de termos desprezando-se o contexto (a categoria) em que cada termo está inserido. É possível, por exemplo, que um termo apontado como não relevante para a base,

de uma forma geral, seja relevante para uma classe específica. A métrica aumentou a densidade dos documentos em cada classe, o que justifica a melhoria da classificação, no entanto, retirou termos dos centroides observando suas relevâncias no contexto geral da base, e não no contexto específico em cada categoria.

Isto posto, foi realizada uma nova tentativa de extração de termos mais relevantes, substituindo a função de *fitness* apresentada na classificação anterior por uma que aumente a similaridade de cada documento ao centróide de sua respectiva classe. A Equação 5.2 foi utilizada como função *fitness* para seleção de termos em cada classe, separadamente, onde d_i é o i-ésimo documento da base, $cent_{C_p}$ é o centroide da classe C_p e $\mid N_{C_p} \mid$ é o número de classes da base.

$$Density(C_p) = \frac{\sum_{i=1}^{|N_{C_p}|} cos(di, cent_{C_p})}{|N_{C_p}|} \tag{5.2}$$

Com esta abordagem, o total de termos retirados foi de 1016, sendo mantidos um total de 710 termos. Os resultados deste experimento estão apresentados na Tabela 5.9.

Classificador	***Recall***	***Precision***	***F1-Measure***
SVM	0.7606	0.7764	0.7684
CBC	**0.8184**	**0.8263**	**0.8224**
KNN	0.7603	0.8025	0.7808

Tabela 5.9: Resultados da classificação (etapa 3.2) - Seleção de características com Algoritmo Genético após etapa 2, tendo como função *fitness* o maior valor da relação obtida na Equação 5.2 em cada Categoria da base de dados (etapa 3.2). Melhor resultado para cada métrica em negrito.

Tendo em vista que os resultados demonstrados na Tabela 5.9 superaram os demonstrados na Tabela 5.8, a métrica indicada na Equação 5.2 foi escolhida para aplicação do Algoritmo Genético (etapa 3.2).

Ressalte-se que ambas as estratégias das etapas 3.1 (SFS) e 3.2 (Genético) foram apli-

cadas à base de dados a partir da etapa 2, ou seja, após serem aplicadas à base de dados a retirada de *stopwords*, extração de sufixos dos termos com uso do RSLP e retirada de termos referenciados em menos de 2 documentos (FD).

5.3.1.6 Etapa 4 - Combinações de remoção dos termos mais raros e mais comuns

Por último, foi aplicada a extração de termos mais raros e termos mais comuns, através da combinação de percentuais de retiradas de termos pertencentes a estes extremos no ranking de pesos dos termos da base, buscando-se a combinação que produzisse aumento na métrica F1-*measure*. Assim, a etapa 4.1 realizou combinações de retirada de termos raros e comuns após a seleção de características aplicada com o SFS (etapa 3.1), enquanto a etapa 4.2 realizou a mesma tarefa, porém após a seleção de características aplicada com o Algoritmo Genético (etapa 3.2).

A aplicação da extração de termos mais raros e de termos mais comuns após a aplicação da seleção de características com o SFS (etapa 4.1) não encontrou uma combinação que provocasse aumento no desempenho da classificação de nenhum dos três classificadores utilizados nos experimentos. Já a aplicação desta técnica na etapa 4.2 (após aplicação do Algoritmo Genético - etapa 3.2), obteve os resultados apresentados na Tabela 5.10 com a extração de 4% de termos mais comuns, restando um total de 681 termos.

Classificador	***Recall***	***Precision***	***F1-Measure***
SVM	0.7986	0.8164	0.8074
CBC	0.8385	0.8448	0.8417
KNN	**0.8556**	**0.8791**	**0.8672**

Tabela 5.10: Resultados da classificação (etapa 4.2) - Seleção de características com extração de termos mais raros e de termos mais comuns da base de dados após etapa 3.2. Melhor resultado para cada métrica em negrito.

Com o classificador SVM, a aplicação do Algoritmo Genético conjugado com a extração de 4% dos termos mais comuns produziu um aumento nas medidas de desempenho em

relação aos obtidos nas etapas anteriores. Os dados desta classificação foram analisados, sendo observado que, em média, 78,45% dos comentários que deveriam ser reprovados realmente foram reprovados pela técnica proposta.

O uso do classificador CBC, seguiu o mesmo padrão dos resultados obtidos com o Classificador SVM. Os dados desta classificação foram analisados, sendo observado que, em média, 83,09% dos comentários que deveriam ser reprovados realmente foram reprovados pela técnica proposta.

É possível também observar um melhor resultado obtido com o uso da aplicação do Algoritmo Genético seguido da aplicação da extração de termos mais raros e mais comuns, obtendo-se o melhor desempenho na etapa 4.2 com o classificador KNN, se comparado com os índices obtidos com os classificadores SVM e CBC. Os dados desta foram analisados, sendo observado que, em média, 84,79% dos comentários que deveriam ser reprovados realmente foram reprovados pela técnica proposta.

Para tornar a avaliação mais criteriosa, foi aplicado em cada caso o teste não paramétrico de Wilcoxon (DEMŠAR, 2006) pareado com nível de significância de 1% sobre os resultados da medida F1-*measure*. Na avaliação foi percebido que os resultados obtidos para os três classificadores nas etapas 3.1 e 3.2 foram estatisticamente superiores aos obtidos nas etapas 1 e 2. Da mesma forma, os resultados obtidos na etapa 4.2 foram estatisticamente superiores aos obtidos na etapa 3.2. Não houve diferença estatística entre os resultados da etapa 1 e etapa 2, e entre os resultados da etapa 3.1 e etapa 4.1. Finalmente, os resultados obtidos na etapa 4.2 foram estatisticamente superiores aos obtidos na etapa 4.1 com os três classificadores.

Os gráficos das medidas alcançadas em cada passo da estratégia utilizando SFS combinado com a extração de termos raros e de termos comuns estão mostradas nas Figuras 5.7, 5.8 e 5.9, para comparação entre as medidas alcançadas com os classificadores KNN, CBC e SVM, respectivamente. Nestes gráficos, a etapa indicada por “FD” refere-se à classificação da base dados após a extração de termos pouco referenciados, a etapa “SFS” refere-se à

classificação após a extração de termos com uso do algoritmo SFS e a etapa “Combinatorial” refere-se à classificação após a extração de termos raros e de termos comuns.

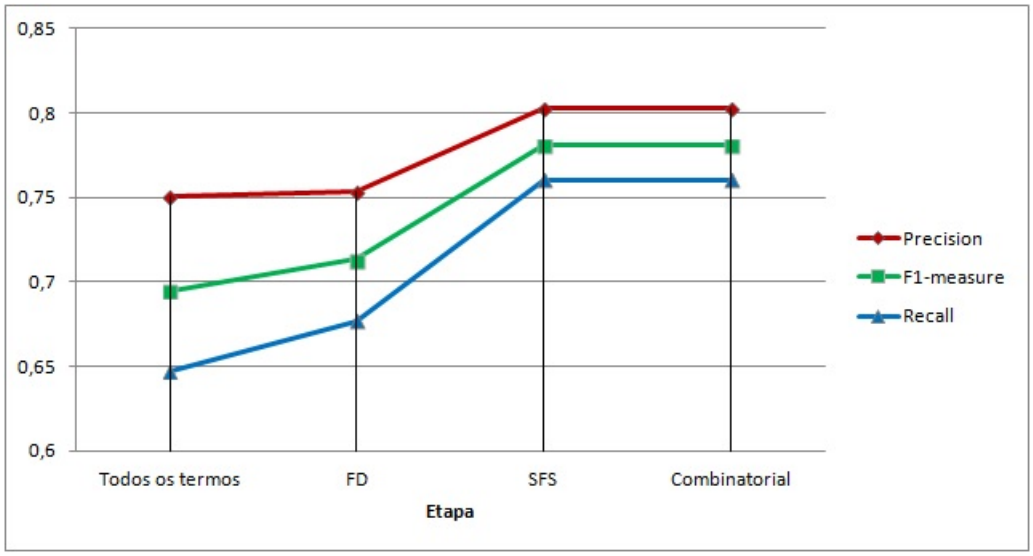

Figura 5.7: Evolução dos resultados das medidas de classificação da base Globo-*Comments* 01, em cada etapa da estratégia usando o SFS, com o classificador KNN

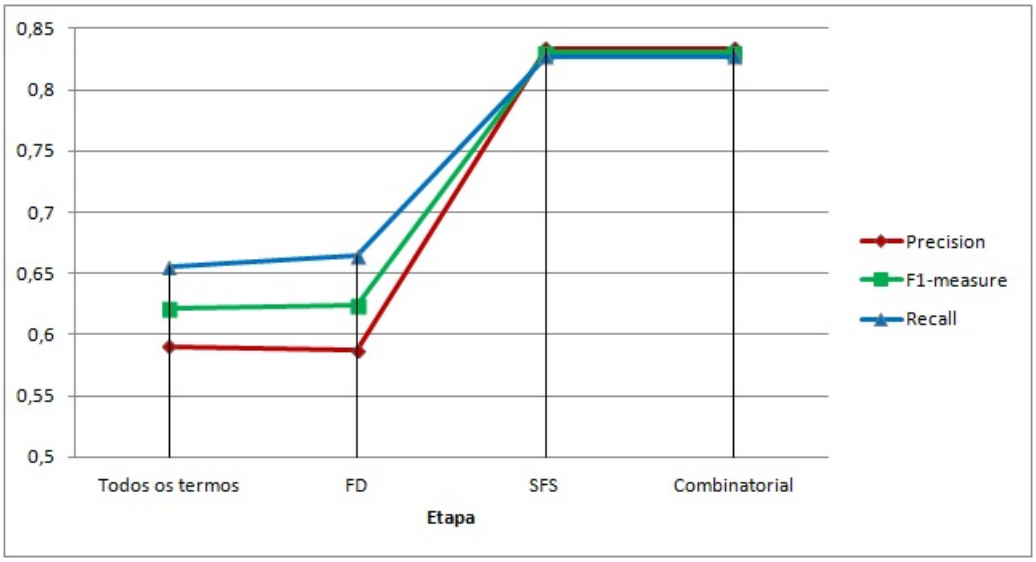

Figura 5.8: Evolução dos resultados das medidas de classificação da base Globo-*Comments* 01, em cada etapa da estratégia usando o SFS, com o classificador CBC

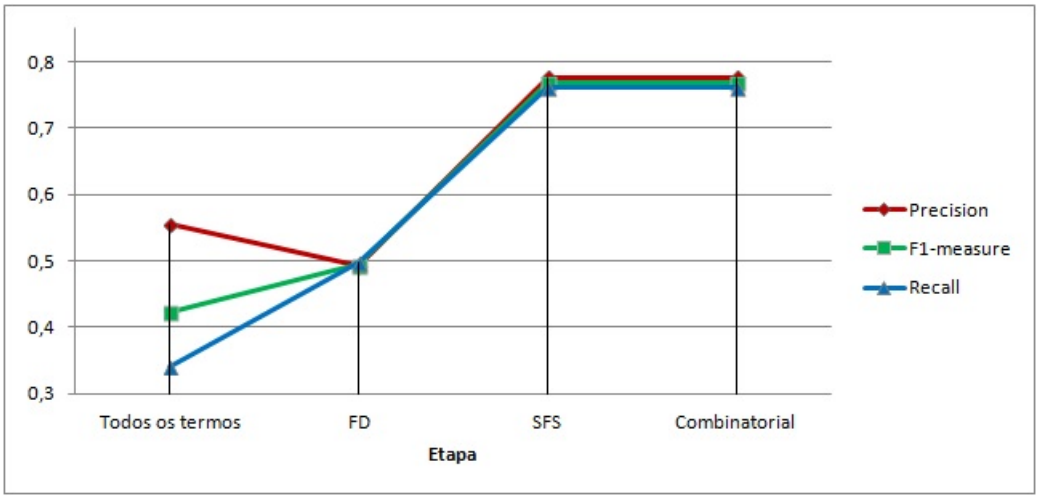

Figura 5.9: Evolução dos resultados das medidas de classificação da base Globo-*Comments* 01, em cada etapa da estratégia usando o SFS, com o classificador SVM

5.3.1.7 Gráficos comparativos dos resultados obtidos pelos experimentos aplicados à base Globo-*Comments* 01

Os gráficos das medidas alcançadas em cada passo da estratégia utilizando Algoritmo Genético combinado com a extração de termos raros e de termos comuns estão mostradas nas Figuras 5.10, 5.11 e 5.12, para comparação entre as medidas alcançadas com os classificadores KNN, CBC e SVM. Nestes gráficos, a etapa indicada por "FD" refere-se à classificação da base dados após a extração de termos pouco referenciados, a etapa "AG01" refere-se à classificação após o uso de Algoritmo Genético buscando a combinação de termos que minimizem o valor $intra/inter\_class$ (Equação 5.1), a etapa "AG02" refere-se à classificação após o uso de Algoritmo Genético buscando a combinação de termos que aumentem o valor da densidade em cada classe (Equação 5.2) e a etapa "Combinatorial" refere-se à classificação após a extração de termos raros e de termos comuns.

A evolução das médias da medida F1-*measure* obtidas com a aplicação dos três classificadores em cada etapa de processamento das técnicas de seleção de características estão demonstradas no gráfico da Figura 5.13.

Analisando os dados da Figura 5.13, é possível observar que a estratégia que apresentou melhor evolução da medida F1-*measure* foi a que utilizou a aplicação de Algoritmo Genético

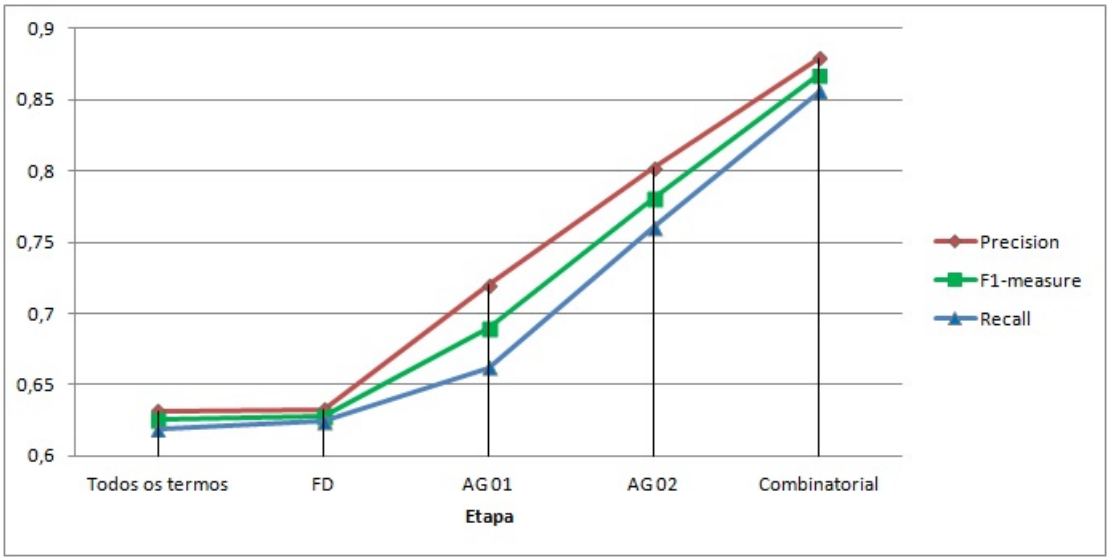

Figura 5.10: Evolução dos resultados das medidas de classificação da base Globo-*Comments* 01, em cada etapa da estratégia usando Algoritmo Genético, com o classificador KNN

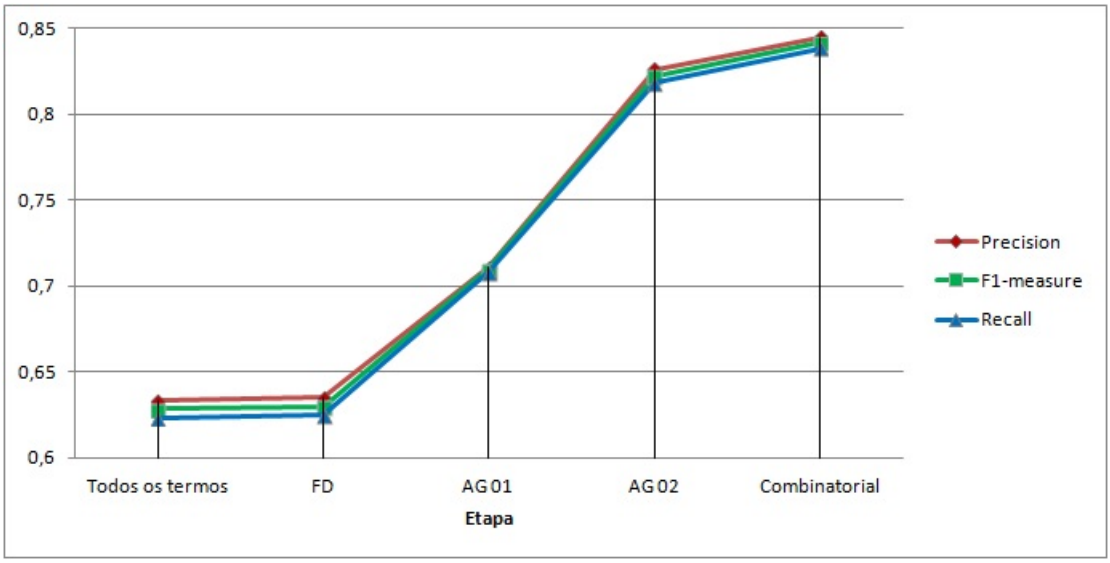

Figura 5.11: Evolução dos resultados das medidas de classificação da base Globo-*Comments* 01, em cada etapa da estratégia usando Algoritmo Genético, com o classificador CBC

combinada com a extração de termos mais raros e mais comuns (SAÚDE et al., 2014b).

Na Figura 5.14 é exibido um gráfico comparativo da evolução da métrica F1-*measure* em função do percentual de remoção de termos da estratégia adotada neste trabalho, para cada um dos classificadores KNN, CBC e SVM, quando usando o algoritmo genético para selecionar os termos.

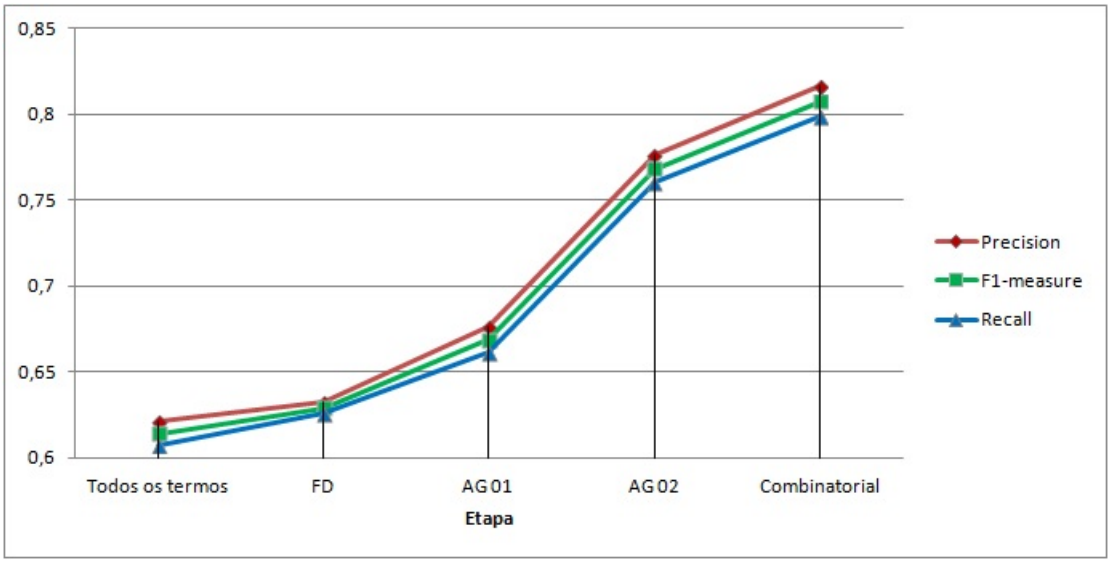

Figura 5.12: Evolução dos resultados das medidas de classificação da base Globo-*Comments* 01, em cada etapa da estratégia usando Algoritmo Genético, com o classificador SVM

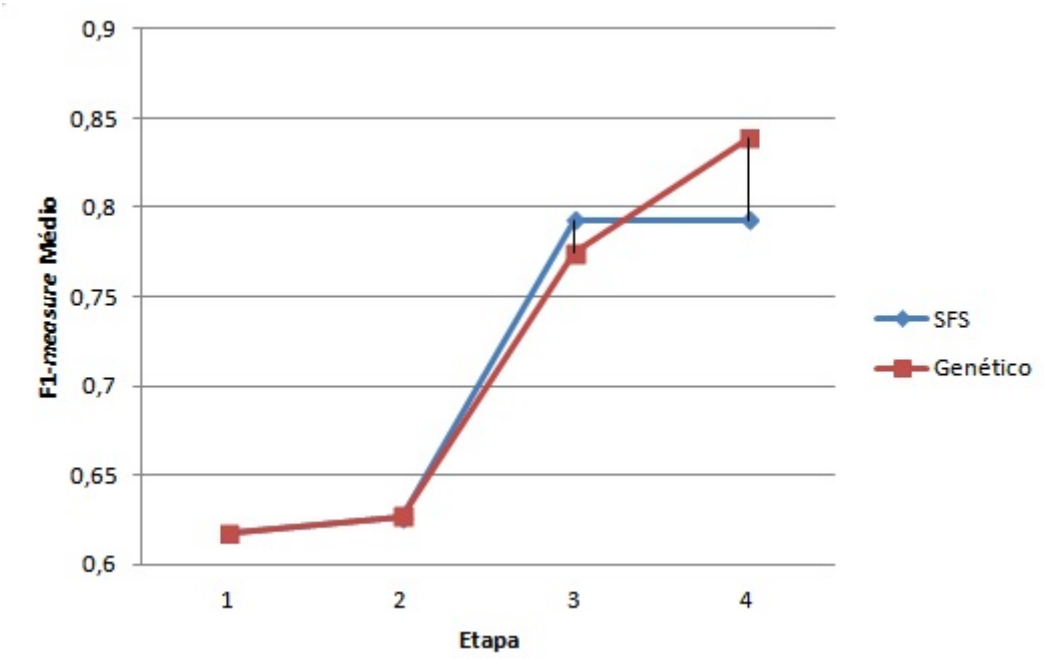

Figura 5.13: Evolução das médias da medida F1-*measure* usando SFS ou Algoritmo Genético em cada etapa

5.3.2 Experimentos com a Base Globo-*Comments* 02

A estratégia apontada pelos experimentos descritos na seção 5.1.1 será utilizada para a base de dados Globo-*Comments* 02, utilizando os classificadores SVM (*Support Vector*

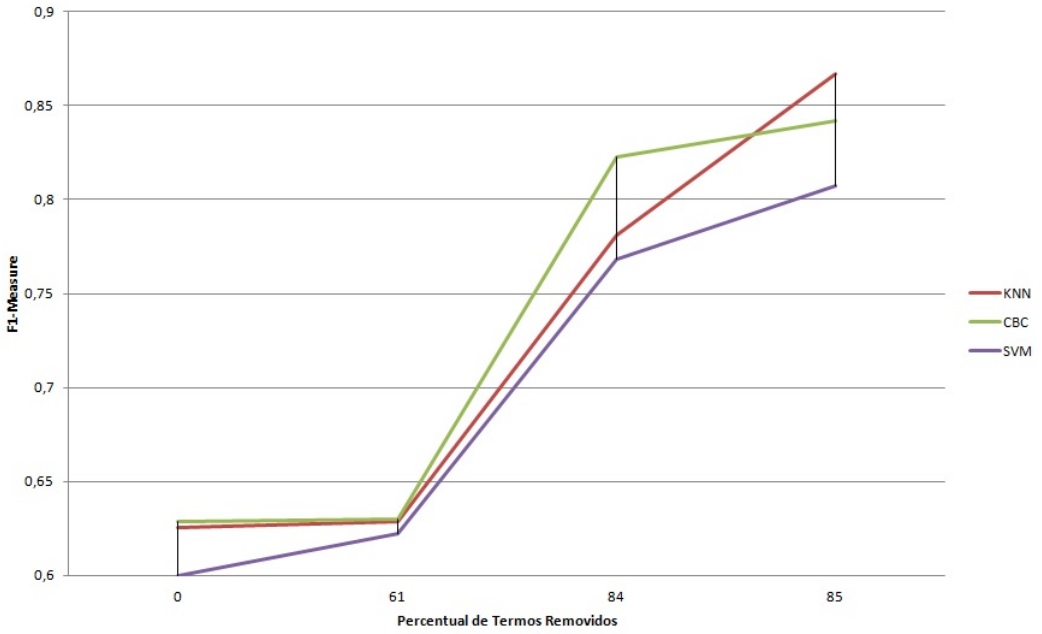

Figura 5.14: Evolução da medida F1-*measure* em função do percentual de termos removidos da base de dados Globo-*Comments* 01

Machine), CBC (*Centroid-based Classification*) e KNN (K-*Nearest Neighbors*).

A sequência de etapas de aplicação das técnicas de redução de dimensionalidade e seleção de características está ilustrada na Figura 5.15.

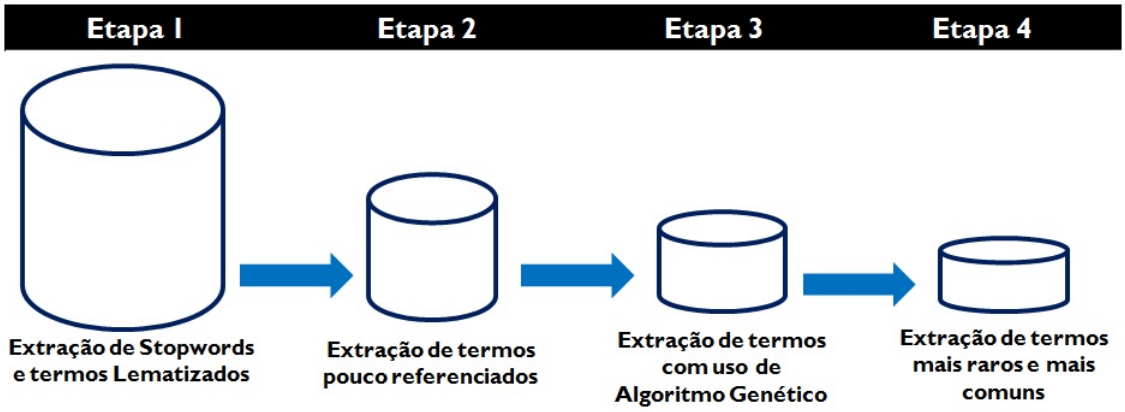

Figura 5.15: Etapas de aplicação das técnicas de redução de dimensionalidade e seleção de características sobre a base Globo-*Comments* 02.

5.3.2.1 Calibração do valor de K para o classificador KNN

Inicialmente, foram testados diversos valores de K para o algoritmo KNN buscando-se aumentar o valor da medida F1-*measure*. O valor de K que apresentou um valor F1-*measure* mais elevado foi K = 5, sendo este o valor escolhido para os experimentos. Um gráfico comparativo desta calibração é mostrado na Figura 5.16, neste gráfico é utilizado valores de K no eixo horizontal com valores da medida F1-*measure* no eixo vertical.

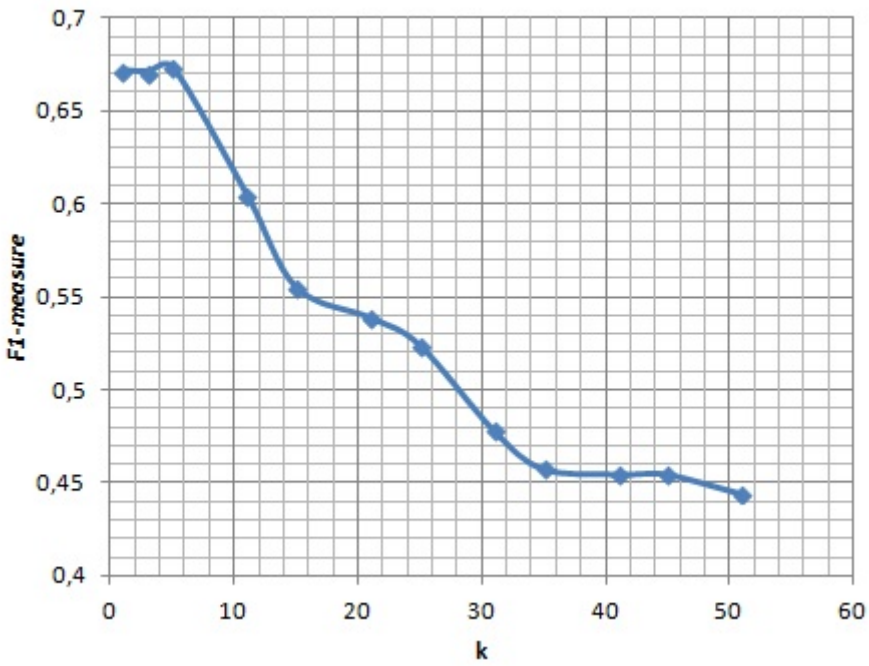

Figura 5.16: Escolha do valor de K para aplicação do classificador KNN na base de dados Globo-*Comments* 02. O valor de K foi selecionado com o objetivo de aumentar o valor do F1-*measure*.

5.3.2.2 Etapa 1 - Remoção de *stopwords* e de sufixos dos termos (*stemming*)

Foi realizado um experimento considerando-se o estado inicial da base de dados, ou seja, com um total de 213,795 termos com sufixos extraídos pelo RSLP. Os resultados são apresentados na Tabela 5.11.

Classifier	Recall	Precision	F1-Measure
SVM	0.3415	0.5561	0.4231
CBC	**0.6551**	0.5903	0.6211
KNN	0.6470	**0.7508**	**0.6951**

Tabela 5.11: Resultados da classificação da base de dados Globo-*Comments* 02(com extração de sufixos de todos os termos, utilizando o RSLP). Melhor resultado para cada métrica em negrito.

5.3.2.3 Etapa 2 - Remoção de termos pouco referenciados (FD)

A seguir, foi aplicada a técnica FD (Frequência de Documentos) para remover termos considerados como ruídos potenciais. Foi escolhido podar os termos referenciados em menos de 0.05% do total de documentos da base de dados. Após essa abordagem, o número de termos, com sufixos extraídos pelo RSLP, foi reduzido de 213,795 para 3,611. Os resultados do experimento da nova classificação com esta base de dados estão mostrados na Tabela 5.12.

Classifier	Recall	Precision	F1-Measure
SVM	0.4974	0.4920	0.4869
CBC	0.6643	0.5876	0.6236
KNN	**0.6771**	**0.7535**	**0.7132**

Tabela 5.12: Resultados da classificação da base de dados Globo-*Comments* 02(utilizando termos com sufixos extraídos pelo RSLP, após o uso da técnica FD). Melhor resultado para cada métrica em negrito.

Apesar de não trazer melhorias consideráveis nas métricas de classificação, a remoção dos termos pela técnica FD foi capaz de reduzir o tempo de processamento da classificação devido a esta redução significativa de termos.

Com um menor número de termos a serem avaliados, o tamanho dos vetores é reduzido e os resultados das classificações KNN, CBC e SVM são obtidas mais rapidamente. Se

um número maior de termos é utilizado, o tempo de análise de cada indivíduo com uso do algoritmo genético é maior. Isto resultaria em um tempo de processamento mais longo (talvez impraticável em alguns casos).

5.3.2.4 Etapa 3 - Seleção de características com uso de Algoritmos Genéticos

Com base nos resultados obtidos até este momento, e da estratégia apontada pelos experimentos realizados com a base Globo-*Comments* 01, o algoritmo SFS não foi aplicado sobre essa base de dados, devido ao seu elevado custo computacional (o que pode ser inviável aplicá-lo sobre essa base) e porque o algoritmo genético foi capaz de fornecer resultados melhores. Por outro lado, as mesmas estratégias aplicadas na base anterior para remoção de termos utilizando Algoritmo Genético foram usadas nesta base de dados.

Uma primeira tentativa do uso do Algoritmo Genético foi selecionar os termos presentes no centroide principal da base de dados que minimzem o valor da medida indicada na Equação 5.1 (medida *intra/inter_class*). Neste experimento, portanto, a função *fitness* seleciona o menor valor da referida medida, tendo em vista que quanto menor esta medida, mais agrupados os documentos estarão dentro da sua respectiva categoria.

Semelhantemente ao aplicado na seção 5.1.1, os parâmetros do Algoritmo Genético foram: tamanho da população igual a 200, taxa de mutação de 5% e número de gerações igual a 50.

Aplicando esta técnica, 1,850 termos foram removidos, restando 1,761 termos na base de dados. A Tabela 5.13 apresenta os resultados obtidos neste experimento.

Houve uma pequena variação nas métricas das classificações, o que confirma o resultado alcançado na seção anterior a respeito do uso dessa medida como função *fitness*.

Um novo experimento foi realizado, desta vez utilizando a medida indicada na Equação 5.2 como a função *fitness* do Algoritmo Genético, com o propósito de extrair os termos que

Classifier	Recall	Precision	F1-Measure
SVM	0.6247	0.6540	0.6390
CBC	0.6425	0.6914	0.6661
KNN	**0.7491**	**0.7879**	**0.7680**

Tabela 5.13: Resultados da classificação da base de dados Globo-*Comments* 02 (após a seleção de termos com Algoritmo Genético usando a Equação 5.1 como função *fitness*). Melhor resultado para cada métrica em negrito.

melhor caracterizam cada categoria, separadamente, da base de dados. Com essa estratégia, foram removidos um total de 1,146 termos, e mantidos um total de 2,465 termos com sufixos extraídos pelo RSLP. Os resultados deste experimento estão apresentados na Tabela 5.14.

Classifier	Recall	Precision	F1-Measure
SVM	0.8134	0.8661	0.8389
CBC	0.8984	0.7636	0.8255
KNN	**0.9192**	**0.9567**	**0.9376**

Tabela 5.14: Resultados da classificação da base de dados Globo-*Comments* 02 (após a seleção de termos com Algoritmo Genético usando a Equação 5.2 como função *fitness*). Melhor resultado para cada métrica em negrito.

Os valores apresentados na Tabela 5.14 mostram que a aplicação de Algoritmo Genético para seleção de termos que melhoram a caracterização de cada categoria, separadamente, mostrou ser uma técnica mais apropriada para a tarefa de classificação.

5.3.2.5 Etapa 4 - Combinações de remoção de termos mais raros e mais comuns

Além disso, a fim de melhorar ainda mais as métricas de classificação, foram realizadas uma série de experiências adicionais, combinando uma remoção simultânea de proporção dos termos mais raros e de termos mais comuns do conjunto de dados. Assim sendo, foi

encontrada uma combinação de extração dos 21% de termos mais raros, e simultaneamente 39% dos termos mais comuns da base de dados, que permitiu aumentar a medida F1-*measure* para 0.9786 para o algoritmo KNN, como mostrado na Tabela 5.15. Foram testadas um total de 5,050 combinações para obter esta combinação bem sucedida. Como resultado, o número de termos foi reduzido para 988 termos.

Classifier	Recall	Precision	F1-Measure
SVM	0.8313	0.8960	0.8624
CBC	0.9163	0.7935	0.8505
KNN	**0.9678**	**0.9897**	**0.9786**

Tabela 5.15: Resultados da classificação da base de dados Globo-*Comments* 02 (após seleção de termos através da experimentação combinatória). Melhor resultado para cada métrica em negrito.

Baseado nos resultados mostrados na Tabela 5.15, é possível concluir que este último experimento foi mais efetivo em gerar melhorias significativas nos indicadores da média geral das métricas de classificação, sendo, no entanto, um processo mais demorado.

5.3.2.6 Gráficos comparativos dos resultados obtidos pelos experimentos aplicados à base Globo-*Comments* 02

Os gráficos das medidas alcançadas em cada passo da estratégia aplicada estão mostradas nas Figuras 5.17, 5.18 e 5.19, para comparação entre as medidas alcançadas com os classificadores KNN, CBC e SVM. Nestes gráficos, a etapa indicada por “FD” refere-se à classificação da base dados após a extração de termos pouco referenciados, a etapa “AG01” refere-se à classificação após o uso de Algoritmo Genético buscando a combinação de termos que minimizem o valor $intra/inter\_class$ (Equação 5.1), a etapa “AG02” refere-se à classificação após o uso de Algoritmo Genético buscando a combinação de termos que aumentem o valor da densidade em cada classe (Equação 5.2) e a etapa “Combinatorial” refere-se à classificação após a extração de termos raros e de termos comuns.

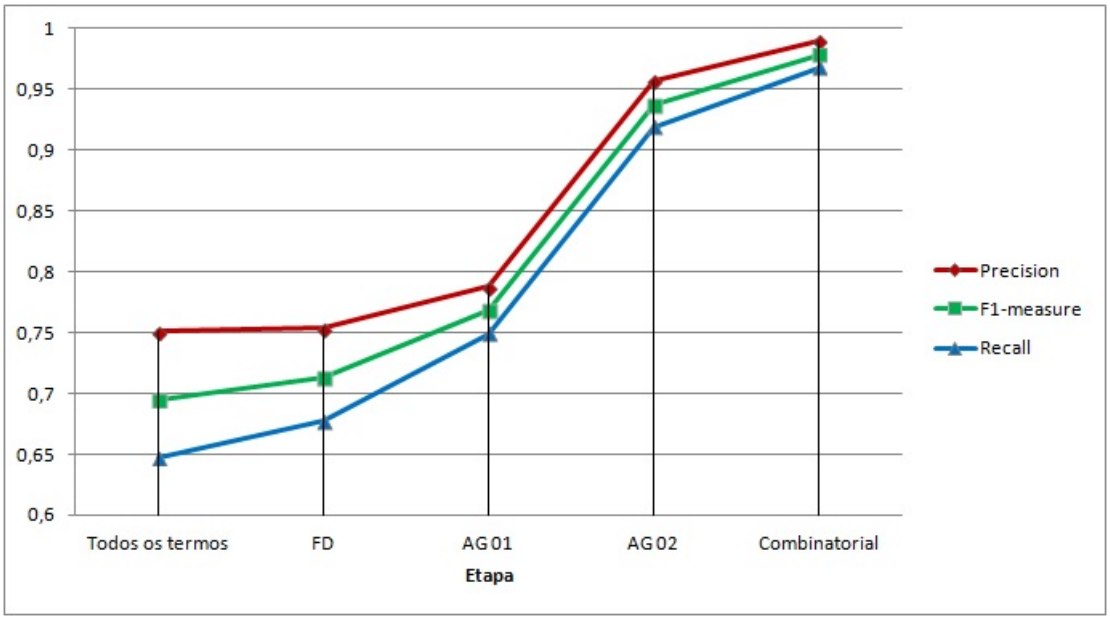

Figura 5.17: Evolução dos resultados das medidas de classificação com o classificador KNN usando diferentes técnicas

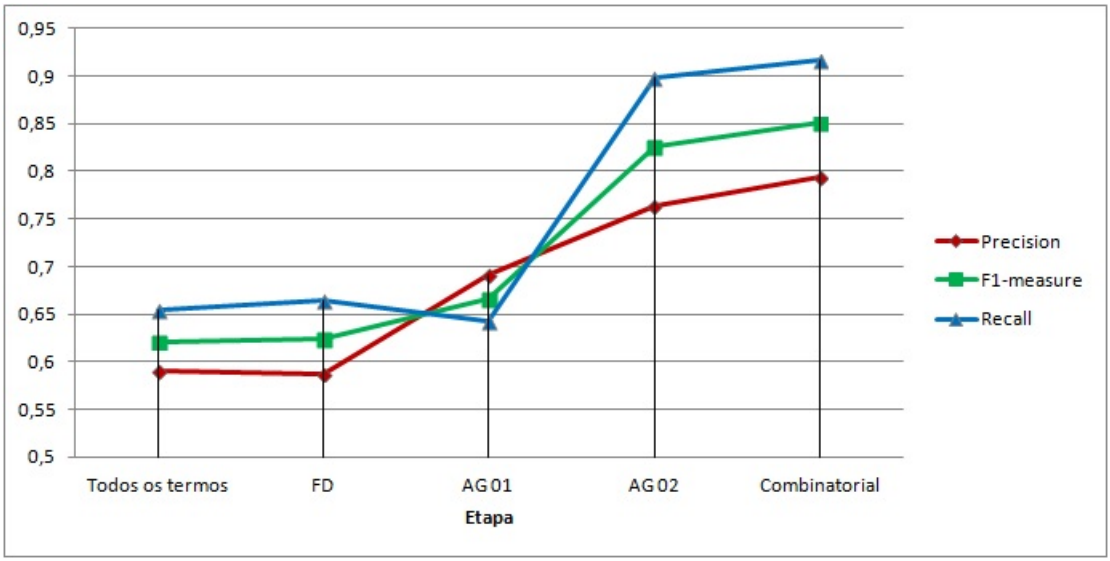

Figura 5.18: Evolução dos resultados das medidas de classificação com o classificador CBC usando diferentes técnicas

Na Figura 5.20 é mostrado um gráfico comparativo da evolução da métrica F1-*measure* em função do percentual de remoção de termos da estratégia adotada neste trabalho, para cada um dos classificadores KNN, CBC e SVM.

Para tornar a avaliação mais criteriosa, nesta base de dados também foi aplicado em cada caso o teste não paramétrico de Wilcoxon (DEMŠAR, 2006) pareado com nível de

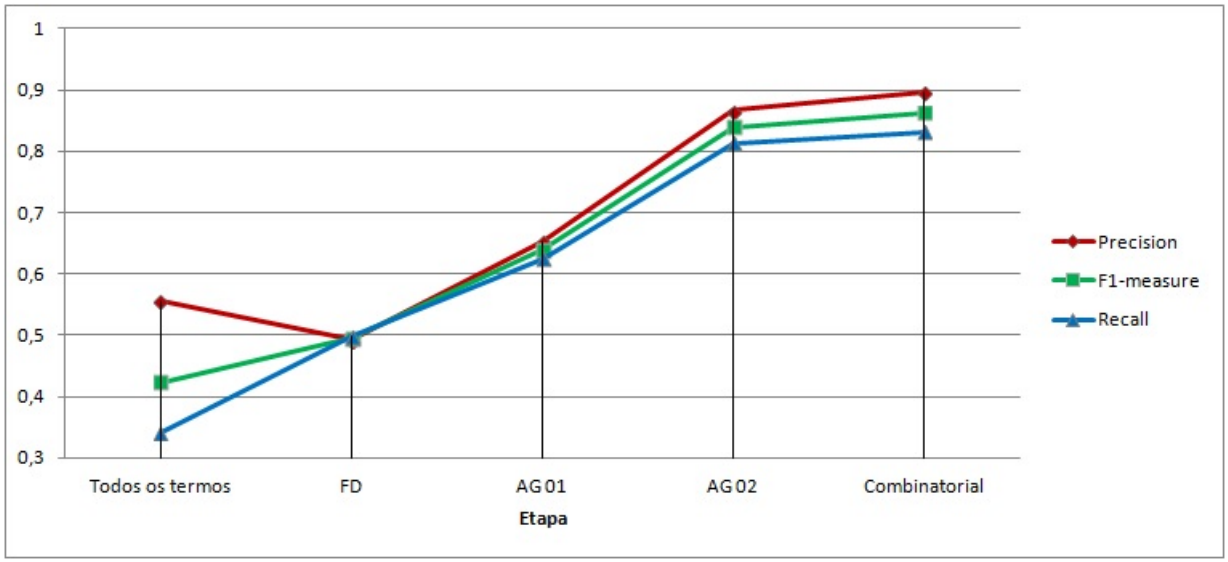

Figura 5.19: Evolução dos resultados das medidas de classificação com o classificador SVM usando diferentes técnicas

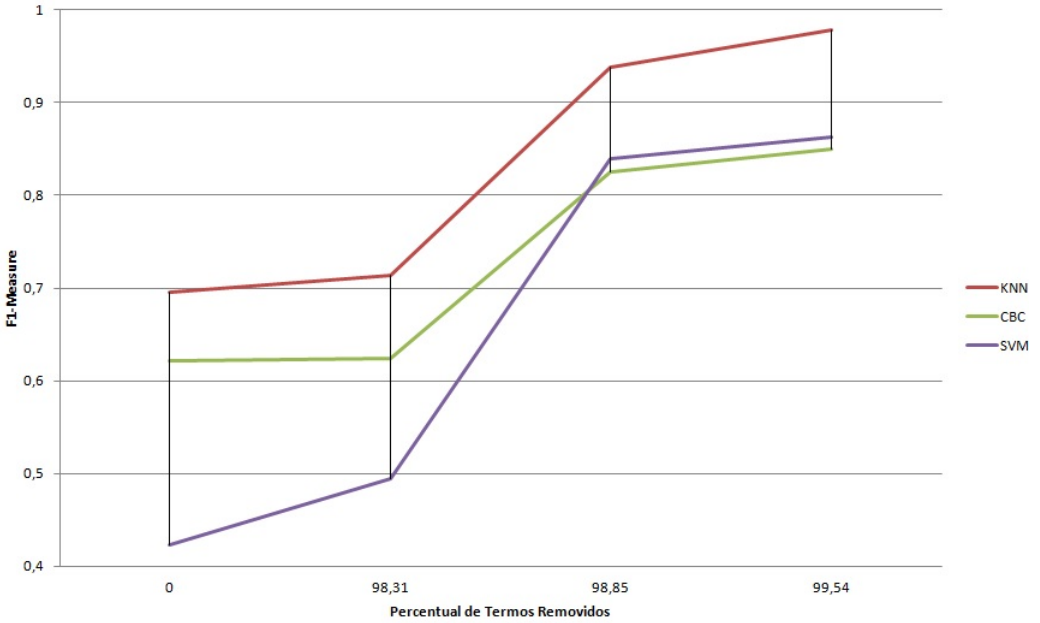

Figura 5.20: Evolução da medida F1-*measure* em função do percentual de termos removidos da base de dados Globo-*Comments* 02

significância de 1% sobre os resultados da medida F1-*measure*. Na avaliação foi percebido que os resultados obtidos para os três classificadores na etapa 3 foram estatisticamente superiores aos obtidos nas etapas 1 e 2. Da mesma forma, os resultados obtidos na etapa 4 foram estatisticamente superiores aos obtidos na etapa 3. Não houve diferença estatística entre os resultados da etapa 1 e etapa 2.

5.4 Discussão dos resultados

Analisando as medidas obtidas nos experimentos, é possível perceber que as técnicas de seleção de características utilizadas nos experimentos produziram melhoria nas métricas de classificação com os três classificadores utilizados.

Os experimentos envolvendo a base de dados Globo-*Comments* 01 indicam que a estratégia de seleção de características com uso de Algoritmo Genético seguido da extração de termos mais raros e de termos mais comuns apresentou maior eficiência no aumento dos índices de classificação, sendo, portanto, a estratégia adotada na classificação da base de dados Globo-*Comments* 02.

O uso da extração de termos que melhor caracterizam a base de dados com uso de Algoritmo Genético obteve melhores resultados quando aplicada à extração de termos que aumentam a densidade em cada categoria da base de dados, separadamente, levando em consideração o contexto em que os termos estão referenciados.

A estratégia adotada neste trabalho mostrou-se eficiente para melhorar a moderação automática de comentários (SAÚDE et al., 2014a). O valor mais alto obtido para o *Recall* foi de 96,78%, significativamente maior do que o obtido em (BUBACK, 2011), que foi de 36,94%. Com estes resultados, a probabilidade de um comentário indesejado ser divulgado diminuiu para apenas 3,22%. É importante considerar que o trabalho em (BUBACK, 2011) não utilizou as técnicas de redução de dimensionalidade discutidos neste trabalho, bem como extração de sufixos (*stemming*) e seleção de características.

Por se tratar de uma base de dados contendo apenas duas classes, era esperado que o classificador SVM apresentasse os melhores resultados. Este fato pode ser explicado em virtude da utilização de um kernel linear para separar as classes, o que pode ter afetado o desempenho deste classificador, pois, como observado pela análise dos dados, as classes não demonstraram ser linearmente separáveis.

É importante ainda salientar que, apesar da classificação com algoritmo KNN ter apresentado os melhores resultados em ambas as bases de dados ao final da aplicação da estratégia, seu uso implicou em maior custo no tempo de processamento da classificação em relação aos algoritmos CBC e SVM.

Capítulo 6

Conclusões e Trabalhos Futuros

Este trabalho apresentou uma estratégia para moderação automática de comentários de usuários. Foram utilizadas duas bases de dados, contendo comentários aprovados e reprovados para divulgação. A primeira base de dados, aqui denominada Globo-*Comments* 01, é composta por um conjunto de dados, num total de 978 comentários, cujos experimentos serviram para a escolha de uma estratégia a ser utilizada numa base de dados maior. A segunda base de dados, aqui denominada Globo-*Comments* 02, é formada por um grande conjunto de dados, num total de 657.705 comentários.

As bases de dados foram submetidas a técnicas de tratamento de textos como a retirada de *stopwords*, a lematização (extração do radical) de cada termo e a retirada de termos pouco referenciados na base. Foram ainda aplicados métodos para seleção de características utilizando SFS (*Sequential Forward Selection*), Algoritmo Genético e combinações de extração de termos raros e de termos comuns.

Os experimentos realizados na base Globo-*Comments* 01 indicaram que o uso de algoritmo genético seguido da extração de termos raros e de termos comuns apresentou melhor desempenho nos índices de classificação do que utilizando o algoritmo SFS seguido da extração de termos raros e de termos comuns.

É importante frisar que a aplicação do algoritmo genético apresentou maior eficácia quando aplicada para extração dos termos que aumentam as densidades de cada classe da base de dados, separadamente.

A estratégia apontada pelos experimentos com a base de dados Globo-*Comments* 01 foi utilizada na classificação da base de dados Globo-*Comments* 02. Os resultados obtidos nos experimentos comprovam que a estratégia utilizada mostrou-se eficiente, apresentando uma medida *Recall* de 96,78%. Com este resultado, a estratégia proposta errou na classificação de apenas 3,22% dos comentários, se comparado à moderação realizada pelo especialista humano.

Para aferição e refinamento da estratégia proposta neste trabalho, sugere-se:

- Verificar a amplitude de sua aplicação em outras bases de dados, comparando seu desempenho com uso de outros classificadores e de extração de características, como o LSI (*Latent Semantic Index*) (DEERWESTER, 1988);
- Aplicar novos testes de classificação do algoritmo SVM com *kernel* não-linear, buscando elevar os resultados das métricas de classificação para este classificador;
- Realizar novos experimentos em bases de dados que considerem o contexto (domínio) da notícia em que os comentários estão inseridos, como forma de aprimorar a moderação automática de comentários.

Referências Bibliográficas

ABE, S. *Support vector machines for pattern classification*. [S.l.]: Springer, 2010.

BAEZA-YATES, R.; RIBEIRO-NETO, B. *Modern information retrieval*. 2th. ed. [S.l.]: ACM press New York, 2011. 320 p.

BASAVARAJU, M.; PRABHAKAR, D. R. A novel method of spam mail detection using text based clustering approach. *International Journal of Computer Applications*, International Journal of Computer Applications, 244 5 th Avenue,# 1526, New York, NY 10001, USA India, v. 5, n. 4, p. 15–25, 2010.

BETTIO, R. W. d. et al. Inter-relaão das técnicas term extration e query expansion aplicadas na recuperação de documentos textuais. Florianópolis, SC, 2007.

BUBACK, S. *Utilizando aprendizado de máquina para construção de uma ferramenta de apoio a moderação de comentários*. [S.l.]: Pontificia Universidade Catolica, 2011.

CASTELLS, M. A sociedade em rede, vol. 1. *Editora Paz e Terra*, 1999.

CATARINA, A. S.; BACH, S. L. Estudo do efeito dos parametros genéticos sobre a solução otimizada e sobre o tempo de convergencia em algoritmos genéticos com codificações binária e real. *Acta Scientiarium, Tecnology*, v. 2, n. 2, p. 147–152, 2003.

CHANG, C.-C.; LIN, C.-J. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, v. 2, p. 27:1–27:27, 2011. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

CHEN, M.-S.; HAN, J.; YU, P. S. Data mining: an overview from a database perspective. *Knowledge and data Engineering, IEEE Transactions on*, IEEE, v. 8, n. 6, p. 866–883, 1996.

CHERKASSKY, V.; MULIER, F. M. *Learning from data: concepts, theory, and methods*. [S.l.]: John Wiley & Sons, 2007.

COVER, T.; HART, P. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, IEEE, v. 13, n. 1, p. 21–27, 1967.

DEERWESTER, S. Improving information retrieval with latent semantic indexing. 1988.

DEMŠAR, J. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, JMLR. org, v. 7, p. 1–30, 2006.

DUARTE, E. S. Sentiment analysis on twitter for the portuguese language. Faculdade de Ciências e Tecnologia, 2013.

FERNEDA, E. *Introdução aos Modelos Computacionais de Recuperação de Informação*. [S.l.]: Editora Ciência Moderna, 2012.

HAN, E.-H. S.; KARYPIS, G. *Centroid-based document classification: analysis and experimental results*. [S.l.]: Springer, 2000.

JOACHIMS, T. Making large scale svm learning practical. Universität Dortmund, 1999.

KODRATOFF, Y.; MICHALSKI, R. S. *Machine learning: an artificial intelligence approach volume III*. [S.l.]: Morgan Kaufmann Publishers Inc., 1990.

KOHAVI, R. et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: *IJCAI*. [S.l.: s.n.], 1995. v. 14, n. 2, p. 1137–1145.

LADHA, L.; DEEPA, T. Feature selection methods and algorithms. *International Journal on Computer Science and Engineering*, Engg Journals Publications, v. 3, n. 5, p. 1787–1797, 2011.

MASAND, B.; LINOFF, G.; WALTZ, D. Classifying news stories using memory based reasoning. In: ACM. *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*. [S.l.], 1992. p. 59–65.

MITCHELL, M. *An introduction to genetic algorithms*. 8th. ed. [S.l.]: MIT Press Cambridge, 2002.

MITCHELL, T. M. Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*, v. 45, 1997.

OLIVEIRA, J. de; NAVES, N. V. *Código penal:(Decreto-lei no. 2,848, de 7-12-1940)*. [S.l.]: Forense, 1984.

ORENGO, V. M.; HUYCK, C. R. A stemming algorithmm for the portuguese language. In: *SPIRE*. [S.l.: s.n.], 2001. v. 8, p. 186–193.

ORRIOLS-PUIG, A.; MACIA, N.; HO, T. K. Documentation for the data complexity library in C++. *Universitat Ramon Llull, La Salle*, v. 196, 2010.

PAK, A.; PAROUBEK, P. Twitter as a corpus for sentiment analysis and opinion mining. In: *LREC*. [S.l.: s.n.], 2010.

PANG, B.; LEE, L. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, Now Publishers Inc., v. 2, n. 1-2, p. 1–135, 2008.

POURHASHEMI, S. M.; OSAREH, A.; SHADGAR, B. E-mail spam filtering by a new hybrid feature selection method using *Chi2* and *CNB Wrapper*. *Int. J. Emerg. Sci*, v. 3, n. 4, p. 410–422, 2013.

REPÚBLICA, S. de Comunicação Social da Presidência da. *Pesquisa Brasileira de Mídia 2014 - Hábitos de Consumo de Mídia pela População Brasileira*. 2014. <http://observatoriodaimprensa.com.br/download/PesquisaBrasileiradeMidia2014.pdf>. [Online; acessado em 14-Setembro-2014].

SALTON, G.; MCGILL, M. J. Introduction to modern information retrieval. 1983.

SALTON, G.; WONG, A.; YANG, C.-S. A vector space model for automatic indexing. *Communications of the ACM*, ACM, v. 18, n. 11, p. 613–620, 1975.

SALTON, G.; YANG, C.-S. On the specification of term values in automatic indexing. *Journal of documentation*, MCB UP Ltd, v. 29, n. 4, p. 351–372, 1973.

SAÚDE, M. R. et al. A strategy for automatic moderation of a large data set of users comments. Centro Latinoamericano de Estudios en Informática, XL Conferencia Latinoamericana de Informática, 2014.

SAÚDE, M. R. et al. Seleção de características aplicada a moderação automática de comentários. Universidade Tecnológica Federal do Paraná, V Meditec, 2014.

SEBASTIANI, F. Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, ACM, v. 34, n. 1, p. 1–47, 2002.

SHAKHNAROVICH, G.; INDYK, P.; DARRELL, T. *Nearest-neighbor methods in learning and vision: theory and practice*. [S.l.: s.n.], 2006.

SHRIVASTAVA, J. N.; BINDU, M. H. E-mail spam filtering using adaptive genetic algorithm. *International Journal of Intelligent Systems & Applications*, v. 6, n. 2, 2014.

SOUZA, F. P. de. *Uma combinação de métodos de ponderação para melhoria da classificação de textos*. [S.l.]: Universidade Federal do Espírito Santo, 2014.

TRENTIN, T. R. D.; TRENTIN, S. S. Internet: publicações ofensivas em redes sociais e o direito a indenização por danos morais. *Revista Direitos Emergentes na Sociedade Global*, v. 1, n. 1, p. 79–93, 2012.

VAPNIK, V.; CORTES, C. Support-vector networks. *Machine learning*, v. 20, n. 3, p. 273–297, 1995.

YANG, Y.; LIU, X. A re-examination of text categorization methods. In: ACM. *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. [S.l.], 1999. p. 42–49.

YANG, Y.; PEDERSEN, J. O. A comparative study on feature selection in text categorization. In: *ICML*. [S.l.: s.n.], 1997. v. 97, p. 412–420.

YONG-FENG, S.; YAN-PING, Z. Comparison of text categorization algorithms. *Wuhan university Journal of natural sciences*, Springer, v. 9, n. 5, p. 798–804, 2004.