Pontifícia Universidade Católica
do Rio de Janeiro

Neide de Oliveira Gomes

Categorização de Textos - Estudo de Caso: Documentos de Pedidos de Patente no Idioma Português

Tese de Doutorado

Tese apresentada como requisito parcial para obtenção do grau de Doutor pelo Programa de Pós-Graduação em Engenharia Elétrica do Departamento de Engenharia Elétrica do Centro Técnico Científico da PUC – Rio.

Orientadora: Prof. Marley Maria B. Rebuzzi Vellasco
Co-Orientador: Prof. Emmanuel Piseces Lopes Passos

Rio de Janeiro
Junho de 2013

Pontifícia Universidade Católica
do Rio de Janeiro

Neide de Oliveira Gomes

Categorização de Textos - Estudo de Caso: Documentos de Pedidos de Patente no Idioma Português

Tese apresentada como requisito parcial para obtenção do grau de Doutor pelo Programa de Pós-Graduação em Engenharia Elétrica do Departamento de Engenharia Elétrica do Centro Técnico Científico da PUC – Rio. Aprovada pela Comissão Examinadora abaixo assinada.

Prof. Marley Maria Bernardes Rebuzzi Vellasco
Orientadora
Departamento de Engenharia Elétrica – PUC–Rio

Prof. Emmanuel Piseces Lopes Passos
Co-Orientador
Aposentado do IME

Prof. Karla Tereza Figueiredo Leite
UEZO

Prof. Rubens Nascimento Melo
Departamento de Informática - PUC–Rio

Prof. Ronaldo Ribeiro Goldschmidt
UFRRJ

Prof. Bernardo Henrique Todt Seelig
INPI

Prof. Anderson da Silva Moreira
INPI

Prof. José Eugenio Leal
Coordenador Setorial do Centro
Técnico Científico – PUC-Rio

Rio de Janeiro, 07 de junho de 2013

Neide de Oliveira Gomes

Graduou-se em Engenharia Elétrica, ênfase em Eletrotécnica, na Pontifícia Universidade Católica do Rio de Janeiro. É Mestre em Ciências em Engenharia Elétrica, ênfase em Sistemas de Potência, pela Universidade Federal do Rio de Janeiro. Especialização em: Análise de Sistemas pela Candido Mendes do Rio de Janeiro; Mecatrônica pela Universidade Estadual do Rio de Janeiro; e Ciências Públicas em Propriedade Intelectual pela UFRJ. Desde 1998 é servidora concursada do Instituto Nacional da Propriedade Industrial (INPI), sendo desde o ano de 2010 chefe da divisão de Física e Eletricidade (DIFEL) da Diretoria de Patentes (DIRPA) do INPI. Exerceu o cargo de engenheira eletricista–projetista em várias empresas por vários anos. Trabalhou como engenheira eletricista no Arsenal de Marinha do Rio de Janeiro.

Ficha catalográfica

Gomes, Neide de Oliveira -

Categorização de textos - estudo de caso: documentos de pedidos de patente no idioma português/ Neide de Oliveira Gomes; orientadora: Marley Maria B. Rebuzzi Vellasco; co-orientador: Emmanuel Piseces Lopes Passos; 2013.
v., 294 f.; il. ; 30 cm

1.Tese (doutorado) – Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica, 2013.

Inclui referências bibliográficas.

1. Engenharia Elétrica – Teses. 2. Categorização de documentos de pedidos de patente. 3. Categorização de textos. 4. Classificação de documentos de pedidos de patente. 5. Classificação de Textos. 6. Descoberta de conhecimento em textos referente a pedidos de patente. 7. Stemização. 8. Algoritmo *k*-Vizinhos-Mais-Próximos (*k*-NN). 9. Algoritmo baseado em centróide ou vetor protótipo. I. Vellasco, Marley Maria B. Rebuzzi. II. Passos, Emmanuel Piseces Lopes. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Elétrica. IV. Título

CDD:621.3

Para a minha mãe Hilda,
pelo apoio e confiança.

Agradecimentos

A minha orientadora Marley Maria B. Rebuzzi Vellasco pelos ensinamentos e pela compreensão pelos problemas surgidos.

Ao meu co-orientador Emmanuel Piseces Lopes Passos, que pela sua vasta experiência, me permitiu a elaboração dessa pesquisa.

A professora Karla Tereza Figueiredo Leite pela orientação pela escolha do programa de doutorado a seguir.

Aos professores que participaram da Comissão examinadora.

Aos meus colegas de trabalho do INPI, Camilo Braga Gomes e Bernardo Todt Seelig pela ajuda, sem o qual meu doutorado não teria sido possível de ter sido realizado. A todas as pessoas que me ajudaram, direta ou indiretamente durante o período do doutorado.

Aos funcionários da PUC pela ajuda durante a fase do programa, em especial a Alcina Portes. A todos os professores do Departamento pelos ensinamentos.

A PUC - Rio pela viabilização dessa pesquisa e pelo auxílio concedido através a bolsa parcial de estudos que me foi oferecida durante o período de elaboração da tese até junho de 2012.

A minha mãe pela educação, atenção e carinho de todas as horas, pelo ótimo relacionamento familiar e pelo entendimento pela minha ausência, mesmo estando acamada.

Resumo

Gomes, Neide de Oliveira Gomes; Vellasco, Marley Maria B. Rebuzzi (orientadora); Passos, Emmanuel Piseces Lopes (co-orientador);. **Categorização de Textos - Estudo de Caso: Documentos de Pedidos de Patente no Idioma Português.** Rio de Janeiro, 2013. 294p. Tese de Doutorado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Atualmente os categorizadores de textos construídos por técnicas de aprendizagem de máquina têm alcançado bons resultados, tornando viável a categorização automática de textos. A proposição desse estudo foi a definição de vários modelos direcionados à categorização de pedidos de patente, no idioma português. Para esse ambiente foi proposto um comitê composto de 6 (seis) modelos, onde foram usadas várias técnicas. A base de dados foi constituída de 1157 (hum mil cento e cinquenta e sete) resumos de pedidos de patente, depositados no INPI, por depositantes nacionais, distribuídos em várias categorias. Dentre os vários modelos propostos para a etapa de processamento da categorização de textos, destacamos o desenvolvido para o Método 01, ou seja, o *k-Nearest-Neighbor* (k-NN), modelo também usado no ambiente de patentes, para o idioma inglês. Para os outros modelos, foram selecionados métodos que não os tradicionais para ambiente de patentes. Para quatro modelos, optou-se por algoritmos, onde as categorias são representadas por vetores centróides. Para um dos modelos, foi explorada a técnica do *High Order Bit* junto com o algoritmo *k*-NN, sendo o *k* todos os documentos de treinamento. Para a etapa de pré-processamento foram implementadas duas técnicas: os algoritmos de stemização de Porter; e o *StemmerPortuguese*; ambos com modificações do original. Foram também utilizados na etapa do pré-processamento: a retirada de *stopwords*; e o tratamento dos termos compostos. Para a etapa de indexação foi utilizada principalmente a técnica de pesagem dos termos intitulada: frequência de termos modificada *versus* frequência de documentos inversa $TF'-IDF$. Para as medidas de similaridade ou medidas de distância destacamos: cosseno; Jaccard; DICE; Medida de Similaridade; HOB. Para a obtenção dos resultados foram usadas as técnicas de predição da relevância e do *rank*. Dos métodos implementados nesse trabalho, destacamos o *k*-NN tradicional, o qual apresentou bons resultados embora demande muito tempo computacional.

Palavras-chave

Categorização de documentos de pedidos de patente; categorização de textos; classificação de documentos de pedidos de patente; classificação de textos; descoberta de conhecimento em textos referente a pedidos de patente; stemização; algoritmo *k*-Vizinhos-Mais-Próximos (*k*-NN); algoritmo baseado em centróide ou vetor protótipo.

Abstract

Gomes, Neide de Oliveira Gomes; Vellasco, Marley Maria B. Rebuzzi (Advisor); Passos, Emmanuel Piseces Lopes (co-Advisor); **Text Categorization - Case Study: Patent's Application Documents in Portuguese**. Rio de Janeiro, 2013. 294p. Doctoral Thesis – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Nowadays, the text's categorizers constructed based on learning techniques, had obtained good results and the automatic text categorization became viable. The purpose of this study was the definition of various models directed to text categorization of patent's application in Portuguese language. For this environment was proposed a committee composed of 6 (six) models, where were used various techniques. The text base was constituted of 1157 (one thousand one hundred fifty seven) abstracts of patent's applications, deposited in INPI, by national applicants, distributed in various categories. Among the various models proposed for the step of text categorization's processing, we emphasized the one devellopped for the 01 Method, the *k-Nearest-Neighbor* (*k*-NN), model also used in the English language patent's categorization environment. For the others models were selected methods, that are not traditional in the English language patent's environment. For four models, there were chosen for the algorithms, centroid vectors representing the categories. For one of the models, was explored the High Order Bit technique together with the k-NN algorithm, being the *k* all the training documents. For the pre-processing step, there were implemented two techniques: the Porter's stemization algorithm; and the *StemmerPortuguese* algorithm; both with modifications of the original. There were also used in the pre-processing step: the removal of the stopwards; and the treatment of the compound terms. For the indexing step there was used specially the modified documents' term frequency versus documents' term inverse frequency $TF'-IDF$. For the similarity or distance measures there were used: cosine; Jaccard; DICE; Similarity Measure; HOB. For the results, there were used the relevance and the rank technique. Among the methods implemented in this work it was emphasized the traditional *k*-NN, which had obtained good results, although demands much computational time.

Keywords

Categorization of patent's application documents; text categorization; classification of patent's application documents; text classification; discovery in text knowledge based in patents; stemming; k-Nearest Neighbor (k-NN) algorithm; centroid or prototype algorithm.

Sumário

Lista de figuras

Lista de tabelas

Apêndice 6

Lista de Abreviaturas

AAC - Média Aritmética do Centróide (em inglês *Arithmetic Average Centroid*)

Árvore PC ou Árvore P - *Peano Count Tree* ou *Predicate Count Tree*

BPNN - Rede Neural *Back-Propagation* (em inglês *Back Propagation Neural N*etwork)

CFC - Centróide com Característica da Categoria (em inglês *Class-Feature-Centroid*)

DF – Frequência de Documentos

DVS – Decomposição de Valor Singular

EI – Extração de Informação (em inglês *Information Extraction*)

EPO – Organização de Patente Européia (em inglês *European Patent Organization*)

GIS – *Generalized Instance Set*

HOB – Bit de maior ordem (em inglês *High Order Bit*)

IDF – Frequência de documentos inversa (em inglês *Inverse Document Frequency*)

INPI – Instituto Nacional da Propriedade Industrial

IPC - Classificação Internacional de Patentes (em inglês *International Patent Classification*)

KDT – Descoberta de Conhecimento em Textos (em inglês *Knowledge Discovery in Texts*)

k-NN - Algoritmo k-Vizinho Mais Próximo (em inglês *k-Nearest Neighbor*)

LLSF – Método *Linear Least Squares Fit*

LSI – Indexação Semântica Latente (em inglês *Latent Semantic Indexing*)

MT – Mineração de Textos (em inglês *Text Mining*)

NB – Algoritmo de *Naïve* Bayes

PCT – Tratado de Cooperação de Patentes (em inglês *Patent Cooperation Treaty*)

PLN – Processamento de Linguagem Natural

RD – Redução da Dimensionalidade (em inglês *Dimensionality Reduction*)

RI – Recuperação de Informação (em inglês *Information Retrieval*)

RPI - Revista da Propriedade Industrial

SMART – *System for Manipulating and Retrieving Text*

SnoW – *Sparse Network of Winnows*

SOM – Método de *Self Organizing Map*

SVM - Método de *Support Vector Machine*

TACSY – *Taxonomy Access and Coding System*

TC – Categorização de textos (em inglês *Text Categorization*)

TF - Frequência de termos (em inglês *Term Frequency*)

TF.IDF – Frequência de termos *versus* frequência de documentos inversa (em inglês *Frequency Term versus Inverse Document Frequency*)

TF'.IDF – Frequência de termos modificada *versus* frequência de documentos inversa (em inglês *Modified Frequency Term versus Inverse Document Frequency*)

USPTO – *United States Patent Trademark Organization*

VSM **-** Modelo Espaço-Vetorial (em inglês *Vector Space Model*)

WIDF - Frequência de documento com pesagem inversa (em inglês *Weighting* Inverse *Document Frequency*)

WIPO – Organização Internacional de Patentes (em inglês *World International Patent Organization*)

Certeza
De tudo ficaram três coisas:
A certeza de que estamos sempre começando...
A certeza de que precisamos continuar...
A certeza de que seremos interrompidos antes de terminar...
Portanto devemos:
Fazer da interrupção um caminho novo...
Da queda um passo de dança...
Do medo, uma escada...
Do sonho, uma ponte...
Da procura um encontro...
(Fernando Pessoa)

1
Introdução

1.1
Motivação

Patentes constituem uma das formas mais antigas de proteção do capital intelectual. Segundo Barbosa (2003), o propósito da patente é incentivar a produção de novas tecnologias, através da garantia jurídica da exclusividade de seu uso.

Quando um pedido de patente é depositado no Instituto Nacional da Propriedade Industrial (INPI), geralmente o mesmo fica em sigilo durante 18 (dezoito) meses e nesse período ele é categorizado manualmente por examinadores de patentes, baseado no seu conteúdo, com o auxílio da Classificação Internacional de Patentes, o IPC (em inglês *International Patent Classification*), uma taxonomia padrão, desenvolvida pela Organização Mundial de Propriedade Intelectual (WIPO – em inglês *World Intellectual Property Organization*). O IPC é um sistema de classificação hierárquico, complexo, cobrindo todas as áreas tecnológicas, incluindo seções de química, física, civil, mecânica, eletricidade (eletrônica, eletrotécnica, telecomunicação), biotecnologia e por causa de seu escopo abrangente, muita mão-de-obra especializada é necessária para realizar a categorização manualmente, além do que devido à complexidade de tal sistema, pessoal não-especializado tem dificuldade em categorizar manualmente pedidos de patentes, segundo o IPC. Uma ferramenta automática pode ajudar na categorização dos documentos de pedidos de patentes no idioma português.

Quando um pedido de patente é examinado, uma busca acerca de invenções anteriores no mesmo campo tecnológico, conhecido com estado-da-arte, se baseia principalmente na exatidão da classificação dos documentos (Fall, 2003).

1.2
Objetivo

A categorização de textos, área escolhida para o estudo deste trabalho, é uma técnica usada em Processamento de Linguagem Natural (PLN), visando à classificação de documentos de textos. Ela permite a definição automática de categorias, bem como a classificação de um conjunto de documentos nessas categorias. Essa técnica pode ser usada como uma forma de organizar os documentos, tanto para recuperação quanto para a armazenagem a partir de categorias pré-definidas (Yang & Pedersen, 1997). Dessa forma o espaço de busca é reduzido, facilitando o acesso à informação, pois ao invés de selecionar um documento entre milhares, pode-se analisar apenas os documentos pertencentes às categorias de interesse (Silva, 2007).

O foco dessa pesquisa é buscar métodos computacionais capazes de categorizar pedidos de patentes, no idioma português, de forma rápida e precisa, envolvendo técnicas de aprendizagem de máquina e lingüística computacional.

Embora vários trabalhos já tenham sido desenvolvidos visando à categorização de patentes, por ter sido a maioria desenvolvida para o idioma inglês, não se pode comprovar que os resultados obtidos seriam os mesmos, caso fosse usado o idioma português.

O IPC é um sistema de categorização hierárquico, iniciando-se a classificação na categoria genérica da hierarquia (raiz) e se repetindo para as subcategorias, até atingir as categorias mais específicas, ou seja, as folhas (Moraes & Lima, 2007).

Embora os documentos de pedidos de patentes sejam categorizados e indexados hierarquicamente, i.e., com a classificação dividida em seções, classes, subclasses, grupos e subgrupos, essa forma de indexação, na maior parte dessa pesquisa, não foi levada em consideração. Em quase todo o estudo, focado em documentos de pedidos de patente de depositantes brasileiros, foram consideradas apenas classificações independentes consistindo da seção, classe e subclasse, visto que em muitas classificações incluindo grupos, há poucos documentos, não comportando uma aprendizagem automática.

No entanto, para fins de avaliação, desenvolveu-se também para um algoritmo específico, a classificação em nível de grupo mais elevado.

Inicialmente, esse trabalho deve ser usado como ponto de partida para a categorização automática, oferecendo sugestões para o encaminhamento dos documentos às divisões.

A classificação IPC tem como propósito principal o estabelecimento de uma ferramenta de busca efetiva para a recuperação de documentos de patentes pelos escritórios de Propriedade Industrial e seus usuários. Tem ainda o propósito de servir como (Pierre, 2010):

- um instrumento de armazenar corretamente documentos de patentes visando a facilitar o acesso de informação contido nele;

- uma base para seletivamente disseminar informação;

- uma base para buscar estado da arte;

- uma base para preparação de estatísticas referentes à Propriedade Industrial.

Os propósitos principais dessa pesquisa são:

- categorizar os documentos de pedidos de patente no idioma português, consistindo somente do Resumo, segundo a classificação IPC, usando uma ferramenta computacional, consistindo tais classificações de: a) seção, classe e subclasse; b) seção, classe, subclasse, grupo mais elevado; a categorização abrangendo:

  - coleta de documentos ou base de dados textuais;
  - limpeza do texto (correções ortográficas);
  - definição de uma lista de *stopwords* (*stoplist*) devendo tal lista ser desconsiderada quando da categorização;
  - tratamento de termos compostos, termos que quando aparecem juntos tem significados diferentes do que tem quando aparecem sozinhos;
  - stemização consistindo da remoção dos sufixos dos termos;
  - transformação dos dados através indexação: cada texto e seu conteúdo devem ser transformados para representações internas;
  - implementação do categorizador;
  - avaliação e interpretação dos resultados (cálculo das medidas de desempenho).

O objetivo mínimo do categorizador automático para categorização de pedidos de patentes deve ser capaz de predizer com exatidão pelo menos 80-90%

das categorias classificadas em nível de subclasse, de acordo com 3 (três) a 4 (quatro) sugestões, sendo que a decisão final deverá ser feita por um especialista (Fall & Benzineb, 2002).

Além do estudo de diferentes técnicas para a Categorização de Pedidos de Patente representados por seus Resumos, também se realizou estudos com duas técnicas para a etapa de stemização dos termos dos textos e algumas técnicas para a pesagem dos termos e medidas de similaridade, visando à obtenção do melhor desempenho.

1.3
Organização do Trabalho

Esse trabalho está organizado em 7 (sete) capítulos, seguida de bibliografia e Apêndices.

O capítulo 2 apresenta fundamentos e etapas do processo da Categorização de Textos. São discutidos ainda aspectos relacionados à Taxonomia IPC e o estado da arte da Categorização de Documentos de Patente.

O capítulo 3 descreve a base de dados utilizada e os modelos propostos para as várias etapas necessárias para o Processo de Categorização de Textos, tais como: entrada de dados; lista de *stopwords*; técnicas de stemização; e técnicas de indexação.

O capítulo 4 descreve os algoritmos utilizados na categorização de documentos de pedidos de patente com o estado da técnica dos vários métodos adotados.

O capítulo 5 descreve as técnicas de Pós-Processamento com as metodologias adotadas para a avaliação dos algoritmos.

O capítulo 6 apresenta os Resultados das Simulações incluindo a Stemização e a Categorização.

O capítulo 7 apresenta as conclusões finais e os trabalhos futuros.

2
Categorização de Textos: Fundamentos

2.1
Introdução

Com o rápido crescimento da quantidade de textos armazenados na *internet*, tornou-se necessária à criação de poderosos algoritmos e ferramentas capazes de tratarem com tais informações, auxiliando na busca, filtragem e gerenciamento de grandes bases de dados textuais (Krishnakumar, 2006).

Na maioria das vezes, essas informações não podem ser facilmente analisáveis, então a área de inteligência buscou suprir suas deficiências adotando técnicas provenientes da área de Recuperação de Informação (RI), Extração de Informação (EI) e Descoberta de Conhecimento em Textos (KDT).

A área de Recuperação de Informação (RI – em inglês *Information Retrieval*) tem por objetivo encontrar documentos que contenham informações relevantes às necessidades definidas por um usuário em uma consulta. Já a área de Extração de Informação (EI – em inglês *Information Extraction*) estuda metodologias, técnicas e sistemas que possam encontrar dados específicos dentro de textos. Para criar tais sistemas é necessário o desenvolvimento de muita engenharia de conhecimento, através o exame de alguns textos, visando à identificação de como a informação é codificado em frases da língua natural.

A partir dessas dificuldades, surgiu a área de Descoberta de Conhecimento em Textos (KDT, em inglês *Knowledge Discovery in Texts*). KDT é a área onde são aplicadas técnicas e ferramentas computacionais com o objetivo de auxiliar na busca de conhecimento novo e útil disponível em coleções textuais. Mesmo que tal conhecimento novo não seja a resposta direta às indagações do usuário, ele deve contribuir para satisfazer as necessidades de informação do mesmo. Os processos a serem executados para efetuar tal ação não são triviais, sendo requerido um alto grau de complexidade no tratamento das informações coletadas. Esse nível de dificuldade dá-se porque além das informações estarem disponíveis de forma semi-estruturada nos documentos, existem objetos que dificultam ainda mais o processamento como imagens, tabelas, planilhas, gráficos, dentre outros (Gomes & Costa, 2005).

Segundo Tan et al. (2005) KDT é o processo de extrair padrões interessantes e não-triviais, a partir de textos de documentos textuais.

Dentre as várias técnicas de KDT temos: Descoberta Tradicional após a Extração; Descoberta por Listas de Conceitos-Chave. Na primeira, os dados são extraídos do texto e colocados em bases de dados estruturados e na segunda, são utilizadas técnicas semelhantes â geração de centróides de categorias, sendo gerada uma lista com os conceitos principais de uma única categoria (Gomes & Costa, 2005).

Uma maneira de se manusear uma grande quantidade de documentos é organizando-as em categorias. As categorias são usualmente hierárquicas porque elas oferecem um meio de se achar e encontrar informações arbitrariamente. Atualmente essa técnica de armazenamento necessita eficiente método automático de categorização.

A categorização de textos é uma técnica de aprendizagem supervisionada usada em Processamento de Linguagem Natural, para categorizar automaticamente documentos de textos, baseada em um conjunto de documentos de treinamento já categorizados e com categorias já predefinidas (Krishnakumar, 2006; Moraes & Lima, 2007).

Processamento de Linguagem Natural (PLN) é a área de conhecimento que se dedica ao estudo, tratamento e compreensão da linguagem humana através de tecnologia computacional (Dias & Malheiros, 2005).

As bases de dados de patentes necessitam do uso de sistemas de categorias hierárquicos para serem categorizados. Patentes cobrem uma vasta área de categorias e cada campo pode ser dividido em subtópicos até que um nível de especialização seja alcançado.

O processo de classificação hierárquica inicia na categoria genérica da hierarquia, ou seja, a raiz e se repete para as subcategorias, até atingir as categorias mais específicas, ou seja, as folhas (Moraes & Lima, 2007).

A técnica de categorização de textos pode ser usada como uma ferramenta para organizar documentos tanto para recuperação, quanto para armazenamento, a partir de categorias pré-definidas.

A categorização automática envolve o uso de vários algoritmos para automaticamente designar categorias a documentos. São usados subconjuntos de documentos para treinamento (conjunto de treinamento). O algoritmo de treinamento é então aplicado a documentos não-classificados. O problema desse método é que a exatidão dos resultados no mundo real das bases de dados não é suficientemente alta. Tais algoritmos tipicamente alcançam entre 75-80% de exatidão (Chow & Perumal, 2009).

Quando há uma grande quantidade de categorias, a complexidade é contornada com categorizadores multi-classes, os quais podem incorrer em um grande tempo de processamento e armazenamento.

Quando se realiza um exame de um pedido de patente, a busca para se achar documentos relevantes pertencentes ao mesmo campo tecnológico, chamada de estado da arte, se baseia entre outros fatores, na exatidão da categorização dos documentos. A recuperação de documentos de patentes é muito importante para o desenvolvimento da tecnologia.

2.2
A Taxonomia IPC e a Categorização de Pedidos de Patente

2.2.1
A Taxonomia IPC

A Classificação Internacional de Patentes (IPC, em inglês *International Patent Classification*), desenvolvida e gerenciada pela WIPO (em inglês *World International Patent Organization*) é um sistema ou *thesaurus* usado pelos escritórios, para classificação de patentes, servindo como guia para que os indexadores e usuários do sistema patentário representem em uma linguagem comum os diversos campos tecnológicos. Esse sistema serve como ferramenta de busca para recuperação dos documentos por usuários e está sendo atualizado periodicamente, acompanhando o desenvolvimento técnico cientifico, tendo sido criado há aproximadamente 40 (quarenta) anos. A Classificação Internacional de Patentes foi implementada pelo Acordo de Estraburgo (1971) e entrou em vigor no Brasil em 1975, por meio do decreto 76.472 (Jannuzzi, 2007).

O sistema de classificação hierárquica IPC está dividido em seções, classes, subclasses, grupos e subgrupos, onde as tecnologias são discriminadas em diversos níveis, em ordem decrescente de hierarquia (http://www.ompi.ch/classifications/ipc/en/ITsupport/Categorization/dataset/wipo-alpha-readme.html).

A taxonomia IPC contém aproximadamente 70352 (setenta mil trezentos e cinquenta e duas) categorias em nível de subgrupo que cobrem todo o domínio da tecnologia industrial (http://ipc.inpi.gov.br versão 2012.01).

A taxonomia IPC, no mais alto da hierarquia, é composta de 8 (oito) seções (denominadas de A até H); 129 (cento e vinte e nove) classes; aproximadamente 648 (seiscentos e quarenta e oito) subclasses; 7200 (sete mil e duzentos) grupos principais; e 70352 (setenta mil, trezentos e cinqüenta e dois) subgrupos nos níveis mais baixos.

Os campos tecnológicos estão representados em seções discriminadas pelas letras de A até H: A representa "Necessidades Humanas"; B representa "Operações de Processamento; Transporte"; C representa a "Química; Metalurgia"; D representa "Têxteis; Papel"; E representa as "Construções Fixas"; F representa a "Engenharia Mecânica; Iluminação; Aquecimento; Armas; Explosão"; G representa a "Física"; H a "Eletricidade". Cada seção é dividida em classes, consistindo de símbolos de seção seguidos de um número de dois dígitos, tal como A01 (em nível de classe). Por sua vez, cada classe é dividida em várias subclasses, cujos símbolos consistem do símbolo da classe seguida de uma letra maiúscula, por exemplo, A01B (subclasse).

A seção A possui 16 (dezesseis) classes, a seção B possui 37 (trinta e sete) classes, a seção C possui 21 (vinte e uma) classes, a seção D possui 9 (nove) classes, a seção E possui 8 (oito) classes, a seção F possui 18 (dezoito) classes, a seção G possui 14 (quatorze) classes e a seção H possui 6 (seis) classes (http://ipc.inpi.gov.br versão 2012.01).

A seção A possui 8485 (oito mil, quatrocentos e oitenta e cinco) subgrupos, a seção B possui 16741 (dezesseis mil, setecentos e quarenta e um) subgrupos, a seção C possui 14469 (quatorze mil, quatrocentos e sessenta e nove) subgrupos, a seção D possui 3050 (três mil e cinqüenta) subgrupos, a seção E possui 3250 (três

mil, duzentos e cinqüenta) subgrupos, a seção F possui 8528 (oito mil, quinhentos e vinte e oito) subgrupos, a seção G possui 7745 (sete mil, setecentos e quarenta e cinco) subgrupos e a seção H possui 8084 (oito mil e oitenta e quatro) subgrupos.

Para a maioria dos documentos de patente, não é atribuída somente uma classificação. É atribuída também um conjunto de classificações secundárias, relacionado a outros aspectos incluídos nos documentos de patente (Fall et al., 2003a) (http://ipc.inpi.gov.br versão 2012.01).

Para a seção A, com 16 (dezesseis) classes, os 8485 (oito mil, quatrocentos e oitenta e cinco) subgrupos estão discriminados (versão 2012.01) como: A01 (Agricultura; Silvicultura; Pecuária; Caça; Captura em Armadilhas; Pesca) com 1434 (hum mil, quatrocentos e trinta e quatro) subgrupos; A21 (Cozedura ao Forno; Equipamento para Preparo ou Processamento de Massas; Massas para Cozedura ao Forno) com 128 (cento e vinte e oito) subgrupos; A22 (Matança de Animais; Beneficiamento de Carne; Processamento de Aves Domésticas ou Peixes) com 63 (sessenta e três) subgrupos; A23 (Alimentos ou Produtos Alimentícios; Seu Beneficiamento, não Abrangido por outras classes) com 660 (seiscentos e sessenta) subgrupos; A24 (Tabaco; Charutos; Cigarros; Artigos para Fumantes) com 237 (duzentos e trinta e sete) subgrupos; A41 (Vestuário) com 234 (duzentos e trinta e quatro) subgrupos; A42 (Chapéus) com 45 (quarenta e cinco) subgrupos; A43 (Calçados) com 386 (trezentos e oitenta e seis) subgrupos; A44 (Artigos de Armarinho; Bijuterias) com 128 (cento e vinte e oito) subgrupos; A45 (Artigos Portáteis ou de Viagem) com 406 (quatrocentos e seis) subgrupos; A46 (Escovas) com 67 (sessenta e sete) subgrupos; A47 (Móveis; Artigos ou Aparelhos Domésticos; Moinhos de Café; Moinhos de Especiaria; Aspiradores em Geral) com 1350 (hum mil, trezentos e cinqüenta) subgrupos; A61 (Ciência Médica ou Veterinária; Higiene) com 2348 (dois mil, trezentos e quarenta e oito) subgrupos; A62 (Salvamento; Combate ao Fogo) com 209 (duzentos e nove) subgrupos; A63 (Esportes; Jogos; Recreação) com 789 (setecentos e oitenta e nove) subgrupos; A99 (Matéria não Incluída em Outro Local desta Seção) com 1 (hum) subgrupo (http://ipc.inpi.gov.br versão 2012.01).

Para a seção H, dividida em 6 (seis) classes (versão 2012.01), os 8084 (oito mil e oitenta e quatro) subgrupos estão discriminados como: H01 (Elementos Elétricos Básicos) com 3785 (três mil, setecentos e oitenta e cinco) subgrupos;

H02 (Produção, Conversão ou Distribuição de Energia Elétrica) com 1087 (hum mil e oitenta e sete) subgrupos; H03 (Circuitos Eletrônicos Básicos) com 1075 (hum mil e setenta e cinco) subgrupos; H04 (Técnicas de Comunicação) com 1753 (hum mil, setecentos e cinqüenta e três) subgrupos; H05 (Técnicas Elétricas não Incluídas em Outro Local) com 383 (trezentos e oitenta e três) subgrupos; H99 (Matéria não Incluída em Outro Local desta Seção) com 1 (hum) subgrupo (http://ipc.inpi.gov.br versão 2012.01).

Os documentos de patentes são categorizados e indexados com o auxílio da Classificação Internacional de Patentes (IPC – em inglês *International Patent Classification*). Nos escritórios europeu e americano, após a indexação, os documentos são inseridos em banco de dados informatizados, para que os usuários do sistema de patentes possam recuperar a informação tecnológica contida nesses ativos de produção intelectual. Quando do inicio desse trabalho, na base brasileira de patentes, somente os Resumos dos pedidos estavam disponibilizados eletronicamente através das RPI's (Revistas da Propriedade Industrial) e do *site* http://worldwide.espacenet.com. No entanto, estão sendo digitalizados, na sua íntegra, todos os pedidos de patentes da base brasileira. Brevemente ela será disponibilizada eletronicamente para o público.

A maioria dos trabalhos publicados realiza as avaliações sobre textos bem escritos, como é o caso da coleção Reuters, que contêm textos jornalísticos, sendo que tais textos contêm informações bem objetivas e são escritos e revisados por profissionais, portanto com pouquíssimos erros ortográficos. Entretanto, os textos dos pedidos de patente brasileiros contêm erros ortográficos e de digitação, escritos às vezes por pessoas não conhecedoras do assunto ou com traduções mal feitas.

Os principais elementos constituintes do documento do pedido de patente são: folha de rosto; relatório descritivo; desenhos (se houver); reivindicações; e resumo. A folha de rosto apresenta os dados formais da patente, tais como nome do(s) inventor(es), país de origem, nome do titular da patente, classificação internacional. O Relatório Descritivo faz a descrição do objeto da invenção de modo a possibilitar a sua realização por um técnico no assunto. O teor das reivindicações, baseado nas informações constantes do Relatório Descritivo é o que determina a extensão da proteção conferida pela patente (Jannuzzi et al.,

2007). Na base de patentes brasileiras, até o término dessa pesquisa, somente o título e o Resumo estavam disponíveis eletronicamente, portanto só foram considerados esses elementos para a elaboração desse trabalho.

A categorização de pedidos de patentes pode ser realizada de dois modos: um algoritmo pode considerar a taxonomia como um sistema de categorias independentes; ou pode incorporar a hierarquia no algoritmo de categorização.

O ato de categorizar uma matéria patenteável perpassa pelas etapas de análise do documento, identificação dos seus principais conceitos e representação desses conceitos em uma linguagem de indexação.

2.2.2
Categorizadores de Documentos de Patente – Revisão da Literatura

Vários algoritmos para categorização de documentos de patentes foram desenvolvidos baseados em diferentes características, tais como: Chakrabarti et al., 1997-1998; Larkey, 1998; Kohonen et al., 2000; Fall et al., 2003-2004; Trappey et al., 2006; Loh et al., 2006; Kim & Choi, 2007; Cong & Tong, 2008; Cong & Loh, 2010. Alguns utilizaram citações de documentos contidos nas patentes para melhorar o desempenho da categorização: Lai & Wu, 2005; Li et al., 2007; enquanto outros empregaram *metadata* referente a patentes, tais como o nome do inventor, conseguindo melhoras no desempenho da classificação, tais como: Richter & MacFarlane, 2005 (Shih & Liu, 2010).

As características extraídas dos documentos de patente podem ser divididos em 3 (três) tipos: características de seu conteúdo; informações dos documentos citados na patente e/ou documentos que citam a patente; e *metadata* (Shih & Liu, 2010).

Para categorizadores baseados nas características do conteúdo da patente, o documento é representado por vetores de pesos dos termos do documento. A similaridade entre 2 (dois) documentos de patente é baseado em alguma técnica de similaridade, sendo a mais comum a do cosseno e para técnica de pesagem do termo sendo a mais comum a TF.IDF (função da freqüência do termo/ freqüência inversa do documento). Vários categorizadores têm sido usados, sendo o *k*-NN (em inglês *k-Nearest Neighbor*) um dos mais usados e a categoria escolhida a que

mais ocorre entre os documentos vizinhos mais próximos. Alguns categorizadores não usam o texto completo da patente, alguns só usam o Resumo (em inglês *Abstract*), o Estado da Arte (em inglês *background*) e os Resultados (em inglês *Results*) (Shih & Liu, 2010).

O sistema de classificação americano (USPTO, em inglês *United States Patents Trademark Organization*) usa nomenclatura diferente do sistema de Classificação Internacional de Patentes (IPC). Classe na USPTO corresponde à subclasse na IPC, subclasse na USPTO corresponde a grupo e a subgrupo na IPC (Smith, 2002).

Os documentos de patentes americanos podem variar em tamanho entre poucos *kilobytes* a 1.5 *megabytes*. Os documentos constam de centenas de campos, no entanto a maioria desses campos é pequena e não é do tipo texto, contendo informações tais como: Número do Depósito do Pedido (em inglês *Application Number*); Número da Patente (em inglês *Patent Number*); Data do Depósito do Pedido (em inglês *Date of Application*); Data de Publicação (em inglês *Publication Date*); etc. Outros campos são pequenos e contêm informações de texto específicas, tais como: nome e endereços dos inventores (em inglês *Authors or Inventors*), depositantes (em inglês *Assignees*); representantes (em inglês *Attorneys*). Os campos de texto mais importantes, do tipo narrativo são: título (em inglês *Title*); Resumo (em inglês *Abstract*); Resumo do Estado da Arte (em inglês *Background Summary*); Descrição Detalhada (em inglês *Detailed Description*). Podem ainda conter as seguintes partes: *Cross-Reference to Related Application*; *Field of the Invention*; *Background of the Invention*; *Summary of the Invention*; (*Brief) Description of the Drawings*; *Detailed Description of Exemplary (Preferred) Embodiments*; *Figures*; *Publication Classification* (Larkey, 1998).

Metadata é descrita como informação que descreve dados. *Metadata* em um documento de patente pode ser o nome do inventor, o nome do depositante e tais informações podem estar relacionadas com o conteúdo do documento e podem ser usadas para auxílio da categorização.

Para categorizadores baseados nas informações dos documentos citados no documento de patente, suponha que sejam citados no seu conteúdo 5 (cinco)

documentos. Caso 3 (três) deles pertençam à categoria C1 e os outros 2 (dois) a categoria C2, o documento de patente será categorizado com a categoria C1 (Shih & Liu, 2010).

O primeiro classificador hierárquico foi desenvolvido por Chakrabarti et al (1997, 1998) e era constituído de 12 (doze) subclasses organizado em 3 (três) níveis e usava o sistema de classificação hierárquico Bayesiano e o discriminante Fisher (Fall et al., 2003a; Seddiqui et al., 2008).

O discriminante Fisher é uma técnica muito conhecida de reconhecimento de padrão estatístico. É usado para distinguir termos característicos de termos com ruídos (em inglês *noise terms*) eficientemente. Para esse categorizador foram levadas em consideração para a categorização de documentos de patente, as classificações dos documentos citados no Relatório Descritivo, as quais, segundo o autor aumentam a precisão dos resultados (Fall et al., 2003a; Seddiqui et al., 2008; Tikk & Biró, 2003).

Larkey (1998) criou uma ferramenta para categorizar patentes americanas segundo a classificação americana, usando o algoritmo *k*-Vizinhos-Mais-Próximos (*k*-NN, em inglês *k-Nearest Neighbor*). A inclusão de frases (termos de palavras compostas) durante a indexação foi a responsável pelo aumento da precisão do sistema para busca de textos de patentes, mas não para a categorização. A precisão do sistema não foi divulgada (Fall et al., 2003a; Seddiqui et al., 2008; Tikk & Biró, 2003).

Larkey (1998), em seu artigo "*Some Issues in the Automatic Classification of U.S. Patents*", cita que junto com o algoritmo *k*-NN foi usado um sistema de recuperação de informação probabilístico denominado *Inquery*, baseado no sistema *Bayesiano*, usando a pesagem TF.IDF. O algoritmo de recuperação retorna uma lista de documentos com uma avaliação dos documentos recuperados. Essa avaliação, realizada através o *Inquery*, pode ser chamada de medida de similaridade e o documento que se localizar no topo da lista é a categoria candidata para a classificação do documento. Para patentes, foram selecionadas seções ou partes de seções do documento, constando dos termos mais frequentes do título, Quadro Reivindicatório, as 20 (vinte) primeiras linhas do *Background*

Summary, sendo que os termos do título receberam valores de peso 3 (três) vezes maiores do que os pesos dados para os demais.

Kohonen et al. (2000) desenvolveu um SOM (*Self Organizing Map*), tendo sido obtida uma precisão de 60.6% para categorização de documentos de patente distribuídos em 21 (vinte e uma) categorias para SOM de duas dimensões (Fall et al., 2003a; Seddiqui et al., 2008; Tikk & Biró, 2003).

Gey et al. (2001) criou uma solução baseada na *web* para categorizar documentos de patentes dos escritórios americano e internacional, mas não realizou testes detalhados acerca das métricas de acerto (Fall et al., 2003a).

Referente a um conjunto de testes de categorização de patentes, Krier & Zacca (2002) desenvolveram um estudo comparativo entre vários categorizadores acadêmicos e comerciais, contudo sem publicar resultados detalhados. O participante que obteve o melhor resultado teve seu trabalho publicado, tendo sido implementado para a categorização uma variante de *Winnow Balanced*, um classificador *online* com um esquema de atualização de pesos multiplicativos. A categorização foi desempenhada a um nível de 44 (quarenta e quatro) e 549 (quinhentas e quarenta e nove) categorias específicas da classificação da EPO (em inglês *European Patent Organization*), tendo sido obtida precisões de 78% e 68% respectivamente, quando medida com um critério de sucesso customizado. As bases de patentes não eram publicadas naquela época e esses dados não puderam ser confirmados (Fall et al., 2003a; Seddiqui et al., 2008; Tikk & Biró, 2003).

O sistema denominado de OWAKE (Delegation of Japan, 2000), de categorização automática, está sendo usado no *Japanese Intellectual Property Cooperation Centre* (IPCC). Ele é basicamente designado para categorização de documentos de patentes no sistema japonês, usando o sistema *F-term* que é um complemento do sistema IPC proporcionando meios de busca de documentos de acordo com as características técnicas dos pedidos japoneses. Para o sistema *F-term* foi alcançada uma precisão de 90%, para classificação de patentes japonesas para 38(trinta e oito) diferentes grupos técnicos. Nesse algoritmo foi usado o esquema de classificação hierárquico, inicialmente sendo usado o algoritmo Rocchio e depois usado o algoritmo *k*-NN para refinamento das categorias

prognosticadas. Foram usados os textos completos dos documentos de patente, tendo sido extraídos *stopwords* (Fall et al., 2003a).

Em 2002, a coleção *WIPO-alpha* em inglês foi publicada e logo após o *corpus* alemão de depósitos de pedidos de patente *WIPO-de* (Fall et al., 2003a; Seddiqui et al., 2008; Tikk & Biró, 2003).

Os documentos da coleção *WIPO-alpha*, no idioma inglês, publicados entre 1998 e 2002, consistem de pedidos de patentes submetidas à WIPO (*World Intellectual Property Organization*) através o Tratado de Cooperação de Patentes - PCT (em inglês *Patent Cooperation Treaty*). O pedido de patente incluiu: título; lista de inventores; lista de depositantes; Relatório Descritivo; Quadro Reivindicatório; Resumo. A coleção WIPO-alpha não incluiu os Desenhos (Fall et al., 2003a, 2003b).

Cada categoria de patentes pode cobrir áreas específicas, portanto é possível que dois escritórios de patentes categorizem documentos similares diferentemente, particularmente em categorias com conteúdo que se sobreponham. Os escritórios de patentes Europeu (EPO), Japonês (JPO) e Americano (USPTO) possuem seus sistemas de classificação independentes, contudo eles usam a IPC como taxonomia adicional. Pedidos de patente são muitas vezes republicados com modificações portanto, a coleção *WIPO-alpha* pode ter documentos de patente duplicados (Fall et al., 2003a, 2003b).

A coleção *WIPO-alpha* foi dividida em duas subcoleções, chamadas de coleção de treinamento e coleção de teste. A coleção de treinamento consistiu de documentos pertencentes aos grupos principais, com a restrição de que cada subclasse deveria conter entre 20 (vinte) e 2000 (dois mil) documentos. Para a coleção de teste foram usados documentos depositados em 2001, com a restrição de que cada subclasse deveria conter entre 10 (dez) e 1000 (hum mil) documentos. Não foram incluídas categorias com poucos documentos, visando o descarte de tecnologias que não são usadas frequentemente (Fall et al., 2003a, 2003b).

Para a coleção *WIPO-alpha* foram usadas as seguintes ferramentas de categorização automática visando a multiclassificação: o pacote *rainbow*, parte do *bag-of-words* (*bow*) e o SnoW (rede esparsa do *Winnows*). O pacote *rainbow* incluía um estudo comparativo entre 4 (quatro) classificadores conhecidos: Naive

Bayes - NB; *Support Vector Machine* – SVM; *k*-NN com *k* igual a 30; e o SnoW uma variante de *Winnow;* e os desenvolveram por meio de medidas de desempenho customizadas para computador tipo PC (Fall et al., 2003a, 2003b).

Na etapa de pré-processamento para o pacote *rainbow*, foram retirados *stopwords*, usada a stemização de PORTER, a seleção de termos foi feita levando-se em consideração o Ganho de Informação e a indexação foi desenvolvida em nível de freqüência de palavras de cada documento. Foi reportado que a stemização teve pouco impacto na efetividade do sistema, contudo diminuiu o tempo de processamento (Fall et al., 2003a, 2003b).

O pacote SnoW usado em espaços esparsos, foi implementado com uma rede esparsa de funções lineares. O SnoW (*Sparse Network of Winnows*) é um categorizador multi-classes implementando algoritmos tipo Perceptron, Naive Bayes e Winnow. Uma variação de regras de atualizações Winnow foi usada para treinamento (Fall et al., 2003a, 2003b).

SnoW foi usado com indexação de pesagem binária, o qual produziu melhores resultados do que a indexação por freqüência de palavras. Foi usada a remoção de *stopwords* e não foi usada a stemização. Levando-se em consideração para a categorização dos documentos de patente: (a) o título; (b) o Quadro Reivindicatório; (c) 300 (trezentas) palavras no máximo incluindo título, inventores, depositantes, Resumo e Relatório Descritivo; (d) título, inventores, depositantes e Resumo, os autores acharam que em nível de classe, para a opção (c) e para o esquema de Prognóstico de Topo, ou seja, considerando para a categoria correta somente o primeiro colocado, os algoritmos NB e SVM foram os melhores (55%), enquanto para o nível de subclasse, SVM sobrepujou os outros métodos (41%). Para em nível de classe, para a opção (c) e para o esquema de Três Prognósticos, ou seja, considerando a categoria correta qualquer um dos 3 (três) primeiros colocados, o algoritmo NB foi o que apresentou melhor resultado (79%) e para nível de subclasse foi o *k*-NN que apresentou o melhor resultado (62%). Em nível de classe, entre as opções de se levar em consideração para a categorização o (a) título, o (b) Quadro Reivindicatório e as (c) 300 (trezentas) primeiras palavras; a última opção foi a que apresentou os melhores resultados. Em nível de subclasse, entre as opções de se levar em consideração para a

categorização as (c) 300 (trezentas) primeiras palavras e o (d) Resumo, a primeira opção foi a que apresentou os melhores resultados (Fall et al., 2003a, 2003b).

A partir daí, muitos trabalhos reportaram o uso da coleção *WIPO-alpha*. A maioria dos trabalhos foi desenvolvida somente com o uso de um subconjunto do *corpus*.

Hofmann et al. (2003) desenvolveu um trabalho usando a seção D (têxtil) com 160 (cento e sessenta) categorias ao nível de subgrupo e obteve 71.9% de exatidão (em inglês *accuracy*) (Tikk & Biró, 2003).

Godbole & Sarawagi (2004) apresentaram outra variante de SVM tendo sido testado o algoritmo na hierarquia completa do IPC, especificamente na seção F (engenharia mecânica, iluminação, aquecimento, armas, explosivos). Foi obtida uma exatidão (em inglês *accuracy*) de 44.1% e 68.8% respectivamente (Tikk & Biró, 2003).

Cai & Hoffmann (2004) testaram seu categorizador também a partir de uma variante do SVM hierárquico, usando cada seção da *WIPO-alpha* e foi obtido 32.4-42.9% de exatidão (em inglês *accuracy*) em nível de grupo principal (Tikk & Biró, 2003).

Rousu et al. (2005) evoluiu seu algoritmo a partir de uma variante do SVM com rede Markov de margem máxima para a seção D da hierarquia e obteve 76.7% de valor de Medida de F1 (em inglês *F-measure*) (Tikk & Biró, 2003).

Um sistema de gerenciamento de conhecimento orientado para depósitos de pedidos de patente foi desenvolvido por Trappey et al. (2006) que incorporou a metodologia de organização de patentes, classificação e busca baseada na tecnologia de rede neural de *Back-Propagation* (BPNN). Essa aproximação se focou no melhoramento do sistema de gerenciamento de documentos de patente, em termos de aplicação e exatidão. Os autores compararam seu método com o modelo estatístico Bayesiano e acharam algum melhoramento quando testado em dois níveis da coleção *WIPO-alpha* (B25 – ferramentas manuais elétricas) com 9 (nove) subníveis de categorias. O artigo colocou ênfase especial na extração de frases-chaves do conjunto de documentos, que foram então usados como entradas para o classificador BPNN (Tikk & Biró, 2003).

O Escritório Europeu de Patentes (EPO, em inglês *European Patent Office*) desenvolveu testes direcionados a *software* categorizadores de pedidos de patente. O objetivo foi de construir um pré-classificador para busca de anterioridades. Os técnicos do Instituto Europeu, na época, que realizavam buscas, estavam divididos em 44 (quarenta e quatro) diretorias e 549 (quinhentos e quarenta e nove) grupos, cada um trabalhando em um domínio específico da classificação IPC. Em nível de diretoria a exatidão para o trabalho manual foi de 81.2%. O objetivo do classificador era de melhorar o desempenho. As conclusões foram as seguintes (Fall et al., 2003a):

- quando usado o texto completo das patentes, a precisão foi de 2-9% maior do que quando usado somente o resumo;

- para uma abrangência de 100%, a precisão foi de 72% para 44 (quarenta e quatro) categorias e 57% para 549 (quinhentos e quarenta e nove) categorias quando usado o categorizador Inxight. Esse resultado se referiu somente ao Prognóstico de Topo e seria maior se o Segundo Prognóstico tivesse sido incluído. O categorizador Inxight, baseado na tecnologia de Processamento de Linguagem Natural (stemização, tokenização, etc) e algoritmo estatístico (*k*-NN método) foi desenvolvido pela Xerox Research;

- num nível desejado de 81.2% de precisão, foi alcançada uma abrangência de 78%. Nesse caso 22% das patentes não foram designadas a um diretório e foram direcionadas para classificação manual;

- a velocidade de categorização não foi um problema, mas o treinamento de taxonomias contendo 100 (cem) categorias, algumas vezes levava uma semana no computador tipo PC;

- considerando as categorizações incorretas, foi indicado que tanto para os procedimentos manuais, quanto para os procedimentos automáticos, tipicamente os erros cometidos eram similares. Isso provavelmente resulta da sobreposição de categorias (por exemplo química orgânica e farmácia).

Segundo Fall et al. (2003a), resultados foram publicados por Koster (2001), o qual realizou simulações comparando o algoritmo Rocchio com o algoritmo Winnow, este último um algoritmo de aprendizagem que refina sua discriminação entre documentos relevantes dos documentos não-relevantes na fase de treinamento. Com o aumento dos documentos de treinamento (Resumo), a

precisão e abrangência da categorização aumentou, sendo que a técnica de Winnow se sobrepujou a de Rocchio para grandes conjuntos de treinamento. O uso da stemização aumenta a precisão, mas diminui a abrangência levando-se em consideração o texto completo do documento, contudo a precisão diminui e a abrangência aumenta levando-se em consideração somente o Resumo. Para seleção do termo do vocabulário completo para categorização de patentes, visando à discriminação das categorias das patentes, as melhores técnicas estão relacionadas a Ganho de Informação para cada termo, particularmente onde somente pequenas quantidade de termos discriminatórios são retidos. De todos os testes dirigidos pela EPO e executados por parceiros acadêmicos e comerciais, o algoritmo de Winnow teve o melhor desempenho (Fall & Benzineb, 2002).

O escritório americano de patentes (USPTO) tem um sistema próprio de categorização consistindo aproximadamente de 400 (quatrocentas) classes e 135000 (cento e trinta e cinco mil) subclasses distribuídas numa hierarquia de várias camadas, Cada subclasse pode conter até 2000 (duas mil) patentes, depois do qual novas subclasses são introduzidas. Muitas subclasses só contêm aproximadamente 20 (vinte) patentes (Fall & Benzineb, 2002).

Foi desenvolvida pela IBM, um sistema de classificação de patentes hierárquico restrito a 12 (doze) subclasses do sistema americano de classificação, organizado em 3 (três) níveis e referentes à Comunicação, Eletricidade e Eletrônica. Foi usado o algoritmo Bayesiano e a geração de pequenos conjuntos de palavras discriminatórias em cada nó da taxonomia hierárquica. A quantidade ótima de palavras dependeu da categoria e variou entre 160 (cento e sessenta) e 9130 (nove mil e cento e trinta) termos. Num teste envolvendo 500 (quinhentos) documentos de patente para treinamento e 300 (trezentos) documentos de patente para teste, em nível de subclasse, a abrangência média obtida foi de 66% para um sistema hierárquico. Não se tem conhecimento se foi usado o texto completo dos documentos. Comparando-se modelos de documentos que levaram em consideração a recorrência de palavras (modelo de Bernoulli) com o modelo Binário, o uso do primeiro modelo aumentou a exatidão em 8%. Comparações entre simulações envolvendo o banco de dados Reuters de novos artigos com o de patentes americano, para o primeiro foi alcançado uma exatidão de 87% para novos artigos, distribuídos em 30 (trinta) categorias e 66% de exatidão para

classificação de patentes, distribuídas em 12 (doze) categorias. Levando-se em consideração os termos dos documentos citados na patente ou os que citam a patente, o desempenho do categorizador piorou. No entanto, levando-se em consideração as categorias dos documentos citados ou os que as citam, o desempenho da simulação aumentou (Fall & Benzineb, 2002).

Segundo Fall et al. (2003a), Leah Larkey desenvolveu um categorizador de patentes baseado na classificação americana. Foram usadas: a remoção de *stopwords*; a stemização; e foi usado o algoritmo *k*-NN combinado com o algoritmo Bayesiano. O melhor desempenho foi alcançado quando foi usado para representar o documento de cada patente: o título; o Resumo; as 20 (vinte) primeiras linhas da primeira página do Resumo do Estado-da-Arte (em inglês *background summary*); e o Quadro Reivindicatório. O categorizador usou o algoritmo *k*-NN, comparando a *string* de busca com o conjunto de patentes mais similares de treinamento. Acredita-se que o categorizador não esteja em uso no escritório americano de patentes.

A Universidade da Califórnia desenvolveu um sistema automático de categorização com a sugestão de várias categorias para documentos de patente. Foi desenvolvido para uso com a classificação americana e para o IPC, através a entrada de uma *query*. A *string* da *query* é automaticamente comparada com os termos associados derivados do vocabulário controlado. Há opções para busca de frases ou palavras. São retornados 10 (dez) subgrupos (Fall & Benzineb, 2002).

Lingway, uma companhia francesa desenvolveu o Sistema TACSY (em inglês *Taxonomy Access and Coding System*) para aplicação ao sistema IPC. O usuário formula uma *query* através de uma sentença, no idioma francês, descrevendo o campo de interesse e o *software* sugere prognósticos de categorias correspondentes a IPC. É desenvolvida inicialmente a indexação lingüística da descrição das categorias, palavras polissêmicas são textualmente desambiguadas e palavras equivalentes são adicionalmente incluídas, sem se ater a nenhum documento. São sugeridas categorias. Testado pela EPO através 350 (trezentas e cinqüenta) *queries*, resultou numa abrangência média de 79% com 55% das classificações corretas aparecendo no topo das 20 (vinte) respostas dadas pelo sistema (Fall et al., 2003a).

Tikk & Biró (2003) desenvolveram um categorizador hierárquico para textos de patentes onde na fase de treinamento é comparado o texto a ser categorizado com os descritores das categorias. Ao texto a ser categorizado é designada à categoria do descritor mais similar. Quando a categoria selecionada não é a correta, aumenta-se os pesos dos termos do descritor da categoria encontrada e abaixa-se os pesos dos termos do descritor da categoria que deveria ter sido selecionada. Faz-se o procedimento até se encontrar pesos ótimos dos descritores das categorias ou não se conseguir mais melhorar o desempenho do método. Foram usadas a base de dados inglesa *WIPO-alpha* e a alemã *WIPO-de*. Os resultados foram diferenciados por nível de confidência. 0 (zero) significa que todas as suposições foram consideradas, enquanto 0.8 significa que somente foram levadas em consideração aquelas cujo nível de confidência não sejam menores que 0.8. Maior o nível de confidência, menor a quantidade de documentos a serem considerados. Foi levada em consideração para a categorização: inventores; depositantes; título; Resumo (em inglês *abstract*); e Reivindicações. A pesagem adotada foi a da TF.IDF e da entropia e a freqüência mínima dos termos foi de 2. Para nível de confidência 0, pesagem TF.IDF e para Prognóstico de Topo, o resultado foi de 64.04% e para Três Prognósticos de Topo o resultado foi de 70.71%.

Khattak & Heyer (2011) desenvolveram um estudo com relação da importância das palavras de pouca frequência nos textos de patentes para sua classificação. Segundo os autores a razão do baixo desempenho encontrado nos categorizadores é devido à inclusão de palavras ruidosas (em inglês *noisy words*) que são necessárias para discriminação entre as palavras dominantes. Para o estudo foi reduzido o tamanho do vocabulário considerando-se somente termos que tivessem frequência acima de um patamar (em inglês *threshold*) baseado em alguma frequência do documento na coleção toda.

Nos experimentos de Khattak & Heyer (2011) foi constatado que termos que ocorrem com baixa frequência mostraram ser termos dominantes e a inclusão desses termos pode aumentar a eficiência da categorização. Para o experimento foi usado para representar o documento o Modelo Espaço Vetorial - VSM (em inglês *Vector Space Model*) e depois aos termos foi dada uma pesagem, através de 3 (três) técnicas: TF.IDF (em inglês *Term Frequency versus Inverse Document*

*Freq*uency); BM25 (em inglês *Best Match*); e SMART (em inglês *System for Manipulating and Retrieving Text*). Para os dois últimos modelos, i.e., BM25 e SMART vide Khattak et al (2011). Os 4 (quatro) categorizadores usados foram: Naive Bayes (NB); *Support Vector Machine* (SVM); Árvores de Decisão (em inglês *Decision Trees*); e *k*-Vizinhos-Mais-Próximos (*k*-NN), para *k* igual a 1 e a 3. Foram levados em consideração no experimento somente o grupo principal e o Quadro Reivindicatório do documento. Foi realizada a stemização dos termos, sendo que todos os termos com menos de 5 (cinco) caracteres foram removidos. Foram obtidos 4351 (quatro mil, trezentos e cinquenta e um) termos de 1484 (hum mil, quatrocentos e oitenta e quatro) documentos, divididos em fase de treinamento (66%) e teste (34%). A base de dados foi obtida do http://www.freepatentsonline.com.

Nos experimentos de Khattak & Heyer (2011) foram feitas as seguintes suposições: termos com baixa frequência são os que ocorrem em mais que 10 (dez) documentos e menos que 101 (cento e um) documentos, tendo sido selecionados 847 (oitocentos e quarenta e sete) termos; termos com frequência normal são os que ocorrem em mais 100 (cem) documentos e menos que 201 (duzentos e um) documentos, tendo sido selecionados 110 (cento e dez) termos; alta frequência de termos são os que ocorrem em mais de 200 (duzentos) documentos, tendo sido selecionados 85 (oitenta e cinco) termos. Foi usado o *software* livre WEKA para o experimento. Constatou-se que termos com baixa frequência contribuem mais para o desempenho da categorização do que termos com frequência normal (obtido para 11 dos 15 casos para medida de desempenho *F-measure*). As exceções encontradas foram quando foi usado o método *k*-NN com pesagem BM25 e SMART, onde se constatou que termos com frequência normal contribuem mais para o desempenho da categorização do que termos com baixa frequência. O experimento também foi desenvolvido usando a biblioteca LIBSVM no octave visando à categorização de 4238 (quatro mil, duzentos e trinta e oito) documentos do banco de dados TREC referentes a patentes químicas. Textos foram extraídos de 21 (vinte e um) grupos principais, onde foram retiradas as *stopwords* e feita a stemização. Nesse caso foram feitas as seguintes suposições: termos que ocorrem em mais de 10 (dez) e menos que 101 (cento e um) documentos foram considerados como sendo de baixa frequência; e termos

que ocorrem em mais de 500 (quinhentos) e menos que 1001 (hum mil e um) documentos foram considerados como de alta frequência. Em todos os casos, constatou-se que termos com baixa frequência contribuem mais para o desempenho da categorização do que termos com alta frequência (Khattak & Heyer, 2011).

O único estudo específico para categorização de pedidos de patentes no idioma português foi realizado no INPI por meio de rede neural (INPI, 2009) e por meio do algoritmo caracterizador de Galho, Thais Silva (2003), no entanto, não foram conseguidos bons resultados.

2.3
Etapas do Processo de Categorização de Textos

Costuma-se dividir o processo de Categorização de Textos em 6 (seis) grandes etapas (Krishnakumar, 2006):

- Coleta de Documentos ou Base de Dados Textuais;
- Preparação de Dados Textuais ou Pré-Processamento;
-Transformação ou Seleção das Características (Indexação);
- Redução da Dimensionalidade;
- Extração de Conhecimento ou Processamento (Implementação do classificador);
- Avaliação e Interpretação dos Resultados (Medidas de Desempenho, Pós-Processamento).

Devido à natureza textual não-estruturada, os documentos necessitam de um pré-processamento para serem submetidos a algoritmos de aprendizagem. É muito importante, a transformação dos documentos textuais em uma representação mais adequada, como em uma tabela atributo-valor.

Na etapa de pré-processamento devem ser aplicadas algumas técnicas que facilitem o processo de seleção de características dos textos tais como: retirada de todas as palavras que não influenciam para a definição da categoria do texto (em inglês *stopwords*); retirada de símbolos; conversão de textos em radicais ou stemização; tratamento de termos compostos; entre outros (Silva & Galho, 2006).

Caso se use na etapa de processamento um algoritmo baseado em palavras ou lista de termos que definem sua categoria, devem ser localizadas, nos textos, todas as palavras que expressam melhor as características desses textos, ou seja, palavras que podem definir sua categoria. A partir dessa lista de termos, é gerada uma lista de termos comum a todos os documentos. Essa lista de termos compõe o índice que representará a categoria (Silva & Galho, 2006). Como alternativa para a representação das categorias podemos ter a construção de um *Thesaurus*, como por exemplo, a partir: do classificador IPC; do índice que representa a categoria; do conhecimento de um especialista; de sinônimos. Essas alternativas devem ser comparadas na etapa de Avaliação dos Resultados.

2.3.1
Coleta de Documentos ou Base de Dados Textuais

A primeira etapa do processo de categorização de textos consiste em recuperar documentos relevantes ao domínio de aplicação do conhecimento a ser extraído (Krishnakumar, 2006).

2.3.2
Preparação de Dados Textuais ou Pré-Processamento

A fase de Preparação dos Dados prepara o conjunto de dados textuais. Esse conjunto de dados deve representar a maior quantidade possível de características relevantes dos documentos (Krishnakumar, 2006).

A fase de Pré-Processamento dos dados textuais é constituída por quatro etapas (Krishnakumar, 2006).

- análise léxica;
- eliminação de *stopwords*;
- aplicação da stemização (em inglês *stemming*);
- uso do *thesaurus.*

2.3.2.1
Análise Léxica

Visa à retirada de tudo que não é significativo, tornando o texto mais curto, a lista de termos das categorias mais sucinta e o tempo de processamento mais curto (Krishnakumar, 2006).

Na análise léxica: eliminam-se os dígitos e os sinais de pontuações; isolam-se os termos; e efetua-se a conversão das letras minúsculas para maiúsculas.

Esse processo acelera comparações no processo de indexação.

Considere uma frase de um documento tomada como exemplo:

“Janeiro começa com grandes liquidações.”

Como a análise léxica consiste na limpeza dos textos, então a frase resultante da aplicação desta etapa encontra-se sem as aspas duplas e o ponto final, devido a esses não serem relevantes para o documento. Nessa etapa, geralmente se retiram os acentos.

JANEIRO COMEÇA COM GRANDES LIQUIDAÇOES

2.3.2.2 *Stopwords* ou Eliminação de Termos Considerados Irrelevantes

Um dos primeiros passos no processo de preparação dos dados é a identificação do que pode ser desconsiderado nos passos do processamento de dados, ou seja, a identificação dos termos frequentes em um texto que não carregam nenhuma informação de maior relevância (Dias & Malheiros, 2004). É a tentativa de retirar tudo que não constitui conhecimento nos textos. Nessa etapa, é formada uma lista contendo palavras a serem descartadas (*stopwords*), chamada de *Stoplist* (Silva, 2007; Krishnakumar, 2006).

Stopwords são palavras que não tem conteúdo semântico significativo no contexto em que ela existe. São palavras consideradas não relevantes na análise de textos, por tratar-se de palavras auxiliares ou conectivas (e, para, a, eles), que não fornecem nenhuma informação discriminatória na expressão do conteúdo dos textos (Guo et al., 2004; Dias & Malheiros, 2004). Na construção de uma lista de *stopwords* incluem-se palavras como artigos, preposições, pronomes, advérbios, conjunções, interjeições, consoantes, vogais, verbos (tais como: ser e estar) e outras classes de palavras auxiliares, pois sua presença pouco contribui para o

valor semântico do texto (Guo et al., 2004; Dias & Malheiros, 2004). *Stopwords* também podem ser palavras que apresentam uma incidência muito alta em uma coleção de documentos.

Normalmente, 40 a 50% do total das palavras de um texto são removidas através de uma *stoplist* (Silva, 2007). O prejuízo semântico dessa estratégia é perder a busca exata por compostos como por exemplo ferro de passar onde a preposição de não pode ser buscada.

A remoção dos termos irrelevantes visa à retirada dos termos de pouca importância para a representatividade dos documentos em um processo de Mineração de Textos. Com base na frase exemplo acima, o resultado da remoção dos termos irrelevantes ficaria:

JANEIRO COMEÇA GRANDES LIQUIDAÇOES

Nesta etapa, para o exemplo, a preposição com foi removida.

2.3.2.3 Normalização Morfológica dos Termos (Remoção de Sufixos) ou Stemização (em inglês *Stemming*) ou Radicalização

A etapa referente à normalização morfológica reduz os termos restantes da etapa anterior aos seus radicais, de forma a agrupar por similaridade variações ortográficas que de outra forma passariam como palavras completamente distintas (Dias & Malheiros, 2004).

Radicalização (em inglês *stemming*) é o processo de combinar formas diferentes de uma palavra em uma representação comum, o radical (em inglês *stem*). Radical é um conjunto de caracteres resultante de um processo de radicalização, permitindo tratar variações diferentes de uma palavra da mesma forma. Por exemplo, conector e conectores são essencialmente iguais, mas sem sofrerem a redução por radicalização serão tratadas como palavras distintas (Dias & Malheiros, 2004).

A vantagem do processo de stemização é identificar similaridades em função da morfologia das palavras, dessa forma reduzindo o número de atributos de um texto. Por exemplo, para a frase exemplo, a normalização morfológica seria:

JANEIR COMEÇ GRANDE LIQUIDAÇ

Observa-se que os termos apresentam-se reduzidos ao seu radical, possibilitando que termos com o mesmo radical sejam considerados unicamente.

A normalização morfológica dos termos (remoção de sufixos) também é denominada de stemização (em inglês *Stemming*).

Na categorização ou classificação de documentos, a variação morfológica pode ser extremamente importante, pois aumenta a discriminação entre documentos, devido à redução de variantes de um mesmo radical para um mesmo conceito.

No processo de normalização morfológica de termos (em inglês *stemming*) cada palavra é considerada isoladamente, tentando reduzi-la a sua provável palavra raiz, eliminando sufixos, indicando formas verbais e/ou plurais. Algoritmos de stemização empregam a lingüística e são dependentes do idioma.

Muitos algoritmos de stemização no idioma inglês foram descritos na literatura: Paice (1983); Lovins (1968); Dawson (1974); Porter (1980).

Para o idioma português temos entre outros: Normalização de Gonzalez et al. (2006); Porter *Stemmer* (1980); *PortugueseStemmer* de Orengo & Huyck (2001b).

A maioria dos algoritmos de stemização não leva em consideração o sentido correto de cada termo, sendo que a sua maioria pode ser considerada como apresentando um significado único. Podemos exemplificar com: "A alta tensão na rede ocasionou avarias nos aparelhos elétricos"; "O uso não correto da palavra mudou o sentido da frase".

Os erros mais comuns associados ao processo de stemização podem ser divididos em dois grupos (Dias & Malheiros, 2004; Lopes, 2004):

- *overstemming* – acontece quando a cadeia de caracteres removida não era um sufixo, mas parte do *stem*. Por exemplo, a palavra confortável após ser processada por um radicalizador, é transformada no radical confor-. Nesse caso o radicalizador removeu parte do radical correto, a saber, confort-;

- *understemming* – acontece quando um sufixo não é removido completamente. Por exemplo, a palavra referência é transformada no radical referênc-, ao invés de ser transformado no radical certo refer-.

A stemização (em inglês *stemming*) não tem sucesso com termos onde a flexão é raramente usada ou inexiste (por exemplo, nomes próprios). A análise flexional e morfológica de termos compostos também é problemática, mesmo para o idioma inglês. Uma solução, para esse casos, é a decomposição do termo e a aplicação da normalização, separadamente, a cada componente (Gonzalez, 2005).

Alguns significados podem ser perdidos na stemização e palavras de famílias de significados diferentes podem ser agrupadas. Seria o caso (Gonzalez, 2005): stemização (livro) = stemização (livre) = livr; stemização (caminhada) = stemização (caminhão) = caminh.

O *stemmer* de Lovins se refere a uma tabela de 294 (duzentos e noventa e quatro) terminações que depois de serem removidas, a forma truncada ou restante é submetida a uma regra, tais como: -iev é alterada para -ief. O *stemmer* de Dawson usa uma lista de aproximadamente 1200 (hum mil e duzentos) sufixos, organizada em um conjunto de árvores de caracteres ramificada visando um rápido acesso. O algoritmo de Porter se baseia em 5 (cinco) etapas, com uma tabela sendo usada por vez (Paice, 1983).

O algoritmo de Normalização para extração dos sufixos deve ser simples e efetivo com o propósito de melhorar a Abrangência (em inglês *Recall*), sem diminuir a Precisão (em inglês *Precision*) (Orengo, 2001a; Orengo, 2001b).

2.3.2.3.1
Algoritmo de Stemização (em inglês *Stemming*) de Porter

O radicalizador de Porter foi desenvolvido por Martin Porter na Universidade de Cambridge em 1980 e é uma adaptação de um algoritmo desenvolvido inicialmente para a Língua Inglesa (Dias & Malheiros, 2004).

Esse radicalizador tem 5 (cinco) passos ou etapas, sendo aplicadas regras dentro de cada passo. Em cada passo, se uma regra de sufixo coincide com uma palavra, então as condições associadas àquela regra são testadas sobre o que seria

o radical resultante, caso aquele sufixo fosse removido. Por exemplo, tal condição poderia ser que a quantidade de vogais (que são seguidas por consoantes no radical) seja maior do que 1 (um) para a regra a ser aplicada. Uma vez que as condições para uma regra sejam satisfeitas, o sufixo é removido e o controle vai para o próximo passo. Se a regra não é satisfeita, então a próxima regra no passo é testada, até que uma outra regra desse passo seja aceita ou até que não existam mais regras, quando se prossegue para o passo seguinte. Esse processo continua por todos os 5 (cinco) passos, retornando o radical resultante após a execução do quinto passo (Dias & Malheiros, 2004).

O algoritmo de stemização de Porter consiste da identificação das diferentes inflexões referentes à mesma palavra e sua substituição por um mesmo *stem*. Temos como exemplo (Lopes, 2004):

CONSIDERAR / CONSIDERADO/ CONSIDERAÇÃO/ CONSIDERAÇÕES, o qual é reduzido ao radical **CONSIDER**.

Esse algoritmo faz uso das definições de regiões R1, R2 e RV, definidas como a seguir (Lopes, 2004):

- R1 é a região depois da primeira não-vogal seguindo a vogal (excluindo-se a primeira letra da palavra), ou é a região vazia no fim da palavra se não existe tal não-vogal;

- R2 é a região depois da primeira não-vogal seguindo a vogal em R1 ou é a região vazia no fim da palavra se não existe tal não-vogal.

A seguir, exemplos de R1 e R2 são mostrados para algumas palavras em português (Lopes, 2004):

```
B a i l a r i n a
      |<---------->| R1
        |<----->| R2
```

Excluindo-se a primeira letra da palavra, a letra l é a primeira não-vogal seguindo a vogal em bailarina, assim R1 é arina. Em R2, r é a primeira não vogal seguindo a vogal. Assim R2 é ina (Lopes, 2004).

```
B a i l e
      |<->| R1
        >|< R2
```

Excluindo-se a primeira letra da palavra, a letra l é a primeira não-vogal seguindo a vogal, assim R1 é apenas a última letra e e R2 não contém nenhuma não-vogal, assim R2 é a região vazia ao final da palavra (Lopes, 2004).

```
B ó i a
    >|< R1
    >|< R2
```

Em bóia R1 e R2 são ambos nulos (Lopes, 2004).

```
A m i g á v e l
      |<-------->| R1
          |<->| R2
```

Excluindo-se a primeira letra, a letra g é a primeira não-vogal seguindo a vogal em amigável, assim R1 é ável. Em R2, v é a primeira não-vogal seguindo a vogal. Assim R2 é el (Lopes, 2004).

```
I n d e p e n d e n t e
        |<---------------------------->| R1
              |<---------------------->| R2
```

Excluindo-se a primeira letra da palavra, a letra p é a primeira não-vogal seguindo a vogal em independente, assim R1 é endente. Em R2, n é a primeira não-vogal seguindo a vogal. Assim R2 é dente (Lopes, 2004).

As regiões R2 e RV têm a mesma definição que no *stemmer* para o idioma espanhol. O algoritmo faz uso também da definição de região RV como a seguir (Lopes, 2004):

- se a segunda letra é uma consoante, RV é a região depois da próxima vogal a seguir;

- se as primeiras duas letras são vogais, RV é a região depois da próxima consoante;

- caso as duas primeiras letras são consoante-vogal, RV é a região depois da terceira letra;

- mas RV é o fim da palavra se essas posições não puderem ser encontradas.

Por exemplo (Lopes, 2004):

m a c h o o l i v a t r a b a l h o á u r e o

|.....| |.....| |.....| |.....|

Em macho, como as duas primeiras letras são consoante-vogal, RV é ho, ou seja, a região depois da terceira letra c.

Em oliva, como a segunda letra é uma consoante, RV é va, ou seja, a região depois da próxima vogal a seguir i.

Em trabalho, como a segunda letra é uma consoante, RV é balho, ou seja, a região depois da próxima vogal a seguir a.

Em áureo a letra r é a primeira consoante seguindo as duas vogais iniciais, assim RV é eo.

De forma a melhorar os resultados, podem ser realizadas as seguintes modificações no algoritmo de Porter original (Lopes, 2004):

- RV = R1;
- sufixos ordenados pelo comprimento.

2.3.2.3.2 Algoritmo de Stemização (em inglês *Stemming*)– *StemmerPortuguese* (*RSLP*)

O algoritmo de *StemmerPortuguese* é um removedor de sufixos para a Língua Portuguesa (RSLP), desenvolvido por Viviane Orengo e Christian Huyck em 2001 (Dias & Malheiros, 2004) (Orengo, 2001a, 2001b) (Lopes, 2004).

Orengo (2001a, 2001b) na aplicação do algoritmo *StemmerPortuguese*, reformulou-o acrescentando listas de exceções, pois se não fossem usadas tais listas, o radicalizador poderia gerar erros de *overstemming* se a regra estivesse presente e erros de *understemming* se a regra fosse retirada (Dias & Malheiros, 2004).

A lista de exceções consiste de: palavras com grafia igual e significados diferentes (ex: casais); verbos irregulares (menos que 1% de erros); mudanças no radical (emitir – emissão); tratamento de nomes próprios, por exemplo, Pereira (Dias & Malheiros, 2004).

Esse algoritmo leva em consideração as classes morfológicas, executando uma série de passos de remoção de sufixos conhecidos. Os passos são aplicados na seguinte sequência (Dias & Malheiros, 2004; Orengo, 2001a, 2001b; Lopes, 2004):

1. Palavra termina em s? Faça redução do plural;

2. Palavra termina em a? Faça redução do feminino;

3. Redução do advérbio;

4. Redução do aumentativo e do diminutivo;

5. Redução das formas nominais;

6. Redução das terminações verbais (raiz + vogal temática + sufixo temporal + desinência pessoal);

7. Redução da vogal temática (caso não stemizada pelos passos 5 e 6);

8. Remoção dos acentos.

O algoritmo RSLP é composto de 8 (oito) passos, sendo que para cada passo há um conjunto de regras, somente podendo ser aplicada uma regra para cada passo e o sufixo mais longo é sempre removido primeiro por causa da ordem das regras dentro de um dado passo. Em cada regra é estabelecido: o sufixo a ser removido; o comprimento mínimo permitido para o *stem*; sufixo a ser reposto, se necessário; e lista de exceções (Orengo, 2001a, 2001b; Dias & Malheiros, 2004; Lopes, 2004).

No trabalho desenvolvido por Dias & Malheiros (2004) foram executadas comparações entre o algoritmo de radicalização *StemmerPortuguese* em relação ao radicalizador de Porter, tendo sido usadas 919 (novecentos e dezenove) dissertações de teses da Biblioteca Digital da UNICAMP. Foram retirados dos textos as fórmulas e gráficos. Na primeira comparação foi formada uma lista (lista bruta) com todos os termos distintos constantes das 919 (novecentos e dezenove) teses, resultando em 602014 (seiscentos e dois mil e quatorze) termos distintos. Usando o radicalizador de Porter foi obtido uma redução de 20% e usando o *StemmerPortuguese* foi obtida uma redução de 27%. Usando uma lista formada com os termos mais frequentes da lista bruta, foram obtidos 32000 (trinta e dois mil) termos, sendo que para o radicalizador Porter foi obtida uma redução de 43%

e usando-se o *StemmerPortuguese* foi obtida uma redução de 48%. Em outra comparação foram usadas 30 (trinta) teses de diversas áreas e quando aplicado o radicalizador de Porter foi obtida uma redução de 12.1% e para o *StemmerPortuguese* foi obtida uma redução de 15.1%.

2.3.2.3.3
Algoritmo de Itens Lexicais Baseada em Sufixos

De acordo com o artigo "Normalização de Itens Lexicais Baseada em Sufixos" de Gonzalez et al. (2003), inicialmente o texto é etiquetado com categorias morfológicas e cada palavra com sua etiqueta é tratada através de um autômato especializado na categoria morfológica identificada. É pesquisado o sufixo em uma das seguintes bases de sufixos: adjetivos (incluindo adjetivos e verbos no particípio); artigos (definidos e indefinidos); numerais (cardinais e ordinais); pronomes (pessoais, demonstrativos, possessivos, indefinidos, relativos); substantivos; e verbos (exceto os verbos no particípio). Os verbos no particípio são tratados na base de adjetivos por terem um comportamento semelhante a estes, quanto à normalização morfológica. Após o reconhecimento do sufixo é definida a ação a ser tomada: inclusão de caracteres; exclusão de caracteres; exclusão seguida de inclusão de caracteres; ou nenhuma delas. Para algumas palavras, a normalização é otimizada em duas etapas que podem ser, por exemplo, do plural para o singular e, após, do feminino para o masculino. As palavras invariáveis, como as preposições e as conjunções, são mantidas na forma original sem passar pela análise dos autômatos.

Por exemplo, no autômato para normalização morfológica de artigos, se for pesquisado o artigo uns será encontrado o s final, depois o n e depois um asterisco, indicando a aceitação de qualquer caractere (ou qualquer conjunto de caracteres). A próxima transição indica "-ns+m", o que significa a exclusão de ns e a inclusão de m. Chega-se assim ao estado final, com a palavra uns sendo normalizado como um (Gonzalez et al., 2003).

2.3.2.3.4
Algoritmo de Lematização

Outra técnica chamada de Lematização é a redução à forma canônica, consistindo em converter os verbos para o infinitivo e os substantivos e adjetivos para sua forma masculina singular. A palavra reduzida dessa forma recebe a denominação de lema ou forma canônica (Dias & Malheiros, 2005; Gonzalez, 2005).

Essa representação gráfica das palavras ocorre nos dicionários, pois todas as ocorrências de uma palavra são reunidas sob uma única forma, em vez de apresentá-las tal como aparecem nos textos, com variações no gênero, no número ou na grafia (Dias & Malheiros, 2005).

A principal diferença entre a stemização e a lematização é que na lematização a categoria morfológica é mantida e na stemização palavras de diferentes categorias morfológicas podem ter o mesmo *stem* (Gonzalez, 2005).

Tanto para a stemização, quanto para a lematização, seus benefícios são reconhecidos para a Recuperação de Informações (RI). Ambos os processos reduzem a quantidade de descritores e economizam no espaço necessário de memória para armazená-los (Gonzalez, 2005).

São exemplos (Gonzalez, 2005): lematização (livre) = lematização (livres) = livre; lematização (caminhar) = lematização (caminhei) = caminhar; lematização (construiu) = construir ≠ lematização (construções) = construção.

Experimentos nos idiomas espanhol e finlandês concluíram que a lematização produz na RI, melhores resultados que a stemização (em inglês *stemming*). Entretanto algumas palavras pertencentes à mesma família de significados podem não ser normalizados (Gonzalez, 2005): lematização (livre) = livre ≠ lematização (liberdade) = liberdade; lematização (caminhei) = caminhar ≠ lematização (caminhada) = caminhada.

2.3.2.3.5
Algoritmo de Stemização (em inglês *Stemming*) Simplificada

Nesse algoritmo é usada a estratégia da identificação dos atores mais difundidos, através da marcação de todos os sujeitos e complementos verbais, que na maioria das vezes são substantivos.

A solução dada para uma primeira etapa é converter todo o texto para letras minúsculas, sem acentos, eliminando os indicadores que caracterizam o singular ou o plural, que na língua portuguesa são: a(s), e(s), i(s), o(s), u(s), al(is), ão(ãos, ães, ões), el(is), il(s), z(es), m(ns). A localização de substantivos se dará pela localização de palavras antecipadas por artigos, preposições e contrações de ambos. Também se devem substituir pronomes relativos pelo símbolo antecedente encontrado pelo método, computando suas ocorrências no símbolo usado para substituí-lo.

Para uma segunda etapa, deve ser definido um limiar que será um percentual que aplicado ao símbolo mais frequente, dará o limite mínimo de ocorrências de um símbolo para que ele seja considerado relevante, devendo os outros serem descartados.

Essa abordagem de localização de substantivos pode selecionar verbos e adjetivos, mas levando-se em consideração que um documento geralmente se utiliza de uma descrição formal e estruturada, se torna muita baixa à ocorrência de locuções e de sujeitos compostos de muitas palavras, sendo a primeira um adjetivo. Esse problema é contornado através o uso do limiar.

Para a terceira etapa, deve-se também selecionar sentenças onde ocorrem palavras como deve, deveria, deverá, deverão, devem, deveriam, mesmo que não contenham um símbolo selecionado, pois em grande parte dos documentos tais palavras estão relacionadas a algo que seja um requisito funcional.

Quando for encontrado o primeiro artigo da sentença, deverão serão ignoradas todas as palavras seguidas de preposição antes de ser encontrado algum artigo, pois na língua portuguesa, não se têm sujeitos preposicionados. Assim serão eliminadas: de acordo com; de vez em quando.

2.3.2.4
Uso do Dicionário ou *Thesaurus*

Usuários definem a mesma *query* usando termos diferentes. Esse problema é resolvido usando-se um *thesaurus* (dicionário) podendo ser definido como um vocabulário controlado que representa sinônimos, abreviações, acrônimos,

hierarquias e relacionamentos associativos entre termos, que ajudam os usuários a encontrar a informação de que eles precisam (Lopes, 2004).

O mapeamento dos termos variantes são representados por um termo preferido único para cada conceito, geralmente generalizados para o termo de mais alto nível, de acordo com a hierarquia de conceitos descrita no *thesaurus*. Para processos de indexação de documentos, o *thesaurus* informa que termos índices devem ser usados para descrever cada conceito (Lopes, 2004).

Um *thesaurus* pode representar também a riqueza de relacionamentos associativos e hierárquicos. Usuários podem expressar a necessidade de informação com um nível de especificidade mais restrito ou mais amplo que o usado pelo indexador para descrever os documentos. O mapeamento de relacionamentos hierárquicos endereçam este problema (Lopes, 2004).

2.3.2.4.1
Termos Compostos

Existe *Thesaurus* que consideram a utilização de termos compostos, nos casos de palavras que aparecem sempre juntas e que quando se reúnem apresentam um significado diferente que cada uma delas tem separadamente (Lopes, 2004).

O uso de descritores consistindo de mais de uma palavra são aceitáveis somente se o termo composto expressa um conceito único, expresso pela associação dos termos considerados (Lopes, 2004).

Segundo Santos (2002) pode-se citar: cabelo branco; pele branca; vinho branco; onde o branco do cabelo é cinza; o branco da pele é rosada; e o branco do vinho é amarelado.

2.3.2.4.2
Relacionamento entre Termos

Podem-se ter vários tipos de relacionamentos no *thesaurus:* de equivalência (sinônimos); de hierarquia (termo amplo e termo restrito); de associação (termo relacionado) (Lopes, 2004).

O relacionamento hierárquico é a primeira característica que distingue um *thesaurus* de uma lista de termos não estruturada, como um glossário (Lopes, 2004).

Ele é baseado em graus ou níveis de superordenação ou subordenação. O descritor superordenado representa uma classe ou um todo; o descritor subordenado refere-se aos membros ou partes de uma classe (Lopes, 2004).

No *thesaurus*, relacionamentos hierárquicos, para o descritor superordenado são representados pelo rótulo termo amplo (em inglês *broader term*) e para o descritor subordenado são representados pelo rótulo termo restrito (em inglês *narrower term*) (Lopes, 2004).

O relacionamento hierárquico cobre três situações logicamente diferentes e mutuamente exclusivas: o relacionamento genérico; o relacionamento modelo; o relacionamento todo-parte. Cada descritor subordenado deve se referir ao mesmo tipo de conceito que o seu descritor superordenado, ou seja, ambos os termos amplo e restrito devem representar um objeto, uma ação, uma propriedade, etc (Lopes, 2004).

O relacionamento genérico identifica a ligação entre a classe e seus membros ou espécies. Nesse tipo de relacionamento a seguinte afirmação sempre pode ser aplicada: termo restrito é um termo amplo (Lopes, 2004).

O relacionamento modelo identifica a ligação entre uma categoria genérica de coisas ou eventos, expressos por um nome comum e um modelo individual daquela categoria, frequentemente um nome próprio (Lopes, 2004).

O relacionamento todo-parte cobre situações na qual um conceito é incluído por herança em outro, independentemente do contexto, de forma que os descritores podem ser organizados em hierarquias lógicas, com o todo sendo tratado como um termo amplo. Alguns exemplos são órgãos do corpo, sistemas ou pontos geográficos (Lopes, 2004).

2.3.3
Transformação dos Dados

Após a eliminação da lista de *stopwords*, utilização do *Thesaurus* e uso do processo de stemização, obtém-se um conjunto de dados reduzido em relação ao original, conhecida como *bag of words* (*bow*) a qual pode ser facilmente convertida em tabelas (Lopes, 2004).

2.3.3.1 Modelo de Espaço-Vetorial

O Modelo Espaço Vetorial (VSM em inglês *Vector Space Model*) é uma das técnicas mais usadas em Mineração de Textos, sendo a aplicação mais comum à categorização automática de documentos. No contexto do tratamento de documentos, o objetivo principal de um modelo de representação é a obtenção de uma descrição adequada da semântica do texto, de uma forma que permita a execução correta da tarefa alvo, de acordo com as necessidades do usuário (Gean, 2004).

Nesse modelo, o documento é conceitualmente representado por um conjunto de termos extraídos da coleção de documentos, criando um espaço Euclidiano m-dimensional, onde m é equivalente ao número de termos da coleção e cada dimensão possuindo um peso associado ao termo, significando a importância do mesmo para o documento (Gomes & Costa, 2005). Geralmente os pesos são obtidos a partir de uma função relacionada à freqüência dos termos no documento e na coleção.

Formalmente, seja $C = (d_1, d_{2,}, d_3, ..., d_n)$ uma coleção qualquer não-ordenada de documentos d_i, contendo n diferentes termos. Então a representação de um documento será $d_i = (f_{i1}, f_{i2}, f_{i3}, ..., f_{in})$ para $i = 1$ até n, onde f_{ij} é uma função de avaliação associada ao termo j no documento i.

Uma função de avaliação (ou peso) f_{ij} bastante utilizada é a freqüência linear das palavras (TF.IDF). Cada termo diferente adiciona uma nova dimensão ao problema. Problemas de Mineração de Textos costumam apresentar dimensões elevadas. Cada documento será então representado por um número de m dimensões indicando a ocorrência do termo no texto.

2.3.3.2
Seleção das Características ou Indexação Automática

A Seleção de Características ou Indexação Automática é a tarefa de definir um conjunto de termos que melhor representem o assunto a ser categorizado. Os documentos de texto podem envolver centenas de características, a maioria das vezes sendo irrelevantes.

A indexação permite que se procure eficientemente em textos por documentos relevantes a uma *query* sem precisar examinar os documentos inteiros (Lopes, 2004).

Os tipos mais comuns de indexação são: a Indexação de Texto Completo; e a Indexação Temática. Existe ainda: a Indexação Tradicional; a Indexação por *Tags*; a Indexação Semântica Latente; e a Indexação por Listas Invertidas (Lopes, 2004).

A Indexação de Texto Completo ocorre automaticamente em várias ferramentas de análise de textos quando os documentos são carregados. Índices guardam informação sobre a localização dos termos dentro do texto, de forma que operadores de proximidade, assim como operadores booleanos possam ser utilizados em *queries* no texto completo. Os operadores mais comuns em *queries* de texto são: *AND*, *OR*, *NOT* (operadores booleanos); *NEAR*, *WITHIN* (operadores de proximidade) (Lopes, 2004).

A Indexação Temática depende do uso do dicionário. O *Thesaurus* é um conjunto de termos que define um vocabulário e é montado usando-se relacionamentos. Ele fornece uma estrutura hierárquica que permite as ferramentas de Mineração de Textos encontrar rapidamente generalizações assim como termos específicos (Lopes, 2004).

A estrutura de um *Thesaurus* mais comumente utilizada para Indexação Temática consiste dos seguintes componentes principais: *Thesaurus*; termo indexador; termo preferido; termo não-preferido (Lopes, 2004).

Na Indexação por *Tags*, algumas partes do texto são selecionadas automaticamente para fazer parte do índice. Para a Indexação por *Tags*, normalmente adota-se o uso de gramáticas *parsers* e expressões regulares para a

definição e reconhecimento das *tags*. As palavras chaves são extraídas com base nessas *tags* (Lopes, 2004).

Na Indexação por Listas ou Arquivos Invertidos, um arquivo invertido contém, para cada termo que aparece no banco de dados, uma lista contendo os números dos documentos contendo aquele termo. Para processar uma consulta ou *query*, um vocabulário é usado para mapear cada termo da *query* para o endereço da lista invertida; as listas invertidas são lidas a partir do disco; e as listas são mescladas, considerando a interseção dos conjuntos de números dos documentos em operações AND, a união em operações OR, e o complemento em operações NOT (Lopes, 2004).

2.3.3.2.1
Indexação de Documentos ou Métrica Definidora de Importância

As técnicas de indexação ou métrica definidora de importância mais usadas são: a pesagem *Booleana*; a pesagem por frequência de termos (TF); a pesagem TF.IDF; a pesagem de documentos inversa (IDF); a pesagem por Entropia; a pesagem por escore de relevância, etc.

A indexação de documentos denota a atividade de mapear um documento d_j, em uma representação compacta do seu conteúdo que pode ser diretamente interpretada por um algoritmo categorizador ou classificador. Na técnica de indexação de documentos geralmente um texto d_j é tipicamente representado como um vetor de pesos:

vetor $d_j = \{w_{1j},...w_{Tj}\}$ onde T é o conjunto de termos que ocorre em pelo menos k documentos.

A técnica de indexação é caracterizada: pela definição do termo (i); e pela definição do método para computar os pesos dos termos (ii). Com relação à (i) a escolha mais frequente é identificar os termos ou com as palavras ocorrendo no documento (com exceção dos *stopwords* tais como artigos, preposições, os quais são eliminados na fase de pré-processamento) ou com suas *stems* (suas raízes morfológicas, obtidas pela aplicação do algoritmo de stemização).

Quanto aos pesos eles podem ter valores binários, i.e. $w_{kj} \varepsilon \{0,1\}$ ou valores reais $0 \leq w_{kj} \leq 1$. Quando os pesos têm representação binária eles representam à presença ou ausência do termo no documento. Quando os termos são não-binários eles são computados ou por técnica probabilística ou por estatística, sendo o primeiro o mais comum (Sebastiani, 2002). Uma função de peso de termos estatísticos é representada por TF.IDF: onde mais frequente o termo t_k ocorre no documento d_j, mais importante para d_j ele é (frequência); mais documentos ocorrem com t_k, menos discriminatório ele é, i.e., menor sua contribuição é em caracterizar a semântica do documento em que ele ocorre (frequência inversa). Pesos computados pela técnica TF.IDF são sempre normalizados para contrastar sua tendência em enfatizar documentos longos.

2.3.4
Redução da Dimensionalidade

Um dos maiores problemas de Mineração de Textos é lidar com a alta dimensionalidade se considerado um espaço-vetorial, onde se cada termo representa uma dimensão, teremos tantas dimensões quanto termos diferentes.

Em Categorização de Textos quase sempre é aplicada uma fase de redução dimensional do documento.

Na Categorização de Textos geralmente um documento é representado por conjunto de palavras desconsiderando a gramática ou a ordem dos termos. Essa representação é chamada de modelo de *bag of words* (*bow*). Desde que um conjunto de documentos pode conter milhares de termos, uma representação de um documento a partir de um modelo de *bag of words* pode ter uma alta dimensionalidade (Tasci & Güngor, 2009).

Uma estratégia bastante utilizada para redução da dimensionalidade é a utilização da Lei de Zipf (Zipf, 1949) e corte de Luhn (Luhn, 1958).

A Lei de Zipf diz que se f é a frequência de ocorrência de qualquer termo do texto e r é a posição da ordenação com relação aos outros termos, então o produto $f \times r$ é aproximadamente constante. Luhn propôs que em um gráfico f *versus* r, pode-se definir um limite superior e um limite inferior de corte. As

palavras que estiverem fora do intervalo são excluídas da análise. Na Figura 1 são mostradas as curvas de Zipf e os Cortes de Luhn (Zipf, 1949; Luhn, 1958).

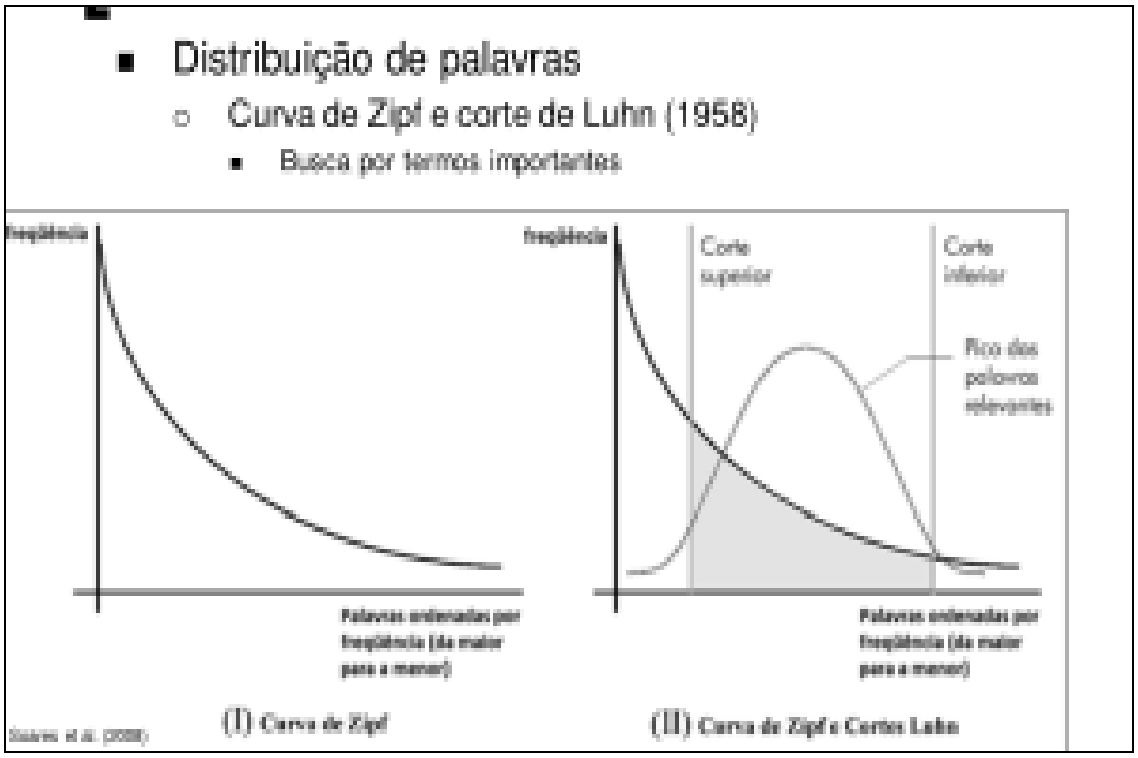


Figura 1 – Curva de Zipf e Corte de Luhn

Especialmente na Categorização de Textos, a alta dimensionalidade da quantidade de termos dos documentos pode ser problemático. A redução da dimensionalidade (em inglês *Dimensionality Reduction* – RD) tende a reduzir o *overfitting*, que é o fenômeno em que o categorizador é tão otimizado que características particulares dos dados de treinamento tornam-se relevantes para as categorias. O categorizador é ajustado de forma muito específica para o conjunto de treinamento, implicando em um baixo desempenho na categorização dos documentos não conhecidos pelo categorizador. Categorizadores que *overfit* os dados de treinamento são bons em reclassificar os dados nos quais eles foram treinados, mas são piores para classificação de dados novos. Experimentos mostraram que para evitar *overfitting* é necessária uma quantidade de exemplos de treinamento proporcionais a quantidade de termos usados. Foi sugerido que entre 50 (cinqüenta) a 100 (cem) exemplos de treinamento por termo pode ser necessário na tarefa de categorização. Isto significa que se a redução é

desempenhada, *overfitting* pode ser evitado mesmo se uma quantidade pequena de exemplos de treinamento é usado. Entretanto, na redução da dimensionalidade temos o risco de remover potencialmente informação útil. Então o processo de redução deve ser desempenhado com cuidado (Sebastiani, 2002).

Para a Redução da Dimensionalidade (RD) pode-se usar a técnica da Seleção de Características (em inglês *Feature Selection*); onde todas as características do documento são *rankeadas* segundo uma métrica estimando sua importância e então as com maiores *ranks* são selecionadas. As métricas de seleção são: Ganho de Informação (IG em inglês *Information Gain*); estatística *Chi-Square* (Chi em inglês *Chi-square statistics*); e freqüência de documentos (em inglês *Document Frequency* - DF). As duas primeiras métricas são supervisionadas (i.e. necessitam de um conjunto de treinamento) enquanto a última DF é uma métrica não-supervisionada (Tasci & Güngor, 2009).

A redução da dimensão pode também usar a técnica de Extração de Características (em inglês *Feature Extraction*), onde um conjunto de termos artificiais é gerado do conjunto do termo original de tal maneira que os termos mais novos gerados são ambos de menor quantidade e estocasticamente mais independentes do que cada um do original usado.

As técnicas da Redução da Dimensionalidade de base de dados textuais, segundo Krishnakumar (2006) são:

- Limiarização da Frequência do Documento (em inglês *Document Frequency Thresholding*);
- Ganho de Informação (em inglês *Information Gain*);
- Estatística *Chi-Square* X^2;
- Indexação Semântica Latente (LSI, em inglês *Latent Semantic Indexing*).

Segundo Sebastiani (2002), ainda existem como técnicas de redução de dimensionalidade: coeficiente NGL (em inglês *NGL coefficient*); Informação Mútua (em inglês *Mutual Information*); *odds ratio*; Escore de Relevância (em inglês *Relevancy Score*); coeficiente GSS (em inglês *GSS coefficient*).

2.3.4.1
Ganho de Informação

A métrica de Ganho de Informação (GI) baseada em entropia é adequada para se achar características relevantes em termos de sua capacidade de discriminação entre categorias. Nessa técnica deve-se remover cada característica que ocorre somente uma vez, pois a mesma não aumenta a razão de erro e deve ser usada como filtro antes de se computar o Ganho de Informação. Entretanto esse método não permite remoção de características baseadas na interação entre características. Esse método de seleção de características reduz agressivamente o tamanho do vocabulário usando a interação de características.

Ganho de Informação é empregado como um critério de importância do termo no campo do Aprendizado de Máquina. Ele mede o número de partes de informação obtidas para predição da categoria, pela presença ou ausência de um termo em um documento. Devido ao conjunto de documentos, o Ganho de Informação é calculado para cada termo, e os termos cujos Ganhos de Informação são menores que um determinado limite são retirados do espaço das características. Este cálculo inclui as estimativas das probabilidades condicionais de uma categoria dado um termo e o cálculo da entropia na definição (Lopes, 2004).

A métrica de Ganho de Informação baseada na entropia faz uso da redução através do conhecimento da existência ou da ausência do termo no documento (Tasci & Güngor, 2009).

$$IG(t_k, c_i) = \sum_{c \in (c_i, \bar{c}_i)} \sum_{t \in (t_k, \bar{t}_k)} P(t, c) \times \log\left[\frac{P(t_k, c_i)}{P(t_k) \times P(c_i)} \right] \quad (1)$$

onde,

$P(t_k, c_i) =$ Probabilidade de t_k e c_i co-ocorrerem;

$P(t_k)$ = Probabilidade da presença do termo t_k ;

$P(c_i)$ = Probabilidade da categoria c_i ocorrer.

2.3.4.2 Estatística *Chi-Square* (X^2)

A estatística *Chi-Square* (X^2) mede a independência de duas variáveis randômicas. Para a categorização de textos, as duas variáveis randômicas são a ocorrência do termo t e a ocorrência da categoria c (Tasci & Güngor, 2009).

$$x^2(t_k, c_i) = N \frac{\left[P(t_k, c_i)P(\bar{t}_k, \bar{c}_i) - P(\bar{t}_k, c_i)P(t_k, \bar{c}_i)\right]^2}{P(t_k)P(\bar{t}_k)P(c_i)P(\bar{c}_i)} \quad (2)$$

onde

$P(t_k, c_i)$ Percentagem de documentos pertencendo à categoria c_i em que o termo t_k ocorre (t_k e c_i co-ocorrem);

$P(\bar{t}_k, \bar{c}_i)$ Percentagem de documentos não pertencendo à categoria c_i em que o termo t_k não ocorre (nem t_k e nem c_i ocorrem);

$P(\bar{t}_k, c_i)$ Percentagem de documentos pertencendo à categoria c_i em que o termo t_k não ocorre (c_i ocorre sem t_k);

$P(t_k, \bar{c}_i)$ Percentagem de documentos não pertencendo à categoria c_i em que o termo t_k ocorre (t_k ocorre sem c_i);

N é a quantidade de documentos na base de dados;

$P(t_k)$ número de vezes que o termo t_k ocorre;

$P(c_i)$ número de vezes que a categoria c_i ocorre;

$P\left(\bar{t}_k\right)$ número de vezes que o termo t_k ocorre sem a categoria c_i mais o número de vezes que nem c_i nem t_k ocorrem;

$P\left(\bar{c}_i\right)$ número de vezes que a categoria c_i ocorre sem o termo t_k mais o número de vezes que nem c_i nem t_k ocorrem.

e onde a estatística (X^2) tem o valor igual a zero se t_k e c_i são independentes (Lopes, 2004).

2.3.4.3 Freqüência do Documento (DF)

A métrica da Frequência do Documento (DF) é uma métrica muito simples. É baseada na suposição que termos que não são frequentes não são confiáveis para a predição da categoria. Somente os termos que ocorrem em maior quantidade de documentos são selecionados. Por sua simplicidade, tem um desempenho similar

ao IG e ao Chi se a quantidade de palavras-chaves não for baixo (Tasci & Güngor, 2009; Sebastiani, 2002).

$$DF(t_k, c_i) = P(t_k, c_i) \quad (3)$$

onde $P(t_k, c_i)$ é a quantidade de documentos pertencendo à base de dados.

Para a DF os termos ocorrendo mais frequentemente na coleção são os mais valiosos para a Categorização de Textos. Isto parece contradizer uma suposição da Recuperação de Informação (IR), no qual os termos com frequência baixa-a-média no documento são os mais informativos. No entanto, isto não é contraditório, visto ser conhecido que a maioria das palavras ocorrendo no *corpus* tem uma Freqüência do Documento pequena. Isto quer dizer que reduzindo o conjunto de termos por 10 (dez) e usando a métrica Frequência do Documento, somente tais palavras são removidas, enquanto palavras de baixo-a-médio a alta-freqüência no documentos são preservadas (Sebastiani, 2002).

2.3.4.4 Entropia

Entropia é o cálculo do Ganho de Informação baseado em uma medida utilizada na teoria da informação. A Entropia caracteriza a (im)pureza dos dados; em um conjunto de dados, é uma medida da falta de homogeneidade dos dados de entrada em relação a sua classificação. Por exemplo, a Entropia é máxima (igual a 1) quando o conjunto de dados é heterogêneo (Silva, 2005).

A pesagem por Entropia é representada por:

$$ENTROPIA_{kj} = \log(TF + 1) \times \left\{ 1 + \frac{1}{\log N} \times \sum_{i=1}^{N} \left[\frac{f_{ki}}{n_k} \times \log\left(\frac{f_{ki}}{n_k} \right) \right] \right\} \quad (4)$$

onde

TF é a ocorrência do k-ésimo termo em d_j;

N é a quantidade de documentos de treinamento;

n_k é a quantidade de documentos pelo qual o k-ésimo termo ocorre pelo menos uma vez;

f_{ki} é a quantidade de vezes que o k-ésimo termo ocorre no documento i.

2.3.4.5
Indexação Semântica Latente (LSI - em inglês *Latent Semantic Indexing*)

O método de Indexação Semântica Latente (LSI - em inglês *Latent Semantic Indexing*) é o processamento dos documentos e a extração nos documentos de uma representação reduzida que facilite a busca.

É usado a Decomposição de Valor Singular (DVS) para reduzir as dimensões do espaço termo-documento, tentando resolver problemas de sinonímia e polissemia (uma palavra que representa mais de um significado). LSI representa explicitamente termos e documentos em um espaço rico e de dimensionalidade alta, permitindo que relacionamentos semânticos subentendidos (latentes) entre termos e documentos sejam explorados durante a procura (Lopes, 2004).

Através dessa técnica é obtido um agrupamento das palavras de acordo com as suas representações conceituais mais apropriados. No processo LSI, as palavras são consideradas dentro do contexto em que estão inseridas, ou seja, o método captura a significação estatística da palavra em relação às palavras que a circundam. O LSI tem sua origem associada a uma área da matemática conhecida como Análise de Fatores. Do grupo de textos extraímos uma matriz X , na qual é transformada em uma decomposição de valores singulares ($T_0S_0D_0$), resultando em uma transformação da matriz de termos original em 3 (três) outras matrizes, sendo que a multiplicação dessas matrizes reconstitui a matriz original. A dimensão dessas matrizes é usualmente muito grande. O passo seguinte é uma redução no tamanho da matriz de valores singulares, com a matriz final tendo apenas valores significativos (TSD truncado). Após o processamento TSD o usuário pode estabelecer uma consulta.

2.3.4.6
Informação Mútua (MI - em inglês *Mutual Information*)

A Informação Mútua (em inglês *Mutual Information*) é um técnica usada em Modelagem Estatística da linguagem em associações de palavras e aplicações correlatas (Lopes, 2004).

A Informação Mútua de um termo t com respeito à categorização C é definida como (Sebastiani, 2002):

$$MI(t_k, c_i) = \log \frac{P(t_k, c_i)}{P(t_k).P(c_i)} \quad (5)$$

onde $P(c_i)$ é a probabilidade da categoria c_i ocorrer, $P(t_k)$ é a probabilidade da presença do termo t_k, $P(t_k, c_i)$ é a probabilidade de c_i e t_k co-ocorrerem.

Considerando-se uma tabela de contingências de um termo t e uma categoria c, A é o número de vezes em que t e c co-ocorrem, B é o número de vezes que t ocorre sem c, C é o número de vezes que c ocorre sem t e N é o número total de documentos, então o critério de Informação Mútua entre t e c é definido como (Lopes, 2004):

$$I(t,c) = \frac{\log \ A \times N}{(A+C) \times (A+B)} \quad (6)$$

onde $I(t,c)$ tem o valor zero se t e c são independentes. Para medir a importância de um termo em uma seleção de características globais, combina-se as pontuações específicas da categoria de um termo em duas formas alternativas (Lopes, 2004):

$$I_{medio}(t) = \sum_{i=1}^{m} P_r(c_i) \times I(t, c_i) \quad (7)$$

$$I_{\max}(t) = \max_{i=1}^{m} \{I(t, c_i)\} \quad (8)$$

Uma deficiência da Informação Mútua é que a pontuação é fortemente influenciada pelas probabilidades marginais dos termos (Lopes, 2004).

2.3.4.7
Escore de Relevância

Na etapa de treinamento, uma técnica de indexação e de redução de dimensionalidade usada é a de Escore de Relevância. Essa técnica atribui à importância de cada termo dentro do contexto, o que é essencial para a análise da similaridade. Diferentes assuntos podem possuir termos iguais como sendo termos relevantes, porém com graus de importância diferentes (Galho & Moraes, 2003).

A idéia dessa técnica está baseada na frequência de um termo em uma categoria e na sua frequência nas demais categorias. A partir desses dados é calculada a relevância do termo para uma dada categoria. Podem-se adotar duas hipóteses: na primeira, naquela(s) categoria(s) em que o termo alcançar um Escore de Relevância maior ele será escolhido para representá-lo (técnica de truncagem); na segunda, todos os termos serão escolhidos para representá-lo (Galho & Moraes, 2003).

O cálculo do escore de relevância r_k, apresentado por Wiener, Pedersen e Weigend (1995) é dado pela fórmula abaixo (Galho & Moraes, 2003):

$$r_k = \log_{10} \frac{\frac{W_{tk}}{d_t} + \frac{1}{6}}{\frac{W_{ik}}{d_i} + \frac{1}{6}} \tag{9}$$

onde,

W_{tk} é a quantidade de documentos pertencentes a uma dada categoria t que contem o termo k

W_{ik} é a quantidade de documentos de outras categorias que contem o termo k

d_t é a quantidade total de documentos da categoria t

d_i é a quantidade de documentos de outras categorias

No trabalho de Galho & Moraes (2003), o cálculo do Escore de Relevância foi executado com as bases 2, 10, 16 e natural, bem como foram efetuados testes também com relação a constante 1/6, que foi substituída por ½, 1/9, 1/15.

Os valores dos pesos dos termos devem estar normalizados entre zero e um para realizar o cálculo do grau de igualdade entre eles (Galho & Moraes, 2003).

2.3.5
Técnicas de Indexação na Categorização

As técnicas de categorização de textos partem do princípio que um documento pode ser representado como um vetor. Isso provoca algumas alterações no texto, pois há perda da ordem das palavras, do contexto gramatical entre outros (Gomes & Costa, 2005).

Uma das etapas de categorização de textos é transformar documentos num conjunto de *strings* de caracteres numa representação apropriada para a tarefa de classificação. Há várias técnicas desenvolvidas: pesagem de termos; frequência de termos (TF); frequência de documento inversa (IDF); frequência de termos (TF) *versus* frequência de documento inversa (IDF); frequência de documento com pesagem inversa (WIDF); entropia, etc.

2.3.5.1
Pesagem de Termos (em inglês *Term Weighting*)

Pesagem de termos (em inglês *Term Weighting*) é um dos métodos de indexação de termos mais simples em Categorização de Textos e corresponde a um valor dado a um termo e que reflete a importância do termo no documento. Há diferentes aproximações de pesagem para indexação de textos, todos compartilhando as seguintes observações (Hadi Wa' el Musa et al., 2008):

- maior quantidade de vezes que um termo ocorre em documentos que pertençam a uma dada categoria, maior é sua importância para aquela categoria;

- maior quantidade de vezes que um termo aparece em diferentes documentos pertencendo a diferentes categorias, menor a sua importância para discriminação da categoria a que pertencem esses documentos.

2.3.5.1.1
Pesagem Booleana

A função de pesagem com características mais simples é designar o mesmo valor a cada termo que ocorre no documento de treinamento, i.e., seja 1 (hum) para ocorrência do termo e 0 (zero) para aqueles termos com não-ocorrência no

texto, o qual é chamado de aproximação de características sem pesagem ou pesagem booleana (Soucy & Mineau, 2001a).

$w(d,t) = 1$ caso o termo ocorra no documento (10)

$w(d,t) = 0$ caso contrário (11)

2.3.5.1.2 Freqüência de Termos (*TF*)

Um dos métodos também simples de pesagem de termo que é usado para medir a importância de cada termo num documento é a Freqüência do Termo – *TF* (em inglês *Term Frequency*). Nesse método, cada termo tem um valor ou peso proporcional à quantidade de vezes que o termo ocorre no texto. Geralmente, para um documento d e um termo t, o peso de t em d é dado como (Hadi, Wa'el Musa et al., 2007, 2008):

$$w(d,t) = TF(d,t) \quad (12)$$

TF pode ajudar a melhorar a medida de desempenho chamada Abrangência (em inglês *Recall*) usada para Categorização de Textos, pois termos que tendem a aparecer com frequência em muitos documentos, tem tais termos pouco poder discriminatório.

Abrangência é a fração de documentos relevantes obtidos corretamente. Pode-se dizer que *TF* segue a curva de distribuição normal, com relação à importância dos termos para o processo de obtenção do resultado correto, que mostra que um termo que aparece com muita frequência ou pouca frequência não melhora a obtenção dos resultados corretos (Hadi Wa'el Musa et al., 2007, 2008).

Quando a frequência dos termos se encontra em intervalos baixos ou altos, então os mesmos devem ser removidos. Devem ser removidos também os *stopwords*, que na maioria das vezes tem alta frequência (Hadi, Wa'el Musa et al., 2007, 2008).

2.3.5.1.3 Frequência de Documentos Inversa (IDF)

Frequência do Termo *TF* reflete a importância do termo num único documento, entretanto, estamos interessados na frequência do termo em um conjunto de documentos, que é chamada de Frequência de Documentos Inversa – *IDF* (em inglês - *Inverse Document Frequency*), que significa a importância de cada termo inversamente proporcional à quantidade de documentos que contém aquele termo. Tabela 1 mostra que para um dado *corpus* de documento, quando a frequência de um termo em um documento aumenta, a importância desse termo diminui de acordo a *IDF*. Quando o termo ocorre em uma pequena quantidade de documentos, esse termo é discriminatório (quantidade maior que 10). Quando o termo ocorre frequentemente em uma grande quantidade de documentos, então ele não é discriminatório de acordo com *IDF* (Hadi Wa'el Musa et al., 2007, 2008).

Para um dado conjunto de *N* documentos, se *n* documentos contem o termo *t*, *IDF* é dado por (Hadi Wa'el Musa et al., 2007, 2008):

$$IDF(t) = \log(\frac{N}{n}) \tag{13}$$

A tabela 1 ilustra a correlação de *IDF* entre a quantidade total de documentos e a quantidade de documentos contendo um termo específico (Hadi Wa'el Musa et al, 2007, 2008).

Tabela 1 – Correlação de *IDF* entre a Quantidade Total de Documentos e a Quantidade de Documentos Contendo um Termo Específico.

Quantidade Total de Documentos	Quantidade de Documentos contendo o Termo	IDF = log(N/n)	Importância do Termo
1000	10	2.000	Máximo
	20	1.699	↓
	40	1.399	↓
	80	1.097	↓
	160	0.795	↓
	320	0.494	
	640	0.190	Mínimo

2.3.5.1.4 Frequência de Termos (TF) X Frequência de Documentos Inversa (IDF)

Segundo Hadi Wa'el Musa et al (2007, 2008), a combinação de TF e IDF, para a pesagem dos termos, segue a definição de Salton (1988), sendo que essa combinação melhora o desempenho com referência à precisão. O produto TF e IDF que consiste em aumentar o peso dos termos que aparecem em poucos documentos e reduzir o peso de termos que aparecem em vários documentos, é dado pela seguinte equação (Hadi et al., 2007, 2008; Krishnakumar, 2006; Gomes, 2005):

$$w(d,t) = TF(t) \times IDF(t) \tag{14}$$

Uma variação da pesagem $TF.IDF$, que leva em consideração os tamanhos diferentes dos documentos é a seguinte:

$$w'(d,t) = \frac{TF}{(\sum TF^2)^{1/2}} \times \log(\frac{N}{n}) \tag{15}$$

onde N é a quantidade de documentos, n é a quantidade de documentos que contem o termo t e TF é a quantidade de vezes que o termo t ocorre no documento d.

2.3.5.1.5 Frequência de Documentos com Pesagem Inversa (WIDF)

Na pesagem IDF todos os documentos contendo certo termo são tratados igualmente devido a uma contagem binária. Em outras palavras se um termo ocorre em 4 (quatro) documentos com diferentes frequências em cada um dos documentos, IDF não considera a quantidade de vezes em que o termo ocorre nesses documentos e somente considera o fato que o termo apareceu. $WIDF$ de um termo t no documento d, corresponde a frequência normalizada do termo, dado por (Hadi, Wa'el Musa et al., 2007, 2008):

$$WIDF(d,t) = \frac{TF(d,t)}{\sum_{i \in D} TF(i,t)} \tag{16}$$

onde $TF(d,t)$ é a ocorrência de t em d e i é o domínio sobre os documentos da coleção D.

O peso do termo com referência a *WIDF* é dado por (Hadi Wa'el Musa et al., 2007, 2008):

$$w(d,t) = WIDF(d,t) \tag{17}$$

2.3.5.1.6
Frequência de Termos Modificada (TF') X Frequência de Documentos Inversa (IDF)

O método de pesagem do termo que é usado para medir a importância de cada termo num documento é modificado nesse método para *TF´* sendo que *TF* é assumido ter um valor proporcional à quantidade de vezes que o termo ocorre no texto. Para um documento d e um termo t, o peso de t em d é dado como (Moraes & Lima, 2007):

$$TF'(d,t) = 1 + \log TF(d,t) \text{ se } TF(d,t) > 0 \tag{18}$$

$$TF'(d,t) = 0 \text{ se } TF(d,t) <= 0 \tag{19}$$

Para $w_{normalizado}$ temos (Moraes & Lima, 2007):

$$w_{normalizado}(d,t) = \frac{w(d,t)}{\left[\sum w^2(d,t)\right]^{1/2}} \tag{20}$$

ou seja,

$$w(d,t) = \frac{(1+\log(TF(d,t))) \times \log(\frac{N}{n})}{\left(\sum[(1+\log(TF(d,t))) \times \log(\frac{N}{n})]^2\right)^{1/2}} \tag{21}$$

onde,

TF é definido como a quantidade de vezes que o termo t ocorre no documento d;

N é o conjunto de documentos;

n é a quantidade de documentos contem que contem o termo t.

2.3.6

Medidas de Similaridade e de Distância

2.3.6.1
Medidas de Similaridade

Há várias técnicas conhecidas de similaridade, tais como: VSM combinado com Cosseno, Jaccard ou DICE; Modelo Probabilístico (PM), etc. (Hadi Wa'el Musa et al., 2007, 2008).

A seguir estão definidas algumas dessas técnicas onde: w_{ik} corresponde ao peso do *k*-ésimo elemento do vetor de termo V_i, i.e., documento pré-categorizado, w_{jk} é o peso do *k*-ésimo elemento do vetor de termo V_j, i.e., texto a ser categorizado. Maior o valor de $Simil(V_i, V_j)$, maior a similaridade entre esses dois documentos (Hadi Wa'el Musa et al., 2007, 2008).

2.3.6.1.1
Medida de Similaridade do Cosseno

O Cosseno identifica o ângulo de proximidade entre documentos ou entre quaisquer objetos que contenham estruturas semelhantes, utilizando os termos existentes nestes, a partir da matriz de termos gerada pela Norma Euclidiana (Gomes & Costa, 2005).

A fórmula é a seguinte (Hadi, Wa'el Musa et al., 2007, 2008; Moraes & Lima, 2007; Gomes & Costa, 2005):

$$Simil(V_i, V_j) = \frac{\sum_{k=1}^{m} \left(w_{ik} \times w_{jk} \right)}{\left(\sum_{k=1}^{m} w_{ik}^2 \times \sum_{k=1}^{m} w_{jk}^2 \right)^{1/2}} \qquad (22)$$

2.3.6.1.2
Medida de Similaridade de Jaccard

Para a medida de similaridade de Jaccard, a fórmula é a seguinte (Hadi, Wa'el Musa et al., 2007, 2008):

$$Simil(V_i, V_j) = \frac{\sum_{k=1}^{m} \left(w_{ik} \times w_{jk} \right)}{\sum_{k=1}^{m} w_{ik}^2 + \sum_{k=1}^{m} w_{jk}^2 - \left(\sum_{k=1}^{m} w_{jk} \times \sum_{k=1}^{m} w_{jk} \right)} \quad (23)$$

2.3.6.1.3 Medida de Similaridade de DICE

Para a medida de similaridade de DICE, a fórmula é a seguinte (Hadi, Wa'el Musa et al., 2007, 2008):

$$Simil(V_i, V_j) = \frac{2 \times \sum_{k=1}^{m} \left(w_{ik} \times w_{jk} \right)}{\sum_{k=1}^{m} w_{ik}^2 + \sum_{k=1}^{m} w_{jk}^2} \quad (24)$$

2.3.6.1.4 Medida de Similaridade do *cosSim*

Para comparar documentos d e i usando a pesagem de termos binário, usamos a fórmula abaixo, onde C é a quantidade de termos que i e d compartilham, A é a quantidade de termos em i e B é a quantidade de termos em d.

$$\cos Sim(i, d) = \frac{C}{(A \times B)^{1/2}} \quad (25)$$

2.3.6.1.5 Medida do Índice de Similaridade

Podemos aplicar mais de uma técnica de contagem e indicar um fator que será o indicativo entre os documentos. Esta técnica é conhecida como Índice de Similaridade (Carvalho et al., 2005):

$$Indice.de.Similaridade = \frac{\alpha \times \cos seno + \beta \times dice + \delta \times jaccard}{\alpha + \beta + \delta} \quad (26)$$

onde α, β, δ são constantes.

2.3.6.2
Distâncias Métricas

Também se pode verificar a similaridade entre dois documentos medindo-se a distância entre eles. A medida métrica de distância mede a dissimilaridade entre dois pontos de dados em termos de algum valor numérico. Também mede a similaridade, portanto podemos dizer que maior à distância, menor a similaridade e menor a distância maior a similaridade (Khan, 2001).

Para definir uma distância métrica precisamos definir um conjunto de pontos e uma regra $d(X,Y)$, para medição da distância entre dois pontos X e Y do espaço (Khan, 2001).

Há a distância de *Minkowski*; distância de *Manhattan*; distância Euclidiana; distância MAX; distância de *Camberra*; distância de Corda Quadrada; distância *Chi-Squared*; distância HOB (bit de maior ordem). A seguir é descrita cada uma dessas distâncias.

2.3.6.2.1
Distância de Minkowski

Considerando pontos X e Y, num espaço de dimensão n, como vetores $< x_1, x_2, x_3, x_4, ...x_n >$ e $< y_1, y_2, y_3, y_4, ...y_n >$ a distância de *Minkowski* é definida como (Wilson & Martinez, 2000; Khan, 2001):

$$d_p(X,Y) = \left\{ \sum_{i=1}^{n} w_i \mid x_i - y_i \mid^p \right\}^{1/p} \qquad (27)$$

onde p é um inteiro positivo, x_i e y_i são os i-ésimos componentes de X e Y, respectivamente, w_i (≥ 0) é o peso associado com a dimensão-i ou característica-i. A associação de pesos permite que algumas características dominem outras na medida de similaridade. Caso seja usado $w_i = 1$ ela é chamada de distância L_p (Khan, 2001).

$$d_p(X,Y) = \left\{ \sum_{i=1}^{n} | x_i - y_i |^p \right\}^{1/p} \quad (28)$$

2.3.6.2.2 Distância de Manhattan

Fazendo-se $p = 1$ na distância de L_p, temos a distância de Manhattan. Essa distância também é conhecida como distância de *City-Block* (Wilson & Martinez, 2000; Khan, 2001; Lopes, 2004).

$$d_1(X,Y) = \sum_{i=1}^{n} | x_i - y_i | \quad (29)$$

2.3.6.2.3 Distância Euclidiana

Para $p = 2$, a distância de Minkowski ou à distância L_p é conhecida como a distância Euclidiana (Wilson & Martinez, 2000; Khan, 2001). A distância Euclidiana é definida como (Lopes, 2004; Khan, 2001):

$$d_2(X,Y) = \left\{ \sum_{i=1}^{n} | x_i - y_i |^2 \right\}^{1/2} \quad (30)$$

2.3.6.2.4 Distância MAX

Para $p = \infty$, a distância de Minkowski ou à distância L_p é conhecida como a distância MAX ou a distância do xadrez. A distância é representada por (Wilson & Marinez, 2000; Khan, 2001):

$$d_\infty(X,Y) = \max_{i=1}^{n} |x_i - y_i| \quad (31)$$

2.3.6.2.5 Distância de Cambera

Considerando os pontos X e Y, num espaço de dimensão n, como vetores $< x_1, x_2, x_3, x_4 ... x_n >$ e $< y_1, y_2, y_3, y_4 ... y_n >$ a distância de Camberra é definida como (Wilson & Martinez, 2000; Khan, 2001):

$$d_c(X,Y) = \sum_{i=1}^{n} \frac{|x_i - y_i|}{x_i + y_i} \tag{32}$$

2.3.6.2.6
Distância da Corda Quadrada (em inglês *Cord Square*)

Considerando os pontos X e Y, num espaço de dimensão n, como vetores $< x_1, x_2, x_3, x_4 ... x_n >$ e $< y_1, y_2, y_3, y_4 ... y_n >$ a distância da Corda Quadrada é definida como (Wilson & Martinez, 2000; Khan 2001):

$$d_{cq}(X,Y) = \sum_{i=1}^{n} \left\{ (x_i)^{1/2} - (y_i)^{1/2} \right\}^2 \tag{33}$$

2.3.6.2.7
Distância da *Chi-Squared*

Considerando pontos X e Y, num espaço de dimensão n, como vetores $< x_1, x_2, x_3, x_4 ..... x_n >$ e $< y_1, y_2, y_3, y_4 .... y_n >$ a distância da *chi-squared* é definida como (Wilson & Martinez, 2000; Khan, 2001):

$$dcs(X,Y) = \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{x_i + y_i} \tag{34}$$

2.3.6.2.8
Distância HOB (Bit de Maior Ordem)

A distância HOB (em inglês *High Order Bit*) é uma distância métrica usada na computação, geralmente quando se usa a técnica da Árvore–P (Rahal & Perrizo, 2004; Khan, 2001).

Árvore-P são estruturas de dados na forma de árvores que armazenam dados numéricos em colunas, no formato de *bit* comprimido, dividindo cada atributo em

bits, i.e., representando cada valor de atributo por seu equivalente binário, agrupando junto todos os *bits* em que cada posição de *bit* para todas as tuplas e representando cada grupamento de *bits* por uma Árvore-P (Rahal & Perrizo, 2004).

A distância HOB é definida para dados onde cada componente de um ponto de dado é um inteiro. Consideramos somente as posições de *bits* consecutivas mais significantes iniciando do *bit* mais a esquerda, o qual é o *bit* de maior ordem. Considere os valores de 8-*bits*, x_i e y_i, representado em binário. O primeiro *bit* é o *bit* mais significativo e o oitavo *bit* é o menos significativo. A tabela 2 a seguir ilustra três números binários e suas proximidades (Rahal & Perrizo, 2004; Khan, 2001).

Tabela 2 – Números Binários e suas Proximidades

Posição do *bit*: 1 2 3 4 5 6 7 8	Posição do *bit:* 1 2 3 4 5 6 7 8
X_1 0 1 1 0 1 0 0 1	x_1 0 1 1 0 1 0 0 1
Y_1 0 1 1 1 1 1 0 1	y_2 0 1 1 0 0 1 0 0

Esses valores são similares nas três posições mais significativas de *bits*: primeiro; segundo; e terceiro (011). Depois eles diferem no quarto *bit*, não consideramos mais nenhum *bit* de ordem menor, embora x_1 e y_1 tenham idênticos *bits* nas quinta, sétima e oitava posição. Desde que estamos procurando proximidade em valores, depois de diferir em *bits* de ordens maiores, similaridade em alguma ordem menor não é significativo com respeito a nosso propósito. Similaridade x_1 e y_2 são idênticas nos quatro *bits* mais significativos (0110). Então x_1 é mais similar a y_2 do que y_1 (Rahal & Perrizo, 2004; Khan, 2001).

2.3.7 Processamento

2.3.7.1 Treinamento

Geralmente diferenciamos os documentos de treinamento como $d \varepsilon D_{treino}$ e os documentos de teste como $d \varepsilon D_{teste}$, onde $D_{treino} \bigcup D_{teste} = D$ e $D_{treino} \bigcap D_{teste} = 0$ (Tikk & Biró, 2001, 2003).

Para implementação ou construção da categorização automática, a maioria dos sistemas necessita de treinamento. Durante essa fase (treinamento), a coleção de documentos classificados manualmente é apresentada ao sistema, seguindo várias etapas. Nessa fase o sistema aprende a reconhecer categorias de acordo com as especificidades dos algoritmos escolhidos (Fall & Benzineb, 2002).

Para tarefas de multi-classificação, i.e., quando uma quantidade de categorias diferentes é associada com cada documento, alguns algoritmos geralmente necessitam de uma etapa de validação depois de uma fase de treinamento, em que um conjunto de limiares (em inglês *threshold*) é estipulado para permitir que o sistema distinga entre documentos relevantes a uma categoria daqueles que não o são. A coleção de documentos usados para validação deve ser diferente do usado para o treinamento (Fall & Benzineb, 2002).

Os documentos de treinamento são usados para construir o classificador e os documentos de teste são usados para testar o desempenho do classificador e geralmente não participam para a construção do classificador (Tikk & Biró, 2001).

2.3.7.2 Teste

É importante testar o algoritmo de categorização automático com um conjunto de documentos que não foi usado durante a fase de treinamento. Os resultados dessas categorizações automáticas devem ser comparados com os resultados dos documentos pré-classificados (Fall & Benzineb, 2002).

2.3.7.3 Algoritmos de Categorização de Textos

Após o Pré-Processamento e a Transformação, surge a etapa de Extração de Conhecimento, com o descobrimento de padrões úteis e desconhecidos presentes

nos documentos, através a execução das tarefas de Mineração. Uma tarefa típica é a categorização, que consiste em examinar as características de um texto e atribuir a ele uma classe pré-definida. Nesse trabalho nos ateremos somente à tarefa de categorização de textos (Krishnakumar, 2006).

Sendo a Categorização de Textos (TC, em inglês *Text Categorization*), o processo de agrupar documentos de texto em uma ou mais categorias pré-definidas, uma grande quantidade de técnicas estatísticas de categorização e técnicas de aprendizagem de máquina já foram aplicadas na categorização de texto (Lopes, 2004).

As categorias são escolhidas para corresponder aos tópicos ou temas dos documentos. O principal objetivo da categorização é a organização automática. Alguns sistemas de categorização retornam uma única categoria para cada documento, enquanto outros retornam categorias múltiplas. Em ambos os casos, um categorizador pode retornar nenhuma categoria ou algumas categorias com confiabilidade muito baixa. Nesses casos, o documento é normalmente associado a uma categoria rotulada como desconhecida para posterior classificação manual (Lopes, 2004).

O início da categorização de textos foi guiada pela engenharia do conhecimento. Dado um conjunto de categorias pré-definidas, um conjunto de regras é definido manualmente para cada categoria por especialistas. Essas regras especificam condições que um documento deve satisfazer para pertencer à categoria correspondente. Nos anos 90, o aprendizado de máquina começou a ficar popular e assumir o processo de categorização. A categorização por aprendizado de máquina provou ser tão acurada como a categorização dirigida por especialistas (Lopes, 2004).

Existem duas maneiras de criar as categorias. A primeira é a criação de um *thesaurus* para definir o conjunto de termos específicos para cada domínio e a relação entre eles. As categorias podem se criadas baseando-se na freqüência das palavras específicas de cada domínio que estão no texto do documento. A segunda seria treinar uma ferramenta de categorização com um conjunto de documentos amostrais. Um conjunto de exemplos representando cada categoria é apresentado à ferramenta que então analisa estatisticamente modelos lingüísticos, tais como

afinidades léxicas e frequências de palavras, para produzir uma assinatura estatística para cada categoria. O categorizador aplica as assinaturas estatísticas a documentos para encontrar os candidatos mais parecidos (Lopes, 2004).

Muitas técnicas de categorização estatística e técnicas de aprendizagem de máquina foram propostos para categorização automática. Os vários algoritmos de categorização de documentos dividem-se em duas categorias. A primeira contém algoritmos de Aprendizado de Máquina tais como: *Support Vector Machines* - SVMs (Vapnik, 1995); classificadores usando a técnica de k-Vizinhos-Mais-Próximos - *k*-NN (em inglês *k-Nearest Neighbor*) (Aha, 1992); classificadores Rocchio; classificador probabilístico *Naïve Bayes* (NB) ou classificadores Bayesianos (Duda & Hart, 1973); classificador Árvore de Decisão (em inglês *Decision Tree classifiers*) (Quinlan, 1986); Redes Neurais; classificadores de Modelo de Regressão; Conjuntos de Regras; Classificadores Baseados em Exemplos dentre outros (Guo et al., 2002; Gomes & Costa, 2005; Lopes, 2004). A segunda contém algoritmos de categorização desenvolvidos a partir da Área de Recuperação de Informação, tais como: algoritmo de *Feedback* de Relevância; classificadores Lineares; classificadores de Exemplos Genéricos (Lopes, 2004).

Nesses categorizadores são utilizadas ferramentas para dar suporte à decisão, baseadas no treinamento (Gomes & Costa, 2005). O objetivo é efetuar uma carga de dados como treinamento da ferramenta com resultados esperados para que este possa reproduzir os resultados com um taxa mínima de erro (Gomes et al, 2005). As simulações dessas técnicas em base de dados textuais, tais como a *Reuters*, apresentaram na maioria das vezes, bons resultados.

A categorização poder ser: de Rótulo Simples *versus* Multi-Rótulo; por Pivotamento de Categoria *versus* Pivotamento por Documento; por Ordenação de Categoria *versus* Ordenação por Documento; Rígida.

A categorização de Rótulo Simples associa cada documento a uma e apenas uma categoria. Nesse caso têm-se categorias não superpostas. A categorização Multi-Rótulo retorna um conjunto de k categorias às quais um documento pode pertencer, onde k é pré-definido (Lopes, 2004).

A categorização por Pivotamento de Categoria encontra todos os documentos que podem ser arquivados sob uma categoria específica. A

categorização por Pivotamento de Documento encontra todas as categorias sob a qual certo documento pode ser arquivado e é usado em aplicações aonde os documentos vem como uma sequência, i.e., não disponível ao mesmo tempo. É usado quando novas categorias podem ser adicionadas dinamicamente, pois um conjunto de documentos não pode ser classificado sob qualquer uma das categorias já definidas (Lopes, 2004).

Na categorização por Ordenação, dado um documentos d para ser categorizado, a categorização retorna uma lista ordenada contendo todas as categorias, de tal forma que categorias com maiores probabilidades de conter o documento d são colocadas no topo da lista. Uma variação poderia ser quando é dada uma categoria c e o sistema retorna uma lista ordenada contendo todos os documentos de tal forma que documentos no topo da lista tem maiores probabilidades de pertencer a c. O primeiro caso é referido com *ranking* de categoria e o último com *ranking* de documento (Lopes 2004).

A categorização rígida é o processo de retornar uma única categoria, dado um documento (Lopes, 2004).

Aplicações em Categorizadores de Texto datam dos anos 60. A Categorização de Texto se expandiu de outras áreas como Recuperação de Informação e Filtragem de Informação até que se tornou uma área de pesquisa própria nos anos 80. As áreas de aplicação em Categorização de Textos são: Indexação Automática; Organização de Documentos; Filtragem de Textos; *Word Sense Disambiguation.*

O algoritmo Conjunto de Exemplos Generalizados (em inglês *Generalized Instance Set* - GIS) proposto por Lam et al. (1998) usa mais de um vetor protótipo para cada categoria, visando superar os problemas do algoritmo tradicional k-NN e do classificador linear. Foram obtidos melhores resultados nos experimentos, contudo o desempenho do GIS depende: da ordem em que os exemplos positivos são escolhidos; do valor de k; e como os k exemplos de topo do conjunto de treinamento depois de obtido o vetor protótipo, irão afetar o cálculo de vetores futuros (Guo et al., 2002).

Guo et al. (2002), propôs o algoritmo chamado Modelo-Baseado no *k*NN (*k*NN Model, em inglês *k*NN *model-based algorithm*), no qual ele usa em sua

simulação 3 (três) algoritmos de categorização: *k*-NN; Rocchio; e *k*NN Model. A pesagem usada para os elementos foi a TF.IDF.

2.3.7.3.1
Classificador *Bayesiano* - Naïve Bayes

O método de *Bayes* é baseado em cálculos probabilísticos. A probabilidade de o elemento pertencer a uma categoria é avaliada pela comparação entre os vetores representativos. O centróide ou vetor protótipo da categoria define os termos que provavelmente aparecem num texto dessa categoria. O peso associado é a probabilidade de o termo aparecer em documentos da categoria. Quanto mais termos da categoria o texto contiver, maior a probabilidade dele pertencer àquela categoria. O método assume que não há dependência entre os termos, i.e., a probabilidade de um termo não é condicionada por outro.

A técnica de categorização probabilística de Naïve Bayes calcula a probabilidade de um documento pertencer a uma dada categoria baseada na suposição de que a distribuição de palavras são variáveis independentes, i.e., de que a presença de uma palavra não tem efeito na distribuição ou presença de outras palavras no mesmo documento, ou seja, que não há dependência entre termos ou que a probabilidade de um termo não é condicionada por outra. A partir de probabilidades estimadas de que palavras pertencem a diferentes categorias, são determinadas as probabilidades dos documentos pertencerem a várias categorias. Os documentos são representados por vetores que levam em consideração a frequência em cada documento (Fall & Benzineb, 2002).

O centróide ou vetor protótipo da categoria define os termos que provavelmente aparecem num texto dessa categoria. O peso associado é a probabilidade de o termo aparecer em documentos da categoria. A probabilidade de o elemento pertencer a uma categoria é avaliada pela comparação entre os vetores representativos. Quanto mais termos da categoria o texto contiver, maior a probabilidade de ele pertencer àquela categoria.

A técnica de Naïve Bayes é simples de implementar, contudo outros algoritmos de categorização têm alcançado melhor desempenho (Fall & Benzineb, 2002).

2.3.7.3.2
Classificador Rocchio

O método Rocchio, que é bem simples, utiliza um vetor protótipo (um centróide) para representar cada classe ou categoria, através da média dos pesos dos termos dos vetores de treinamento de cada categoria. A avaliação de pertinência de um elemento na classe é feita usando uma função de similaridade ou de distância entre os dois vetores representativos, os dos protótipos das categorias e do elemento a ser categorizado. Dependendo do grau de similaridade, o elemento sendo testado será ou não pertencente à categoria. Tal algoritmo pode ser encarado como baseado na similaridade. O algoritmo baseado no método de Rocchio pode tratar com ruídos (em inglês *noise*) através das contribuições dos exemplos pertinentes a cada categoria. Por exemplo, se uma característica aparece em muitos exemplos de treinamento de uma específica categoria, seu peso correspondente no vetor protótipo da dita categoria terá um valor maior. Se uma característica aparece nos exemplos de treinamento de muitas categorias, seu peso no vetor protótipo tende a zero. No entanto, o categorizador Rocchio está restrito ao espaço do conjunto de regiões do hiperplano separável linear que é menos expressivo do que o do algoritmo *k*-NN (Guo et al., 2002).

Esse algoritmo é extremamente simples de ser implementado, contudo a dificuldade surge quando um documento é designado a várias categorias. Nesse caso é comparado o documento a ser categorizado com uma média dos documentos protótipos que podem ser totalmente diferentes da categoria certa, causando erros (Fall & Benzineb, 2002).

Um refinamento usado para melhorar o desempenho do algoritmo Rocchio consiste em incluir exemplos de treinamento negativos, tomados fora de cada categoria e os usando como indicadores negativos quando da determinação dos vetores de documentos protótipos (Fall & Benzineb, 2002).

O algoritmo de Rocchio é um classificador linear. Dado um conjunto de treinamento T é calculada diretamente uma categoria $(c_i) = < v_{1i}, ..., v_{Ti} >$ por meio da fórmula:

$$v_{ki} = \beta \times \sum_{|d \in POSI|} \left[\frac{w_{kj}}{|\ POS_I\ |} \right] - \gamma \times \sum_{|d \varepsilon NEGI|} \left[\frac{w_{kj}}{|\ NEG_I\ |} \right] \qquad (35)$$

onde,

w_{kj} é o peso do termo t_j no documento d_j.

$POS_I = \{d_j \varepsilon T_r \mid \Phi(d_j, c_i) = T\}$

$NEG_I = \{d_j \varepsilon T_r \mid \Phi(d_j, c_i) = F\}$

$\Phi(d_j, c_i = T)$ ou $\Phi(d_j, c_i = F)$

Isso significa que o documento d_j pertence a (ou não pertence) a categoria c_i. As constantes β e γ representam parâmetros de controle usados para realçar a importância relativa dos exemplos positivos e negativos. O perfil de c_i é o centróide dos exemplos de treinamento positivos. Um classificador construído de acordo com o algoritmo de Rocchio recompensa a proximidade dos documentos de teste em relação ao centróide dos exemplos de treinamento positivos e seu afastamento do centróide dos exemplos de treinamento negativos. Uma desvantagem do classificador Rocchio é sua característica de dividir o espaço de dados linearmente.

2.3.7.3.3 Classificador dos *k*-Vizinhos-Mais-Próximos *k*-NN (em inglês *k-Nearest Neighbor*)

A técnica dos *k*-Vizinhos-Mais-Próximos (*k*-NN– *k-Nearest-Neighbor*) foi inicialmente analisada em (Fix & Hodges, 1951), sendo sua aplicação em problemas de classificação, pioneiramente realizada em (Johns, 1962). Foi somente a partir dos resultados apresentados em (Aha, 1992), que essa abordagem ganhou popularidade como método de classificação nas área de Aprendizado de Máquina e Mineração de Dados (Borsato, 2007).

O método dos k-Vizinhos-Mais-Próximos decide a categoria de um documento de teste pelas categorias associadas aos seus *k*-vizinhos mais próximos. Para determinar a categoria de um elemento que não pertença ao conjunto de treinamento, o categorizador *k*-NN procura os k-elementos do conjunto de treinamento que estejam mais próximos deste elemento desconhecido ou de teste, ou seja, que tenham a menor distância. Estes k-elementos são

chamados de *k* -vizinhos-mais-próximos. Uma das técnicas para se categorizar o elemento desconhecido ou de teste é verificando quais são as categorias desses *k* -vizinhos e a categoria mais frequente será atribuída à categoria do elemento desconhecido.

k -NN é um categorizador que possui apenas um parâmetro livre (o número de *k* -vizinhos) que é controlado pelo usuário com objetivo de se obter uma melhor categorização (Silva, 2005). Em parte, o sucesso da categorização depende do valor de *k* (Guo et al., 2002).

Alguns trabalhos abordaram a escolha de um valor adequado para o parâmetro *k.* Em (Wettschereck & Dietterich, 1994) foram apresentadas 4 (quatro) variações do *k*-NN clássico. Essas variações determinam o valor do parâmetro *k* a ser usado na classificação de uma nova instância através de uma avaliação da vizinhança da mesma. Essas estratégias apresentaram um desempenho similar ao do *k*-NN clássico para 12 (doze) bases de dados comumente utilizadas em trabalhos de classificação e mostraram-se superiores ao *k*-NN para 3 (três) bases geradas pelos próprios autores. Em Wang (2003) foi proposto a técnica *nok*NN, que realiza a classificação considerando-se não apenas um, mas vários conjuntos de vizinhos mais próximos. Os resultados mostraram que a exatidão do método proposto foi ligeiramente superior à do *k*-NN clássico, para *k* variando entre 1 e 10 (Borsato, 2007).

Esse processo de categorização pode ser computacionalmente exaustivo se considerado um conjunto com muitos dados. Para determinadas aplicações, o processo é bem aceitável (Silva, 2005).

Para a categorização da base de dados *Reuters* de histórias de *newswire*, o algoritmo *k*-NN é conhecido como sendo um dos mais efetivos (Guo et al., 2002).

Uma das variações desse algoritmo é a seleção de pontos que estão dentro de uma hiper-esfera de raio R (decidido pelo usuário) e a categoria predominantemente dentro dessa hiperesfera será a categoria do ponto desconhecido ou de teste. A desvantagem desse processo é que pode existir uma hiper-esfera sem qualquer ponto (Silva, 2005).

Na técnica *k*-NN quando um novo documento é categorizado, ele é comparado ao conjunto existente de documentos pré-classificados para

determinação dos mais similares. Similaridade entre documentos é determinada para comparação da distribuição de palavras. Nesse algoritmo, *k* indica a quantidade de documentos vizinhos a serem examinados. As categorias sugeridas para o novo documento podem ser estimadas dos documentos vizinhos pesando sua contribuição de acordo com suas distâncias e de acordo com os scores ordenados das categorias sugeridas (Fall & Benzineb, 2002).

Esse algoritmo proporciona um conjunto de documentos similares ao documento a ser classificado. Tal técnica é vantajosa no caso de categorização de documentos de patente segundo IPC, pois também proporciona base para busca no estado da arte (Fall & Benzineb, 2002).

k-NN é um método de aprendizagem, que usa todos os dados de treinamento para categorização. Por isso ele é chamado de algoritmo de aprendizagem preguiçoso e não é aconselhável para aplicações em *web* dinâmica de mineração para um grande conjunto de documentos. Um modo de melhorar a eficiência é achar algumas representações do conjunto de treinamento para categorização por meio da construção de um modelo de aprendizagem indutivo do conjunto de treinamento.

2.3.7.3.4
Classificador *Support Vector Machine* (SVM)

SVM é um método criado por Vapnick em 1995 para solucionar problemas no reconhecimento de padrões (Gomes & Costa, 2005).

O objetivo desse algoritmo é encontrar a superfície de decisão no espaço de documentos possíveis que melhor separam documentos relevantes para uma categoria daqueles que não o são. O método *Support Vector Machines* (SVM) encontra a fronteira ótima que separa os elementos da coleção em dois conjuntos. A limitação é que só se trabalha com 2 (duas) categorias. O método se baseia somente nos documentos que delimitam as fronteiras da categoria. A vantagem dessa categoria é que não há parâmetros a configurar e não há a necessidade de seleção de termos, portanto vocabulários extensos podem ser manipulados facilmente (Fall & Benzineb, 2002).

2.3.7.3.5
Classificador Redes Neurais

Redes Neurais são sistemas computacionais criados com base na modelagem do cérebro humano e sua principal característica é aprender através de exemplos. Classificações de padrões e previsão são as principais áreas de aplicação das redes neurais (Sebastiani, 2002).

Uma camada da rede recebe a entrada, na forma de uma coleção de termos e pesos representativos do documento, camadas intermediárias processam os pesos e a camada de saída sugere a categoria relevante (Fall & Benzineb, 2002; Sebastiani, 2002).

Uma maneira típica de treinar uma rede neural é o *Backpropagation*, onde os pesos dos termos de um documento de treinamento são carregados em unidades de entrada, e se misclassificação ocorre, o erro é backpropagado ocasionando mudanças nos parâmetros da rede e eliminação ou minimização do erro. O tipo mais simples de categorizador de rede neural é o *Perceptron*, que é um classificador linear (Sebastiani, 2002).

É na fase do treinamento que a rede neural aprende o problema. Existem várias técnicas para se treinar uma rede neural; a escolha mais adequada depende do problema e da aplicação. Na fase do treinamento é escolhido o algoritmo de aprendizado juntamente com os parâmetros de aprendizado, que são: taxa de aprendizado (em inglês *learning rate*); taxa de momento (*momentum*); critérios de parada; e forma de treinamento (Silva, 2005).

Se for usado para a etapa de categorização o método de Redes Neurais serão necessários muitos e bons casos-exemplos para o processo de aprendizagem supervisionada.

Uma rede neural artificial é composta por várias unidades de processamento, cujo funcionamento é bastante simples. Essas unidades são geralmente conectadas por canais de comunicação que estão associados a determinado peso. A unidades fazem operações apenas sobre seus dados locais, que são entradas recebidas pelas suas conexões. O comportamento inteligente de uma Rede Neural Artificial vem das interações entre as unidades de processamento da rede. A operação de uma

unidade de processamento, proposta por Mc Cullock e Pitts em 1943, pode ser resumida da seguinte maneira (http://www.icmc.usp.br):

- sinais são apresentados à entrada;

- cada sinal é multiplicado por um número, ou peso, que indica a sua influência na saída da unidade;

- é feita a soma ponderada dos sinais que produz um nível de atividade;

- se este nível de atividade exceder certo limite (em inglês *threshold*) a unidade produz uma determinada resposta de saída.

A maioria dos modelos das redes neurais possui uma regra de treinamento, onde os pesos de suas conexões são ajustados de acordo com os padrões apresentados, i.e., eles aprendem através de exemplos (http://www.icmc.usp.br).

Arquiteturas neurais são tipicamente organizadas em camadas, com unidades que podem estar conectadas às unidades da camada posterior. Usualmente as camadas são classificadas em três grupos (http://www.icmc.usp.br):

- camada de entrada: onde os padrões são apresentados à rede;

- camadas intermediárias ou escondidas: onde é feita a maior parte do processamento, através das conexões ponderadas; podem ser consideradas como extratoras de características;

- camada de saída: onde o resultado final é concluído e apresentado.

Uma rede neural é especificada, principalmente por sua topologia, pelas características dos nós e pelas regras de treinamento (http://www.icmc.usp.br).

A propriedade mais importante das redes neurais é a habilidade de aprender de seu ambiente e com isso melhorar seu desempenho. Isso é feito através de um processo iterativo de ajustes aplicado a seus pesos, o treinamento. O aprendizado ocorre quando a rede neural atinge uma solução generalizada para uma classe de problemas (http://www.icmc.usp.br).

Denomina-se algoritmo de aprendizado a um conjunto de regras bem definidas para a solução de um problema de aprendizado. Existem muitos tipos de algoritmos de aprendizado específicos para determinados modelos de redes neurais, sendo que estes algoritmos diferem entre si principalmente pelo modo

como os pesos são modificados. Outro fator importante é a maneira pela qual uma rede neural se relaciona com o ambiente. Nesse contexto existem os seguintes paradigmas de aprendizado (http://www.icmc.usp.br):

- Aprendizado Supervisionado quando é utilizado um agente externo que indica à rede a resposta desejada para o padrão de entrada;

- Aprendizado Não-Supervisionado (auto-organização), quando não existe um agente externo indicando a resposta desejada para os padrões de entrada;

- Reforço, quando um crítico externo avalia a resposta fornecida pela rede.

O treinamento supervisionado do modelo de rede Perceptron, consiste em ajustar os pesos e os limiares (em inglês *thresholds*) de suas unidades para que a categorização desejada seja obtida. Para a adaptação dos limiares juntamente com os pesos podemos considerá-lo como sendo o peso associado a uma conexão, cuja entrada é sempre igual à -1 e adaptar o peso relativo a essa entrada. Quando um padrão é inicialmente apresentado à rede, ela produz uma saída. Após medir a distância entre a resposta atual e a desejada, são realizados os ajustes apropriados nos pesos das conexões de modo a reduzir esta distância. Este procedimento é conhecido como Regra Delta. As respostas geradas pelas unidades são calculadas através de uma função de ativação. Existem vários tipos de funções de ativação, as mais comuns são: *Hard Limiter*; *Threshold Logic*; e *Sigmoid* (http://www.icmc.usp.br).

2.3.7.3.6
Classificador de Regras de Decisão (em inglês *Decision Rules*)

Esse algoritmo categoriza um documento seguindo um conjunto de diretrizes de classificação ou regras. As regras indicam quando uma palavra ou coleção de palavras, ou a ausência de uma palavra, é um bom indicador de que o documento pertence a uma dada categoria. As regras podem ser combinadas na forma de uma complexa árvore de decisão (Fall & Benzineb, 2002).

Tais regras de decisão de categorias podem ser aprendidas automaticamente, examinando-se quais palavras são discriminatórias entre as categorias ou especialistas podem formulá-las manualmente. Essa aproximação é única, pois permite a categorização de documentos sem a necessidade de um conjunto de

treinamento de documentos pré-classificados, visto que especialistas classificam esses documentos com precisão (Fall & Benzineb, 2002).

No caso da categorização IPC, uma grande quantidade de regras de classificação e conhecimento já está disponível fazendo desse algoritmo uma opção atraente, além de não ter sido usado extensivamente em categorização de patentes (Fall & Benzineb, 2002).

2.3.7.3.7
Classificador de Árvores de Decisão

Árvores de decisão são modelos estatísticos que utilizam um treinamento supervisionado para a categorização e previsão de dados. Em sua construção é utilizado um conjunto de treinamento formado por entradas e saídas. Estas últimas são as categorias (Silva, 2005).

As árvores de decisão estão entre os mais populares algoritmos de inferência e tem sido aplicado em várias áreas como, por exemplo, diagnóstico médico e risco de crédito e deles pode-se extrair regras do tipo se-então que são facilmente compreendidas (Silva, 2005).

Numa árvore de decisão cada nó de decisão contem um teste para algum atributo, cada ramo descendente corresponde a um possível valor desse atributo, o conjunto de ramos são distintos, cada folha está associada a uma categoria, cada percurso da árvore, da raiz à folha corresponde uma regra de categorização. No espaço definido pelos atributos, cada folha corresponde a um hiper-retângulo onde a interseção destes é vazia e a união é todo o espaço (Silva, 2005).

O critério usado para realizar as partições é o da utilidade do atributo para a categorização. Aplica-se, por esse critério, um determinado Ganho de Informação para cada atributo. O atributo escolhido como atributo teste para a corrente nó é aquele que possui o maior Ganho de Informação. A partir desta aplicação, inicia-se um novo processo de partição (Silva, 2005).

Nos casos em que a árvore é usada para categorização, os critérios de partição mais conhecidos são baseados na entropia e no índice Gini (Silva, 2005).

2.3.7.3.8
Método *Sleeping Experts*

O método *Sleeping Experts* é semelhante ao Rocchio, contudo os pesos dos termos são ajustados em sessões de treino. Esse método funciona melhor com pares e trios de palavras, do que com termos únicos. O método *Winnow* é semelhante ao *Sleeping Experts* com a diferença de que os pesos somente são ajustados se produzirem algum tipo de erro. O resultado final é conseguido após várias iterações, quando os pesos permanecerem estáveis.

2.3.7.3.9
Método *Linear Least Squares Fit* (LLSF)

O método *Linear Least Squares Fit* (LLSF) cria um modelo de regressão a partir dos casos de treino para caracterizar cada categoria, utilizando computações complexas.

2.3.7.3.10
Método Baseado em Lista de Termos e Similaridade Difusa

Pode ser usado também na etapa de categorização, depois dos conceitos definidos, um processo de raciocínio *fuzzy* ou técnica do método de similaridade difusa, onde os pesos dos sinais (termos) encontrados são computados para avaliar a possibilidade de presença de uma lista de termos no texto. A técnica de similaridade difusa permite efetuar a categorização graduada de um texto em uma ou mais categorias. O processo consiste em determinar o grau de semelhança de uma lista de termos de um novo texto com a lista de termos de cada categoria. Para cada termo comum à lista de termos do texto e de uma determinada categoria, calcula-se o grau de igualdade de seus escores de relevância através das funções difusas.

2.3.7.3.11
Algoritmo Usando a Técnica das Árvores-P (em inglês *P-Trees*)

Aqui é apresentado um modelo para representação de dados textuais baseado na idéia de colocar dados em intervalos predefinidos usando a tecnologia de

Árvore-P ou Árvore PC (em inglês *Peano Count Tree* ou *Predicate Count Tree*) (Rahal & Perrizo, 2004).

A estrutura de dados básica explorada na tecnologia da Árvore-P é a contagem de atributos da Árvore (PC-tree) ou simplesmente Árvore-P. Formalmente, a Árvore-P são estruturas de dados na forma de árvores que armazenam dados numéricos em colunas, no formato de *bit* comprimido, dividindo cada atributo em *bits* (i.e., representando cada valor de atributo por seu equivalente binário), agrupando junto todos os *bits* em cada posição de *bit* para todas as tuplas e representando cada grupamento de *bits* por uma Árvore-P. As Árvores-P provêm muita informação e são estruturadas para facilitar processos de Mineração de Dados (Rahal & Perrizo, 2004).

Depois de representar cada valor de atributo numérico por sua representação de *bits*, armazenamos todos os *bits* para cada posição separadamente. Agrupamos juntos todos os valores de *bits* na posição de *bit* x para cada atributo para todas as tuplas t. Inicialmente convertemos todos os valores representativos de atributos de numérico para binário. Formam-se Árvores-P, para cada coluna de grupo de *bits* que formam um atributo. A tabela 3 apresenta os dados numéricos de uma tabela de atributos convertidos em binários de 8 (oito) *bits* cada. A tabela 4 mostra as Árvores-P formadas por cada atributo de 8 (oito) *bits* (Rahal & Perrizo, 2004).

A tabela 5 mostra um grupo de 16 (dezesseis) *bits* transformada em uma Árvore-P depois de dividida em quadrantes ou subgrupos de 4 (quatro). Cada árvore é chamada uma árvore-P. 7 (sete) é a quantidade total de bits 1's no grupo total de *bits* mostrada na parte superior. 4, 2, 1 e 0 são a quantidade de 1's no primeiro, segundo, terceiro e quarto quadrante de *bits* respectivamente. Desde que o primeiro quadrante (o nó denotado por 4 no segundo nível da árvore) é formada por 1 *bit* em sua totalidade, nenhuma sub-árvore é necessária. Similarmente, quadrantes formados por 0 *bits* (o nó denominado de 0 no segundo nível da árvore) é chamada quadrante de puros 0 e não tem sub-árvores. É assim que a compactação é feita. Quadrantes não-puros tais como nós 2 e 1 no segundo nível na árvore são recursivamente particionados em quatro quadrantes com um nó para cada quadrante. A partição recursiva do nó termina quando ele se torna puros 1 ou puros 0 e eventualmente se alcança um ponto onde um nó é composto de *bits* simples 1 ou 0 (Rahal & Perrizo, 2004).

Árvore-P inclui operações de AND, OR, NOT e contagem da raiz (quantidade de 1 na árvore) (Rahal & Perrizo, 2004).

Tabela 3 – Dados Numéricos de uma Tabela de Atributos Convertidos em Binários de 8(oito) *bits* cada.

ATRIBUTO 1	ATRIBUTO 2	ATRIBUTO 3	ATRIBUTO 1	ATRIBUTO 2	ATRIBUTO 3
5	6	2	00000101	00000110	00000010
12	15	0	00001100	00001111	00000000
14	24	1	00001110	00001000	00000001
29	64	255	00011101	01000000	11111111

Tabela 4 - Árvore-P Oriunda da Tabela 3 de 3(três) Atributos e 4(quatro) Tuplas

ATRIBUTO	Árvore-P 1	Árvore-P 2	Árvore-P 3	Árvore-P 4	Árvore-P 5	Árvore-P 6	Árvore-P 7	Árvore-P 8
ATRIBUTO1	0000	0000	0000	0001	0111	1111	0010	1001
ATRIBUTO2	0000	0001	0000	0000	0110	1100	1100	0100
ATRIBUTO3	0001	0001	0001	0001	0001	0001	1001	0011

Tabela 5 – Grupo de 16(dezesseis) *bits* Convertidos em Árvore-P.

1111	1010	0001	0000
7 (*bits* de 1)			
4 (*bits* de 1)	2(*bits* de 1)	1(*bit* de 1)	0
----	1010	0001	----

A matriz de documentos x termos deve ter uma representação *TF.IDF* e depois cada valor dos termos dessa matriz deve ser normalizada para valores de medidas de freqüência entre 0 e 1. Para inicializar a fase de dividir em intervalos, deve-se decidir a quantidade de intervalos e o domínio de cada intervalo. Depois se substituem os valores dos termos dos vetores de documentos pelos seus respectivos intervalos (Rahal & Perrizo, 2004).

Por exemplo, podem-se usar quatro intervalos lógicos: $I0 = [0,0]$; $I1 = (0,0.1]$; $I2 = (0.1,0.2]$; $I3 = (0.2,1]$ onde "(" e ")" são exclusivos e "[" e "]" são inclusivos. A quantidade ótima de intervalos e seus domínios dependem do tipo de documentos e deve haver uma ordenação no intervalo escolhido, de tal

maneira que $I0 \leq I1 \leq I2 \leq I3$. Para cada posição de *bit* em cada termo t_i será criada uma árvore-P. Como temos 2 (dois) *bits* por termo (desde que cada valor de termo é agora um dos quatro intervalos, cada representado por 2 *bits*). Então são necessárias 2 (duas) árvores-P (uma para cada posição de *bit*), $P_{i,1}$ e $P_{i,2}$ onde $P_{i,j}$ é a representação da árvore-P onde j representa a j-ésima posição de *bit* e i representa o i-ésimo termo para todos os documentos. Cada $P_{i,j}$ representa a quantidade de documentos que tem o *bit* 1 na posição j para o termo i. Se o valor binário desejado para o termo i é 10, calculamos $P_{i,10} = P_{i,1}$ AND $P_{i,0}^{'}$ onde ' indica o complemento do *bit* ou a operação NOT (que é a contagem do complemento em cada quadrante) (Rahal & Perrizo, 2004).

Cada documento é um vetor de termos representado por intervalos de valores. A similaridade entre dois documentos d_1 e d_2 pode ser medido pela quantidade de termos comuns. Um termo t é considerado a ser comum entre d_1 e d_2 se o valor do intervalo dado ao termo t em ambos os documentos é o mesmo. Maior a quantidade de termos comuns em d_1 e d_2, maior o grau de similaridade entre eles. Entretanto, nem todos os termos participam igualmente na similaridade. A ordem do intervalo participa na similaridade. Se usarmos quatro intervalos, I0, I1, I2 e I3, onde $I0 \leq I1 \leq I2 \leq I3$, então termos comuns tendo intervalos com valores maiores tal como I3 contribuem mais para a similaridade do que termos tendo intervalos com valores menores tal como I0 (Rahal & Perrizo, 2004).

Usando termos comuns para medir a similaridade entre documentos, precisamos checar quão perto termos não-comuns estão. Se documentos d_1 e d_2 têm para certos termos, intervalos com valores diferentes, então maiores e mais próximos estiverem esses intervalos, maior o grau de similaridade entre d_1 e d_2. Por exemplo, se o termo t tiver um valor 11 em d_1 e um valor 01 em d_2, então a similaridade entre d_1 e d_2 será maior do que se o termo t tivesse um valor 10 em d_1 e um valor 00 em d_2 porque 11 contribui mais para o contexto de d_1 do que 10 e o mesmo se aplica para d_2. Entretanto, a similaridade entre d_1 e d_2 será maior se comparada ao primeiro caso, se o termo t tiver um valor 11 em d_1 e um

valor 10 em d_2 porque o *gap* entre 11 e 10 é menor do que o *gap* entre 11 e 01 (Rahal & Perrizo, 2004).

Resumindo, a similaridade entre documentos, está implicitamente especificada na representação da Árvore-P e é baseada: na quantidade de termos comuns entre dois documentos; pela proximidade dos intervalos de termos não-comuns; e pelos valores dos intervalos por si só (valores maiores significam similaridades maiores) (Rahal & Perrizo, 2004).

2.3.7.3.12
Categorização em Taxonomias Hierárquicas

Quando a taxonomia é hierárquica, deve-se considerar se o algoritmo de categorização pode ou não explorar esse fato. A complexidade de classificadores hierárquicos pode ser maior do que o linear, pois uma quantidade de categorizadores separados pode ser treinada para cada nível da hierarquia (Fall & Benzineb, 2002).

Se for usado um classificador hierárquico, deve ser decidido como treinar o sistema. Pode ser vantajoso treinar subclassificadores somente com exemplos negativos, derivados da mesma categoria família, do que de todo o *corpus*. Isso pode prover menor discriminação de subcategorias e menor tempo de treinamento (Fall & Benzineb, 2002).

3.0
Modelos Propostos

Foram desenvolvidos vários algoritmos, na linguagem C++ Builder versão 2009, usando-se o *SQL Server Management Studio* como Banco de Dados, distribuídos nas seguintes etapas: Entrada de Dados; Eliminação dos *StopWords*; Tratamento de Palavras Compostas; Stemizações baseadas nos Algoritmos de *StemmerPortuguese* e Porter; Divisão de Dados em Treinamento e Teste; e Algoritmos de Categorização incluindo Medidas de Desempenho.

3.1
Base de Dados

A base de dados usada foi a de pedidos de patentes do Instituto Nacional da Propriedade Industrial (INPI), disponível no *site espacenet*, concernente a alguns depósitos de pedidos feitos por depositantes nacionais, abrangendo as subclasses: Aquecimento e Iluminação (H05B); Cabos ou Linhas Elétricas (H02G); Painéis (H02B); Magnetos e Indutâncias (H01F); Máquinas Elétricas (H02K); Conversão de Energia (H02M); Controle ou Regulação (H02P); Tubos de Descarga, Lâmpadas de Descarga (H01J); Mesas, Escrivaninhas, Móveis de Escritório, Armários, Gavetas, Detalhes Gerais de Móveis (A47B); Cadeiras (A47C). A seção H foi escolhida devido da mesma estar direcionada a Eletricidade e a seção A estar direcionada às Necessidades Humanas.

Devido à pequena quantidade de documentos em algumas classificações à nível de grupo, os documentos foram agrupados independentemente por subclasse. Por exemplo, todas as classificações H02B1/00, H02B3/00, H02B5/00, H02B7/00, H02B11/00, H02B13/00, H02B15/00, foram agrupadas na subclasse Painéis (H02B).

Posteriormente, os textos já classificados foram reunidos pelo grupo mais alto hierarquicamente, por exemplo, as classificações H05B37/02, H05B37/04, H05B37/00, etc. foram agrupadas na categoria H05B37/00. A categoria H05B37/00 corresponde à seção H, a classe 05, a subclasse B e ao grupo 37/00.

Para as categorias ou subclasses H05B, H02G, H02B, H01F, H02K, H02M, H02P, H01J foram analisados 3081 (três mil e oitenta e um) documentos. Para as

categorias A47B e A47C, foram analisados 777 (setecentos e setenta e sete) documentos. Os documentos na forma *.txt consistiram do Resumo dos pedidos de patentes, pois o Relatório Descritivo e o Quadro Reivindicatório não se encontravam disponíveis eletronicamente para o público na base de dados brasileira ou em outra base de dados.

A categoria H05B por tratar-se de Aquecimento e Iluminação foi dividida em H05B(A) (H05B1/00 a H05B11/00 - Aquecimento) e H05B(I) (H05B31/00 a H05B43/00 - Iluminação).

No Apêndice 1 estão indicadas as quantidades dos documentos usados nos experimentos, discriminadas por subclasse e grupo (IPC), para as seguintes categorias: H02P (Tabela 24); A47C (Tabela 25); H05B, H02G e H02B (Tabela 26); H01F e H02M (Tabela 27); H01J (Tabela 28); H02K (Tabela 29); A47B (Tabela 30).

3.2
Entrada de Dados

A primeira etapa utilizada para a Categorização de Textos consistiu da Entrada de Dados. A tela correspondente a Entrada de Dados (ver Figura 2), consistiu dos seguintes dados: Categoria do documento em nível de classe segundo IPC; Número do documento; Categorias do documento segundo IPC; Resumo do documento.

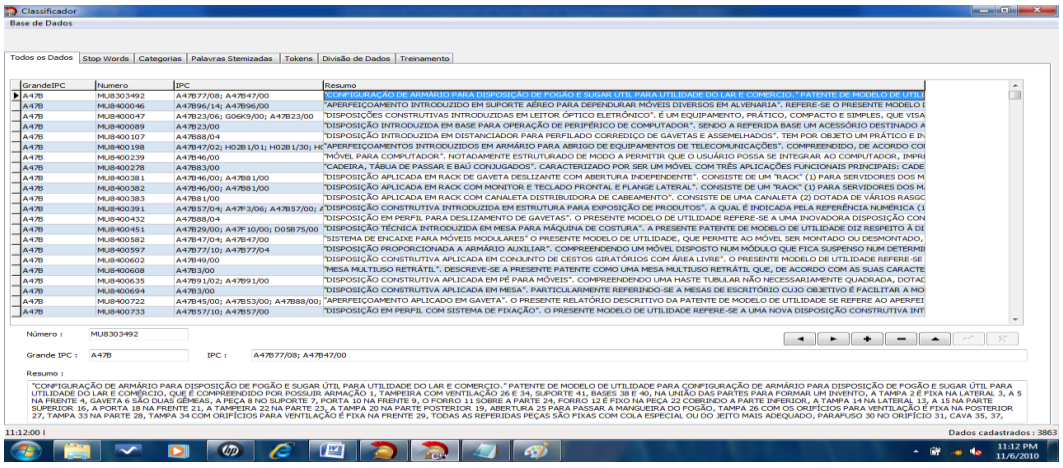


Figura 2 – Tela Correspondente a Entrada de Dados

3.3
Lista de *Stopwords* e Tratamento de Palavras Compostas

Para que o processo de descoberta de conhecimento em si seja iniciado, é necessário submeter os textos por algumas etapas de preparação. Essas etapas se encarregarão de preparar o texto, limpar, retirar terminologias conhecidas e que não são relevantes no processo de análise, dentre outras (Gomes, 2005).

Foi implementado um algoritmo visando a limpeza do texto de um dado documento, representado por seu Resumo, através da retirada dos pronomes, das palavras comuns do idioma, dos símbolos, da pontuação, dos caracteres estranhos, das palavras que não agregam informações ao texto, conhecidas como *stopwords*.

A lista de *stopwords* (*Stoplist*) foi gerada com base em listas disponíveis na *internet*, contudo levando-se em consideração também termos encontrados nos textos dos Resumos dos pedidos de patente analisados nesse trabalho, que não carregavam nenhuma informação de maior relevância, tais como: acordo; adição; alternativa; antigos; aperfeiçoados; aperfeiçoamentos; certificado; definitivamente; invenção; caracterizado, etc. Nessa etapa não foram retirados os acentos.

Para a elaboração da *Stoplist*, optou-se além da eliminação de termos com pouco valor para a categorização dos documentos de pedidos de patentes, termos que se encontravam com erros datilográficos ou sem valor semântico. A lista constou de 1337 (hum mil, trezentos e trinta e sete) termos que foram considerados sem valor.

Há uma probabilidade muito grande de que os termos contidos nos textos a serem analisados contenham erros, sinônimos. Há, portanto a necessidade da intervenção humana para minimizar os possíveis ruídos que podem advir de tais ocorrências.

Há palavras que não devem ser stemizadas por serem exceções, sendo a maioria palavras em idiomas que não o português, por exemplo: *application*; balancim; *boost*; *brake*; *brushless*; *buffer*; *buzzer*; chassi*; chip*; *chips*; *circuit*; *clamping*; *common*; *container*; *cronomatics*; *current*; *déficit*; *delay*; *design*; *diac*; *dimer*; *dimmer*; *diode*; *display*; *driver; eastern; eeproms; epoxi; esfiha; estresse*; *flag*; *flash*; *flexpower*; *flip flop; fly-back*; *forward; gap*; *graetz*; *hardware*;

highway; insitu; *integrated*; *internet*; *jump*; *jumpers*; lápis; *lead*; *light*; *link*; *loop*; *micro*; *microswitch*; *mode*; *modulation*; *mosfets*; *néon*; *neom*; *nobreak*; nylon; optotriac; oring; pelton; pet; pizza; *plug*; *plug*; *polyester*; *polyswitch*; *power*; *pull*; *pulse*; *push*; *reactance*; *reedswitch*; *ripple*; *scaner*; *scanner*; *score*; shunt; *skate*; *software*; *source*; *specific*; *start*; *starter*; *system*; *switch*; *switching*; *supply*; *timer*; *tiristor*; *transformation*; transistor; triac; *tripple*; *versus*; *voltage*; zener; *web*; well; width; etc.

Palavras compostas foram tratadas, onde foram usados os seguintes vocabulários controlados:

-termos diferentes com o mesmo significado, geralmente termos com erros ortográficos ou abreviações técnicas, por exemplo: analisador *versus* analizador; captação *versus* capitação; contacto *versus* contato; contacto *versus* contator; detectar *versus* detetar; detecção *versus* deteção; energizada *versus* energisada; flurescente *versus* fluorescente; magnetron *versus* magnetrão; néon *versus* neom; óptica *versus* ótica; resinas *versus* rezinas; secções *versus* seções; etc, foram substituídos por somente um dos termos, geralmente o correto;

- para termos compostos, nos casos de palavras que aparecem sempre juntas e que quando se reúnem apresentam um significado diferente que cada uma delas tem separadamente, optou-se por economia processual a junção de alguns termos compostos, tais como: foto diodos para fotodiodos; corrente alternada para correntealternada; moto redutor para motoredutor, etc. que quando aparecem juntos mudam o significado que cada uma delas tem separadamente. Também foram considerados os termos compostos separados por hífen, que às vezes apareceram nos textos com hífen e às vezes sem hífen, portanto optou-se por padronizar esses termos considerando-os escritos conjuntamente, com as seguintes exceções: anti; auto; extra; micro; multi; não; pós; pré; semi; sub; etc. os quais foram considerados escritos separadamente.

No Apêndice 2, através as tabelas 31A a 31G, está discriminada a lista de *Stopwords* ou *Stoplist* levada em consideração no experimento.

No Apêndice 3, através as tabelas 32A a 32C, estão discriminados alguns termos compostos que foram modificados, bem como termos mais comuns com erros ortográficos.

3.4
Stemização ou Normalização (vide item 2.3.2.3.1 e item 2.3.2.3.2)

Nessa etapa foram extraídas as palavras consideradas chaves e foram realizadas suas stemizações por meio de 2 (dois) métodos baseados nos seguintes radicalizadores: radicalizador de stemização de Porter; e radicalizador *StemmerPortuguese*.

3.4.1
Algoritmo de Stemização de Porter Modificado para a Língua Portuguesa

O algoritmo de Porter adaptado para a língua português, com modificações (mod) introduzidas é o seguinte (Lopes, 2004):

·Procure por exceções tais como: lápis; através; *bit*; *software*; dois; *scanner*; plug; etc. (mod)

·Faça Etapa 1

·Se não foi alterada palavra na Etapa 1

·· Então

···Faça Etapa 2

···Se foi alterada palavra na Etapa 2

····Então

·····Faça Etapa 3

···Fim-Se

··Senão

···Faça Etapa 3

·Fim-Se

·Se não foi alterada palavra na Etapa 1 e 2

··Então

··· Faça Etapa 4

·Fim-Se

·Faça Etapa 5

- Descrição da Etapa 1 – Remoção de sufixo padrão

(http://snowball.tartarus.org/algorithms/portuguese/stemmmer.html) (Lopes, 2004)

(Vide item 2.3.2.3.1 para definição das regiões R1, R2 e RV).

Procure pelo mais longo entre os seguintes sufixos e execute a ação indicada.

eza ezas ico ica icos icas ismo ismos ável ível ista istas oso osa osos osas amento amentos imento imentos adora ador ação adoras adores ações (apague o sufixo se em R2)

avel ivel (apague o sufixo se em R2)

ância(s) ancia(s) (apague o sufixo se em R2)

ante(s) (apague o sufixo se em R2) (mod)

logía logías (substitua o sufixo com ***log*** se em R2)

logia logias (substitua o sufixo com ***log*** se em R2)

ência ências (substitua o sufixo por ***ente*** se em R2)

encia encias (substitua o sufixo por ***ente*** se em R2)

osamente (apague o sufixo ***osamente*** se em R2);

icamente (apague o sufixo ***icamente*** se em R2);

adamente (apague o sufixo ***adamente*** se em R2);

atamente (apague o sufixo ***atamente*** se em R2);

ivamente (apague o sufixo ***ivamente*** se em R2);

amente (apague o sufixo ***amente*** se em R2); (mod)

ívelmente (apague o sufixo ***ívelmente*** se em R2);

ivelmente (apague o sufixo ***ivelmente*** se em R2);

avelmente (apague o sufixo ***avelmente*** se em R2);

antemente (apague o sufixo ***antemente*** se em R2); (mod)

mente (apague o sufixo se em R2);

ividade(s) (apague o sufixo ***idade(s)*** se em R2);
icidade(s) (apague o sufixo ***idade(s)*** se em R2);
abilidade(s) (apague o sufixo ***idade(s)*** se em R2);
idade idades (apague o sufixo ***idade(s)*** se em R2);

ativa (s) (apague o sufixo *iva(s)* se em R2);
ativo (s) (apague o sufixo *ivo(s)* se em R2);
iva ivo ivas ivos (apague o sufixo se em R2);
eira (s) (apague o sufixo ***eira(s)*** se em R1 (RV));
ira iras (substitua com ***ir*** se em R1 (RV));

- Descrição da Etapa 2 – Sufixos de Verbos

(http://snowball.tartarus.org/algorithms/portuguese/stemmmer.html) (Lopes, 2004)

(Vide item 2.3.2.3.1 para definição das regiões R1, R2 e RV).

Procure pelo mais longo entre os seguintes sufixos em R1 (RV), e se encontrado, apague:

ada ida ia ária eria iria ará ara erá era irá ava asse esse isse
aste este iste ei arei erei irei am iam ariam eriam iriam aram
eram iram avam em arem erem irem assem essem issem ado ido
ando endo indo arão erão irão ar er ir as adas idas ias
arias erias irias arás aras erás eras irás avas es ardes erdes irdes
ares eres ires asses esses isses astes estes istes is ais eis íeis aríeis
eríeis iríeis áreis areis éreis ereis íreis ireis asseis ésseis ísseis áveis
ados idos ámos amos íamos aríamos eríamos iríamos áramos éramos
iramos ávamos emos aremos eremos iremos ássemos êssemos íssemos
imos armos ermos irmos eu iu ou ira iras
aria ieis arieis (mod)
erieis irieis esseis isseis aveis (mod)
iamos ariamos eriamos iriamos aramos éramos (mod)
avamos assemos essemos issemos (mod)

- Descrição da Etapa 3

(http://snowball.tartarus.org/algorithms/portuguese/stemmmer.html) (Lopes, 2004)

(Vide item 2.3.2.3.1 para definição das regiões R1, R2 e RV).

Apague sufixo ***ci*** se em R1 (RV),

Apague sufixo ***i*** se em R1 (RV).

- Descrição da Etapa 4 – Sufixo residual

(http://snowball.tartarus.org/algorithms/portuguese/stemmmer.html) (Lopes, 2004)

(Vide item 2.3.2.3.1 para definição das regiões R1, R2 e RV).

Se a palavra termina com um dos sufixos: ***os a as i o á í ó*** em R1 (RV), apague-o.

- Descrição da Etapa 5

(http://snowball.tartarus.org/algorithms/portuguese/stemmmer.html) (Lopes, 2004)

(Vide item 2.3.2.3.1 para definição das regiões R1, R2 e RV).

Se a palavra termina com ***cie*** em R1 (RV), apague ***ie***;
Se a palavra termina com ***cié*** em R1 (RV), apague ***ié***;
Se a palavra termina com ***ciê*** em R1 (RV), apague ***iê***;
Se a palavra termina com ***gue*** em R1 (RV), apague ***ue***;
Se a palavra termina com ***gué*** em R1 (RV), apague ***ué***;
Se a palavra termina com ***guê*** em R1 (RV), apague ***uê***;
Se a palavra termina com: ***e é ê*** em R1 (RV), apague;
Se a palavra termina com ***ç*** remova a cedilha

Além das modificações (mod) introduzidas no algoritmo de stemização de Porter proposmo também a inclusão no algoritmo de uma etapa 6 e uma etapa 7:

Se não foi alterada palavra em nenhuma etapa anterior faça etapa 6.
- Etapa 6 – Remova o plural (mod)
- Etapa 7 – Remova os acentos (mod).

- Descrição da etapa 6 (mod):

Se a palavra terminar em ***ões*** altera para ***ão***, *stem* mínimo restante 3 letras;

Se a palavra terminar em ***ães*** altera para ***ão***, *stem* mínimo restante 1 letra;

Se a palavra terminar em ***ais*** alterar para ***al,*** *stem* mínimo restante 1 letra; ;

Se a palavra terminar em ***eis*** altera para ***el***, *stem* mínimo restante 2 letras;

Se a palavra terminar em ***éis*** altera para ***el***, *stem* mínimo restante 2 letras;

Se a palavra terminar em ***ois*** altera para ***ol***, *stem* mínimo restante 2 letras;

Se a palavra terminar em ***óis*** altera para ***ol***, *stem* mínimo restante 2 letras;

Se a palavra terminar em ***les*** altera para ***l***, *stem* mínimo restante 2 letras;

Se a palavra terminar em ***res*** altera para ***r***, *stem* mínimo restante 2 letras;

Se a palavra terminar em ***zes*** altera para ***z***, *stem* mínimo restante 2 letras;

Se a palavra terminar em ***ses*** altera para ***s***, *stem* mínimo restante 2 letras;

Se a palavra terminar em ***is*** altera para ***il***, *stem* mínimo restante 2 letras;

Se a palavra terminar em ***ns*** altera para ***m***, *stem* mínimo restante 2 letras;

Se a palavra terminar em ***s*** retirar, *stem* mínimo restante 2 letras.

3.4.2
Algoritmo de Stemização Modificado *StemmerPortuguese*

Algumas modificações (mod) foram implementadas no algoritmo (vide item 2.3.2.3.2), conforme especificado nos passos a seguir, pois foram encontradas algumas inconsistências:

Passo 0: Inicialmente não retirar o acento, principalmente o til. (mod)

Passo 1. Procure por exceções, tais como: lápis; através; *bit*; *software*, dois, etc. e retorne. (mod)

Passo 2. Faça etapa de Redução do Plural [A]. [vide tabela 33]

Passo 3. Faça etapa de Redução do Feminino [B]. [vide tabela 34]

Passo 4. Faça etapa da Redução do Aumentativo/Diminutivo [C] [vide tabela 36]. Se houve redução vá para passo 9.

Passo 5. Faça etapa da Redução do Advérbio [D] [vide tabela 35]. Se houve redução vá para passo 9.

Passo 6. Se houve redução no passo 2, então vá para o passo 8. Se houve redução no passo 3, então vá para o passo 9. (mod)

Passo 7. Faça etapa da Redução do Sufixo do Nome ou Substantivo [E] [vide tabelas 37A a 37C]. Se houve redução vá para passo 9.

Passo 8. Faça Redução do Sufixo do Verbo [F] [vide tabelas 38A a 38D]. Se houve redução, então vá para passo 10.

Passo 9. Faça etapa da Remoção da Vogal [G]. [vide tabela 39]

Passo 10. Faça etapa da Remoção dos Acentos [H] [vide tabela 40]. Retorne.

No Apêndice 4 estão discriminadas nas tabelas 33 a 40, as regras mostradas nos passos 2[A], 3[B], 5[D], 4[C], 7[E], 8[F], 9[G], 10[H] respectivamente do Algoritmo de Stemização Modificado *StemmerPortuguese* implementado nesse trabalho. Cada regra estabelece: o sufixo a ser removido; o tamanho mínimo do radical restante; o tamanho mínimo da palavra original; evitando que a regra seja aplicada em palavras erroneamente e cada regra tem exceções não mostradas. (http://cpansearch.perl.og/scr/XERN/Lingua-PT-Stemmer-0.01/lib/Lingua/PT/Stemmer..)

3.5
Divisão da Base de Dados em Treinamento e Teste

Foram utilizadas Três Modalidades para a divisão de documentos da Base de Dados, concernentes aos pedidos de patentes, em Conjuntos de Treinamento e Teste. Para os algoritmos 01, 02, 03 e 05 foi utilizado a Primeira Modalidade e para os algoritmos 04 e 06 foram utilizadas a Segunda e a Terceira Modalidade, conforme descrição a seguir.

Para a Primeira Modalidade da pesquisa, a divisão dos documentos entre Conjuntos de Treinamento e Teste foi realizada de acordo com a seguinte regra:

- se a quantidade de documentos da categoria for maior ou igual a 450 (quatrocentos e cinquenta), então a quantidade de documentos para treinamento será igual a 2/3 da quantidade total e a quantidade de documentos para teste será igual a 1/3. A divisão será randômica;

- se a quantidade de documentos da categoria estiver compreendido entre 380 (trezentos e oitenta) inclusive e 450 (quatrocentos e cinqüenta) exclusive, então a quantidade de documentos de treinamento será igual a 300 (trezentos) e a quantidade de documentos de teste será igual a quantidade total menos a quantidade de documentos de treinamento;

- se a quantidade de documentos da categoria estiver compreendido entre 310 (trezentos e dez) exclusive e 380 (trezentos e oitenta) exclusive, então a

quantidade de documentos de treinamento será igual a 300 (trezentos) e a quantidade de documentos de teste será igual a quantidade total de documentos menos a quantidade de documentos de treinamento, acrescido de uma quantidade de documentos já escolhidos para treinamento de tal maneira que o total de documentos de teste perfaça no máximo 80 (oitenta) documentos;

- se a quantidade de documentos da categoria for menor ou igual a 310 (trezentos e dez) então a quantidade de documentos de treinamento será igual a quantidade total de documentos menos 10 (dez) e a quantidade de documentos de teste será igual aos 10 (dez) restantes acrescido de documentos já escolhidos para treinamento, perfazendo no máximo o total de 80 (oitenta) documentos.

Para a Segunda Modalidade da pesquisa, para a Etapa de Teste, foram excluídos os documentos que já tinham sido selecionados para Treinamento. A divisão dos documentos entre Treinamento e Teste obedeceu a seguinte regra:

- se a quantidade de documentos da categoria for maior ou igual a 450 (quatrocentos e cinqüenta) então a quantidade de documentos para treinamento será igual a 2/3 da quantidade total e a quantidade de documentos para teste será igual a 1/3. A divisão será randômica;

- se a quantidade de documentos da categoria estiver compreendido entre 310 (trezentos e dez) inclusive e 450 (quatrocentos e cinqüenta) exclusive, então a quantidade de documentos de treinamento será igual a 300 (trezentos) e a quantidade de documentos de teste será igual a quantidade total menos a quantidade de documentos de treinamento que é de 300 (trezentos);

- se a quantidade de documentos da categoria for menor que 310 (trezentos e dez) então a quantidade de documentos de treinamento será igual a quantidade total de documentos menos 10 (dez) e a quantidade de documentos de teste será igual aos 10 (dez) restantes.

Para uma Terceira Modalidade de pesquisa, todos os documentos foram considerados de Teste. Para essa Modalidade não houve Etapa de Treinamento.

No Apêndice 5 estão indicadas nas tabelas 41 a 47, as quantidades de documentos discriminadas por etapa de treinamento e teste, para as Modalidades 1 e 2, para as seguintes categorias: tabela 41 (categorias H05B e H02G); tabela 42 (categorias H01F e H02M); tabela 43 (categoria H01J); tabela 44 (categoria

H02K); tabela 45 (categoria A47B); tabela 46 (categorias A47C e H02P); e tabela 47 (categoria H02B).

3.6
Transformação dos Dados Por Meio da Indexação

3.6.1
Modelo Espaço-Vetorial (VSM, em inglês *Vector Space Model*)

Adotou-se o Modelo Espaço-Vetorial para representação dos termos dos documentos a serem categorizados. No modelo, a representação é feita a partir de um vetor de termos simples, não ordenado, sem ligações, tendo sido assumido que todos os vetores dos termos tem relacionamentos de mesmo grau, i.e., não há relação direta entre os termos e todos são considerados de mesmo nível. Associado com cada termo no vetor, há um peso, descrevendo seu grau de importância e descrevendo a importância relativa do termo, identificando se o mesmo está presente ou não no texto.

O peso de um termo no vetor deve ser normalizado para uma escala entre zero e um.

O uso do Modelo Espaço-Vetorial pode levar a interpretações erradas, pois o contexto dos termos não é analisado. Por exemplo, o termo "não" pode alterar completamente o significado de uma expressão.

3.6.2
Indexação

As técnicas de indexação usadas nos algoritmos de categorização desenvolvidos foram:

-freqüência de termos modificada (TF') x freqüência de documentos inversa (IDF) (vide item 2.3.5.6);

- freqüência de termos (TF) x freqüência de documentos inversa (IDF) (vide item 2.3.5.4);

- escore de relevância (vide item 2.3.4.6.7).

4.0
CATEGORIZAÇÃO

Vários algoritmos já foram desenvolvidos e testados para a categorização de documentos de patentes (ver item 2.2.2) em idiomas que não o português, tais como: k-Vizinhos-Mais-Próximos (*k*-NN); *Support Vector Machine* (SVM); Bayesiano (Naive Bayes); Árvores de Decisão; Rocchio; Winnow; e algumas variantes desses algoritmos.

Devido a EPO ter usado para a categorização de documentos de patentes, no idioma inglês, o algoritmo *k*-NN e devido a ter obtido bons resultados (Fall et al., 2003a), optou-se para esse estudo, como primeiro algoritmo categorizador a ser explorado, o *k*-Vizinhos-Mais-Próximos (*k*-NN).

Outros categorizadores direcionados para pedidos de patente foram explorados nesse estudo, tendo sido escolhido outros métodos, que não os tradicionais usados em outros idiomas, tais como métodos baseados: em centróides ou conceitos-chaves; e na técnica de *High Order Bit* (HOB).

4.1
Algoritmo *k*-Vizinhos-Mais-Próximos (*k*-NN, em inglês *K-Nearest Neighbor*)

4.1.1
Estado da Técnica

O algoritmo *k*-NN é um algoritmo estatístico, usado na categorização de textos, baseado na comparação texto-texto que tem sido estudado intensivamente por quatro décadas (Hadi, Wa' el Musa et al., 2007, 2008).

O algoritmo *k*-NN é bem simples. O método *k*-Vizinhos-Mais-Próximos (em inglês *k-Nearest Neighbor*) escolhe a categoria de um documento pelas categorias associadas aos seus *k*-vizinhos mais próximos (Hadi, Wa' el Musa et al., 2007, 2008; Baoli et al., 2003).

Separando um conjunto de documentos em treinamento e teste, para um dado documento de teste, o algoritmo acha os *k*-vizinhos mais próximos entre os

documentos de treinamento e usa as categorias desses *k*-vizinhos para categorizar o documento de teste (Hadi, Wa' el Musa et al., 2007, 2008).

Desde que o algoritmo *k*-NN armazena todos os exemplos de treinamento, ele precisa de grande memória. Ele deve procurar através de todos os exemplos disponíveis para categorização os que são semelhantes ao novo documento, portanto ele demanda muito tempo para o processamento de categorização. Também por ele armazenar todos os exemplos no conjunto de treinamento, exemplos ruidosos (i.e., aqueles com erros, no vetor de entrada ou categoria de saída ou aqueles que não são representativos de casos especiais) são também armazenados e podem degradar a exatidão da generalização (Wilson & Martinez, 2000).

Quando se utiliza o algoritmo *k*-NN, depois de se calcular os *k*-vizinhos, muitas estratégias podem ser tomadas para predizer a categoria de um documento de teste. Contudo usualmente para esse método, um valor fixo de *k* é usado para todas as categorias, independente de suas diferentes distribuições (Baoli et al., 2003).

Hadi, Wa'el Musa et al., em seus artigos "*A Comparative Study Using Vector Space Model with k-Nearest Neighbor on Text Categorization Data*" (2007) e "*A Comprehensive Comparative Study Using Vector Space Model with k-Nearest Neighbor on Text Categorization Data*" (2008), usa o algoritmo *k*-NN para a categorização de textos com variações do Modelo Espaço Vetorial (VSM) e usa essa técnica com diversas medidas de similaridade, tais como: Cosseno (ver item 2.3.6.1.1); Jaccard (ver item 2.3.6.1.2); DICE (ver item 2.3.6.1.3). São exploradas, também nos artigos, várias técnicas de pesagem de termos, entre as quais: *IDF* (ver item 2.3.5.1.3); *WIDF* (ver item 2.3.5.1.5); *TF*.*IDF* (ver item 2.3.5.1.4).

Para o experimento, segundo o artigo de Hadi, Wa' el Musa et al. (2007) foram usadas 8 (oito) bases, escolhidas da base de 20NovosGrupos – 20NG (1999). Para cada base de dados foram selecionados 100 (cem) documentos arbitrariamente, sendo que 70% dos documentos foram selecionados para treinamento e 30% para teste. Não foram selecionados textos duplicados e cabeçalhos identificadores, tais como: *Date*; *Follow up-To*; *Path*; *Newsgroups*, *X-*

ref, etc. A medida de desempenho usada foi a Medida de F1 (ver item 5.1) e o valor arbitrado para *k* foi 5 (Hadi, Wa' el Musa et al., 2007).

Para esse método, calcula-se as medidas de similaridade entre cada documento de teste e os documentos dos *k*-vizinhos. Se vários dos *k*-vizinhos-mais-próximos compartilham de uma mesma categoria, então os pesos das similaridades entre esses vizinhos e o documento de teste são somados e a soma resultante é usada como *score* para a categorização do documento de teste. Ordenando os *scores* das categorias candidatas, é obtida uma lista para a categorização dos documentos de teste (Hadi, Wa' el Musa et al., 2007, 2008).

Segundo o artigo de Hadi, Wa' el Musa et al. (2007), usando-se a medida de desempenho da Medida de F1, os resultados obtidos, usando a Medida de Similaridade do Coeficiente de Cosseno com a pesagem *IDF*, apresentaram melhor desempenho do que os obtidos usando: a medida de similaridade do Coeficiente de Cosseno com a pesagem *WIDF*; a medida de similaridade de DICE com a pesagem *IDF*; a medida de similaridade de DICE com a pesagem *WIDF*; a medida de similaridade de Jaccard com a pesagem *IDF*; e a medida de similaridade de Jaccard com a pesagem *WIDF*. Foi constatado que há consistência entre os resultados obtidos quando usado a medida de similaridade do Cosseno com a pesagem *WIDF* e a medida de similaridade do Cosseno com a pesagem *TF.IDF* e que ambos sobrepujam os resultados obtidos quando usado a: medida de similaridade de DICE com pesagem *TF.IDF*; medida de similaridade de DICE com pesagem *WIDF*; medida de similaridade de Jaccard com pesagem *TF.IDF*; medida de similaridade de Jaccard com pesagem *WIDF*. Foram encontradas similaridades entre os resultados obtidos para a Medida de F1 usando-se: a medida de similaridade de DICE com pesagem *TF.IDF* e a medida de similaridade de Jaccard com pesagem *TF.IDF*; a medida de similaridade de DICE com pesagem *WIDF* e a medida de similaridade de Jaccard com pesagem *WIDF* (Hadi, Wa' el Musa et al., 2007).

Em outro artigo de Hadi, Wa'el Musa et al. (2008), foram feitas simulações com os seguintes valores dos *k*-vizinhos: 3; 5; 7; 9; e 11. Segundo o artigo, valores maiores de *k*, reduzem o ruído (em inglês *noise*) no processo de categorização. Os resultados obtidos para a Medida de F1, usando a medida de similaridade do coeficiente de Cosseno foram: para *k* igual a 5 e pesagem

TF.IDF e *WIDF*, de 47.52% e 50.75% respectivamente; e para *k* igual a 11, os resultados obtidos foram 53.08% e 58.43% respectivamente. Os resultados referentes a Medida de F1, usando-se as medidas de similaridade dos coeficientes de Jaccard e DICE foram para *k* igual a 5 e pesagem *TF.IDF* e *WIDF*, de 55.47% e 51.28% respectivamente e para *k* igual a 11, os resultados obtidos foram de 59.67% e 58.50% respectivamente.

Em seu artigo, "*An Improved k-Nearest Neighbor Algorithm for Text Categorization*", Baoli et al. (2003), utiliza o algoritmo *k*-NN para categorização de textos e depois de calcular os *k*-vizinhos, usa 2 (duas) estratégias para predizer a categoria de um documento de teste.

As equações (36) e (37) abaixo mostram 2 (duas) das estratégias usadas para esse método (Baoli et al., 2003):

$$y(d_i) = \max_k \sum_{x_j \varepsilon\ kNN} y(x_j, c_k) \quad \text{.....} \quad (36)$$

$$y(d_i) = \max_k \sum_{x_j \varepsilon\ kNN} Simil(d_i, x_j) \times (x_j, c_k) \quad (37)$$

onde d_i é o documento de teste, x_j é um dos vizinhos no conjunto de treinamento, $y(x_j, c_k)\ \varepsilon\ [0,1]$ indica se x_j pertence a categoria c_k e $Simil(d_i, x_j)$ é a função similaridade entre d_i e x_j.

De acordo com a estratégia usada na equação (36) (Baoli et al., 2003), a categoria selecionada será a que tiver a maior quantidade de documentos de treinamento nos *k*-vizinhos mais próximos (limiar baseado em ordenação – em inglês *rank*). As categorias dos *k*-vizinhos são ordenadas decrescentemente conforme os seus valores de medidas de similaridade e as n categorias melhor colocadas serão escolhidas como as categorias do documento d_i. Assim, para $n = 1$, a categoria de maior importância será a categoria do documento.

De acordo com a estratégia usada na equação (37) (Baoli et al., 2003), a categoria com a soma máxima de similaridades entre os *k*-vizinhos-mais-próximos será a ganhadora (limiar baseado em relevância – em inglês *relevance*). Podem ser utilizados valores limiares por categoria: limiar (c_m). Durante o

processo de categorização, as categorias dos *k*-vizinhos, cujo fator de relevância atingirem esse limiar, relevância $(d_i, c_m) \geq \lim iar(c_m)$ serão definidas como as categorias de d_i.

Segundo Baoli et al. (2003), a estratégia usada na equação (37) é mais difundida do que a descrita na equação (36).

Devido a que a distribuição de documentos nas categorias serem irregulares, pode ser que um valor fixo de *k* resulte em uma escolha incorreta da categoria. Caso seja usada a estratégica de escolha segundo a Relevância, valores pequenos de similaridade irão se acumulando até resultar em um valor total grande, levando a escolha de uma categoria que tenha muitos exemplos. Para superar esse problema foi proposta a estratégia mostrada a seguir (Baoli et al., 2003).

No estudo de Baoli et al. (2003) foram usadas 5 (cinco) configurações para teste, variando-se o valor de *k* com os seguintes valores: 13; 17; 23; 25; e 31. Em cada configuração variou-se o valor de *k*, conforme a quantidade de documentos de treinamento.

Ainda no estudo de Baoli et al. (2003), para a escolha da categoria, é calculada a probabilidade de que um documento pertença a uma categoria usando-se somente os n vizinhos mais próximos de topo para aquela categoria, onde n é derivado de *k* de acordo com a quantidade de documentos da categoria no conjunto de treinamento, ou seja, é usada uma quantidade de vizinhos mais próximos diferentes para cada categoria. Para categorias com grande quantidade de documentos, são usados mais vizinhos próximos. A seleção dinâmica é baseada na distribuição de categorias no conjunto de treinamento. Para a nova estratégia são usadas as probabilidades da divisão entre o somatório dos valores de similaridade de vizinhos pertencendo a uma categoria pelo somatório total de valores de similaridade de todos os vizinhos selecionados para essa categoria. A equação (38) seguinte descreve a função que representa o algoritmo proposto, semelhante à apresentada na equação (37), porém com o valor de *k* variável (Baoli et al., 2003):

$$y(d_i) = \max_m \frac{\sum_{x_j \ \varepsilon \ top-n-kNN(c_m)} Simil(d_i, x_j) \times y(x_j, c_k)}{\sum_{x_j \ \varepsilon \ top-n-kNN(c_m)} Simil(d_i, x_j)} \quad (38)$$

onde

$top - n - kNN(c_m) =$ { n vizinhos de topo do k-vizinhos-mais-próximos originais de k-NN sendo n um número inteiro}.

$$n = \frac{k \times N(c_m)}{\max\{N(c_j) \mid j = 1,...N_C\}} \quad (39)$$

onde $N(c_m)$ denota a quantidade de documentos da categoria c_m no conjunto de treinamento e $\max\{N(c_j) \mid j = 1,...N_C\}$ é a quantidade de documentos de uma dada categoria com mais documentos no mesmo conjunto (Baoli et al., 2003).

Em outro estudo, ainda segundo artigo de Baoli et al. (2003), o *corpus* usado para simulações foi o disponibilizado pelo "*Computer Network and Distributed Systems Laboratory, Department of Computer Science and Technology, Peking Univers*ity" contendo 19892 (dezenove mil, oitocentos e noventa e duas) *web pages* chinesas num total de 19852 (dezenove mil, oitocentos e cinqüenta e dois) documentos distribuídos em 12 (doze) categorias sendo cada documento designado a somente uma categoria. O *corpus* foi dividido em 10 (dez) partes randomicamente e para as simulações foram usadas somente 2 (duas) partes, sendo uma para treinamento e a outra para teste. Os documentos foram representados por meio de um Modelo Espaço-Vetorial (em inglês *Vector Space Model* - VSM). A função de similaridade usada foi a de cosseno (ver item 2.3.6.1.1) e a pesagem usada foi a *TF´.IDF* (ver item 2.3.5.1.6) normalizada. Foram usados 12 (doze) diferentes valores para o parâmetro *k*, variando de 5 até 60, com intervalos de 5.

Os resultados obtidos nos experimentos desenvolvidos por Baoli et al. (2003), entre dois algoritmos, o tradicional *k*-NN (kNN-A) e o *k*-NN proposto (kNN-B), com diferentes valores de *k*, foram os seguintes:

- *k*NN-A conseguiu seus melhores resultados para *k* igual a 10 (dez), com os valores de 72.24% para micro-média Medida de F1 e 68.11% para macro-média Medida de F1;

- *k*NN-B conseguiu seus melhores resultados para *k* igual a 15 (quinze), com os valores de 71.94% para micro-média Medida de F1 e 67.05% para macro-média Medida de F1;

- para macro-média Medida de F1, o resultado para *k*NN-A diminuiu de 68.11% (*k*=10) para 59.04% (*k*=60); e para micro-média Medida de F1, o resultado diminuiu de 72.24% (*k*=10) para 65.93% (*k*=60);

- em média, o desempenho de *k*NN-B, para k igual a 60, foi de pelo menos 1.4% maior do que o obtido para o algoritmo *k*NN-A;

- para diferentes valores de *k*, não houve variações com relação ao desempenho do algoritmo *k*NN-B.

Moraes & Lima (2007) apresentaram no "V Workshop em Tecnologia da Informação e da Linguagem Humana" um trabalho onde foi desenvolvido e implementado um categorizador hierárquico formado por vários classificadores multicategorias, que implementam o algoritmo *k-Nearest Neighbor* (*k*-NN) sobre uma coleção de 26606 (vinte e seis mil, seiscentos e seis) textos jornalísticos, escritos em língua portuguesa, não categorizados, organizados sob títulos de 29 (vinte e nove) seções da Folha de São Paulo, pertencentes ao *corpus* PLN-BR CATEG. O textos foram lematizados pelas ferramentas CHAMA e FORMA, desenvolvidas por Marco Gonzalez e foram retiradas dos textos uma lista de 365 (trezentos e sessenta e cinco) *stopwords*. Os documentos foram representados segundo a abordagem *bag-of-words*, na qual um documento é definido como um vetor de pesos. A técnica *TF´.IDF* (ver item 2.3.5.1.6) foi utilizada para calcular os pesos dos termos. A métrica usada foi a do cosseno e a estratégias usadas para categorização foram: limiar baseada em ordenação (em inglês *rank*), ou seja, $\sum_{z=1}^{k} \cos(d_i, v_z)$ onde $v_Z \in c_m$, d_i é um documento, c_m é a categoria; e limiar baseada em relevância (em inglês *relevance*).

Segundo Moraes & Lima (2007), na primeira estratégia, as categorias dos k-vizinhos são ordenadas conforme seus valores de similaridade e as n

categorias melhores colocadas são escolhidas como as categorias do documento d_i. Foi utilizada sempre $n=1$, ou seja, a categoria de maior similaridade é a categoria do documento. Na segunda estratégia são definidos valores limiares por categoria: limiar(c_m). Durante o processo de categorização, as categorias do *k*-vizinhos cujo fator de similaridade atingirem esse limiar, serão definidos como as categorias de d_i. Foram realizados 5 (cinco) experimentos com o objetivo de analisar parâmetros como o valor de corte usado na seleção de atributos (Min_df) e número k de documentos vizinhos que são considerados durante a execução do algoritmo. A configuração dos experimentos, segundo Moraes & Lima (2007), sendo q a quantidade de documentos, está indicada na Tabela 6.

Tabela 6- Configuração dos Experimentos de Moraes & Lima (2007)

Configuração	Min_df	*k*
1	4	13
2	4	7 se 1<=q<=250 13 se 251<=q<=500 17 se q>500
3	4	13 se 1<=q<=250 17 se 251<=q<=500 23 se q>500
4	3 se 1<=q<=500 4 se q>500	13
5	4 se 1<=q<=250 5 se 251<=q<=500 6 se q>500	13

Segundo Moraes & Lima (2007), da coleção PLN-BR CATEG, os resultados da estratégia de limiar por relevância para configuração 3, foram para valores de macro-média (Precisão–40%; Abrangência–76%; Medida de F1–40%) e para valores de micro-média (Precisão–24%; Abrangência-88%; Medida de F1-37%).

Em seu artigo, Krishnakumar (2006) ilustrou com um Resumo resultados de trabalhos anteriores, realizados com a coleção Reuters – 21578. As tabelas 7A e 7B a seguir mostram os resultados.

Soucy & Mineau (2001a) no seu trabalho intitulado "*A Simple k-NN Algorithm for Text Categorization*" ilustra um algoritmo baseado no *k*-NN

(algoritmo *k*-NN simplificado). Nele o peso dos termos pode ser estabelecido de acordo com vários critérios: freqüência; *TF.IDF* (ver item 2.3.5.1.4); ou por um *score* designado por sua característica de ter a capacidade de dividir exemplos em alguns conjuntos de categorias (i.e. o Ganho de Informação – ver item 2.3.4.1). Para a similaridade entre o documento d_i de teste com o documento x_j de treinamento foi escolhida a função $\cos Sim$ (ver item 2.3.6.1.4), que é particularmente simples, usando a aproximação de pesagem do termo binário.

Tabela 7A - Resultados de Precisão Obtidos com a Coleção Reuters segundo Krishnakumar (2006)

Autor	Quant Doc Trein	Quant Doc Teste	Tópicos	Indexação	Redução da Dimensão	Método	Medida
Joachims	9603	3299	90	Pesagem Tfc	Ganho de Informação	binário	Break-even
Duamis	9603	3299	118	Pesagem boolena	MI	binário	Break-even
Shapire	9603	3299		Pesagem TF.IDF	Nenhuma	Multi classe	Precisão
Weiss	9603	3299	95	Pesagem Frequência Palavras	-	binário	Break-even
Yang	7789	3309	93	LTC	Estatística x^2	binário	Break-even

Tabela 7B - Resultados de Precisão Obtidos com a Coleção Reuters segundo Krishnakumar (2006)

Autor	Rocchio(%)	Bayes(%)	k-NN(%)	Árvore de Decisão(%)	SVM(%)
Joachims	79.9	72.0	82.3	79.4	86.0
Duamis	61.7	75.2	-	-	-
Shapire	(x)	(x)	-	-	(x)
Weiss	78.7	73.4	86.3	78.9	86.3
Yang	75.0	71.0	85.0	79.0	-

(x) – Método foi testado mas foi usada uma medida diferente de break-even.
MI – Informação Mútua (ver item 2.3.4.6)
Ganho de Informação (ver item 2.3.4.1)

Foi usada a equação (40) para estimativa se um documento pertence à categoria c_k (Soucy & Mineau, 2001a):

$$y(d_i) = \arg \max_m \frac{\sum Simil(d_i, x_j) \times y(x_j, c_k)}{\sum_{k_i \varepsilon K} Simil(d_i, x_j)} \quad (40)$$

onde $Simil(d_i, x_j)$ é o valor do resultado da função similaridade entre d_i e x_j usada para comparar o documento d_i de teste com os seus vizinhos (x_j é um dos vizinhos no conjunto de treinamento), $y(x_j, c_k) \varepsilon [0,1]$ indica se x_j pertence a categoria c_k. Isto é, para cada vizinho na vizinhança do conjunto K (de tamanho k) pertencendo à categoria particular c, somamos as similaridades do documento d e dividimos pela soma de todas as similaridades dos k vizinhos do documento d_i. O conjunto de vizinhança K de d_i compreende os documentos n que tem a maior posição de acordo com essa medida (Soucy & Mineau, 2001a).

Na simulação do trabalho de Soucy & Mineau (2001a) foi usada a Seleção de Características que é o processo de selecionar subconjuntos entre todas as características disponíveis. A categorização de documentos de texto pode envolver milhares de características, a maioria delas irrelevantes.

No estudo de Soucy & Mineau (2001a) foi constatado de que a métrica do Ganho de Informação baseada na Entropia (ver item 2.3.4.4) é recomendável para se encontrar características relevantes dos termos visando sua capacidade de discriminação entre categorias. Também foi constatado que removendo todas as características que ocorrem somente uma vez, não aumenta a taxa de erro e deve ser usada como um filtro para aumentar a velocidade antes de se computar o Ganho de Informação. Foi usado no experimento, um método de Seleção de Característica denominada de µ-Ocorrência (vide *A Simple Feature Selection Method for Text Classification*, Soucy & Mineau, 2001a), que reduz agressivamente o tamanho do vocabulário por meio da interação de características. No experimento desenvolvido por Soucy & Mineau (2001a), cada conjunto envolveu duas categorias. Foram usados os categorizadores *k*-NN e o Naive-Bayes para comparação, sendo que não foram usados todos os documentos, sendo um bom valor encontrado o de 2000 (dois mil), selecionados através o Ganho de Informação.

A seguir é apresentado na tabela 8 os resultados encontrados no experimento de Soucy & Mineau (2001a).

Tabela 8 - Resultados Encontrados no Experimento de Soucy & Mineau (2001a)

Base de Dados	Relação Doc. Treinam./ Teste	µ-Ocorrência *k*-NN		µ-Ocorrência Bayes		Bayes 2000 Melhores	Todos Bayes	
		#f	%	#f	%	%	#f	%
WebKBCourse (WebKB)	80/320	35	96.9	35	95	95.6	12150	94.2
Reuters1 (Reuters-21578)	40/360	8	98.3	8	97.4	89.2	3393	81.1
Spam (LingSpam)	40/80	77	95	77	86.3	90	4484	86.3
Prisoner* (WWW)	40/20	9	90	9	70	80	5076	70
Beethoven* (WWW)	40/20	8	85	8	90	85	3327	90
News1* (Usenet)	40/27	130	85.2	130	92.7	96.3	8522	92.6
* Base de Dados Manualmente Criada								

No trabalho de Tikk & Biró (2001) intitulado "*Text Categorization on a Multi-Lingual Corpus*", também é proposto um algoritmo baseado no *k*-NN (algoritmo *k*-NN Melhorado), a qual usa diferentes valores de *k* para diferentes categorias. A distribuição de documentos nas categorias é irregular. No algoritmo tradicional de *k*-NN, o valor de *k* é fixado anteriormente ao processamento. Se *k* é muito grande, categorias com grande quantidade de documentos vão sobrepujar as categorias com pequena quantidade de documentos. Caso *k* seja muito pequeno, o algoritmo não usa a vantagem de se terem muito exemplos de treinamento. Na prática, o valor de *k* é otimizado através de muitas tentativas na etapa de treinamento.

No trabalho de Tikk & Biró (2001) foram usadas para a escolha da categoria: a que tiver a maior quantidade de membros nos *k*-vizinhos mais próximos; e a que tiver maior soma de similaridades entre os vencedores dos *k*-vizinhos mais próximos.

Para esse método (Tikk & Biró, 2001) é usado o algoritmo original *k*-NN e é calculada a probabilidade de que um documento pertença a uma categoria usando-se somente os n-vizinhos mais próximos de topo para aquela categoria, onde n é derivado de k de acordo com a quantidade de documentos da categoria no conjunto de treinamento, ou seja, é usado uma quantidade de vizinhos mais próximos diferentes para as categorias. Para categorias com grande quantidade de documentos, é usado mais vizinhos próximos. A seleção dinâmica é baseada na distribuição de categorias no conjunto de treinamento. Para a estratégia são usadas as probabilidades da proporção entre o somatório da similaridade de vizinhos pertencendo a uma categoria dividido pelo somatório total de similaridade de todos os vizinhos selecionados para essa categoria. As equações que descrevem a função que representa o algoritmo proposto são idênticas às apresentadas nas equações (38) e (39) apresentadas por Baoli et al. (2003).

No trabalho de Tikk & Biró (2001), os documentos são representados pelo Modelo de Espaço-Vetorial (item 2.3.3.1) tendo sido usada a pesagem de termos representada pela freqüência de termos modificada *TF´.IDF* descrita no item 2.3.5.1.6. A função cosseno, descrita no item 2.3.6.1.1 foi usada para computar a similaridade entre documentos.

4.1.2
Simulação do Método 01

Para o algoritmo denominado Método 01 foi usado o Algoritmo de Categorização *k*-Vizinhos-Mais-Próximos (*k*-NN). O método foi definido nos itens 2.3.7.3.3 e 4.1.1.

A base de dados foi agrupada por subclasse, segundo classificação IPC (ver item 3.1) e optou-se por representar o *corpus* por meio de um Modelo Espaço-Vetorial (VSM – ver item 2.3.3.1).

A divisão da Base de Dados em Treinamento e Teste foi realizada segundo a Primeira Modalidade descrita no item 3.5.

Foram retirados os *stopwords* através uma *Stoplist*, conforme item 3.3. Foram tratados os termos compostos.

O método de stemização adotado foi o do Algoritmo Modificado de *StemmerPortuguese* para a Língua Portuguesa (ver item 3.4.2), pois a redução de termos obtida nesse algoritmo foi maior do que a obtida para o algoritmo de Stemização de Porter.

Optou-se para a pesagem aquela adotada no algoritmo de Baoli et al. (2003), ou seja, a Freqüência de Termos Modificada (TF') x Freqüência de Documentos Inversa (IDF), tanto para os documentos de treinamento quanto para os de teste, conforme definida no item 2.3.5.1.6.

Optou-se para a medida de Similaridade aquela adotada por quase todos os algoritmos, ou seja, a do Cosseno (definida no item 2.3.6.1.1) e a Medida do Índice de Similaridade para α, β, γ iguais a 1 (um) (definida no item 2.3.6.1.5), denominada nesse trabalho como ABS. A Medida do Índice de Similaridade usa as técnicas de cosseno, DICE e Jaccard, técnicas também exploradas em muitos algoritmos de categorização, tal como em Hadi et al. (2007, 2008).

Foram usados 5 (cinco) métodos de resolução: resolução 1 (k igual a 13); resolução 2 (k igual a 17); resolução 3 (k igual a 23); resolução 4 (k igual a 25); resolução 5 (k igual a 31). Os valores de k foram considerados fixos, sem levar em consideração a quantidade de documentos de teste.

Foram usadas ainda na etapa de processamento, para definição da categoria, as seguintes estratégias: do Método *Rank* (ver equação 36); e do Método Relevância (ver equação 37); técnicas usadas também nos algoritmos de Baoli et al. (2003), contudo sem levar em consideração o limiar no Método de Relevância.

Para o método *Rank* ou Ordenação a categoria selecionada como vencedora é a que obtém a maior quantidade de documentos de treinamento nos k-vizinhos–mais-próximos. Para o método de Relevância, a categoria selecionada é a que obtém a soma máxima de similaridades entre os k-vizinhos-mais-próximos.

Na tabela 9 a seguir estão ilustradas as técnicas usadas no algoritmo denominado Método 01 e nas técnicas usadas nos algoritmos das anterioridades mais relevantes.

Tabela 9 - Técnicas Usadas no Algoritmo do Método 01 e nas Anterioridades mais Relevantes.

Autor	Mod. VSM	Indexação – Doc. Treinamento/ Teste				Similaridade				Predição		*k*
		(a)	(b)	(c)	(d)	Cos	DICE	Jac	Ind. Simil	Rank	Relev	
Meto-do 01	x	-	-	-	x	x	-	-	x	x	x	(1)
Hadi et al. (2007)	x	x	x	x	-	x	x	x	-	-	x (7)	(2)
Hadi et al. (2008)	x	-	x	x	-	x	x	x	-	-	x (7)	(3)
Baoli et al. (2003)	x	-	-	-	x	x	-	-	-	x	x (7)	(4)
Baoli et al. (2003)	x	-	-	-	x	x	-	-	-	x	x (7)	(5)
Moraes & Lima (2007)	x	-	-	-	x	x	-	-	-	x	x (7)	(6)

(a) IDF (b) WIDF (c) TF.IDF (d) TF'.IDF

(1) k = 13, 17, 23, 25, 31. (2) k = 5. (3) k = 3, 5, 7, 9, 11.
(4) k = 13, 17, 23, 25, 31. (5) k = 5 a 60, intervalos de 5, k variando de acordo com a quantidade de documentos de teste.

(6) k= 7, 13, 17, 23, k variando de acordo com a quantidade de documentos de teste

(7) Método de predição -Relevância com limiar (vide tabela 6).

Método 01 aplicado ao idioma português.

4.2
Algoritmo Classificador Baseado em Centróides ou Lista de Conceitos-Chaves ou Conjunto de Termos Descritores e Funções Difusas

A Categorização por Centróides ou Descoberta por Lista de Conceitos-Chave ou Conjunto de Termos Descritores é baseada na idéia de representar uma categoria com uma lista de termos com os conceitos principais de todos os documentos que fazem parte de uma determinada categoria. O ponto principal está na idéia de que o significado de um texto não é identificado por sua leitura

sequencial, mas por uma análise dos elementos léxicos presentes, ou seja, pelas palavras chaves mais importantes do texto (Gomes, 2005).

Depois da etapa da preparação de textos da base de dados, separa-se a base de dados em documentos de treinamento e de teste e os de treinamento em categorias. Cada documento é analisado e são aplicadas técnicas que facilitam o processo de Seleção de Características. Depois devem ser localizadas no texto, todas as palavras que expressam melhor suas características, ou seja, palavras ou termos que podem definir sua categoria e a partir desses termos é gerada uma lista com os termos comuns a todos os documentos. Essa lista compõe o índice que representará a categoria. Na fase de categorização ou teste, o novo documento a ser categorizada também passa pela etapa de preparação. Após essa etapa também é necessário descobrir as características desse documento para definir a sua lista de termos. A categorização ocorre através de uma comparação entre a lista de termos descritores das categorias e a lista de termos do novo documento a ser categorizado. A categoria que possuir a lista de termos mais similar à lista do documento novo será escolhida como sua categoria.

4.2.1
Estado da Técnica

No trabalho de Galho & Moraes (2003) foi escolhido o Método de Similaridade Difusa como técnica para categorização de novos documentos. A técnica de Similaridade Difusa permite efetuar a categorização graduada de um texto em uma ou mais categorias.

Segundo Galho & Moraes (2003) esse método foi utilizado nos experimentos de Wives (1999) e de Loh (2001) e desde que ambos alcançaram bons resultados em seus projetos, esse método foi utilizado no seu estudo. O processo consiste em determinar o Grau de Semelhança da lista de termos do documento a ser categorizado com a lista de termos de cada categoria. Para cada termo comum à lista de termos do documento a ser categorizado e de uma determinada categoria, calcula-se o Grau de Igualdade (ver equação 41) de seus Escores de Relevância (ver item 2.3.4.7) através de suas funções difusas apresentadas a seguir.

As variáveis a e d representam os pesos do termo no documento a ser categorizado e na categoria (lista de termos descritores da categoria calculada através da técnica de Escore de Relevância na etapa de treinamento) respectivamente (Galho & Moraes, 2003).

A idéia dessa técnica está baseada na frequência de um termo em uma categoria e na sua frequência nas demais categorias. A partir desses dados é calculada a relevância do termo para uma dada categoria. Naquela(s) categoria(s) em que o termo alcançar um Escore de Relevância maior ele será escolhido para representá-lo.

O Grau de Igualdade é calculado pela seguinte equação (41) (Galho & Moraes, 2003):

$$g_i(a,d) = \frac{1}{2}\left[(a \to d) \wedge (d \to a) + (\bar{a} \to \bar{d}) \wedge (\bar{d} \to \bar{a})\right] \quad (41)$$

onde:

$a \to d = \max\{c \varepsilon [0,1] \mid a \times c \leq d\}$ ou d/a

$d \to a = \max\{c \varepsilon [0,1] \mid d \times c \leq a\}$ ou a/d

$\bar{a} = 1 - a$

$\bar{d} = 1 - d$

$\wedge = mínimo$

$\bar{a} \to \bar{d} = \max\{c \varepsilon [0,1] \mid \bar{a} \times c \leq \bar{d}\}$ ou $\bar{d}/\bar{a}$

$\bar{d} \to \bar{a} = \max\{c \varepsilon [0,1] \mid \bar{d} \times c \leq \bar{a}\}$ ou $\bar{a}/\bar{d}$

Os valores calculados devem pertencer ao intervalo [0,1]. Quando isso não acontece existe a necessidade de normalização, assumindo-se o valor 1 quando a implicação resultar em um valor maior que 1 e 0 quando for menor que 0 (Galho & Moraes, 2003).

Em seguida é calculado o somatório dos Graus de Igualdade de todos os termos comuns e divide-se esse valor pelo total de termos do texto e da categoria

juntos, excluindo-se os termos semelhantes. Considerando X o novo documento e Y a categoria, pode-se definir o Grau de Similaridade entre o documento e a categoria pela equação (42) (Galho & Moraes, 2003):

$$G_s(X,Y) = \sum_{h=1}^{k} \frac{g_{ih}(a,d)}{M} \tag{42}$$

onde:

$G_s(X,Y)$ é o Grau de Similaridade entre o documento X e a categoria Y

$g_{ih}(a,d)$ é o Grau de Igualdade (ver equação 41) entre os pesos do termo h (peso a no documento X e peso d na categoria Y)

M é o número total de termos nos documento X e na categoria Y sem contagem repetida.

h é um índice para os termos comuns ao documento X e a categoria Y

k é o número total de termos comuns ao documento X e a categoria Y

Segundo Galho & Moraes (2003), o processo é repetido para cada categoria. A categoria para a qual o texto obtiver maior Grau de Similaridade (ver equação 42) será a categoria sugerida para sua categorização

A base usada no trabalho de Galho & Moraes (2003) foi uma coleção de notícias em língua portuguesa obtida através de jornais e revistas e seção de divulgação de notícias dos principais provedores, do ano de 2003. Foram notícias distribuídas em 5 (cinco) categorias diferentes: Economia; Esportes; Policial; Saúde; e Tecnologia. Cada categoria continha 60 (sessenta) textos, totalizando uma coleção de 300 (trezentos) documentos. Metade da coleção foi usada para treinamento e metade para teste. O grau de precisão total alcançado foi de 91% e o de abrangência foi de 100% para Economia, 73% para Esportes, 97% para Policial, 93% para Saúde e 90% para Tecnologia. O baixo desempenho alcançado na categoria Esportes ocorreu devido ao fato de que os textos considerados para treinamento tratavam de muitos esportes diferentes como mergulho, esgrima, sumo, balonismo, tiro e outros (Galho & Moraes, 2003).

No trabalho de Galho & Moraes (2003) foi feita inicialmente uma preparação do texto, tais como retiradas de todas as palavras que não influenciavam para a definição da categoria do texto, retirada de símbolos, conversão de termos em radicais.

4.2.2
Simulação do Método 02

Para o algoritmo do Método 02, a base de dados foi agrupada por subclasse, segundo classificação IPC (ver item 3.1) e optou-se por representar o *corpus* por meio de um Modelo Espaço-Vetorial (VSM – ver item 2.3.3.1).

A Base de Dados foi dividida em Treinamento e Teste conforme Primeira Modalidade definida no item 3.5.

Foram retirados os *stopwords* através uma *Stoplist*, conforme item 3.3. Foram tratados os termos compostos.

O método de stemização adotado foi o do Algoritmo Modificado de *StemmerPortuguese* para a Língua Portuguesa (ver item 3.4.2), pois a redução de termos obtida nesse algoritmo foi maior do que a obtida para o algoritmo de Stemização de Porter.

O algoritmo do Método 02 foi baseado no artigo "Categorização Automática de Documentos de Texto Utilizando Lógica Difusa" (Galho & Moraes, 2003) onde é usado o Método de Similaridade Difusa (ver item 4.2.1) como técnica para categorização de novos documentos e onde todos os documentos em uma dada categoria formam um conjunto representado por vetores centróides ou protótipos, i.e., um vetor para cada categoria e os pesos de cada termo de cada vetor centróide de cada categoria são calculados através os Escores de Relevância definidos no item 2.3.4.7. Os valores calculados devem pertencer ao intervalo [0,1]. Quando isso não acontece existe a necessidade de normalização, assumindo-se o valor 1 quando a implicação resultar em um valor maior que 1 e 0 quando for menor que 0. Para a representação das categorias foram selecionados todos os termos dos documentos de treinamento.

Para a pesagem dos termos dos documentos de teste, usou-se duas técnicas: para a primeira modalidade usou-se a *TF´.IDF* (ver item 2.3.5.1.6), a mesma

técnica selecionada para o algoritmo do Método 01. Para a segunda modalidade ou modalidade (a), o cálculo dos pesos dos termos dos documentos a serem categorizados (teste) constituiu-se dos seguintes passos, sendo que foram selecionados todos os termos constantes dos documentos de treinamento:

1- para cada documento de teste, seleciona-se todos os termos;

2- para cada termo k, se usa a pesagem do Escore de Relevância definido no item 2.3.4.7, para uma dada categoria t (i.e., W_{tk} é igual a todos os documentos de treinamento pertencendo a uma dada categoria t que contem o termo k acrescido de um e W_{ik} é igual a todos os documentos de treinamento pertencendo a outras categorias que não a categoria t que contem o termo k);

3- repete-se a operação 2 para todas as categorias;

4- seleciona-se entre os pesos encontrados entre todas as categorias o de maior valor;

5- esse peso é o escolhido para o peso do termo k do documento de teste.

O processo dessa técnica consiste em determinar o Grau de Semelhança da lista de termos de um novo documento com a lista de termos de cada categoria ou centróide. Para cada termo comum à lista de termos do documento a ser categorizado e de uma determinada categoria, calcula-se o Grau de Igualdade de seus Escores de Relevância através suas funções difusas apresentadas no item 4.2.1, equação (41).

Em seguida é calculado o somatório dos Graus de Igualdade de todos os termos comuns, dividindo-se este valor pelo total dos termos do documento e da categoria juntos, excluindo-se os termos semelhantes conforme item 4.2.1, equação (42).

O processo é repetido para cada categoria.

Na tabela 10 a seguir estão indicadas as técnicas usadas no algoritmo denominado Método 02 e técnicas usadas no algoritmo da anterioridade mais relevante.

Para o algoritmo do Método 02, a escolha das técnicas foram baseadas em Galho & Moraes (2003), com exceção do método de pesagem (indexação) para os

documentos de teste, onde foram consideradas duas modalidades: uma com pesagem *TF'.IDF* ; e outra com Escore de Relevância Modificado.

Tabela 10 - Técnicas Usadas no Algoritmo do Método 02 e nas Anterioridades mais Relevantes.

Autor/ Método	Mod VSM	Indexação (Documento Treinamento)	Indexação (Doc. Teste)			Similaridade Difusa	
		Escore de Relevancia	TF.IDF	TF'.IDF	Escore Relev. Modificado	Grau de Igualdade	Grau de Similaridade
Metodo 02	x	x	-	x	x	x	x
Galho & Moraes (2003)	x	x	x	-	-	x	x
Método 02 e Galho & Moraes (2003) ambos aplicados ao idioma português.							

4.3
Algoritmo Classificador Baseado na Característica do Centróide das Categorias (em inglês *Class-Feature-Centroid* - CFC)

4.3.1 – Estado da Técnica

Guan, Hu et al. (2009), em seu artigo "*A Class-Feature-Centroid Classifier for Text Categorization*", descreve em seu artigo um método de categorização de textos baseado no conceito do centróide para cada categoria, considerando características de distribuição de termos: internamente a categoria; e entre categorias.

Segundo o artigo, um vetor protótipo, i.e. um centróide, foi construído para cada categoria. Os pesos dos vetores centróides não foram normalizados. Os pesos dos vetores dos documentos foram normalizados. Quando se categoriza um novo documento, o vetor representando o documento é comparado com os vetores protótipos e o documento é designado à categoria cujo vetor protótipo é o mais similar (Guan, Hu et al., 2009).

O *corpu*s foi representado por meio de um Modelo Espaço-Vetorial (VSM), onde todos os documentos em uma categoria formam um conjunto e um documento é representado por um vetor de pesos normalizado (Guan, Hu et al., 2009).

Segundo o artigo, para a elaboração do vetor centróide para cada categoria (C_j), foi usada a equação (43) a seguir (Guan, Hu et al., 2009):

$$w_{ij} = b^{DF/|Cj|} \times \ln\left(\frac{|C|}{CF_t}\right) \quad (43)$$

onde

w_{ij} = vetor de termos t_i do vetor centróide para categoria Cj;

t_i = termo sem repetição de cada documento da categoria Cj;

b = constante > 1;

DF = frequência do termo t_i na categoria Cj;

Cj = quantidade de documentos na categoria Cj;

$|C|$ = quantidade de categorias C;

CF_t = quantidade de categorias contendo termo t_i;

$b^{DF/|C_j|}$ é o índice interno do termo na categoria

$\ln\left(\frac{|C|}{CF_t}\right)$ é o índice entre categorias do termo na categoria

Na equação (43), o primeiro componente é o índice de termo interno da categoria e o segundo componente representa o índice de termo entre categorias (Guan, Hu et al., 2009).

O primeiro termo ou o termo interno da categoria é decorrente de que se o termo aparece muitas vezes em documentos da categoria C, então um documento de teste contendo o termo tem mais probabilidade de ser da categoria C. O peso interno do termo da categoria é limitado ao domínio $(1, b]$ e o denominador C_j suaviza a diferença da frequência de documentos através das categorias (Guan, Hu et al., 2009).

O segundo termo da equação (43) ou o termo entre categorias é decorrente de que se o termo aparece somente em umas poucas categorias, então o termo é um bom discriminante dessas categorias. Se o termo aparece em todas as categorias então ele não é um bom discriminante da categoria. Se o termo aparece

em todas as categorias então o valor se torna zero. Quando o termo ocorre em somente uma categoria, o valor se torna $\ln(|C|)$ (Guan, Hu et al., 2009).

Segundo o artigo, os pesos dos termos dos documentos, na etapa de teste, são calculados segundo o score $TF.IDF$ relatado no item 2.3.5.1.4. Para o cálculo da Similaridade, foi adotada a Medida do Cosseno (ver item 2.3.6.1.1 e equação 44) sem normalização, pois segundo Guan, Hu et al., dessa maneira a capacidade de discriminação das características é preservada (Guan, Hu et al., 2009):

$$C' = \arg \max (d_i \times Centroide_j) \quad (44)$$

onde

d_i é o vetor documento para documento i normalizado;

Centróide j é o vetor centróide da categoria j não normalizado.

No artigo de Guan, Hu et al. (2009) foi comparado o algoritmo proposto com outros algoritmos baseados em 3 (três) métodos: Vetor Centróide baseado na Média Aritmética; Vetor Centróide baseado na Soma Acumulada; técnica de *SVMLight*, *SVMTorch* e *LibSVM*. Os testes foram realizados usando o banco de dados *Reuters* –21578 (vinte e um mil, quinhentos e setenta e oito) e o banco de dados formado com 19997 (dezenove mil, novecentos e noventa e sete) textos distribuídos em 20 (vinte) novos grupos (Guan, Hu et al., 2009).

Para a base de dados *Reuters* foram removidos: *stopwords*; termos unigramos que ocorreram menos de três vezes; e documentos que pertenciam a mais de uma categoria. Foram incluídos categorias que tinham no mínimo um documento, tanto no grupo de treinamento, quanto no grupo de teste. Para as palavras do título foram dados pesos 10 (dez) maiores do que as correspondentes incluídas no Resumo. Restaram 6495 (seis mil, quatrocentos e noventa e cinco) documentos de treinamento e 2557 (dois mil, quinhentos e cinquenta e sete) documentos de teste, distribuídos em 52 (cinquenta e duas) categorias (Guan, Hu et al., 2009).

Para o banco de dados de novos grupos, consistindo de 19899 (dezenove mil, oitocentos e noventa e nove) textos agrupados em 20 (vinte) grupos, foram removidos *stopwords*, e expressões tais como: *"SUBJECT"; "sender"; "path", etc.* (Guan, Hu et al., 2009).

Não foi aplicada a stemização e nem o agrupamento de palavras no algoritmo (Guan, Hu et al., 2009).

Segundo o artigo, para o banco de dados *Reuters*, o método CFC foi o que apresentou melhores resultados, com *micro*-Medida de F1 e *macro*-Medida de F1, valores respectivamente iguais a 99.41% e 99.60%; e os piores resultados foram obtidos para a Média Aritmética e a Soma Acumulada, que apresentaram ambos os valores de 86.47% e 73.44% respectivamente. Para a base de 20 (vinte) grupos os valores obtidos foram de 92.72%, 92.75% e 83.07%, 82.92% respectivamente (Guan, Hu et al., 2009).

Segundo o artigo de Guan, Hu et al. (2009), a não-normalização do vetor centróide só se aplica para o método CFC, pois o seu uso no método da Média Aritmética e da Soma Acumulada gera degradação nos resultados.

4.3.2
Simulação do Método 03

Para o Método 03, a base de dados foi agrupada por subclasse, segundo classificação IPC (ver item 3.1).

A Base de Dados foi dividida em Treinamento e Teste conforme Primeira Modalidade definida no item 3.5.

Para esse Método, o *corpus* foi representado por meio de um Modelo Espaço-Vetorial (VSM), onde todos os documentos em uma categoria formam um conjunto e um documento é representado por um vetor de pesos.

Foram retirados os *stopwords* através uma *Stoplist*, conforme item 3.3. Foram tratados os termos compostos.

O método de stemização adotado foi o Algoritmo Modificado de *StemmerPortuguese* para a Língua Portuguesa (ver item 3.4.2), pois a redução de termos obtida nesse algoritmo foi maior do que a obtida para o algoritmo de Stemização de Porter.

Para o cálculo dos pesos dos termos em cada documento, para os documentos de teste foi adotada a pesagem normalizada *TF´.IDF* descrita no item 2.3.5.1.6, a mesma adotada para o algoritmo do Método 01.

Foi usada a técnica descrita no item 4.3.1, i.e, a construção de um vetor protótipo para cada categoria. Para o vetor centróide não foi feita a normalização dos pesos. A constante b adotada foi a sugerida no artigo de Guan, Hu et al (2009), ou seja, [exp(1) – 1.7].

Quando se categoriza um documento novo, o vetor representando o documento é comparado com os vetores protótipos de cada categoria e o documento é designado à categoria cujo vetor protótipo é o mais similar. Foram usadas as medidas de Similaridade: do Cosseno (ver item 2.3.6.1.1); de Jaccard (ver item 2.3.6.1.2); de DICE (ver item 2.3.6.1.3); e Índice de Similaridade (ver item 2.3.6.1.5).

Nesse algoritmo, os valores do pesos dos termos do vetor centróide foram computados com os dados dos documentos de treinamento e para os resultados foram considerados os documentos de teste.

Na tabela 11 a seguir estão indicadas as técnicas usadas no algoritmo do Método 03 e no algoritmo da anterioridade mais relevante.

Tabela 11 - Técnicas Usadas no Algoritmo do Método 03 e na Anterioridade Mais Relevante.

Autor	Mod. VSM	Indexação			Similaridade			
		Centroide	Doc. Teste					
		Escore de Relevância	TF.IDF	TF'.IDF	Cosseno	DICE	Jaccard	Indice Simil.
Método 03	x	x	-	x	x	x	x	x
Guan, Hu et al. -2009	x	x	x	-	x	-	-	-
Método 03–aplicado ao idioma português; Guan et al. - aplicado ao idioma inglês.								

Para esse algoritmo (Método 03), a escolha das técnicas foi baseada no algoritmo de Guan, Hu et al. (2009), com exceção: do método de pesagem (indexação) para os documentos de teste; e do cálculo de similaridade, que além do método do cosseno, foram usados também os métodos de Jaccard, DICE, e Índice de Similaridade.

4.4

Algoritmo Usando a Média Aritmética dos Pesos dos Termos dos Documentos da Categoria para o Vetor Centróide

4.4.1
Estado da Técnica

Guan, Hu et al., em seu artigo "*A Class-Feature-Centroid Classifier for Text Categorization*" (2009), cita métodos clássicos usados na categorização de textos, baseados no conceito do centróide, i.e., um vetor protótipo para cada categoria (C_j), onde seus pesos são formados por uma combinação dos pesos de todos os documentos de uma dada categoria. Entre os métodos citados é relatado que o "*Arithmetical Average Centroid* (AAC)", ou seja, a média aritmética de todos os pesos dos vetores dos documentos de uma categoria é o mais comum entre os métodos de inicialização baseado no conceito do centróide.

O centróide, ou seja, o vetor protótipo de uma dada categoria, é um vetor com todos os termos distintos de uma dada categoria e seus pesos são a média aritmética de todos os pesos dos termos dos documentos de treinamento d da dita categoria C_j e Q_{cj} é a quantidade de termos distintos de todos os documentos da categoria C_j conforme equação (45) :

$$Centróide_j = \left[\frac{1}{Q_{cj}}\right]\sum_{d \varepsilon C_j} d \tag{45}$$

Esse método foi usado no artigo de Guan, Hu et al. (2009), visando comparação com o método proposto CFC, descrito no item 4.3.1, tendo sido exposto no artigo, que os resultados obtidos não foram bons, quando comparados com o CFC.

A pesagem dos documentos obedeceu ao método de *TF.IDF* definida no item 2.3.5.1.4. Não foi aplicada a stemização e nem o agrupamento de palavras no algoritmo.

Segundo o artigo, para o banco de dados *Reuters* e o banco de 20-novos-grupos, o método da Média Aritmética apresentou os seguintes resultados, para micro-média Medida de F1 e macro-média Medida de F1 valores,

respectivamente iguais a 86.47% e 73.44%; e 83.97% e 82.92%, i.e., os piores resultados quando comparados com o método de CFC e técnicas de *SVMLight*, *SVMTorch* e *LibSVM*.

4.4.2 Simulação do Método 04

Para o algoritmo denominado Método 04, foi usada a técnica da "Média Aritmética dos Pesos dos Termos dos Documentos da Categoria para o Vetor Centróide", onde todos os documentos em uma dada categoria formam um conjunto representado por vetores centróides, um para cada categoria e os pesos de cada termo de cada vetor centróide de uma categoria específica são calculados pela média aritmética de todos os pesos dos ditos termos dos vetores de todos os documentos da categoria específica. Portanto, um vetor protótipo, i.e. um centróide, foi construído para cada categoria.

Para esse Método, o *corpus* foi representado por meio de um Modelo Espaço-Vetorial (VSM), onde todos os documentos em uma categoria formam um conjunto e um documento é representado por um vetor de pesos (ver item 2.3.3.1).

A Base de Dados foi agrupada por subclasse, segundo classificação IPC (ver item 3.1) e dividida em Treinamento e Teste conforme Segunda Modalidade definida no item 3.5. Para esse método, os documentos de Teste foram selecionados distintivamente dos de treinamento.

Foram retirados os *stopwords* através uma *Stoplist*, conforme item 3.3. Foram tratados os termos compostos.

O método de stemização adotado foi: o Algoritmo Modificado de Stemização *StemmerPortuguese* (ver item 3.4.2), pois a redução de termos obtida nesse algoritmo foi maior do que a obtida para o algoritmo de Stemização de Porter.

Para o algoritmo do Método 04, usou-se para cada documento de treinamento e de teste de uma dada categoria a seguinte pesagem $TF'.IDF$ descrita em 2.3.5.1.6, a mesma utilizada para o algoritmo do Método 01.

Para a Medida de Similaridade, optou-se pela medida de Similaridade de Cosseno (definida no item 2.3.6.11), tanto para os documentos de treinamento, quanto para os documentos de teste.

Na tabela 12 a seguir estão ilustradas as técnicas usadas no algoritmo do Método 04 e no algoritmo da anterioridade mais relevante.

Para esse algoritmo (Método 04), a escolha das técnicas foi baseada no algoritmo de Guan, Hu et al. (2009), com exceção do método de pesagem (indexação) para os documentos de teste.

Tabela 12 - Técnicas Usadas no Algoritmo do Método 04 e na Anterioridade Mais Relevante.

Autor	Mod. VSM	Indexação			Similaridade Cosseno
		Centróide	Doc. Teste		
		Média Aritmética dos Pesos dos Termos dos Documentos da Categoria	TF.IDF	TF'.IDF	.
Método 04	x	x	-	x	x
Guan, Hu et al. (2009)	x	x	x	-	x

Método 04 – aplicado ao idioma português.
Guan, Hu et al. - aplicado ao idioma inglês.

4.4.3 Simulação do Método 06

Para o algoritmo denominado Método 06, foi usada também a técnica da "Média Aritmética dos Pesos dos Termos dos Documentos da Categoria para o Vetor Centróide", onde todos os documentos em uma dada categoria formam um conjunto representado por vetores centróides, um para cada categoria e os pesos de cada termo de cada vetor centróide de uma categoria específica são calculados pela média aritmética de todos os pesos dos ditos termos dos vetores de todos os documentos da categoria específica. Portanto, um vetor protótipo, i.e. um centróide, foi construído para cada categoria.

Para esse Método, o *corpus* foi representado por meio de um Modelo Espaço-Vetorial (VSM), onde todos os documentos em uma categoria formam um

conjunto e um documento é representado por um vetor de pesos. (ver item 2.3.3.1).

A Base de Dados foi representada em nível de grupo, agrupada no nível mais alto, segundo classificação IPC (ver item 3.1) e dividida em Treinamento e Teste conforme Segunda Modalidade definida no item 3.5. Para esse método, os documentos de Teste foram selecionados distintivamente dos de treinamento.

Foram retirados os *stopwords* através uma *Stoplist*, conforme item 3.3. Foram tratados os termos compostos.

O método de stemização adotado foi o do Algoritmo Modificado de Stemização *StemmerPortuguese* (ver item 3.4.2), pois a redução de termos obtida nesse algoritmo foi maior do que a obtida no algoritmo de Stemização de Porter.

Para esse algoritmo, usou-se para cada documento de treinamento e de teste de uma dada categoria a seguinte pesagem $TF´.IDF$, descrita em 2.3.5.1.6, a mesma técnica usada para o algoritmo do Método 01.

Para a Medida de Similaridade, optou-se pela medida de Similaridade de Cosseno (definida no item 2.3.6.11), tanto para os documentos de treinamento, quanto para os documentos de teste.

Na tabela 13 a seguir estão indicadas as técnicas usadas no algoritmo do Método 06 e no algoritmo da anterioridade mais relevante.

Para esse algoritmo (Método 06), a escolha das técnicas foi baseada no algoritmo de Guan, Hu et al. (2009), com exceção do método de pesagem (indexação) para os documentos de teste.

Tabela 13 - Técnicas Usadas no Algoritmo do Método 06 e na Anterioridade Mais Relevante.

Autor	Mod. VSM	Indexação			Similaridade Cosseno
		Centróide	Doc. Teste		
		Média Aritmética dos Pesos dos Termos dos Documentos Categoria	TF.IDF	TF'.IDF	
Método 06	x	x	-	x	x
Guan, Hu et al. (2009)	x	x	x	-	x
Método 06 – aplicado ao idioma português. Guan, Hu et al. - aplicado ao idioma inglês					

4.5
Algoritmo Usando a Distância HOB

4.5.1
Estado da Técnica

Um novo modelo para representação de dados textuais, baseado na idéia de colocar dados em intervalos pré-definidos, usando a tecnologia da distância HOB (em inglês *High Order Bit*) é apresentado por Khan (2001) e Rahal & Perrizo (2004).

Segundo Khan (2001) e Rahal & Perrizo (2004), inicialmente a matriz S de documentos de treinamento x termos deve ter uma representação usando a técnica de pesagem $TF.IDF$, definida no item 2.3.5.1.4, intitulada Frequência de Termos (TF) x Frequência de Documentos Inversa (IDF) e depois essa representação deve ser normalizada para valores de medidas entre 0 e 1. Na fase de dividir os dados em intervalos, deve-se decidir a quantidade de intervalos e o domínio de cada um. Depois se substituem os valores dos pesos dos termos dos vetores de documentos pelos seus respectivos intervalos. Se forem usados 4 (quatro) intervalos lógicos, então cada intervalo pode ser definido por: I0 = [0, 0]; I1 = (0, 0.35]; I2 = (0.35, 0.70]; I3 = (0.70, 1]. Se forem usados 8 (oito) intervalos lógicos, cada intervalo pode ser definido por: I0 = [0, 0]; I1 = (0, 0.15]; I2 = (0.15, 0.30]; I3 = (0.30, 0.45]; I4= (0.45, 0.60]; I5 = (0.60, 0.75]; I6 = (0.75, 0.9]; I7 = (0.9, 1.0], onde "(" e ")" são exclusivos e "[" e "]" são inclusivos. A mesma representação deve ter o documento a ser categorizado (Khan, 2001) (Rahal & Perrizo, 2004).

A quantidade ótima de intervalos e seus domínios dependem do tipo de documentos, devendo haver uma ordenação no intervalo escolhido, de tal maneira que I0 < I1 < I2 < I3 ou I0 < I1 < I2 < I3 < I4 < I5 < I6 < I7, dependendo se estão sendo usados 4 (quatro) ou 8 (oito) intervalos respectivamente (Khan, 2001) (Rahal & Perrizo, 2004).

Caso seja usado uma representação de 4 (quatro) intervalos I0, I1, I2 e I3, os valores dos pesos dos termos serão representados por uma representação de 2 (dois) bits ou seja, 00, 01, 10 e 11 respectivamente. Caso seja usado uma

representação de 8 (oito) intervalos I0, I1, I2, I3, I4, I5, I6 e I7, os valores dos pesos dos termos serão representados por uma representação de 3 (três) bits ou seja, 000, 001, 010, 011, 100, 101, 110 e 111 respectivamente (Khan, 2001) (Rahal & Perrizo, 2004).

Deve ser ordenado, em ordem decrescente, o conjunto de todos os termos do conjunto S de documentos de treinamento, obedecendo à ordem também decrescente dos valores dos termos intervalizados representantes do documento de teste a ser categorizado (Khan, 2001) (Rahal & Perrizo, 2004).

Cada documento é um vetor de termos representado por intervalos de valores. A similaridade entre dois documentos d_1 e d_2 pode ser medido pela quantidade de termos comuns. Um termo t é considerado a ser comum entre d_1 e d_2 se o valor do intervalo dado ao termo t em ambos os documentos é o mesmo. Maior a quantidade de termos comuns em d_1 e d_2, maior o grau de similaridade entre eles. Entretanto, nem todos os termos participam igualmente da similaridade. A ordem do intervalo participa da similaridade. Se usarmos 8 (oito) intervalos, i.e., I0, I1, I2, I3, I4, I5, I6 e I7 onde I0<I1<I2<I3<I4<I5<I6<I7, então termos comuns, tendo intervalos com valores maiores tal como I7 contribuem mais para a similaridade do que termos tendo intervalos com valores menores tal como I1 (Khan, 2001).

Usando termos comuns para medir a similaridade entre documentos, precisamos checar quão perto termos não-comuns estão. Se documentos d_1 e d_2 têm para certos termos, intervalos com valores diferentes, então maiores e mais próximos estiverem esses intervalos, maior o grau de similaridade entre d_1 e d_2. Por exemplo, se o termo t tiver um valor 111 em d_1 e um valor 001 em d_2, então a similaridade entre d_1 e d_2 será maior do que se o termo t tivesse um valor 010 em d_1 e um valor 000 em d_2 porque 111 contribui mais para o contexto de d_1 do que 010 e o mesmo se aplica para d_2. Entretanto, a similaridade entre $d1$ e $d2$ seria maior se comparada ao primeiro caso, se o termo t tiver um valor 111 em $d1$ e um valor 101 em $d2$ porque o *gap* entre 111 e 101 é menor do que o *gap* entre 111 e 001 ou o *gap* entre 010 e 000 (Khan, 2001).

Resumindo, a similaridade entre documentos é baseada: na quantidade de termos comuns entre dois documentos; pela proximidade dos intervalos de termos não-comuns; e pelos valores dos intervalos por si só (valores maiores significam similaridades maiores) (Khan, 2001).

Para categorizar o novo documento d_{novo}, o algoritmo deve encontrar os vizinho mais próximo. Primeiro devemos ordenar a Matriz Documentos de Treinamento x Termos S de acordo com a ordenação decrescente dos valores intervalizados do vetor de teste d_{novo} que se deseja categorizar. Depois se procura por categoria, quantos termos da matriz S são idênticos em todos os 3 (três) *bits*, ou seja, $d(t_1,t_2)=0$ e depois de se achar as quantidades de termos similares separadamente, esses valores são multiplicados pelos pesos de voto baseados nas suas similaridades com d_{novo} convertidos para numérico acrescido de um. Esses valores são guardados num vetor $w(c_i)$ onde c_i é a categoria selecionada (Khan, 2001).

Em seguida, procuramos pelos termos que são idênticos nas 2 (duas) posições mais significativas, não levando em conta o terceiro *bit* à direita, ou seja $d(t_1,t_2)\leq 1$. Refazemos a contagem de termos idênticos para todos os documentos por categoria e repete-se a operação de multiplicação e armazenamento. Remove-se mais um bit da direita, e assim sucessivamente (Khan, 2001).

O inconveniente desse método é que a expansão não ocorre nos dois lados simetricamente e a exatidão se torna baixa. Por exemplo, para a representação 101 (5), inicialmente procuramos pelos vizinhos (5,5). Quando retiramos o bit menos significativo, a representação torna-se 10-, e passamos a procurar pelos vizinhos (4,5). Retirando mais um bit menos significativo, a representação torna-se 1—, e passamos a procurar pelos vizinhos (4,7). O centro da vizinhança torna-se (4+7)/2=5.5, no entanto o valor procurado é 5, além do tamanho da vizinhança ser expandida pela potência de 2 (Khan, 2001).

Outro método proposto é a expansão da vizinhança usando o Centro Perfeito. Nesse método expandimos a vizinhança exatamente por 1, em ambos os lados do domínio mantendo o valor requerido precisamente no centro. Na primeira expansão da vizinhança procuramos pelos vizinhos $[a-1,a+1]$,

baseados na distância de Minkowski para $p = \infty$ ou seja, a distância MAX (Khan, 2001):

$$d_{\infty}(X,Y) = \max_{i=1}^{n} |x_i - y_i| \tag{48}$$

A seguir, procuramos pelos vizinhos $[a-2, a+2]$, aumentando a distância de 1. Como estamos trabalhando com valores de somente 3 (três) bits, a computação não é muito maior quando comparada com o primeiro método (Khan, 2001).

Da mesma maneira que o método anterior, para toda categoria c_i, um loop é realizado através de todos os termos t em d_{novo} e é calculado a quantidade de termos (mais próximos) tendo o mesmo valor de t com d_{novo} sendo que é multiplicado esses valores pelos pesos de voto baseados nas suas similaridades com d_{novo}. Repita o procedimento para $[a-1, a+1]$ e $[a-2, a+2]$. Armazene esses valores num vetor $w(c_i)$ onde c_i é a categoria selecionada (Khan, 2001).

4.5.2
Simulação do Método 05

No experimento do Método 05, utilizou-se a Distância HOB (em inglês *High Order Bit*), técnica da distância descrita no trabalho de Khan (2001) intitulado "*Fast Distance Metric Based Data Mining Using P-Trees; k-Nearest–Neighbor Classification and k-Clustering*".

A Base de Dados foi agrupada por subclasse, segundo classificação IPC (ver item 3.1) onde todos os documentos em uma categoria formam um conjunto e um documento é representado por um vetor de pesos. Optou-se por representar o *corpus* por meio de um Modelo Espaço Vetorial (VSM - ver item 2.3.3.1)

A Base de Dados foi dividida em Treinamento e Teste conforme Primeira Modalidade definida no item 3.5.

Foram retirados os *stopwords* através uma *Stoplist*, conforme item 3.3. Foram tratados os termos compostos.

O método de stemização adotado foi o Método de Stemização StemerMétodo01 (radicalizador *StemmerPortuguese*) para a Língua Portuguesa (ver item 3.4.2), pois a redução de termos obtida nesse algoritmo foi maior do que a obtida para o algoritmo de Stemização de Porter.

A pesagem de termos usada, tanto na etapa de treinamento quanto na etapa de teste, foi a definida no item 2.3.5.1.6, intitulada Frequência de Termos Modificada (TF´) x Frequência de Documentos Inversa (IDF), a mesma técnica usada para o algoritmo do Método 01.

Optou-se por representar os valores dos termos da matriz documentos x termos por 8 (oito) intervalos, sendo os valores dos pesos dos termos representados por 3 (três) bits, ou seja: 000 (I0); 001 (I1); 010 (I2); 011 (I3); 100 (I4); 101 (I5); 110 (I6); e 111 (I7).

Como foram usados 8 (oito) intervalos lógicos, definiu-se para cada intervalo, tanto para os documentos de treinamento, quanto para os de teste: I0 = [0, 0]; I1 = (0, 0.15]; I2 = (0.15, 0.30]; I3 = (0.30, 0.45]; I4= (0.45, 0.60]; I5 = (0.60, 0.75]; I6 = (0.75, 0.9]; I7 = (0.9, 1.0], onde "(" e ")" são exclusivos e "[" e "]" são inclusivos.

Os valores dos pesos dos termos dos vetores dos documentos de treinamento e de teste foram substituídos pelos seus respectivos intervalos.

A similaridade entre dois documentos d_1 e d_2 pode ser medido pela quantidade de termos comuns. Um termo t é considerado a ser comum entre d_1 e d_2 se o valor do intervalo dado ao termo t em ambos os documentos é o mesmo. Maior a quantidade de termos comuns em d_1 e d_2, maior o Grau de Similaridade entre eles. Entretanto, nem todos os termos participam igualmente na similaridade. A ordem do intervalo participa na similaridade (Rahal & Perrizo, 2004).

Usando termos comuns para medir a similaridade entre documentos, precisamos checar quão perto termos não-comuns estão. Se documentos d_1 e d_2 têm para certos termos, intervalos com valores diferentes, então maiores e mais próximos estiverem esses intervalos, maior o grau de similaridade entre d_1 e d_2 (Rahal & Perrizo, 2004).

Resumindo, a similaridade entre documentos é baseada: na quantidade de termos comuns entre dois documentos; pela proximidade dos intervalos de termos não-comuns; e pelos valores dos intervalos por si só (valores maiores significam similaridades maiores) (Rahal & Perrizo, 2004).

Para o experimento, optou-se para o Algoritmo do Método 05, tanto para os documentos de teste quanto para os documentos de treinamento, os seguintes valores para INT, $INT1$ e $INT2$ (valores decimais para representação binária):

- se o peso na base decimal for igual a zero (intervalo I0 ou binário 000) então faço INT igual a 0 (similaridade em todos os bits); $INT1$ igual a 0 (similaridade nos dois bits de maior ordem, ou seja, 00); $INT2$ igual a 0 (similaridade no bit de maior ordem, ou seja, 0);

- se peso na base decimal for maior que zero e menor ou igual a 0.15 (intervalo I1 ou binário 001) então faço INT igual a 1 (similaridade em todos os bits); $INT1$ igual a 0 (similaridade nos dois bits de maior ordem, ou seja 00) igual a 0; $INT2$ igual a 0 (similaridade no bit de maior ordem, ou seja, 0);

- se peso na base decimal for maior que 0.15 e menor ou igual a 0.30 (intervalo I2 ou binário 010) então faço INT igual a 2 (similaridade em todos os bits); $INT1$ igual a 1 (similaridade nos dois bits de maior ordem, ou seja 01); $INT2$ igual a 0 (similaridade no bit de maior ordem, ou seja 0);

- se peso na base decimal for maior que 0.30 e menor ou igual a 0.45 (intervalo I3 ou binário 011) então faço INT igual a 3 (similaridade em todos os bits); $INT1$ igual a 1 (similaridade nos dois bits de maior ordem, ou seja 01); $INT2$ igual a 0 (similaridade no bit de maior ordem, ou seja 0);

- se peso na base decimal for maior que 0.45 e menor ou igual a 0.60 (intervalo I4 ou binário 100) então faço INT igual a 4 (similaridade em todos os bits); $INT1$ igual a 2 (similaridade nos dois bits de maior ordem, ou seja 10); $INT2$ igual a 1 (similaridade no bit de maior ordem ou seja 1);

- se peso na base decimal for maior que 0.60 e menor ou igual a 0.75 (intervalo I5 ou binário 101) então faço INT igual a 5 (similaridade em todos os bits); $INT1$ igual a 2 (similaridade nos dois bits de maior ordem, ou seja 10); $INT2$ igual a 1 (similaridade no bit de maior ordem, ou seja 1);

- se peso na base decimal for maior que 0.75 e menor ou igual a 0.90 (intervalo I6 ou binário 110) então faço INT igual a 6 (similaridade em todos os

bits); $INT1$ igual a 3 (similaridade nos dois bits de maior ordem, ou seja 11); $INT2$ igual a 1 (similaridade no bit de maior ordem, ou seja 1);

- se peso na base decimal for maior que 0.90 e menor ou igual a 1 (intervalo I7 ou binário 111) então faço INT igual 7 (similaridade em todos os bits) igual a 7; $INT1$ igual a 3 (similaridade nos dois bits de maior ordem, ou seja 11); $INT2$ igual a 1 (similaridade no bit de maior ordem, ou seja 1).

Devido a que nem todos os termos participam igualmente na similaridade e que se documentos $d1$ e $d2$ têm para certos termos, intervalos com valores diferentes, então maiores e mais próximos estiverem esses intervalos, maior o grau de similaridade entre $d1$ e $d2$, então usamos diferentes pesos para a verificação das similaridades dos intervalos.

Testou-se o algoritmo para 2 (duas) Modalidades. Para a primeira modalidade (Modalidade 05), os fatores de pesos acham-se discriminados no Apêndice 12, tabelas 131 e 132. Na tabela 131 acham-se discriminados os pesos, levando-se em consideração todos os bits, para os termos comuns e não-comuns. Na tabela 132 acham-se discriminados os pesos, levando-se em consideração um e dois bits de maior ordem, somente para os termos comuns. Para a segunda modalidade (Modalidade 05V1), no Apêndice 12, tabela 133 acham-se discriminados os pesos, levando-se em consideração todos os bits para os termos comuns e os termos não-comuns. Nas tabelas 134 (134A e 134B) e 135 do Apêndice 12, encontram-se os pesos, levando-se em consideração os dois e um bit de Maior Ordem, respectivamente, para os termos comuns e os termos não-comuns.

O algoritmo do Método 05, para a Modalidade 05V1, se baseia nas seguintes etapas, depois da pesagem intervalizada:

1. para cada documento de teste, pego todos os termos com os devidos pesos;

2- faça um somatório referente a cada categoria igual a zero;

3- para cada documento de teste específico, pego todos os documentos de treinamento, um de cada vez;

4- para um determinado documento de treinamento pego todos os termos com os devidos pesos;

5- para cada termo comum ao documento de teste e ao de treino, vou fazendo o somatório (referente a categoria específica do documento de treino), do peso do termo no documento de teste multiplicado pelo fator de peso, para verificação da similaridade dos intervalos (veja tabelas do Apêndice 12);

6- pego outro documento de treino e vou para a etapa 4 até não ter mais documentos de treino;

7- para cada documento de teste tenho um conjunto de somatórios, um para cada categoria;

8- coloco esses somatórios em ordem decrescente;

9- o somatório de maior valor indicará qual categoria o documento de teste será categorizado;

10- pego outro documento de teste e repito a operação indo para a etapa 1.

Para o algoritmo do Método 05 e Modalidade 05V1, levou-se em consideração para verificação das similaridades dos intervalos: os termos comuns e os não-comuns, levando-se em consideração tanto os três bits de maior ordem quanto os dois bits e um bit de maior ordem.

Para o algoritmo do Método 05 e Modalidade 05, levou-se em consideração os termos comuns e os não-comuns para verificação das similaridades dos intervalos, levando-se em consideração os três bits de maior ordem. Para os de dois bits e um bit de maior ordem, levou-se em consideração somente os termos comuns.

Para um dado documento de teste, o algoritmo acha o Vizinho-Mais-Similar entre os documentos de treinamento e usa a categoria desse vizinho para categorizar o documento de teste (Prognóstico de Topo) ou usa os Três-Vizinhos-Mais-Similares para categorizar o documento de teste, podendo ser a categoria correta qualquer uma dessas categorias (Três Prognósticos de Topo).

Na tabela 14 acham-se discriminadas as técnicas usadas no Algoritmo do Método 05 e no algoritmo da anterioridade mais relevante .

Tabela 14 - Técnicas Usadas no Algoritmo do Método 05 e na Anterioridade Mais Relevante.

Autor	Mod. VSM	Documentos de Treino e. Teste		Intervalos Logicos (Quant. Bits)	Similaridade		Categorização	
					Quant. Termos Comuns (1)		Todos Documentos de Treinamento (Vizinhos)	k-Vizinhos
		TF.IDF	TF'.IDF		Proximidade de Termos Não Comuns (2)	Proximidade de Termos Não Comuns (1)		
					Valores Intervalos	Valores Intervalos		
Método 05	x	-	x	8 (3)	x	x	x	-
Khan (2001)	x	x	-	(8)	Usando a Árvore P e *k*-NN		-	x

(1) Levando-se em consideração todos os Três Bits, Dois e Um Bit de Maior Ordem

(2) Levando-se em consideração todos os Três Bits

Método 05 - Aplicado ao idioma português.

Khan – Aplicado ao idioma inglês.

4.6. Várias Técnicas Adotadas

Na tabela 15 a seguir é discriminado para cada algoritmo de categorização (Método 01; Método 02; Método 03; Método 04; e Método 05) as várias técnicas adotadas, sendo que a Base de Dados foi agrupada por subclasse, segundo classificação IPC (ver item 3.1). Para o Método 06, considerou-se que os documentos de teste tinham sido categorizados acertadamente em suas categorias em nível de subclasse e testaram-se todos os documentos, em nível de grupo, agrupados no nível mais alto.

Tabela 15 - Discriminação dos Algoritmos de Categorização Simulados

Algoritmo	**Discriminação**
1.*k*-Nearest Neighbor (*k*-NN)	Peso dos documentos de treinamento e teste: TF' x IDF [normalizado] Similaridade entre doc. teste e treinamento: cosseno/ ABS Conjunto de vizinhança → $y(d_i) = \max_k \sum y(x_j, c_k)$ (*rank*) Conjunto. vizinhança. → $y(d_i) = \max_k \sum Simil(d_i, x_j)\ .\ y(x_j, c_k)$ (relevância) Resoluções: k=13; 17; 23; 25; 31.
2.Método Baseado em Lista de Termos Descritores	Centróide (treinamento): Escore de Relevância [normalizado] Peso do documento de teste: TF'x IDF ou Escore de Relevância [norm.] Similaridade (centróide e doc. teste): $g_i(a,d)$ = ½ [(a→d) ^ (d→a) + (ā→đ) ^ (đ→ ā)] Escolha da categoria: $G_s(X,Y) = \frac{\sum_{i=1}^{k} g_{ih}(a,d)}{M}$
3.Classif. Baseado Centróide por Categorias	Centróide (treinamento): $w_{ij} = b^{DF/\|C\ j\|} \times \ln(\|C\| / CF_t)$ [não normalizado] Peso do documento de teste: TF' x IDF [normalizado] Similaridade entre centróide e doc. teste: cosseno; DICE; Jaccard; Índice de Similaridade.
4 e 6. Método de Vetor Centróide Média Aritmética	$Centróide_j = \left[\frac{1}{Q_{cj}}\right] \sum_{d \varepsilon C_j} d$ [normalizado] Peso dos documentos de teste e de treinamento: TF' x IDF [normalizado] Similaridade entre centróide e doc. teste: cosseno
5- Método Usando Distância HOB.	Peso dos documentos de treinamento e de teste: TF'.IDF [normalizado] Intervalização dos pesos: I0 = [0, 0]; I1 = (0, 0.15]; I2 = (0.15, 0.30]; I3 = (0.30, 0.45]; I4= (0.45, 0.60]; I5 = (0.60, 0.75]; I6 = (0.75, 0.9]; I7 = (0.9, 1.0]. Similaridade: HOB [quantidade de termos comuns entre dois documentos; pela proximidade dos intervalos dos termos não-comuns; valores dos intervalos por si só]

5.0
Pós-Processamento

Sistemas de Recuperação de Informação (RI) podem ser avaliados sob dois enfoques: avaliação de desempenho, quanto ao tempo gasto no processamento e quanto ao espaço de memória exigido; e quanto a avaliação de desempenho de recuperação, ou seja, quanto à qualidade da saída de dados (Gonzalez, 2005).

5.1
Medidas de Desempenho

Em geral, os sistemas de Recuperação de Informação (RI) são avaliados através do uso de duas métricas precisão (em inglês *precision*) e abrangência (em inglês *recall*) (Hadi et al., 2008). Usa-se também a métrica exatidão (em inglês *accuracy*) e Medida de F1 (em inglês *F-measure*).

Supondo-se uma classificação binária de D documentos, ou seja, o documento pertencente a classe c ou não, a tabela Verdade *versus* Falso pode ser representada pela tabela 16 a seguir.

Tabela 16 – Tabela Verdade *versus* Falso para uma Classificação Binária.

Prognóstico **Real**	**Classe c**	**Classe não c**
Classe c	Positivo Verdadeiro (True Positive) TP	Negativo Falso (False Negative) FN
Classe não c	Positivo Falso (False Positive) FP	Negativo Verdadeiro (True Negative) TN

As várias métricas citadas acima são obtidas pelas fórmulas abaixo (Hadi et al, 2008) (Guo et al., 2004) (Khattak & Heyer, 2011):

$$\text{Pr}ecisão = \frac{TP}{TP + FP} \times 100 \qquad (49)$$

$$Abrangência = \frac{TP}{TP + FN} \times 100 \quad (50)$$

$$Exatidão = \frac{TP + TN}{TP + FP + TN + FN} \quad (51)$$

Admitindo-se que tanto a Precisão tanto quanto a Abrangência são importantes, se utiliza também a Medida de F1 (em inglês *F-measure*) calculada pela seguinte fórmula (Guo et al., 2004) (Khattak & Heyer, 2011):

$$F = \frac{2 \times \text{Pr}ecisão \times Abrangência}{\text{Pr}ecisão + Abrangência} \quad (52)$$

Usam-se também as medidas de micro-média (em inglês *microaveraging*) e macro-média (em inglês *macroaveraging*) para se obter o resultado para a coleção toda. *Microaveraging* considera a coleção toda como uma única categoria e então avalia os graus de Abrangência e Precisão sem distinção de categorias. A medida de *Macroaveraging* primeiro calcula a Precisão e Abrangência em cada categoria e então extrai os valores médios para a coleção toda (Guo et al., 2004).

Fall, C. J et al. (2003a), em seu artigo "*Automated Categorization in the International Patent Classification*" descreve alguns esquemas de medidas de desempenho, entre os quais destacamos: Prognóstico de Topo; e Três Prognósticos. No esquema de Prognóstico de Topo, a categoria prognosticada de topo é comparada com a categoria principal do documento. No esquema de Três Prognósticos, as três primeiras categorias prognosticadas de topo são comparadas com a categoria principal do documento. Se houver acerto em qualquer um dos três prognósticos, o resultado é considerado correto (Fall, C. J et al., 2003a)

Segundo Tikk & Biró (2001, 2003), além dos valores de Prognóstico de Topo e Três Prognósticos, ainda é definida a seguinte medida:

- Cinco Prognósticos: Comparam-se as cinco categorias de topo encontradas no categorizador com todas as categorias associadas com o documento. A quantidade de suposições corretas é contada e normalizada para cada documento e o resultado final é acumulado sobre toda a coleção.

A tabela 17 a seguir ilustra a medida de desempenho representada por vários prognósticos ou sugestões (Tikk & Biró, 2001).

Tabela 17 - Medidas de Desempenho Representadas por Vários Prognósticos.

Prognóstico de topo		**Três Prognósticos**		**Cinco Prognósticos**	
Prognóstico	**Real**	**Prognósticos**	**Real**	**Prognósticos**	**Real**
Categoria de Topo 1 →	Categoria principal 1	Categoria de Topo 1 → Categoria de Topo 2 → Categoria de Topo 3 →	Categoria principal 1	Categoria de Topo 1 → Categoria de Topo 2 → Categoria de Topo 3 → Categoria de Topo 4 → Categoria de Topo 5 →	Categoria principal 1 ou Categoria principal 2 ou Categoria principal 3 ... ou Categoria principal n

Consideramos também nesse trabalho, a Medida de Desempenho de Dois Prognósticos, ou seja, qualquer uma das duas primeiras categorias prognosticadas de topo serão comparadas com a categoria principal do documento. Se houver acerto em qualquer um dos dois prognósticos, o resultado é considerado correto.

6.0
Resultados das Simulações

Foram executadas as seguintes simulações: Método 01 (ver item 4.1) onde foi usado o Algoritmo de Categorização *k* -Vizinhos-Mais-Próximos *k*-NN (em inglês *k-Nearest Neighbor*); Métodos 02 e 03 (ver itens 4.2 e 4.3 respectivamente) onde foram usados Algoritmos baseados em Lista de Termos Descritores, Conceitos-Chave ou Centróides; Métodos 04 e 06 (ver item 4.4) onde foi usado o Método do Vetor Centróide baseado na Média Aritmética dos Termos Descritores; Método 05 (ver item 4.5) onde foi usada a técnica da Distância HOB. Para o Método 06 os dados foram representados, segundo classificação IPC, em nível de grupo, agrupados no nível mais alto e nos outros Métodos os dados foram representados em nível de subclasse.

As simulações foram desenvolvidas na linguagem C++ Builder versão 2009, usando-se o *SQL Server Management Studio* como Banco de Dados, distribuídos nas seguintes etapas: Entrada de Dados; Eliminação dos *StopWords* e Tratamento de Palavras Compostas; Stemizações baseadas nos radicalizadores de Porter e *StemmerPortuguese*; Divisão de Dados em Treinamento e Teste; e Algoritmos de Categorização incluindo Medidas de Desempenho.

6.1
Stemização

A tabela 18 a seguir mostra a quantidade de termos dos documentos que constituem o Banco de Dados usado nesse estudo, discriminados pelos seguintes Métodos de Stemização: *StemmerPortuguese* (StemerMetodo01); Stemização de Porter (StemerMetodo02); e sem stemização (Palavra). No total para o Método de Stemização *StemmerPotuguese* houve uma redução de 8.54% da quantidade total dos termos e para o método de Stemização de Porter houve uma redução de 6.12%. A maior redução de termos ocorreu: para o método de Stemização *StemmerPortuguese* (10.04%) para a categoria H02P; e para o método de Stemização de Porter, as maiores reduções ocorreram para as categorias: H05BI (6.60%); H02M (6.57%); e H01F(6.46%). A média de termos por documento foi aproximadamente de 41 (quarenta e um).

A tabela 19 mostra a quantidade distinta dos termos dos documentos considerados por categoria e por Método de Stemização.

Tabela 18 - Quantidade de Termos Discriminados por Método de Stemização

Categorias	Quant. Termos Discriminados por Método Stemização			Total Documentos	Média Termos por Doc.
	StemerMetodo01	StemerMetodo02	Palavra		
A47B	16352 (91.91%)	16697 (93.85%)	17791 (100%)	447	40
H05BA	12606 (91.15%)	12957 (93.69%)	13830 (100%)	294	47
H02G	19149 (91.35%)	19628 (93.63%)	20963 (100%)	500	42
A47C	11999 (93.08%)	12232 (94.89%)	12891(100%)	329	39
H02P	11603 (89.96%)	12092 (93.75%)	12898 (100%)	307	42
H02M	10082 (90.75%)	10380 (93.43%)	11110 (100%)	274	41
H05BI	8619 (91.60%)	8788 (93.40%)	9409 (100%)	205	46
H01F	15710 (91.35%)	16086 (93.54%)	17197 (100%)	434	40
H02K	17809 (91.33%)	18362 (94.16%)	19500 (100%)	500	39
H02B	11760 (91.09%)	12111 (93.81%)	12910 (100%)	293	44
H01J	9982 (92.66%)	10198 (94.66%)	10773 (100%)	278	39
Média Total	145671 (91.46%)	149531 (93.88%)	159272 (100%)	3861	41

Tabela 19– Quantidade de Termos Distintos por Categoria

Categorias	Quantidade Termos Distintos Discriminados por Método de Stemização			Total de Documentos
	StemerMetodo01	StemerMetodo02	Palavra	
A47B	2510	2989	4550	447
H05BA	2419	2807	4077	294
H02G	2852	3406	5190	500
A47C	2360	2773	4034	329
H02P	2113	2497	3641	307
H02M	2035	2381	3394	274
H05BI	2009	2269	3206	205
H01F	2791	3336	4972	434
H02K	2769	3307	4968	500
H02B	2346	2758	4081	293
H01J	2382	2757	3879	278
Total	26586 (57.81%)	31280 (68.01%)	45992 (100%)	3861

Levando-se em consideração a quantidade de termos distintos por categoria, a redução dos termos usando-se o radicalizador de *StemmerPortuguese* foi de 42.19% e para o radicalizador de Porter foi de 31.99%.

6.2
Categorização

6.2.1
Método 01

Para o Método 01, os métodos de desempenho usados foram a Precisão e a Abrangência para cada categoria.

Ainda foram usados como Medidas de Desempenho as seguintes medidas: Prognóstico de Topo; e Três Prognósticos de Topo. As medidas de Similaridade usadas foram Cosseno e Índice de Similaridade (nesse trabalho denominada de ABS) para α, β, e γ iguais a 1(um). O método de stemização usado foi o do *StemmerPortuguese* (StemerMétodo01)

Muitos resultados foram obtidos com as simulações. Foi calculada a média dos vários resultados obtidos (precisão e abrangência) entre as categorias (*macroaveraging*): H05B(A); H05B(I); H02G; H02B; H01F; H02K; H02M; H02P; H01J; A47B; e A47C. A Medida de F1 final foi calculada a partir dos valores médios da Precisão e Abrangência, ambos obtidos por *macroaveraging*.

As tabelas 48 a 87, constantes do Apêndice 6, mostram os resultados, encontrados nas simulações referentes ao Método 01 para as Resoluções 1 a 5, para as Medidas de Similaridade do Cosseno e do Índice de Similaridade (ABS) e para o Método de Stemização de *StemmerPortuguese* (StemerMetodo01), referentes as medidas de Precisão, Abrangência e Medida de F1 e para os métodos de desempenho de Prognóstico de Topo e Três Prognósticos de Topo. Nas tabelas 20A a 20D se encontram os resultados encontrados nas várias simulações referentes ao algoritmo do Método 01 e Resoluções 1 a 5 (valores médios).

Tabela 20 A– Resultados Encontrados nas Várias Simulações Referentes ao Método 01 e Resoluções 1 e 2.

Técnicas	Método de Similaridade	Abrangência Média	Precisão Média	Medida de F1	Categorias com maiores valores de Abrangência	Categorias com menores valores de Abrangência
Resolução 1 / Prognóstico de Topo	RankCosseno	0.6887	0.6994	0.6940	A47B (0.932) H01J(0.895)	H01F(0.448) A47C(0.507) H02G(0.515)
	Relevância Cosseno	0.7983	0.7725	0.7852	H05BA(0.949) H02P(0.937) H02M(0.935) H01J(0.934)	H01F(0.388) H02G(0.569)
	RankABS	0.6588	0.6697	0.6642	A47B (0.973) H01J (0.788)	H01F (0.396) H02P (0.50)
	Relevância ABS	0.7141	0.7157	0.7149	A47B (0.979) H05BA(0.875)	H01F (0.373) H02G(0.563)
Resolução 1 / Três Prognósticos de Topo	RankCosseno	0.9134	--	--	H02P (1.0) A47B (0.986)	H01F (0.672)
	Relevância Cosseno	0.9344	--	--	H05BA (1.0) H02P (1.0) H05BI (1.0) H01J (1.0)	H01F (0.6567)
	RankABS	0.9072	--	--	A47B(0.993) H05BA(0.987)	H01F(0.672)
	Relevância ABS	0.9263	--	--	H05BA(1.0) H05BI(1.0)	H01F(0.649)
Resolução 2 / Prognóstico de Topo	RankCosseno	0.6868	0.6957	0.6912	A47B (0.932) H01J(0.855)	H01F(0.403) H02G(0.521) A47C(0.547)
	Relevância Cosseno	0.7769	0.7750	0.7759	H05BA(0.938) A47B(0.925)	H01F(0.403) H02G(0.569)
	RankABS	0.6902	0.7033	0.6967	A47B (0.98) H05BA (0.838)	H01F (0.388) H02P (0.563)
	Relevância ABS	0.7340	0.7268	0.7304	A47B (0.98) H05BA(0.899)	H01F (0.366) H02G(0.587)

Tabela 20B – Resultados Encontrados nas Várias Simulações Referentes ao Método 01 e Resoluções 2 e 3.

Técnicas	Método de Similaridade	Abrangência Média	Precisão Média	Medida de F1	Categorias com maiores valores de Abrangência	Categorias com menores valores de Abrangência
Resolução 2 / Três Prognósticos de Topo	RankCosseno	0.9019	--	--	H02P (1.0) H05BA(0.987) A47B (0.986)	H01F (0.657)
	Relevância Cosseno	0.9358	--	--	H01J (1.0) H05BA (1.0) H05BI (1.0)	H01F (0.657)
	RankABS	0.9064	--	--	H05BA(1.0) A47B(0.993)	H01F(0.604)
	Relevância ABS	0.9292	--	--	H05BA(1.0) A47B(1.0) H05BI(1.0)	H01F(0.634)
Resolução 3 / Prognóstico de Topo	RankCosseno	0.6878	0.7020	0.6949	A47B (0.932) H01J(0.829)	H01F(0.366) A47C(0.507) H02G(0.515)
	Relevância Cosseno	0.7850	0.7699	0.7774	H05BA(0.924) A47B(0.918)	H01F(0.358) H02G(0.563)
	RankABS	0.6962	0.7057	0.7009	A47B (0.986) H05BA(0.823)	H01F (0.403) H02P (0.544)
	Relevância ABS	0.7338	0.7329	0.7333	A47B(0.986) H05BA(0.899)	H01F (0.381) H02G(0.581)
Resolução 3 / Três Prognósticos de Topo	RankCosseno	0.9140	--	--	H02P (0.987) H05BA(0.987) A47B (0.986)	H01F (0.671)
	Relevância Cosseno	0.9430	--	--	H05BA (1.0) H05BI(1.0) H01J (1.0)	H01F (0.694)
	RankABS	0.9081	--	--	A47B(1.0) H05BA(0.975) H05BI(0.974)	H01F(0.642)
	Relevância ABS	0.9167	--	--	A47B(1.0) H05BA(1.0) H05BI(1.0)	H01F(0.634)

Tabela 20C – Resultados Encontrados nas Várias Simulações Referentes ao Método 01 e Resoluções 4.e 5.

Técnicas	Método de Similaridade	Abrangência Média	Precisão Média	Medida de F1	Categorias com maiores valores de Abrangência	Categorias com menores valores de Abrangência
Resolução 4 / Prognóstico de Topo	Rank Cosseno	0.6912	0.7116	0.7013	A47B (0.939) H01J(0.829)	H01F(0.373) A47C(0.493) H02G(0.539)
	Relevância Cosseno	0.7820	0.7718	0.7769	H05BA(0.937) A47B(0.932)	H01F(0.366) H02G(0.557)
	RankABS	0.6923	0.6978	0.6950	A47B (0.973) H05BA(0.810)	H01F (0.366) H02G (0.557) H02P(0.557)
	Relevância ABS	0.7265	0.7266	0.7265	A47B(0.973) H05BA(0.886)	H01F (0.358) H02G(0.575)
Resolução 4 / Três Prognósticos de Topo	Rank Cosseno	0.9242	--	--	H05BI(1.0) H02P (0.987) H05BA(0.987) A47B (0.986)	H01F (0.649)
	Relevância Cosseno	0.9454	--	--	H05BA (1.0) H05BI(1.0) H02P(1.0) H01J (1.0)	H01F (0.701)
	RankABS	0.9098	--	--	A47B(1.0) H05BA(0.987) H05BI(0.974)	H01F(0.634)
	Relevância ABS	0.9203	--	--	A47B(1.0) H05BA(1.0) H05BI(1.0)	H01F(0.634)
Resolução 5 / Prognóstico de Topo	Rank Cosseno	0.6877	0.7122	0.6997	A47B (0.939) H01J(0.842)	H01F(0.396) A47C(0.533) H02G(0.533)
	Relevância Cosseno	0.7828	0.7742	0.7785	H05BA(0.949) A47B(0.939)	H01F(0.373) H02G(0.539)
	RankABS	0.7123	0.7184	0.7153	A47B (0.98) H05BA(0.835)	H01F (0.403) H02G (0.611) H02B (0.615) H02P (0.62)
	Relevância ABS	0.7365	0.7371	0.7368	A47B(0.98) H05BA(0.886)	H01F (0.381) H02G(0.617)

Tabela 20D – Resultados Encontrados nas Várias Simulações Referentes ao Método 01 e Resolução 5

Técnicas	Método de Similari-dade	Abrang ência Média	Precisão Média	Medida de F1	Categorias com maiores valores de Abrangência	Categorias com menores valores de Abrangência
Resolu ção 5 / Três Prognósti cos de Topo	RankCosseno	0.922	--	--	H02P (1.0) H05BI(0.987) A47B (0.986)	H01F (0.642)
	Relevância Cosseno	0.9435	--	--	H05BA (1.0) H02P (1.0) H01J (1.0)	H01F (0.664)
	RankABS	0.8862	--	--	A47B(0.993) H05BA(0.95) H05BI(0.95)	H01F(0.627)
	Relevância ABS	0.8945	--	--	A47B(0.993) H05BA(0.963) H05BI(0.95)	H01F(0.642)

A categoria selecionada para o método de Ordenação (em inglês *Rank*) foi a que teve maior quantidade de documentos de treinamento nos *k*-Vizinhos-Mais-Próximos. Para o método de Relevância, a categoria selecionada foi a que obteve a soma máxima de similaridade entre os *k*-Vizinhos-Mais-Próximos.

Para todos os resultados encontrados nas simulações referentes ao Método 01, Prognóstico de Topo e para todas as 5 (cinco) Resoluções, os melhores resultados, referente a Medida de F1, calculada a partir dos valores médios de Precisão e Abrangência (*macroaveraring*), ocorreram usando-se as técnicas do Cosseno e Relevância (RelevânciaCos). O mesmo ocorreu para a Medida de Desempenho de Três Prognósticos de Topo.

Com relação as técnicas de Resolução, Método de Desempenho de Prognóstico de Topo, Método de Similaridade do Cosseno e método de predição de Relevância, em média, para a Medida de F1 calculada, não houve mudanças significativas entre os resultados obtidos entre as Resoluções (78.52%, 77.59%, 77.74%, 77.69%, 77.85%), entretanto para a Resolução 1 (78.52%), o resultado foi um pouco superior.

Referindo-se as técnicas de Resolução, Método de Desempenho de Três Prognósticos de Topo, Método de Similaridade do Cosseno e Método de predição de Relevância, em média, os resultados obtidos para a Abrangência (93.44%, 93.58%, 94.30%, 94.54%, 94.35%), a Resolução 4 apresentou resultado um pouco melhor do que os obtidos para as outras resoluções.

Referente as técnicas de Prognóstico de Topo, Relevância, Cosseno e Resolução 1, as categorias que apresentaram os melhores resultados, para a Medida de Abrangência, em ordem decrescente foram: H05B(A) (94.9%); H02P (93.7%); H02M (93.5%); H01J (93.4%); A47B (91.8%); H05B(I) (88.5%); H02B (82.1%); A47C (73.3%); e H02K (71.3%). Os piores foram: H02G (56.9%); e H01F (38.8%).

Na tabela 21A acham-se discriminados os valores de Abrangência encontrados no Método 01 para as categorias: A47B; H01J; H05B(A); H02K; e A47C; e na tabela 21B para a categoria H01F.

Para a categoria A47B, para o Método de Desempenho de Prognóstico de Topo, o melhor resultado, para Resolução 1, foi obtido para as técnicas de Relevância/Índice de Similaridade; para as Resoluções 2 a 5, os melhores resultados foram obtidos para as técnicas Rank/Índice de Similaridade e Relevância/Índice de Similaridade.

Para a categoria A47B, para o Método de Desempenho de Três Prognósticos de Topo, para a Resolução 1, os melhores resultados foram obtidos tanto para as técnicas de Rank/Índice de Similaridade e Relevância/Índice de Similaridade quanto para a técnica de Relevância/Cosseno. Para Resolução 2, o melhor resultado foi obtido para as técnicas de Relevância/Índice de Similaridade e para as Resoluções 3 a 5, os melhores resultados foram obtidos tanto para as técnicas de Rank/Índice de Similaridade quanto para as técnicas de Relevância/Índice de Similaridade.

Para as categorias H05B(A) e H01J, para o Método de Desempenho de Prognóstico deTopo, as técnicas que obtiveram os melhores resultados foram a da Relevância/Cosseno, para todas as resoluções. Para a categoria H05B(A), Método de Desempenho de Prognóstico de Topo, para as Resoluções 1 e 5, os resultados obtidos para as técnicas de Relevância/Cosseno foram um pouco melhores do que

as obtidas para as outras Resoluções. Para a categoria H01J, Método de Desempenho de Prognóstico de Topo, o melhor resultado ocorreu para a Resolução 1.

Para Três Prognósticos de Topo, para a categoria H05B(A), tanto para as técnicas de Relevância/Cosseno, para todas as Resoluções, quanto para as técnicas de Relevância/Índice de Similaridade, para as Resoluções 1 a 4, foram obtidos resultados com valores máximos. Para a Resolução 5, o melhor resultado ocorreu para a técnica de Relevância/Cosseno. Para Três Prognósticos de Topo, para a categoria H01J, as técnicas de Relevância/Cosseno, para todas as Resoluções, obtiveram também resultados máximos.

Para a categoria H02K, Prognóstico de Topo e Resolução 1, os melhores resultados foram obtidos para as técnicas de Rank/Cosseno e Relevância/Cosseno. Para as Resoluções 2 e 5, os melhores resultados foram obtidos para as técnicas de Relevância/Cosseno e para as Resoluções 3 e 4, os melhores resultados foram obtidos para as técnicas de Rank/Cosseno. O melhor resultado foi obtido para as Resoluções 4 e 5.

Para a categoria H02K, para Três Prognósticos de Topo, o melhor resultado para a Resolução 1, foi obtido para as técnicas de Rank/Cosseno. Para as Resoluções 2 e 3, os melhores resultados foram obtidos para as técnicas de Relevância/Cosseno e para as Resoluções 4 e 5, os melhores resultados foram obtidos tanto para as técnicas de Rank/Cosseno quanto para a de Relevância/Cosseno. O melhor resultado foi obtido para Resolução 3.

Para a categoria A47C, para o Método de Desempenho de Prognóstico de Topo, para a Resolução 1, o melhor resultado foi obtido para as técnicas de Relevância/Cosseno. Para as Resoluções 2 a 5, os melhores resultados foram obtidos para as técnicas de Relevância/Índice de Similaridade. O melhor resultado foi obtido para a Resolução 5.

Para a categoria A47C, para Três Prognósticos de Topo, os melhores resultados, para todas as resoluções, foram obtidos para as técnicas de Relevância/Cosseno.

Tabela 21 A– Valores de Abrangência Obtidos para o Método 01 e para as Categorias A47B, H01J, H05BA, H02K, A47C

Categoria	Resolução	Prognóstico de Topo				Três Prognósticos de Topo			
		Método A	Método B	Método C	Método D	Método A	Método B	Método C	Método D
A47B 147 doc.	1	93.2%	91.8%	97.3%	98%	98.6%	99.3%	99.3%	99.3%
	2	93.2%	92.5%	98%	98%	98.6%	99.3%	99.3%	100%
	3	93.2%	91.8%	98.6%	98.6%	98.6%	98.6%	100%	100%
	4	93.9%	93.2%	97.3%	97.3%	99%	99%	100%	100%
	5	93.9%	93.9%	98%	98%	98.6%	98.6%	99.3%	99.3%
H01J 76 doc	1	89.5%	93.4%	78.8%	80%	97.4%	100%	97.4%	98.7%
	2	85.5%	88.8%	78.8%	84.2%	96.1%	100%	94.7%	98.7%
	3	82.9%	89.5%	80.3%	82.9%	96.1%	100%	94.7%	96.1%
	4	82.9%	89.5%	78.9%	80.3%	96.1%	100%	94.7%	97.4%
	5	84.2%	89.5%	78.9%	80.3%	96.1%	100%	91.3%	91.3%
H05B (A) 79 doc	1	78.5%	94.9%	76.3%	87.5%	97.5%	100%	98.7%	100%
	2	78.5%	93.8%	83.8%	89.9%	98.7%	100%	100%	100%
	3	75.9%	92.4%	82.3%	89.9%	98.7%	100%	97.5%	100%
	4	78.5%	93.7%	81%	88.6%	98.7%	100%	98.7%	100%
	5	78.5%	94.9%	83.5%	88.6%	97.5%	100%	95%	96.3%
H02K 167 doc	1	71.3%	71.3%	68.3%	66.5%	89.8%	89.2%	84.4%	86.8%
	2	73.1%	73.7%	71.3%	73.1%	89.2%	91%	88.6%	89.2%
	3	74.9%	74.3%	71.3%	72.5%	91%	92.2%	90.4%	91%
	4	76.1%	74.9%	71.9%	71.3%	91%	91%	88.6%	89.2%
	5	74.3%	76%	71.3%	72.5%	91.6%	91.6%	88%	88.6%
A47C 75 doc	1	50.7%	73.3%	62.5%	70%	97.3%	98.7%	96%	94.7%
	2	54.7%	70%	68.8%	74.7%	97.3%	98.7%	94.7%	97.3%
	3	50.7%	74.7%	69.3%	76%	97.3%	98.7%	96%	94.7%
	4	49.3%	69.3%	72%	74.7%	96%	98.7%	96%	96%
	5	53.3%	69.3%	73.3%	78.7%	93.3%	98.7%	90%	90%

Método A - Método Rank e Cosseno // Método B – Método Relevância e Cosseno
Método C – Método Rank e Índice de Similaridade // Método D – Método Relevância e Índice de Similaridade

Tabela 21 B – Valores de Abrangência Obtidos para o Método 01 e para a Categoria H01F

Categoria	Resolução	Prognóstico de Topo				Três Prognósticos de Topo			
		Método A	Método B	Método C	Método D	Método A	Método B	Método C	Método D
H01F 134 doc.	1	44.8%	38.8%	39.6%	37.3%	67.2%	65.7%	67.2%	64.9%
	2	40.3%	40.3%	38.8%	36.6%	65.7%	65.7%	60.4%	63.4%
	3	36.6%	35.8%	40.3%	38.1%	67.2%	69.4%	64.2%	63.4%
	4	37.3	36.6%	36.6%	35.8%	64.9%	70.1%	63.4%	63.4%
	5	39.6%	37.3%	40.3%	38.1%	64.2%	66.4%	62.7%	64.2%

Método A - Método Rank e Cosseno // Método B – Método Relevância e Cosseno
Método C – Método Rank e Índice de Similaridade // Método D – Método Relevância e Índice de Similaridade

Para a categoria H01F, para Prognóstico de Topo, para as Resoluções 1 e 4, os melhores resultados foram obtidos para as técnicas de Rank/Cosseno; para a Resolução 2, os melhores resultados foram obtidos para as técnicas de Rank/Cosseno e Relevância/Cosseno; para as Resoluções 3 e 5, os melhores resultados foram obtidos para as técnicas de Rank/Índice de Similaridade. O melhor resultado foi obtido para a Resolução 1.

Para a categoria H01F, para Três Prognósticos de Topo, para a Resolução 1, os melhores resultados foram obtidos para as técnicas de Rank/Cosseno e Rank/Índice de Similaridade; para a Resolução 2 os melhores resultados foram obtidos para as técnicas de Rank/Cosseno e Relevância/Cosseno; e para as Resoluções 3 a 5, os melhores resultados foram obtidos para as técnicas de Relevância/Cosseno. O melhor resultado ocorreu para a Resolução 4.

Como mostrado não há uma técnica que melhor se adeque a todas as categorias. Portanto, optou-se para o melhor resultado, os resultados obtidos para a Resolução 1 e técnicas de Relevância/Cosseno.

6.2.2
Método 02

No Apêndice 7 estão apresentados, através das tabelas 88 e 89, os resultados obtidos para o algoritmo do Método 02. Na tabela 88, estão apresentados os resultados, usando-se a Medida de Desempenho de Prognóstico de Topo e na tabela 89 os resultados usando-se a Medida de Desempenho de Três Prognósticos de Topo. Nos resultados apresentados em ambas as tabelas é usada a Stemização StemerMétodo01 (radicalizador *StemmerPortuguese*). A Primeira Modalidade corresponde a pesagem dos termos dos documentos de teste através a técnica $TF' \times IDF$ descrita no item 2.3.5.1.6 e a Segunda Modalidade (a) corresponde a pesagem descrita no item 2.3.4.7.

Os resultados encontrados para a Medida de Desempenho de Prognóstico de Topo não foram satisfatórios, tendo sido obtido em média, uma Precisão de 56.37% (Primeira Modalidade) e 57.50% (Segunda Modalidade) e uma Abrangência de 51.66% (Primeira Modalidade) e 55.21% (Segunda Modalidade), levando-se em consideração o método do *macroaveraging.*

Os resultados encontrados para a Medida de Desempenho de Três Prognóstico de Topo foram melhores, tendo sido obtido na média uma Abrangência de 84.65% (Primeira modalidade) e 83% (Segunda Modalidade), levando-se em consideração o método de *macroaveraging*.

Para o experimento, a categoria para a qual o texto obteve o menor Grau de Similaridade foi a categoria sugerida para sua categorização, podendo haver mais de uma sugestão para a categoria a ser selecionada. Foi selecionado o menor Grau de Similaridade, pois o Grau de Igualdade é calculado através a média aritmética entre uma função *fuzzy* e uma função *não-fuzzy*. Os melhores resultados foram encontrados quando a função *não-fuzzy* teve um valor pequeno.

Outra constatação foi que houve a necessidade de se levar em consideração todos os termos para a representação da categoria. Uma trucagem com apenas os 50 (cinqüenta) primeiros termos com pesos maiores para a elaboração da representação da categoria não foi satisfatória.

6.2.3
Método 03

No Apêndice 8, nas tabelas 90 a 93, acham-se discriminados os resultados encontrados para o algoritmo do Método 03, usando-se o Método de Desempenho de Prognóstico de Topo e as Medidas de Similaridade de Cosseno, Jaccard, DICE e Índice de Similaridade (ABS) respectivamente. Nas tabelas 94 a 97 do Apêndice 8, acham-se discriminados os resultados encontrados para o algoritmo do Método 03, usando-se o Método de Desempenho de Três Prognósticos de Topo e as Medidas de Similaridade de Cosseno, Jaccard, DICE e Índice de Similaridade (ABS) respectivamente. Nos resultados apresentados em todas as tabelas (90 a 97) é usada a Stemização StemerMétodo01 (radicalizador *StemmerPortuguese*).

A Medida de F1 foi calculada através das medidas de Precisão e Abrangência obtidas pelo método de *macroaveraging*.

Com relação ao algoritmo do Método 03, para a categoria A47B (147 documentos) e com relação a quantidade de documentos categorizados corretamente, para a Medida de Desempenho de Prognóstico de Topo, houve uma variação entre 128 (cento e vinte e oito) e 129 (cento e vinte e nove) acertos para os Métodos de Similaridade de Cosseno, Jaccard, DICE e Índice de Similaridade (ABS), sendo a Abrangência obtida em média de 88%. Com relação a Medida de Desempenho de Três Prognósticos de Topo, para a mesma categoria, os resultados foram os mesmos que os obtidos para o Prognóstico de Topo, portanto não foram obtidos acertos para as Medidas de Desempenho de Dois e Três Prognósticos de Topo.

Para as categorias H05B(A) (79 documentos) e H02P (79 documentos) não houve acertos para a Medida de Desempenho de Prognóstico de Topo, para nenhum dos Métodos de Similaridade. Os acertos foram obtidos para as Medidas de Desempenho de Dois e Três Prognósticos de Topo. Obteve-se para a Medida de Desempenho de Três Prognósticos de Topo uma Abrangência, em média de 97% e 100% respectivamente para as categorias H05B(A) e H02P.

Para a Medida de Similaridade para as categorias H02G (167 documentos), H02K (167 documentos), A47C (75 documentos), H02M (77 documentos), H01J (76 documentos) e H02B (78 documentos), com relação a Medida de Desempenho de Prognóstico de Topo, os valores de Abrangências obtidos foram muito baixos, respectivamente em média, para o Método de Cosseno 7.8%,

10.2%, 16%, 22.1%, 28.9% e 37.2%. Os valores obtidos com relação a Abrangência e com relação as Medidas de Desempenho de Três Prognósticos de Topo foram para a Medida de Similaridade de Cosseno de aproximadamente de 83.2%, 80.2%, 97.3%, 100%, 97.4% e 93.6% respectivamente para H02G, H02K, A47C, H02M, H01J e H02B.

Para a categoria H05B(I) (78 documentos), para a Medida de Desempenho de Prognóstico de Topo, não houve acertos para nenhum dos Métodos de Similaridade. Os acertos ocorreram para as Medidas de Similaridade correspondentes a Dois e Três Prognósticos de Topo, em média de 97.4% (Abrangência).

Para a categoria H01F (134 documentos), com relação à quantidade de documentos categorizados corretamente para a Medida de Desempenho de Prognóstico de Topo, o índice de acerto (Abrangência) foi em média de 72% (Medida de Similaridade de Cosseno, Jaccard, DICE). Com relação a Medida de Desempenho de Três Prognósticos de Topo, os valores obtidos para a Abrangência foram de 80%, 78.4%, 78.4% e 81.3% para respectivamente as Medidas de Similaridade de Cosseno, Jaccard, DICE e Índice de Similaridade (ABS).

A Abrangência Média (*macroaveraging*) obtida para a Medida de Desempenho de Prognóstico de Topo foi muito baixa, ou seja, 25.67%; 25.09%; 25.09%; e 25.46% para os Métodos de Similaridade respectivamente de Cosseno, Jaccard, DICE e Índice de Similaridade (ABS). Para a Medida de Desempenho de Três Prognósticos de Topo, os resultados obtidos para as Abrangências Médias (*macroaveraging*) foram de 92.32%, 91.28%, 91.28%, 91.61% para respectivamente as Medidas Similaridade de Cosseno, Jaccard, DICE e Índice de Similaridade (ABS).

Do exposto, constatamos que o Método 03 não apresentou bons resultados para a Medida de Desempenho de Prognóstico de Topo, contudo para a Medida de Desempenho de Três Prognósticos de Topo, os valores obtidos para a Abrangência, ou seja, os índices de acerto, foram bem satisfatórios, para todas as Medidas de Similaridade, ou seja, Cosseno, Jaccard, DICE e Índice de Similaridade (ABS).

6.2.4
Método 04

Conforme descrito no item 4.2.4, para o algoritmo do Método 04, todos os documentos em uma dada categoria, formam um conjunto representado por vetores centróides, um para cada categoria e os pesos de cada termo de cada vetor centróide de uma dada categoria específica são calculados pela média aritmética de todos os pesos do dito termo dos vetores de todos os documentos da dita categoria. Portanto, um vetor protótipo, i.e., um centróide, foi construído para cada categoria.

Para uma Primeira Modalidade, testou-se o algoritmo com os documentos de treinamento, segundo o Método de Similaridade de Cosseno e Método de Stemização StemerMétodo01 (radicalizador *StemmerPortuguese*), i.e., com os mesmos documentos que formaram os centróides das categorias. A quantidade de documentos categorizados corretamente para Prognóstico de Topo foi de 3157 (três mil, cento e cinquenta e sete), de uma quantidade total de 3157 (três mil, cento e cinquenta e sete) documentos. Todos os documentos de treinamento foram categorizados corretamente. Nesse caso ocorreu o *overfitting*, i.e., o categorizador foi tão otimizado que características particulares dos dados de treinamento tornaram-se relevantes para as categorias. Os resultados encontram-se apresentados na Tabela 98 do Apêndice 9.

Para uma Segunda Modalidade, testou-se o algoritmo com os documentos de teste, segundo o Método de Similaridade de Cosseno e Método de Stemização StemerMétodo01 (radicalizador *StemmerPortuguese*). Para essa Modalidade, os documentos de teste são distintos dos documentos de treinamento. Utilizou-se para o cálculo dos pesos dos termos dos documentos de teste, o método usado no algoritmo do Método 01. Atualizou-se os vetores centróides com os pesos dos termos dos documentos de teste, para verificar se continuava a ocorrer *overfitting* com a introdução dos documentos de teste.

Para essa Segunda Modalidade foram selecionados para a etapa de teste, conforme item 3.5 (Segunda Modalidade da Divisão da Base de Dados em Treinamento e Teste), 706 (setessentos e seis) documentos sendo: 147 (cento e

quarenta e sete) documentos da categoria A47B; 10 (dez) da categoria H05B(A); 167 (cento e sessenta e sete) da categoria H02G; 31 (trinta e um) da categoria A47C; 10 (dez) da categoria H02P; 10 (dez) da categoria H02M; 10 (dez) da categoria H05B(I); 134 (cento e trinta e quatro) da categoria H01F; 167 (cento e sessenta e sete) da categoria H02K; 10 (dez) da categoria H02B; e 10 (dez) da categoria H01J.

Para essa Segunda Modalidade, para categorização dos documentos de teste, os valores obtidos para a quantidade de documentos categorizados corretamente, para o método de Desempenho de Prognóstico de Topo, foi de 595 (quinhentos e noventa e cinco), de uma quantidade total de 706 (setecentos e seis) documentos. A precisão média (*macroaveraging*) obtida foi de 86.60%, a abrangência média (*macroaveraging*) foi de 86.25% e a Medida de F1 calculada através da precisão e abrangência média (*macroaveraging*) foi 86.42%. Mais 103 (cento e três) documentos foram categorizados corretamente levando-se em consideração o Método de Desempenho de Dois e/ou Três Prognósticos de Topo. Somente 8 (oito) documentos dos 706 (setessentos e seis) documentos não foram categorizados corretamente fora do alcance dos Três Prognósticos de Topo, para os documentos de teste, todos distintos dos documentos de treinamento. O valor médio obtido para a abrangência (*macroaveraging*), para o Método de Desempenho de Três Prognósticos de Topo foi de 98.87% (698 acertos de 706). Os resultados acham-se discriminados nas Tabelas 99 e 100 do Apêndice 9.

Para uma Terceira Modalidade, testou-se o algoritmo com os documentos de teste, segundo o Método de Similaridade de Cosseno e Método de Stemização StemerMétodo01 (radicalizador *StemmerPortuguese*). Os documentos de teste também foram selecionados distintivamente dos documentos de treinamento, conforme item 3.5 (Segunda Modalidade da Divisão da Base de Dados em Treinamento e Teste). Utilizou-se para o cálculo dos valores dos pesos dos termos dos documentos de teste, o método usado no algoritmo do Método 01. Os vetores centróides foram construídos com os pesos dos termos dos documentos de treinamento.

Para essa Terceira Modalidade, a quantidade de documentos categorizados corretamente para o Método de Desempenho de Prognóstico de Topo foi 323 (trezentos e vinte e três) documentos para um total de 706 (setecentos e seis)

documentos. A quantidade de documentos categorizados corretamente, para o Método de Desempenho de Dois Prognósticos de Topo, foi de 458 (quatrocentos e cinqüenta e oito) documentos e para Três Prognósticos de Topo, a quantidade de documentos categorizados corretamente foi de 543 (quinhentos e quarenta e três) documentos. Para o Método de Desempenho de Prognóstico de Topo, a Precisão Média obtida foi de 33.14% (*macroaveraging*) e a Abrangência Média foi de 36.29% (*macroaveraging*). Para o Método de Desempenho de Dois Prognósticos de Topo, a Precisão Média obtida foi de 60.96% (*macroaveraging*) e para Três Prognósticos de Topo, a Precisão Média foi de 75.03% (*macroaveraging*). Os resultados acham-se discriminados na Tabela 101 do Apêndice 9.

Para essa Terceira Modalidade, onde se testou os documentos de teste, frente aos centróides, construídos a partir dos documentos de treinamento, em geral, para Prognóstico de Topo, os resultados não foram satisfatórios. Para esse caso, devido ter ocorrido *overfiting* com os dados de treinamento, o categorizador se ajustou de forma muito específica para o conjunto de treinamento, implicando em um baixo desempenho na categorização de documentos não conhecidos pelo categorizador, ou seja, documentos de teste.

Quando os documentos passarem a ser totalmente disponibilizados ao público, poder-se-à levar em consideração outros dados dos documentos, que não somente o Resumo, presumindo-se que os vetores centróides das categorias, retratarão com mais realidade cada categoria, pois de acordo com os resultados obtidos na Primeira Modalidade, o categorizador se ajusta de forma muito específica para o conjunto de treinamento. Poderá se testar o algoritmo também com mais documentos.

Como a quantidade de documentos de teste para o Algoritmo da Terceira Modalidade é diferente da considerada nos algoritmos dos outros Métodos Simulados, só pudemos comparar os resultados para as categorias A47B; H02G; H01F; e H02K, pois são as que contem a quantidade igual dos documentos de teste. O algoritmo do Método 01, levado em consideração para comparação usou Métodos de Similaridade de Cosseno e técnicas de Relevância e Resolução 1. A Tabela 102 do Apêndice 9 apresenta comparações entre os algoritmos dos Métodos 01 e 04 (Terceira Modalidade), para as categorias os quais as quantidades de documentos de teste são as mesmas para ambos os algoritmos. Os

resultados encontrados para o Algoritmo do Método 01 foram melhores do que os encontrados para o Algoritmo do Método 04.

6.2.5
Método 05

Para o algoritmo de categorização do Método 05, foi usada a técnica do Vizinho-Mais-Similar (Prognóstico de Topo), podendo também ser considerada como a categoria correta, qualquer um dos três primeiros acertos (Três Prognósticos de Topo).

Tanto para a Modalidade 05V1 quanto para a Modalidade 05, os resultados obtidos nas simulações foram satisfatórios com as seguintes exceções: para a categoria H02G, o qual não foi obtido nenhum acerto, nem para o Método de Desempenho de Prognóstico de Topo nem para o de Três Prognósticos de Topo; e para a categoria H01F, os quais os resultados não foram satisfatórios.

Para Prognóstico de Topo, para a modalidade 05V1, documentos de teste com classificação em H02G com 167 (cento e sessenta e sete) documentos, 88 (oitenta e oito) documentos foram classificados em H02B, 40 (quarenta) em H05B(A), 14 (quatorze) em A47B, 10 (dez) em H02K, 5 (cinco) em A47C, 6 (seis) em H02P, 1 (um) em H01F, 2 (dois) em H02M e 1 (um) em H01J. Para a Modalidade 05, 81 (oitenta e um) documentos foram classificados em H02B, 39 (trinta e nove) em H05B(A), 19 (dezenove) em A47B, 12 (doze) em H02K, 5 (cinco) em A47C, 5 (cinco) em H02P, 3 (três) em H01F, 2 (dois) em H02M e 1 (um) em H01J.

Para o Método de Desempenho de Prognóstico de Topo, para a Modalidade 05V1 (Precisão Média 67.60%; Abrangência Média 76.77% - *macroaveraging*), os resultados obtidos foram um pouco superiores aos obtidos para a Modalidade 05 (Precisão Média 66.02%; Abrangência Média 75.59% - *macroaveraging*). Para o Método de Desempenho de Três Prognósticos de Topo, para a Modalidade 05V1 (Abrangência Média 86.98% - *macroaveraging*), os resultados obtidos foram quase os mesmos dos resultados alcançados para a Modalidade 05 (Abrangência Média 86.89% - *macroaveraging*).

Para a Modalidade 05V1, Método de Desempenho de Prognóstico de Topo, as categorias que apresentaram as Abrangências mais elevadas foram: H05B(A) (98.73%); A47C (96%); H02M (94.81); H02P (92.41%). As categorias que obtiveram bons resultados, tanto para precisão quanto para abrangência foram: A47B (Precisão 89.3%; Abrangência 90.5%; Medida de F1 89.9%); A47C (Precisão 81.8%; Abrangência 96.0%; Medida de F1 88.33%); H05BI (Precisão 84.8%; Abrangência 85.9%; Medida de F1 85.35%); H01J (Precisão 88.6%; Abrangência 81.6%; Medida de F1 84.96%).

Para a modalidade 05V1, Método de Três Prognósticos de Topo, as categorias com suas respectivos valores de Abrangência, em ordem decrescente foram: H05BA (100%); H02P (100%); A47B (99.3%); A47C (98.7%); H02M (98.7%); H02B (98.7%); H05BI (98.7%); H02K (97.0%); H01J (94.7%); H01F (70.9%); H02G (0%).

No Apêndice 10, as Tabelas 103 a 106 apresentam os resultados para o algoritmo do Método 05, Modalidades 05 e 5V1.

Para pesquisas futuras, sugere-se que sejam realizadas outras simulações, através dos ajustes dos valores dos pesos, visando melhores resultados usando-se a técnica da distância H0B, tal como a expansão da vizinhança usando o Centro Perfeito. Nesse método expande-se a vizinhança exatamente por 1, em ambos os lados do domínio, mantendo o valor requerido precisamente no centro.

6.2.6
Método 06

Foi usada para o Método 06 a técnica da "Média Aritmética dos Pesos dos Termos dos Documentos da Categoria para o Vetor Centróide" descrito no item 4.4.1 e o Método de Stemização de *StemmerPortuguese* (StemerMétodo01).

Para uma Primeira Modalidade, testaram-se todos os documentos, sem separá-los em treinamento e teste, em nível de grupo, agrupados no nível mais alto. Para esse método, foi considerado que os documentos foram categorizados acertadamente em suas categorias em nível de subclasse.

Conforme mostrado na tabela 107 do Apêndice 11, os resultados encontrados foram satisfatórios. De 3861 (três mil, oitocentos e sessenta e um)

documentos, 3788 (três mil, setecentos e oitenta e oito) foram categorizados corretamente em nível de grupo de mais alto nível para Prognóstico de Topo, 3841 (três mil, oitocentos e quarenta e um) foram categorizados corretamente levando-se em consideração o Método de Desempenho de Dois Prognósticos de Topo e 3855 (três mil, oitocentos e cinqüenta e cinco) levando-se em consideração o Método de Desempenho de Três Prognósticos de Topo.

Para essa Primeira Modalidade, para Prognóstico de Topo, 73 (setenta e três) documentos de 3861 (três mil, oitocentos e sessenta e um) tiveram sua categorização incorreta; para o método de Desempenho de Dois Prognósticos de Topo, 20 (vinte) documentos tiveram sua categorização incorreta; e para Três Prognósticos de Topo, somente 6 (seis) documentos não conseguiram ser categorizados corretamente.

Exemplificamos com a categoria H02K, que com 500 (quinhentos) documentos, 483 (quatrocentos e oitenta e três) documentos foram categorizados corretamente usando-se o Método de Desempenho de Prognóstico de Topo, 496 (quatrocentos e noventa e seis) documentos foram categorizados corretamente para o Método de Desempenho de Dois Prognósticos de Topo e 499 (quatrocentos e noventa e nove) documentos categorizados corretamente no Método de Desempenho de Três Prognósticos de Topo. Somente 1 (um) documento não conseguiu ser categorizado corretamente levando-se em consideração o método de desempenho de Três Prognósticos de Topo.

Para essa Primeira Modalidade, a Abrangência Média (*macroaveraging*) obtida para Prognóstico de Topo foi de 98.06%. Os resultados obtidos em ordem crescente para Abrangência (segundo o Método de Desempenho de Prognóstico de Topo) foram: 95.07% (H02G); 96.60% (H01F); 96.94% (H05BI); 97.09% (H02K); 97.92% (H01J); 98.14% (A47C); 99.03% (A47B); 99.11% (H02P); 99.43% (H05BA); 99.64% (H02B); 99.67% (H02M). Os resultados obtidos em ordem crescente, para Abrangência (segundo o Método de Desempenho de Dois Prognósticos de Topo) foram: 96.63% (H02G); 99.19% (H01F); 99.24% (H05BI); 99.24% (H02K); 99.73% (A47C); 99.74% (H05BA); 99.84% (H02M); 99.92% (H02P); 99.96% (A47B); 100% (H01J); 100% (H02B). Os resultados obtidos em ordem crescente para Abrangência (segundo Método de Desempenho de Três Prognósticos de Topo) foram: 97.92% (H02G); 99.66% (H01F); 99.72% (H02K);

99.84% (H02M); 99.96% (A47B); 100% (H05BI); 100% (A47C); 100% (H05BA); 100% (H02P); 100% (H01J); 100% (H02B).

Para a Primeira Modalidade, os resultados obtidos para Precisão, segundo Medida de Desempenho de Prognóstico de Topo, em ordem crescente foram: 97.87% (H02G); 98.58% (A47B); 98.74% (H05BI); 98.76% (A47C); 98.88% (H01F); 98.88% (H02K); 99.14% (H02P); 99.30% (H05BA); 99.35% (H02M); 99.88% (H01J); e 99.93% (H02B).

Para a Primeira Modalidade, a média dos valores para a Medida de F1 (em inglês *F-Measure*), segundo a medida de Desempenho de Prognóstico de Topo, obtido através a Precisão (*macroaveraging*) e Abrangência (*macroaveraging*), foi de 98.54%. Os resultados obtidos para Medida de F1, segundo Prognóstico de Topo, em ordem crescente foram: 96.45% (H02G); 97.73% (H01F); 97.83% (H05BI); 97.98% (H02K); 98.45% (A47C); 98.80% (A47B); 98.89% (H01J); 99.12% (H02P); 99.36% (H05BA); 99.51% (H02M); 99.78% (H02B).

Tabelas 108A e 108B do Apêndice 11 apresentam os resultados obtidos para o Método 06, Primeira Modalidade, para os Métodos de Desempenho de Topo, Dois e Três Prognósticos, para a categoria A47B em nível de grupo. Tabelas 109, 110, 111, 112, 113, 114, 115 e 116 do Apêndice 11 apresentam os resultados para o Método 06, Primeira Modalidade, para as seguintes categorias, em nível de grupo, respectivamente: H05B(A); H02G; A47C; H02P; H02M; H05B(I); H02B; e H01F. Tabelas 117A e 117B apresentam os resultados obtidos para o Método 06, Primeira Modalidade, para a categoria H02K em nível de grupo e Tabela 118 para a categoria H01J também em nível de grupo.

Para uma Segunda Modalidade, testaram-se os documentos de teste (distintos dos de treinamento), em nível de grupo agrupados no nível mais alto. Para esse método foi considerado que os documentos foram categorizados acertadamente em suas categorias em nível de subclasse. Conforme mostrado na tabela 119 do Apêndice 11, dos 706 (setessentos e seis) documentos, 686 (seissentos e oitenta e seis) documentos foram categorizados corretamente em nível de grupo de mais alto nível, para Método de Desempenho de Prognóstico de Topo; 699 (seissentos e noventa e nove) levando-se em consideração o Método de

Desempenho de Dois Prognósticos; e 702 (setessentos e dois) foram categorizados corretamente levando-se em consideração Três Prognósticos de Topo.

Para a Segunda Modalidade, para o Método de Desempenho de Prognóstico de Topo, 20 (vinte) documentos dos 706 (setessentos e seis) tiveram sua categorização incorreta; para Dois Prognósticos de Topo, 7 (sete) documentos tiveram sua categorização incorreta; e para Três Prognósticos de Topo somente 4 (quatro) documentos não conseguiram ser categorizados corretamente.

Para a Segunda Modalidade do algoritmo do Método 06, a Abrangência Média (*macroaveraging*) obtida para Prognóstico de Topo foi de 98.81%. Na Tabela 119 do Apêndice 11 é apresentado um resumo dos resultados obtidos para as diversas categorias, usando-se para teste os documentos distintos dos de treinamento, para os Métodos de Topo, Dois e Três Prognósticos, usando-se o Método de Stemização de *StemmerPortuguese*.

Para essa Segunda Modalidade, tabelas 120A e 120B do Apêndice 11, apresentam os resultados obtidos nas simulações para a categoria H02K; tabelas 121A e 121B para a categoria A47B; tabelas 122 a 130 para as categorias H05B(A), H02G, A47C, H02P, H02M, H05B(I), H02B, H01F e H01J respectivamente.

Exemplificamos com a categoria H02K, que com 167 (cento e sessenta e sete) documentos, 160 (cento e sessenta) documentos foram categorizados corretamente, levando-se em consideração o Método de Desempenho de Prognóstico de Topo; 163 (cento e sessenta e três) documentos categorizados corretamente, levando-se em consideração Dois Prognósticos de Topo; e 166 (cento e sessenta e seis) documentos categorizados corretamente levando-se em consideração Três Prognósticos de Topo.

7.0
Conclusão

A proposição desse estudo foi a definição de vários modelos direcionados a categorização de textos de pedidos de patente de depositantes brasileiros, no idioma português. Para esse ambiente foi proposto um comitê composto de 6 (seis) modelos, onde foram usadas várias técnicas. A base de dados consistiu de 1157 (hum mil, cento e cinquenta e sete) resumos de pedidos de patentes depositados no INPI por depositantes nacionais, distribuídos em várias categorias. A seção H foi escolhida por tratar-se de Eletricidade e a seção A por referir-se as Necessidades Humanas. Foram selecionadas as seguintes classificações: H02G (cabos e linhas elétricas); H02P (controle ou regulação); H02M (conversão de energia); H02B (painéis); H01F (magnetos e indutâncias); H01J (tubos e lâmpadas de descarga); H02K (máquinas elétricas); H05B (aquecimento e iluminação); A47C (cadeiras); A47B (móveis, artigos ou aparelhos domésticos).

Os textos que constituem os resumos dos pedidos de patentes de depositantes brasileiros fazem parte de uma base não estruturada e devido a esses textos conterem erros ortográficos e de digitação, devido a maioria das vezes por terem sido escritos por pessoas não conhecedoras do assunto ou estarem com tradução mal feita, depois de coletados os dados, foi realizada uma limpeza dos textos com a correção dos erros ortográficos.

Em seguida, foi definida uma lista de *stopwords* (*stoplist*), a qual foi retirada dos textos. Para a definição da lista de *stopwords* foi levada tambem em consideração, palavras que não carregam nenhuma informação de maior relevância para a base de dados considerada, isto é, termos com pouca representatividade para os documentos considerados, principalmente termos sem representatividade para a categorização de um pedido de patente. A lista constou de 1337 (hum mil, trezentos e trinta e sete) termos que foram considerados sem valor para a base de dados considerada.

Em seguida, foi realizado um tratamento com os termos compostos, termos que quando aparecem juntos tem significados diferentes do que tem quando aparecem sozinhos. Por economia processual, optou-se pela junção de alguns termos compostos, relevantes para a base de dados considerada, tais como: foto

diodos foi substituído por fotodiodos; corrente alternada por correntealternada, etc., principalmente termos direcionados para as técnicas usadas nos pedidos de patente a serem categorizados.

Referente aos métodos de stemização e levando-se em consideração a quantidade de termos distintos por categoria, a redução dos termos usando-se o radicalizador de *StemmerPortuguese* foi de 42.19% e para o radicalizador de Porter foi de 31.99%. Portanto, optou-se para a etapa de processamento da categorização, nesse estudo, o uso do radicalizador de *StemmerPortuguese.*

Dentre os vários modelos propostos para a etapa de processamento da categorização de textos, destacamos o desenvolvido para o Método 01, ou seja, o *k-Nearest-Neighbor* (*k*-NN), modelo também usado para o ambiente de patentes no idioma inglês. Para esse modelo, também foram usadas as técnicas: de Similaridade do Cosseno e do Índice de Similaridade; Medida de Desempenho de Prognóstico de Topo e Três Prognósticos de Topo; Método de Predição de Rank e de Relevância. As técnicas usadas nesse modelo foram baseadas: no modelo do algoritmo de Baoli (2003), com exceção da técnica de similaridade denominada Índice de Similaridade; e em algumas técnicas dos modelos dos algoritmos de Hadi et al. (2007, 2008) e Moraes & Lima (2007).

Para os outros modelos, foram escolhidos métodos que não os tradicionais, usados em ambiente de patentes em outros idiomas que não o português. Para quatro modelos, optou-se por algoritmos, que na etapa de treinamento, as categorias são representadas por vetores centróides.

Para um dos modelos, foi explorada a técnica do *High Order Bit* com o algoritmo *k*-NN, sendo o *k* todos os algoritmos de treinamento.

Para o primeiro modelo desenvolvido (método 01), para a técnica de Similaridade do Cosseno, método de predição de Relevância, método de Stemização *StemmerPortuguese* e para *k* igual a 13, foi alcançado um resultado para a Medida de F1 de 78.52% e para a Abrangência Média o valor de 79.83% (*macroaveraging*). Não houve muita diferença entre os resultados obtidos para a Medida de F1, obtido a partir dos valores de Precisão (*macroaveraging*) e Abrangência (*macroaveraging*) para *k* igual a 17 (77.59%), para *k* igual a 23 (77.74%), para *k* igual a 25 (77.69%) e para *k* igual a 31 (77.85%). Também para

o primeiro modelo, para o Método de Desempenho de Três Prognósticos de Topo, para a técnica de Stemização *StemmerPortuguese*, os melhores resultados obtidos foram para a técnica de Similaridade do Cosseno, Método de Predição de Relevância, *k* igual a 25, onde foi obtido para a Abrangência Média (*macroaveraging*) o valor de 94.54%. Também não houve muita diferença entre os resultados obtidos para a Abrangência Média (*macroaveraging*) para *k* igual a 13 (93.44%), para *k* igual a 17 (93.58%), para *k* igual a 23 (94.30%) e para *k* igual a 31 (94.35%).

Para primeiro modelo desenvolvido ou Método 01, para os valores obtidos através a técnica de *macroaveraging*, dentre os métodos de Similaridade simulados, o que obteve o melhor resultado foi o do Cosseno e para o método de predição o melhor resultado foi o de Relevância.

Para o primeiro modelo, referente às técnicas de Prognóstico de Topo, Relevância, Cosseno e Resolução 1, as categorias que apresentaram os melhores resultados, para a Medida de Abrangência, em ordem decrescente foram: H05B(A) (94.9%); H02P (93.7%); H02M (93.5%); H01J (93.4%); A47B (91.8%); H05B(I) (88.5%); H02B (82.1%); A47C (73.3%); H02K (71.3%); H02G (56.9%); e H01F (38.8%).

Referente ao primeiro modelo, para a categoria H02K, para Prognóstico de Topo e Resolução 1, os melhores resultados foram os obtidos para as técnicas de Rank/Cosseno e Relevância/Cosseno. Para as Resoluções 2 e 5, os melhores resultados foram os obtidos para as técnicas de Relevância/Cosseno e para as Resoluções 3 e 4, os melhores resultados foram os obtidos para as técnicas de Rank/Cosseno. O melhor resultado foi obtido para as Resoluções 4 e 5.

Também referente ao primeiro modelo, para a categoria H01F, para Prognóstico de Topo, para as Resoluções 1 e 4, os melhores resultados foram os obtidos para as técnicas de Rank/Cosseno, para a Resolução 2, o melhor resultado foi o obtido para as técnicas de Rank/Cosseno e Relevância/Cosseno e para as Resoluções 3 e 5, os melhores resultados foram os obtidos para as técnicas de Rank/Índice de Similaridade. O melhor resultado ocorreu para a Resolução 1.

Referente ao primeiro modelo ou Método 01, conforme mostrado, não há uma técnica que melhor se adeque a todas as categorias. Portanto, optou-se como

o melhor resultado, o obtido na Resolução 1 para as técnicas de Relevância e Cosseno, que foram as que conseguiram melhores resultados no cômputo geral, para valores de *macroaveraging*.

Para o Método 02, para a técnica de Stemização *StemmerPortuguese*, não foram obtidos bons resultados para o Método de Desempenho de Prognóstico de Topo, contudo para Três Prognósticos de Topo, o resultado para a Abrangência Média (*macroaveraging*) foi satisfatório, ou seja, 84.65%.

Para o Método 03, a Abrangência Média (*macroaveraging*) obtida para a Medida de Desempenho de Prognóstico de Topo foi muito baixa, ou seja, 25.67%; 25.09%; 25.09%; e 25.46% para os Métodos de Similaridade respectivamente de Cosseno, Jaccard, DICE e Índice de Similaridade (ABS). Para a Medida de Desempenho de Três Prognósticos de Topo, os resultados obtidos para as Abrangências Médias (*macroaveraging*) foram de 92.32%, 91.28%, 91.28%, 91.61% para respectivamente as Medidas Similaridade de Cosseno, Jaccard, DICE e Índice de Similaridade (ABS).

Com relação ao Método 03, para a categoria A47B (147 documentos) e com relação à quantidade de documentos categorizados corretamente, para a Medida de Desempenho de Prognóstico de Topo, houve uma variação entre 128 (cento e vinte e oito) e 129 (cento e vinte e nove) acertos para os Métodos de Similaridade de Cosseno, Jaccard, DICE e Índice de Similaridade (ABS), sendo a Abrangência obtida em média de 88%. Com relação à Medida de Desempenho de Três Prognósticos de Topo, para a mesma categoria, os resultados foram os mesmos que os obtidos para o Prognóstico de Topo, portanto não foram obtidos acertos para as Medidas de Desempenho de Dois e Três Prognósticos de Topo.

Ainda com relação ao Método 03, para as categorias H05B(A) (79 documentos) e H02P (79 documentos) não houve acertos para a Medida de Desempenho de Prognóstico de Topo, para nenhum dos Métodos de Similaridade. Os acertos foram obtidos para as Medidas de Desempenho de Dois e Três Prognósticos de Topo. Obteve-se para a Medida de Desempenho de Três Prognósticos de Topo uma Abrangência, em média de 97% e 100% respectivamente para as categorias H05B(A) e H02P.

Para a categoria H05B(I) (78 documentos), para a Medida de Desempenho de Prognóstico de Topo, não houve acertos para nenhum dos Métodos de Similaridade. Os acertos ocorreram para as Medidas de Similaridade correspondentes a Dois e Três Prognósticos de Topo, em média de 97.4% (Abrangência).

Para a categoria H01F (134 documentos), com relação à quantidade de documentos categorizados corretamente para a Medida de Desempenho de Prognóstico de Topo, o índice de acerto (Abrangência) foi em média de 72% (Medida de Similaridade de Cosseno, Jaccard, DICE). Com relação à Medida de Desempenho de Três Prognósticos de Topo, os valores obtidos para a Abrangência foram de 80%, 78.4%, 78.4% e 81.3% para respectivamente as Medidas de Similaridade de Cosseno, Jaccard, DICE e Índice de Similaridade (ABS).

Do exposto, constatamos que o Método 03 não apresentou bons resultados para a Medida de Desempenho de Prognóstico de Topo, contudo para a Medida de Desempenho de Três Prognósticos de Topo, os valores obtidos para a Abrangência, ou seja, os índices de acerto, foram bem satisfatórios, para todas as Medidas de Similaridade, ou seja, Cosseno, Jaccard, DICE e Índice de Similaridade (ABS). A Medida de F1 foi calculada através das medidas de Precisão e Abrangência obtidas pelo método de *macroaveraging*.

Para o Método 04, para a Terceira Modalidade, onde foram simulados somente os documentos de teste, para a técnica de Stemização de *StemmerPortuguese* e Método de Similaridade do Cosseno, os resultados para o Método de Desempenho de Prognóstico de Topo não foram muito bons (precisão média 33.14%; abrangência média 36.29% - *macroaveraging*), contudo para o Método de Desempenho de Três Prognósticos de Topo, os resultados obtidos foram um pouco melhores, ou seja, 75.03% para a abrangência média (*macroaveraging*). Vale salientar que para o Método 04, Terceira Modalidade, os documentos selecionados para teste foram totalmente distintos dos de treinamento. Para as categorias H05B (H05BA e H05BI), H02P, H02M, H02B e H01J, para a fase de teste, foram usados somente 10 (dez) documentos para cada categoria e para A47C foram usados 31 (trinta e um) documentos. Para as categorias H02G, H01F, H02K e A47B, a quantidade de documentos usados para

teste foi idêntica aos usados para os métodos 01, 02, 03 e 05. Quando se testou o algoritmo do Método 04 com os centróides das categorias oriundas dos documentos testados, os resultados foram muito bons, portanto, isso é um indício que é necessário uma maior precisão no retratamento dos vetores centroides visando à obtenção de melhores resultados.

Para o Método 04, quando se testou o algoritmo com os documentos de treinamento, i.e., com os mesmos documentos que formaram os centróides das categorias, usando-se o Método de Similaridade do Cosseno e técnica de Stemização *StemmerPortuguese* ocorreu um *overfitting*, com acerto de 100%.

Para o Método 05, os resultados para o Método de Desempenho de Prognóstico de Topo e técnica de stemização *StemmerPortuguese*, foram considerados médios, ou seja, para a medida de F1 obteve-se o valor de 71.89% (Abrangência – 76.77% *macroaveraging*). Para o Método de Desempenho de Três Prognósticos de Topo, o valor obtido para a Abrangência (*macroaveraging*) foi de 86.89%, considerado satisfatório.

Para o Método 06, assumindo-se de que os documentos de teste foram categorizados corretamente em nível de subclasse e testando-se o algoritmo com as classificações em nível de grupo mais alto, os resultados foram ótimos, tanto para Prognóstico de Topo, quanto para Dois Prognósticos de Topo e Três Prognósticos de Topo.

Segundo Fall & Benzineb (2002), o objetivo mínimo do categorizador automático para categorização de pedidos de patente deve ser capaz de predizer com exatidão pelo menos 80-90% das categorias categorizadas em nível de subclasse, de acordo com 3 (três) [Método de Desempenho de Três Prognósticos de Topo] a 4 (quatro) sugestões, sendo que a decisão final deverá ser feita por um especialista.

Portanto, segundo Fall & Benzineb (2002), o método 01 (*k*-NN), tanto para *k* igual a 13, 17, 23, 25 ou 31 são bons categorizadores, tanto para o método de similaridade de Cosseno quanto para o Método de Índice de Similaridade e tanto para o método de predição de *Rank* quanto para o de Relevância (segundo Método de Desempenho de Três Prognósticos de Topo).

Ainda segundo Fall & Benzineb (2002), os métodos 02 e 05 simulados também são bons categorizadores e o método 03 tanto para o método de similaridade de Cosseno quanto para os métodos de Jaccard, DICE e Índice de Similaridade (ABS) são também bons categorizadores (segundo Método de Desempenho de Três Prognósticos de Topo).

Futuramente, quando os documentos passarem a ser totalmente disponibilizados ao público, poder-se-á levar em consideração outros dados dos documentos, que não somente o Resumo, presumindo-se que para o algoritmo do Método 04, os vetores centróides das categorias, retratarão com mais realidade cada categoria. Também poderia se testar o algoritmo com mais documentos.

Também como sugestões para trabalhos futuros podem destacar:

- os termos que compõem os títulos dos documentos poderiam ter pesos três vezes maiores (pesos de voto) que os termos do resumo;

- inclusão de termos específicos (direcionados aos existentes nas classificações da IPC) nos centroides dos algoritmos;

- particionamento dos documentos em N segmentos de iguais tamanhos. O procedimento seria executado N vezes. O teste seria feito em 1 (uma) partição de cada vez e o treinamento em N-1 partições restantes. O conjunto de teste seria diferente do conjunto de treinamento. O resultado final seria a média dos resultados de cada tentativa;

- poderia ter sido usada uma ontologia.

Para o algoritmo 05, para a predição dos resultados, poderiam ser selecionados somente os *k*-Vizinhos-Mais-Próximos de cada categoria (valor dependente da quantidade de documentos). Também relacionado com o algoritmo 05, outros pesos de voto poderiam ser selecionados para serem testados com o algoritmo.

Na tabela 22 acham-se transcritos as médias dos resultados (*macroaveraging*) encontrados nas simulações para o algoritmo do Método 01.

Tabela 22 - Resultados Encontrados nas Simulações para o Método 01						
Método	Resolução	Desempenho	Similaridade	Precisão	Abrangência	Medida F
Método 1 *k*-Nearest Neighbor	Resolução1 k=13	Prognóstico de Topo	RankCos	0.6994	0.6887	0.6940
			RelevCos	0.7725	0.7983	0.7852
			RankABS	0.6697	0.6588	0.6642
			RelevABS	0.7157	0.7141	0.7149
		3 Prognós-ticos de Topo	RankCos	-	0.9134	-
			RelevCos	-	0.9344	-
			RankABS	-	0.9072	-
			RelevABS	-	0.9263	-
	Resolução 2 k=17	Prognóstico de Topo	RankCos	0.6957	0.6868	0.6912
			RelevCos	0.7750	0.7769	0.7759
			RankABS	0.7033	0,6902	0.6967
			RelevABS	0.7268	0.7340	0.7304
		3 Prognós-ticos de Topo	RankCos	-	0.9019	-
			RelevCos	-	0.9358	-
			RankABS	-	0.9064	-
			RelevABS	-	0.9292	-
	Resolução 3 k=23	Prognós-tico de Topo	RankCos	0.7020	0.6878	0.6949
			RelevCos	0.7699	0.7850	0.7774
			RankABS	0.7057	0.6962	0.7009
			RelevABS	0.7329	0.7338	0.7333
		3 Prognós-ticos de Topo	RankCos	-	0.9140	-
			RelevCos	-	0.9430	-
			RankABS	-	0.9081	-
			RelevABS	-	0.9167	-
	Resolução 4 k=25	Prognóstico de Topo	RankCos	0.7116	0.6912	0.7013
			RelevCos	0.7718	0.7820	0.7769
			RankABS	0.6978	0.6923	0.6950
			RelevABS	0.7266	0.7265	0.7265
		3 Prognós-ticos de Topo	RankCos	-	0.9242	-
			RelevCos	-	0.9454	-
			RankABS	-	0.9098	-
			RelevABS	-	0.9203	-
	Resolução 5 k=31	Prognóstico de Topo	RankCos	0.7122	0.6877	0.6997
			RelevCos	0.7742	0.7828	0.7785
			RankABS	0.7184	0.7123	0.7153
			RelevABS	0.7371	0.7365	0.7368
		3 Prognós-ticos de Topo	RankCos	-	0.9215	-
			RelevCos	-	0.9435	-
			RankABS	-	0.8862	-
			RelevABS	-	0.8945	-

Tabela 23 - Resultados Encontrados nas Simulações para os Métodos 02, 03, 04, 05 e 06

Método	Modalidade	Desempenho	Similaridade	Precisão	Abrangência	Medida de F
Método 02	Modalidade1	Prognós-tico de Topo		0.5637	0.5166	
	Modalidade2			0.5750	0.5521	
	Modalidade1	3 Prognós ticos Topo		-	0.8465	-
	Modalidade2			-	0.83	-
Método 03		Prognóstico Topo	Cos	0.2028	0.2567	0.2266
			Jaccard	0.1954	0.2509	0.2197
			DICE	0.1954	0.2509	0.2197
			ABS	0.2	0.2546	0.224
		3 Prognós ticos Topo	Cos	-	0.9232	-
			Jaccard	-	0.9128	-
			DICE	-	0.9128	-
			ABS	-	0.9161	-
Método 04	Modalidade 1	Progn.Topo	Cos	1.0	1.0	1.0
	Modalidade 2	Progn.Topo	Cos	0.8660	0.8625	0.8642
		3 Prg Topo	Cos	-	0.9887	-
	Modalidade 3	Progn.Topo	Cos	0.3314	0.3629	-
		2 Prg.Topo	Cos	-	0.6096	-
		3 Prg Topo	Cos	-	0.7503	-
Método 05	Modalidade05	Prognós-tico de Topo	HOB	0.6602	0.7559	0.7048
	Modalidade 05V1		HOB	0.6760	0.7677	0.7189
	Modalidade05	3 Prognós-ticos de Topo	HOB	-	0.8689	-
	Modalidade 05V1		HOB	-	0.8698	-
Método 06	Modalida-de 1	Prognóstico de Topo	Cos	0.9903	0.9806	0.9854
		2 Prognósti-cos Topo	Cos	-	0.9941	-
		3 Prognósti-cos Topo	Cos	-	0.9974	-
Método 06	Modalida-de 2	Prognóstico de Topo	Cosseno	0.9946	0.9881	-
		2 Prognósti-cos Topo	Cosseno	-	0.9929	-
		3 Prognósti-cos Topo	Cosseno	-	0.9941	-

Na tabela 23 acham-se transcritos as médias (*macroaveraging*) dos resultados encontrados nas simulações para os algoritmos dos Métodos 02, 03, 04, 05 e 06.

8.0
Referências bibliográficas

ADAMS, Stephen. **Using the International Patent Classification in an Online Environment**. Patent Information Users' Group (PIUG), Annual Meeting in Washington DC, USA 1999. World Patent Information 22, 2000, pp. 291-300.

AGRAWAL, R; Imielinski, T; Swami, A. **Mining Association Rules Between Sets of Items in Large Databases**. Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington D. C, May 26-28, 1993, pp. 207-216.

AHA, D. **Tolerating Noisy, Irrelevant and Novel Atributes in Instance-Based Learning Algorithms**. International Journal of Man-Machine Studies, 1992, v. 3, n. 2, pp. 267-287.

ALVARES, Reinaldo Viana. **Investigação do Processo de Stemming na Língua Portuguesa**. Dissertação do Mestrado em Programação em Computação da Universidade Federal Fluminense, março 2005.

ARANHA, Christian N.; Passos, Emmanuel Piceses Lopes. **A Tecnologia de Mineração de Textos**. RESI - Revista Eletrônica de Sistemas de Informação, nº 2, Lab. ICA Elétrica PUC – Rio, 2006.

ARANHA, Christian N. **Uma Abordagem de Pré-Processamento Automático para Mineração de Textos em Português: sob o Enfoque da Inteligência Computacional**. Tese de Doutorado, Programa de Pós-Graduação da Engenharia Elétrica do Departamento de Engenharia Elétrica da PUC-Rio, março 2007.

[1]BAOLI, Li; [2]Shiwen, Yu; Qin Lu. **An Improved k-Nearest Neighbor Algorithm for Text Categorization**. [1]Institute of Computacional Linguistics, Department of Computer Science and Technology, Peking University, Beijing, P. R. China; [2]Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong; Proceedings of the 20th International Conference on Computer Processing of Oriental Languages, Shenyang, China, 2003.

BARBOSA, Denis. **Uma Introdução à Propriedade Intelectual.** 2ª Edição Revista e Atualizada. Rio de Janeiro, 2003.

[1]BAX, Marcello Peixoto; [2]Souza, Renato Rocha. **Uma Proposta de Uso de Agentes e Mapas Conceituais para Representação de Conhecimentos Altamente Contextualizados.** [1]Universidade Federal de Minas Gerais, Departamento de Ciência da Informação; [2]Pontifícia Universidade Católica de Minas Gerais – Data PUC; 4º Simpósio Internacional de Gestão do Conhecimento/Gestão de Documentos – ISKM/DM, Curitiba, 2001.

[1]BORSATO, Bruno; Merschmann Luiz; [2]Plastino Alexandre. ***k*-NN: Estimando o Valor do Parâmetro *k***. [1]Instituto de Computação - Universidade Federal Fluminense – Niterói - Rio de Janeiro;

[2]Departamento de Ciências Exatas e Aplicadas – Universidade Federal de Ouro Preto João Monlevade - Minas Gerais. WAAMD 2007, III Worshop em Algoritmos e Aplicações de Mineração de Dados, 2007.

BUFFET, Pierre. **IPC – A Tool for Classifying Patent Documents Rather Than Concepts**. Executive Vice-President Questel Orbit Intellectual Property Group S. A. Paris, France. IPC Workshop, Geneve, February 2010.

CAI, Lijuan; Hofmann, Thomas. **Hierarchical Document Categorization with Support Vector Machine**. Brown University, Providence, R. I. In Proc. of the 13th ACM International Conference on Information and Knowledge Management (CIKM'04), Washington D.C.: ACM Press, New York, 2004, pp. 78-87.

CAMARGO, Yuri Barwick Lannes. **Abordagem Lingüística na Classificação Automática de Texto em Português**. Dissertação de mestrado do Programa de Pós-Graduação de Engenharia Elétrica da UFRJ, junho de 2007.

CARRILHO, J. **Desenvolvimento de uma Metodologia para Mineração de Textos**. Dissertação de Mestrado, Departamento de Engenharia Elétrica, PUC-Rio, 2007.

CARVALHO, Gustavo; Sayão, Miriam; Gatti, Maíra. **Técnicas de PLN na Análise de Domínio em SMAs Abertos**. Laboratório de Engenharia de Software, PUC, 2005.

CHAKRABARTI, S.; Dom, B. E. T.; Indyk P. **Enhanced Hypertext Categorization Using Hiperlinks**. Proceedings of SIGMOD-98, ACM International Conference on Management of Data, eds. L. M. Haas, 1998; A. Tiwary, ACM Press, New York, US: Seattle, US, pp. 307-318.

CHAKRABARTI, S.; Dom, B. E.; Agrawal, R.; Raghavan, P. **Scalable Feature Selection, Classification and Signature Generation for Organizing Large Text Databases into Hierarchical Topic Taxonomies**. Journal of Very Large Data Bases, 1998, 7(3), pp. 163-178.

CHOW, Yuan Kai; Perumal, Nagendran. **A Document Categorization System**. Technology Park of Malaysia, Kuala, MY; Pedido de Patente WO 2009/145605 A2; PCT/MY2009/000065.

COELHO, Alexandre Ramos. **Stemming para a Língua Portuguesa: estudo, análise, melhoria do algoritmo RSLP**. Trabalho de Graduação da Universidade Federal do Rio Grande do Sul, Instituto de Informática, Curso de Ciência da Computação, Porto Alegre, junho de 2007.

Delegation of Japan. OWAKE Systems – **Primary Automatic Classification**. Section 59 of WIPO Report IPC/CE/29/11, on the 29th Session of the Committee of Experts of the IPC Union, 13-17 March 2000.

DIAS, Maria Abadia Laceda; Malheiros, Marcelo de Gomensoso. **Estudo de Técnicas de Radicalização para a Língua Portuguesa**. Centro Universitário UNIVATES, Lajeado, Rio Grande do Sul, Brasil, 2004.

DIAS, Maria Abadia Lacerda; Malheiros, Marcelo de Gomensoro. **Extração Automática de Palavras-Chave de Textos da Língua Portuguesa**. Centro Universitário UNIVATES, Lajeado – RS, 2005.

DING, Qin; Khan Maleq; Roy Amalendu; Perrizo William. **The P-Tree Algebra**. Computer Science Department, North Dakota State University, Fargo, USA. Proceedings of ACM Symposium on Applied Computing (SAC'02), Madrid, Spain, March 2002, pp. 426-431.

DUDA, R. O.; Hart, P. E. **Pattern Classification and Scene Analysis**. John Wiley & Sons, New York., 1973.

DUMAIS, S. T.; Platt, J.; Heckerman, D.; Sahami, M. **Inductive Learning Algorithms and Representations for Text Categorization**. Proceedings of CIKM-98, 7th ACM International Conference on Information and Knowledge Management, eds. G. Gardarin, J. C. French, N. Pissinou, K. Makki, L. Bouganim, ACM Press, New York, US: Bethesda, US, 1998, pp. 148-155.

DUMAIS, S. T.; Chen, H. **Hierarchical Classification of Web Content**. Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval, eds. N. J. Belkin, P. Ingwersen, M. K. Leong. ACM Press, New York, US: Athens, GR, 2000, pp. 256-263.

FALL, C. J.; Benzineb, K. **Literature Survey: Issues to be Considered in the Automatic Classification of Patents**. World Intellectual Property Organization, Geneva, 2002.

[1]FALL, C. J.; [2]Törcsvári, A.; [3]Benzineb K.; [4]Karetka G. **Automated Categorization in the International Patent Classification**. [1]ELCA Informatique SA, Lausanne, Switzerland; [2]Arcanum Development, Budapest, Hungary; [3]Metaread SA, Genève-Acacias, Switzerland; [4]World Intellectual Property Organization, Genève, Switzerland. ACM SIGIR Forum, 2003(a), archive 37(1), pp. 10-25.

FALL, C. J.; Benzineb K.; Guyot J.; Törcsvári, A; Fiévet P. **Computer Assisted Categorization of Patent Documents in the International Patent Classification.** Proceedings of the International Chemical Information Conference, Nîmes, October, 2003(b), (ICIC´03)

FELDMAN, R.; Sanger, J. **The Text Mining Handbook - Advanced Approaches in Analyzing Unstructured Data**. Cambridge University Press, 2007.

FIX, E; Hodges, J. **Discriminatory Analysis, non-Parametric Discrimination: Consistency Properties**. Technical Report 21-49-004(4). USAF School of Aviation Medicine, Randolph Field, Texas, 1951.

GALHO, Thaís Silva; Moraes, Sílvia Maria Wanderley. **Categorização Automática de Documentos de Texto Utilizando Lógica Difusa**. Monografia desenvolvida para obtenção do título de Bacharel em Ciência de Computação, Universidade Luterana do Brasil (ULBRA), Campus Gravataí, 2003.

GEAN, C. C.; Kaestner, C. A. A. **Classificação Automática de Documentos Usando Subespaços Aleatórios e Conjuntos de Classificadores**. In: TIL 2004 – 2^0 Workshop em Tecnologia da Informação e da Linguagem Humana, Salvador, Anais do SBC, 2004, v.1, p. 1-8.

GODBOLE, S.; Sarawagi, S. **Discriminative Methods for Multi-Labeled Classification**. Proc. of the 8th Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD'04) (pp.22-30), Sydney, Australia: Springer-Verlag, Berlin Heidelberg, LNAI 3056, 2004.

GOLDSCHMIDT, Ronaldo Ribeiro. **Assistência Inteligente à Orientação do Processo de Descoberta de Conhecimento em Base de Dados**. Tese de Doutorado, Programa de Pós-Graduação da Engenharia Elétrica do Departamento de Engenharia Elétrica da PUC-Rio, 2003.

GOLDSCHMIDT, Ronaldo; Passos, Emmanuel Piceses Lopes. **Data Mining: Um guia Prático**. Rio de Janeiro, 2005, Editora Campus.

GOMES, Anderson Lago; Costa, Danilo Freitas da. **Classificação de Documentos Usando Mineração de Textos**. Trabalho de Projeto Final do curso de Bacharelado em Informática da Universidade Católica do Salvador, dezembro de 2005.

GONZALEZ, Marco; Toscani, Daniela; Rosa, Letícia; Dorneles Rita; Lima, Vera L. S. de. **Normalização de Itens Lexicais Baseada em Sufixos**. PUCRS – Faculdade de Informática, Porto Alegre, Brasil; XVI Brazilian Symposium on Computer Graphics and Image Processing SIBGRAPI I Workshop em Tecnologia da Informação e Linguagem Humana, São Carlos, 2003.

GONZALEZ, Marco Antonio Insaurriaga. **Termos e Relacionamentos em Evidência na Recuperação de Informação**. Universidade Federal do Rio Grande do Sul, Instituto de Informática, Tese de Doutorado em Ciência de Computação, julho de 2005.

GONZALEZ, Marco. **Lematização – ferramenta CHAMA e FORMA**. 2006

GONZALEZ, M; Lima, V. L. S; Lima, J.V. **Tools for Normalization: an Alternative for Lexical Normalization**. In: VII Workshop on Comp. Proc. Of Portuguese Lang – Written and Spoken, 7. PROPOR, Proceedings…Springer-Verlag, 2006, pp.100-109.

GUAN, Hu; Zhou Jingyu; Guo Minyi. **A Class-Feature-Centroid Classifier for Text Categorization**. Computer Science Dept, Shanghai

Jiao Tong University, Shanghai, China. International World Wide Web Conference Committee; Madrid, 2009.

[1]GUO, Gongde; [1]Wang Hui; [2]Bell David; [2]Bi Yaxin; [1]Greer Kieran. **An k-NN Model based Approach and Its Application in Text Categorization**. [1]School of Computing and Mathematics, University of Ulster Newtownabbey, Northern Ireland, UK; [2]School of Computer Science, Queen's University Belfast, Belfast, UK; Lectures Notes in Computer Science, 2004, Vol. 2945/2004.

[1]GUO, Gongde; [1]Wang Hui; [2]Bell David; [2]Bi Yaxin; [1]Greer Kieran. **Using *k*NN Model for Automatic Text Categorization**. [1]School of Computing and Mathematics, University of Ulster Newtownabbey, Northern Ireland, UK; [2]School of Computer Science, Queen's University Belfast, Belfast, UK; Soft Computing – A Fusion of Foundations, Methodologies and Applications, 2006, v. 10, n. 5, pp. 423-430.

[1]HADI, Wa'el Musa; [2]Thabtah, Fadi; [3]Abdel-jaber Hussein. **A Comparative Study Using Vector Space Model with k-Nearest Neighbor on Tex Categorization Data**. [1]Department of Computer Information Systems - Arab Academy for Banking and Financial Sciences - Amman; Jordan; [2]Department of MIS, Philadelphia University, Amman, Jordan; [3]Department of Computing, University of Bradford, Bradford, UK. Proceedings of the World Congress on Engineering 2007, v. 1, pp. 296-300, WCE 2007, July 2-4, London, U.K.

[1]HADI, Wa'el Musa; [2]Thabtah, Fadi; [1]Mousa, Salahideen; [1]Al Hawari, Samer; [1]Kanaan, Ghassan; [1]Ababnih, Jafar. **A Comprehensive Comparative Study Using Vector Space Model with k-Nearest Neighbor on Text Categorization Data**. [1]Department of Computer Information Systems - Arab Academy for Banking and Financial Sciences - Amman; Jordan; [2]Department of MIS, Philadelphia University, Amman, Jordan; Asian Journal of Information Management, 2008, v. 2, Issue 1, pp. 14-22.

HAN, Eui-Hong; Karypis, George; Kumar, Vipin. **Text Categorization Using Weight Adjusted k-Nearest Neighbor Classification (WAKNN)**. Department of Computer Science and Engineering, Army HPC Research Center, University of Minnesota, Minneapolis, USA, 17/05/1999.

HOFFMANN, T.; Cai, L.; Ciaramita, M. **Learning with Taxonomies: Classifying Documents and Words**. In Workshop on Syntax, Semantics and Statistics (NIPS'03) Whistler, BC, Canada, 2003.

[1]HOSSAIN, Mohammad Kabir; [1]Reaz Abu Ahmed Sayeem; [1]Alam Rajibul; [2]Perrizo William. **Automatic Face Recognition System Using P-tree and k-Nearest Neighbor Classifier**. [1]Department of Compute Science and Engineering, North South University, Dhaka; [2]Department of Computer Science, North Dakota State University, Fargo, USA, 2002.

JANNUZZI, Anna Haydée Lanzillotti; Amorim, Rita de Cássia Rocha; Souza, Cristina Gomes de. **Implicações da Categorização e Indexação**

na Recuperação da Informação Tecnológica Contida em Documentos de Patentes. Ci. Inf., Brasília, maio/ago 2007, v.36, n. 2, p. 27-34.

JOACHIMS, T. **Text Categorization with Support Vector Machines: Learning with Many Relevant Features**. In Proceedings of the 10th European Conference on Machine Learning (ECML), New York: Springer 1998, pp. 137-142.

JOHNS, M. **Studies in Item Analysis and Prediction**. Stanford University Press, Palo Alto, CA, 1961.

KESTERING, David Alexandre de Sousa; Santos, Júlio César Lopes do. **Desenvolvimento de um Sistema de Informação**. Universidade da Amazônia, Centro de Ciências Exatas e Tecnológicas. Trabalho de Conclusão do Curso de Bacharel em Ciência de Computação, Belém, 2007.

KHAN, Md Abdul Maleq. **Fast Distance Metric Based Data Mining Techniques Using P-trees: k-Nearest –Neighbor Classification and k-Clustering**. Faculty of the North Dakota State University of Agriculture and Applied Science. Thesis of Degree of Master of Science, Fargo, North Dakota, december 2001.

KHAN, Maleq; Ding, Qin; Perrizo, William. **k-Nearest Neighbor Classification on Spatial Data Streams Using P-Trees**. Computer Science Department, North Dakota State University, Fargo, US, 2002.

KHATTAK, Akmal Saeed; Heyer, Gerhard. **Significance of Low Frequent Words in Patent Classification**. Natural Language Processing, Department of Computer Science, University of Leipzig, Leipzig, Germany. ICCG1 2011: The Sixth International Multi-Conference on Computing in the Global Information Technology.

KO, Youngjoong; Seo, Jungyun. **Text Categorization Using Feature Projections**. Department of Computer Science, Sogang University, Seoul, Korea. International Conference on Computational Linguistics, 2002.

KONCHADY, M. **Text Mining Application Programming** (1 ed.). Charles River Media, 2006.

KRIER, M.; Zacca, F. **Automatic Categorization Applications at the European Patent Office**. World Patent Information 24, 2002, pp. 187-196.

KRISHNAKUMAR, Anit. **Building a kNN Classifier for Reuters-21578 Collection**. 2006.

LARKEY, Leah S. **Some Issues in the Automatic Classification of U. S. Patents**. Working Notes for the AAAI-98 Workshop on Learning for Text Categorization. American Association for Artificial Intelligence (AAAI) Technical Report WS-98-05, 1998.

LARKEY, L. S. **A Patent Search and Classification System**. Proceedings of DL-99, 4th ACM Conference on Digital Libraries, eds. E.A.

Fox, N. Rowe, ACM Press, New York, US: Berkeley, US, 1999, pp. 179-187.

LIN, Chung-hsin; Chen, Hsinchun. **An Automatic Indexing and Neural Network Approach to Concept Retrieval and Classification of Multilingual (Chinese-English) Documents**. IEEE Transactions on Systems, Man and Cybernetics-part B: Cybernetics, February 1996, V. 26, nº1.

LOH, Stanley; Wives, Leandro Krug; Frainer Antônio Severo. **Uma Abordagem para Busca Contextual de Documentos na Internet**. CNPq/PROTEM e FAPERGS. Journal RITA, 1997, V. 4, N. 2, pp. 79-92.

LOH, Stanley; Wives, Leandro Krug. **Recuperação de Informações Usando a Expansão Semântica e Lógica Difusa**. Curso de Pós-Graduação em Ciência da Computação, Universidade Federal do Rio Grande do Sul. Congresso Internacional En Ingenieria Informatica, ICIE, 1998.

LOH, Stanley; Wives, Leandro Krug; Oliveira, José Palazzo M. de. **Concept-Based Knowledge Discovery in Texts Extracted from the Web**. ACM SIGKDD Explorations Newsletter, July 2000, V. 2, Ed. 1, pp. 29-39.

LOH, Stanley; Oliveira, José Palazzo M. de; Gastal, Fábio Leite. **Knowledge Discovery in Textual Documentation: Qualitative and Quantitative Analyses**. Journal of Documentation, 2001, V. 57, N. 5, pp.577-590.

LOH, Stanley. **Abordagem Baseada em Conceitos para Descoberta de Conhecimento em Textos**. Tese de Doutorado em Ciência de Computação. Universidade do Rio Grande do Sul, Instituto de Informática, outubro de 2001.

LOH, Stanley; Oliveira, José Palazzo M. de; Gameiro, Mauricio A. **Knowledge Discovery in Texts for Constructing Decision Support Systems**. Professor de Análise de Sistemas e Ciência da Computação da ULBRA, Applied Intelligence, 2003, V. 18, N. 3, pp. 357-366.

LOH, Stanley; Lichtnow, Daniel; Saldaña, Ramiro; Borges, Thiago; Primo, Tiago; Kickhöfel, Rodrigo Branco; Simões, Gabriel. **Investigação sobre a Identificação de Assuntos em Mensagens de Chat**. TIL XXIV Congresso da SBC, Salvador, 2004, p. 187-195.

LOH, Stanley; Amaral, Leonardo Albernaz; Wives, Leandro Krug; Oliveira, José Palazzo Moreira de. **Descoberta de Conhecimento em Textos Através da Análise de Seqüências Temporais**. 2006.

LOPES, Maria Célia Santos. **Mineração de Dados Textuais Utilizando Técnicas de Clustering para o Idioma Português**. Tese de Doutorado; Departamento de Engenharia Civil; UFRJ; outubro de 2004.

LOVINS, J. B. **Development of a Stemming Algorithm. Mechanical Translation and Computacional Linguistics**. 1968, V. 11, pp. 22-31.

LUHN, H. P. **The Automatic Creation of Literature Abstracts**. IBM Journal of Research and Development 2(2), 1958, pp. 159-165.

MAGALHÃES, Lúcia Helena de; Arbex, Márcio Aarestrup. **O Text Mining para Apoio a Tomada de Decisão**. Faculdades Integradas do Instituto Vianna Júnior, 2006.

MARTINS, Claudia A.; Monard Maria Carolina; Matsubara, Edson T. **Uma Ferramenta Computacional para Auxiliar no Pré-Processamento de Textos**. Lab. Inteligência Computacional. Anais do XXIII Congresso da http://www.icmc.usp.br/~mcmonard/public/enia/2003.pdf.

MORAES, Silvia Maria Wanderley; Lima, Vera Lúcia Strube de. **Um Estudo Sobre Categorização Hierárquica de uma Grande Coleção de Textos em Língua Portuguesa**. PUCRS - Programa de Pós-Graduação em Ciência da Computação, Pontifícia Universidade Católica do Rio Grande do Sul. Anais de XXVII Congresso de SBC 2007, Workshop em Tecnologia de Informação e de Linguagem Humana.

MORAES, Silvia Maria Wanderley; Lima, Vera Lúcia Strube de. **Categorização de Textos Baseada em Conceitos**. Encontro de PLN; PUCRS - Programa de Pós-Graduação em Ciência da Computação. Pontifícia Universidade Católica do Rio Grande do Sul, 2008.

Multimedia ‘07 Augsburg, Germany. **Semantic Concept-Based Query Expansion and Re-ranking for Multimedia Retrieval**. A Comparative Review and New Approaches; Paper #1569046180, 2007.

ORENGO, Viviane Moreira. **A Stemming Algorithm for the Portuguese Language**. In Proceedings of the SPIRE Conference, Laguna de San Raphael, November 13-15, 2001(a).

ORENGO, V. M.; Huyck, C. **A Stemming Algorithm for the Portuguese Language**. Proceedings of the Eight International Symposium on String Processing and Information Retrieval, 2001(b), pp. 186-193.

PAICE, Chris D. **Another Stemmer**; Department of Computing Lancaster University, Bailrigg, Lancaster, U. K, 1983.

PEREIRA, Rachel; Ricarte, Ivan; Gomide, Fernando. **Ontologia Relacional Fuzzy em Sistemas de Recuperação de Informação**. XXV Congresso da Sociedade Brasileira de Computação – SBC. Unisinos – São Leopoldo – RS, 2005.

PERES, Sarajane Marques; Boscarioli, Clodis. **Sistemas Gerenciadores de Banco de Dados Relacionais Fuzzy; Uma Aplicação em Recuperação de Informação**. Colegiado de Informática, Universidade Estadual do Oeste do Paraná, Acta Scientiarum Technology, Maringá, 2002, V. 24, N. 6, pp. 1733-1743.

PERRIZO, William. **System and Method for Organizing Compressing and Structuring Data for Data Mining Readiness**. North Dakota State University Research Foundation, United States Patent Application Publication US2003/0208488 A1, 2003.

PORTER, M. **An Algorithm for Suffixing Stripping**. Program: electronic library and information systems, 1980, V. 14, pp. 130-137.

QUINLAN, J. **Induction of Decision Trees**. Machine Learning 1986, V. 1, n. 1, pp. 81-106.

RAHAL, Imad; Perrizo William. **An Optimized Approach for kNN Text Categorization Using P-trees**. Computer Science Department, North DakotaState University, Fargo, ND, USA, 2004. ACM Symposium on Applied Computing.

[1]RICHTER, Georg; [2]MacFarlane, Andrew. **The Impact of Metadata on the Accuracy of Automated Patent Classification.** (1) Thomson Scientific; (2) Department of Information Science, Centre for Interactive Systems Research London, World Patent Information 27(2005) pp. 13-26.

ROITBLAT, Herbert L. **Tools for Text Categorization**. Orca Tec, EDI – Leadership Summit Santa Monica California, 2013.

ROUSU, J.; Saunders, C.; Szedmark, S; Shawe-Taylor, J. **Learning Hierarchical Multi-Category Text Classification Models**. In Proc. of the 22nd Int. Conf. on Machine Learning (pp. 745-752); Bonn, Germany: Omnipress, 2005.

SALTON, G. **Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer**. Addison-Wesley; Reading; Pennsylvania, 1989.

SALTON, G.; Buckley, C. **Term-Weighting Approaches in Automatic Text Retrieval**. Information Processing & Management, 1988, Vol. 24(5), pp. 513-523.

SANTOS, Cássia Trojahn dos; Osório Fernando Santos. **O Uso de Técnicas de Aprendizado de Máquina na Categorização de Documentos**. Universidade do Vale do Rio dos Sinos, Centro de Ciências Exatas e Tecnológicas, São Leopoldo, 2003.

SANTOS, Maria Angela Moscalewski Roveredo dos. **Extraindo Regras de Associação a Partir de Textos**. Dissertação de Mestrado em Informática Aplicada. Universidade Católica do Paraná, Curitiba, 2002, disponível em http://www.ppgia.pucpr.br/ensino/defesas/Maria_Angela%20_2002.PDF

SCHIJVENAARS, Bob J. A.; Schuemie, Martijn J.; Mulligen, Erick M. van; Weeber, Marc; Jelier, Rob; Mons, Barend; Kors, Jan A. **A Concept-Based Approach to Text Categorization**. Notebook Paper TREC 2005 Genomics Track, Department of Medical Informatics.

SCHÜTZE, H.; Hull, D. A.; Pedersen, J. O. **A Comparison of Classifiers and Document Representations for the Routing Problem**. Proceedings of SIGIR-95, 18th ACM International Conference on Research and Development Information Retrieval, eds. E. A. Fox, P. Ingwersen, R. Fidel, ACM Press, New York, US: Seattle, US, 1995, pp. 229-237.

SEBASTIANI, Fabrizio. **Machine Learning in Automated Text Categorization**. Consiglio Nazionale delle Ricerche, Italy. ACM Computing Surveys, March 2002, V. 34, N^0 1, pp. 1-47. Tech. Rep. IEI-B4-31-1999.

SEDDIQUI, Md. Hanif; Seki, Yohei; Aono, Masaki. **A Semantic Approach to Patent Mining for Relating IPC to a Research Paper Abstract**. Toyohashi University of Technology, Aichi, Japan. Computer Science & Engineering, 2008.

SHIH, Meng-Jung; Liu, Duen-Ren. **Patent Classification Using Ontology- Based Patent Network Analysis**. Institute of Information Management; National Chiao Tung University. Hsinchu, Taiwan, PACIS 2010 Proceedings Paper 95.

SILVA, A. A. **Aîruri: Um Portal para Mineração de Textos Integrado a Grids**. Dissertação de Mestrado. Engenharia Civil, UFRJ, 2007.

SILVA, Cassiana Fagundes[1]; Galho, Thaís Silva[2]. **Mineração de Textos Utilizando Técnicas de Aprendizado de Máquina e Lógica Difusa**.[1]Faculdade de Seama, Macapá, AP; [2]Universidade Luterana do Brasil (ULBRA). II Congresso Sul Catarinense de Computação Criciúma, 2006.

SILVA, Luiza Maria Oliveira da. **Uma Aplicação de Árvores de Decisão, Redes Neurais e kNN para a Identificação de Modelos ARMA Não-Sazonais e Sazonais**. Tese de Doutorado PUC-RJ. 09/2005.

SMITH, Harold. **Automation of Patent Classification**. US Patent and Trademark Office, US Department of Commerce, Washington, DC, USA, World Patent Information 24 (2002), pp. 269-271.

SPHINX Brasil – **Manual do Sphinx v4– Análise Sintática e Lematização**. www.sphinxbrasil.com

SOUCY, Pascal; Mineau Guy W. **A Simple k-NN Algorithm For Text Categorization**. Department of Computer Science, Université Laval, Québec, Canada. Proceedings of the 2001 IEEE International Conference on Data Mining (ICDM'01) (2001a).

SOUCY, Pascal; Mineau Guy W. **A Simple Feature Selection Method for Text Classification**. In Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI-01). (2001b).

TANABE, Akeo. **A Verbal Stemmer for Brazilian Portuguese Language**. Departamento de Informática, PUC – RJ, Monografia em Ciência de Computação n^0 26/08, 2008.

TAN, P.-N.; Steinbach, M.; Kumar, V. **Introduction to Data Mining**. Pearson Addison Wesley, 2005.

TASCI, Serafettin; Güngör, Tunga. **LDA-based Keyword Selection in Text Categorization**. Computer Engineering Department. Bogazici University, Istambul, Turkey. In Proceedings: ISCIS 2009 pp 230-235 Export: BIBTEX LNCS IEEE ACM.

[1]TIKK, Domonkos; [2]Biró, György. **Experiment with Hierarchical Text Categorization Method on the WIPO-Alpha Patent Collection**. [1]Department of Telecom and Media Informatics, Budapest University of Technology and Economics, Budapest, Hungary; [2]Department of Informatics, Eötvös Loránd Science University, Budapest, Hungary; Proceedings of the Fourth International Symposium on Uncertainty Modeling and Analysis, 2003 IEEE.

[1]TIKK, Domonkos; [2]Biró György. **Text Categorization on a Multi–Lingual Corpus**. [1]Department of Telecommunication and Media Informatics, Budapest University of Technology and Economics; [2]Department of Informatics, Eötvös Loránd Science University, Department of Informatics, 2001.

[1]TIKK, Domonkos; [2]Biró György; [3]Törcsvári, Attila. **A Hierarchical Online Classifier for Patent Categorization**. [1]Department of Telecommunication and Media Informatics, Budapest University of Technology and Economics; [2]Textminer, Budapest, Hungary; [3]Arcanum Development Ltd., Budapest, Hungary, 2008.

TRAPPEY, A. J. C.; Hsu, F.-C.; Trappey, C. V.; Lin, C.–I. **Development of a Patent Document Classification and Search Platform Using a Back-Propagation Network**. Expert Systems with Applications, 2006, Vol.31, pp. 755-765.

UBER, José Lino. **Descoberta de Conhecimento com o Uso de Text Mining Aplicada ao SAC**. Universidade Regional de Blumenau. Centro de Ciências Exatas e Naturais – Curso de Ciências da Computação – Bacharelado, 2004.

VAPNICK, V. **The Nature of Statistical Learning Theory**. Springer-Verlag, New York, 1995.

WAGELAAR, Dennis. **A Concept-Based Approach for Early Aspect Modelling**. Vrije Universiteit Brussel. AOSD 2003 Workshop on Early Aspects, Boston, MA, USA.WANG, H. (2003). **Nearest Neighbours Without k: A Classification Formalism Based on Probability**. Tech. Report CS-03-02, Faculty of Informatics, University of Ulster, UK, 2003.

WANG, Wei; Li, Sujian; Wang Chen; ICL at NTCIR-7. **An Improved kNN Algorithm for Text Categorization; Inst. of Computational Linguistics**. Peking University. Proceedings of NTCIR-7 Worshop Meeting. December 2008, Tokyo, Japan.

WETTSCHERECK, D.; Dietterich, T. **Locally Adaptive Nearest Neighbor Algorithms**. In Advances in Neural Information Processing Systems. v. 6, pp. 184-191. Morgan Kaufmann, San Mateo, CA, 1994.

WILSON, D. RANDALL; MARTINEZ, TONY R. **Reduction Techniques for Instance-Based Learning Algorithms**. Neural Network & Machine Learning Laboratory, Computer Science Department. Brigham Young University, USA. Machine Learning 38, 2000, pp. 257-286.

World Intellectual Property Organization – IP Services. **Readme Information for WIPO-alpha Autocategorization Training Set**. Geneva, june 2009.

YANG,Y.; Pedersen, J. O. **A comparative study on feature selection in text categorization**. Proceedings of ICML-97, 14th International Conference on Machine Learning, ed. D. H. Fisher, Morgan Kaufmann Publishers, San Francisco, US, Nashville, US, 1997, pp. 412-420.

ZHANG, Tong; Oles, Frank J. **Text Categorization Based on Regularized Linear Classification Methods**. Mathematical Sciences Department IBM T. J. Watson Research Center, Information Retrieval 4 (2001), pp. 5-31.

ZIPF, G. K. **Human Behavior and the Principle of Least Effort**. Addison-Wesley.

Portuguese Stemming Algorithms

http://snowball.tartarus.org/algorithms/portuguese/stemmmer.html disponível em 14/05/2009

http://snowball.tartarus.org/algorithms/portuguese/stop.txt disponível em 14/05/2009

http://linguateca.di.uminho.pt/Paulo/stopwords/folha.MF100.txt disponível em 14/05/2009

http://linguateca.di.uminho.pt/Paulo/stopwords/folha.MF300.txt disponível em 14/05/2009

http://linguateca.di.uminho.pt/Paulo/stopwords/publico.MF300.txt disponível em 14/05/2009

http://linguateca.di.uminho.pt/Paulo/stopwords/chave.MF300.txt disponível em 14/05/2009.

http://meta.wikimedia.org/wiki/Stop_word_list/pt disponível em 14/05/2009

[http://www.ranks.nl/stopwords/portuguese.html disponível em 14/05/2009

http://members.unine.ch/jacques.savoy/clef/portugueseST2.txt disponível em 14/05/2009

http://www.unine.ch/info/clef/portugueseST.txt disponível em 14/05/2009

http://r-forge.r-project.org/plugins/scmsvn/viewcvs.php/trunk/tm/inst/stopwords/portuguese
disponível em 14/05/2009

http://cpansearch.perl.org/src/XERN/Lingua-PT-Stemmer-0.01/lib/Lingua/PT/Stemmer... disponível em 14/05/2009

[http://www.icmc.usp.br/~andre/research/neural/index.htm disponível em 12/03/2010

http://members.unine.ch/jacques.savoy/clef/portugueseStemmer.txt
disponível em 14/05/2009

http://www.ompi.ch/classifications/ipc/en/ITsupport/Categorization/dataset/wipo.alpha-readme.html

http://ipc.inpi.gov.br

9.0
Apêndice 1
Quantidade de Documentos Discriminados por Grupo e Subclasse IPC

Tabela 24 – Quantidade de Documentos Discriminados por Grupo e Subclasse (IPC) para a Categoria H02P.

Subclasse (IPC)	Grupo (IPC)	Quantidade Total	
		Grupo	Subclasse
H02P CONTROLE OU REGULAÇÃO	H02P 1/00	55	307
	H02P 3/00	14	
	H02P 5/00	31	
	H02P 6/00	41	
	H02P 7/00	93	
	H02P 8/00	9	
	H02P 9/00	44	
	H02P 11/00	0	
	H02P 13/00	3	
	H02P 15/00	1	
	H02P 17/00	0	
	H02P 19/00	0	
	H02P 21/00	6	
	H02P 23/00	6	
	H02P 25/00	2	
	H02P 27/00	2	

Tabela 25 – Quantidade de Documentos Discriminados por Grupo e Subclasse (IPC) para a Categoria A47C

Subclasse (IPC)	Grupo (IPC)	Quantidade Total	
		Grupo	Subclasse
A47C CADEIRAS	A47C1/00	46	331
	A47C3/00	22	
	A47C4/00	33	
	A47C5/00	9	
	A47C7/00	52	
	A47C9/00	3	
	A47C11/00	3	
	A47C12/00	0	
	A47C13/00	4	
	A47C15/00	5	
	A47C16/00	11	
	A47C17/00	59	
	A47C19/00	15	
	A47C20/00	8	
	A47C21/00	8	
	A47C23/00	8	
	A47C25/00	0	
	A47C27/00	33	
	A47C29/00	2	
	A47C31/00	10	

Tabela 26 – Quantidade de Documentos Discriminados por Grupo e Subclasse (IPC) para as Categorias H05B, H02G e H02B.

Subclasse (IPC)	**Grupo (IPC)**	**Quantidade Total**	
		Grupo	**Subclasse**
H05B AQUECIMEN-TO H05B(A)	H05B 1/00	97	294
	H05B3/00	111	
	H05B6/00	67	
	H05B7/00	18	
	H05B11/00	1	
H05B ILUMINAÇÃO H05B(I)	H05B31/00	0	205
	H05B 33/00	26	
	H05B 35/00	1	
	H05B 37/00	63	
	H05B 39/00	22	
	H05B 41/00	88	
	H05B 43/00	5	
H02G CABOS E LINHAS ELÉTRICAS	H02G 1/00	71	500
	H02G 3/00	221	
	H02G 5/00	12	
	H02G 7/00	77	
	H02G 9/00	11	
	H02G 11/00	9	
	H02G 13/00	3	
	H02G 15/00	96	
H02B PAINÉIS	H02B 1/00	218	293
	H02B 3/00	5	
	H02B 5/00	13	
	H02B 7/00	3	
	H02B11/00	12	
	H02B13/00	40	
	H02B15/00	2	

Tabela 27 – Quantidade de Documentos Discriminados por Grupo e Subclasse (IPC) para as Categorias H01F e H02M.

Subclasse (IPC)	**Grupo (IPC)**	**Quantidade Total**	
		Grupo	**Subclasse**
H01F MAGNETOS E INDUTÂNCIAS	H01F 1/00	58	434
	H01F 3/00	5	
	H01F 5/00	21	
	H01F 6/00	4	
	H01F 7/00	41	
	H01F 10/00	5	
	H01F 13/00	9	
	H01F 15/00	2	
	H01F 17/00	5	
	H01F 19/00	1	
	H01F 21/00	2	
	H01F 27/00	139	
	H01F 29/00	14	
	H01F 30/00	8	
	H01F 35/00	3	
	H01F 36/00	1	
	H01F 37/00	21	
	H01F 38/00	52	
	H01F 39/00	3	
	H01F 40/00	2	
	H01F 41/00	38	
H02M CONVERSÃO DE ENERGIA	H02M 1/00	53	274
	H02M 3/00	71	
	H02M 5/00	39	
	H02M 7/00	102	
	H02M 9/00	2	
	H02M 11/00	7	

Tabela 28 – Quantidade de Documentos Discriminados por Grupo e Subclasse (IPC) para a Categoria H01J.

Subclasse (IPC)	**Grupo (IPC)**	**Quantidade Total**		
		Grupo	**Parcial**	**Subclasse**
H01J TUBOS DE DESCARGA	H01J 1/00	22	237	278
	H01J 3/00	5		
	H01J 5/00	15		
	H01J 7/00	15		
	H01J 9/00	17		
	H01J 11/00	2		
	H01J 13/00	3		
	H01J 15/00	0		
	H01J 17/00	8		
	H01J 19/00	2		
	H01J 21/00	0		
	H01J 23/00	2		
	H01J 25/00	4		
	H01J 27/00	0		
	H01J 29/00	76		
	H01J 31/00	6		
	H01J 33/00	1		
	H01J 35/00	3		
	H01J 37/00	41		
	H01J 40/00	5		
	H01J 41/00	0		
	H01J 43/00	0		
	H01J 45/00	2		
	H01J 47/00	1		
	H01J 49/00	7		
H01J LÂMPADAS DE DESCARGA	H01J 61/00	32	41	
	H01J 63/00	0		
	H01J 65/00	9		

Tabela 29 – Quantidade de Documentos Discriminados por Grupo e Subclasse (IPC) para a Categoria H02K.

Subclasse (IPC)	**Grupo (IPC)**	**Quantidade Total**	
		Grupo	**Subclasse**
H02K MÁQUINAS ELÉTRICAS	H02K 1/00	78	500
	H02K 3/00	28	
	H02K 5/00	69	
	H02K 7/00	65	
	H02K 9/00	26	
	H02K11/00	26	
	H02K 13/00	11	
	H02K15/00	31	
	H02K 16/00	6	
	H02K 17/00	16	
	H02K 19/00	3	
	H02K 21/00	19	
	H02K 23/00	8	
	H02K 24/00	0	
	H02K 25/00	1	
	H02K 26/00	0	
	H02K 27/00	1	
	H02K 29/00	12	
	H02K 31/00	0	
	H02K 33/00	16	
	H02K 35/00	6	
	H02K 37/00	1	
	H02K 39/00	0	
	H02K 41/00	12	
	H02K 44/00	3	
	H02K 47/00	4	
	H02K 49/00	5	
	H02K 51/00	6	
	H02K 53/00	25	
	H02K 55/00	13	
	H02K 57/00	9	

Tabela 30– Quantidade de Documentos Discriminados por Grupo e Subclasse (IPC) para a Categoria A47B.

Subclasse (IPC)	Grupo - IPC (Quantidade por Grupo)		Quantidade Total por Subclasse
A47B MÓVEIS; ARTIGOS OU APARELHOS DOMÉSTICOS; MOINHOS DE CAFÉ; MOINHOS DE ESPECIARIA; ASPIRADORES EM GERAL.	A47B1/00 (4)	A47B69/00 (0)	447
	A47B3/00 (21)	A47B71/00 (1)	
	A47B5/00 (3)	A47B73/00 (1)	
	A47B7/00 (2)	A47B75/00 (3)	
	A47B9/00 (7)	A47B77/00 (16)	
	A47B11/00 (6)	A47B79/00 (0)	
	A47B13/00 (15)	A47B81/00 (21)	
	A47B17/00 (2)	A47B83/00 (5)	
	A47B19/00 (3)	A47B85/00 (4)	
	A47B21/00 (30)	A47B87/00 (13)	
	A47B23/00 (12)	A47B88/00 (31)	
	A47B25/00 (1)	A47B91/00 (32)	
	A47B27/00 (1)	A47B95/00 (16)	
	A47B29/00 (2)	A47B96/00 (54)	
	A47B31/00 (1)	A47B97/00 (17)	
	A47B33/00 (0)		
	A47B35/00 (4)		
	A47B37/00 (24)		
	A47B39/00 (5)		
	A47B41/00 (5)		
	A47B43/00 (3)		
	A47B45/00 (2)		
	A47B46/00 (12)		
	A47B47/00 (23)		
	A47B49/00 (5)		
	A47B51/00 (1)		
	A47B53/00 (1)		
	A47B55/00 (4)		
	A47B57/00 (13)		
	A47B61/00 (9)		
	A47B63/00 (3)		
	A47B65/00 (2)		
	A47B67/00 (7)		

10.0
Apêndice 2
Lista de *Stopwards*

Tabela 31A - *StopList* ou Lista de *Stopwords*

(a) (as)	abaix(o) (a) (ado) (ada)	abrangente	ach(o) (ar) (ou) (ado)	acima	acordo	ademais
adequad(a) (o) (as)(os) (amente)	adiç(ão) (ões)	adicional (mente)	afi(m) (nidade)	afirm(a) (ou)	agora	ainda
além	algo	alguém	algu(m) (ma) (mas) (ns)	alternativa (s) (mente)	aludid(a) (o) (as) (os)	amanhã
amb(as) (os)	ampl(a) (as) (o) (os)	anex(o) (os) (a) (as)	ano (s)	anterior (mente)	antig(a) (o) (as) (os) (amente)	apenas
aperfeiçoa-d(a) (o) (as) (os) (amente)	aperfeiço-amento (s)	após	apreço (s)	apresent(a) (o) (amos) (ada) (ado) (adas)(ados) (ando) (am)	apresent(e) (ar) (ará) (arem) (em) (ou) (ação) (ações)	apresenta-se
apropria-d(a) (o) (as) (os)	aproxi-m(a)(o) (as) (adamente)	aquel(a) (as) (e) (es)	aqui	aquilo	área	assim
até	através	atual (mente)	aument(ar) (ou)	aumento	avanço	básic(a) (o) (as) (os) (amente)
beleza	bem	benefíci(o) (os)	bilhões	boa	bom bons	cada
característ-ic(a) (as) (o) (os)	caracteri-z(ada) (adas) (ado) (ados) (ando)(ante)	caso	cem	cento	centr(al) (ais)	centro (s)
cerca	certa (mente)	certificad(o) (a) (os) (as)	cert(o) (a) (amente)	cheg(a) (ada) (o) (as) (am) (ando) (ar)	cinco	concebid(o) (a) (os) (as)
cita (da) (o) (do)	coempre-ga (o) (as) (os)	coisa (s)	coloc(a) (o) (as) (ação) (amos) (ada) (adas) (ado) (ados)	coloc(ar) (amos) (ando)	com	comerci-(al) (ais) (alização) (alizado) (alizados)

Tabela 31B - *StopList* ou Lista de *Stopwords*

comérci(o) (os)	como	compatí-v(el) (eis)	complet(a) (amente) (o) (as) (os) (ando)	compost(a) (o) (os) (as)	compreen-d(er) (endo) (a) (o) (e) (em) (ida) (erem) (ido) (idas) (idos)	conceb(er) (ido) (endo) (ida) (idas) (idos)
concern(e) (ente)	concordân-ci(a) (as)	concordante	condizente	conferem-lhe	confer(ir) (irmos)	confiáv(el) (eis)
conforme	congênere (s)	consecuti-v(a) (o) (as) (os) (amente)	consegue-se	conse-gu(ida) (ir) (ido) (idas) (idos) (indo)	conse-qu(ência) (ente) (entemente)	consider (ação) (ável)(áveis (ando-se) (avelmente)
consist(e) (em) (ir) (indo)	constat(ar) (ado) (ados) (ei)	constitu(ir) (ido) (i) (em) (ida) (idas) (ido) (idos) (indo) (tiva)	constituindo -se constituir-se constitui-se	cont(er) (em) (endo) (ida) (idas) (ido) (idos)	contempl(a) (ando)	contexto
continua-mente	contra	contudo	convencio-n(al) (ais)	conveniente (s) (mente)	correspon-dente (mente) (s)	cri(a) (ei) (ado) (ados) (ada) (adas) (ando)
crise	cuj(a) (as) (o) (os)	custo	customiza-d(o) (a) (as) (os)	d(a) (e) (o) (as) (os)	dad(o) (a) (os) (as)	dando-lhe
daquel(a) (e) (as) (es)	daquilo	dar	decorrente (s)	definid(a) (s) (o) (os)	definitiv(o) (a) (os) (as) (amente)	deix(ar) (o) (e) (ada) (ado) (adas) (ados) (ando) (am)
del(a) (e) (as) (es)	dentre	depend(e) (endo) (erá) (eremos)	depois	deposit(ar) (o) (a) (ada) (ado) (adas) (ados) (os)	destacando (-se)	destacável
descrev(er) (e) (endo) (e-se)	descrit(a) (as) (ivo) (o)	desde	desej(ar) (a) (e) (ada) (em) (adas) (ado) (ados) (am) (áveis) (ável)	desenho (s)	desenvolvi-mento	design(a) (ação) (ando) (ada) (adas) (ado) (ados)
desperdi-ç(ar) (ando)	desperdício	dess(a) (as) (e) (es)	dest(a) (as) (e) (es)	destac(ar) (a) (o) (ada) (adas) (ado) (ados) (amos) (ando)	destacando-se	destaque

Tabela 31C - *StopList* ou Lista de *Stopwords*

desvanta-g(em) (ens)	detalhe (s)	determina-d(a) (o) (as) (os)	dev(er) (em) (e) (endo) (ia) (iam) (erá) (erão) (eria) (eriam)	dez dois duas	dia (s)	diminu(ir) (ia) (i)
dispendio dispêndio	disse	disso	disto	dit(a) (as) (o) (os)	divers(a) (as) (o) (os)	divulgad(a) (o) (as) (os)
diz (em)	dois duas	dotad(a) (o) (as) (os)	drastic(a) (o) (as) (os) (amente)	durante	duzentos	e
ecologi(a) (as) (co) (ca) (cos) (cas) (camente)	economi(a) (as) (camente)	eficiente (s) (mente) eficiência	el(a) (e) (as) (es)	em	embora	enquanto
Entre	era (m)	especi(al) (almente) (ais)	específic(a) (as) (o) (os) (ada) (ado) (adas) (ados) (amente)	esquema	ess(a) (as) (e) (es)	essencial (mente)
est(ar) (á) (amos) (ão) (ará) (arão) (eve) (arem) (ava) (avam) (avamos)	est(a) (as) (e) (es)	estej(a) (am) (amos)	estenderem-se	estiv(er) (emos) (era) (eram) (erem) (ermos) (esse) (essem) (este)	est(ar) (ou)	eu
exagera-d(a) (as) (o) (os) (amente)	excepcio-n(al) (ais) (almente)	excessiv(a) (as) (o) (os) (amente)	exempl(o) (os) (ificando) (ificativa)	fácil (mente)	falta (s)	fato (s)
faz (er) (em) (endo) (-se) (endo-se)	fé	feit(a) (as) (o) (os)	fez	ficou	figura (s)	fim final finalidade
for foi fomos fora foram forem	form(a) (as) (ada) (adas) (ado) (ados) (am) (ando) (ar) (arem)	form(e) (em)	formos fui	forneça	fornec(er) (e) (em) (endo) (ida) (idas) (ido) (idos)	fornecimen-t(o) (os)

Tabela 31D - *StopList* ou Lista de *Stopwords*

fosse (m)	frequente-mente	funcionali-zada (s)	generic(a) (as) (amente)	ger(al) (ais) (almente)	gingatesc(a) (o) (as) (os)	graças
grande (s)	grupo (s)	há houve	habitual (mente)	haj(a) (am) (amos)	havemos havia hei	história (s)
historic(o) (a) (amente)	hoje	houv(e) (emos) (er) (era) (eram) (eramos) (erem) (ermos)	houvess(e) (em) (emos)	idealiz(ar) (a) (ado) (ada) (ados) (adas) (ou)	ilustr(ar) (a) (ou) (ada) (ado) (adas) (ados)	imediat(o) (amente)
inclu(ir) (e) (em) (i) (ida) (idas) (ido) (idos) (indo)	independen-te (mente)	industri(a) (as) (almente)	inerente (mente)	infinit(a) (s) (o) (os)	influenci(a) (as) (ação) (ar) (ável) (áveis)	influenci-(ada) (adas) (ado) (ados)
inici(o) (almente)	inov(a) (ar) (ação) (ações) (ando)	inspeç(ão) (ões)	integrad(a) (o) (adas) (ados)	integral-mente	intempérie (s)	intencio-n(ar) (ado) (ada) (ados) (adas) (o) (os)
interdepen-dência	interroga-ção	Introdução	invenç(ão) (ões)	inventiv(o) (s) (a) (as)	invento	isso isto
já	janeiro	lá	lado	lh(e) (es)	ligeir(a) (o) (as) (os) (amente)	lo
loc(al) (ais) (almente)	lugar (es)	maior (es)	mais	maneira	mante-l(a) (as) (o) (os)	mantem-se
mantendo-(a) (se) (o) (as) (os)	mantenha (m)	mant(er) (inha)	mão	marcante	mas	me
média	mediante	medid(a) (o) (as) (os)	meio	melhor (es) (ia)	melhora-ment(o) (os)	melhorando (-se)
menor (es)	menos	mês (es)	mesm(a) (as) (o) (os)	meu (s) minha	mil (hões)	minh(a) (as)
mo	modalida-d(e) (es)	model(o) (os)	modern(o) (a) (os) (as) (idade)	modo	momenta-nea (mente)	momento
muit(a)(as) (o) (os)	mundial (mente)	n(a) (as) (o) (os)	nada	não	naquel(a) (as) (e) (es)	naturez(a) (as)

Tabela 31E - *StopList* ou Lista de *Stopwords*

necessá-ri(a) (as) (o) (os)	necessidade	nem	nenhum (a)	ness(a) (as)	nest(a) (as) (e) (es)	ninguém
norm(al) (ais) (almente)	noss(a) (as) (o) (os)	notad(a) (o) (as) (os) (amente)	noutr(a) (o) (as) (os)	nov(a) (o) (as) (os) (amente)	nove	novidade (s)
num (a) (as)	número (s)	nunca	objetivo	obra	obt(er) (emos) (enção) (era) (endo) (enha) (ermos)	obstante
obt(er-se) (em-se) (endo-se)	obtid(a) (o) (as) (os)	ocasião	ocup(ar) (a) (o) (am) (ando)	oitenta oito	onde	ontem
opcional (mente)	operacional (mente)	ou	outr(a) (o) (as) (os)	para	parci(al) (ais) (almente)	parte (s)
particular (mente)	passado	pass(ivel) (iveis)	patente (s)	peculari-da-de (s)	pedido (s)	pela (s)
pelo (s)	pequen(a) (o) (as) (os)	per perante	perecív(el) (eis)	performan-ce	permeia	pertencente
pi	pod(e) (em) (emos) (ia) (iam) (er) (endo)	pod(e-se) (endo-se) (em-se)	poder(á) (ão) (em) (ia) (iam)	poder-se-á	põe	pois
por	porção	porém	porque	possa (m)	possibilida-de (s)	possibilit(a) (ada) (adas) (ado) (ados) (am) (ando)
possibili-t(ar) (ará) (e) (em)	possivel (mente)	posso	possu(a) (am) (i) (indo) (ir)	pouc(a) (s) (o) (os)	pouquíssi-m(a) (o) (as) (os)	pratic(o) (amente)
pré	precisa-mente	preço	predetermi-nad(a) (o) (as) (os)	predomi-nante (mente)	preestabele-cid(a) (o) (as) (os)	preferen-c(ia) (ial) (ialmente)
preferen-t(e) (emente)	preferida	preferivel (mente)	presente (mente)	previsto (a) (os) (as)	primeira (as) (o) (os)	principal (mente)

Tabela 31F- *StopList* ou Lista de *Stopwords*

privilégio	procedi-mento	prontamen-te	propiciad(a) (o) (as) (os)	propici(ar) (ando)	propõe-se	proporcio-n(ar) (a) (am) (ada) (adas) (ado) (ados)
proporcio-n(al) (ais) (ando)	proporcio-n(ará)(arão) (arem)	proposit(o) (almente)	propost(a) (as) (o) (os)	propri(a) (amente) (as) (o) (os)	prove-se	qu(al) (ais)
qualquer quaisquer	quando	quant(o) (a) (os) (as)	quase	quatro (s)	que (m)	quer
razoáv(el) (eis)	re(al) (ais)	realiz(ar) (o) (a) (ada) (adas) (am) (ados) (e)	realizaç(ão) (ões)	realiz(ará) (arem) (ando) (a-se)	reciproc(a) (amente) (o)	redundante (mente) (s)
refere-se referem-se	referenci(a) (ais) (as)	referente (s)	refer(ida) (idas) (ido) (idos) (ir)	referindo	regularmen-te	reivindica-ç(ão) (ões)
relaç(ão) (ões)	relacion(a) (o) (as) (os) (ada) (adas) (ado) (ados)	relativ(a) (o) (amente)	relatório	relevante (s)	reparo (s)	requer (em) (endo)
requeiram	requerente (s)	requer(er) (ida) (idas) (ido) (idos)	requerimen-to (s)	requisita-d(a) (o) (as) (os)	requisito (s)	respectiv(a) (amente) (as) (o) (os)
respeito	restante	resultante	resum(e) (o)	revelad(a) (as) (o) (os)	revis(ão) (ões)	são
satisfeita	se	segundo	seis	sej(a) (am) (amos)	sem	sempre
sendo	ser (em) (á) (ão) (ei) (emos) (ia) (iam)	sete setenta seis	seu (s) sua (s)	si	sido	significati-v(a) (o) (as) (os) (amente)
similar (es) (idade)	simples (mente)	simplific(a) (o) (as) (os) (ação) (ada) (ado) (ando) (adora) (ar)	só somente	sob	sobre	sobretudo
solucion(a) (o)	somos	sou	sub	submet(er) (endo) (ida) (ido)	subsequente (mente)	substancial (mente)
substitui-ç(ão) (ões)	substituir	sucedid(a) (o) (as) (os)	suficiente (s)	super(a) (ada) (adas) (ado) (ados) (ar) (ando)	t(al) (ais)	talvez

Tabela 31G - *StopList* ou Lista de *Stopwords*

também	tampouco	tanto	tão	técnic(a) (as)	t(er) (endo) (em) (emos)	ter(ei) (emos) (ia) (iam)
tenh(a) (am) (amos) (o)	teoric(a) (o) (amente) (as) (os)	teu (s) tua (s)	teve	ti	tido	tinh(a) (am)
tipic(a) (o) (as) (os) (amente)	tiv(er) (emos) (e) (esse) (essem)	tiver(a) (em) (am) (mos)	tod(a) (as) (o) (os)	todavia	tornad(a) (o) (as) (os)	torn(am) (ando) (ar) (ara) (e) (em) (o)
tornar-se tornaram-se torna-se	tot(al) (ais)	traduzir traduz-se	trata-se	três	trezentos	tu tua (s)
tudo	últim(a) (o) (as) (os)	um (a) (as) uns	usuário (s)	út(il) (eis)	utilidade (s)	vai
vantag(em) (ens)	vantaj(osa) (oso) (osamente)	vão	vasta (mente)	vem	vendo	ver
versatil (idade)	vez (es)	viag(em) (ens)	vindo	vinte	vir	virtude
vis(a) (ando) (ar)	visto	visual (mente)	voce (s)	vós	W X Y Z	A B C D
E F G H	J K L M	P Q R S	I O U	N T V		

11.0
Apêndice 3
Lista de Termos Compostos Modificados e com Erros Ortográficos

Tabela 32A – Termos Compostos Modificados e com Erros Ortográficos

Termo Original	Termo Modificado	Termo Original	Termo Modificado
alta tensão alta-tensão	altatensão	alta voltagem alta-voltagem	altavoltagem
alta potência alta-potência	altapotência	amarelo-verde	amareloverde
alto-falante	altofalante	auto-falante	autofalante
anti-brilho	antibrilho	auto-comandada	autocomandada
auto-sublimação	autosublimação	anti-estático	antiestático
auto cozimento auto-cozimento	autocozimento	auto-protegid(o) (s)(a)(as)	autoprotegid(o) (a)(os)(as)
auto-corrige	autocorrige		
auto-suportado(a)	autosuportado(a)	auto sustentad(o)(a)	autosustentado(a)
auto-transformador	autotransformador	auto- transformadores	autotransformadores
baixa tensão baixa-tensão	baixatensão	baixa e alta tensão	baixatensão e altatensão
baixa voltagem baixa-voltagem	baixavoltagem	baixíssima tensão baixíssima-tensão	baixíssimatensão
baixa potência	baixa potência	bi-volt	bivolt
chumbo-ácido	chumboácido	co-axialmente	coaxialmente
C. A. corrente alternada	correntealternada	contra-eletrodo	contraeletrodo
contra-porca	contraporca	corpo-suporte	corposuporte
curto-circuito	curtocircuito		
descarga gasosa	descargagasosa	descarga de gás	descargadegás
diodo-capacitor	diodocapacitor		
eletro-depositar	eletrodepositar	eletro-magnético	eletromagnético

Tabela 32B – Termos Compostos Modificados e com Erros Ortográficos

Termo Original	Termo Modificado	Termo Original	Termo Modificado
eletro-eletrônico(s)	eletroeletrônico(s)	eletro-eletrônica(s)	eletroeletrônica(s)
eletro-mecânico	eletromecânico	eletro-ótica	eletroótica
em-linha	emlinha		
fase/terra	faseterra	fibra-ótica fibra-óptica	fibraótica
fio-terra	fioterra	físico químicas físico-químicas	físicoquímicas
foto-acoplador	fotoacoplador	foto detetor	fotodetetor
foto diodo	fotodiodo	foto sensível	fotosensível
infra-vermelho infra vermelho	infravermelho	inter ligante	interligante
in-situ	insitu		
justa-posto	justaposto		
lâmpada de descarga	lâmpadadedescarga		
lâmpada de descarga de baixa pressão	lâmpadadedescarga debaixapressão	lâmpada de descarga de alta pressão	lâmpadadedescargadealtapressão
orelha de gato	orelhadegato	onda completa	ondacompleta
pára-brisas	parabrisas	pára-raios; para-raios pára-raio; para-raio	pararaios
passa-baixo	passabaixo	pisca-pisca	piscapisca
ponte completa	pontecompleta	onda completa	ondacompleta
poli-transformador	politransformador	porta-amostra	portaamostra
porta-catodo	portacatodo	porta-fusíveis	portafusíveis
porta-lâmpada	portalâmpada	porta-resistência	portaresistência
porta-substrato	portasubstrato	pré-pagas	prépagas (exceção pré)

Tabela 32C – Termos Compostos Modificados e com Erros Ortográficos

Termo Original	Termo Modificado	Termo Original	Termo Modificado
radio-frequência	radiofrequência	raios-X; raio-X	raiosX
retro-alimentar	retroalimentar		
sobre-cargas	sobrecargas	sobre-tensões	sobretensões
sub-tensões	subtensões	sub-pré	subpre
tensão alternada	tensãoalternada	tensão contínua	tensãocontínua
termo magnético	termomagnético	terras-raras	terrasraras
UV; ultra-violeta ultra violeta	ultravioleta	ultra-alto	ultraalto
vapor de sódio	vapordesódio	vapor metálico	vapormetálico
vapor de mercúrio	vapordemercúrio	vapor de mercúrio, metálico e sódio	vapordemercúrio vapormetálico vapordesódio
vidro-cerâmico	vidrocerâmico	vice-versa	viceversa
voltagem baixa	baixavoltagem		

12.0
Apêndice 4
Regras Usadas no Algoritmo de Stemização Modificado *StemmerPortuguese*

Tabela 33- Regras de Redução do Plural **[A]**				
Começar do maior para o menor sufixo e somente uma regra dentro do passo pode ser aplicada.				
Sufixo a Remover	**Tamanho Mínimo do *Stem* (Radical)**	**Tamanho Mínimo Palavra Original**	**Substituição**	**Exemplos**
-ões	3	6	-ão	balões → balão
-ães	1	4	-ão	capitães → capitão
-ais	1	4	-al	normais → normal animais → animal
-eis	2	5	-el	amáveis→ amável possíveis→possível
-éis	2	5	-el	papéis → papel
-óis -ois	2	5	-ol	lençóis → lençol caracóis→caracol
-les	2	5	-l	males → mal
-res	2	5	-r	mares→mar/doutores → doutor
-zes	2	5	-z	felizes→feliz /luzes→luz
-ses	2	5	-s	gases → gás
-is	2	4	-il	barris → barril
-ns	2	4	-m	bons → bom
-s	2	3	-	casas → casa

Tabela 34 - Regras de Redução do Feminino (a) **[B]**				
Começar do maior para o menor sufixo e somente uma regra dentro do passo pode ser aplicada.				
Sufixo a Remover	**Tamanho Mínimo do *Stem***	**Tamanho Mínimo Palavra Original**	**Substituição**	**Exemplo**
-íaca -iaca	3	7	-íaco	maníaca → maníaco
-inha	3	7	-inho	sozinha → sozinho velhinha → velhinho
-eira	3	7	-eiro	primeira → primeiro
-ona	3	6	-ão	chefona → chefão
-ora	3	6	-or	professora → professor vendedora → vendedor
-esa	3	6	-ês	inglesa → inglês francesa → francês
-osa	3	6	-oso	famosa → famoso bondosa → bondoso
-ica	3	6	-ico	prática → prático médica → médico
-ada	3	6	-ado	cansada → cansado
-ida	3	6	-ido	mantida → mantido corrida → corrido
-ída	3	6	-ido	
-ima	2	5	-imo	prima → primo
-iva	3	6	-ivo	passiva → passivo
-na	4	6	-no	americana → americano menina → menino chilena → chileno
-ã	2	3	-ão	vilã → vilão
-a	3	4	-	água → agu ;baixa →baix

Tabela 35 - Regras de Redução do Advérbio] **[D]**			
Sufixo a Remover	**Tamanho Mínimo do Stem**	**Substituição**	**Exemplo**
-mente	4	-	felizmente → feliz

Tabela 36 - Regras de Redução do Aumentativo/Diminutivo **[C]**				
Começar do maior para o menor sufixo e somente uma regra dentro do passo pode ser aplicada.				
Sufixo a Remover	**Tamanho Mínimo do Stem**	**Tamanho Mínimo Palavra Original**	**Substituição**	**Exemplo**
-abilíssimo -abilissimo	2		-a	amabilíssimo → ama
-díssimo -dissimo	5		-d	cansadíssimo → cansad
-íssimo -issimo	3		-	fortíssimo → fort
-érrimo -errimo	4		-	chiquérrimo → chiqu
-quinho	4		-c	maluquinho → maluc
-adinho	3		-ad	cansadinho → cansad
-ésimo -esimo	3		-	
-zinho	2		-	pezinho → pe
-uinho	4		-	amiguinho → amig
-alhão -alhao	4		-	grandalhão → grand
-arraz	4		-	pratarraz → prat
-inho	3		-	carrinho → carr
-adão -adao	3		-	casadão → cãs
-ázio -azio	3		-	corpázio → corp
-arra	3		-	bocarra → boc
-uça -uca	4		-	dentuça → dent
-aço -aco	3		-	ricaço → ric
-zão -zao	2		-	calorzão → calor
Exceção: ão	3		-	meninão → menin

Tabela 37A - Regras de Redução do Nome **[E]**

Começar do maior sufixo para o menor sufixo e somente uma regra dentro do passo pode ser aplicada.

Sufixo a Remover	**Tamanho Mínimo do Stem**	**Substituição**	**Exemplo**
1) -encialista	4	-	existencialista → exist
2) -abilidade	5	-	comparabilidade → compar
3) -icionista	4	-	abolicionista → abol
4) -cionista	5	-	intervencionista → interven
5) -iamento	4	-	gerenciamento → gerenc
6) –alizado	4	-	comercializado→comerci
7) –atizado	4	-	traumatizado→traum
8) –ividade	5	-	produtividade → produt
9) –alista	5	-	minimalista → minim
10) -amento	3	-	monitoramento→ monit
11) -imento	3	-	nascimento → nasc
12) -atória 13) -atoria	5	-	obrigatória → obrig
14) -edouro	3	-	bebedouro → beb abatedouro → abat
15) -encial	5	-	existencial → exist
16) -ástico 17) -astico	4	-	bombástico → bomb
18) -queiro	3	-c	fofoqueiro → fofoc
19) -alização	5	-	comercialização → comerci
20) –izado	5	-	alfabetizado → alfabet
21) –ativo	4	-	associativo → associ
22) –ional	4	-	profissional → profiss
23) -ência 24) –encia	3	-	dependência → depend referência → refer
25) -ância 26) –ancia	4	-	instância → inst repugnância → repugn
27) –quice	4	-c	maluquice → maluc

Tabela 37B -Regras de Redução do Nome **[E]**			
Sufixo a Remover	**Tamanho Mínimo do Stem**	**Substituição**	**Exemplos**
28) -ático 29) –atiço	3	-	problemático → problem
30) –idade	5	-	profundidade → profund
31) –agem	3	-	contagem → cont
32) -ização	5	-	concretização → concret
33) -ário 34) –ario	3	-	anedotário → anedot
35) -ério 36) –erio	6	-	ministério → minist
37) –tivo	4	-	contraceptivo→contracep
38) –ador	3	-	ralador → ral
39) –edor	3	-	entendedor → entend
40) –idor	4	-	cumpridor → cumpr
41) –eiro	3	-	brasileiro → brasil
42) –ante	2	-	ocupante → ocup
43) –esco	4	-	parentesco → parent
44) –ismo	3	-	consumismo → consum
45) –oria	4	-	aposentadoria→aposentad
46) –ista	3	-	artista → art
47) –inal	3	-	criminal → crim
48) –ente	4	-	decorrente → decorr
49) -ável 50) –avel	2	-	amável → am
51) -íaco 52) –iaco	3	-	demoníaco → demon
53) -ível 54) –ivel	4	-	combustível → combust
55) –eza	3	-	beleza → bel
56) –ivo	4	-	esportivo → esport
57) –ado	2	-	abalado → abal
58) –ido	3	-	impedido → imped
59) –oso	3	-	bondoso → bond gostoso → gost
60) –iço	4	-	polêmico → polêm
61) –ano	4	-	americano → americ

Tabela 37C -Regras de Redução do Nome **[E]**			
Sufixo a Remover	**Tamanho Mínimo do Stem**	**Substituição**	**Exemplo**
62) –ice	4	-	chatice → chat meninice → menin
63) –ura	4	-	cobertura → cobert
64) –ual	3	-	consensual → consens
65) –ial	3	-	mundial → mund
66) –ação	3	-	alegação → aleg
67) –ição	3	-	abolição → abol
68) –ês 69) –es	4	-	chinês → chin
70) –ez	4	-	rigidez → rigid
71) –or	2	-	produtor → produt
72) –al	4	-	experimental →experiment

Tabela 38A - Regras de Redução do Verbo **[F]**			
Deve ser removido do maior para o menor sufixo e somente uma regra dentro do passo pode ser aplicada.			
Sufixo a Remover	**Tamanho Mínimo do Stem**	**Substituição**	**Exemplo**
-aríamo -ariamo	2	-	cantaríamo→cant
-ássemo -assemo	2	-	cantássemo→cant
-eríamo -eriamo	2	-	beberíamo→beb
-êssemo -essemo	2	-	bebêssemo→beb
-iríamo -iriamo	3	-	partiríamo→part
-íssemo -issemo	3	-	partíssemo→part
-áramo -aramo	2	-	cantáramo→cant
-aremo	2	-	cantaremo→cant
-ariam	2	-	cantariam→cant
-aríei -ariei	2	-	cantaríei→cant
-ássei -assei	2	-	cantássei→cant
-assem	2	-	cantassem→cant
-ávamo -avamo	2	-	cantávamo→cant

Tabela 38B - Regras de Redução do Verbo **[F]**			
Deve ser removido do maior para o menor sufixo e somente uma regra dentro do passo pode ser aplicada.			
Sufixo a Remover	**Tamanho Mínimo do Stem**	**Substituição**	**Exemplo**
-êramo -eramo	3	-	bebêramo→beb
-eremo	3	-	beberemo→beb
-eriam	3	-	beberiam→beb
-eríei -eriei	3	-	beberíei→beb
-êssei -essei	3	-	bebêssei→beb
-essem	3	-	bebessem→beb
-íramo -iramo	3	-	partíramo→part
-iremo	3	-	partiremo→part
-iriam	3	-	partiriam→part
-iríei -iriei	3	-	partiríei→part
-íssei -issei	3	-	partíssei→part
-issem	3	-	partissem→part
-árei -arei	2	-	cantárei→cant
-ando	2	-	cantando→cant
-endo	3	-	bebendo→beb
-indo	3	-	partindo→part
-eria	3	-	beberia→beb
-ermo	3	-	bebermo→beb
-esse	3	-	bebesse→beb
-este	3	-	bebeste→beb
-íamo -iamo	3	-	bebíamo→beb
-iram	3	-	partiram→part
-íram	3	-	concluíram→conclu
-irde	2	-	partirde→part
-irei	3	-	partirei→part
-irem	3	-	partirem→part
-iria	3	-	partiria→part
-irmo	3	-	partirmo→part
-isse	3	-	partisse→part

Tabela 38C - Regras de Redução do Verbo **[F]**			
(Deve ser removido do maior para o menor sufixo e somente uma regra dentro do passo pode ser aplicada).			
Sufixo a Remover	**Tamanho Mínimo do Stem**	**Substituição**	**Exemplo**
-iste	4	-	partiste→part
-ondo	3	-	propondo→prop
-aram	2	-	cantaram→cant
-arde	2	-	cantarde→cant
-arei	2	-	cantarei→cant
-arem	2	-	cantarem→cant
-aria	2	-	cantaria→cant
-armo	2	-	cantarmo → cant
-asse	2	-	cantasse → cant
-aste	2	-	cantaste → cant
-avam	2	-	cantavam → cant
-ávei -avei	2	-	cantávei → cant
-eram	3	-	beberam → beb
-erde	3	-	beberde → beb
-erei	3	-	beberei → beb
-êrei	3	-	
-erem	3	-	beberem → beb
-amo	2	-	cantamo→cant
-ara	2	-	cantara→cant
-ará	2	-	cantará→cant
-are	2	-	cantare→cant
-ava	2	-	cantava→cant
-emo	2	-	cantemo→cant
-era	3	-	bebera→beb
-erá	3	-	beberá→beb
-ere	3	-	bebere→beb
-iam	3	-	bebiam→beb
-íei -iei	3	-	bebíei→beb
-imo	3	-	partimo→part
-ira	3	-	partira→part
-irá	3	-	partirá→part
-ire	3	-	partire→part

Tabela 38D - Regras de Redução do Verbo **[F]**			
Deve ser removido do maior para o menor sufixo e somente uma regra dentro do passo pode ser aplicada.			
Sufixo a Remover	**Tamanho Mínimo do Stem**	**Substituição**	**Exemplo**
-ire	3	-	partire→part
-omo	3	-	compomo → comp
-ear	4	-	barbear → barb
-uei	3	-	cheguei → cheg
-ai	2	-	cantai → cant
-am	2	-	cantam → cant
-ar	2	-	cantar → cant
-ei	3	-	cantei → cant
-em	2	-	cantem → cant
-er	2	-	beber → beb
-eu	3	-	bebeu → beb
-ia	3	-	bebia → beb
-ir	3	-	partir → part
-iu	3	-	partiu → part
-ou	3	-	chegou → cheg
-i	3	-	bebi → beb

Tabela 39 - Regras de Redução das Vogais Finais **[G]**			
Deve ser removido do maior para o menor sufixo e somente uma regra dentro do passo pode ser aplicada.			
Sufixo a Remover	**Tamanho Mínimo do Stem**	**Substituição**	**Exemplo**
-ão	3	5	invenção → invenç
-e	3	4	grande → grand maxixe → maxix
-a	3	4	relva → relv
-o	3	4	menino → menin pequeno → pequen

Tabela 40 - Regras de Redução do Acento **[H]**			
Sufixo a Remover	**Tamanho Mínimo do Stem**	**Substituição**	
-á	-	-a	
-é	-	-e	
-í	-	-i	
-ó	-	-o	
-ú	-	-u	
-ã	-	-a	
-õ	-	-o	
-â	-	-a	
-ê	-	-e	
-î	-	-i	
-ô	-	-o	
-û	-	-u	
-ä	-	-a	
-ë	-	-e	
-ï	-	-i	
-ö	-	-o	
-ü	-	-u	
-à	-	-a	
-è	-	-e	
-ì	-	-i	
-ò	-	-o	
-ù	-	-u	
-ç	-	-c	

13.0
APÊNDICE 5
Quantidade de Documentos Discriminados por Etapa de Treinamento e Teste

Tabela 41 – Primeira e Segunda Modalidades - Divisão da Base de Dados em Treinamento e Teste para as categorias H05B e H02G.

Subclasse (IPC)	**Grupo (IPC)**	**Quantidade Total**		
		Treina-mento	**Teste**	**Total**
H05B AQUECIMENTO **H05B(A)** **Total: 294 documentos**	H05B 1/00 H05B3/00 H05B6/00 H05B7/00 H05B11/00	284	Primeira Modalidade 10 + 69 (trein.) ---------- Segunda Modalidade 10	Primeira Modalidade 294 + 69 (trein.) ---------- Segunda Modalidade 294
H05B ILUMINAÇÃO **H05B(I)** **Total: 205 documentos**	H05B31/00 H05B 33/00 H05B 35/00 H05B 37/00 H05B 39/00 H05B 41/00 H05B 43/00	195	Primeira Modalidade 10 + 68 (trein.) ---------- Segunda Modalidade 10	Primeira Modalidade 205 + 68 (trein.) ---------- Segunda Modalidade 205
H02G CABOS E LINHAS ELÉTRICAS **Total: 500 documentos**	H02G 1/00 H02G 3/00 H02G 5/00 H02G 7/00 H02G 9/00 H02G 11/00 H02G 13/00 H02G 15/00	333	Primeira e Segunda Modalidades: 167	Primeira e Segunda Modalidades: 500

Tabela 42 – Primeira e Segunda Modalidades - Divisão da Base de Dados em Treinamento e Teste para as Categorias H01F e H02M.

Subclasse (IPC)	**Grupo (IPC)**	**Quantidade Total**		
		Treina-mento	**Teste**	**Total**
H01F MAGNETOS E INDUTÂNCIAS **Total: 434 documentos**	H01F 1/00 H01F 3/00 H01F 5/00 H01F 6/00 H01F 7/00 H01F 10/00 H01F 13/00 H01F 15/00 H01F 17/00 H01F 19/00 H01F 21/00 H01F 27/00 H01F 29/00 H01F 30/00 H01F 35/00 H01F 36/00 H01F 37/00 H01F 38/00 H01F 39/00 H01F 40/00 H01F 41/00	300	Primeira e Segunda Modalidades: 134	Primeira e Segunda Modalidades: 434
H02M CONVERSÃO DE ENERGIA **Total: 274 documentos**	H02M 1/00 H02M 3/00 H02M 5/00 H02M 7/00 H02M 9/00 H02M 11/00	264	Primeira Modalidade 10 + 67(trein.) ---------- Segunda Modalidade 10	Primeira Modalidade 274+67 (trein.) ---------- Segunda Modalidade 274

Tabela 43 – Primeira e Segunda Modalidades - Divisão da Base de Dados em Treinamento e Teste para a categoria H01J.

Subclasse (IPC)	**Grupo (IPC)**	**Quantidade Total**		
		Treina-mento	**Teste**	**Total**
H01J TUBOS DE DESCARGA **Total: 237 documentos**	H01J 1/00	267	Primeira Modalidade 10 + 66(trein.) ---------- Segunda Modalidade 10	Primeira Modalidade 278+66(trein.) ---------- Segunda Modalidade 278 sendo 237 (tubos de descarga) e 41 (lâmpadas de descarga)
	H01J 3/00			
	H01J 5/00			
	H01J 7/00			
	H01J 9/00			
	H01J 11/00			
	H01J 13/00			
	H01J 15/00			
	H01J 17/00			
	H01J 19/00			
	H01J 21/00			
	H01J 23/00			
	H01J 25/00			
	H01J 27/00			
	H01J 29/00			
	H01J 31/00			
	H01J 33/00			
	H01J 35/00			
	H01J 37/00			
	H01J 40/00			
	H01J 41/00			
	H01J 43/00			
	H01J 45/00			
	H01J 47/00			
	H01J 49/00			
H01J LÂMPADAS DE DESCARGA **Total: 41**	H01J 61/00			
	H01J 63/00			
	H01J 65/00			

Tabela 44 – Primeira e Segunda Modalidades - Divisão da Base de Dados em Treinamento e Teste para a categoria H02K

Subclasse (IPC)	**Grupo (IPC)**	**Quantidade Total**		
		Treina-mento	**Teste**	**Total**
H02K MÁQUINAS ÉTRICAS **Total: 500 cumentos**	H02K 1/00	333	Primeira e Segunda Modalidades: 167	Primeira e Segunda Modalidades: 500
	H02K 3/00			
	H02K 5/00			
	H02K 7/00			
	H02K 9/00			
	H02K11/00			
	H02K 13/00			
	H02K15/00			
	H02K 16/00			
	H02K 17/00			
	H02K 19/00			
	H02K 21/00			
	H02K 23/00			
	H02K 24/00			
	H02K 25/00			
	H02K 26/00			
	H02K 27/00			
	H02K 29/00			
	H02K 31/00			
	H02K 33/00			
	H02K 35/00			
	H02K 37/00			
	H02K 39/00			
	H02K 41/00			
	H02K 44/00			
	H02K 47/00			
	H02K 49/00			
	H02K 51/00			
	H02K 53/00			
	H02K 55/00			
	H02K 57/00			

Tabela 45 – Primeira e Segunda Modalidades - Divisão da Base de Dados em Treinamento e Teste para a categoria A47B.

Subclasse (IPC)	**Grupo (IPC)**		**Quantidade Total**		
			Treina-mento	**Teste**	**Total**
A47B MÓVEIS; ARTIGOS OU APARELHOS DOMÉSTI-COS; MOINHOS DE CAFÉ; MOINHOS DE ESPECIARIA; ASPIRADO-RES EM GERAL. **Total: 447 documentos**	A47B1/00	A47B63/00	300	Primeira e Segunda Modalidades 147	Primeira e Segunda Modalida-des: 447
	A47B3/00	A47B65/00			
	A47B5/00	A47B67/00			
	A47B7/00	A47B69/00			
	A47B9/00	A47B71/00			
	A47B11/00	A47B73/00			
	A47B13/00	A47B75/00			
	A47B17/00	A47B77/00			
	A47B19/00	A47B79/00			
	A47B21/00	A47B81/00			
	A47B23/00	A47B83/00			
	A47B25/00	A47B85/00			
	A47B27/00	A47B87/00			
	A47B29/00	A47B88/00			
	A47B31/00	A47B91/00			
	A47B33/00	A47B95/00			
	A47B35/00	A47B96/00			
	A47B37/00	A47B97/00			
	A47B39/00				
	A47B41/00				
	A47B43/00				
	A47B45/00				
	A47B46/00				
	A47B47/00				
	A47B49/00				
	A47B51/00				
	A47B53/00				
	A47B55/00				
	A47B57/00				
	A47B61/00				

Tabela 46 – Primeira e Segunda Modalidades - Divisão da Base de Dados em Treinamento e Teste para as categorias A47C e H02P

Subclasse (IPC)	**Grupo (IPC)**	**Quantidade Total**		
		Treinamento	**Teste**	**Total**
A47C CADEIRAS **Total: 331 documentos**	A47C1/00	300	Primeira Modalidade 31+ 44(trein.) ---------- Segunda Modalidade 31	Primeira Modalidade 331+ 44(trein.) ---------- Segunda Modalidade 331
	A47C3/00			
	A47C4/00			
	A47C5/00			
	A47C7/00			
	A47C9/00			
	A47C11/00			
	A47C12/00			
	A47C13/00			
	A47C15/00			
	A47C16/00			
	A47C17/00			
	A47C19/00			
	A47C20/00			
	A47C21/00			
	A47C23/00			
	A47C25/00			
	A47C27/00			
	A47C29/00			
	A47C31/00			
H02P CONTROLE OU REGULAÇÃO **Total: 307 documentos**	H02P 1/00	297	Primeira Modalidade 10+ 69(trein.) ---------- Segunda Modalidade 10	Primeira Modalidade 307+ 69(trein.) ---------- Segunda Modalidade 307
	H02P 3/00			
	H02P 5/00			
	H02P 6/00			
	H02P 7/00			
	H02P 8/00			
	H02P 9/00			
	H02P 11/00			
	H02P 13/00			
	H02P 15/00			
	H02P 17/00			
	H02P 19/00			
	H02P 21/00			
	H02P 23/00			
	H02P 25/00			
	H02P 27/00			

Tabela 47 – Primeira e Segunda Modalidades - Divisão da Base de Dados em Treinamento e Teste para a categoria H02B.

Subclasse (IPC)	**Grupo (IPC)**	**Quantidade Total**		
		Treina-mento	**Teste**	**Total**
H02B PAINÉIS **Total: 293 documentos**	H02B 1/00 H02B 3/00 H02B 5/00 H02B 7/00 H02B11/00 H02B13/00 H02B15/00	283	Primeira Modalidade 10 + 68 (trein.) ---------- Segunda Modalidade 10	Primeira Modalidade 293 + 68 (trein.) ---------- Segunda Modalidade 293

14.0
Apêndice 6
Resultados do Algoritmo do Método 01

Tabela 48 - Método 01 - Prognóstico de Topo - StemerMétodo01 - Método RankCos – Resolução 1

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.559184	0.931973	137	245	147
2	H05BA	0.712644	0.784810	62	87	79
3	H02G	0.741379	0.514970	86	116	167
4	A47C	0.745098	0.506667	38	51	75
5	H02P	0.694118	0.746835	59	85	79
6	H02M	0.686047	0.766234	59	86	77
7	H05BI	0.72	0.692308	54	75	78
8	H01F	0.779221	0.447761	60	77	134
9	H02K	0.7	0.712575	119	170	167
10	H02B	0.592105	0.576923	45	76	78
11	H01J	0.764045	0.894737	68	89	76
Média:		0.6994	0.6887	Total: 787	1157	1157
Medida de F1:		0.6940				

Tabela 49 - Método 01 - Prognóstico de Topo - StemerMétodo01 - Método RelevânciaCos – Resolução 1

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.699482	0.918367	135	193	147
2	H05BA	0.742574	0.949367	75	101	79
3	H02G	0.818966	0.568862	95	116	167
4	A47C	0.833333	0.733333	55	66	75
5	H02P	0.691589	0.936709	74	107	79
6	H02M	0.765957	0.935065	72	94	77
7	H05BI	0.802326	0.884615	69	86	78
8	H01F	0.838710	0.388060	52	62	134
9	H02K	0.767742	0.712575	119	155	167
10	H02B	0.680851	0.820513	64	94	78
11	H01J	0.855422	0.934211	71	83	76
Média:		0.7725	0.7983	Total: 881	1157	1157
Medida de F1:		0.7852				

Tabela 50 - Método 01 - Prognóstico de Topo - StemerMétodo01 - Método RankABS – Resolução 1

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.665116	0.972889	143	215	147
2	H05BA	0.648936	0.7625	61	94	79
3	H02G	0.729508	0.532934	89	122	167
4	A47C	0.819672	0.625	50	61	75
5	H02P	0.533333	0.5	40	75	79
6	H02M	0.733333	0.6875	55	75	77
7	H05BI	0.675	0.675000	54	80	78
8	H01F	0.646341	0.395522	53	82	134
9	H02K	0.658960	0.682635	114	173	167
10	H02B	0.632911	0.625	50	79	78
11	H01J	0.623762	0.787500	63	101	76
Média:		0.6697	0.6588	Total: 772	1157	1157
Medida de F1:		0.6642				

Tabela 51 - Método 01 - Prognóstico de Topo - StemerMétodo01 - Método RelevânciaABS – Resolução 1

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.738462	0.979592	144	195	147
2	H05BA	0.654206	0.875	70	107	79
3	H02G	0.746032	0.562874	94	126	167
4	A47C	0.875	0.7	56	64	75
5	H02P	0.578947	0.6875	55	95	79
6	H02M	0.789474	0.75	60	76	77
7	H05BI	0.701149	0.7625	61	87	78
8	H01F	0.769231	0.373134	50	65	134
9	H02K	0.680982	0.664671	111	163	167
10	H02B	0.636364	0.7	56	88	78
11	H01J	0.703297	0.800000	64	91	76
Média:		0.7157	0.7141	Total: 821	1157	1157
Medida de F1:		0.7149				

Tabela 52 - Método 01 - Três Prognósticos de Topo - StemerMétodo01 - Método RankCos – Resolução 1

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.986395	145	-	147
2	H05BA	-	0.974684	77	-	79
3	H02G	-	0.826347	138	-	167
4	A47C	-	0.973333	73	-	75
5	H02P	-	1	79	-	79
6	H02M	-	0.935065	72	-	77
7	H05BI	-	0.961538	75	-	78
8	H01F	-	0.671642	90	-	134
9	H02K	-	0.898204	150	-	167
10	H02B	-	0.846154	66	-	78
11	H01J	-	0.973684	74	-	76
Média:			0.9134	Total: 1039		1157

Tabela 53 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RelevânciaCos – Resolução 1

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.993197	146	-	147
2	H05BA	-	1	79	-	79
3	H02G	-	0.826347	138	-	167
4	A47C	-	0.986666	74	-	75
5	H02P	-	1	79	-	79
6	H02M	-	0.974026	75	-	77
7	H05BI	-	1	78	-	78
8	H01F	-	0.656716	88	-	134
9	H02K	-	0.892216	149	-	167
10	H02B	-	0.948718	74	-	78
11	H01J	-	1	76	-	76
Média:			0.9344	Total: 986		1157

Tabela 54 - Método 01 - Três Prognósticos de Topo - StemerMétodo01 - Método RankABS – Resolução 1

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.993197	146	-	147
2	H05BA	-	0.987342	78	-	79
3	H02G	-	0.844311	141	-	167
4	A47C	-	0.96	72	-	75
5	H02P	-	0.949367	75	-	79
6	H02M	-	0.883117	68	-	77
7	H05BI	-	0.961539	75	-	78
8	H01F	-	0.671642	90	-	134
9	H02K	-	0.844311	141	-	167
10	H02B	-	0.910256	71	-	78
11	H01J	-	0.973684	74	-	76
Média:			0.9072	Total: 1031		1157

Tabela 55 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RelevânciaABS – Resolução 1

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.993197	146	-	147
2	H05BA	-	1	79	-	79
3	H02G	-	0.886228	148	-	167
4	A47C	-	0.946667	71	-	75
5	H02P	-	0.974684	77	-	79
6	H02M	-	0.961039	74	-	77
7	H05BI	-	1	78	-	78
8	H01F	-	0.649254	87	-	134
9	H02K	-	0.868263	145	-	167
10	H02B	-	0.923077	72	-	78
11	H01J	-	0.986842	75	-	76
Média:			0.9263	Total: 1052		1157

Tabela 56 - Método 01 - Prognóstico de Topo - StemerMétodo01 - Método RankCos – Resolução 2

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.590517	0.931973	137	232	147
2	H05BA	0.704545	0.784810	62	88	79
3	H02G	0.731092	0.520958	87	119	167
4	A47C	0.820000	0.546667	41	50	75
5	H02P	0.625	0.759494	60	96	79
6	H02M	0.686747	0.740260	57	84	77
7	H05BI	0.746667	0.717949	56	75	78
8	H01F	0.760563	0.402985	54	71	134
9	H02K	0.701149	0.730539	122	174	167
10	H02B	0.564103	0.564103	44	78	78
11	H01J	0.722222	0.855263	65	90	76
Média:		0.6957	0.6868	Total: 785	1157	1157
Medida de F1:		0.6912				

Tabela 57 - Método 01 - Prognóstico de Topo - StemerMétodo01 - Método RelevânciaCos – Resolução 2

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.693878	0.925170	136	196	147
2	H05BA	0.75	0.9375	75	100	79
3	H02G	0.778689	0.568862	95	122	167
4	A47C	0.861538	0.70	56	65	75
5	H02P	0.682692	0.8875	71	104	79
6	H02M	0.766667	0.862500	69	90	77
7	H05BI	0.797619	0.837500	67	84	78
8	H01F	0.870968	0.402985	54	62	134
9	H02K	0.783440	0.736527	123	157	167
10	H02B	0.673684	0.800000	64	95	78
11	H01J	0.865854	0.8875	71	82	76
Média:		0.7750	0.7769	Total: 881	1157	1157
Medida de F1:		0.7759				

Tabela 58 - Método 01 – Prognóstico de Topo - StemerMétodo01 - Método RankABS – Resolução 2

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.702439	0.979592	144	205	147
2	H05BA	0.690722	0.837500	67	97	79
3	H02G	0.748092	0.586826	98	131	167
4	A47C	0.901639	0.6875	55	61	75
5	H02P	0.548780	0.5625	45	82	79
6	H02M	0.691358	0.700000	56	81	77
7	H05BI	0.743243	0.6875	55	74	78
8	H01F	0.742857	0.388060	52	70	134
9	H02K	0.672316	0.712575	119	177	167
10	H02B	0.638554	0.662500	53	83	78
11	H01J	0.65625	0.787500	63	96	76
Média:		0.7033	0.6902	Total: 832	1157	1157
Medida de F1:		0.6967				

Tabela 59 - Método 01 – Prognóstico de Topo - StemerMétodo01 - Método RelevânciaABS – Resolução 2

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.753927	0.979592	144	191	147
2	H05BA	0.676190	0.898734	71	105	79
3	H02G	0.759690	0.586826	98	129	167
4	A47C	0.933333	0.746667	56	60	75
5	H02P	0.586957	0.683544	54	92	79
6	H02M	0.776316	0.766234	59	76	77
7	H05BI	0.726190	0.782051	61	84	78
8	H01F	0.765625	0.365672	49	64	134
9	H02K	0.677778	0.730539	122	180	167
10	H02B	0.627907	0.692308	54	86	78
11	H01J	0.711111	0.842105	64	90	76
Média:		0.7268	0.7340	Total: 832	1157	1157
Medida de F1:		0.7304				

Tabela 60.- Método 01 – Três Prognósticos de Topo -StemerMétodo01 - Método RankCos – Resolução 2

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.986395	145	-	147
2	H05BA	-	0.987342	78	-	79
3	H02G	-	0.862275	144	-	167
4	A47C	-	0.973333	73	-	75
5	H02P	-	1	79	-	79
6	H02M	-	0.935065	72	-	77
7	H05BI	-	0.948718	74	-	78
8	H01F	-	0.656716	88	-	134
9	H02K	-	0.892216	149	-	167
10	H02B	-	0.871795	68	-	78
11	H01J	-	0.960526	73	-	76
Média:			0.9019	Total: 1043		1157

Tabela 61 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RelevânciaCos – Resolução 2

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.993197	146	-	147
2	H05BA	-	1	79	-	79
3	H02G	-	0.862275	144	-	167
4	A47C	-	0.986667	74	-	75
5	H02P	-	0.987342	78	-	79
6	H02M	-	0.974026	75	-	77
7	H05BI	-	1	78	-	78
8	H01F	-	0.656716	88	-	134
9	H02K	-	0.910180	152	-	167
10	H02B	-	0.923077	72	-	78
11	H01J	-	1	76	-	76
Média:			0.9358	Total: 1062		1157

Tabela 62 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RankABS – Resolução 2

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.993197	146	-	147
2	H05BA	-	1	79	-	79
3	H02G	-	0.862275	144	-	167
4	A47C	-	0.946667	71	-	75
5	H02P	-	0.962025	76	-	79
6	H02M	-	0.909091	70	-	77
7	H05BI	-	0.974359	76	-	78
8	H01F	-	0.604478	81	-	134
9	H02K	-	0.886228	148	-	167
10	H02B	-	0.884615	69	-	78
11	H01J	-	0.947368	72	-	76
Média:			0.9064	Total: 1032		1157

Tabela 63 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RelevânciaABS – Resolução 2

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	1	147	-	147
2	H05BA	-	1	79	-	79
3	H02G	-	0.862275	144	-	167
4	A47C	-	0.973333	73	-	75
5	H02P	-	0.974684	77	-	79
6	H02M	-	0.974026	75	-	77
7	H05BI	-	1	78	-	78
8	H01F	-	0.634328	85	-	134
9	H02K	-	0.892216	149	-	167
10	H02B	-	0.923077	72	-	78
11	H01J	-	0.986842	75	-	76
Média:			0.9292	Total: 1054		1157

Tabela 64 - Método 01 – Prognóstico de Topo - StemerMétodo01 - Método RankCos – Resolução 3

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.580508	0.931973	137	236	147
2	H05BA	0.638298	0.759494	60	94	79
3	H02G	0.788991	0.514970	86	109	167
4	A47C	0.844444	0.506667	38	45	75
5	H02P	0.613861	0.784810	62	101	79
6	H02M	0.710843	0.766234	59	83	77
7	H05BI	0.727273	0.717949	56	77	78
8	H01F	0.790323	0.365672	49	62	134
9	H02K	0.690608	0.748503	125	181	167
10	H02B	0.666667	0.641026	50	75	78
11	H01J	0.670213	0.828947	63	94	76
Média:		0.7020	0.6878	Total: 785	1157	1157
Medida de F1:		0.6949				

Tabela 65 - Método 01 – Prognóstico de Topo – StemerMétodo01 - Método RelevânciaCos – Resolução 3

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.671642	0.918367	135	201	147
2	H05BA	0.701923	0.924051	73	104	79
3	H02G	0.831858	0.562874	94	113	167
4	A47C	0.875000	0.746667	56	64	75
5	H02P	0.645455	0.898734	71	110	79
6	H02M	0.804598	0.909091	70	87	77
7	H05BI	0.795181	0.846154	66	83	78
8	H01F	0.872727	0.358209	48	55	134
9	H02K	0.756098	0.742515	124	164	167
10	H02B	0.714286	0.833333	65	91	78
11	H01J	0.800000	0.894737	68	85	76
Média:		0.7699	0.7850	Total: 870	1157	1157
Medida de F1:		0.7774				

Tabela 66- Método 01 – Prognóstico de Topo - StemerMétodo01 - Método RankABS – Resolução 3

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.683962	0.986395	145	212	147
2	H05BA	0.637255	0.822785	65	102	79
3	H02G	0.757813	0.580838	97	128	167
4	A47C	1	0.693333	52	52	75
5	H02P	0.544304	0.544304	43	79	79
6	H02M	0.666667	0.753247	58	87	77
7	H05BI	0.740260	0.730769	57	77	78
8	H01F	0.794118	0.402985	54	68	134
9	H02K	0.664804	0.712575	119	179	167
10	H02B	0.644737	0.628205	49	76	78
11	H01J	0.628866	0.802632	61	97	76
Média:		0.7057	0.6962	Total: 800	1157	1157
Medida de F1:		0.7009				

Tabela 67 - Método 01 – Prognóstico de Topo - StemerMétodo01 - Método RelevânciaABS – Resolução 3

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.728643	0.986395	145	199	147
2	H05BA	0.639640	0.898734	71	111	79
3	H02G	0.763780	0.580838	97	127	167
4	A47C	1	0.760000	57	57	75
5	H02P	0.588889	0.670886	53	90	79
6	H02M	0.702381	0.766233	59	84	77
7	H05BI	0.75	0.807692	63	84	78
8	H01F	0.809524	0.380597	51	63	134
9	H02K	0.679775	0.724551	121	178	167
10	H02B	0.666667	0.666667	52	78	78
11	H01J	0.732558	0.828947	63	86	76
Média:		0.7329	0.7338	Total: 832	1157	1157
Medida de F1:		0.7333				

Tabela 68 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RankCos – Resolução 3

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.986395	145	-	147
2	H05BA	-	0.987342	78	-	79
3	H02G	-	0.910180	152	-	167
4	A47C	-	0.973333	73	-	75
5	H02P	-	0.987342	78	-	79
6	H02M	-	0.948052	73	-	77
7	H05BI	-	0.961538	75	-	78
8	H01F	-	0.671642	90	-	134
9	H02K	-	0.910180	152	-	167
10	H02B	-	0.871795	68	-	78
11	H01J	-	0.960526	73	-	76
Média:			0.9140	Total: 1057		1157

Tabela 69 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RelevânciaCos – Resolução 3

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.986395	145	-	147
2	H05BA	-	1	79	-	79
3	H02G	-	0.886228	148	-	167
4	A47C	-	0.986667	74	-	75
5	H02P	-	0.987342	78	-	79
6	H02M	-	0.974026	75	-	77
7	H05BI	-	1	78	-	78
8	H01F	-	0.694030	93	-	134
9	H02K	-	0.922156	154	-	167
10	H02B	-	0.935897	73	-	78
11	H01J	-	1	76	-	76
Média:			0.9430	Total: 1073		1157

Tabela 70 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RankABS – Resolução 3

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	1	147	-	147
2	H05BA	-	0.974684	77	-	79
3	H02G	-	0.856287	143	-	167
4	A47C	-	0.96	72	-	75
5	H02P	-	0.962025	76	-	79
6	H02M	-	0.922078	71	-	77
7	H05BI	-	0.974359	76	-	78
8	H01F	-	0.641791	86	-	134
9	H02K	-	0.904192	151	-	167
10	H02B	-	0.846154	66	-	78
11	H01J	-	0.947368	72	-	76
Média:			0.9081	Total: 1037		1157

Tabela 71 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RelevânciaABS – Resolução 3

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	1	147	-	147
2	H05BA	-	1	79	-	79
3	H02G	-	0.850299	142	-	167
4	A47C	-	0.946667	71	-	75
5	H02P	-	0.974684	77	-	79
6	H02M	-	0.948052	73	-	77
7	H05BI	-	1	78	-	78
8	H01F	-	0.634328	85	-	134
9	H02K	-	0.910180	152	-	167
10	H02B	-	0.858974	67	-	78
11	H01J	-	0.960526	73	-	76
Média:			0.9167	Total: 1044		1157

Tabela 72 - Método 01 – Prognóstico de Topo - StemerMétodo01 - Método RankCos – Resolução 4

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.577406	0.938776	138	239	147
2	H05BA	0.681319	0.784810	62	91	79
3	H02G	0.803571	0.538922	90	112	167
4	A47C	0.860465	0.493333	37	43	75
5	H02P	0.616162	0.772152	61	99	79
6	H02M	0.719512	0.766234	59	82	77
7	H05BI	0.75	0.730769	57	76	78
8	H01F	0.793651	0.373134	50	63	134
9	H02K	0.675532	0.760480	127	188	167
10	H02B	0.657534	0.615385	48	73	78
11	H01J	0.692308	0.828947	63	91	76
Média:		0.7116	0.6912	Total: 792	1157	1157
Medida de F1:		0.7013				

Tabela 73 - Método 01 – Prognóstico de Topo - StemerMétodo01 - Método RelevânciaCos – Resolução 4

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.665049	0.931973	137	206	147
2	H05BA	0.711538	0.936709	74	104	79
3	H02G	0.845455	0.556886	93	110	167
4	A47C	0.896552	0.693333	52	58	75
5	H02P	0.657407	0.898734	71	108	79
6	H02M	0.793103	0.896104	69	87	77
7	H05BI	0.795181	0.846154	66	83	78
8	H01F	0.859649	0.365672	49	57	134
9	H02K	0.739645	0.748503	125	169	167
10	H02B	0.706522	0.833333	65	92	78
11	H01J	0.819277	0.894737	68	83	76
Média:		0.7718	0.7820	Total: 869	1157	1157
Medida de F1:		0.7769				

Tabela 74 - Método 01 – Prognóstico de Topo - StemerMétodo01 - Método RankABS – Resolução 4

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.677725	0.972789	143	211	147
2	H05BA	0.633663	0.810127	64	101	79
3	H02G	0.732283	0.556886	93	127	167
4	A47C	0.964286	0.72	54	56	75
5	H02P	0.530120	0.556962	44	83	79
6	H02M	0.674419	0.753247	58	86	77
7	H05BI	0.727273	0.717949	56	77	78
8	H01F	0.777778	0.365672	49	63	134
9	H02K	0.659341	0.718563	120	182	167
10	H02B	0.653846	0.653846	51	78	78
11	H01J	0.645161	0.789474	60	93	76
Média:		0.6978	0.6923	Total: 792	1157	1157
Medida de F1:		0.6950				

Tabela 75 - Método 01 – Prognóstico de Topo - StemerMétodo01 - Método RelevânciaABS – Resolução 4

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.704434	0.972789	143	203	147
2	H05BA	0.625	0.886076	70	112	79
3	H02G	0.75	0.574850	96	128	167
4	A47C	0.982456	0.746667	56	57	75
5	H02P	0.586957	0.683544	54	92	79
6	H02M	0.701149	0.792208	61	87	77
7	H05BI	0.75	0.807692	63	84	78
8	H01F	0.8	0.358209	48	60	134
9	H02K	0.676136	0.712575	119	176	167
10	H02B	0.698630	0.653846	51	73	78
11	H01J	0.717647	0.802632	61	85	76
Média:		0.7266	0.7265	Total: 822	1157	1157
Medida de F1:		0.7265				

Tabela 76 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 – Método RankCos – Resolução 4

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.986395	145	-	147
2	H05BA	-	0.987342	78	-	79
3	H02G	-	0.892216	149	-	167
4	A47C	-	0.96	72	-	75
5	H02P	-	0.987342	78	-	79
6	H02M	-	0.961039	74	-	77
7	H05BI	-	1	78	-	78
8	H01F	-	0.649254	87	-	134
9	H02K	-	0.910180	152	-	167
10	H02B	-	0.871795	68	-	78
11	H01J	-	0.960526	73	-	76
Média:			0.9242	Total: 1054		1157

Tabela 77 - Método 01 – Três Prognósticos de Topo - StemerMétodo 01 - Método RelevânciaCos – Resolução 4

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.986395	145	-	147
2	H05BA	-	1	79	-	79
3	H02G	-	0.904192	151	-	167
4	A47C	-	0.986667	74	-	75
5	H02P	-	1	79	-	79
6	H02M	-	0.974026	75	-	77
7	H05BI	-	1	78	-	78
8	H01F	-	0.701493	94	-	134
9	H02K	-	0.910180	152	-	167
10	H02B	-	0.935897	73	-	78
11	H01J	-	1	76	-	76
Média:			0.9454	Total: 1076		1157

Tabela 78 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RankABS – Resolução 4

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	1	147	-	147
2	H05BA	-	0.987342	78	-	79
3	H02G	-	0.862275	144	-	167
4	A47C	-	0.96	72	-	75
5	H02P	-	0.949367	75	-	79
6	H02M	-	0.948052	73	-	77
7	H05BI	-	0.974359	76	-	78
8	H01F	-	0.634328	85	-	134
9	H02K	-	0.886228	148	-	167
10	H02B	-	0.858974	67	-	78
11	H01J	-	0.947368	72	-	76
Média:			0.9098	Total: 1037		1157

Tabela 79 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RelevânciaABS – Resolução 4

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	1	147	-	147
2	H05BA	-	1	79	-	79
3	H02G	-	0.868263	145	-	167
4	A47C	-	0.96	72	-	75
5	H02P	-	0.962025	76	-	79
6	H02M	-	0.961039	74	-	77
7	H05BI	-	1	78	-	78
8	H01F	-	0.634328	85	-	134
9	H02K	-	0.892216	149	-	167
10	H02B	-	0.871795	68	-	78
11	H01J	-	0.973684	74	-	76
Média:			0.9203	Total: 1047		1157

Tabela 80 - Método 01 – Prognóstico de Topo - StemerMétodo01 - Método RankCos – Resolução 5

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.567901	0.938776	138	243	147
2	H05BA	0.659574	0.784810	62	94	79
3	H02G	0.780702	0.532934	89	114	167
4	A47C	0.930233	0.533333	40	43	75
5	H02P	0.6	0.759494	60	100	79
6	H02M	0.697674	0.779221	60	86	77
7	H05BI	0.774648	0.705128	55	71	78
8	H01F	0.791045	0.395522	53	67	134
9	H02K	0.685083	0.742515	124	181	167
10	H02B	0.651515	0.551282	43	66	78
11	H01J	0.695652	0.842105	64	92	76
Média:		0.7122	0.6877	Total: 788	1157	1157
Medida de F1:		0.6997				

Tabela 81 - Método 01 – Prognóstico de Topo - StemerMétodo 01 - Método RelevânciaCos– Resolução 5

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.660287	0.938776	138	209	147
2	H05BA	0.694444	0.949367	75	108	79
3	H02G	0.841121	0.538922	90	107	167
4	A47C	0.928571	0.693333	52	56	75
5	H02P	0.676190	0.898734	71	105	79
6	H02M	0.795455	0.909090	70	88	77
7	H05BI	0.802469	0.833333	65	81	78
8	H01F	0.862069	0.373134	50	58	134
9	H02K	0.747059	0.760479	127	170	167
10	H02B	0.727273	0.820513	64	88	78
11	H01J	0.781609	0.894737	68	87	76
Média:		0.7742	0.7828	Total: 870	1157	1157
Medida de F1:		0.7785				

Tabela 82 - Método 01 – Prognóstico de Topo - StemerMétodo01 - Método RankABS – Resolução 5

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.679245	0.979592	144	212	147
2	H05BA	0.640777	0.835443	66	103	79
3	H02G	0.766917	0.610778	102	133	167
4	A47C	0.964912	0.733333	55	57	75
5	H02P	0.550562	0.620253	49	89	79
6	H02M	0.709302	0.792208	61	86	77
7	H05BI	0.753247	0.743590	58	77	78
8	H01F	0.805970	0.402985	54	67	134
9	H02K	0.691860	0.712575	119	172	167
10	H02B	0.657534	0.615385	48	73	78
11	H01J	0.681818	0.789474	60	88	76
Média:		0.7184	0.7123	Total: 816	1157	1157
Medida de F1:		0.7153				

Tabela 83 - Método 01 – Prognóstico de Topo - StemerMétodo01 - Método RelevânciaABS – Resolução 5

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.72	0.979592	144	200	147
2	H05BA	0.630631	0.886076	70	111	79
3	H02G	0.762963	0.616766	103	135	167
4	A47C	1	0.786667	59	59	75
5	H02P	0.602273	0.670886	53	88	79
6	H02M	0.727272	0.831169	64	88	77
7	H05BI	0.705882	0.769231	60	85	78
8	H01F	0.836066	0.380597	51	61	134
9	H02K	0.699422	0.724551	121	173	167
10	H02B	0.689189	0.653846	51	74	78
11	H01J	0.734940	0.802632	61	83	76
Média:		0.7371	0.7365	837	1157	1157
Medida de F1:		0.7368				

Tabela 84 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RankCos – Resolução 5

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.986395	145	-	147
2	H05BA	-	0.974684	77	-	79
3	H02G	-	0.916168	153	-	167
4	A47C	-	0.933333	70	-	75
5	H02P	-	1.0	79	-	79
6	H02M	-	0.961039	74	-	77
7	H05BI	-	0.987179	77	-	78
8	H01F	-	0.641791	86	-	134
9	H02K	-	0.916168	153	-	167
10	H02B	-	0.858974	67	-	78
11	H01J	-	0.960526	73	-	76
Média:			0.9215	Total: 1054		1157

Tabela 85 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RelevânciaCos – Resolução 5

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.986395	145	-	147
2	H05BA	-	1.0	79	-	79
3	H02G	-	0.928144	155	-	167
4	A47C	-	0.986667	74	-	75
5	H02P	-	1.0	79	-	79
6	H02M	-	0.974026	75	-	77
7	H05BI	-	0.987179	77	-	78
8	H01F	-	0.664179	89	-	134
9	H02K	-	0.916168	153	-	167
10	H02B	-	0.935897	73	-	78
11	H01J	-	1.0	76	-	76
Média:			0.9435	Total: 1075		1157

Tabela 86 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RankABS – Resolução 5

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.993197	146	-	147
2	H05BA	-	0.95	76	-	79
3	H02G	-	0.898204	150	-	167
4	A47C	-	0.90	72	-	75
5	H02P	-	0.925	74	-	79
6	H02M	-	0.90	72	-	77
7	H05BI	-	0.95	76	-	78
8	H01F	-	0.626866	84	-	134
9	H02K	-	0.880240	147	-	167
10	H02B	-	0.8125	65	-	78
11	H01J	-	0.9125	73	-	76
Média:			0.8862	Total: 1035		1157

Tabela 87 - Método 01 – Três Prognósticos de Topo - StemerMétodo01 - Método RelevânciaABS – Resolução 5

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.993197	146	-	147
2	H05BA	-	0.9625	77	-	79
3	H02G	-	0.88024	147	-	167
4	A47C	-	0.90	72	-	75
5	H02P	-	0.9375	75	-	79
6	H02M	-	0.9125	73	-	77
7	H05BI	-	0.95	76	-	78
8	H01F	-	0.641791	86	-	134
9	H02K	-	0.886228	148	-	167
10	H02B	-	0.8625	69	-	78
11	H01J	-	0.9125	73	-	76
Média:			0.8945	Total: 1042		1157

15.0
Apêndice 7
Resultados do Algoritmo do Método 02

Tabela 88 - Método 02 - Prognóstico de Topo - StemerMétodo01 (Todos os Termos Selecionados)

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.536481	0.850340	125	233	147
1ª	A47B	0.666667	0.666667	98	147	147
2	H05BA	0.549296	0.493671	39	71	79
2ª	H05BA	0.491071	0.696203	55	112	79
3	H02G	0.683099	0.580838	97	142	167
3ª	H02G	0.761062	0.514970	86	113	167
4	A47C	0.678571	0.506667	38	56	75
4ª	A47C	0.626667	0.626667	47	75	75
5	H02P	0.348993	0.658228	52	149	79
5ª	H02P	0.299595	0.936709	74	247	79
6	H02M	0.351190	0.766234	59	168	77
6ª	H02M	0.433333	0.844156	65	150	77
7	H05BI	0.529412	0.576923	45	85	78
7ª	H05BI	0.476190	0.384615	30	63	78
8	H01F	0.696970	0.171642	23	33	134
8ª	H01F	0.795918	0.291045	39	49	134
9	H02K	0.701613	0.520958	87	124	167
9ª	H02K	0.708861	0.335329	56	79	167
10	H02B	0.342466	0.320513	25	73	78
10ª	H02B	0.436782	0.487179	38	87	78
11	H01J	0.782609	0.236842	18	23	76
11ª	H01J	0.628571	0.289474	22	35	76
Média		0.5637	0.5166	Total: 608	1157	1157
Média[a]		0.5750	0.5521	Total[a]:610	1157	

Tabela 89 - Método 02 -Três Prognósticos de Topo - StemerMétodo01 - (Todos os Termos Selecionados)

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant Total
1	A47B	-	0.979592	144	-	147
1ª	A47B	-	0.918367	135	-	147
2	H05BA	-	0.784810	62	-	79
2ª	H05BA	-	0.962025	76	-	79
3	H02G	-	0.904192	151	-	167
3ª	H02G	-	0.844311	141	-	167
4	A47C	-	0.853333	64	-	75
4ª	A47C	-	0.853333	64	-	75
5	H02P	-	0.949367	75	-	79
5ª	H02P	-	0.987342	78	-	79
6	H02M	-	0.883117	68	-	77
6ª	H02M	-	0.961039	74	-	77
7	H05BI	-	0.897436	70	-	78
7ª	H05BI	-	0.948718	74	-	78
8	H01F	-	0.462687	62	-	134
8ª	H01F	-	0.641791	86	-	134
9	H02K	-	0.790419	132	-	167
9ª	H02K	-	0.898204	150	-	167
10	H02B	-	0.641026	50	-	78
10ª	H02B	-	0.730769	57	-	78
11	H01J	-	0.473684	36	-	76
11ª	H01J	-	0.565789	43	-	76
Média			0.8465	Total: 914		1157
Média[a]			0.83	Total[a]: 978		

16.0
Apêndice 8
Resultados do Algoritmo do Método 03

Tabela 90 - Método 03 – Método Cosseno - Prognóstico de Topo - StemerMétodo01

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.381657	0.877551	129	338	147
2	H05BA	0	0	0	0	79
3	H02G	0.265306	0.077844	13	49	167
4	A47C	0.126316	0.16	12	95	75
5	H02P	0	0	0	5	79
6	H02M	0.425000	0.220779	17	40	77
7	H05BI	0	0	0	0	78
8	H01F	0.282799	0.723881	97	343	134
9	H02K	0.283333	0.101796	17	60	167
10	H02B	0.295918	0.371795	29	98	78
11	H01J	0.170543	0.289474	22	129	76
Média:		0.2028	0.25665	Total: 336	1157	1157
Medida de F1:		0.2266				

Tabela 91 - Método 03 – Método Jaccard - Prognóstico de Topo – StemerMétodo01

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.353591	0.870748	128	362	147
2	H05BA	0	0	0	0	79
3	H02G	0.224490	0.065868	11	49	167
4	A47C	0.123810	0.173333	13	105	75
5	H02P	0	0	0	2	79
6	H02M	0.414634	0.220779	17	41	77
7	H05BI	0	0	0	0	78
8	H01F	0.291793	0.716418	96	329	134
9	H02K	0.293103	0.101796	17	58	167
10	H02B	0.239583	0.294872	23	96	78
11	H01J	0.208696	0.315789	24	115	76
Média:		0.1954	0.2509	Total: 329	1157	1157
Medida de F1:		0.2197				

Tabela 92 - Método 03 – Método DICE - Prognóstico de Topo - StemerMétodo01

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.353591	0.870748	128	362	147
2	H05BA	0	0	0	0	79
3	H02G	0.224490	0.065868	11	49	167
4	A47C	0.123810	0.173333	13	105	75
5	H02P	0	0	0	2	79
6	H02M	0.414634	0.220779	17	41	77
7	H05BI	0	0	0	0	78
8	H01F	0.291793	0.716418	96	329	134
9	H02K	0.293103	0.101796	17	58	167
10	H02B	0.239583	0.294872	23	96	78
11	H01J	0.208696	0.315789	24	115	76
Média:		0.1954	0.2509	Total: 329	1157	1157
Medida de F1:		0.2197				

Tabela 93 - Método 03 – Método ABS - Prognóstico de Topo - StemerMétodo01

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.372832	0.877551	129	346	147
2	H05BA	0	0	0	0	79
3	H02G	0.234043	0.065868	11	47	167
4	A47C	0.127660	0.16	12	94	75
5	H02P	0	0	0	6	79
6	H02M	0.452381	0.246753	19	42	77
7	H05BI	0	0	0	0	78
8	H01F	0.288630	0.738806	99	343	134
9	H02K	0.293103	0.101796	17	58	167
10	H02B	0.247525	0.320513	25	101	78
11	H01J	0.183333	0.289474	22	120	76
Média:		0.2	0.2546	Total: 334	1157	1157
Medida de F1:		0.2240				

Tabela 94 - Método 03 – Método Cosseno - Três Prognósticos de Topo - StemerMétodo01

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.877551	129	-	147
2	H05BA	-	0.987342	78	-	79
3	H02G	-	0.832335	139	-	167
4	A47C	-	0.973333	73	-	75
5	H02P	-	1	79	-	79
6	H02M	-	1	77	-	77
7	H05BI	-	0.974359	76	-	78
8	H01F	-	0.798507	107	-	134
9	H02K	-	0.802395	134	-	167
10	H02B	-	0.935897	73	-	78
11	H01J	-	0.973684	74	-	76
Média:			0.9232	Total: 1039		1157

Tabela 95 - Método 03 – Método Jaccard - Três Prognósticos de Topo - StemerMétodo01

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.870748	128	-	147
2	H05BA	-	0.962025	76	-	79
3	H02G	-	0.844311	141	-	167
4	A47C	-	0.973333	73	-	75
5	H02P	-	1	79	-	79
6	H02M	-	0.987013	76	-	77
7	H05BI	-	0.974359	76	-	78
8	H01F	-	0.783582	105	-	134
9	H02K	-	0.748503	125	-	167
10	H02B	-	0.935897	73	-	78
11	H01J	-	0.960526	73	-	76
Média:			0.9128	Total: 1025		1157

Tabela 96 - Método 03 – Método DICE - Três Prognósticos de Topo - StemerMétodo01

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.870748	128	-	147
2	H05BA	-	0.962025	76	-	79
3	H02G	-	0.844311	141	-	167
4	A47C	-	0.973333	73	-	75
5	H02P	-	1	79	-	79
6	H02M	-	0.987013	76	-	77
7	H05BI	-	0.974359	76	-	78
8	H01F	-	0.783582	105	-	134
9	H02K	-	0.748503	125	-	167
10	H02B	-	0.935897	73	-	78
11	H01J	-	0.960526	73	-	76
Média:			0.9128	Total: 1025		1157

Tabela 97 - Método 03 – Método ABS - Três Prognósticos de Topo - StemerMétodo01

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.877551	129	-	147
2	H05BA	-	0.974684	77	-	79
3	H02G	-	0.826347	138	-	167
4	A47C	-	0.973333	73	-	75
5	H02P	-	1	79		79
6	H02M	-	0.987013	76	-	77
7	H05BI	-	0.974359	76	-	78
8	H01F	-	0.813433	109	-	134
9	H02K	-	0.766467	128	-	167
10	H02B	-	0.923077	72	-	78
11	H01J	-	0.960526	73	-	76
Média:			0.9161	Total: 1030		1157

17.0
Apêndice 9
Resultados do Algoritmo do Método 04

Tabela 98 - Método 04 – Similaridade Cos – Etapa Doc. Treino x Centróide - Prognóstico de Topo – StemerMétodo01 (Primeira Modalidade)

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	1	1	300	300	300
2	H05BA	1	1	284	284	284
3	H02G	1	1	333	333	333
4	A47C	1	1	300	300	300
5	H02P	1	1	297	297	297
6	H02M	1	1	264	264	264
7	H05BI	1	1	195	195	195
8	H01F	1	1	300	300	300
9	H02K	1	1	333	333	333
10	H02B	1	1	283	283	283
11	H01J	1	1	268	268	268
Média:		1	1 Total:	3157	3157	3157

Tabela 99 - Método 04 – Similaridade Cos – Etapa Doc. Teste x Centróide Prognóstico de Topo - StemerMétodo01 (Segunda Modalidade)

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.680556	1	147	216	147
2	H05BA	1	0.8	8	8	10
3	H02G	0.991736	0.718563	120	121	167
4	A47C	0.947368	0.580645	18	19	31
5	H02P	0.769231	1	10	13	10
6	H02M	1	1	10	10	10
7	H05BI	1	0.9	9	9	10
8	H01F	0.908333	0.813433	109	120	134
9	H02K	0.895706	0.874251	146	163	167
10	H02B	0.666667	0.8	8	12	10
11	H01J	0.666667	1	10	15	10
Média:		0.8660	0.8625 Total:	595	706	706
Medida de F1:		0.8642				

Tabela 100 - Método 04 – Similaridade Cos – Etapa Doc. Teste x Centróide
Três Prognósticos de Topo - StemerMétodo 01 (Segunda Modalidade)

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	1	147	-	147
2	H05BA	-	1	10	-	10
3	H02G	-	0.988024	165	-	167
4	A47C	-	1	31	-	31
5	H02P	-	1	10	-	10
6	H02M	-	1	10	-	10
7	H05BI	-	1	10	-	10
8	H01F	-	0.970149	130	-	134
9	H02K	-	0.994012	166	-	167
10	H02B	-	0.9	9	-	10
11	H01J	-	1	10	-	10
Média:			0.9887	Total: 698		706

Tabela 102 –Método 01 *versus* Método 04 – Topo, Dois e Três Prognósticos - StemerMétodo01

Item	Classif.	Técnica	Precisão	Abrangência.	Quant. Categ Corret.	Quant Total	Prognóstico
1	A47B	Método 01 CosRelev Resol1	0.699482	0.918367 [0.993197]	135 146	147	Topo [Três]
		Método 04 Terceira Modal.	0.486381	0.850340 (0.931973) [0.952381]	125 137 140		Topo (Dois) [Três]
2	H02G	Método 01 CosRelev Resol1	0.818966	0.568862 [0.826347]	95 138	167	Topo [Três]
		Método 04 Terceira Modal.	0.692308	0.215569 (0.497006) [0.688623]	36 83 115		Topo (Dois) [Três]
3	H01F	Método 01 CosRelev Resol1	0.838710	0.38806 [0.656716]	52 88	134	Topo [Três]
		Método 04 Terceira Modal.	0.590164	0.268657 (0.447761) [0.611940]	36 60 82		Topo (Dois) [Três]
4	H02K	Método 01 CosRelev Resol1	0.767742	0.712575 [0.892216]	119 149	167	Topo [Três]
		Método 04 Terceira Modal.	0.574713	0.598802 (0.748503) [0.826347]	100 125 138		Topo [Três]

Tabela 101 - Método 04 – Similaridade Cos – Etapa Doc. Teste x Centróide - Topo, Dois e Três Prognósticos - StemerMétodo01 (Terceira Modalidade)

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	A47B	0.486381	0.850340 0.931973 0.952381	125 (12) 137 (3) 140	257	147	Topo Dois Três
2	H05BA	0.15	0.3 0.6 0.8	3 (3) 6 (2) 8	20	10	Topo Dois Três
3	H02G	0.692308	0.215569 0.497006 0.688623	36 (47) 83 (32) 115	52	167	Topo Dois Três
4	A47C	0.470588	0.2580645 0.5806452 0.7741935	8 (10) 18 (6) 24	17	31	Topo Dois Três
5	H02P	0.214286	0.6 0.8 0.8	6 (2) 8 (0) 8	28	10	Topo Dois Três
6	H02M	0.2	0.3 0.8 0.8	3 (5) 8 (0) 8	15	10	Topo Dois Três
7	H05BI	0.142857	0.1 0.3 0.6	1 (2) 3 (3) 6	7	10	Topo Dois Três
8	H01F	0.590164	0.268657 0.447761 0.611940	36 (24) 60 (22) 82	61	134	Topo Dois Três
9	H02K	0.574713	0.598802 0.748503 0.826347	100 (25) 125 (13) 138	174	167	Topo Dois Três
10	H02B	0.03125	0.1 0.2 0.5	1 (1) 2 (3) 5	32	10	Topo Dois Três
11	H01J	0.093023	0.4 0.8 0.9	4 (4) 8 (1) 9	43	10	Topo Dois Três
(*macroaveraging*)							
Média		0.3314	0.3629	Total: 323	706	706	Topo
Média			0.6096	458			Dois
Média			0.7503	543			Três
(*microaveraging*)							
Média			0.4575				Topo
Média			0.6487				Dois
Média			0.7691				Três

18.0
Apêndice 10
Resultados do Algoritmo do Método 05

Tabela 103 - Modalidade 05 - Prognóstico de Topo - StemerMétodo01

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.855263	0.884354	130	152	147
2	H05BA	0.490566	0.987342	78	159	79
3	H02G	0	0	0	0	167
4	A47C	0.808989	0.96	72	89	75
5	H02P	0.579365	0.924051	73	126	79
6	H02M	0.75	0.935065	72	96	77
7	H05BI	0.855263	0.833333	65	76	78
8	H01F	0.886792	0.350746	47	53	134
9	H02K	0.751445	0.778443	130	173	167
10	H02B	0.414634	0.871795	68	164	78
11	H01J	0.869565	0.789474	60	69	76
Média:		0.6602	0.7559 Total:	795	1157	1157
Medida de F1:		0.7048				

Tabela 104 – Modalidade 05 - Três Prognósticos de Topo – StemerMétodo01

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.993197	146	-	147
2	H05BA	-	1	79	-	79
3	H02G	-	0	0	-	167
4	A47C	-	0.986667	74	-	75
5	H02P	-	1	79	-	79
6	H02M	-	0.987013	76	-	77
7	H05BI	-	0.987179	77	-	78
8	H01F	-	0.686567	92	-	134
9	H02K	-	0.970060	162	-	167
10	H02B	-	0.987179	77	-	78
11	H01J	-	0.960526	73	-	76
Média:			0.8689 Total:	935		1157

Tabela 105 – Modalidade 05V1 - Prognóstico de Topo - StemerMétodo01

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	0.892617	0.904762	133	149	147
2	H05BA	0.484472	0.987342	78	161	79
3	H02G	0	0	0	0	167
4	A47C	0.818181	0.96	72	88	75
5	H02P	0.593496	0.924051	73	123	79
6	H02M	0.768421	0.948052	73	95	77
7	H05BI	0.848101	0.858974	67	79	78
8	H01F	0.959184	0.350746	47	49	134
9	H02K	0.779762	0.784431	131	168	167
10	H02B	0.405714	0.910256	71	175	78
11	H01J	0.885714	0.815789	62	70	76
Média:		0.6760	0.7677 Total:	807	1157	1157
Medida de F1:		0.7189				

Tabela 106 – Modalidade 05V1 - Três Prognósticos de Topo - StemerMétodo01

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total
1	A47B	-	0.993197	146	-	147
2	H05BA	-	1	79	-	79
3	H02G	-	0	0	-	167
4	A47C	-	0.986667	74	-	75
5	H02P	-	1	79	-	79
6	H02M	-	0.987013	76	-	77
7	H05BI	-	0.987179	77	-	78
8	H01F	-	0.708955	95	-	134
9	H02K	-	0.970060	162	-	167
10	H02B	-	0.987179	77	-	78
11	H01J	-	0.947368	72	-	76
Média:			0.8698 Total:	937		1157

19.0
Apêndice 11
Resultados do Algoritmo do Método 06

Tabela 107 - Método 06 – Topo, Dois e Três Prognósticos - StemerMétodo01 –Treino e Teste (Primeira Modalidade)

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	A47B	0.9858	0.9903 (0.9996) [0.9996]	443 (446) [446]	447	447	Topo (Dois) [Três]
2	H05BA	0.9930	0.9943 (0.9974) [1]	291 (293) [294]	294	294	Topo (Dois) [Três]
3	H02G	0.9787	0.9507 (0.9663) [0.9792]	485 (494) [498]	500	500	Topo (Dois) [Três]
4	A47C	0.9876	0.9814 (0.9973) [1]	324 (328) [329]	329	329	Topo (Dois) [Três]
5	H02P	0.9914	0.9911 (0.9992) [1]	303 (306) [307]	307	307	Topo (Dois) [Três]
6	H02M	0.9935	0.9967 (0.9984) [0.9984]	272 (273) [273]	274	274	Topo (Dois) [Três]
7	H05BI	0.9874	0.9694 (0.9924) [1]	199 (204) [205]	205	205	Topo (Dois) [Três]
8	H01F	0.9888	0.9660 (0.9919) [0.9966]	420 (430) [433]	434	434	Topo (Dois) [Três]
9	H02K	0.9888	0.9709 (0.9924) [0.9972]	483 (496) [499]	500	500	Topo (Dois) [Três]
10	H02B	0.9993	0.9964 (1)	292 (293)	293	293	Topo (Dois)
11	H01J	0.9988	0.9792 (1)	276 (278)	278	278	Topo (Dois)
Média:		0.9903	0.9806 (0.9941) [0.9974]	Total: 3788 (3841) [3855]	3861	3861	Topo (Dois) [Três]
Medida de F1:			0.9854				

Tabela 108A - Método 06 - Topo, Dois e Três Prognósticos – StemerMétodo01 – Categoria A47B (Primeira Modalidade)

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	A47B1/00	1	1	4	4	4	Topo
2	A47B11/00	1	0.833333 (1)	5 (6)	5	6	Topo (Dois)
3	A47B13/00	1	1	15	15	15	Topo
4	A47B17/00	1	1	2	2	2	Topo
5	A47B19/00	1	1	3	3	3	Topo
6	A47B21/00	0.967742	1	30	31	30	Topo
7	A47B23/00	1	0.916667 (1)	11 (12)	11	12	Topo (Dois)
8	A47B25/00	1	1	1	1	1	Topo
9	A47B27/00	1	1	1	1	1	Topo
10	A47B29/00	1	1	2	2	2	Topo
11	A47B3/00	1	1	21	21	21	Topo
12	A47B31/00	1	1	1	1	1	Topo
13	A47B35/00	1	1	4	4	4	Topo
14	A47B37/00	1	1	24	24	24	Topo
15	A47B39/00	1	1	5	5	5	Topo
16	A47B41/00	1	1	5	5	5	Topo
17	A47B43/00	1	1	3	3	3	Topo
18	A47B45/00	1	1	2	2	2	Topo
19	A47B46/00	0.923077	1	12	13	12	Topo
20	A47B47/00	1	1	23	23	23	Topo
21	A47B49/00	1	1	5	5	5	Topo
22	A47B5/00	1	1	3	3	3	Topo
23	A47B51/00	1	1	1	1	1	Topo
24	A47B53/00	1	1	1	1	1	Topo
25	A47B55/00	1	1	4	4	4	Topo
26	A47B57/00	1	1	13	13	13	Topo
27	A47B61/00	1	1	9	9	9	Topo
28	A47B63/00	1	1	3	3	3	Topo
29	A47B65/00	1	1	2	2	2	Topo

Tabela 108B - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria A47B (Primeira Modalidade)

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
30	A47B67/00	1	0.857143 (1)	6 (7)	6	7	Topo (Dois)
31	A47B7/00	1	1	2	2	2	Topo
32	A47B71/00	1	1	1	1	1	Topo
33	A47B73/00	0.5	1	1	2	1	Topo
34	A47B75/00	1	1	3	3	3	Topo
35	A47B77/00	1	1	16	16	16	Topo
36	A47B81/00	1	1	21	21	21	Topo
37	A47B83/00	1	1	5	5	5	Topo
38	A47B85/00	1	1	4	4	4	Topo
39	A47B87/00	1	1	13	13	13	Topo
40	A47B88/00	0.96875	1	31	32	31	Topo
41	A47B9/00	1	1	7	7	7	Topo
42	A47B91/00	1	1	32	32	32	Topo
43	A47B95/00	1	1	16	16	16	Topo
44	A47B96/00	1	0.981481 0.981481 0.981481	53 (53) [53]	53	54	Topo (Dois) (Três)
45	A47B97/00	1	1	17	17	17	Topo
Média:		0.9858	0.9903 (0.9996) [0.9996]	Total: 443 (446) [446]	447	447	Topo (Dois) [Três]
Medida de F1:		0.9880					Topo

Tabela 109 – Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H05B(A) (Primeira Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H05BA1/00	0.9796	0.9897 (0.9897) [1]	96 (96) [97]	98	97	Topo (Dois) [Três]
2	H05BA11/00	1	1	1	1	1	Topo
3	H05BA3/00	1	0.9820 (1)	109 (111)	109	111	Topo (Dois)
4	H05BA6/00	0.9853	1	67	68	67	Topo
5	H05BA7/00	1	1	18	18	18	Topo
Média:		0.9930	0.9943 0.9974 [1]	Total:291 (293) [294]	294	294	Topo (Dois) [Três]
Medida de F1:		0.9936					Topo

Tabela 110 - Método 06 – Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H02G (Primeira Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H02G1/00	0.9221	1	71	77	71	Topo
2	H02G11/00	1	1	9	9	9	Topo
3	H02G13/00	1	1	3	3	3	Topo
4	H02G15/00	0.9495	0.9792 (0.9896) [1]	94 (95) [96]	99	96	Topo (Dois) [Três]
5	H02G3/00	0.9953	0.9593 (0.991) [1]	212 (219) [221]	213	221	Topo (Dois) [Três]
6	H02G5/00	1	0.6667 (0.75) [0.8333]	8 (9) [10]	8	12	Topo (Dois) [Três]
7	H02G7/00	0.9625	1	77	80	77	Topo
8	H02G9/00	1	1	11	11	11	Topo
Média:		0.9787	0.9507 (0.9663) [0.9792]	Total: 485 (494) [498]	500	500	Topo (Dois) [Três]
Medida de F1:		0.9645					

Tabela 111 - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria A47C (Primeira Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	A47C1/00	0.958333	1	46	48	46	Topo
2	A47C11/00	1	1	3	3	3	Topo
3	A47C13/00	1	1	4	4	4	Topo
4	A47C15/00	1	1	5	5	5	Topo
5	A47C16/00	1	1	11	11	11	Topo
6	A47C17/00	1	0.966102 (1)	57 (59)	57	59	Topo (Dois)
7	A47C19/00	1	1	15	15	15	Topo
8	A47C20/00	0.888889	1	8	9	8	Topo
9	A47C21/00	1	1	8	8	8	Topo
10	A47C23/00	1	0.875 (1)	7 (8)	7	8	Topo (Dois)
11	A47C27/00	0.970588	1	33	34	33	Topo
12	A47C3/00	1	0.954545 (0.954545) [1]	21 (21) [22]	21	22	Topo (Dois) [Três]
13	A47C31/00	1	1	10	10	10	Topo
14	A47C4/00	0.970588	1	33	34	33	Topo
15	A47C5/00	1	0.888889 (1)	8 (9)	8	9	Topo (Dois)
16	A47C7/00	1	1	52	52	52	Topo
17	A47C9/00	1	1	3	3	3	Topo
Média:		0.9876	0.9814 (0.9973) [1]	Total: 324 (328) [329]	329	329	Topo (Dois) [Três]
Medida de F1:		0.9845					Topo

Tabela 112 - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H02P (Primeira Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H02P1/00	1	1	55	55	55	Topo
2	H02P13/00	1	1	3	3	3	Topo
3	H02P15/00	1	1	1	1	1	Topo
4	H02P21/00	1	1	6	6	6	Topo
5	H02P23/00	1	1	6	6	6	Topo
6	H02P25/00	1	1	2	2	2	Topo
7	H02P27/00	1	1	2	2	2	Topo
8	H02P3/00	1	0.928571 (1)	13 (14)	13	14	Topo (Dois)
9	H02P5/00	0.96875	1	31	32	31	Topo
10	H02P6/00	0.930233	1	41	43	41	Topo
11	H02P7/00	0.989130	0.978495 (0.989247) [1]	91 (92) [93]	92	93	Topo (Dois) [Três]
12	H02P8/00	1	1	9	9	9	Topo
13	H02P9/00	1	0.977273 (1)	43 (44)	43	44	Topo (Dois)
Média:		0.9914	0.9911 (0.9992) [1]	Total: 303 (306) [307]	307	307	Topo (Dois) [Três]
Medida de F1:		0.9912					Topo

Tabela 113 - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H02M (Primeira Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H02M1/00	1	1	53	53	53	Topo
2	H02M11/00	1	1	7	7	7	Topo
3	H02M3/00	0.986111	1	71	72	71	Topo
4	H02M5/00	0.975	1	39	40	39	Topo
5	H02M7/00	1	0.980392 (0.990196) [0.990196]	100 (101) [101]	100	102	Topo (Dois) [Três]
6	H02M9/00	1	1	2	2	2	Topo
Média:		0.9935	0.9967 (0.9984) [0.9984]	Total: 272 (273) [273]	274	274	Topo (Dois) [Três]
Medida de F1:		0.9951					Topo

Tabela 114 - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – H05B(I) (Primeira Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H05BI33/00	1	1	26	26	26	Topo
2	H05BI35/00	1	1	1	1	1	Topo
3	H05BI37/00	0.967742	0.952381 (1)	60 (3)	62	63	Topo (Dois)
4	H05BI39/00	1	0.863636 (0.954545) [1]	19 (2) [1]	19	22	Topo (Dois) [Três]
5	H05BI41/00	0.956522	1	88	92	88	Topo
6	H05BI43/00	1	1	5	5	5	Topo
Média:		0.9874	0.9694 (0.9924) [1]	Total: 199 (204) [205]	205	205	Topo (Dois) [Três]
Medida de F1:		0.9783					Topo

Tabela 115 - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H02B - (Primeira Modalidade).

Item	Classif.	Precisão	Abrangên-cia	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H02B1/00	0.995434	1	218	219	218	Topo
2	H02B3/00	1	1	5	5	5	Topo
3	H02B5/00	1	1	13	13	13	Topo
4	H02B7/00	1	1	3	3	3	Topo
5	H02B11/00	1	1	12	12	12	Topo
6	H02B13/00	1	0.975 (1)	39 (40)	39	40	Topo (Dois)
7	H02B15/00	1	1	2	2	2	Topo
Média:		0.9993	0.9964 (1)	Total: 292 (293)	293	293	Topo (Dois)
Medida de F1:		0.9978					Topo

Tabela 116 - Método 06 -Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H01F (Primeira Modalidade).

Item	Classif.	Precisão	Abran-gência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H01F1/00	0.852941	1	58	68	58	Topo
2	H01F3/00	1	1	5	5	5	Topo
3	H01F5/00	1	1	21	21	21	Topo
4	H01F6/00	1	1	4	4	4	Topo
5	H0F7/00	1	1	41	41	41	Topo
6	H01F10/00	1	1	5	5	5	Topo
7	H01F13/00	1	1	9	9	9	Topo
8	H01F15/00	1	1	2	2	2	Topo
9	H01F17/00	1	1	5	5	5	Topo
10	H01F19/00	1	1	1	1	1	Topo
11	H01F21/00	1	1	2	2	2	Topo
12	H01F27/00	0.985401	0.971223 (0.992806) [1]	135 (138) [139]	137	139	Topo (Dois) [Três]
13	H01F29/00	1	0.785714 (0.857143) [0.928571]	11 (12) [13]	11	14	Topo (Dois) [Três]
14	H01F30/00	1	1	8	8	8	Topo
15	H01F35/00	1	1	3	3	3	Topo
16	H01F36/00	1	1	1	1	1	Topo
17	H01F37/00	0.954545	1	21	22	21	Topo
18	H01F38/00	1	0.942308 (0.980769) [1]	49 (51) [52]	49	52	Topo (Dois) [Três]
19	H01F39/00	1	0.666667 (1)	2 (3)	2	3	Topo (Dois)
20	H01F40/00	1	1	2	2	2	Topo
21	H01F41/00	0.972222	0.921053 (1)	35 (38)	36	38	Topo (Dois)
Média:		0.9888	0.9660 0.9919 0.9966	Total: 420 (430) [433]	434	434	Topo (Dois) [Três]
Medida de F1:		0.9773					Topo

Tabela 117A - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H02K (Primeira Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H02K1/00	0.8864	1	78	88	78	Topo
2	H02K11/00	1	0.961538 (1)	25 (26)	25	26	Topo (Dois)
3	H02K13/00	1	1	11	11	11	Topo
4	H02K15/00	1	1	31	31	31	Topo
5	H02K16/00	1	1	6	6	6	Topo
6	H02K17/00	1	0.875 (1)	14 (16)	14	16	Topo (Dois)
7	H02K19/00	1	1	3	3	3	Topo
8	H02K21/00	0.947368	0.947368 (1)	18 (19)	19	19	Topo (Dois)
9	H02K23/00	1	1	8	8	8	Topo
10	H02K25/00	1	1	1	1	1	Topo
11	H02K27/00	1	1	1	1	1	Topo
12	H02K29/00	1	0.916667 (1)	11 (12)	11	12	Topo (Dois)
13	H02K3/00	0.962963	0.928571 (0.964286) [1]	26 (27) [28]	27	28	Topo (Dois) [Três]
14	H02K33/00	1	0.8125 (1)	13 (16)	13	16	Topo
15	H02K35/00	1	1	6	6	6	Topo
16	H02K37/00	1	1	1	1	1	Topo
17	H02K41/00	1	1	12	12	12	Topo
18	H02K44/00	1	1	3	3	3	Topo
19	H02K47/00	1	1	4	4	4	Topo
20	H02K49/00	1	1	5	5	5	Topo
21	H02K5/00	0.985075	0.956522 (0.985507) [1]	66 (68) [69]	67	69	Topo (Dois) [Três]
22	H02K51/00	1	1	6	6	6	Topo
23	H02K53/00	0.961538	1	25	26	25	Topo
24	H02K55/00	1	0.846154 (0.846154) [0.923077]	11 (11) [12]	11	13	Topo (Dois) [Três]
25	H02K57/00	1	1	9	9	9	Topo

Tabela 117B - Método 06 Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H02K - (Primeira Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
26	H02K7/00	0.954545	0.969231 [1]	63 (65)	66	65	Topo (Dois)
27	H02K9/00	1	1	26	26	26	Topo
Média:		0.9888	0.9709 (0.9924) [0.9972]	Total: 483 (496) [499]	500	500	Topo (Dois) [Três]
Medida de F1:		0.9798					Topo

Tabela 118 - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H01J (Primeira Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H01J1/00	1	1	22	22	22	Topo
2	H01J11/00	1	1	2	2	2	Topo
3	H01J13/00	1	1	3	3	3	Topo
4	H01J17/00	1	0.875 (1)	7 (8)	7	8	Topo (Dois)
5	H0J19/00	1	1	2	2	2	Topo
6	H01J23/00	1	1	2	2	2	Topo
7	H01J25/00	1	1	4	4	4	Topo
8	H01J29/00	0.974359	1	76	78	76	Topo
9	H01J3/00	1	1	5	5	5	Topo
10	H01J31/00	1	1	6	6	6	Topo
11	H01J33/00	1	1	1	1	1	Topo
12	H01J35/00	1	0.666667 (1)	2 (3)	2	3	Topo (Dois)
13	H01J37/00	1	1	41	41	41	Topo
14	H01J40/00	1	1	5	5	5	Topo
15	H01J45/00	1	1	2	2	2	Topo
16	H01J47/00	1	1	1	1	1	Topo
17	H01J49/00	1	1	7	7	7	Topo
18	H01J5/00	1	1	15	15	15	Topo
19	H01J61/00	1	1	32	32	32	Topo
20	H01J65/00	1	1	9	9	9	Topo
21	H01J7/00	1	1	15	15	15	Topo
22	H01J9/00	1	1	17	17	17	Topo
Média:		0.9988	0.9792 (1) [1]	Total: 276 (278) [278]	278	278	Topo (Dois) [Três]
Medida de F1:		0.9889					Topo

Tabela 119 - Método 06 – Topo, Dois e Três Prognósticos - StemerMétodo01– Teste (Segunda Modalidade)

Item	Classif.	Precisão	Abran-gência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	A47B	0.9951	0.9836	145 (146)	147	147	Topo (Dois)
2	H05BA	1	1	10	10	10	Topo
3	H02G	0.9724	0.9607 (0.9688)	161 (166)	167	167	Topo (Dois)
4	A47C	1	1	31	31	31	Topo
5	H02P	1	1	10	10	10	Topo
6	H02M	1	1	10	10	10	Topo
7	H05BI	1	1	10	10	10	Topo
8	H01F	0.9873	0.9694 (0.9896)	129 (133)	134	134	Topo (Dois)
9	H02K	0.9858	0.9557 0.9798 0.9933	160 (163) [166]	167	167	Topo (Dois) [Três]
10	H02B	1	1	10	10	10	Topo
11	H01J	1	1	10	10	10	Topo (Dois)
Média:		0.9946	0.9881 0.9929 0.9941	686 (699) [702]	706	706	Topo (Dois) [Três]

Tabela 120A - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H02K (Segunda Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H02K1/00	0.866667	1	26	30	26	Topo
2	H02K11/00	1	0.875 (1)	7 (8)	7	8	Topo (Dois)
3	H02K13/00	1	1	6	6	6	Topo
4	H02K15/00	1	1	13	13	13	Topo

Tabela 120B - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H02K - (Segunda Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
5	H02K16/00	1	1	2	2	2	Topo
6	H02K17/00	1	1	4	4	4	Topo
7	H02K19/00	-	-	-	-	-	Topo
8	H02K21/00	1	1	10	10	10	Topo
9	H02K23/00	1	1	4	4	4	Topo
10	H02K25/00	-	-	-	-	-	Topo
11	H02K27/00	1	1	1	1	1	Topo
12	H02K29/00	1	0.666667 (1)	2 (3)	2	3	Topo (Dois)
13	H02K3/00	0.875	0.875 (0.875) [1]	7 (7) [8]	8	8	Topo (Dois) [Três]
14	H02K33/00	1	0.857143 (1)	6 (7)	6	7	Topo (Dois)
15	H02K35/00	1	1	3	3	3	Topo
16	H02K37/00	1	1	1	1	1	Topo
17	H02K41/00	1	1	4	4	4	Topo
18	H02K44/00	1	1	2	2	2	Topo
19	H02K47/00	1	1	1	1	1	Topo
20	H02K49/00	1	1	1	1	1	Topo
21	H02K5/00	0.952381	0.952381 (0.952381) [1]	20 (20) [21]	21	21	Topo (Dois) [Três]
22	H02K51/00	1	1	2	2	2	Topo
23	H02K53/00	1	1	8	8	8	Topo
24	H02K55/00	1	0.666667 (0.666667) [0.833333]	4 (4) [5]	4	6	Topo (Dois) [Três]
25	H02K57/00	1	1	2	2	2	Topo
26	H02K7/00	0.95	1	19	20	19	Topo
27	H02K9/00	1	1	5	5	5	Topo
Média:		0.9858	0.9557 Total:	160	167	167	Topo
			0.9798	(163)			(Dois)
			0.9933	[166]			[Três]
Medida de F1:		0.9705					

Tabela 121A - Método 06 - Topo, Dois e Três Prognósticos – StemerMétodo01 – Categoria A47B (Segunda Modalidade)							
Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	A47B1/00	1	1	2	2	2	Topo
2	A47B11/00	1	1	2	2	2	Topo
3	A47B13/00	1	1	5	5	5	Topo
4	A47B17/00	-	-	-	-	-	Topo
5	A47B19/00	1	1	2	2	2	Topo
6	A47B21/00	1	1	9	9	9	Topo
7	A47B23/00	1	0.5 (1)	1 (2)	1	2	Topo (Segundo)
8	A47B25/00	-	-	-	-	-	Topo
9	A47B27/00	-	-	-	-	-	Topo
10	A47B29/00	1	1	1	1	1	Topo
11	A47B3/00	1	1	11	11	11	Topo
12	A47B31/00	-	-	-	-	-	Topo
13	A47B35/00	1	1	4	4	4	Topo
14	A47B37/00	1	1	5	5	5	Topo
15	A47B39/00	1	1	1	1	1	Topo
16	A47B41/00	1	1	1	1	1	Topo
17	A47B43/00	1	1	1	1	1	Topo
18	A47B45/00	1	1	1	1	1	Topo
19	A47B46/00	0.833333	1	5	6	5	Topo
20	A47B47/00	1	1	7	7	7	Topo
21	A47B49/00	1	1	2	2	2	Topo
22	A47B5/00	1	1	1	1	1	Topo
23	A47B51/00	-	-	-	-	-	Topo
24	A47B53/00	-	-	-	-	-	Topo
25	A47B55/00	-	-	-	-	-	Topo
26	A47B57/00	1	1	4	4	4	Topo
27	A47B61/00	1	1	1	1	1	Topo
28	A47B63/00	1	1	1	1	1	Topo
29	A47B65/00	-	-	-	-	-	Topo

Tabela 121B - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria A47B (Segunda Modalidade)

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
30	A47B67/00	1	1	1	1	1	Topo
31	A47B7/00	-	-	-	-	-	Topo
32	A47B71/00	-	-	-	-	-	Topo
33	A47B73/00	-	-	-	1	-	Topo
34	A47B75/00	1	1	1	1	1	Topo
35	A47B77/00	1	1	8	8	8	Topo
36	A47B81/00	1	1	9	9	9	Topo
37	A47B83/00	1	1	2	2	2	Topo
38	A47B85/00	1	1	2	2	2	Topo
39	A47B87/00	1	1	4	4	4	Topo
40	A47B88/00	1	1	9	9	9	Topo
41	A47B9/00	1	1	4	4	4	Topo
42	A47B91/00	1	1	12	12	12	Topo
43	A47B95/00	1	1	5	5	5	Topo
44	A47B96/00	1	0.941176	16	16	17	Topo
45	A47B97/00	1	1	5	5	5	Topo
Média:		0.9951	0.9836 Total:	145 (146) [146]	147	147	Topo (Segundo) [Terceiro]
Medida de F1:		0.9893					

Tabela 122 – Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H05B(A) - (Segunda Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H05BA1/00	1	1	5	5	5	Topo
2	H05BA11/00	-	-	-	-	-	Topo
3	H05BA3/00	1	1	1	1	1	Topo
4	H05BA6/00	1	1	3	3	3	Topo
5	H05BA7/00	1	1	1	1	1	Topo
Média:		1	1	Total: 10	10	10	Topo

Tabela 123 - Método 06 – Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H02G (Segunda Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H02G1/00	0.923077	1	24	26	24	Topo
2	H02G11/00	1	1	4	4	4	Topo
3	H02G13/00	1	1	1	1	1	Topo
4	H02G15/00	0.939394	1	31	33	31	Topo
5	H02G3/00	1	0.9359 (1)	73 (78)	73	78	Topo (Dois)
6	H02G5/00	1	0.75	3	3	4	Topo
7	H02G7/00	0.916667	1	22	24	22	Topo
8	H02G9/00	1	1	3	3	3	Topo
Média: Topo		0.9724	0.9607 Total: 161		167	167	
(Dois)			0.9688	(166)			
Medida de F1:		0.9665					

Tabela 124 - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria A47C (Segunda Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	A47C1/00	1	1	3	3	3	Topo
2	A47C11/00	1	1	1	1	1	Topo
3	A47C13/00	-	-	-	-	-	Topo
4	A47C15/00	-	-	-	-	-	Topo
5	A47C16/00	1	1	2	2	2	Topo
6	A47C17/00	1	1	4	4	4	Topo
7	A47C19/00	1	1	1	1	1	Topo
8	A47C20/00	1	1	2	2	2	Topo
9	A47C21/00	-	-	-	-	-	Topo
10	A47C23/00	-	-	-	-	-	Topo
11	A47C27/00	1	1	7	7	7	Topo
12	A47C3/00	1	1	3	3	3	Topo
13	A47C31/00	1	1	1	1	1	Topo
14	A47C4/00	1	1	1	1	1	Topo
15	A47C5/00	1	1	1	1	1	Topo
16	A47C7/00	1	1	5	5	5	Topo
17	A47C9/00	-	-	-	-	-	Topo
Média: Topo		1	1	Total: 31	31	31	
Medida de F1:		1					

Tabela 125 - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H02P (Segunda Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H02P1/00	1	1	1	1	1	Topo
2	H02P13/00	-	-	-	-	-	Topo
3	H02P15/00	-	-	-	-	-	Topo
4	H02P21/00	-	-	-	-	-	Topo
5	H02P23/00	-	-	-	-	-	Topo
6	H02P25/00	-	-	-	-	-	Topo
7	H02P27/00	-	-	-	-	-	Topo
8	H02P3/00	-	-	-	-		Topo
9	H02P5/00	1	1	3	3	3	Topo
10	H02P6/00	-	-	-	-	-	Topo
11	H02P7/00	1	1	5	5	5	Topo
12	H02P8/00	-	-	-	-	-	Topo
13	H02P9/00	1	1	1	1	1	Topo
Média:		1	1	Total: 10	10	10	Topo
Medida de F1 :		1					

Tabela 126- Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H02M (Segunda Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H02M1/00	1	1	2	2	2	Topo
2	H02M11/00	-	-	-	-	-	Topo
3	H02M3/00	1	1	4	4	4	Topo
4	H02M5/00	1	1	1	1	1	Topo
5	H02M7/00	1	1	3	3	3	Topo
6	H02M9/00						Topo
Média:		1	1	Total: 10	10	10	
Topo							
Medida de F1:		1					

Tabela 127 - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H05B(I) - (Segunda Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H05BI33/00	1	1	1	1	1	Topo
2	H05BI35/00	-	-	-	-	-	Topo
3	H05BI37/00	1	1	2	2	2	Topo
4	H05BI39/00	-	-	-	-	-	Topo
5	H05BI41/00	1	1	5	5	5	Topo
6	H05BI43/00	1	1	2	2	2	Topo
Média:		1	1	Total: 10	10	10	

Topo
Medida de F1: 1

Tabela 128 - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H02B - (Segunda Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H02B1/00	1	1	7	7	7	Topo
2	H02B3/00	-	-	-	-	-	Topo
3	H02B5/00	-	-	-	-	-	Topo
4	H02B7/00	-	-	-	-	-	Topo
5	H02B11/00	-	-	-	-	-	Topo
6	H02B13/00	1	1	3	3	3	Topo
7	H02B15/00	-	-	-	-	-	Topo
Média:		1	1	Total: 10	10	10	

Topo
Medida de F1: 1

Tabela 129 - Método 06 -Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H01F - (Segunda Modalidade).

Item	Classif.	Precisão	Abran-gência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H01F1/00	0.85	1	17	20	17	Topo
2	H01F3/00	1	1	1	1	1	Topo
3	H01F5/00	1	1	10	10	10	Topo
4	H01F6/00	1	1	1	1	1	Topo
5	H0F7/00	1	1	12	12	12	Topo
6	H01F10/00	1	1	1	1	1	Topo
7	H01F13/00	1	1	3	3	3	Topo
8	H01F15/00	-	-	-	-	-	Topo
9	H01F17/00	1	1	2	2	2	Topo
10	H01F19/00	-	-	-	-	-	Topo
11	H01F21/00	1	1	1	1	1	Topo
12	H01F27/00	0.947368	0.972973 (1)	36 (37)	38	37	Topo (Dois)
13	H01F29/00	1	0.666667 (0.833333)	4 (5)	4	6	Topo (Dois)
14	H01F30/00	1	1	4	4	4	Topo
15	H01F35/00	-	-	-	-	-	Topo
16	H01F36/00	-	-	-	-	-	Topo
17	H01F37/00	1	1	7	7	7	Topo
18	H01F38/00	1	0.933333 (1)	14 (15)	14	15	Topo (Dois)
19	H01F39/00	-	-	-	-	-	Topo (Dois)
20	H01F40/00	1	1	1	1	1	Topo
21	H01F41/00	1	0.9375 (1)	15 (16)	15	16	Topo (Dois)
Média Topo		0.9873	0.9694	Total: 129	134	134	
(Dois)			0.9896	(133)			

Tabela 130 - Método 06 - Topo, Dois e Três Prognósticos - StemerMétodo01 – Categoria H01J - (Segunda Modalidade).

Item	Classif.	Precisão	Abrangência	Quant. Categor. Corret.	Quant. Categ Classif	Quant. Total	Prognóstico
1	H01J1/00	1	1	2	2	2	Topo
2	H01J11/00	-	-	-	-	-	Topo
3	H01J13/00	-	-	-	-	-	Topo
4	H01J17/00	1	1	1	1	1	Topo
5	H0J19/00	-	-	-	-	-	Topo
6	H01J23/00	-	-	-	-	-	Topo
7	H01J25/00	1	1	2	2	2	Topo
8	H01J29/00	1	1	2	2	2	Topo
9	H01J3/00	-	-	-	-	-	Topo
10	H01J31/00	-	-	-	-	-	Topo
11	H01J33/00	-	-	-	-	-	Topo
12	H01J35/00	-	-	-	-	-	Topo
13	H01J37/00	1	1	1	1	1	Topo
14	H01J40/00	-	-	-	-	-	Topo
15	H01J45/00	-	-	-	-	-	Topo
16	H01J47/00	-	-	-	-	-	Topo
17	H01J49/00	-	-	-	-	-	Topo
18	H01J5/00	1	1	1	1	1	Topo
19	H01J61/00	1	1	1	1	1	Topo
20	H01J65/00	-	-	-	-	-	Topo
21	H01J7/00	-	-	-	-	-	Topo
22	H01J9/00	-	-	-	-	-	Topo
Média: Topo		1	1	Total: 10	10	10	
Medida de F1:		1					

20.0
Apêndice 12
Fatores de Pesos Usados no Algoritmo do Método 05

Tabela 131- Fatores de Peso para Verificação das Similaridades dos Intervalos dos Termos Comuns e Não-Comuns, Levando-se em Consideração Todos os Bits para o Algoritmo Modalidade 05.

Intervalos		Pesos	Intervalos		Pesos	Intervalos		Pesos
D1	D2		D1	D2		D1	D2	
111(7)	111(7)	1.60	-	-	-	-	-	-
111(7)	110(6)	1.00	110(6)	110(6)	1.60	-	-	-
111(7)	101(5)	0.85	110(6)	101(5)	0.90	101(5)	101(5)	1.60
111(7)	100(4)	0.70	110(6)	100(4)	0.75	101(5)	100(4)	0.80
111(7)	011(3)	0.55	110(6)	011(3)	0.60	101(5)	011(3)	0.65
111(7)	010(2)	0.40	110(6)	010(2)	0.45	101(5)	010(2)	0.50
111(7)	001(1)	0.25	110(6)	001(1)	0.30	101(5)	001(1)	0.35
111(7)	000(0)	0	110(6)	000(0)	0	101(5)	000(0)	0
100(4)	100(4)	1.60	-	-	-	-	-	-
100(4)	011(3)	0.70	011(3)	011(3)	1.60	-	-	-
100(4)	010(2)	0.55	011(3)	010(2)	0.60	010(2)	010(2)	1.60
100(4)	001(1)	0.40	011(3)	001(1)	0.45	010(2)	001(1)	0.30
100(4)	000(0)	0	011(3)	000(0)	0	010(2)	000	0
001(1)	001(1)	1.60						
001(1)	000(0)	0						

Tabela 132- Fatores de Peso para Verificação das Similaridades dos Intervalos dos Termos Comuns, Levando-se em Consideração Somente Um ou Dois Bits de Maior Ordem para o Algoritmo Modalidade 05.

Intervalos		Pesos	Intervalos		Pesos	Intervalos		Pesos
D1	D2		D1	D2		D1	D2	
11-(7)	11-(7)	1.40	11-(6)	11-(6)	1.40	10-(5)	10-(5)	1.40
10-(4)	10-0(4)	1.40	01-(3)	01-(3)	1.40	01-(2)	01-(2)	1.40
00-(1)	00-(1)	0						
1--(7)	1--(7)	1.20	1--(6)	1--(6)	1.20	1--(5)	1--(5)	1.20
1--(4)	1--(4)	1.20	0--(3)	0--(3)	1.20	0--(2)	0--(2)	1.20
0--(1)	0--(1)	0						

Tabela 133 - Fatores de Peso para Verificação das Similaridades dos Intervalos dos Termos Comuns e Não-Comuns Levando-se em Consideração Todos os Bits para o Algoritmo. Modalidade 05V1.

Intervalos		Pesos	Intervalos		Pesos	Intervalos		Pesos
D1	D2		D1	D2		D1	D2	
111(7)	111(7)	1.7	-	-	-	-	-	-
111(7)	110(6)	1.0	110(6)	110(6)	1.6	-	-	-
111(7)	101(5)	0.90	110(6)	101(5)	0.85	101(5)	101(5)	1.5
111(7)	100(4)	0.80	110(6)	100(4)	0.75	101(5)	100(4)	0.70
111(7)	011(3)	0.70	110(6)	011(3)	0.65	101(5)	011(3)	0.60
111(7)	010(2)	0.60	110(6)	010(2)	0.55	101(5)	010(2)	0.50
111(7)	001(1)	0.50	110(6)	001(1)	0.45	101(5)	001(1)	0.40
111(7)	000(0)	0	110(6)	000(0)	0	101(5)	000(0)	0
100(4)	100(4)	1.4	-	-	-	-	-	-
100(4)	011(3)	0.55	011(3)	011(3)	1.3	-	-	-
100(4)	010(2)	0.45	011(3)	010(2)	0.40	010(2)	010(2)	1.2
100(4)	001(1)	0.35	011(3)	001(1)	0.30	010(2)	001(1)	0.25
100(4)	000(0)	0	011(3)	000(0)	0	010(2)	000	0
001(1)	001(1)	1.1						
001(1)	000(0)	0						

Tabela 134A – Fatores de Peso para Verificação das Similaridades dos Intervalos dos Termos Comuns e Não-Comuns, Levando-se em Consideração os Dois Bits de Maior Ordem para o Algoritmo Modalidade 05V1.

Intervalos		Pesos	Intervalos		Pesos	Intervalos		Pesos
D1	D2		D1	D2		D1	D2	
11-(7)	11-(7)	1.0	-	-	-	-	-	-
11-(7)	11-(6)	1.0	11-(6)	11-(6)	1.0	-	-	-
11-(7)	10-(5)	0.75	11-(6)	10-(5)	0.75	10-(5)	10-(5)	0.70
11-(7)	10-(4)	0.75	11-(6)	10-(4)	0.75	10-(5)	10-(4)	0.70
11-(7)	01-(3)	0.55	11-(6)	01-(3)	0.55	10-(5)	01-(3)	0.45
11-(7)	01-(2)	0.55	11-(6)	01-(2)	0.55	10-(5)	01-(2)	0.45
11-(7)	00-(1)	0.45	11-(6)	00-(1)	0.45	10-(5)	00-(1)	0.40
11-(7)	00-(0)	0	11-(6)	00-(0)	0	10-(5)	00-(0)	0

Tabela 134B– Fatores de Peso para Verificação das Similaridades dos Intervalos dos Termos Comuns e Não-Comuns, Levando-se em Consideração os Dois Bits de Maior Ordem para o Algoritmo Modalidade 05V1.

Intervalos		Pesos	Intervalos		Pesos	Intervalos		Pesos
D1	D2		D1	D2		D1	D2	
10-(4)	10-0(4)	0.70	-	-	-	-	-	-
10-(4)	01-(3)	0.45	01-(3)	01-(3)	0.40	-	-	-
10-(4)	01-(2)	0.45	01-(3)	01-(2)	0.40	01-(2)	01-(2)	0.40
10-(4)	00-(1)	0.40	01-(3)	00-(1)	0.25	01-(2)	00-(1)	0.25
10-(4)	00-(0)	0	01-(3)	00-(0)	0	01-(2)	00-(0)	0
00-(1)	00-(1)	0						

Tabela 135- Fatores de Peso para Verificação das Similaridades dos Intervalos dos Termos Comuns e Não-Comuns, Levando-se em Consideração Somente o Bit de Maior Ordem para o Algoritmo Modalidade 05V1.

Intervalos		Pesos	Intervalos		Pesos	Intervalos		Pesos
D1	D2		D1	D2		D1	D2	
1--(7)	1--(7)	0.65	-	-	-	-	-	-
1--(7)	1--(6)	0.65	1--(6)	1--(6)	0.65	-	-	-
1--(7)	1--(5)	0.65	1--(6)	1--(5)	0.65	1--(5)	1--(5)	0.65
1--(7)	1--(4)	0.65	1--(6)	1--(4)	0.65	1--(5)	1--(4)	0.65
1--(7)	0--(3)	0.35	1--(6)	0--(3)	0.35	1--(5)	0-1-(3)	0.35
1--(7)	0--(2)	0.35	1--(6)	0--(2)	0.35	1--(5)	0--(2)	0.35
1--(7)	0--(1)	0.35	1--(6)	0--(1)	0.35	1--(5)	0--(1)	0.35
1--(4)	1--(4)	0.65	-	-	-	-	-	-
1--(4)	0--(3)	0.35	0--(3)	0--(3)	0	-	-	-
1--(4)	0--(2)	0.35	0--(3)	0--(2)	0	0--(2)	0--(2)	0
1--(4)	0--(1)	0.35	0--(3)	0--(1)	0	0--(2)	0--(1)	0

21.0
APÊNDICE 13

ICIEA - The 6 th IEEE CONFERENCE ON INDUSTRIAL ELECTRONICS AND APPLICATIONS

Accept Paper - Interactive Session TuMP19 – Tuesday, 21 june, 2011, 1330-1550 – 16:30 (Paper submetido e aceito no ICIEA)

TEXT CATEGORIZATION

STUDY CASE: PATENTS' APPLICATION DOCUMENTS

NEIDE DE OLIVEIRA GOMES, M.Sc.
of Electric Engineer Department
Pontifical Catholic University, PUC
Rio de Janeiro, RJ
nog@inpi.gov.br

Emmanuel Piceses Lopes Passos, D.Sc.
of Electric Engineer Department
Pontifical Catholic University, PUC
Rio de Janeiro, RJ
manupas@uninet.com.br

Abstract—This paper presents computational methods aiming to patent's text categorization in Portuguese language, involving techniques from machine learning and computational linguistics. The algorithm used was the k-Nearest Neighbor method (k-NN) modified which showed good results, although it requires much computational time in the training stage. For the pre-processing step, it was implemented, with modifications, the stemming method called StemmerPortuguese that includes the removal of suffixes, besides the removal of stopwords and treatment of compound terms.

Keywords: Text Categorization; Text Classification; Knowledge Discovery in Texts; Categorization of Patents' Applications; Classification of Patent's Applications.

I. INTRODUCTION

A patent application represents the first step to protect an invention, which is considered an indicator of innovation in the country and in the economy. Patents cover a wide range of categories and each field can be divided into subtopics until a reasonable level of expertise is achieved. Patent's searching to find prior art inventions, called the state of the art, are based primarily on the accuracy of patents' categorization.

Patent documents are categorized and indexed with the help of the International Patent Classification (IPC). In the European office, after indexing, the documents are entered into computerized database so that users of the patents' system can recover the technological information contained in these assets to intellectual production. In the Brazilian Institute of Industrial Property the classification of patents' application are made by specialists. Until the beginning of this research, based on the Brazilian patent Institute database, only the abstract of the applications were made available electronically through RPI's (Magazine of Industry Property).

The IPC is a complex indexing system. Because of its comprehensive scope, human classifiers that are not experts in all areas have difficulty to categorize manually a document according to IPC. An automated tool can also help patents' offices to categorize the patents' applications where specialists are not numerous.

Most papers published performance of test algorithms based on well-written texts, such as the Reuters collection, which contains newspaper articles, and such texts contain well objective information and are written and reviewed by professionals, so with very few spelling errors. However, the Brazilian texts of patents' applications contain misspellings and typing, sometimes written by people with not knowledgeable of the subject or with no well misspelling.

Due to the nature of unstructured text, documents (texts) need a pre-processing to undergo learning algorithms. The transformation of documents in a more appropriate representation, as in an attribute-value table is very important.

The focus of this research is to find computational methods aiming to text categorization of patents, in Portuguese language, able to classify patents' applications accurately and quickly, involving techniques of machine learning and computational linguistics.

II. CLASSIFICATION OF TEXTS

The text categorization is a supervised learning technique used in natural language processing to classify text documents, based on probabilities suggested by the set of training documents already categorized. The technique can be used as a way to organize documents for both recovery and for storage, from pre-defined categories. For definition of categories is required: collect the database for pre-processing step; development of indexing; reduction of the dimensionality of the base; implementation of the classifier; and calculation of the performance measures.

In the pre-processing step, each document is analysed. Should be applied techniques that facilitate the process of features selection of texts such as: removal of all words (stopwords) that do not influence the definition of the category of text; removal of symbols; treatment of compound terms; conversion of radical texts, among others.

For the conversion of the radical of the words of the text, it was used the StemmerPortuguese method in the Portuguese language [23] with modifications. Also, not to have any misunderstanding of some compound terms, which when appear together have meanings different that each one has separately, that terms were combined, to avoid changing the meaning of the expression. For example: "alta tensão" has been

TABLE I. RESOLUTIONS FOR THE K-NN ALGORITHM

Resolutions	k	Resolutions	k
1	13	4	25
2	17	5	31
3	23	-	-

In the kNN algorithm, due to the difficulty of determining the best value for k, should be conducted a series of experiments with different values of k to determine the best value.

In this study we used five (5) resolutions in the test step, as showed in table (1) [9].

After calculating the k-neighbors, many strategies can be taken to predict the category of a test document. A fixed value of k, as chosen in table (1) is usually used for all categories, regardless of their different distributions [2].

The equations (6) and (7) show 2(two) strategies used to this method: [2] [14]

$$y(d_i) = \arg\max_k \sum_{x_j \in kNN} y(x_j, c_k) \quad (6)$$

$$y(d_i) = \arg\max_k \sum_{x_j \in kNN} Sim(d_i, x_j) \cdot y(x_j, c_k) \quad (7)$$

where d_i is the test document, x_j is one of the neighbors in the training set, $y(x_j, c_k) \in [0,1]$ indicates that x_j belongs to the class c_k and $Sim(d_i, x_j)$ is the function similarity between d_i and x_j. [2] [14]

According to the strategy used in the first equation (6), the category selected will be that having the greatest amount of training documents in the k-nearest neighbor (threshold based on rank). The categories of k-neighbors are ordered decreasingly according to their similarity measures and N categories in the best position will be chosen as the categories of document d_i. For n = 1, i.e., the category of greatest importance will be the type of the document. [2] [14] [9]

According to the strategy used in the second equation (7), the category with the maximum sum of similarities is the winner (threshold based on relevance). During the process of categorization, the categories of k-neighbors, whose relevance factor reach the best positions will be chosen as the categories of document.

We consider that the algorithm returns an ordered quantity of predictions or suggestions for categories, where the order is determined by the level of performance used as weight. The following measures are defined: [21] [22]

- Top Prognostic: The top category provided by the categorizer is compared with the main category of the document;
- Three Top Prognostic: The top three categories provided by the categorizer are compared with the category of the document. If only one category is correct, the categorizer is considered successful.

III. PÓS-PROCESSING

A. *PERFORMANCE MEASUREMENTS*

In general, information retrieval systems are evaluated by using two metrics: precision and recall. It was also used the metric accuracy and F-measure. In this trial is being shown only the Precision metric which consists of the amount of documents correctly categorized divided by the amount of the documents belonging to a given category.

B. *DATABASES*

The database used was of the Brazilian database of patents, concerning the patents' applications deposited in Brazil, including the subclasses: Heating and Lighting (H05B); Cable or Power Lines (H02G); Panels (H02B); Magnets and Inductances (H01F); Electrical Machines (H02K); Energy Conversion (H02M); Control or Regulation (H02P); Discharge Tubes, Discharge Lamps (H01J); Tables, Desks, Office Furnitures, Cupboards, Drawers, Details General Furniture Details (A47B); Chairs (A47C). The subclass H05B was divided in H05BA (Heating) and H05BI (Lightning).

For the subclasses H05B, H02G, H02B, H01F, H02K, H02M, H02P, H01J were analyzed 3085 (three thousand and eighty-five) documents. For categories A47B and A47C were analyzed 778 (seven hundred and seventy-eight) documents.

The documents, in the form *.txt, consist of only the Abstract of the patents' applications, because in the beginning of this research, the Description, Claims and Drawings of the applications were not available electronically on the Brazilian database. The documents were divided between training and testing step.

C. RESULTS

Many results were obtained with the simulations. We have calculated the average of the various results obtained among the categories: H05B; H02G; H02B; H01F; H02K; H02M; H02P; H01J; A47B; and A47C. In the figure (1) were illustrated the average of the various results representing: 1- Rank Cosine Method, Resolution 1st; 2- Relevance Cosine Method, Resolution 1st; 3- Rank ABS Method, Resolution 1st; 4- Relevance ABS Method, Resolution 1st; 5- Rank Cosine Method, Resolution 2nd; 6- Relevance Cosine Method, Resolution 2nd; 7- Rank ABS Method, Resolution 2nd; 8 – Relevance ABS Method, Resolution 2nd; 9- Rank Cosine Method, Resolution 3rd; 10- Relevance Cosine Method, Resolution 3rd; 11– Rank ABS Method, Resolution 3rd; 12– Relevance ABS Method, Resolution 3rd; 13- Rank Cosine Method, Resolution 4th; 14- Relevance Cosine Method, Resolution 4th; 15– Rank ABS Method, Resolution 4th; 16– Relevance ABS Method, Resolution 4th; 17- Rank Cosine Method, Resolution 5th; 18- Relevance Cosine Method, Resolution 5th; 19 – Rank ABS Method, Resolution 5th; 20 – Relevance ABS Method, Resolution 5th. The lower curve (blue) represents the results using the Top Prognostic (serie1) technique and upper curve (gray) represents the results using the Three Top Prognostic technique (Serie2).

Referring to the Similarity measurements as Jaccard and DICE, both don't presented good results, so their results were

not presented here. Concerning to Cosine and ABS Similarities measurements, both presented good results.

Referring to the measures of the Top Prognostics, the performance, on average, the Relevance method presented better results when compared with the Rank method and the Cosine Relevance method had better performance that the ABS Relevance method.

Concerning to the resolution method and to the Top Prognostic, there were not significant changes among the results of them, however, on average, the results that had used the resolution 2nd, showed a slight improvement over the others.

Referring to the measures of the Three Top Prognostics, on average, the method of Relevance presented better results when compared to the method of Rank. The Relevance cosine method had better results than the method of Relevance ABS. Concerning to resolution, resolution 4th showed a slight improvement over the others.

The categories that presented better results, on average, in descending order for Top Diagnostic were: A47B; H05B(A); H01J; H02M; H05B(I); H02P; H02K; H02B. The worst were: H01F; H02G; A47C.

In Figure (2) and in the Figure (3) were illustrated the various results obtained in the simulations of the techniques used for the A47B and H01J categories respectively. Series 1 is related with Top Diagnostic technique and Series 2 is concerned to Three Top Diagnostic Method.

As illustrated in Figure (2) and referring only to A47B category and Top Diagnostic Technique, Relevance Method do not presented significant better results when compared to Rank Method. Comparing Cosine versus ABS techniques, Rank ABS and Relevance ABS present best results if compared with Rank Cosine and Relevance Cosine Method. Concerning to Three Diagnostic Technique, all methods presented good results and do not presented significatively differences, however concerning to resolution 3rd and resolution 4th, both present slight performance referring to Rank ABS and Relevance ABS.

As illustrated in Figure (3) and referring only to H01J category and Top Diagnostic Technique, Relevance Method presented better results when compared to Rank Method. Comparing Cosine versus ABS techniques, Rank Cosine and Relevance Cosine present best results if compared with Rank ABS and Relevance ABS Method. Concerning to Three Diagnostic Technique, all methods presented good results, but Relevance Method had presented better results when compared with Rank Method. Of all methods, Relevance Cosine had presented the best results.

Figure 1. Results of the Simulation concerning the k-NN Method

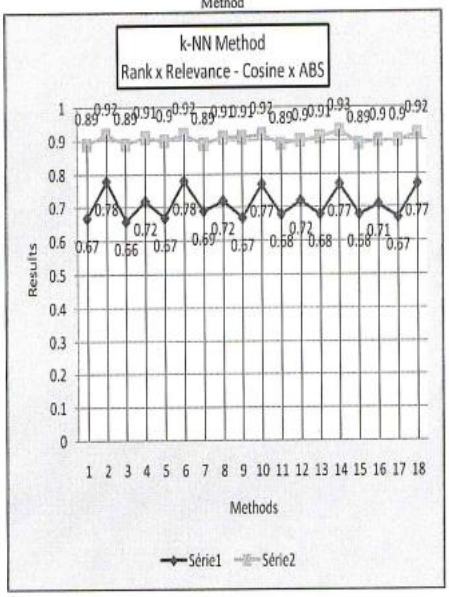


Figure 2. Results of the Simulation concerning the k-NN Method

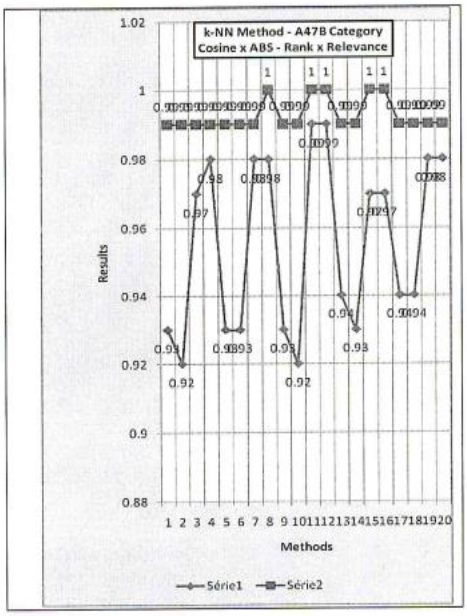
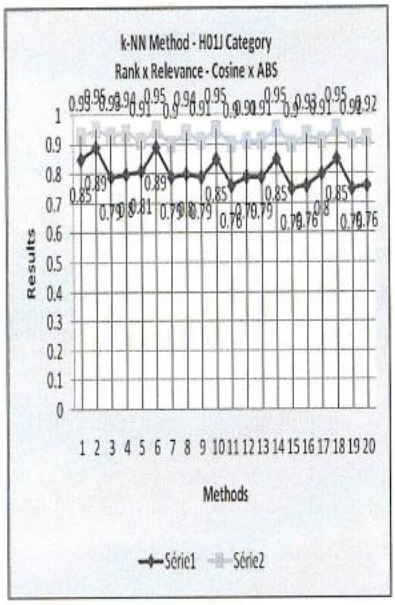


Figure 3. Results of the Simulation Concerning the k-NN Method



IV. CONCLUSION

The results of simulations used for the categorization of patents' applications in Portuguese consisted of a modified Algorithm K-Nearest Neighbor (k-NN) included the following steps: removal of the stopwords; treatment of the compound terms; stemming with the modified algorithm Stemmer Portuguese; weighting with the "Frequency of Modified Terms (TF') x Inverse Document Frequency (IDF)" technique; cosine, Jaccard, DICE, ABS similarity techniques; categorization with the k-NN algorithm using Rank and Relevance method and the results presented according to Top Prognostic and Three Top Prognostic.

For future research will be made other simulations, as variants of k-NN algorithm, to help to conclude which method should be used for the categorization of patents' applications.

REFERENCES

[1] Alvarez, Reinaldo Viana; Research Process in Stemming Portuguese; thesis in Master Program in Computer Science from, 2005.

[2] [1]Baoli, Li; [2]Shiwen, Yu, Lu Qin; An Improved K-Nearest Neighbor Algorithm for Text Categorization; [1]Institute of Computational Linguistics, Department of Computer Science and Technology, Peking University, Beijing, P. R. China, [2]Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong ~ 2004.

[3] Campbell, Yuri Barwick Lannes; Linguistic Approach to Automatic Text Classification in Portuguese; thesis UFRJ, 2007.

[4] Dias, Maria Abadia Lacerda; Malheiros, Marcelo Gomensoro; Automatic Extraction of Keywords Texts of Portuguese; University Center UNIVATES; Tilings - RS, 2005.

[5] Fall, C. J.; Benzineb, K; Literature Survey: Issues to Be Considered in the Automatic Classification of Patents, World Intellectual Property Organization, Geneva.

[6] Goldschmidt, Ronaldo, Passos, Emmanuel Pices Lopes; Intelligent Aiding in the Orientation Process of Knowledge Discovery in Databases, PUC - RJ, 2004.

[7] Goldschmidt, Ronaldo, Passos, Emmanuel Pices Lopes; Data Mining A Practical Guide, RJ, 2005, Editora Campus.

[8] Gonzalez, Marco Antonio Insaurriaga; Terms and Relationships of Evidence in Information Retrieval; Federal University of Rio Grande do Sul, Institute of Computer Sciences, Doctoral Thesis in Computer Science, July 2005.

[9] [1]Hadi, Wa'el Musa; [2]Thabtah, Fadi; [1]Mousa, Salahideen; [1]AL Hawari, Samer; [1]Kanaan, Ghassan; [1]Ababnih, Jafar; A Comprehensive Comparative Study Using Vector Space Model with k-Nearest Neighbor on Text Categorization Data ; [1]Department of Computer Information Systems - Arab Academy for Banking and Financial Sciences - Amman, Jordan; [2]Department of MIS, Philadelphia University, Amman, Jordan.

[10] Krishnakumar, Anita; Building a kNN Classifier for Reuters-21578 Collection.

[11] Loh, Stanley; Oliveira, José Palazzo M. de; Gameiro, Maurício A.; Knowledge Discovery in Texts for Constructing Decision Support Systems.

[12] Magellan, Lúcia Helena; Arbex, Márcio Aarestrup, Text Mining Support for Decision Aiding, Institute of Integrated Schools Junior Vianna.

[13] Martins, Claudia A.; Maria Carolina Monard, Matsubara, Edson T.; A Computational Tool to Assist in Pre-Processing of Texts, Computational Intelligence Lab.

[14] Moraes, Silvia Maria Wanderley, Lima, Vera Lúcia Strube de; A Study of Hierarchical Proceedings Categorization of a Great Collection of Texts in Portuguese; PUCRS - Graduate Program in Computer Science, Catholic University of Rio Grande do Sul; Proceedings of the XXVII Congress SBC 2007, Workshop on Information Technology and Human Language.

[15] Orengo, Viviane Moreira; A Stemming Algorithm for the Portuguese Language; In Proceedings of the SPIRE Conference, Laguna San Raphael, November 13-15, 2001.

[16] Santos, Maria Angela Moscalewski of Roveredo; Extracting Association Rules from Text, Master's Thesis in Applied Computer Science, Catholic University of Parana, Curitiba, 2002.
Available in http://www.ppgia.pucpr.br/ensino/defesas/Maria_Angela_2002.PDF

[17] Schijvenaars, Bob J. A.; Schuemie, Martijn J.; Mülligen, Erick M. van; Weeber, Marc; Jelier, Rob; Mons, Barend; Kors, Jan A.; A Concept-Based Approach to Text Categorization, Notebook Paper TREC 2005 Genomics Track, Department of Medical Informatics

[18] Seddiqui, Md. Hanif, Seki, Yohei; Aono, Masaki; The Semantic Approach to Mining for Patent Relating to the IPC Research Paper Abstract.

[19] Soucy, Pascal, Guy W. Mineau, A Simple k-NN Algorithm For Text Categorization, Department of Computer Science, Université Laval, Quebec, Canada.

[20] Tanabe, Akeo; The Verbal Stemmer for Brazilian Portuguese Language, Department of Informatics, PUC - RJ.

[21] [1]Tikk, Domonkos; [2]Biró, György; Hierarchical Text Categorization Experiment with Method-Alpha on the WIPO Patent Collection; [1]Department of Telecom and Media Informatics, Budapest University of Technology and Economics, Budapest, Hungary, [2]Department of Informatics, Eötvös Loránd Science University, Budapest, Hungary, Proceedings of the Fourth International Symposium on Uncertainty Modeling and Analysis, IEEE 2003.

[22] [1]Tik, Domonkos; [2]Biró György; Text Categorization on the Multi-Lingual Corpus; [1]Department of Telecommunication and Media Informatics, Budapest University of Technology and Economics; [2]Department of Informatics, Eötvös Loránd Science University, Department of Informatics, 2001.