COPPE/UFRJ

MODELO COMPUTACIONAL PARA MINERAÇÃO DE TEXTO E ANÁLISE DE QUESTÕES DE CONCURSOS

Jorge da Cunha Morgado Júnior

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Civil, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Civil.

Orientador: Nelson Francisco Favilla Ebecken

Rio de Janeiro
Dezembro de 2008

MODELO COMPUTACIONAL PARA MINERAÇÃO DE TEXTO E ANÁLISE DE QUESTÕES DE CONCURSOS

Jorge da Cunha Morgado Júnior

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA CIVIL.

Aprovada por:

_____

Prof. Nelson Francisco Favilla Ebecken, D.Sc.

_____

Profª. Valeria Menezes Bastos, D.Sc.

_____

Profª. Beatriz de Souza Leite Pires de Lima, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

DEZEMBRO DE 2008

Morgado Júnior, Jorge da Cunha

Modelo Computacional para Mineração de Texto e Análise de Questões de Concursos/Jorge da Cunha Morgado Júnior. – Rio de Janeiro: UFRJ/COPPE, 2008.

XII, 85 p.: il.; 29,7 cm.

Orientador: Nelson Francisco Favilla Ebecken

Dissertação (mestrado) – UFRJ/ COPPE/ Programa de Engenharia Civil, 2008.

Referencias Bibliográficas: p. 81-85.

1. Mineração de Textos. 2. Questões de Concursos. 3. Análise de Links. 4. Classificação. 5. Agrupamento. I. Ebecken, Nelson Francisco Favilla. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Civil. III. Titulo.

Agradecimentos

Em primeiro lugar, agradeço enormemente aos meus pais, Jorge e Stella, por me darem sempre total apoio em todos os momentos de minha vida.

Agradeço à minha namorada, Renata, pela paciência e apoio incondicional, e por ter transformado todo o difícil e longo caminho em algo mais prazeroso de ser vivido.

Agradeço ao meu orientador, Prof. Nelson Francisco Favilla Ebecken, pela orientação e incentivo mesmo nos momentos que pareciam mais difíceis, fazendo com que todas as dúvidas se esclarecessem.

Agradeço à minha sogra Lila, meu sogro Maurílio e meus tios, Yeda e Jaime, que praticamente me adotaram e são pessoas maravilhosas.

Agradeço às minhas irmãs Isabele, Michele e Gisele pelo amor e carinho que sempre tiveram comigo.

Agradeço aos colegas de curso e funcionários do programa pelo companheirismo, apoio e incentivos sempre presentes ao longo dessa trajetória.

Agradeço minha equipe da Petrobras, Max, André Luiz, Luís Cogliatti, Dalila, Alexandre, Sabrina, Diogo e Righetto pelo apoio nos momentos difíceis.

Agradeço aos meus chefes Flávio Gondin e Luis Antonio Pereira de Araújo, pelo incentivo e liberação de horas para que eu pudesse realizar este trabalho.

Por fim, agradeço a Deus por tudo que tenho e por ter me suprido a ótima saúde e coragem de que tanto precisei nesta gloriosa etapa da minha vida.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

MODELO COMPUTACIONAL PARA MINERAÇÃO DE TEXTO E ANÁLISE DE QUESTÕES DE CONCURSOS

Jorge da Cunha Morgado Júnior

Dezembro/2008

Orientador: Nelson Francisco Favilla Ebecken

Programa: Engenharia Civil

Este trabalho apresenta uma proposta metodológica para a análise de informações não-estruturadas contidas em questões de concursos públicos realizados no período de 2000 e 2006 no Brasil. Para realizar o estudo foram utilizadas as seguintes tarefas envolvidas no processo de mineração de textos: coleta de dados, pré-processamento textual, exploração dos dados, classificação, agrupamento e análise de *links*. Para a etapa de pré-processamento, desenvolveu-se um sistema específico para tratar documentos em português. No processamento dos textos buscou-se estabelecer conexões entre os registros, através da análise de *links* nos documentos, com o propósito de reconhecer os padrões de relações existentes nas questões.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

COMPUTER MODEL FOR TEXT MINING AND ANALYSIS OF EXAM QUESTIONS

Jorge da Cunha Morgado Júnior

December/2008

Advisor: Nelson Francisco Favilla Ebecken

Department: Civil Engineering

This work presents a methodology for the analysis of non-structured information contained on issues of procurement made in the period 2000 to 2006 in Brazil. To conduct the study was used the following tasks involved in text mining: data collection, pre-processing text, use data, classification, clustering and link analysis. For the pre-processing stage, it has developed a specific system to handle documents in Portuguese. In the processing of texts trying to establish connections between records, through the link analysis to documents in order to recognize the patterns of relationships in existing issues.

Índice

Lista de Figuras

Lista de Tabelas

Lista de Abreviações e Siglas

SGDB	-	Sistema Gerenciador de Banco de Dados
ODBC	-	Open Data Base Connectivity
KDD	-	Knowledge Discovery Database
KDT	-	Knowledge Discoveryin Texts
KE	-	Knowledge Engineering
TG	-	Teoria dos Grafos
MVC	-	Model View Controller
ML	-	Machine learning
NB	-	Naive Bayes
RSLP	-	Removedor de Sufixo da Língua Portuguesa
ASCII	-	American Standard Code for Information Interchange
PA	-	PolyAnalyst
IDE	-	Integrated Development Environment
JSP	-	JavaServer Pages
SVM	-	Support Vector Machine
PDF	-	Portable Document Format
HTML	-	Hyper Text Markup Language
XML	-	Extensible Markup Language
URL	-	Uniform Resource Locator
SQL	-	Structured Query Language
CSV	-	Comma-Separated Values
MD	-	Mineração de Dados
MT	-	Mineração de Textos

1 Introdução

Métodos de recuperação de textos sempre foram utilizados para organizar documentos, porém, com o aumento do volume de textos que vem ocorrendo, principalmente pela digitalização do conteúdo e pela Internet, técnicas de tratamento automático de textos começaram a se tornar cada vez mais importantes para se encontrar e trabalhar a informação. Para solucionar esses problemas surge uma nova linha de pesquisa, a mineração de textos.

O emprego de técnicas de mineração de textos pode auxiliar na extração de informações não-triviais de repositórios de documentos não estruturados.

Pesquisas mostram que inúmeras novas páginas contendo textos são lançadas diariamente na Internet, assim como outros tipos de documentos (como relatórios de acompanhamento, atas de reuniões, históricos pessoais, etc.) são periodicamente gerados, atualizados e armazenados nas empresas. Por esses motivos, a importância da análise automática de textos é reconhecida em todos os segmentos que lidam com informação e conhecimento.

Adicionalmente, grande parte das atividades de tomada de decisões, hoje, envolve a análise de grandes volumes de texto. O processo decisório, que era orientado a análise de séries temporais e fluxo de dados desde os anos 70, está cada vez mais, principalmente das áreas estratégicas das empresas, orientado pelas informações (information-driven) [48].

Entretanto, o grande volume dessas informações faz com que as organizações e as pessoas tenham dificuldade para gerenciar adequadamente estas informações, principalmente as não-estruturadas. Durante muito tempo as técnicas de mineração de dados [49] cresceram para elaborar soluções para as informações estruturadas da empresa. Seguindo esse mesmo caminho, a área de mineração de textos surge para minimizar o problema de tratar dados não-estruturados, ajudando a explorar conhecimento armazenado em meios textuais e assim gerar algum tipo de vantagem competitiva.

A tarefa de gerar inteligência a partir da análise das informações capturadas e documentadas em textos livres já é realizada atualmente e demanda cada vez mais tempo dos participantes envolvidos devido ao volume cada vez maior a ser tratado. É exatamente nesse ponto que a mineração de textos pode contribuir.

1.1 Motivação

Nos últimos anos muitos concursos públicos foram abertos aos brasileiros com o intuito de preencher as vagas oferecidas na administração pública, seja direta ou indireta. Como um bom emprego no Brasil está cada dia mais escasso, esse tipo de seleção torna-se cada vez mais disputada e acirrada, sendo que as melhores vagas são conquistadas por candidatos muito bem preparados e que dedicaram um tempo precioso de estudo para alcançar seus objetivos. Alguns concursos trazem em seu edital o conteúdo programático das disciplinas muito parecido com o de outros concursos, fazendo com que o estudo, dedicado na preparação para uma prova, possa ser aproveitado para outra prova. O candidato preparado sabe, ao resolver inúmeras questões sobre um determinado assunto, que a probabilidade de aparecerem questões semelhantes em provas distintas é grande. A partir desse cenário surgiu o interesse em aplicar as técnicas de mineração de textos baseadas em questões de concursos públicos, escritos na língua portuguesa do Brasil, envolvendo as tarefas de categorização, agrupamento e análise de links.

1.2 Objetivo

O objetivo principal desse trabalho é estudar a importância da utilização, em texto, de técnicas computacionais de descoberta de conhecimento, buscando categorizar, agrupar, correlacionar os dados e descobrir as similaridades existentes entre as questões aplicadas pelas bancas organizadoras para concursos, utilizando a análise das questões de provas passadas. Além disso, busca-se fazer uma contribuição para a língua portuguesa do Brasil, com o desenvolvimento de uma ferramenta que minera os documentos, na etapa de pré-processamento.

1.3 Estado da Arte

Mineração de textos é um conjunto de técnicas e processos que descobrem conhecimento inovador nos textos. Ela está sendo empregada atualmente em projetos de diversas áreas, por exemplo, para descobrir fatos na área de inteligência competitiva. Em relação a concursos públicos, não foi encontrado nenhum trabalho científico que abordasse o tema, somente livros de auto-ajuda voltados para estudantes que pretendem

prestar algum tipo de concurso. A análise das ferramentas presentes no mercado privilegiou aplicações livres e comerciais, mas nenhuma com o propósito exclusivo de fazer mineração de textos de questões de concursos.

1.4 Relevância

A contribuição deste trabalho está baseada na exploração e descoberta do conhecimento em provas disponíveis na *Web*. Sua importância está na análise eficiente e a aplicação de técnicas de *text mining* na busca de conhecimento escondido sejam eles relevantes e/ou inesperados e no fato de incorporar técnicas de processamento voltadas para a língua portuguesa, princípio básico para o seu desenvolvimento e para um nicho específico contextualizado em provas de concursos.

1.5 Descrição do Trabalho

Este trabalho está dividido em seis capítulos da seguinte forma:

- O capítulo 1 apresenta a motivação, que serviu como base para a pesquisa de uma metodologia de descoberta de conhecimento em dados disponíveis em texto, bem como uma rápida identificação do desenvolvimento de um sistema aplicativo baseado nessa metodologia, e que atendesse ao problema de forma adequada;
- O capítulo 2 apresenta a teoria em que se baseia a mineração de textos e as técnicas que foram selecionadas na implementação do sistema;
- O capítulo 3 trata da metodologia proposta e define as etapas necessárias para desenvolver as tarefas de classificação, agrupamento e análise de links de uma coleção de documentos;
- O capítulo 4 trata do ambiente computacional desenvolvido, descrevendo suas funcionalidades e implementação, e das ferramentas utilizadas para a mineração dos dados;
- No capítulo 5, são identificados alguns estudos de casos realizados, o que demonstra a utilização do sistema, e os resultados obtidos.
- Finalmente, o capítulo 6 apresenta as considerações finais, definindo as conclusões do trabalho e sugestões de desenvolvimentos futuros que podem ser efetuados.

2 Mineração de Texto

Mineração de textos (MT), de maneira análoga à mineração de dados (MD), é o processo utilizado para descobrir conhecimento útil em uma coleção de documentos textuais através da identificação e exploração de padrões interessantes nesses documentos. Mineração de Textos é uma área multidisciplinar que incorpora técnicas de diversas áreas como Recuperação de Informação, Aprendizado de Máquina, Estatística, Lingüística Computacional, Extração de Informação, Visualização e especialmente Mineração de Dados. Neste capítulo será apresentada uma introdução ao processo de MT, bem como as principais etapas e tarefas relacionadas ao processo. Será dada ênfase à etapa de pré-processamento e às tarefas de categorização, agrupamento de bases textuais e análise de links, uma vez que esse é o objetivo de estudo desta dissertação.

2.1 Visão Geral

Mineração de textos refere-se ao processo não trivial de extração de padrões úteis e interessantes (conhecimento) a partir de um conjunto de documentos textuais não estruturados [3]. A MT inclui métodos inteligentes e ferramentas automáticas para auxiliar pessoas na análise de grandes volumes de textos a fim de minerar o conhecimento útil. Portanto, uma característica importante do processo de mineração de textos, tal como ocorre com a mineração de dados, é que o conhecimento extraído seja compreensível a humanos. A MT utiliza técnicas das áreas de Extração de Informação, Processamento de Língua Natural (PLN) e Recuperação de Informação, juntamente com algoritmos e métodos de *Knowledge Discovery in Database* (KDD), Aprendizado de Máquina e estatística.

Apesar de similar à MD, a MT é, no entanto, um processo mais complexo, pois trabalha com dados textuais que são inerentemente não estruturados e que, eventualmente, possuem ambigüidade [1]. Dörre [4] considera como o primeiro desafio da mineração de textos a manipulação e análise de informações textuais expressas em língua natural, que, inicialmente, não foram projetadas para serem processadas por computadores, mas para serem interpretadas por humanos. Pesquisas recentes na área de MT têm enfocado em problemas de representação de textos, classificação, clustering, extração de informação e busca ou modelagem de padrões. Nesse contexto, a seleção de

atributos relevantes e a influência do conhecimento do domínio possuem uma importante função.

2.2 Etapas do Processo de Mineração de Textos

O processo de mineração de textos inclui etapas similares ao processo de mineração de dados, no entanto os documentos textuais são o foco da análise. O processo de MT pode ser dividido em quatro etapas fundamentais: coleta de documentos, pré-processamento, extração de padrões e avaliação dos resultados (pós-processamento), como apresentado na figura 2.1.

Figura 2-1 - Etapas do Processo de Mineração de Textos

2.2.1 Coleta de Documentos

A primeira etapa do processo é a coleta de documentos. Esta etapa consiste na busca de documentos relevantes ao domínio de aplicação do conhecimento a ser extraído. Estes documentos estão disponíveis tanto na internet quanto nos livros.

Em muitos cenários de mineração de textos, os documentos relevantes, inicialmente, podem estar disponíveis ou ser parte da descrição do problema. Entretanto, em algumas aplicações é imprescindível o processo de coleta de documentos. Podem ser consideradas diversas fontes para a coleta, tais como livros (pelo uso de um scanner ou cujas páginas possuem cópias eletrônicas) e, especialmente, documentos provenientes da internet. Entretanto, é possível que os documentos coletados estejam em uma grande variedade de formatos e, dessa forma, pode ser útil converter tais documentos a um formato padrão como, por exemplo, o XML, antes de prosseguir para a etapa de pré-processamento.

A seguir é apresentado em maiores detalhes o pré-processamento de textos, a extração de padrões, bem como as tarefas de categorização, agrupamento e análise de links, focos de estudo desta dissertação.

2.2.2 Pré-processamento

Após a coleta de documentos é necessário formatar os documentos selecionados, pois eles serão submetidos aos algoritmos de extração automática de conhecimento. Essa segunda etapa denomina-se pré-processamento.

O pré-processamento de textos consiste em um conjunto de transformações realizadas sobre alguma coleção de textos com o objetivo de fazer com que esses passem a ser estruturados em uma representação atributo-valor. De modo geral, a etapa de pré-processamento tem por finalidade melhorar a qualidade dos dados já disponíveis e organizá-los. A etapa de pré-processamento demanda a maior parte do tempo do processo de extração de conhecimento. Além disso, ela exige planejamento e processamento, pois durante a transformação dos textos em formato estruturado existe a possibilidade de que informação intrínseca ao conteúdo dos textos seja perdida. Um desafio, nesse caso, é obter uma boa representação minimizando a perda de informação.

A etapa de pré-processamento em um processo de MT é, portanto, fundamental para o desempenho de todo o processo [5]. A Figura 2.2 apresenta o modelo do fluxo de tarefas realizadas no pré-processamento neste trabalho. A seguir, será apresentado em maiores detalhes.

Figura 2-2 - Etapas do Processo de Pré-Processamento

2.2.2.1 Remoção de caracteres indesejados

Em um texto existem sinais de pontuação que são indesejados dentro de uma análise. Podem também existir palavras com símbolos desconhecidos por erros de digitação, ou simplesmente caracteres esparsos sem significado semântico. É comum ainda a existência de caracteres matemáticos (“%”, “+”, “<”, etc), monetários (“$”, “€”, etc), números, caracteres de formatação (retorno de carro, newlines, caracter de tabulação, etc), entre outros que devem ser retirados do texto. Para executar esta tarefa, é necessário identificar os caracteres que se deseja excluir do texto, ou o contrário: identificar os que não devem ser excluídos e remover os demais. Isso é feito com o auxílio de uma lista em disco, ou acoplada no próprio fonte do método responsável. Um exemplo pode ser mostrado na sentença abaixo: “Os meios empregados para isso – técnicos, sociais, legais e políticos – são cada vez mais sofisticados.”. O trecho destacado contém caracteres que representam sinais de pontuação (“,” “-” “.”) e também o caractere de tabulação, utilizado para formatar o texto. Assim sendo, estes caracteres deverão ser removidos para dar início à próxima etapa. Depois da remoção de caracteres indesejados, o texto ficará da seguinte forma: “Os meios empregados para isso técnicos sociais legais e políticos são cada vez mais sofisticados”.

2.2.2.2 Transformação das letras (Case Folding)

É necessário integrar todos os caracteres restantes da etapa anterior dentro de uma mesma caixa para utilização posterior: letras maiúsculas ou letras minúsculas. No sistema desenvolvido, os textos foram todos passados para caixa baixa. As sentenças seguintes exemplificam:

“O Internet Explorer é o Navegador Mais Utilizado no Mercado”

A frase acima contém caracteres em caixa alta. Depois da etapa de integração da caixa, a sentença ficará como abaixo:

“o internet explorer é o navegador mais utilizado no mercado”

Diversas linguagens de programação acompanham em suas bibliotecas padrão mecanismos automáticos para a integração da caixa. Caso contrário, é possível alterar

facilmente a caixa fazendo simples cálculos utilizando o código ASCII de cada caractere.

2.2.2.3 Tokenização

A tarefa de tokenização consiste na identificação dos termos dentro do texto. Isso é feito utilizando o espaço (“ ”), que é o único caractere fora do alfabeto restante da etapa de remoção de caracteres indesejados, e que por natureza é um caractere separador de palavras.

Cada grupo de caracteres entre espaços é denominado token, que é o termo em si. O texto, que é visto como uma seqüência de caracteres, será transformado em uma coleção de tokens, como mostra o exemplo abaixo:

“Venda de celulares cairá em 2009”

Utilizando os espaços como separadores, os tokens (termos) da sentença acima serão os seguintes:

“venda”, “de”, “celulares”, “cairá”, “em”, “2009”.

Esta tarefa é muito importante uma vez que todas as etapas posteriores a este ponto utilizarão os textos como termos. Muitas linguagens de programação já acompanham bibliotecas e classes que auxiliam nesta etapa (biblioteca strtok.h em C, classe StringTokenizer em Java, etc).

2.2.2.4 Retirada de palavras desnecessárias (*Stopwords*)

No processo de análise é relevante a eliminação de palavras que não possuem importância significativa no texto, no intuito de limitar a quantidade de termos-índices, com a visão de se manter apenas os termos que representam realmente o contexto de cada documento.

São exemplos destas palavras artigos, preposições, conjunções, pronomes, tais como: de, assim, afim, agora, onde, outro, outros, ainda, a, o, que, vários, e, do, da, uns, em, um, para, é, etc.

Na maioria dos casos estas palavras possuem apenas a finalidade de conectividade entre termos (no auxílio à formulação de frases), não havendo assim a necessidade de adicioná-las na estrutura de índices.

Além dessas, outras palavras que são tidas como irrelevantes são aquelas que aparecem com freqüência na coleção de documentos. Desta forma, são consideradas incapazes de discriminar os mesmos, tornando-se desnecessária a permanência destas na estrutura de índices [6].

A remoção das *stopwords* é feita com o auxílio de uma lista, tipicamente em disco, denominada *stoplist*. Após uma palavra ser reconhecida no processo de indexação, sua presença na s*toplist* é verificada. Caso exista na lista de palavras negativas, ela não é adicionada ao índice.

Na Figura 2.3 é apresentado o documento resultante da etapa anterior, após ser validado por uma *stoplist*. Neste caso a lista de s*topwords* contém artigos, preposições, conjunções e algumas seqüências de caracteres que não devem ser adicionadas ao índice por possuírem freqüência elevada.

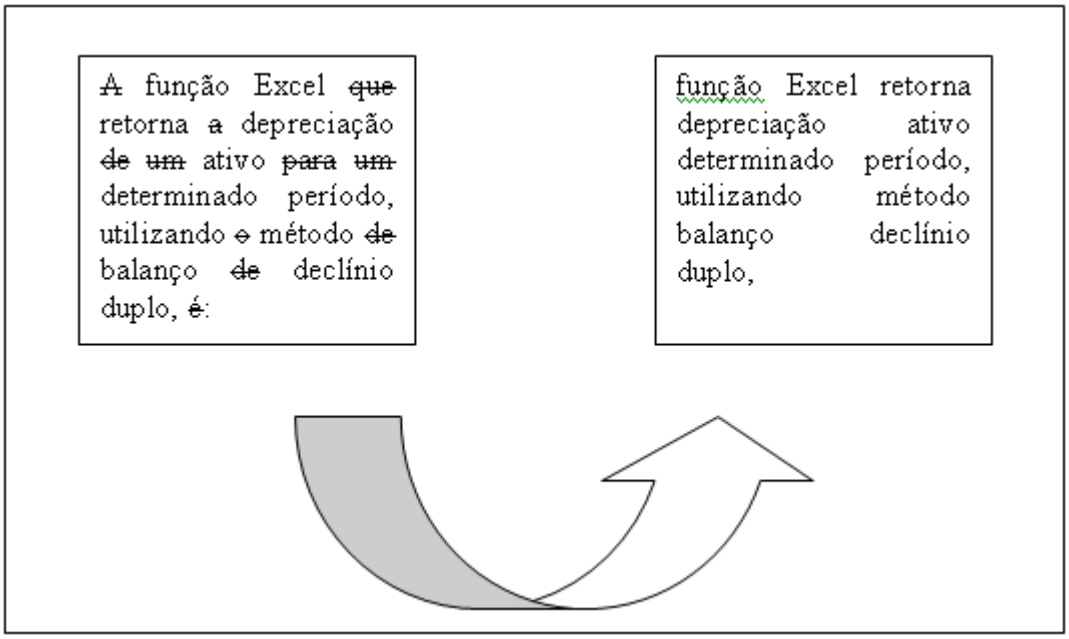


Figura 2-3 - Remoção de Stopwords

Portanto, *stopwords* são todas as palavras que influenciam negativamente no processo de análise. Assim, a sua existência nos textos implicaria na perda de

desempenho e qualidade nas etapas seguintes da tarefa de mineração de textos, por isso é necessária a execução deste processo antes das próximas etapas.

2.2.2.5 Redução ao menor radical de cada palavra (*Stemming*)

A tarefa de *stemming* consiste em reduzir um termo ao seu radical. Este processo é de grande importância para as etapas posteriores.

O processo de *stemming* é responsável por reduzir as diversas formas de um termo a uma forma comum (raiz) denominada *stem*. Um *stem* é um grupo natural de termos que compartilham interpretações semânticas iguais ou similares. Os algoritmos de *stemming* aplicam uma série de normalizações lingüísticas para remover sufixos de termos, ou inclusive mapear verbos para a sua forma no infinitivo. Por exemplo, os termos *speaks, spoke, speaking e spoken* são reduzidos ao seu radical único, o *steam speak*, que expressa o significado comum aos quatros termos.

Assim, termos com o mesmo radical poderão ser processados em conjunto, pois palavras cujo radical é o mesmo serão consideradas como equivalentes.

Com essa técnica, diminui-se o tamanho do dicionário, isto é, o número de termos distintos que representam o conjunto de documentos fica menor, resultando em redução no espaço de armazenamento necessário e no tempo de processamento.

Algoritmos de *stemming* empregam lingüística e são dependentes do idioma. Através da Figura 2.4 pode ser visualizada a aplicação do processo de *stemming*. Pode ser observado que o mesmo diminui consideravelmente a quantidade de termos, possibilitando, assim, uma melhora na etapa de criação da estrutura de termos-índices.

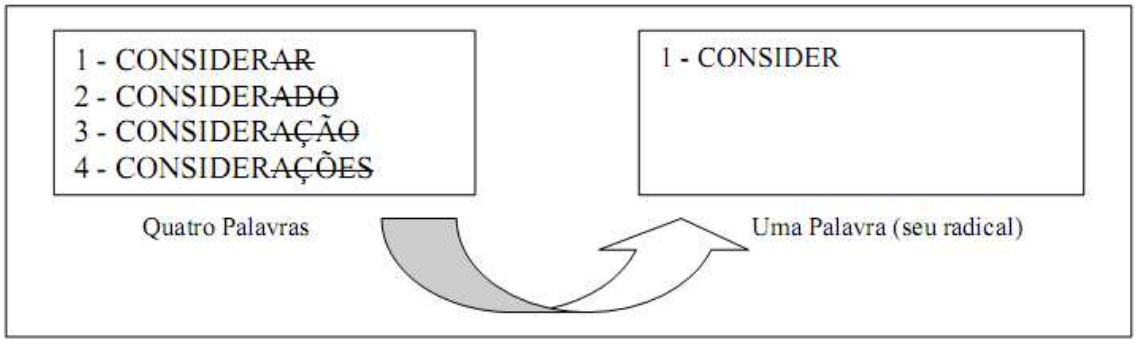


Figura 2-4 – Técnica *Stemming*

Existem vários algoritmos de *stemming* na literatura para a língua inglesa, dentre os mais conhecidos destacam-se os algoritmos do Porter [7] e o algoritmo do Lovins [8] que removem sufixos de termos. Como os textos abordados neste trabalho estão na língua portuguesa, será citado apenas o método mais conhecido desenvolvido para esta língua, o RSLP [9].

O stemming RSLP, ou *Portuguese Stemming*, ou Removedor de Sufixo da Língua Portuguesa é composto de oito passos que devem ser executados na ordem correta. A Figura 2.5 apresenta a seqüência que os passos devem seguir. Cada passo tem um conjunto de regras, cada uma destas regras deve ser processada em determinada ordem e somente uma regra em cada passo pode ser aplicada. O sufixo mais longo possível é sempre removido primeiro, por causa da ordem das regras no passo. Por exemplo, o sufixo plural “es” deve ser testado antes do sufixo “s”.

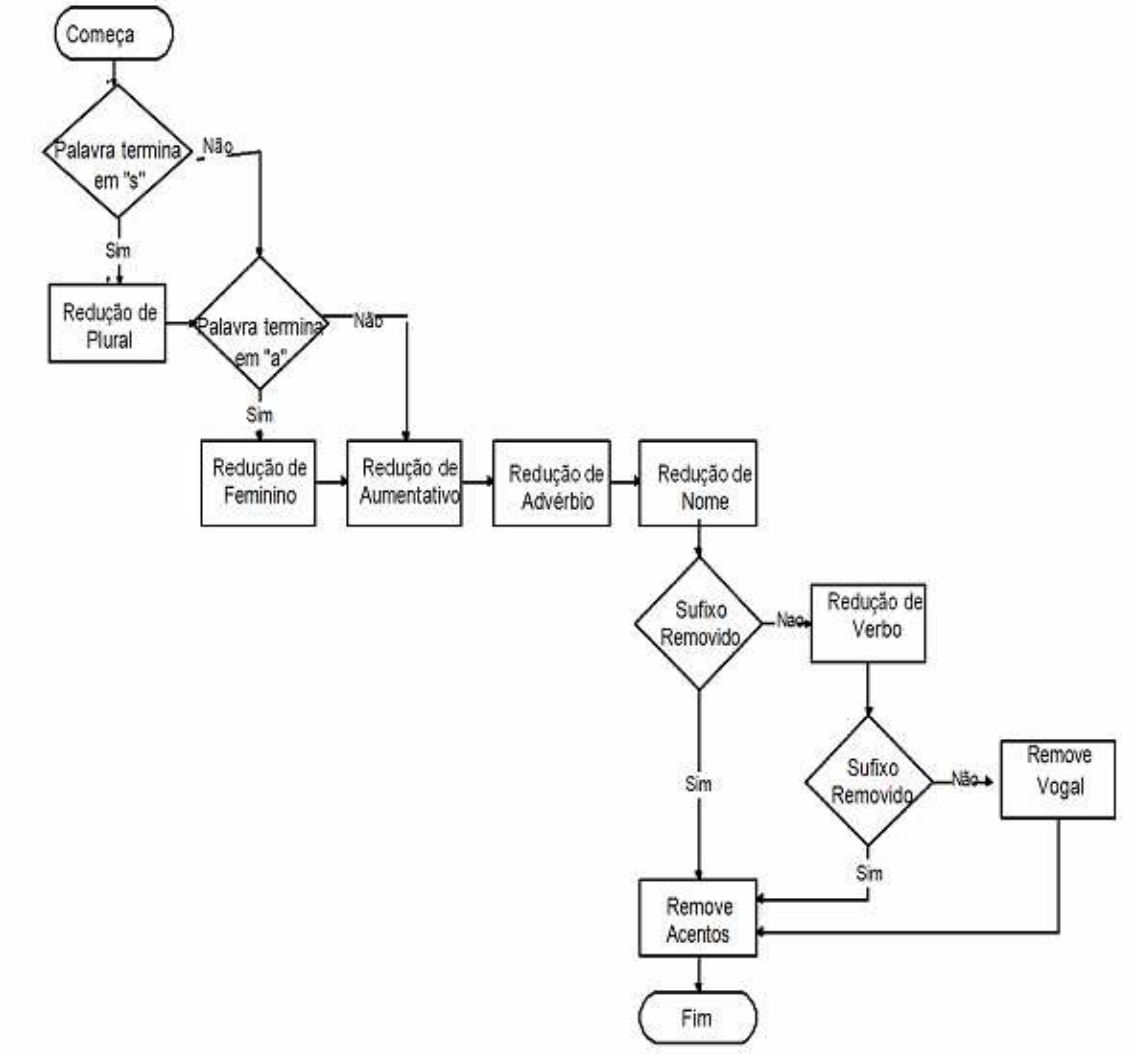


Figura 2-5 – Oito passos do RSLP [10]

Apesar de ser uma poderosa ferramenta para melhorar o desempenho da aplicação de mineração de texto, se o RSLP for mal trabalhado, os algoritmos de *stemming* podem prejudicar consideravelmente o resultado da análise. Por se tratar de um processo lingüístico, a fase de *stemming* pode apresentar falha, uma vez que é muito difícil englobar todas as exceções de um idioma em uma lista. Também é muito difícil estabelecer regras para remoção de sufixos que contemplem todas as palavras de uma língua. Há ainda a possibilidade de existirem palavras digitadas erroneamente e isto acarretará em um radical incorreto. Por estas razões existem duas falhas comuns na etapa de *stemming*:

- *Overstemming*: ocorre quando um termo for reduzido mais que o necessário, e o resultado então não expressará o radical correto da palavra. Ex: a palavra "gramática" foi reduzida ao radical "grama". O radical correto seria "gramat".

- *Understemming*: ocorre quando uma palavra for reduzida insuficientemente e o resultado não expressará o radical correto da palavra. Ex: "facilitador" foi reduzido ao radical "facilit", quando o correto seria o radical "facil".

Para contornar esse problema verifica-se o radical ao final da etapa de stemming para corrigir *understemmings* e *overstemmings*. Isto é feito com o auxílio de uma lista que indica o radical correto caso a palavra tenha sido reduzida a um radical incorreto contido na lista.

2.2.2.6 Dicionário de dados (Thesaurus)

Uma boa alternativa para melhorar os resultados de uma aplicação é utilizar um dicionário de dados que correlaciona palavras diferentes e comuns a uma única palavra em todo o texto, ou seja, montar uma relação de várias palavras para uma única palavra que possa substituí-las sem alterar o contexto. Como exemplo, podemos citar as palavras "rua", "avenida", "estrada", que poderiam ser associadas a uma única palavra que é "rua". Outro exemplo que se relaciona com uma aplicação jurídica, são as palavras "defere", "deferido", "procedente", "procede", que poderiam ser padronizadas como "deferido".

2.2.3 Extração de Padrões

A extração de padrões consiste na escolha e aplicação de um método de mineração. Sendo que para cada método escolhido existe um tipo diferente de extração de conhecimento de textos. Esses métodos precisam ser definidos para que as tarefas cabíveis possam ser executadas a fim de identificar padrões e relacionamentos entre os documentos. Dentre as principais tarefas relacionadas ao processo de MT, destacam-se:

Categorização: esta técnica visa identificar os tópicos principais em um documento e associar este documento a uma ou mais categorias pré-definidas [11]. Muitas das técnicas de extração de padrões utilizadas em categorização de documentos são similares às utilizadas em MD [12]; [13].

Agrupamento: esta técnica busca agrupar, em um ou mais grupos, um conjunto de exemplos de acordo com a similaridade ou dissimilaridade de seu conteúdo. A função de similaridade entre os exemplos é definida através dos termos que aparecem nos documentos.

Sumarização: é uma técnica que identifica os termos e frases mais importantes de um documento, ou conjunto de documentos, gerando a partir destes um resumo ou sumário [14]. Segundo Habn and Mani [15], esta tarefa pode ser considerada como o processo de redução da quantidade de texto em um documento, porém mantendo seus significados-chave.

2.2.4 Avaliação e Interpretação dos Resultados

A última etapa do processo de mineração de textos é responsável pela avaliação e interpretação dos padrões extraídos. Esta etapa visa constatar se o objetivo almejado foi alcançado, ou se todas ou algumas etapas do processo necessitam ser refeitas. Os padrões descobertos podem ser avaliados pelo usuário final, especialista do domínio e analista de dados, com o intuito de validar o conhecimento obtido [2].

O desempenho de um algoritmo de mineração de textos pode ser estimado em termos de várias medidas objetivas. Freqüentemente, os resultados obtidos pelos algoritmos de extração de informação são, estatisticamente, avaliados em termos das métricas *abrangência, precisão e F-measure* [16]; [17]; [18] e podem ser obtidas através das seguintes relações [19]:

Abrangência – consiste em avaliar a habilidade do sistema em recuperar os documentos mais relevantes para o usuário [20], medindo a quantidade de itens recuperados, dentre os relevantes na base de dados.

$$abrangência = \frac{n\_recuperados\_relevantes}{n\_relevantes} \tag{2.1}$$

Precisão – avalia a habilidade do sistema em manter os documentos irrelevantes fora do resultado de uma consulta [20].

$$precisão = \frac{n\_recuperados\_relevantes}{n\_total\_recuperados} \tag{2.2}$$

O exame das medidas de Precisão e Abrangência separadamente pode levar a uma má avaliação do sistema, pois em geral, ao se aumentar a Precisão de um sistema, diminui-se sua Abrangência. Portanto, há a necessidade de se investigar outras formas de avaliar o sistema de modo a obter a configuração mais adequada.

As medidas do ponto de equilíbrio (*breakeven point*) [21] e a medida do *F-measure* [19] combinam os valores de Precisão e Abrangência de modo a se obter o desempenho geral do sistema. O ponto de equilíbrio já foi bastante utilizado em sistemas de categorização: através do traçado dos vários pares de Precisão e Abrangência obtidos, pode-se obter por interpolação o ponto de equilíbrio, isto é, o ponto em que a Precisão e a Abrangência se igualam. A medida do *F-measure* foi definida por Rijsbergen, [19] e permite um balanceamento entre os valores de Precisão e Abrangência através da expressão:

$$F = \frac{(\beta^2 + 1) * P * C}{\beta^2 * (P + C)} \tag{2.3}$$

Onde β é o parâmetro que permite a atribuição de diferentes pesos para as medidas de Precisão (P) e Abrangência (C), sendo 1 o valor geralmente adotado. O valor de F é maximizado quando a Precisão e a Abrangência são iguais ou muito próximas, de modo que nesta situação, por definição, o valor do *F-measure* é o próprio valor da Precisão ou da Abrangência, que por sua vez, é o ponto de equilíbrio do sistema.

No entanto, as técnicas e ferramentas para visualização de dados visam melhorar a compreensão dos resultados obtidos e a comunicação entre os usuários [22], tornando-se instrumentos indispensáveis ao processo de MT. Segundo Fayyad et al. [22], poderosas ferramentas de visualização que gerem diversas formas de visualização (árvores, regras, gráficos 2D, 3D) combinadas com técnicas de mineração de textos podem melhorar muito o processo de MT.

Entretanto, o conhecimento de um especialista é importante ao longo do processo de extração de conhecimento em mineração de textos, e tem como objetivo auxiliar a resolver situações de conflito, indicando os melhores caminhos e complementando informações.

As diferentes medidas de avaliação de resultados podem fazer uso de variadas técnicas de MT. A seção que segue apresenta mais detalhadamente as tarefas de MT adotadas nesta dissertação: categorização e agrupamento de bases textuais.

2.3 Tarefas de Mineração de Textos

Cada tipo de tarefa de MT extrai um tipo diferente de informação dos textos independente da tarefa escolhida. O objetivo é descobrir conhecimento útil e inovador dos textos de forma a auxiliar os usuários na obtenção de informações relevantes. Na seção que segue estas tarefas serão mais detalhadas.

2.3.1 Categorização de Textos

A categorização é um processo que visa à identificação de tópicos principais em um documento e a sua associação a uma ou mais categorias pré-definidas [11]. Esse processo tenta identificar a qual domínio ou categoria um determinado documento pertence.

A categorização de documentos textuais é uma tarefa tipicamente realizada por humanos, especialistas no domínio de interesse, que lêem os documentos e os classificam em categorias temáticas pré-definidas. Porém, esse processo manual é muito custoso. O site de buscas *yahoo*, por exemplo, possui mais de duzentos especialistas para rotular manualmente ou categorizar os sites, que recebe centenas de páginas diariamente [25].

Então, com o número de documentos potenciais para exame por um humano cada vez mais excedendo a quantidade de documentos que uma pessoa precisa ler, técnicas como a categorização automática de textos (ou classificação de textos) tem testemunhado um interesse crescente nos últimos tempos. Esse interesse também se deve ao fato desta técnica estar conseguindo atingir níveis de desempenho competitivos em relação às classificações realizadas por profissionais treinados [12].

Pode-se afirmar que a categorização automática de textos possui diversas vantagens em relação à categorização manual. A categorização feita manualmente por pessoas, segundo Wives [26], acarreta problemas de atraso ou imprecisão (onde a pessoa que indexa pode não categorizar corretamente a informação, colocando-a em uma categoria diferente da categoria que a informação realmente pertence).

O estudo de Categorização de Textos utilizando computadores data do inicio de 1960, porém, somente no início da década de 90 tornou-se um sub-campo maior na disciplina de sistemas de informação, graças ao aumento do interesse por aplicativos e à disponibilidade de hardwares mais poderosos. [27]

Até a década de 80 a abordagem mais popular para a categorização de textos, pelo menos na comunidade “operacional” (aplicações do mundo real), era a Engenharia do Conhecimento (*Knowledge Engineering* - KE), que consiste na definição manual de um conjunto de regras codificadas por um especialista, de como classificar os documentos sobre as categorias fornecidas. Na década de 90 esta abordagem foi perdendo popularidade (especialmente na comunidade de pesquisa) para a abordagem de aprendizado de máquina (ML), onde um processo geral indutivo constrói automaticamente um classificador, que aprende de um conjunto de documentos pré-classificados as características da categoria [12].

As vantagens da abordagem do aprendizado de máquina são a acurácia comparável com a alcançável por especialistas humanos, e a economia da mão de obra do especialista, assim como nenhuma intervenção de um engenheiro de conhecimento ou especialista do domínio é necessária para a construção do classificador ou para encaminhar a um conjunto diferente de categorias.

As categorias são escolhidas para corresponder aos tópicos ou temas dos documentos. O principal objetivo da categorização de textos é a organização automática. Alguns sistemas de categorização (ou categorizadores) retornam uma única categoria para documento, enquanto outros retornam categorias múltiplas. Em ambos os casos, um categorizador pode retornar nenhuma categoria ou algumas categorias com

confiabilidade muito baixa. Nestes casos, o documento é normalmente associado a uma categoria rotulada como “desconhecida”, para posterior classificação manual [28].

Em resumo, a categorização de textos é uma ferramenta utilizada para classificar automaticamente um conjunto de documentos em uma ou mais categorias preexistentes, não tendo outra finalidade senão recuperar a informação. Na figura 2.6 é ilustrada a representação do processo de categorização.

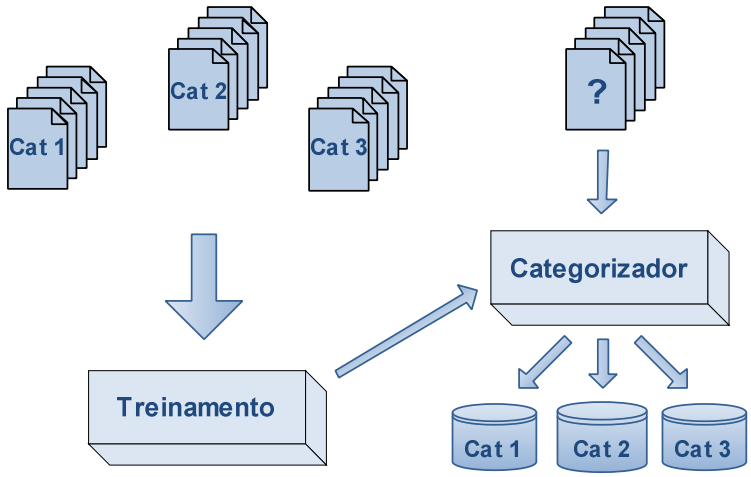


Figura 2-6 - Processo de Categorização

Os vários algoritmos de categorização de documentos que têm sido desenvolvidos dividem-se em duas categorias gerais. A primeira categoria contém algoritmos de aprendizado de máquina [24] tais como árvores de decisão, conjuntos de regras, classificadores baseados em exemplos, classificadores probabilísticos, *support vector machines (SVM)*, etc., que tem sido usados diretamente, ou adaptados para uso no contexto de dados em forma de documentos. A segunda categoria contém algoritmos de categorização especializados e desenvolvidos a partir da área de recuperação de informação. Exemplos de tais algoritmos incluem feedback de relevância, classificadores lineares, classificadores de conjuntos de exemplos genéricos, etc.

No presente trabalho são realizados experimentos com os categorizadores Bayesiano e SVM.

2.3.1.1 Classificador Bayesiano

Dentre os vários métodos utilizados para classificar textos, destaca-se neste trabalho o método Bayesiano. O classificador Naive Bayes (NB) é provavelmente o

mais utilizado para classificação de documentos. É baseado no Teorema de Bayes, que é usado na inferência de estatística para atualizar estimativas da probabilidade de que diferentes hipóteses sejam verdadeiras, baseando-se nas observações e no conhecimento de como essas observações se relacionam com as hipóteses. O Teorema de Bayes pode ser representado da seguinte forma:

$$p(X|Y) = \frac{p(Y|X)p(X)}{p(Y)} \tag{2.4}$$

O teorema determina que, para eventos X e Y, a probabilidade de X, dado que Y tenha acontecido denotado por p(X|Y) é igual à probabilidade de Y, dado que X denotado por p(Y|X) tenha acontecido, vezes a probabilidade de X acontecer p(X), dividido pela probabilidade de Y acontecer p(Y) [30].

Este teorema é uma das pedras angulares da estatística das probabilidades combinadas e é largamente utilizado em áreas à primeira vista pouco relacionadas, como Medicina e Informática.

Na primeira, o paradigma embasado em evidências é todo construído em cima do Teorema de Bayes. Baseado na experiência acumulada de exames e testes para tentar diagnosticar uma doença, o médico enquadra seus pacientes e pode estimar qual a probabilidade de que uma dada doença esteja se manifestando. Ou seja, dada uma probabilidade inicial (por exemplo, o paciente é fumante) e aplicado um exame em que, se sabe, há uma probabilidade de falso-positivos e falso-negativos (por exemplo, uma biópsia de pulmão), o médico sabe qual a probabilidade resultante de aquele paciente ter a doença (por exemplo, câncer de pulmão).

Na informática, muitos dos sistemas de classificação automática são baseados no Teorema de Bayes. Inicialmente, o sistema é treinado aceitando entrada de humanos que dizem que uma dada entrada pertence a determinado grupo. Com o tempo o sistema acumula um grande banco dessas informações e, aplicando o Teorema de Bayes, consegue estimar a probabilidade de cada novo dado de pertencer a cada grupo já classificado [31].

A classificação de Naive Bayes é feita utilizando dados de treinamento para estimar a probabilidade de o documento pertencer a cada classe. São utilizados os termos do documento com seus respectivos pesos para realizar a classificação. Para cada termo do documento é calculada a probabilidade de o mesmo pertencer à categoria [32]. É feita uma combinação das probabilidades levando em consideração o peso dos

termos de acordo com a regra de Bayes. Se o resultado for maior que determinado coeficiente, o documento é incluído na categoria.

2.3.1.2 Classificador SVM

O classificador *Support Vector Machines* ou simplesmente SVM é a junção de dois tipos de classificadores: hierárquico e não-hierárquico de documentos, apresentado em [36][37], que usa uma estrutura de dados baseada em árvore de categorias, com poucos níveis, preferencialmente dois, e os resultados das buscas estão presentes nas folhas desta árvore.

O processo de classificação de textos baseado em SVM [13] é incremental e seletivo, orientado para a organização das categorias em hierarquias. Neste tipo de classificação, características particulares dos documentos, tais como tags e links são considerados pelo método.

Os documentos são divididos em dois conjuntos definidos como base de treinamento e de teste. A base de treinamento é usada para o algoritmo de classificação obter as características das categorias da coleção. A base de teste valida o desempenho do classificador, determinando as categorias as quais os novos documentos pertencem [23].

Na fase de análise, o desempenho do algoritmo é medido de acordo com o resultado obtido na classificação original dos documentos, o que é feito geralmente através de classificação manual.

Em sua forma linear, SVM gera um hiperplano que separa um conjunto de amostras positivas de um conjunto de amostras negativas. Em termos geométricos este método pode ser visto como uma tentativa de busca da melhor superfície s i, no conjunto de todas as superfícies S_1 S_2 .... S_n no espaço r-dimensional que separa os exemplos de treinamento positivos dos negativos (superfície de decisão) [12]. A melhor superfície S_i separa os exemplos positivos dos negativos pela mais larga margem possível.

2.3.2 Agrupamento

O agrupamento tem sido amplamente utilizado em padrões de análises exploratórias de dados, tomada de decisão e situações de aprendizado de máquina,

incluindo mineração de dados, recuperação de informação, segmentação de imagens e classificação de padrões.

O agrupamento é uma técnica de aprendizado não supervisionado, portanto não existem rótulos pré-determinados para os padrões de treinamento. Técnicas de agrupamento buscam agrupar em um ou mais grupos, um conjunto de exemplos similares entre si. Ao final do processo, cada grupo irá conter um representante que permite identificar aquele grupo. O resultado do agrupamento é dependente da medida de similaridade (critério) adotada.

Theodoridas [33] define agrupamento como sendo um dos processos mais primitivos do ser humano, e desenvolvido para auxiliar a processar grandes quantidades de informações e agrupá-las de acordo com seus atributos comuns.

Conforme descrito em Jain and Dubes [34], para se executar uma tarefa de agrupamento normalmente os seguintes passos devem ser seguidos:

- Selecionar as características;
- Estabelecer uma medida de similaridade;
- Determinar um critério de agrupamento;
- Escolher um algoritmo de agrupamento;
- Validar os resultados do agrupamento;
- Interpretar os resultados.

Faz-se necessário selecionar características devido aos métodos de agrupamento estarem de alguma forma baseados em alguma medida de similaridade entre os exemplos. Portanto, para que os exemplos possam ser agrupados, é preciso identificar as características dos exemplos e agrupá-los de acordo com a quantidade de características similares entre eles.

A segunda etapa do processo de agrupamento consiste em analisar todos os exemplos com o objetivo de selecionar semelhanças entre os exemplos. O grau de semelhança entre os exemplos é dado, em geral, por uma fórmula de similaridade. Sendo que essa fórmula analisa todas as características semelhantes que os exemplos possuem e retorna um valor, indicando um grau de similaridade entre os exemplos.

Conforme descrito em Cole [35], para agrupar objetos de acordo com sua similaridade, deve-se definir uma medida de quão próximos dois objetos estão, ou quão bem seus valores se comparam. Uma pequena distância entre os objetos deve indicar uma alta similaridade.

Como a distância é uma função que envolve somente atributos relativos a dois exemplos, o processo de agrupamento baseado em similaridade pode ser feito de forma simples e sem necessitar de nenhum tipo de conhecimento sobre o assunto tratado. Entretanto, devido a variedade de tipos e escalas dos valores dos atributos, a medida de similaridade deve ser escolhida cuidadosamente. As características utilizadas para definir similaridade são especificadas e combinadas em uma medida a ser calculada para todos os exemplos.

Na literatura existem várias medidas de similaridade, tais como Euclidiana, Manhattan e Minkowski. Dentre essas medidas [35] destacam-se a distância Euclidiana, que é a mais utilizada.

Depois de concluída a etapa de cálculo de similaridades (ou distância), tem-se uma matriz que indica os valores de similaridade entre os exemplos. Com base nesta matriz é possível identificar os grupos de exemplos, especificando algum tipo de regra de relacionamento entre os mesmos e então escolher um dos tipos de agrupamentos (tais como, *K-means*, *Cobweb*, etc.). Por fim, os resultados do agrupamento são validados e interpretados com o intuito de verificar o desempenho do processo.

Fundamentalmente, existem dois tipos de agrupamento quando nos referimos à MT: um baseado em termos e outro em documentos. No primeiro tipo, os grupos de termos similares são identificados, com o intuito de construir um dicionário de termos que definam assuntos similares. Em contraste, o agrupamento por documentos visa identificar os documentos de assuntos similares e alocá-los em um grupo. Esse método é extremamente útil quando não se tem uma idéia dos assuntos (das classes) tratados em cada documento e deseja-se separá-los por assunto.

Logo, o agrupamento tem como finalidade identificar os exemplos (tais como, documentos ou termos) que apresentem características em comum, agrupando-os em subconjuntos de exemplos similares, sendo que estes possam ser os mais variados possíveis antes de agrupados.

2.3.2.1 Algoritmo *Suffix Tree Clustering*

Para a geração dos Clusters deste trabalho, utilizou-se uma variação do algoritmo *Suffix Tree Clustering*, que foi apresentado originalmente em [46]. Esse algoritmo apresenta algumas qualidades, entre elas a sua velocidade de processamento, que é próxima de um processamento linear, ou seja, o tempo é proporcional ao número de

registros. Ele apresenta resultados de fácil interpretação por parte dos usuários. Também possui mecanismos de identificação das frases.

As frases apresentam a vantagem de ter um poder descritivo mais alto que os termos isolados. Daí, elas se prestam melhor para descrever o conteúdo dos grupos para os usuários, e de uma maneira mais concisa.

O processo envolve dois passos principais: no primeiro, o algoritmo faz uma busca por registros que compartilham frases; no segundo, ele agrupa os documentos a partir da freqüência da ocorrência dessas frases.

2.3.3 Análise de Links

Em várias situações é possível identificar relacionamentos entre indivíduos, lugares, objetos ou mesmo conceitos, tais como os que acontecem entre pessoas que conversam por meio de ligações telefônicas, entre documentos de hipertextos ou entre pesquisadores, quando estes são co-autores em publicações. Outros exemplos podem ser vistos em companhias aéreas e de transportes que ligam cidades, Estados e países. Também como exemplos de relacionamentos têm-se as citações bibliográficas e até mesmo um grupo de pessoas que se conhecem. Esses relacionamentos podem conter informações úteis e, para estudá-los, surgiu a técnica de análise de links.

Trata-se de uma técnica de Mineração de Dados que tem por finalidade revelar a estrutura e o conteúdo de um conjunto de informações por meio de unidades (entidades ou objetos) interconectadas.

A análise de links é baseada em Teoria dos Grafos (TG). Uma entidade (chamada de vértice ou nó) pode ser uma pessoa, um lugar, um documento, um objeto qualquer ou até mesmo um conceito. A conexão entre as unidades (chamada de aresta ou arco) representa a relação entre essas unidades.

Segundo Jensen [50], muitos conjuntos de dados podem ser representados naturalmente como coleções de objetos ligados. Por exemplo, coleções de documentos podem ser representadas como documentos conectados por citações e referências de hipertexto. Similarmente, organizações podem ser representadas como pessoas conectadas por relacionamentos sociais e/ou padrões de comunicação. Outros exemplos são: coleção de dados de chamadas telefônicas (Figura 2.7), dados de transações financeiras entre contas bancárias, observações de encontros individuais, seus endereços, e outras interações comerciais e sociais.

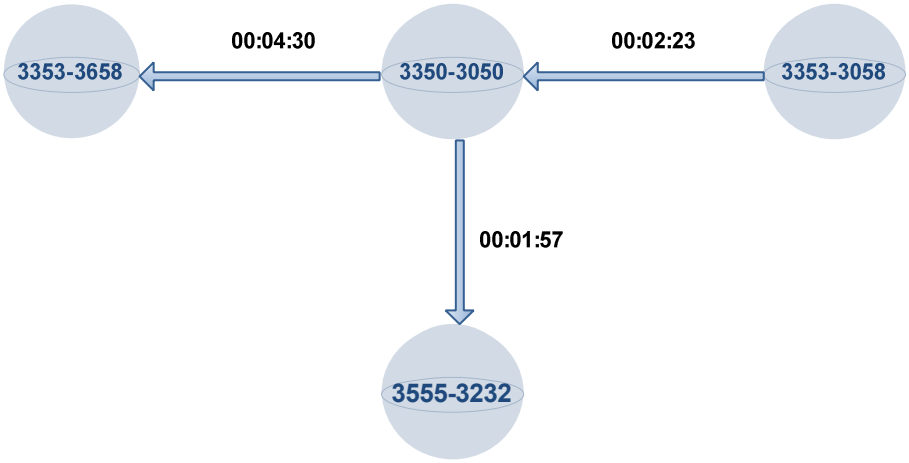


Figura 2-7 - Ligações telefônicas relacionando números de telefones

Na Figura 2.7 tem-se um exemplo de um grafo representando chamadas telefônicas: cada vértice representa um número de telefone; os arcos entre os vértices representam uma chamada; a orientação das setas indica quem originou a ligação e os valores de cada link indicam a duração de cada ligação. Esse tipo de representação de dados é também facilitado pela crescente disponibilidade de base de dados e sistemas hipertextos. A representação na forma de relacionamentos está no âmbito da tarefa de modelagem de dependência em KDD. A natureza das bases de dados disponíveis na maior parte das organizações (i.e. bases relacionais) facilita a identificação de relacionamentos entre elementos de um domínio. Bases relacionais, orientadas a objetos e documentos hipertexto têm sua estrutura adequada ao tratamento de dependência. É nesse contexto que a análise de links assume especial relevância.

A análise de tais dados, que pode ser realizada através de análise de links, está se tornando importante em diversos campos: investigações criminais, detecção de fraudes, epidemiologia, recuperação da informação, etc. Alguns dados ligados podem ser simples, mas volumosos (e.g. chamadas de telefone), com uma uniformidade de nós e tipo de links e com bastante regularidade. Outros dados podem ser extremamente ricos e variados, apesar de escassos (e.g. dados sobre investigação criminal), com elementos que possuem muitos atributos de domínio específico e também com valores que podem mudar com o tempo.

Segundo Goldberg e Senator [51], para descobrir informações úteis e interessantes sobre uma pessoa específica ou sobre grupos de pessoas, é necessário primeiro identificar precisamente os indivíduos representados no banco de dados. Esse processo

de tornar não ambíguo e de combinar a informação de identificação em chave única, as quais se referem a indivíduos específicos, é chamado de consolidação. E para, descobrir informações úteis, tais como anomalias que podem indicar fraude, freqüentemente se requerem a construção de redes de indivíduos relacionando transações e um padrão de atividade. O processo de criar essas redes é chamado de formação de links, o qual pode ser usado em domínios em que há relacionamentos escondidos. A idéia de consolidação e formação de links foi apresentada na análise de pessoas e seus relacionamentos, mas esses dois conceitos podem ser aplicados a outros tipos de entidades.

Um exemplo de formação de links é visto em Pinheiro e Sun [29]. Os autores desenvolveram um método para relacionar registros de diferentes bancos de dados. Nesse método foi usada uma medida de similaridade entre duas palavras. Assim, foi possível relacionar registros do tipo texto que não tinham um identificador único em comum. Além das técnicas para consolidação e formação de links, também se estuda como examinar, modificar, analisar, pesquisar e mostrar essas redes. Um de seus principais objetivos é a apresentação visual das relações para que o usuário possa melhor compreender o significado das inter-relações e, ainda, ver relações desconhecidas. Contudo, há limitações para a apresentação visual das informações. Nesse sentido, Grady, Tufano e Flanery Junior [41] afirmam que há a necessidade de novas técnicas para organizar a exibição das informações, e afirmam que vêm sendo realizadas pesquisas na área de HCI (*Human Computer Interaction*) com o objetivo de desenvolver interfaces mais adequadas para apresentação das informações ao usuário.

Como afirma Lyons [42], as questões a seguir são freqüentemente consideradas em Link Analysis:

- Quais nós são chaves ou centrais (hubs) na rede formada?
- Quais links podem ser reforçados para aumentar a eficiência das operações da rede?
- É possível descobrir links ou nós não detectados a partir dos dados conhecidos?
- Existem similaridades na estrutura de partes da rede que podem indicar um relacionamento não conhecido?
- Quais são as sub-redes relevantes dentro de uma rede com muitos nós?
- Quais modelos de dados e níveis de agregação melhor revelam certos tipos de relações e sub-redes?

3 Metodologia Proposta

A metodologia proposta neste trabalho define as etapas necessárias para desenvolver as tarefas de classificação, agrupamento e análise de links de uma coleção de documentos.

Com o objetivo de obter enriquecimento prático sobre a metodologia de MT estudada, foi implementado um sistema de investigação textual contemplando a etapa de pré-processamento descrita no capítulo 2. O desenvolvimento deste sistema tem como objetivo atuar na etapa de pré-processamento de documentos e será alimentado da seguinte forma: Obtenção de conteúdos colhidos de provas de concursos, as quais serão armazenadas em uma base de dados central, e assim, será criada uma interface de acesso a essas informações, que permitirá ao usuário a execução de consultas definidas a partir de determinados contextos.

Na etapa de processamento será utilizado o software comercial PolyAnalyst 6.0 [47], para dar suporte às tarefas de classificação, agrupamento e análise de links entre os documentos. Os resultados da etapa de processamento serão analisados através dos módulos de visualização do software.

A metodologia é apresentada em cinco etapas, conforme a figura 3.1. A etapa identificada pelo número I requer envolvimento direto do usuário, a de número II são atividades desenvolvidas no escopo deste trabalho e a de número III utiliza software comercializado.

Cada etapa tem suas atividades definidas e é realizada seqüencialmente, mas a retroação entre as etapas é considerada sempre que for necessário melhorar os resultados finais.

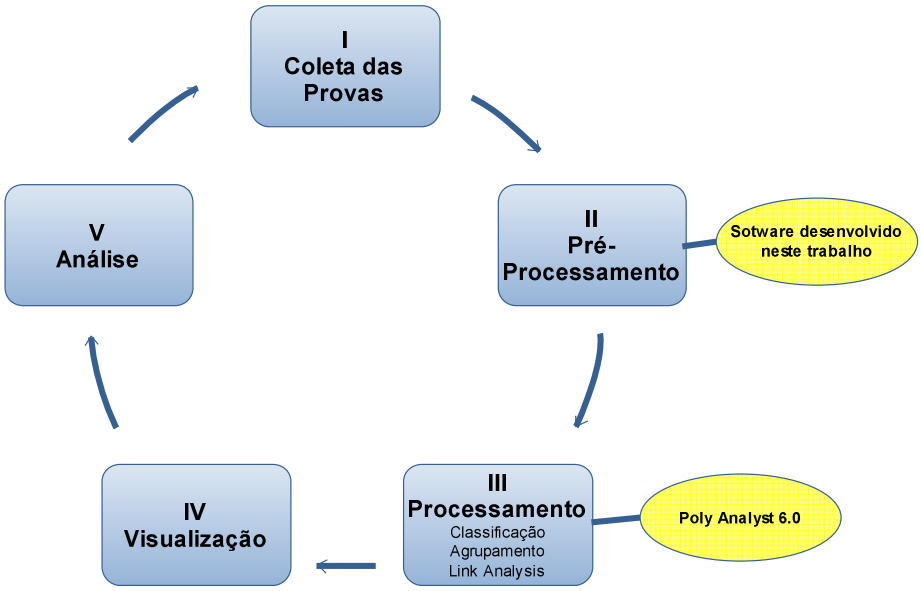


Figura 3-1 – Etapas da Metodologia proposta

3.1 Coleta de Documentos

As provas coletadas estão em formato PDF e se encontram espalhadas nos *sites* das instituições organizadoras de concurso público. Essas provas datam do ano de 2000 até o final de 2006. O primeiro passo foi converter o arquivo em PDF para o formato de texto e, em seguida, inserir os textos no banco de dados utilizado na aplicação. Serão apresentados dois estudos de casos neste trabalho. No primeiro, foram selecionadas aproximadamente 1400 questões de Informática já pré-classificadas, nos seguintes assuntos: Arquivos, Conceitos Básicos, Email, Excel, Hardware, Internet, Internet Explorer, Linux, Outlook, Redes, Segurança, Softwares, Teclas de Atalho, Windows e Word. No segundo estudo de caso foram selecionadas aproximadamente 3000 questões já pré-classificadas, da disciplina de Direito Administrativo, nos seguintes assuntos: Diversos, Órgãos e Agentes Públicos, Princípios Fundamentais, Poderes e Deveres, Servidores Públicos, Atos Administrativos, Licitação Pública, Responsabilidade Civil do Estado, Bens Públicos, Controle da Administração, Processo Administrativo, Serviços Públicos, Ética na administração e Contratos Administrativos.

3.2 Pré-Processamento

A fase de pré-processamento compreende a identificação de todos os termos existentes nos documentos, bem como o armazenamento de várias informações referentes aos termos que serão utilizados na fase seguinte, pelas ferramentas de descoberta de conhecimento. O objetivo desta fase é identificar cada termo existente no documento. Sendo assim, são executadas: a análise léxica, efetuada por um *parser* que identifica cada termo como sendo uma seqüência de letras e/ou dígitos; a análise morfológica, efetuada por um *stemmer*, que reduz cada termo ao seu radical; e a retirada de *stopwords* ou termos que não agregam valor, tais como: preposições, conjunções, artigos, etc.

Para esta fase desenvolveu-se uma ferramenta, com o objetivo de sistematizar todo o processo de preparação dos dados, desde o momento da coleta das provas até a saída do arquivo pronto para ser executado pelos algoritmos de mineração de textos. No capítulo 4 será apresentado todo o ambiente computacional utilizado neste trabalho.

3.3 Processamento dos Dados

Com os arquivos gerados pela aplicação, passamos para a etapa de processamento dos dados. Trataremos nesta dissertação das tarefas de categorização, agrupamento de documentos e análise de links.

3.3.1 Mineração de Textos

Existem vários softwares aplicados à mineração de textos, que podem ser classificados como livres ou comerciais. Foi utilizado o sistema PolyAnalyst para gerar os resultados para o proposto neste trabalho. A configuração do ambiente é apresentada na figura 3.2.

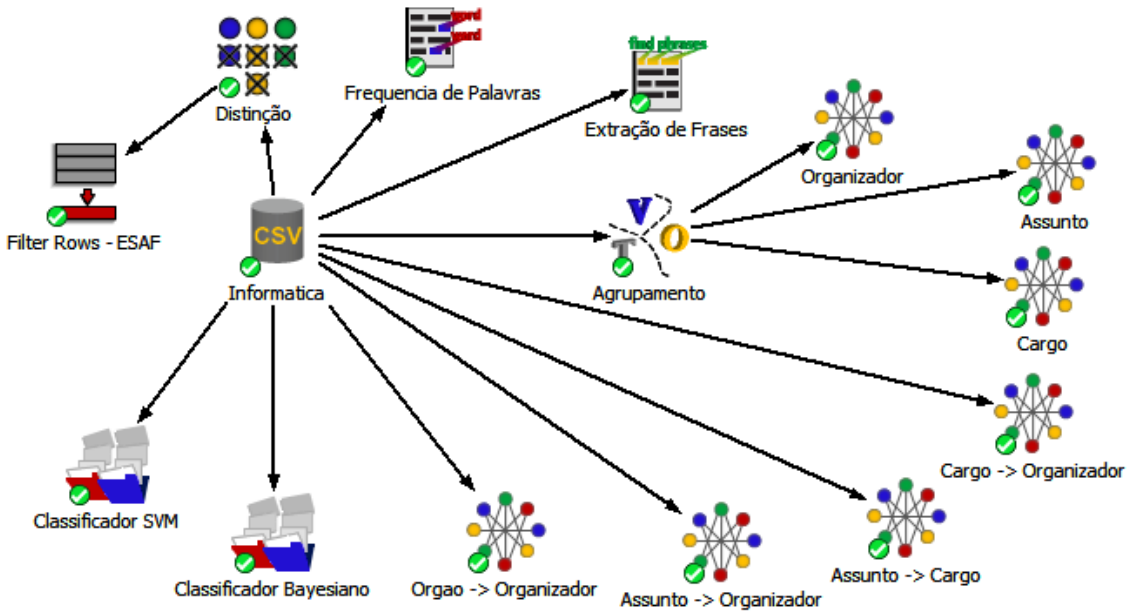


Figura 3-2 – Configuração do ambiente

No módulo de geração de arquivos, da aplicação desenvolvida, são gerados os arquivos de entrada para o PolyAnalyst. A figura 3.3 apresenta a estrutura do arquivo CSV gerado. Neste exemplo o *pipe(*"|") é o delimitador dos campos.

```
1  Questão|Assunto|Cargo|Orgao|Organizador|Data
2  hardware computadores pessoais iten subsequentes
   disquetes possuem mecanismo impedir gravacao exclusao
   arquivos mouses conecta computador porta paralela
   computador muitas impressoras tipo jato tinta utilizam
   cartucho separados cor preta cores disquetes atuais
   capazes armazenar superior milhao bytes enquanto
   scanners conectados traseira caixa cpu teclados pcs
   conectam-se diretamenta traseira monitor
   video|Hardware|Treinee Área 2|BANESE|CESPE|7/4/2002
```

Figura 3-3 – Arquivo de entrada para as tarefas de Mineração de Textos

3.3.2 Exploração dos Dados

Nesta etapa buscou-se conhecer melhor a base de dados através de análise estatística dos dados e a distribuição dos registros na base. Como primeira análise procurou-se entender a distribuição das questões por assunto e por organizadores. Em seguida foram encontrados os termos com maior freqüência nas questões. Analisou-se

também a freqüência das frases mais significantes. Esta técnica foi adotada para encontrar questões repetidas aplicadas em diferentes concursos.

3.3.3 Categorização

A primeira tarefa realizada após a exploração dos dados foi a categorização dos documentos. O objetivo maior desta etapa é encontrar informações úteis sobre as classes dos dados que possam estar escondidas dentro da base de dados. Além disso, buscou-se avaliar o melhor classificador para o contexto em que a base de dados está inserida. A classificação auxiliará o especialista do domínio a categorizar automaticamente os assuntos das questões, pois algumas questões não aparecem pré-classificadas, o que torna o trabalho do especialista mais demorado.

A classificação foi realizada em dois algoritmos disponíveis no sistema, o Bayesiano e o SVM.

3.3.4 Agrupamento

Após a categorização, aplicou-se as técnica de Clustering de textos. Clustering ou agrupamento é uma técnica de aprendizado não supervisionado, com o objetivo de encontrar uma estrutura em uma coleção de dados que não sejam pré-classificados. Esse procedimento é utilizado para agrupar documentos semelhantes [11]. Um sistema de agrupamento de texto deve desempenhar a tarefa fundamental de descobrir elementos compartilhados por documentos. O critério de similaridade entre os documentos é dado pela distância existente entre eles. Dois ou mais documentos irão pertencer ao mesmo grupo se estiverem próximos uns dos outros no espaço vetorial. A técnica de agrupamento aplicada na descoberta de conhecimento em provas irá identificar questões que têm um comportamento parecido, ou que podem ter alguma similaridade. O algoritmo utilizado pelo software é uma variação do algoritmo *Suffix Tree Clustering*, apresentado no capítulo 2. A figura 3.4 apresenta o resultado do algoritmo de agrupamento utilizado. Na figura observa-se o cluster de correio eletrônico achado na execução do algoritmo.

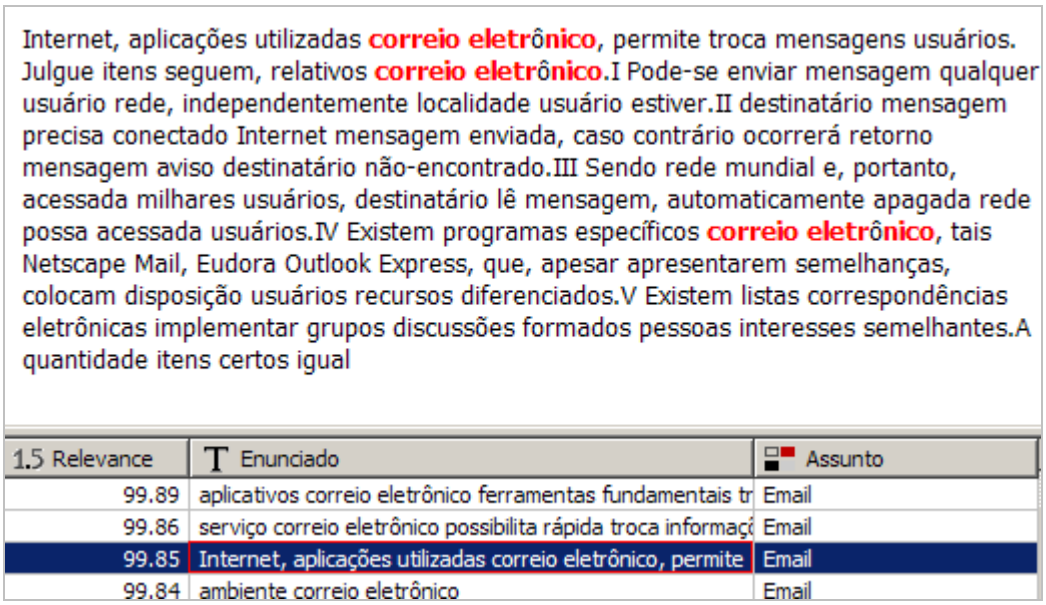


Figura 3-4 – Grupos formados pelo algoritmo

3.3.5 Análise de Links

A análise de links é considerada uma técnica de mineração de dados na qual é possível estabelecer conexões entre registros com o propósito de desenvolver modelos baseados em padrões de relações.

Neste trabalho esta técnica será aplicada no sentido de buscar relações dentro das questões que mostre ocorrências de correlacionamento entre as organizadoras de concurso, o assunto cobrado nas provas, o cargo pretendido e as questões, com enunciado e alternativas dentro de um mesmo campo. Detalhe da técnica de análise de links é abordado no capítulo 2 deste trabalho e o uso da ferramenta no capítulo 4. No capítulo 5 serão aplicados os estudos de casos envolvendo esta técnica.

3.4 Pós-Processamento

Essa etapa envolve a apresentação, a análise e a interpretação dos resultados, a fim de validar as descobertas obtidas na etapa anterior. Nela, o especialista em mineração de textos e o especialista no domínio da aplicação podem, a partir da avaliação dos resultados alcançados, definir novas alternativas de investigação dos dados. Nesta etapa busca-se responder algumas perguntas iniciais sobre a base de dados, que motivaram o desenvolvimento deste trabalho: Qual a probabilidade de uma questão

se repetir em outros concursos? Quais bancas examinadoras apresentam questões mais parecidas? Quais são os assuntos mais pedidos por determinada entidade organizadora?

3.4.1 Visualização

As ferramentas de visualização dos resultados apóiam a interpretação e a avaliação do conhecimento extraído. Elas podem disponibilizar técnicas de extração de informações específicas sobre os textos, como a recuperação de um termo dentro do seu contexto no documento, a extração de um sumário (técnica também conhecida como Sumarização), a seleção do assunto principal, a extração da lista de temas, ou ainda a extração de uma versão do documento com os termos especificados em destaque, entre outras. Essas técnicas eventualmente se utilizam de gráficos em duas e em três dimensões para a apresentação dos resultados, a fim de facilitar ou até mesmo viabilizar a sua compreensão e interpretação.

3.4.2 Análise

Cada etapa tem sua importância no desenvolvimento da metodologia e é avaliada em função das características dos documentos analisados para obter melhores resultados. A retroação, entre as etapas da metodologia, é realizada para a utilização de medidas de similaridades diferentes, e também para a inclusão de palavras na lista de *Stopwords* ou inclusão de termos no dicionário em busca de resultados melhores.

4 Ambiente Computacional

O ambiente computacional configurado para a utilização do software produzido para essa dissertação envolve ferramentas de código aberto, de maneira a permitir futuras modificações e evoluções que se fizerem necessárias. A seguir são apresentadas a arquitetura do sistema, as ferramentas utilizadas e os módulos de execução do sistema.

4.1 Arquitetura Geral do Sistema

Foi implementado um sistema de investigação textual contemplando a etapa de preparação dos dados descrita no capítulo 2. O ambiente de desenvolvimento do sistema é composto pelo sistema operacional Windows XP, pelo ambiente de programação *Netbeans* 6.0, pela linguagem de programação Java com *JSP* e pelo banco de dados *MySQL* 5.0.

O sistema foi desenvolvido de forma clássica, isto é, com acesso às funcionalidades principais através de uma barra de menus que levam a formulários de preenchimentos, consultas, ações e visualização de resultados.

A arquitetura do sistema segue o padrão de projeto denominado MVC (do inglês, Model-View-Controller) [43], conhecido por prover uma solução útil a sistemas complexos, a qual separa a camada de apresentação (interface gráfica) da camada de negócio (modelo), através de uma camada intermediária (Controller). Assim, alterações na interface gráfica não alteram a lógica do sistema. A Figura 4.1 apresenta a interação do padrão.

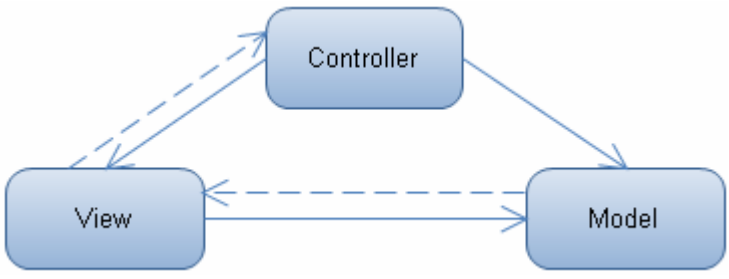


Figura 4-1 – Diagrama simples da relação entre Model, View e Controller

4.1.1 Ambiente de Programação Netbeans

A IDE NetBeans é um ambiente de desenvolvimento - uma ferramenta pra programadores, que lhes permite escrever, compilar, debugar e instalar programas. A IDE é completamente escrita em Java, mas pode suportar qualquer linguagem de programação. Existem também um grande número de módulos para estender a IDE NetBeans. A IDE NetBeans é um produto livre, sem restrições de como ela pode ser usada. O NetBeans IDE oferece aos desenvolvedores todas as ferramentas necessárias para criar aplicativos profissionais de desktop, empresariais, Web e móveis multiplataformas [38].

4.1.2 Linguagem de Programação Java

A linguagem Java [39] foi escolhida por ser uma linguagem computacional completa e adequada ao desenvolvimento de aplicações baseadas na Web, redes fechadas ou programas stand-alone. Java é poderosa em ambientes distribuídos complexos, como uma rede Internet. Porém, por sua versatilidade, oferece ao programador uma poderosa linguagem de programação de uso geral, com recursos para a construção de uma variedade de aplicativos, podendo ou não depender do uso de recursos de conectividade. Atualmente a linguagem disponibiliza ao programador alguns avanços tecnológicos, fundamentais para o desenvolvimento de poderosas ferramentas, bem como uma estreita integração com diversos gerenciadores de bancos de dados.

4.1.3 Java Server Pages

O JSP [40] é uma tecnologia para desenvolvimento de aplicações web que tem a vantagem de ser multi-plataforma, como também é de fácil codificação, facilitando assim a construção e manutenção de uma aplicação. Esta tecnologia permite separar a programação lógica (parte dinâmica) da programação visual (parte estática), possibilitando o desenvolvimento de aplicações mais robustas, em que o programador e o designer podem trabalhar no mesmo projeto, porém, de maneira independente.

JSP utiliza *tags* (comandos que são executados e identificados por algum separador) semelhantes ao XML, que encapsulam a lógica de programação e geram o

conteúdo das páginas. Além disso, a lógica de aplicação pode residir em recursos baseados no servidor, que a página acessa através das *tags*. Toda a formatação tanto HTML como XML é passada diretamente para a página de resposta.

É necessário ter um servidor de aplicação para se executar uma página JSP, que receba a solicitação do usuário, efetue o processamento e envie a resposta ao mesmo. Existem vários servidores de aplicação disponíveis na web, dentre os que têm o conceito de software livre estão o Jakarta Tomcat, JBoss etc. Nesta dissertação o Tomcat é usado como servidor de aplicações.

Na Figura 4.2 é ilustrado o funcionamento da requisição de uma página JSP.

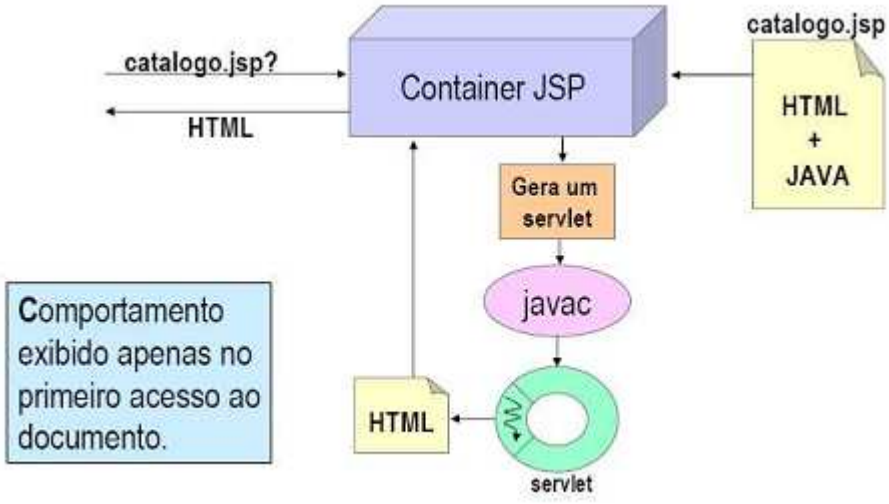


Figura 4-2 - Funcionamento de uma página JSP [45]

4.1.4 Banco de Dados MySQL

O MySQL é um sistema de gerenciamento de banco de dados (SGBD) que utiliza a linguagem SQL (Structured Query Language - Linguagem de Consulta Estruturada) como interface. É, atualmente, um dos bancos de dados mais populares, com mais de 10 milhões de instalações pelo mundo.

É reconhecido pelo seu desempenho e robustez e também por ser multi-tarefa e multi-usuário. A Wikipédia, usando o programa *MediaWiki*, utiliza o MySQL para gerenciar seu banco de dados, demostrando que é possível utilizá-lo em sistemas de produção de alta exigência e em aplicações sofisticadas [44].

4.2 Modelagem do Sistema

4.2.1 Casos de Uso

Os casos de uso (Figura 4.3) definem as funcionalidades principais disponíveis no sistema. A partir dessas, outras funcionalidades estão definidas, porém são detalhadas na seção 5.3 que descreve cada um dos módulos do sistema.

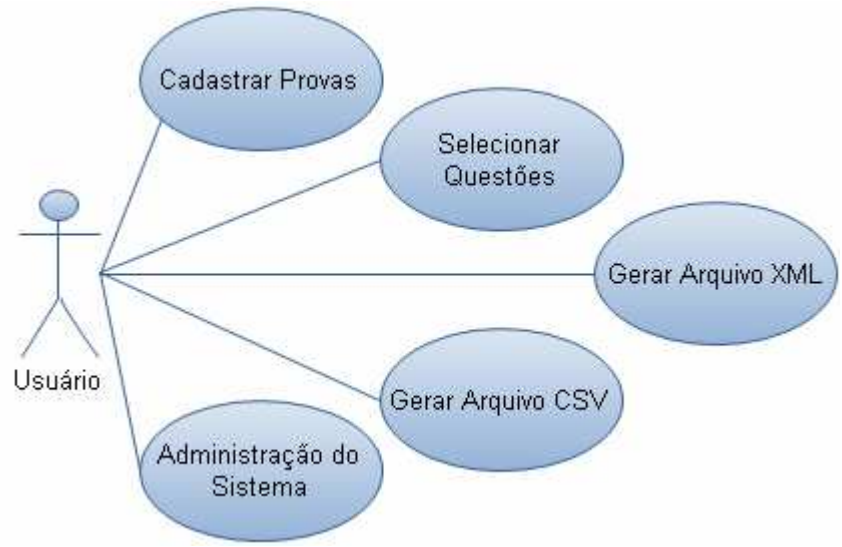


Figura 4-3 - Funcionalidades Gerais dos Casos de Uso

4.2.2 Descrição dos casos de uso

Os casos de uso são baseados nas funcionalidades disponíveis no sistema. A seguir são descritos os principais casos de uso.

a) Cadastrar Provas

Atores: Usuários cadastrados no sistema

Descrição: O usuário entra com os dados nos campos adequados no sistema através de uma prova que tenha em mãos.

b) Selecionar Questões

Atores: Usuários cadastrados no sistema

Descrição: O usuário entra em uma página que contém um filtro, onde poderá selecionar as questões por disciplina, assunto, concurso, organizador ou órgão.

c) Gerar arquivo XML

Atores: Usuários cadastrados no sistema

Descrição: O usuário após ter listado as questões através do filtro, poderá gerar o arquivo XML que será usado por programas específicos de mineração de textos. Este arquivo poderá ser gerado com as opções de parâmetros escolhidas no momento da geração (..) .

d) Gerar arquivo CSV

Atores: Usuários cadastrados no sistema

Descrição: O usuário após ter listado as questões através do filtro, poderá gerar o arquivo CSV que será usado por programas específicos de mineração de textos. No momento da geração do arquivo, serão apresentadas as opções de parâmetros que o usuário irá escolher para a mineração (..).

e) Administração do Sistema

Atores: Usuários administradores

Descrição: O usuário administrador recebe pedidos de cadastramento e envia as respostas contendo a confirmação ou não do acesso ao site da aplicação

4.3 Módulos do Sistema

4.3.1 Iniciando o sistema

O sistema de pré-processamento é iniciado através do browser disponível no sistema do usuário, através da URL: http://localhost:8084/Mestrado, que vai ativar a execução da tela mostrada na Figura 4.4. Nela, o usuário tem disponível todas as funcionalidades do sistema.

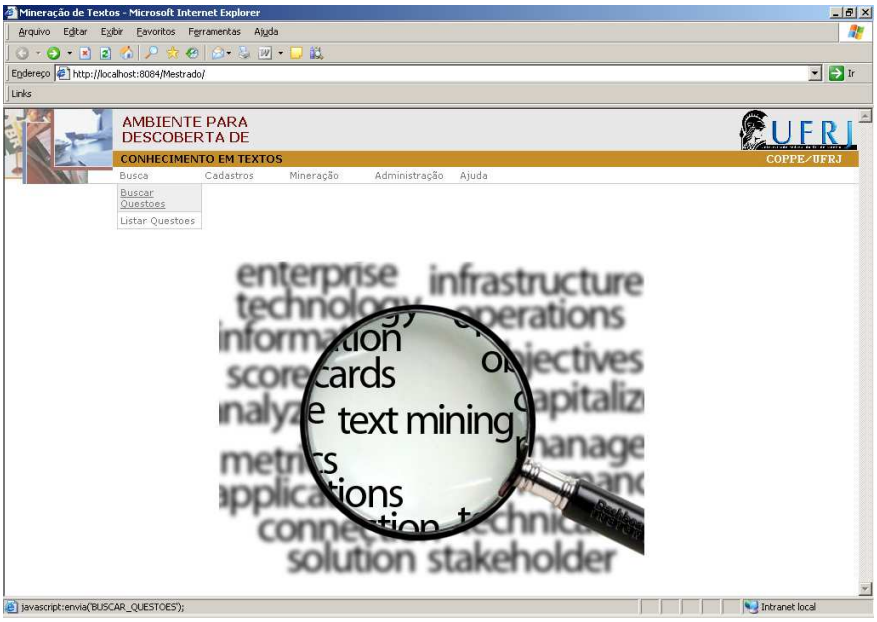


Figura 4-4 - Tela principal do sistema

4.3.2 Cadastros

No modulo de entrada de dados, o usuário poderá inserir novas questões de provas.

4.3.3 Busca

No menu “Busca” se encontram duas opções de busca. A primeira se refere ao filtro de questões que poderão ser selecionadas para se obter uma pesquisa mais restrita de questões. Na segunda opção são listadas todas as questões cadastradas no sistema.

A Figura 4.5 apresenta a tela de filtro, onde o usuário tem a opção de restringir a busca de questões por disciplina, assunto, concurso, organizador, órgão e cargo.

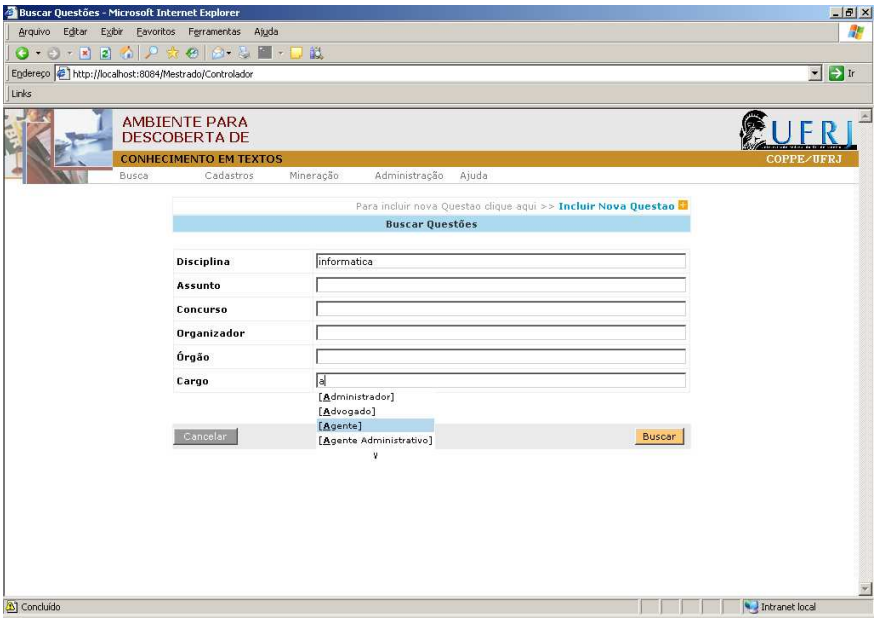


Figura 4-5 – Filtro com as questões que serão procuradas.

O sistema foi criado de tal forma que os documentos sejam pré-selecionados através de uma consulta inicial. Através do filtro principal, seleciona-se um grupo mais específico de documentos. Caso o especialista precise pesquisar em todos os documentos, esta opção também fica disponível.

Após o filtro inicial são listadas as questões de acordo com as opções do filtro. A Figura 4.6 apresenta uma lista das questões classificadas pelo assunto.

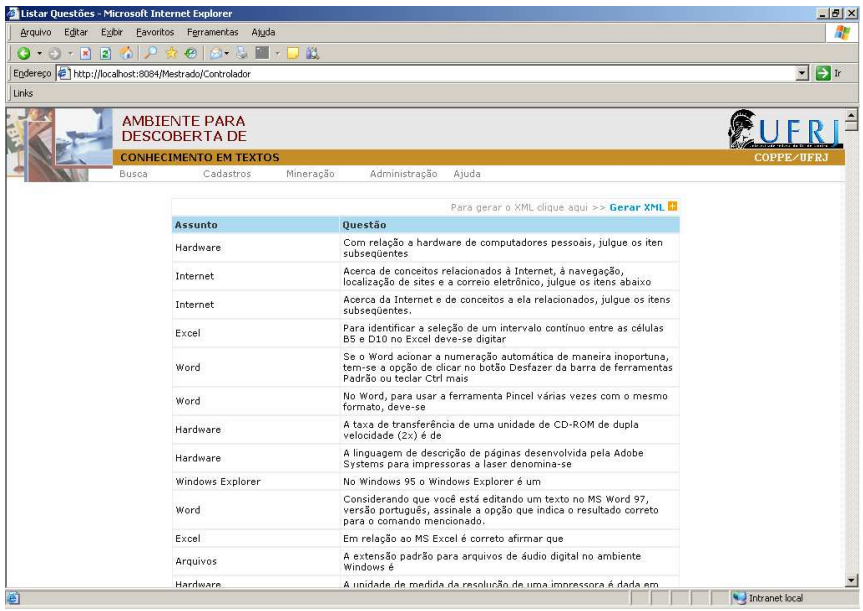


Figura 4-6 – Lista de questões selecionadas pela consulta

4.3.4 Geração do Arquivo

Com as questões listadas na tela, o usuário poderá gerar o arquivo que será usado para o processo de MT, conforme a Figura 4.7.

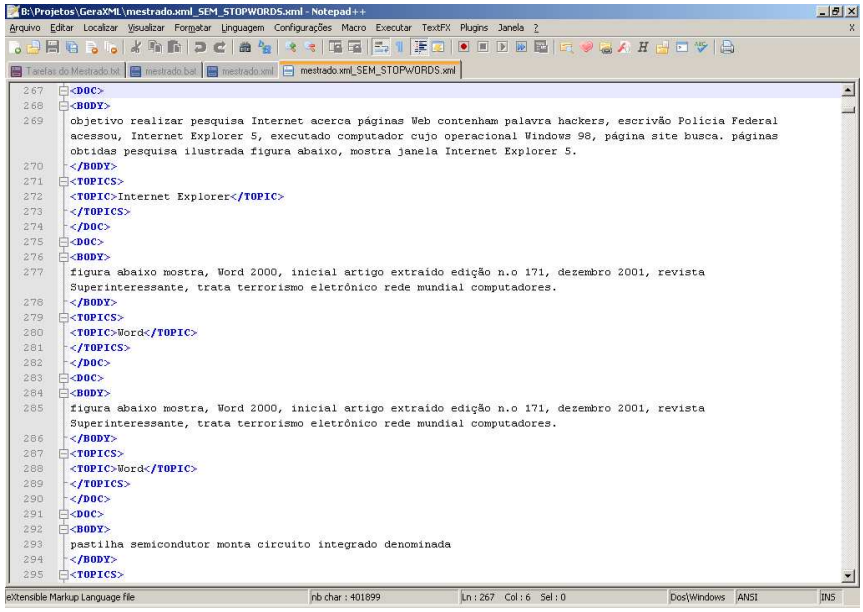


Figura 4-7 – Arquivo XML gerado pelo sistema

Na geração do arquivo, o usuário poderá marcar as opções de gerar o arquivo sem nenhum parâmetro selecionado, ou seja, as questões serão geradas no formato XML, com o mesmo conteúdo original, sem nenhuma mineração. Ou o usuário poderá selecionar os seguintes parâmetros, de acordo com o seu interesse:

- ✓ Remoção de caracteres indesejados
- ✓ Integração de caixa
- ✓ Remoção de termos estrangeiros
- ✓ Remoção de stopwords
- ✓ Stemming

4.3.5 Banco de Dados

O banco de dados construído para este trabalho possui uma tabela resultante de uma consulta onde as questões das provas estão armazenadas. Na tabela 4.1, onde se apresenta a tabela, pode-se observar que cada linha representa uma questão. Todas as questões mostradas na tabela são da disciplina de informática. Na primeira linha da

tabela de Concursos (tabela 4.2) entende-se que a questão foi retirada da prova para o cargo de Auditor Fiscal da Receita Estadual, realizada em 3 de julho de 2005 e a entidade organizadora do concurso foi a ESAF. Nas colunas “Enunciado” e “Alternativa”, estão o enunciado e as alternativas da questão e na coluna “Assunto” encontra-se o assunto da questão, representando a categoria já pré-classificada.

Tabela 4-1 - Questão de uma prova de informática

CodTe	Enunciado	Assunto	Alternativa
2	Com relação a hardware de computadores pessoais, julgue os iten subseqüentes	Hardware	a) Os disquetes de 3½" possuem mecanismo para impedir gravação e a exclusão de arquivos b) A maioria dos mouses se conecta ao computador por meio da porta paralela do computador c) Muitas impressoras do tipo jato de tinta utilizam cartucho separados: um para a cor preta e outro para as demais cores d) Os disquetes de 3½" atuais são capazes de armazenar um número superior a um milhão de bytes e) Em geral, enquanto os scanners são conectados na parte traseira da caixa da CPU, os teclados de PCs conectam-se diretamenta na parte traseira do monitor de vídeo

Tabela 4-2 - Tabela de Concursos

CodConcurso	Cargo	Orgao	EntOrganizador	TotQuest	Data
131	Auditor Fiscal da Receita Estadual	SEFAZ/MG	ESAF	5	3/7/2005
132	Analista Judiciário	TRE/MG	FCC	5	18/7/2005
133	Auditor - Direito	Estado/MT	NCE/UFRJ	8	19/12/2004
134	Técnico Judiciário	TRE/RN	FCC	5	3/7/2005
135	Analista Judiciário	TRE/RN	FCC	5	3/7/2005
136	Analista Judiciário - Administrativa	TRE/PI	FCC	5	12/5/2002
137	Técnico Judiciário - Administrativa	TRE/PI	FCC	5	12/5/2002
138	Analista Judiciário - Administrativa	TRE/ES	FESAG	6	22/5/2005
139	Analista Judiciário - Judiciária	TRE/ES	FESAG	6	22/5/2005
140	Técnico Judiciário - Administrativa	TRE/ES	FESAG	6	22/5/2005
141	Auditor Fiscal de Tributos Estaduais	SEFAZ/AM	NCE/UFRJ	6	18/9/2005
142	Assistente Administrativo da Fazenda Estadual	SEFAZ/AM	NCE/UFRJ	20	18/9/2005
143	Técnico Judiciário - Administrativa	TRE/SC	FAPEU	7	19/6/2005
144	Analista Judiciário - Contador	TRE/SC	FAPEU	3	19/6/2005

4.4 Software PolyAnalyst

O PolyAnalyst é um software desenvolvido pela empresa Megaputer Intelligence Inc e encontra-se, atualmente, em sua versão 6.0. Trata-se de um programa de mineração que incorpora as últimas tecnologias na descoberta automatizada do conhecimento e trabalha com dados estruturados e não estruturados, facilitando as pesquisas e a tomada de decisões a partir da análise dos dados. O software contém várias ferramentas para exploração de dados, que possibilitam ligá-los, alterá-los, separá-los, analisá-los e sumarizá-los. Estas ferramentas estão disponíveis para que o

usuário possa compor seus projetos de análise, de acordo com as tarefas que pretende realizar. A Figura 4.8 apresenta o ambiente de trabalho do PolyAnalyst onde se tem acesso a toda funcionalidade do sistema.

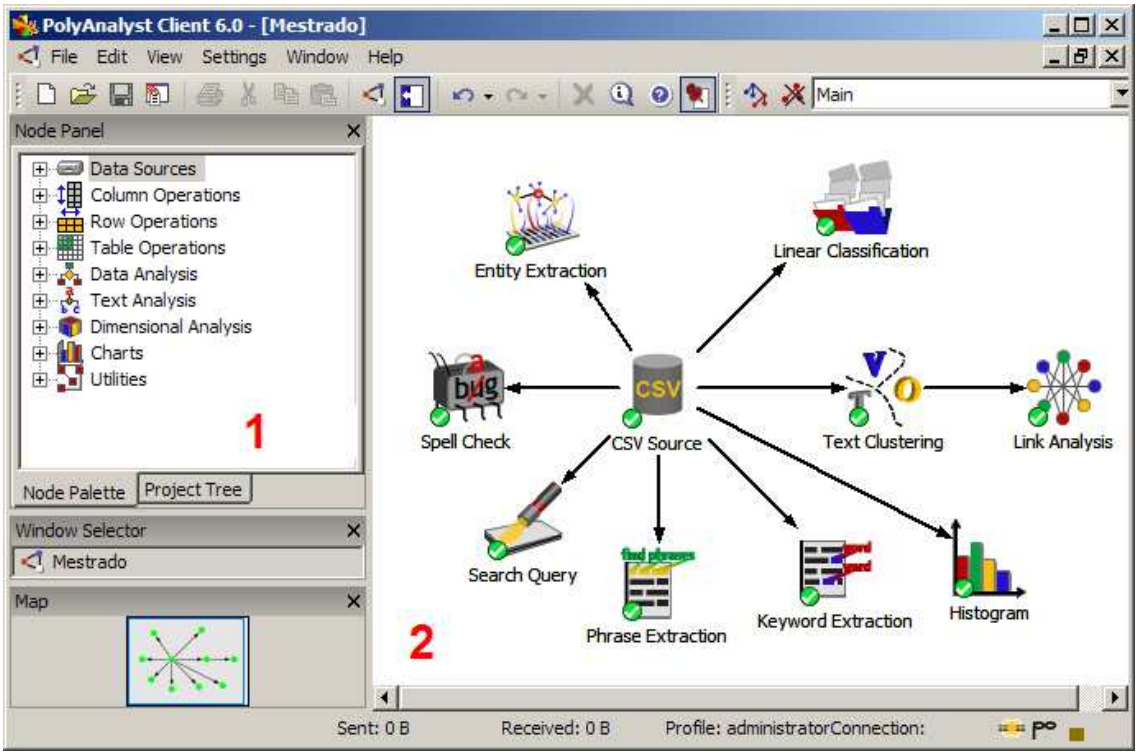


Figura 4-8 – Ambiente de trabalho do PolyAnalyst

Nesta dissertação, na qual o objetivo é minerar documentos, procurou-se explorar o ambiente operacional do software dando ênfase às tarefas inerentes a Mineração de Textos. Nas próximas seções serão apresentadas as funcionalidades do PolyAnalyst utilizadas nesta dissertação.

4.4.1 A Arquitetura do PolyAnalyst

O software PolyAnalyst utiliza a arquitetura cliente/servidor, que permite uma administração eficiente dos recursos disponíveis. Tipicamente são montadas várias estações clientes que são atendidas por um único servidor. Essas estações podem ser configuradas individualmente para dar suporte a tarefas específicas. Por exemplo, algumas podem estar configuradas para as tarefas de análises de dados propriamente ditas, enquanto outras podem estar voltadas para usuários de negócio, que, basicamente, terão interesse nos relatórios das análises realizadas.

No programa cliente - ***Analytical Client*** – o analista de dados constrói o workflow responsável por todo o processo de extração do conhecimento. Na figura 4.8 pode-se observar um exemplo da estrutura montada para esta dissertação.

O *Analytical Client* possui um painel de menu estruturado como uma árvore onde cada nó propõe uma tarefa específica. Esses nós são agrupados de acordo com as características das tarefas realizadas por eles. Um nó é uma operação que processa uma entrada e produz uma saída.

Abaixo são apresentados os grupos das tarefas presente no aplicativo:

- ***Data Sources*** – dá suporte às diferentes fontes de dados passíveis de uso pelo sistema;
- ***Column Operations*** – dá suporte às tarefas de tratamento de colunas;
- ***Row Operations*** – dá suporte as tarefas de tratamento de registros;
- ***Table Operations*** – dá suporte às tarefas que envolvem operações nas tabelas;
- ***Data Analysis*** – dá suporte às tarefas de análise de dados convencionais;
- ***Text Analysis*** – dá suporte às tarefas de análise de dados textuais;
- ***Dimensional Analysis*** – dá suporte às tarefas de segmentação de tabelas;
- ***Charts*** – dá suporte às tarefas de geração de gráficos para a visualização dos resultados;

Neste trabalho foram utilizados os grupos de tarefas *Data Sources*, *Data Analysis* e *Text Analysis* para a realização das tarefas propostas.

4.4.2 Data Source

O sistema permite que se trabalhe com arquivos de diferentes formatos para a entrada de dados, a saber:

- Arquivos texto em geral: txt, doc, rtf, html, pdf, etc.
- Formato CSV (Comma-Separated Values);
- Fonte de Dados ODBC
- Microsoft Access
- Microsoft Excel
- Formato de webpages (arquivos carregados diretamente da Internet);
- Formatos de texto diversos, para serem processados em rotinas de análises de texto;

- Formato RSS.

Os *Data Sources* tem o propósito de tratar os diferentes tipos de fonte de dados possíveis de alimentar o sistema. No escopo deste trabalho optou-se pela fonte de dados CSV, pela comodidade de se gerar este tipo de arquivo pela aplicação proposta.

4.4.3 Text Analysis

Esse grupo trata das tarefas de processamento dos dados textuais, que dão suporte à Classificação e à Clusterização de documentos, bem como à extração de frases, palavras-chave ou entidades contidas nos textos, assim como a tarefas de tratamento de erros. Os resultados gerados consistem em relatórios e tabelas que exibem o conteúdo extraído.

O grupo é composto por oito módulos, responsáveis por tarefas distintas, conforme apresentado a seguir:

- ***Linear Classification*** – utilizado para o desenvolvimento de tarefas de Classificação, através de dois algoritmos distintos: o SVM e o Bayesiano Simples;
- ***Text Clustering*** – utilizado para tarefas de Clusterização, a partir de palavras-chave contidas nos textos;
- ***Phrase Extraction*** – utilizado para extrair automaticamente a freqüência com que frases ocorrem dentro dos textos. A tabela 4.3 mostra uma tabela apresentada pelo software com as ocorrências das frases dentro dos documentos da base de questões de informática.

Tabela 4-3 - Ocorrência de frase nos documentos

Phrase	Freque...	Support
usu rio	184	135
mem ria	140	70
afirma verdadeira	111	111
op contenha afirma verdadeira	108	108
indique op contenha afirma verdadeira	106	106
microsoft word	98	90
correto afirmar	98	98
internet explorer	96	74
usu rios	82	60
seguinte afirma relativa	75	75
analise seguinte afirma relativa	74	74
analise seguinte afirma relacionada	43	43

- ***Keyword Extraction*** - utilizado para identificar as ocorrências de palavras-chave com maiores significância dentro dos textos. O resultado é apresentado de forma tabular, vide tabela 4.4.

Tabela 4-4 - Extração de palavras-chaves

Term	1.5 Significance	1 Support	1 Frequency
arquivo	81.79	385	890
computador	81.09	309	557
usuario	80.79	289	520
rede	80.62	216	479
documento	80.41	203	430
pagina	80.40	185	428
celula	80.37	123	422
texto	80.37	229	422
internet	80.06	297	680
memoria	79.91	128	310

- ***Entity Extraction*** - utilizado para extrair automaticamente entidades que ocorrem nos textos. O software trabalha com 8 entidades pré-cadastradas. Data, nome, organização, endereço, telefone, moeda, local geográfico e objeto usuário;
- ***Search Query*** – utilizado para a pesquisa de termos e frases. O PolyAnalyst trás várias funções interessantes para consultar palavras dentro de textos. Uma destas funções é a “Soudex()”, que procura palavras com fonemas parecidos.
- ***Link Terms*** – utilizado para identificar associações entre termos e frases;
- ***Spell Check*** – utilizado para identificar erros e sugerir correção na grafia dos termos;

Na próxima seção daremos ênfase ao funcionamento das principais tarefas utilizadas neste trabalho: *Linear Classification* , *Text Clustering e Link Analysis* .

4.4.3.1 Linear Classification

Este módulo desenvolve um modelo de Classificação dependente de um atributo estruturado, através da utilização de uma coluna independente com os textos. Esse modelo é baseado na freqüência e na distribuição dos termos no texto. A partir disso, o programa treina um modelo para a classificação automática de textos. O PolyAnalyst apresenta duas abordagens de classificação de textos, baseadas em algoritmos distintos, a saber:

- Baseada no algoritmo SVM – um algoritmo que requer um processamento mais intensivo em termos computacionais e que tipicamente apresenta uma maior acurácia nos resultados;
- Baseada no algoritmo Bayesiano Simples – um algoritmo de processamento computacional mais rápido, que é mais escalável, e que, em geral, obtém resultados menos precisos que o SVM.

Os detalhes de como os dois algoritmos trabalham não são apresentados na documentação disponibilizada pelo fabricante. Na configuração dos parâmetros para se executar a Classificação é necessário definir o atributo a ser utilizado como fonte dos dados, o algoritmo de processamento, o uso ou não de uma *stoplist* e a definição da mesma, o tipo de dado a ser tratado, entre outros. Durante o processamento, o programa armazena as palavras-chave de forma booleana em uma tabela, indicando se elas aparecem ou não em cada documento, e a freqüência com que isso acontece.

Após o processamento, o resultado é apresentado em duas abas: uma contendo as informações da configuração adotada, e a outra contendo as informações da classificação propriamente dita, incluindo uma matriz com as taxas de erro por classe.

A Figura 4.9 apresenta a tela de propriedades do classificador Linear, onde são configurados os principais parâmetros do classificador.

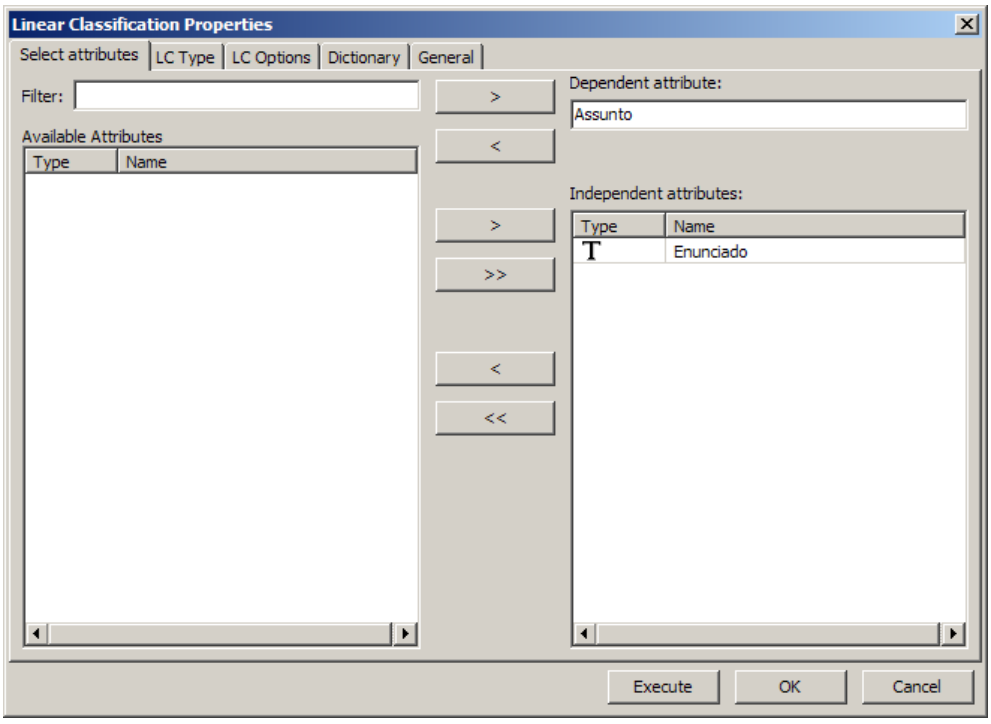


Figura 4-9 – Propriedades do Classificador linear

4.4.3.2 Text Clustering

Este módulo é utilizado para a Clusterização de documentos. Para a geração dos Clusters, ele utiliza uma variação do algoritmo *Suffix Tree Clustering*, que foi apresentado originalmente em [46]. Maiores detalhes do funcionamento deste algoritmo foi descrito no capítulo 2 deste trabalho.

Na configuração dos parâmetros para a execução da Clusterização, primeiro é preciso definir o atributo utilizado como fonte de dados, a partir da tabela de entrada especificada. Então, é necessário configurar alguns parâmetros matemáticos que manipulam o comportamento do algoritmo, como os apresentados a seguir:

- Escolher se o agrupamento será base, exclusivo ou hierárquico;
- O número máximo de agrupamentos básicos, que diz respeito ao número máximo de frases individuais e palavras que serão buscadas, no primeiro passo do algoritmo;
- Os números de percentual mínimo e máximo de registros por grupo;
- O uso ou não de um thesaurus;
- O uso ou não de um dicionário e a definição de qual.

A Figura 4.10 mostra a tela de configuração do cluster.

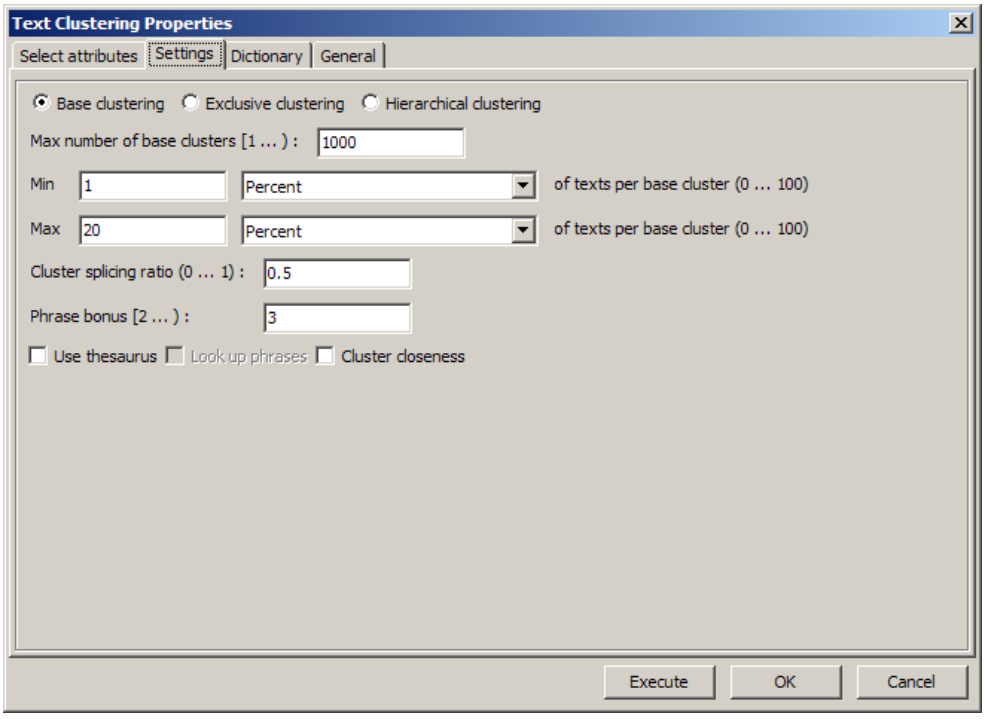


Figura 4-10 – Tela de propriedade da Clusterização de textos

O resultado do processamento é apresentado em várias abas, contendo a configuração adotada no processamento, um gráfico estatístico com a distribuição dos documentos pelos Clusters, uma tabela com os Clusters gerados, contendo as suas descrições, a quantidade e a identificação dos documentos presentes, um gráfico mostrando a proximidade dos Clusters, entre outros. Cabe ressaltar que na abordagem do algoritmo implementado para essa tarefa não é possível definir a quantidade de grupos a ser encontrada, na configuração dos parâmetros.

4.4.4 Análise de Links

O software possui um módulo para se trabalhar com Análise de Links, o qual revela padrões complexos de correlações entre valores dos atributos representando-os visualmente. Resultados da análise são apresentados como um grafo *d* e objetos conectados que suporta vários tipos de manipulação e operações *drill-down*. A saída visual da análise de links facilita em um melhor entendimento sobre estruturas escondidas dos dados investigados e ajuda rapidamente a isolar padrões de interesse para uma posterior investigação. Na Figura 4.15, apresenta-se um exemplo de uso do Link Analysis.

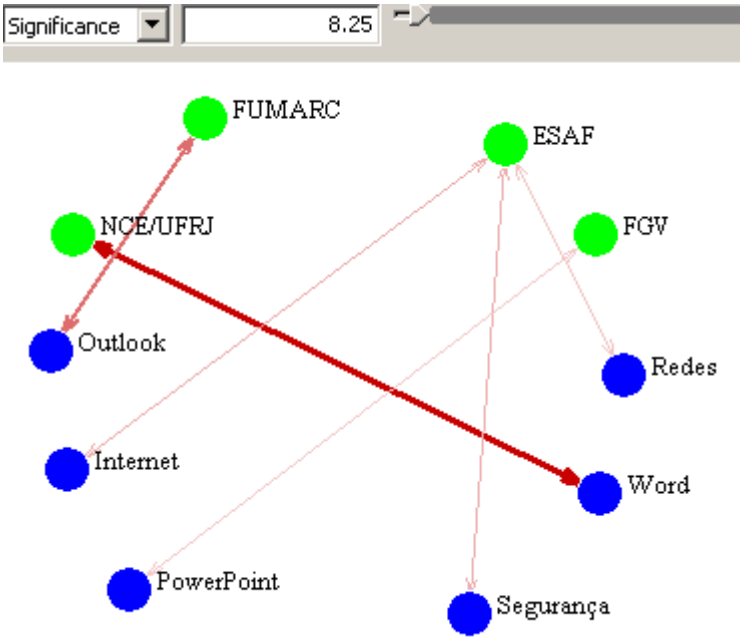


Figura 4-11 – Exemplo de visualização do Link Analysis

A habilidade de mostrar visualmente e posteriormente investigar e agrupar termos extraídos de notas textuais faz desta técnica um componente interessante na análise de dados não estruturados.

5 Estudo de Casos e Resultados

Neste capítulo são descritos o corpus, os métodos utilizados no pré-processamento dos textos, as tarefas de categorização, de agrupamento e a análise de links. São apresentados dois estudos de casos, onde foram utilizadas as funcionalidades desenvolvidas no ambiente de descoberta de conhecimento, aplicadas sobre questões de concursos. Pretende-se com isso mostrar que a ferramenta é capaz de apresentar resultados satisfatórios para consultas não convencionais. Para a realização dos experimentos foram utilizados os softwares descritos no capítulo 4.

5.1 O Corpus

Para a construção do modelo de mineração de textos é importante que se tenha um conjunto de documentos pré-classificados nas diversas categorias temáticas do domínio de interesse. Esse conjunto de documentos é denominado corpus.

O corpus utilizado para realização dos experimentos foi retirado dos *sites* de instituições organizadoras de concurso público no Brasil. As provas coletadas foram realizadas entre os anos de 2000 e 2006. Os dados iniciais estavam originalmente em arquivos do tipo PDF. As provas se referem a concursos para o provimento dos mais variados cargos da esfera pública estadual e federal.

A primeira opção para transformar os arquivos PDF para textos foi usar conversores automáticos encontrados na internet. O problema desta abordagem está na perda de formatação do arquivo original, causando um trabalho extenuante para identificação e correção das palavras “quebradas”. Então para poder trabalhar com as provas, montou-se um sistema de cadastro de questões junto a uma estrutura de banco de dados com o intuito de organizar as questões pelos assuntos correspondentes. O sistema, apresentado no capítulo 4, trabalha os documentos, prepara os dados e gera arquivos CSV e XML com as questões prontas para serem processadas. As figuras 5.1 e 5.2 apresentam a origem dos dados, de acordo como aparecem nas provas coletadas.

Da figura 5.1 retiram-se as seguintes informações:

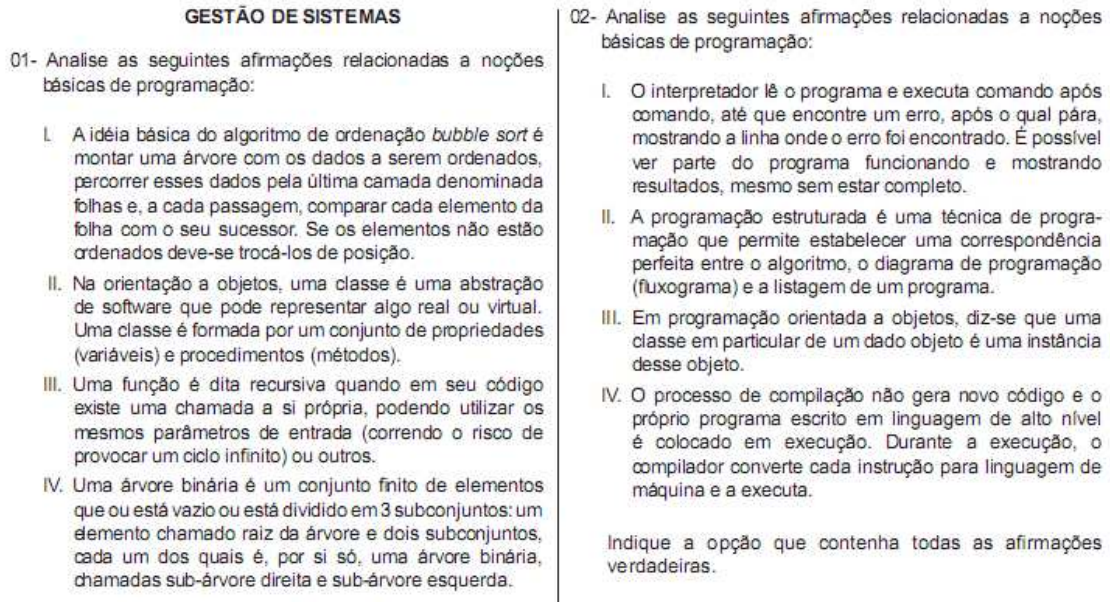
- Órgão: Receita Federal do Brasil
- Organizadora: ESAF
- Cargo: Auditor Fiscal da Receita Federal do Brasil

- Ano: 2005

E na figura 5.2 são coletados os seguintes campos:

- Assunto: Gestão de Sistemas
- Enunciado
- Alternativas

Figura 5-1 – Exemplo de prova



GESTÃO DE SISTEMAS

01- Analise as seguintes afirmações relacionadas a noções básicas de programação:

I. A idéia básica do algoritmo de ordenação *bubble sort* é montar uma árvore com os dados a serem ordenados, percorrer esses dados pela última camada denominada folhas e, a cada passagem, comparar cada elemento da folha com o seu sucessor. Se os elementos não estão ordenados deve-se trocá-los de posição.

II. Na orientação a objetos, uma classe é uma abstração de software que pode representar algo real ou virtual. Uma classe é formada por um conjunto de propriedades (variáveis) e procedimentos (métodos).

III. Uma função é dita recursiva quando em seu código existe uma chamada a si própria, podendo utilizar os mesmos parâmetros de entrada (correndo o risco de provocar um ciclo infinito) ou outros.

IV. Uma árvore binária é um conjunto finito de elementos que ou está vazio ou está dividido em 3 subconjuntos: um elemento chamado raiz da árvore e dois subconjuntos, cada um dos quais é, por si só, uma árvore binária, chamadas sub-árvore direita e sub-árvore esquerda.

02- Analise as seguintes afirmações relacionadas a noções básicas de programação:

I. O interpretador lê o programa e executa comando após comando, até que encontre um erro, após o qual pára, mostrando a linha onde o erro foi encontrado. É possível ver parte do programa funcionando e mostrando resultados, mesmo sem estar completo.

II. A programação estruturada é uma técnica de programação que permite estabelecer uma correspondência perfeita entre o algoritmo, o diagrama de programação (fluxograma) e a listagem de um programa.

III. Em programação orientada a objetos, diz-se que uma classe em particular de um dado objeto é uma instância desse objeto.

IV. O processo de compilação não gera novo código e o próprio programa escrito em linguagem de alto nível é colocado em execução. Durante a execução, o compilador converte cada instrução para linguagem de máquina e a executa.

Indique a opção que contenha todas as afirmações verdadeiras.

Figura 5-2 Exemplo de questão de prova

O arquivo CSV gerado a partir do sistema possui os seguintes campos: Questão, Assunto, Cargo, Órgão, Organizador e Data. O campo “Questão” é composto tanto pelo enunciado quanto pelas alternativas da questão. Este campo sofre um tratamento na etapa de preparação de dados para retirada de *stopwords*, retirada de acentos, integração

de caixa e remoção de caracteres indesejados. Foram feitos testes com o texto *stemmizado*, mas o resultado não foi satisfatório.

5.2 Questões de Informática

Optou-se por selecionar as questões relacionadas à disciplina de Informática, pois vem sendo cobrada na maioria dos concursos e também porque é importante que um especialista no assunto esteja presente no momento de preparação dos dados.

São 6 os atributos (colunas) e 1463 os registros (linhas). Na tabela 5.1 pode-se ver a estrutura dos dados do arquivo CSV gerado no pré-processamento. A primeira coluna apresenta o cabeçalho que identifica os atributos.

Tabela 5-1 - Linha do arquivo CSV com uma questão de informática.

Questão	relativos hardware software computadores tipo configuracao parametros bios computador feita programa setup barramento agp cuja funcao permitir conexao computador modem instalado mesmo possui desempenho inferior barramento isa zip drive unidade disco utiliza midia capacidade memoria superior mbytes ventilador cooler instalado processador deixa funcionar importante substitui-lo evitar dano processador placas fax modem configuradas porta
Assunto	Hardware
Cargo	Técnico de Informática
Órgão	TJ/AC
Organizador	CESPE
Data	11/9/2002

5.2.1 Exploração dos Dados

As classes principais onde os arquivos foram alocados foram escolhidas pelo especialista. Na tabela 5.2 estão descritas as classes e a quantidade de documentos por cada classe.

Tabela 5-2 – Categorias

Assunto	Count	Perce...
Hardware	207	14.15
Internet	142	9.71
Excel	170	11.62
Word	296	20.23
Windows	153	10.46
Arquivos	34	2.32
Softwares	102	6.97
Email	30	2.05
Painel de Controle	28	1.91
PowerPoint	17	1.16
Outlook	47	3.21
Internet Explorer	72	4.92
Conceitos Básicos	12	0.82
Redes	37	2.53
Segurança	77	5.26
Teclas de Atalho	26	1.78
Linux	13	0.89

As palavras encontradas com maior freqüência nas questões estão listadas na tabela 5.3. Observa-se que a palavra “arquivo” foi a que teve maior freqüência entre as palavras. Apesar de o assunto “arquivos” só possuir 34 questões dentro do universo de 1463 questões, a palavra “arquivo” tem uma importância grande dentro das demais questões, aparecendo com freqüência nos demais assuntos.

Tabela 5-3 – Freqüência dos termos mais significantes nas questões de informática

Term	Significance	Support	Frequency
arquivo	81.79	385	890
computador	81.09	309	557
usuario	80.79	289	520
rede	80.62	216	479
documento	80.41	203	430
pagina	80.40	185	428
celula	80.37	123	422
texto	80.37	229	422
internet	80.06	297	680
memoria	79.91	128	310
programa	79.67	191	304
endereco	79.35	126	264
tipo	79.35	192	264
ferramenta	79.31	173	259

Buscou-se também evidenciar as frases ou conjunto de palavras que aparecem juntas, com freqüência, em uma mesma questão. Dentre essas frases que se destacam se

encontram “internet explorer” e “correio eletrônico”, destacados na Tabela 5.4. Nestas frases observa-se o interesse em se cobrar questões relativas ao uso de ferramentas de internet, e o conhecimento que o candidato tem sobre estas ferramentas.

Tabela 5-4 – Freqüência das frases mais significantes nas questões de informática

Phrase	Frequency	Support
internet explorer	134	92
correio eletronico	125	74
disco rigido	114	96
microsoft word	105	93
banco dado	83	46
pode se	80	74
barra ferramenta	76	50
window explorer	72	63
memoria ram	72	56
deve se	68	62
arquivo pasta	67	41
pagina web	62	46
operacional window	55	49
painel controle	54	40

Com esta mesma técnica é possível encontrar questões que se repetem em concursos distintos, comprovando que se estudar resolvendo questões de provas anteriores pode ser um dos recursos utilizados pelos candidatos ao se preparar para um concurso. No conjunto de questões de informática apareceram, com freqüência, questões semelhantes em provas distintas. A maioria delas se repete pelo menos uma vez. Nas tabelas 5.5 e 5.6, são apresentados dois casos em que as questões se repetem. O texto em azul reflete o trecho em que as frases estão idênticas.

Tabela 5-5 – Questão repetida em duas provas diferentes

Prova A	Prova B
produzir documento word cabecalho **sequencia adequada inserir cabecalho rodape exibir cabecalho rodape formatar cabecalho rodape inserir comentario cabecalho rodape formatar estilo cabecalho rodape**	jose deseja inserir cabecalho documento **sequencia adequada inserir cabecalho rodape exibir cabecalho rodape formatar cabecalho rodape inserir comentario cabecalho rodape formatar estilo cabecalho rodape**

Tabela 5-6 – Questão repetida em duas provas diferentes

Prova X	Prova Y
windows disponibiliza proprias ferramentas utilizadas regularmente manter disco rigido boas condicoes operacionais dentre destacamos scandisk cuja funcao disco verificar existencia virus **elimina-lo verificar erros estado superficie fisica desfragmentar arquivos acelerando desempenho aumentar espaco disponivel agrupar arquivos pesquisar clusters organizar fat**	windows disponibiliza proprias ferramentas utilizadas regularmente manter disco rigido boas condicoes operacionais dentre destaca scandisk cuja funcao disco verificar existencia virus extensao disco **elimina-lo verificar erros estado superficie fisica desfragmentar arquivos acelerando desempenho aumentar espaco disponivel agrupar arquivos pesquisar clusters organizar fat**

Outro ponto observado no conjunto de questões foi a distribuição por organizadores de concursos. A tabela 5.7 apresenta a freqüência de questões por entidades organizadoras. As entidades organizadoras FCC, ESAF e NCE/UFRJ aparecem com maior freqüência no conjunto de questões. Do total de questões, 25% abrangem questões aplicadas em concursos promovidos pela entidade organizadora FCC. Junto com a ESAF, esse percentual fica próximo de 49%, ou seja, quase metade das questões de informática foram extraídas de concursos feitos por estas entidades.

Tabela 5-7 - Estatística de distribuição das questões por Organizador.

Organizador	1 Count	1.5 Percent
FCC	375	25.63
ESAF	337	23.03
NCE/UFRJ	207	14.15
VUNESP	97	6.63
VÁRIOS	75	5.13
CESPE	73	4.99
FUMARC	72	4.92
FGV	53	3.62
CESGRANRIO	23	1.57
ACP/SP	19	1.30
FDRH	19	1.30
FESAG	18	1.23
FGSVP	15	1.03
FAPEU	10	0.68
UECE	10	0.68
COMVEST/UEPB	10	0.68
FEC	10	0.68
TRE/PB	10	0.68
UEG	7	0.48
ACP/SC	6	0.41
CONSULT	5	0.34
JPF	5	0.34
COVEST	4	0.27
ACAFE	3	0.21

5.2.2 Classificação

A presente etapa descreve os resultados obtidos com a tarefa de Classificação. O objetivo maior desta etapa é encontrar informações úteis sobre as classes dos dados que possam estar escondidas dentro da base de dados. Além disso, busca-se avaliar o melhor classificador para este tipo de base de dados. Outro ponto a se destacar é que a classificação auxiliará o especialista do domínio a categorizar automaticamente os assuntos das questões, pois muitas questões não aparecem pré-classificadas, o que torna o trabalho do especialista mais demorado.

Avaliou-se o desempenho de 2 tipos de classificadores sobre uma base com 1463 registros, 6 atributos e 17 classes. Os classificadores utilizados são implementações dos métodos Naive Bayes e SVM, contidos no software PolyAnalyst.

Para avaliação dos resultados, optou-se pela visualização através de uma matriz de confusão. Esta matriz mostra como os erros de classificação foram distribuídos. O modelo é considerado bom quando os valores da diagonal principal da matriz são altos, enquanto os outros são próximos ou, iguais a zero. A figura 5.11 exemplifica uma matriz de confusão. A classe representada na figura 5.11 como classe A é a classe que está sendo analisada e a classe B é o somatório das outras classes.

		Classe prevista	
		A	**B**
Classe real	**A**	Quantidade de registros da classe A classificados como A	Quantidade de registros da classe A classificados como B
	B	Quantidade de registros da classe B classificados como A	Quantidade de registros da classe B classificados como B

Figura 5-3 - Exemplo de uma matriz de confusão

5.2.2.1 Bayesiano Simples

real/pred	Redes	Conceitos Básicos	Painel de Controle	Word	Segurança	Hardware	Internet Explorer	Windows	Linux	PowerPoint	Excel	Internet	Arquivos	Teclas de Atalho	Outlook	Email	Softwares
Redes	34	0	0	0	0	2	0	0	0	0	0	1	0	0	0	0	0
Conceitos Básicos	0	9	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1
Painel de Controle	0	0	25	0	0	0	0	3	0	0	0	0	0	0	0	0	0
Word	2	0	0	271	2	4	1	4	0	0	6	2	0	4	2	1	2
Segurança	0	0	0	0	74	0	0	1	0	0	0	2	0	0	0	0	0
Hardware	3	0	0	0	0	198	0	1	0	1	1	0	2	0	0	0	1
Internet Explorer	0	0	0	0	0	0	69	0	0	0	1	2	0	0	0	0	0
Windows	2	0	2	4	0	4	0	139	0	0	0	3	0	2	0	0	2
Linux	1	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0
PowerPoint	0	0	0	2	0	0	0	0	0	15	0	0	0	0	0	0	0
Excel	0	0	0	6	0	0	1	0	0	0	162	0	0	0	1	0	2
Internet	0	0	0	0	0	1	1	0	0	0	0	138	0	0	0	1	1
Arquivos	0	0	0	0	0	0	0	6	0	0	0	0	26	1	0	0	1
Teclas de Atalho	0	0	0	1	0	0	0	3	0	0	0	0	0	22	0	0	0
Outlook	0	0	0	1	0	0	0	0	0	0	0	0	0	0	45	1	0
Email	0	0	0	0	0	0	0	0	0	0	0	4	0	0	1	25	0
Softwares	1	0	1	3	0	6	0	2	0	1	4	3	2	1	0	0	78

Figura 5-4 – Matriz de Confusão do Classificador Bayesiano Simples

Tabela 5-8 - Eficiência do classificador

Total de Erro de Classificação:	9,02%
Casos de indefinição na predição:	0,00%
Probabilidade de classificação:	90,98%
Eficiência:	88,67%

Tabela 5-9 - Erros de classificação por categoria

Redes :	8,11%	PowerPoint :	11,76%
Conceitos Básicos :	25,00%	Excel :	5,81%
Painel de Controle :	10,71%	Internet :	2,82%
Word :	9,97%	Arquivos :	23,53%
Segurança :	3,90%	Teclas de Atalho :	15,38%
Hardware :	4,35%	Outlook :	4,26%
Internet Explorer :	4,17%	Email :	16,67%
Windows :	12,03%	Softwares :	23,53%
Linux :	7,69%		

O classificador Bayesiano obteve excelente desempenho para as categorias Excel, Internet, Outlook, Segurança, Hardware e Internet Explorer. A categoria que obteve o pior desempenho no percentual de acertos foi a de Conceitos Básicos, com uma taxa de 25% de erro na classificação. Isso pode ser explicado pelo fato de que as questões listadas nesta categoria vez ou outra tratam de assuntos básicos referidos nas demais categorias. Então o desempenho pouco satisfatório para esta categoria já era esperado. Uma categoria que trouxe surpresa no baixo índice de acertos foi a categoria "Softwares". Depois de analisar o resultado do algoritmo Bayesiano, o especialista reconheceu que esta categoria pode estar mal empregada, pois outras categorias tratam de softwares, o que pode induzir esta categoria ao erro. A eficiência do classificador ficou em 88,67%.

5.2.2.2 Classificação Linear – Algoritmo SVM

real/pred	Redes	Conceitos Básicos	Painel de Controle	Word	Segurança	Hardware	Internet Explorer	Windows	Linux	PowerPoint	Excel	Internet	Arquivos	Teclas de Atalho	Outlook	Email	Softwares
Redes	36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Conceitos Básicos	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Painel de Controle	0	0	28	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Word	2	0	0	298	0	1	0	0	0	0	0	0	0	0	0	0	0
Segurança	0	0	0	0	77	0	0	0	0	0	0	0	0	0	0	0	0
Hardware	0	0	0	0	0	207	0	0	0	0	0	0	0	0	0	0	0
Internet Explorer	0	0	0	0	0	0	72	0	0	0	0	0	0	0	0	0	0
Windows	0	0	1	2	0	2	0	152	0	0	0	0	0	0	0	0	1
Linux	0	0	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0
PowerPoint	0	0	0	0	0	0	0	0	0	17	0	0	0	0	0	0	0
Excel	0	0	0	0	0	0	0	0	0	0	172	0	0	0	0	0	0
Internet	0	0	0	0	0	0	0	0	0	0	0	142	0	0	0	0	0
Arquivos	0	0	0	0	0	0	0	0	0	0	0	0	34	0	0	0	0
Teclas de Atalho	0	0	0	0	0	0	0	0	0	0	0	0	0	26	0	0	0
Outlook	0	0	0	0	0	0	0	0	0	0	0	0	0	0	47	0	0
Email	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	30	0
Softwares	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	102

Figura 5-5 – Matriz de Confusão na Classificação Linear – SVM

Tabela 5-10 - Eficiência do classificador

Total de Erro de Classificação:	0,68%
Probabilidade de Classificação:	99,32%
Eficiência:	99,15%

Tabela 5-11 – Erros de classificação por categoria

Redes :	2,70%	PowerPoint :	0,00%
Conceitos Básicos :	0,00%	Excel :	0,00%
Painel de Controle :	0,00%	Internet :	0,00%
Word :	1,00%	Arquivos :	0,00%
Segurança :	0,00%	Teclas de Atalho :	0,00%
Hardware :	0,00%	Outlook :	0,00%
Internet Explorer :	0,00%	Email :	0,00%
Windows :	3,80%	Softwares :	0,00%
Linux :	0,00%		

O classificador SVM obteve excelente resultado para esta base de questões de informática. Das 17 categorias existentes, somente “Redes”, “Word” e “Windows” tiveram algum erro na classificação. Além disso, o percentual de erro nestas categorias está em um nível bem satisfatório. A eficiência do classificador é de 99,15%.

5.2.2.3 Análise dos Resultados da Classificação

Os dois algoritmos foram satisfatórios nos resultados obtidos para a classificação da base de questões de informática. O SVM teve uma leve vantagem sobre o Bayesiano, pois sua eficiência na classificação foi superior em aproximadamente 10%.

5.2.3 Agrupamento

A tarefa de agrupamento de textos no PolyAnalyst utiliza uma variação do algoritmo *Suffix Tree Clustering.* O processo de descoberta dos grupos envolve dois passos principais: no primeiro, o algoritmo faz uma busca por registros que compartilham frases; no segundo, ele agrupa os documentos a partir da freqüência da ocorrência dessas frases.

Após alguns experimentos iniciais, foi adotada uma configuração de parâmetros que resultou na descoberta de 27 grupos. A figura 5-6 apresenta o gráfico de distribuição para a tarefa de agrupamento

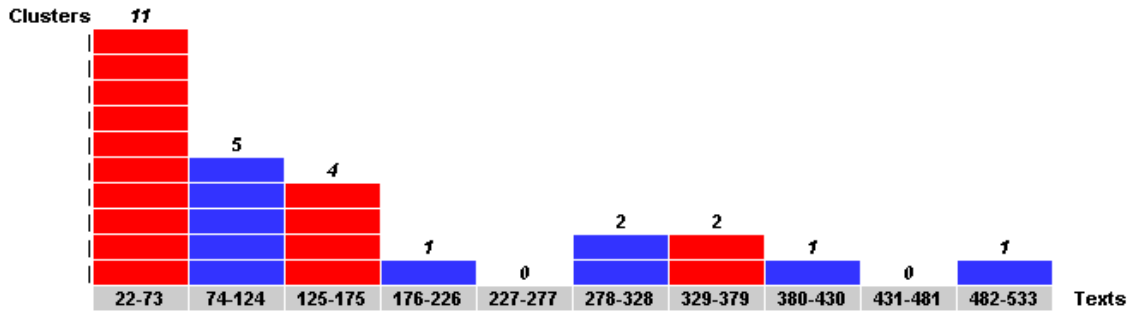


Figura 5-6 – Distribuição de textos nos Cluster

Tabela 5-12 - Clusters gerados para a base de informática

	Description	Text count
1	ctrl alt; alt shift; ctrl ctrl; ctrl shift;	40
2	texto; documento; microsoft word;	533
3	mail; tcp ip; express; outlook; mensagem; servidor; mensagens; protocolo; correio eletronico;	320
4	rede dial; rede mundial;	25
5	celula serum; celula celula;	26
6	ftp; web; www; http; site; pagina; internet explorer;	380
7	botao direito; direito mouse; botao esquerdo; esquerdo mouse;	34
8	excel; linha; celula; coluna; tabela; formula; planilha;	355
9	hardware; software;	128
10	menu; barra; ferramenta;	323
11	botao; mouse; clicar;	184
12	play; plug;	22
13	ram; rom; memoria;	154
14	ctrl; shift; tecla;	101
15	video; teclado;	87
16	rede; acesso; computadore;	339
17	painel; controle;	93
18	empresa; intranet; tecnologia;	119
19	seguranca; informacao;	126
20	saida; entrada;	62
21	disco; rigido;	163
22	enviar; destinatario;	65
23	imprimir; impressao;	61
24	editar; exibir;	109
25	colar; copiar;	62
26	rodape; cabecalho;	35
27	dial; download;	40

5.2.3.1 Análise dos Resultados do Agrupamento

Na configuração adotada para a geração dos grupos, o parâmetro de distância entre os grupos ficou com o valor de 0,3, onde o máximo é 1. Com essa configuração encontrou-se 27 grupos. Na distribuição dos grupos, verifica-se que 11 grupos são constituídos com uma média entre 22 e 73 documentos, enquanto que 1 grupo contém entre 482 e 533 documentos. Dos 1475 documentos iniciais, 6% (ou 88 documentos) não foram alocados a nenhum grupo. Alguns grupos refletem as categorias pré-estabelecidas, outros trouxeram novas informações sobre a base, que não foram encontradas através de consultas simples. Por exemplo, no grupo 10, formado pelas palavras “menu”, “barra” e “ferramenta” (tabela 5.12), há um número considerável de questões que tratam dessas palavras. Esse grupo está correlacionado com as assunto Word, como pode ser observado na figura 5.7. Com isso, conclui-se que se trata de um assunto importante nas provas e que provavelmente vem associado a estes assuntos.

Na figura 5.7 abaixo, as bolas vermelhas representam os Assuntos e as bolas coloridas representam os grupos formados.

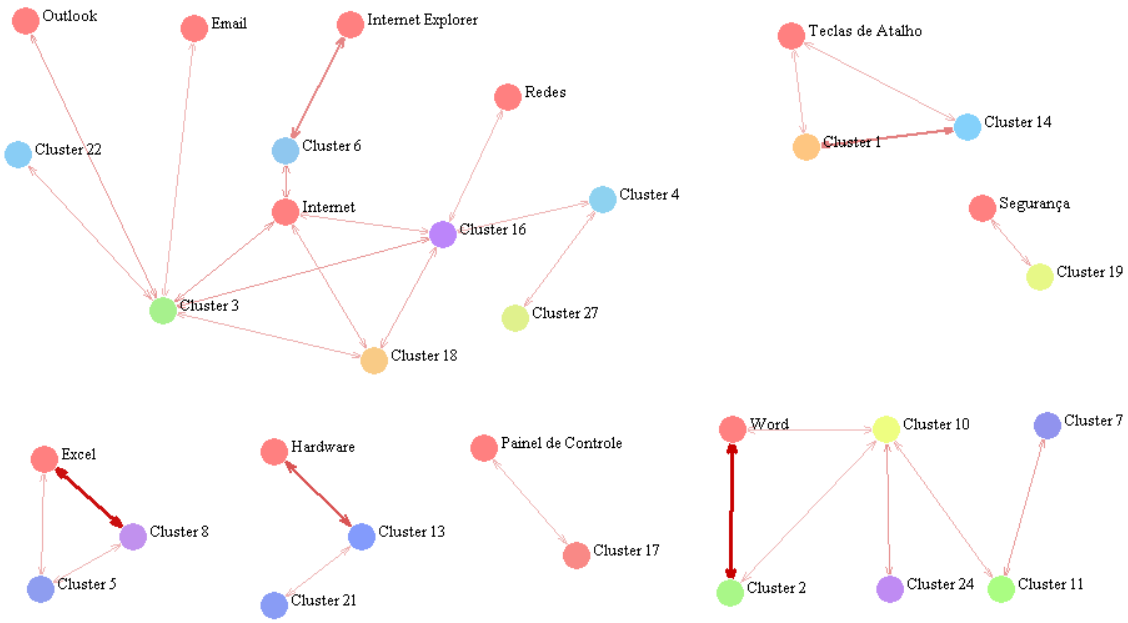


Figura 5-7 - Correlação entre as categorias(Assuntos) e os grupos formados.

O resultado da execução do algoritmo foi satisfatório, pois evidenciou a correlação dos grupos formados com as categorias originais, como visto na figura 5.7. O

algoritmo apresentou bom desempenho em sua execução e, em todos os testes realizados, os grupos foram formados em menos de 1 minuto.

5.2.4 Análise de Links

Os parâmetros configurados para a execução da tarefa de análise de links foram os seguintes:

- Mostrar links: positivos
- Significância min: 6,0
- Links mais significantes: 3000
- Suporte min: 0,0

A opção “Mostrar links” determina qual a linha de correlação que aparecerá nos links, correlação positiva, ou correlação negativa. Nessa execução serão apresentadas somente as correlações positivas. Cada associação tem uma medida de significância. Pode-se configurar um limite mínimo e máximo para esta associação e na execução desta tarefa foi configurado o limite mínimo de 6,0. Há também a possibilidade de se restringir o numero de *links*. Configura-se esta opção para definir um limite superior sobre o número de *links* identificados pelo algoritmo. Provavelmente, esse elevado número nunca será encontrado, mas este é um limiar útil quando se lida com grandes quantidades de dados.

Cada correlação é parcialmente baseada na medida de suporte, que é o número de registros envolvidos onde tanto os valores do antecedente quanto do conseqüente da correlação estão presentes. Opcionalmente especifica-se um valor mínimo e um máximo para filtrar as correlações que podem ser encontradas pelo algoritmo.

As linhas vermelhas indicam correlação positiva entre os valores dos atributos, enquanto as linhas azuis indicam correlações negativas. A intensidade da cor e do peso de cada linha representa visualmente a força da associação, onde as linhas espessas e mais escuras têm maiores correlações.

Na figura 5.8 as bolas verdes representam a entidade organizadora e as bolas azuis o assunto cobrado na questão. Observa-se forte ligação entre o NCE/UFRJ com o assunto Word. Outra ligação forte se observa entre a organizadora FUMARC e o assunto Outlook. A ESAF mantém ligações com os assuntos Internet, Segurança, Redes e Softwares.

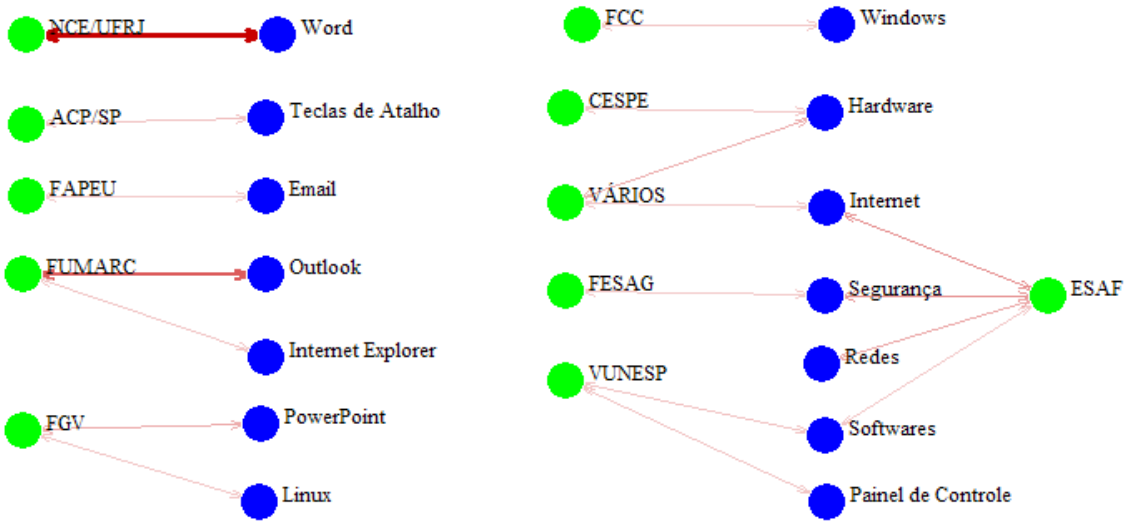


Figura 5-8 - Link entre Assunto e Organizadores de Concurso

Outro estudo foi feito entre Assuntos e Cargos no serviço público – figura 5.9. Os cargos são representados pelas bolas verdes enquanto que os assuntos são representados pelas bolas azuis. Para o Cargo de "Técnico da Receita Federal" o assunto "Redes" é muito importante, pois em várias questões existe esta ligação entre estes dois campos. Outra ligação forte foi observada nos cargos de "Auditor do Tesouro Municipal" e "Auxiliar de Marcas". Ambos apresentam forte ligação com o assunto "Word".

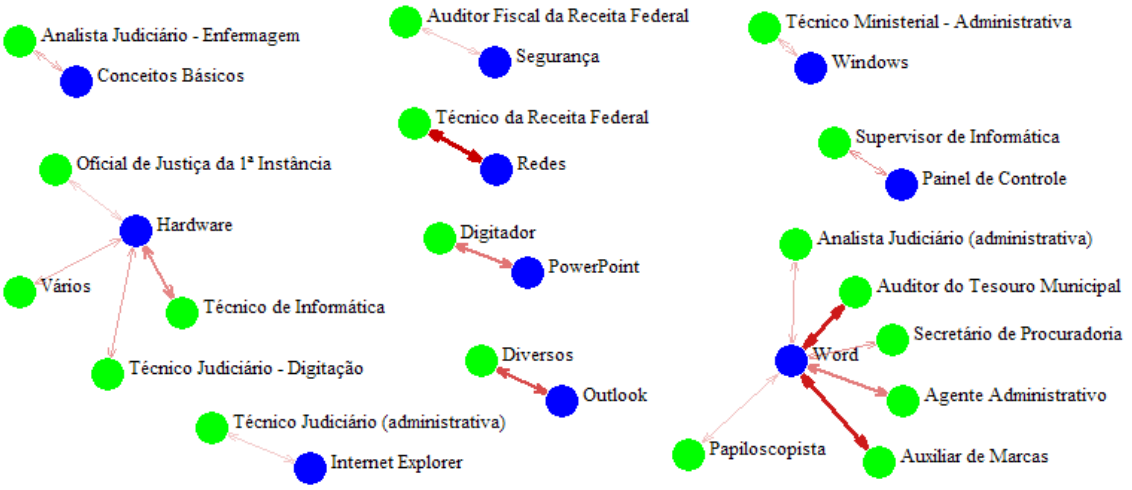


Figura 5-9 - Link entre Assunto e Cargo

Em relação aos *links* entre os grupos formados e as entidades organizadoras de concursos públicos, da figura 5.10 podem-se extrair informações das ligações existentes entre os grupos, entre as organizadoras, e também entre ambos. Observa-se que os grupos 2 e 16 são de grande importância tanto para as entidades organizadoras quanto para os demais grupos. Fazendo a análise do conteúdo destes grupos, tem-se que o grupo 2 se refere os termos “texto”, “documento” e “word”, enquanto que o grupo 16 se refere aos termos “rede”, “acesso” e “computadores”. Na associação com os assuntos existentes, a maioria dessas questões trata dos assuntos “Word” e “Redes” respectivamente.

Dentre as entidades organizadoras, aparecem em associação aos grupos o CESPE, a ESAF, a FUMARC e o NCE/UFRJ. O grupo 16 é importante tanto para a ESAF quanto para o CESPE, pois mantém uma relação de significância com essas entidades. Existe ligação entre as entidades organizadoras NCE/UFRJ e FUMARC e o grupo 2, que tem em seu conteúdo termos que se ligam ao assunto Word.

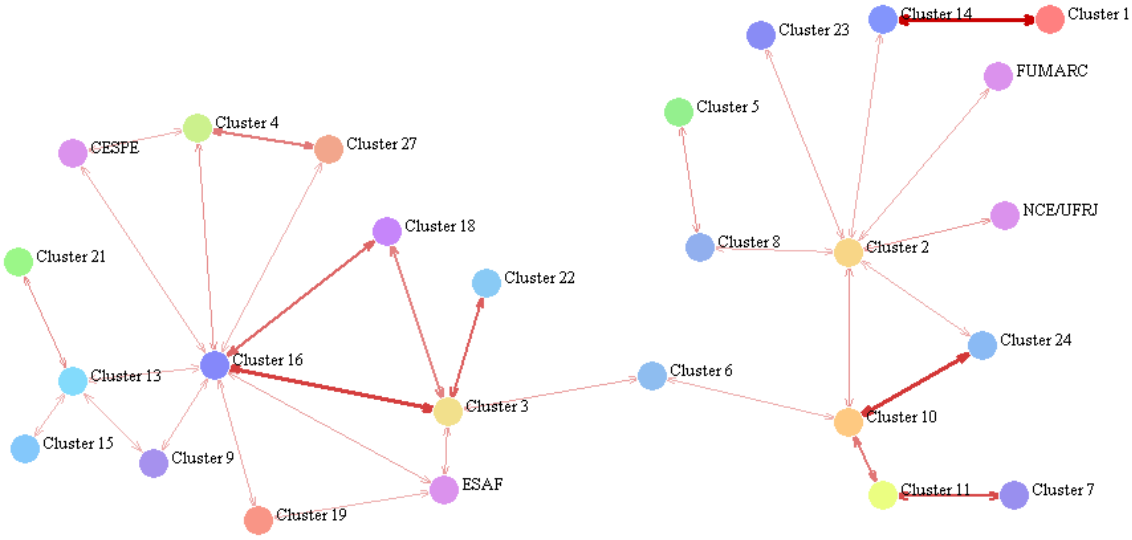


Figura 5-10 - Link entre Organizador e Grupos

Conclui-se nesta visualização que dois grupos mantêm fortes ligações com os demais e também com algumas entidades organizadoras, evidenciando a importância das questões que estão em seu conteúdo. Como estes grupos tratam de questões sobre “Redes” e de questões do assunto “Word”, pode-se aferir que o NCE/UFRJ cobra em seus concursos com freqüência o assunto “Word” e também a ESAF costuma cobrar o assunto “Redes” em suas provas.

5.3 Questões de Direito Administrativo

Optou-se por selecionar as questões relacionadas à disciplina de Direito Administrativo, pois vem sendo cobrada na maioria dos concursos e também porque se trata de uma disciplina da área jurídica, onde as provas são basicamente compostas por textos, com poucas imagens, o que torna o trabalho de mineração de textos mais relevante.

São 6 os atributos (colunas) e 3285 os registros (linhas). Na tabela 5.13 pode-se ver a estrutura dos dados do arquivo CSV. A primeira coluna apresenta o cabeçalho que identifica os atributos.

Tabela 5-13 – Linha do arquivo CSV com uma questão de direito administrativo.

Questão	limitacao exercicio direitos individuais caracteriza policia administrativa discricionario hierarquico regulamentar disciplinar
Assunto	Poderes e Deveres
Cargo	Agente Tributário Estadual
Órgão	SEFAZ/PI
Organizador	ESAF
Data	23/12/2001

5.3.1 Exploração dos dados

As classes principais, onde os arquivos foram alocados, foram escolhidas pelo especialista. Na tabela 5.14 estão descritas as classes e a quantidade de documentos por cada classe.

Tabela 5-14 – Categorias de Direito Administrativo

Assunto	1 Count	1.5 Pe...
Servidores Públicos	590	18.08
Atos Administrativos	503	15.42
Licitação Pública	375	11.49
Contratos Administrativos	251	7.69
Orgãos e Agentes Públicos	235	7.20
Ética na administração	228	6.99
Bens Públicos	191	5.85
Controle da Administração	168	5.15
Poderes e Deveres	162	4.96
Princípios Fundamentais	150	4.60
Responsabilidade Civil do Estado	142	4.35
Processo Administrativo	135	4.14
Serviços Públicos	133	4.08

Os termos encontrados com maior freqüência nas questões estão listados na tabela 5.15. Observa-se que o termo “publico” foi o que teve maior freqüência entre as termos. Observou-se que alguns assuntos têm em seu título este termo, como “Servidores públicos”, “Licitação pública”, “Bens públicos” e “Serviços públicos”. Nota-se que realmente é um termo muito comum até mesmo na categoria em que ele está inserido.

Tabela 5-15 - Termos com maior freqüência nas questões de Direito Administrativo

Term	Significance	Support	Frequency
publico	83.62	1,601	2,756
administracao	83.39	1,381	2,350
administrativo	83.19	1,310	2,238
ato	83.04	909	1,858
publica	82.92	1,160	1,884
servico	82.59	880	1,527
servidor	82.40	668	1,244
direito	82.21	695	1,111
contrato	82.13	536	1,157
licitacao	82.01	474	990
atos	81.96	565	960

Buscou-se também evidenciar as frases ou conjunto de palavras que aparecem com freqüência em uma mesma questão. Dentre as frases que se destacam se encontram “administração pública” e “ato administrativo”, destacados na Tabela 5.16.

Tabela 5-16 – Freqüência das frases mais significantes nas questões

Phrase	Frequency	Support
administracao publica	968	678
ato administrativo	672	456
servidor publico	428	348
servico publico	417	308
atos administrativo	389	275
contrato administrativo	346	264
interesse publico	303	267
processo administrativo	300	223
regime juridico	255	200
empresa publica	234	172
economia mista	221	171

No conjunto de questões de Direito Administrativo ocorreram diversas referências de questões semelhantes em provas distintas. A maioria delas se repete pelo menos uma vez. Nas tabelas 5.17 e 5.18 são apresentados dois casos em que as questões se repetem. O texto em azul reflete o trecho em que os textos estão idênticos.

Tabela 5-17 – Questão repetida em duas provas diferentes

Prova x	Prova y
alteracao unilateral contrato administrativo ocorrer houver modificacao projeto tecnico contratado existe limite percentual acrescimos supressoes objeto contratado faculdade alteracao unilateral contrato corresponde obrigacao publico manter equilibrio economico-financeiro avenca insere-se chamadas clausulas exorbitantes contratos administrativos determinada judiciario	alteracao unilateral contrato administrativo ocorrer houver modificacao projeto tecnico contratado existe limite percentual acrescimos supressoes objeto contratado faculdade alteracao unilateral contrato corresponde obrigacao publico manter equilibrio economico-financeiro avenca insere-se chamadas clausulas exorbitantes contratos administrativos determinada judiciario

Tabela 5-18 – Questão repetida em duas provas diferentes

Prova x	Prova y
atividade administracao publica limitando disciplinando direito interesse liberdade regula pratica ato abstencao fato razao interesse publico concernente seguranca higiene ordem costumes disciplina producao mercado exercicio atividades economicas dependentes concessao autorizacao publico tranquilidade publica respeito propriedade direitos individuais coletivos consiste exteriorizacao policia regulamentar disciplinar hierarquico discricionario	atividade administracao publica limitando disciplinando direito interesse liberdade regula pratica ato abstencao fato razao interesse publico concernente seguranca higiene ordem costumes disciplina producao mercado exercicio atividades economicas dependentes concessao autorizacao publico tranquilidade publica respeito propriedade direitos individuais coletivos definicao presente art codigo tributario nacional aplica-se hierarquico revisional vinculado policia

Em relação à distribuição por organizadores de concursos, a tabela 5.19 mostra a freqüência de questões por entidade organizadora. As entidades organizadoras FCC e ESAF aparecem com maior freqüência no conjunto de questões. Do total de questões, 38,40% abrangem questões extraídas de concursos promovidos pela entidade organizadora FCC. Junto com a ESAF, esse percentual fica próximo de 64,50%, ou seja, quase 2/3 das questões de Direito Asministrativo foram extraídas de concursos feitos por estas entidades.

Tabela 5-19 - Estatística de distribuição das questões por Organizador.

Organizador	1 C...	1.5 Perc...
FCC	1,370	38.40
ESAF	931	26.09
VUNESP	206	5.77
CESPE	185	5.18
NCE/UFRJ	154	4.32
FEC	80	2.24
OAB/DF	67	1.88
FAPEU	47	1.32
CESGRANRIO	38	1.07
TRT 14ª	34	0.95
TRT 12ª	27	0.76
FESAG	24	0.67
MPT	23	0.64
MPU	23	0.64

5.3.2 Classificação

Avaliou-se o desempenho de 2 tipos de classificadores sobre uma base com 3285 registros, 6 atributos e 13 classes. Os classificadores utilizados são implementações dos métodos Naive Bayes e SVM, contidos no software PolyAnalyst.

Para avaliação dos resultados, optou-se pela visualização através de uma matriz de confusão. Esta matriz mostra como os erros de classificação foram distribuídos. O modelo é considerado bom quando os valores da diagonal principal da matriz são altos, enquanto os outros são próximos ou, iguais a zero.

O objetivo desta tarefa neste trabalho foi de constatar se a base está classificada de acordo pelo especialista. Uma boa classificação das questões resultará numa avaliação mais precisa dos links que serão gerados posteriormente.

5.3.2.1 Bayesiano Simples

	Serviços Públicos	Controle da Administr	Licitação Pública	Princípios Fundamen	Processo Administrat	Atos Administrativos	Bens Públicos	Orgãos e Agentes Pú	Contratos Administral	Responsabilidade Civ	Servidores Públicos	Ética na administraçã	Poderes e Deveres	Undefined
Serviços Públicos	124	0	1	1	1	0	3	2	0	0	2	1	0	0
Controle da Administração	0	148	0	0	5	6	1	2	0	0	1	8	0	0
Licitação Pública	2	1	348	4	1	1	1	3	15	0	1	0	0	0
Princípios Fundamentais	0	1	0	147	0	4	0	0	0	0	0	0	0	0
Processo Administrativo	0	0	0	3	124	0	1	1	0	2	2	3	0	0
Atos Administrativos	4	4	0	4	5	484	1	0	0	0	0	0	7	0
Bens Públicos	3	0	4	1	1	0	176	3	1	0	2	0	1	0
Orgãos e Agentes Públicos	0	1	0	0	0	2	1	230	0	0	1	0	0	0
Contratos Administrativos	6	3	11	0	0	0	2	1	226	1	0	1	1	0
Responsabilidade Civil do Estado	1	1	0	0	0	0	1	2	0	137	0	1	0	0
Servidores Públicos	0	6	1	0	10	5	0	24	1	2	517	32	0	0
Ética na administração	0	3	1	0	10	1	0	4	0	2	20	190	0	0
Poderes e Deveres	1	1	1	0	0	12	1	2	1	0	0	0	145	0

Figura 5-11 – Matriz de Confusão do Classificador Bayesiano Simples

Tabela 5-20 - Eficiência do classificador

Total de Erro de Classificação:	9.1047 %
Probabilidade de Classificação:	90.8952 %
Eficiência:	88.8765 %

Tabela 5-21 – Erros de classificação por categoria

Serviços Públicos :	8.1481%	Órgãos e Agentes Públicos :	2.5531%
Controle da Administração :	13.4502%	Contratos Administrativos :	9.9206%
Licitação Pública :	7.9575%	Responsabilidade Civil do Estado :	4.1958%
Princípios Fundamentais :	3.2894%	Servidores Públicos :	13.5451%
Processo Administrativo :	8.8235%	Ética na administração :	17.7489%
Atos Administrativos :	5.1080%	Poderes e Deveres :	10.9756%
Bens Públicos :	8.3333%		

O classificador Bayesiano obteve excelente desempenho para as categorias “Princípios Fundamentais”, “Atos Administrativos”, “Órgãos e Agentes Públicos” e “Responsabilidade Civil do Estado”. A categoria que obteve o pior desempenho no percentual de acertos foi a de “Etica na Administração”, com taxa de erro de 17, 75%.

5.3.2.2 Classificação Linear – Algoritmo SVM

real/pred	Serviços Públicos	Controle da Administração	Licitação Pública	Princípios Fundamentais	Processo Administrativo	Atos Administrativos	Bens Públicos	Orgãos e Agentes Público	Contratos Administrativos	Responsabilidade Civil do	Servidores Públicos	Ética na administração	Poderes e Deveres	Undefined
Serviços Públicos	134	0	0	0	0	0	0	0	0	0	1	0	0	0
Controle da Administração	0	171	0	0	0	0	0	0	0	0	0	0	0	0
Licitação Pública	0	0	377	0	0	0	0	0	0	0	0	0	0	0
Princípios Fundamentais	0	0	0	152	0	0	0	0	0	0	0	0	0	0
Processo Administrativo	0	0	0	0	136	0	0	0	0	0	0	0	0	0
Atos Administrativos	0	1	0	0	0	508	0	0	0	0	0	0	0	0
Bens Públicos	0	0	0	0	0	0	192	0	0	0	0	0	0	0
Orgãos e Agentes Públicos	0	0	0	0	0	0	0	235	0	0	0	0	0	0
Contratos Administrativos	0	0	0	0	0	0	0	0	252	0	0	0	0	0
Responsabilidade Civil do Estado	0	0	0	0	0	0	0	0	0	143	0	0	0	0
Servidores Públicos	0	0	0	0	0	0	0	0	0	0	598	0	0	0
Ética na administração	0	0	0	0	0	0	0	0	0	0	1	230	0	0
Poderes e Deveres	0	0	0	0	0	0	0	0	0	0	0	0	164	0

Figura 5-12 – Matriz de Confusão na Classificação Linear – SVM

Tabela 5-22 - Eficiência do classificador

Total de Erro de Classificação:	0.0910%
Probabilidade de Classificação:	99.9089%
Eficiência:	99.8887%

Tabela 5-23 – Erros de classificação por categoria

Serviços Públicos :	0,7407%	Órgãos e Agentes Públicos :	0,0000%
Controle da Administração :	0,0000%	Contratos Administrativos :	0,0000%
Licitação Pública :	0,0000%	Responsabilidade Civil do Estado :	0,0000%
Princípios Fundamentais :	0,0000%	Servidores Públicos :	0,0000%
Processo Administrativo :	0,0000%	Ética na administração :	0,4329%
Atos Administrativos :	0,1965%	Poderes e Deveres :	0,0000%
Bens Públicos :	0,0000%		

O classificador SVM obteve excelente resultado para esta base de questões de direito administrativo. Das 13 categorias existentes, somente “Serviços Públicos”, “Atos Administrativos” e “Ética na administração” tiveram erro na classificação. Além disso, o percentual de erro nestas categorias está em um nível bem satisfatório. A eficiência do classificador é de 99,88%.

5.3.2.3 Análise dos Resultados da Classificação

Os dois algoritmos foram satisfatórios nos resultados obtidos para a classificação da base de questões de direito administrativo. O SVM teve uma leve vantagem sobre o Bayesiano, pois sua eficiência na classificação foi superior em 10%. A eficiência do classificador bayesiano foi de 88.8765 % enquanto que o SVM foi de 99,88%.

5.3.3 Agrupamento

Após alguns experimentos iniciais, foi adotada uma configuração de parâmetros que resultou na descoberta de 36 grupos. A figura 5.13 apresenta o gráfico de distribuição para a tarefa de agrupamento

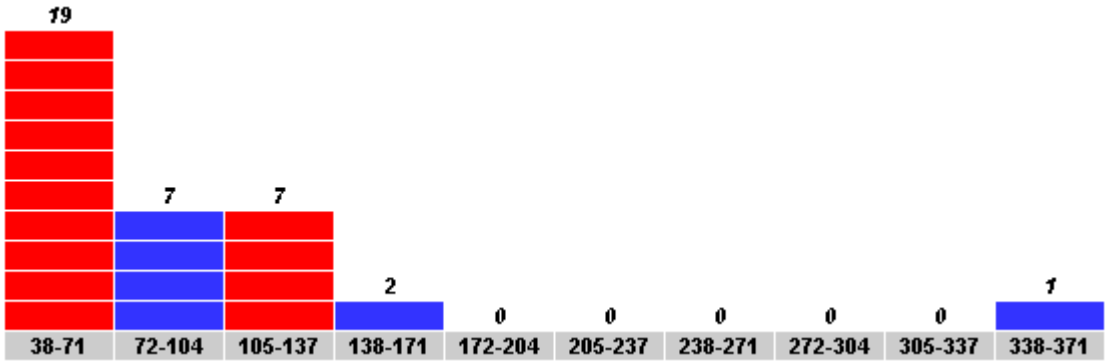


Figura 5-13 - Distribuição de textos nos Grupos de D.Administrativo

Tabela 5-24 – Formação dos grupos para a base de D.Administrativo

1	Description	1 Text count
1	empresa publica sociedades; publica sociedades economia mista;	56
2	ambito administracao publica; administracao publica federal;	39
3	controle externo; tribunal contas uniao;	52
4	pessoa juridica direito; juridica direito privado; juridica direito publico;	93
5	penalidade; processo administrativo; administrativo disciplinar;	145
6	concessao; servico publico;	131
7	publicos civis; servidore publicos;	72
8	agente publico; improbidade administrativa;	86
9	uso comum; ben publicos;	55
10	tomada precos; modalidade licitacao;	65
11	juridica direito; personalidade juridica;	42
12	contratado; contrato administrativo;	118
13	mista; empresa; economia;	66
14	cargo; servidor;	371
15	regime; juridico; publicos; servidore;	130
16	dia; prazo;	155
17	pessoa; privado; juridica;	75
18	principio; legalidade; moralidade;	132
19	entidade; indireta;	85
20	autoridade; competente;	130
21	auto; atributo;	46
22	dano; civil; culpa; agente; causado; objetiva; responsabilidade;	110
23	licitatorio; procedimento;	82
24	precos; tomada;	44
25	efetivo; nomeacao; provimento;	47
26	efeitos; revogacao;	129
27	eficiencia; principios;	61
28	economico; financeiro;	54
29	convite; contrato; licitacao; modalidade; concorrencia;	98
30	concedente; concessionaria;	38
31	presuncao; legitimidade;	51
32	imoveis; alienacao;	46
33	rescisao; unilateral;	51
34	republica; presidente;	45
35	estagio; probatorio;	59
36	executoriedade; imperatividade;	52

5.3.3.1 Análise dos Resultados do Agrupamento

Na configuração adotada para a geração dos grupos, o parâmetro de distância entre os grupos ficou com o valor de 0,3, onde o máximo é 1. Quando este valor é elevado, mais próximo de 1, o número de grupos formados diminui e o número de documentos sem agrupamento aumenta. Com essa configuração encontrou-se 36 grupos. Na distribuição dos grupos, verifica-se que 19 grupos são constituídos com uma média entre 38 e 71 documentos, 7 grupos são constituídos com uma média entre 72 e 104 documentos, 7 grupos são constituídos com uma média entre 105 e 137 documentos, 2 grupos são constituídos com uma média entre 138 e 171 documentos, enquanto que 1 grupo contém entre 338 e 371 documentos. Dos 3295 documentos iniciais, 6% (ou 184 documentos) não foram alocados a nenhum grupo.

Na figura 5.14 abaixo, as bolas azuis representam os Assuntos e as bolas verdes representam os grupos formados.

Alguns grupos refletem as categorias pré-estabelecidas, outros trouxeram novas informações sobre a base, que não foram encontradas através de consultas simples. Por exemplo, o grupo 14, formado pelos termos “cargo” e “servidor” (tabela 5.24), aparece como o grupo com a maior contagem de questões relacionadas a um mesmo assunto. Esse grupo está correlacionado com os assuntos “Servidores Públicos” e “Ética na Administração”, como se observa na figura 5.14. Apesar de ter ligação com os 2 assuntos, sua ligação mais forte, onde o suporte e a significância são elevados é com o assunto “Servidores Públicos”.

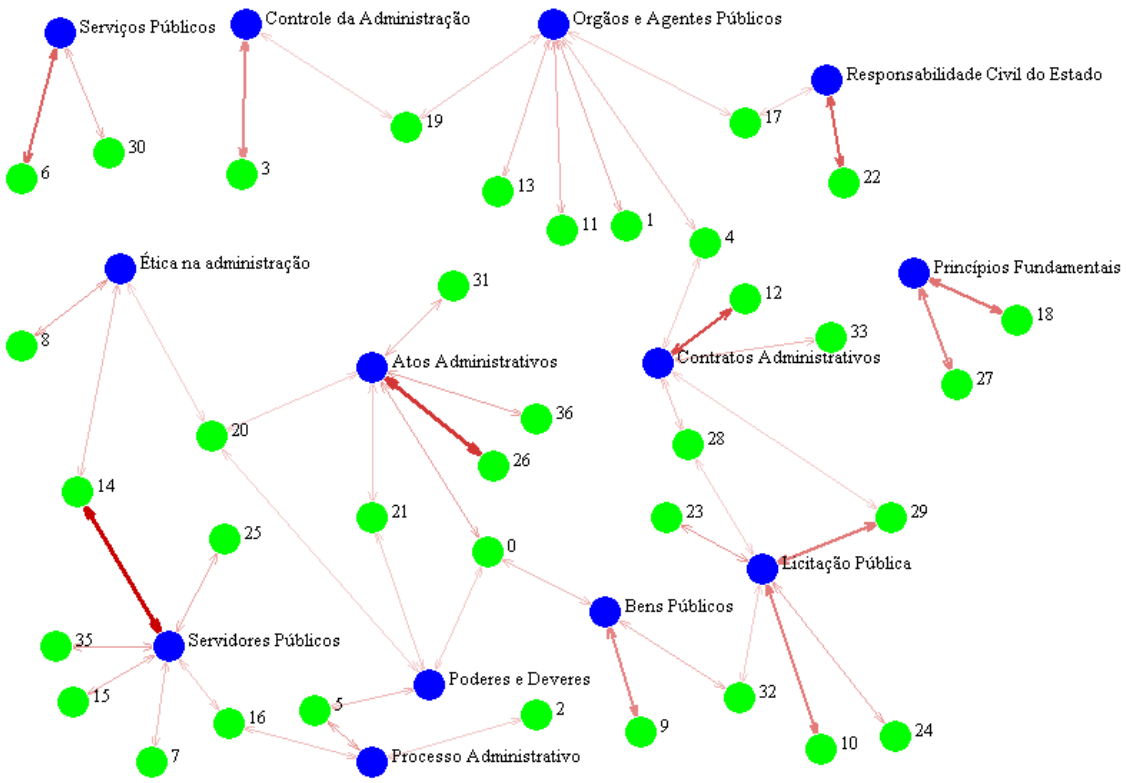


Figura 5-14 - Correlação entre as Categorias(assuntos) e os 36 grupos formados.

5.3.4 Análise de Links

Os parâmetros configurados para a execução da tarefa de análise de links para a base de direito administrativo foram os seguintes:

- Mostrar links: positivos
- Significância min: 6,0
- Links mais significantes: 3000
- Suporte min: 0,0

Na figura 5.15 as bolas verdes representam a entidade organizadora e as bolas azuis o assunto cobrado na questão. As setas vermelhas indicam a força da relação. Quanto mais intensa a cor, maior a relação. Observa-se forte ligação entre o CESGRANRIO com o assunto “Licitação Pública”. Outra ligação forte se observa entre a organizadora FCC e o assunto “Ética na Administração”.

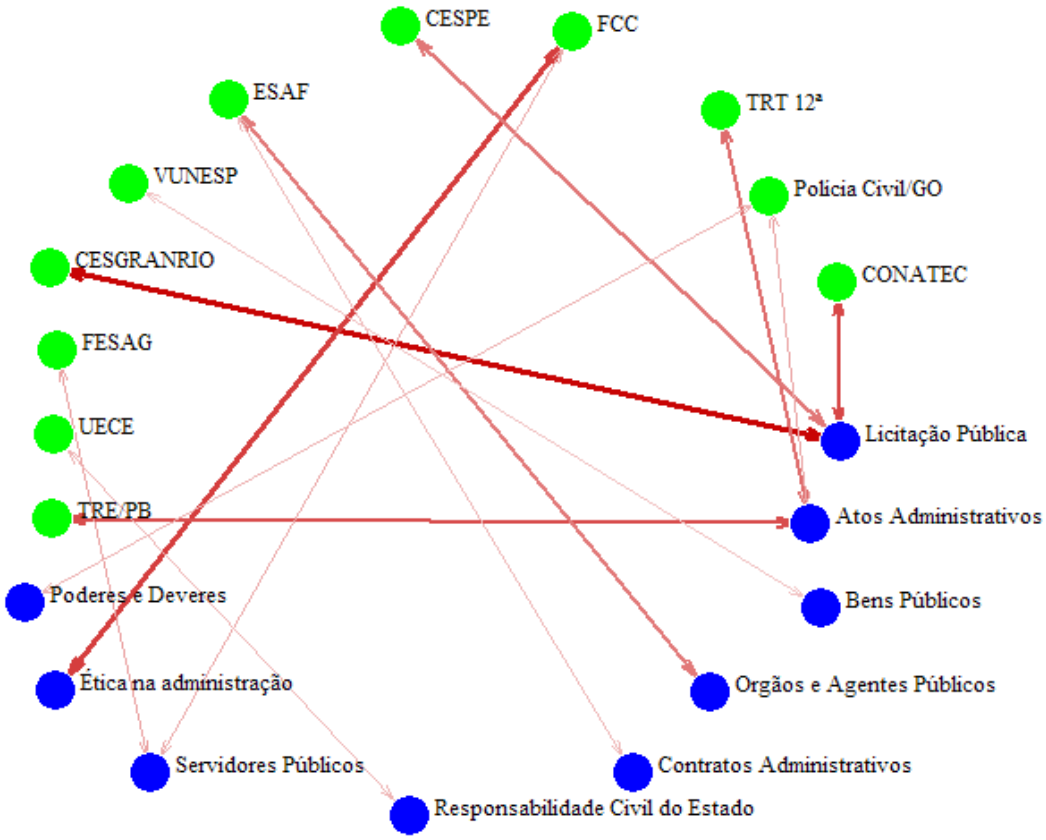


Figura 5-15 - Link entre Assunto e Organizadores de Concurso

Outro estudo foi feito entre Assuntos e Cargos no serviço público – figura 5.16. Os cargos são representados pelas bolas verdes e os assuntos são representados pelas bolas azuis. Para o Cargo de “Arquivista - SPU” o assunto “Bens Públicos” é muito importante, pois em várias questões existe esta ligação entre estes dois campos. Outra ligação forte foi observada nos cargos de “Técnico Judiciário - Administrativa” que apresenta forte ligação com o assunto “Servidores Públicos”.

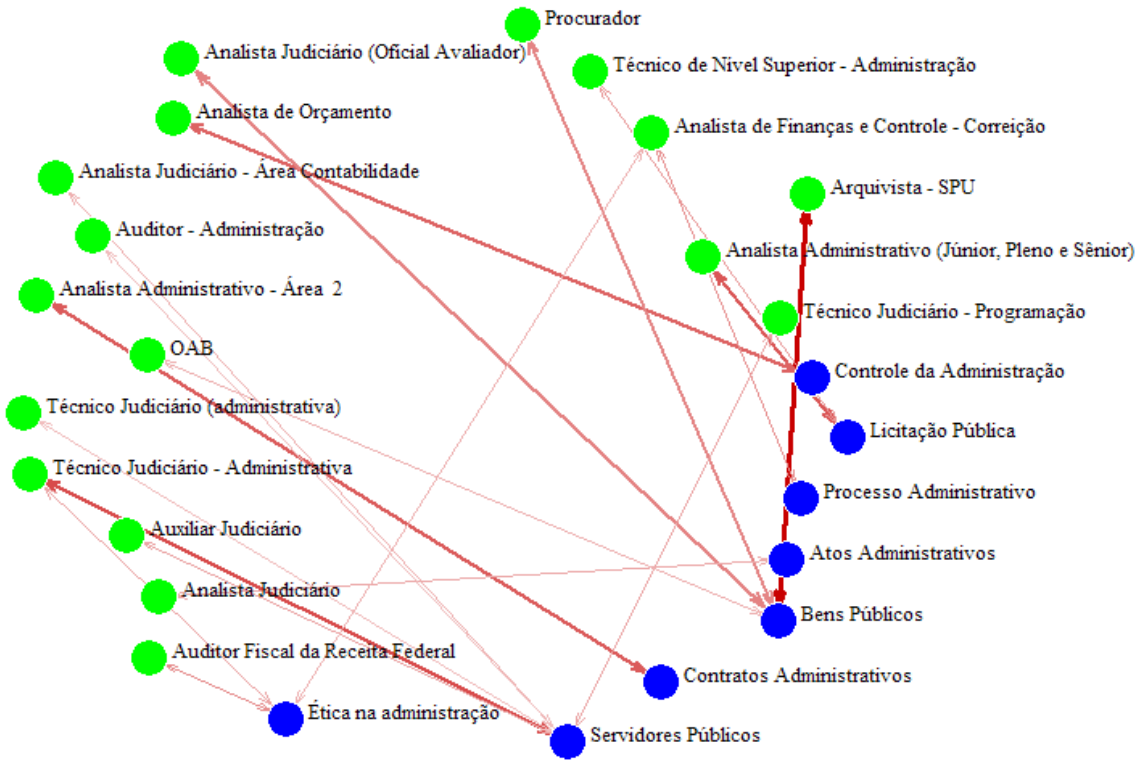


Figura 5-16 - Link entre Assunto e Cargo

Em relação aos *links* entre os grupos formados (36 clusters) e as entidades organizadoras de concursos públicos, da figura 5.17 pode-se extrair informações das ligações existentes entre os grupos, entre as organizadoras, e também entre ambos. Observa-se que o grupo 23(“licitatorio; procedimento”) tem alta relevância para o CESPE, onde a significância da relação apresentou o valor mais alto entre todas as ligações, de 10,10, porém o seu suporte foi de 14. Os grupos 14(“cargo; servidor”), 16(“dia; prazo”) e 20(“autoridade; competente”) estão ligados com alta relevância com a entidade FCC. A ligação do grupo 14 tem suporte de 184 e significância de 9,74, configurando esta relação como a mais forte entre as apresentadas.

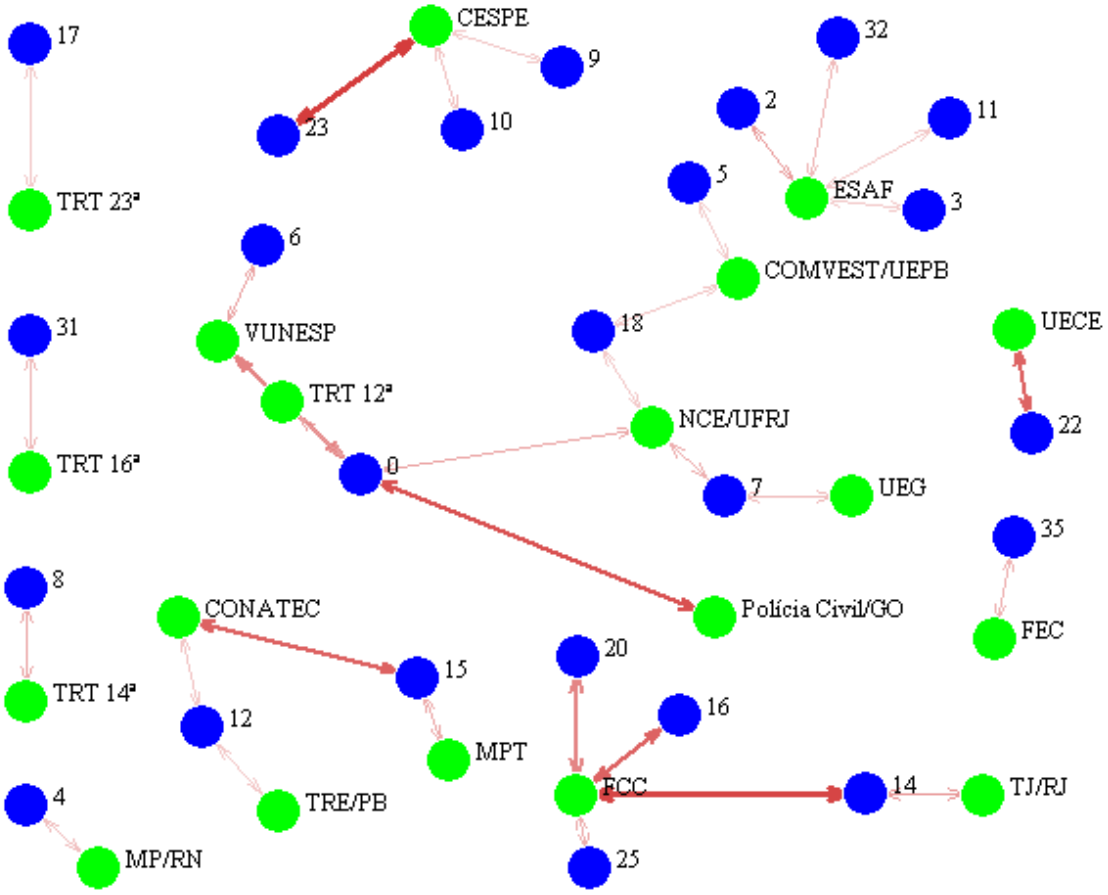


Figura 5-17 - Link entre Organizador e Grupos

As relações entre cargos e grupos são vistas na figura 5.18. Nesta relação observa-se o grupo 23 com ligação com o cargo de Analista administrativo com alta significância. O grupo 14 está ligado a 8 cargos, sendo o cluster mais correlacionado aos cargos apresentados. Este grupo apresenta alto suporte e alta significância com o cargo de Técnico Judiciário – Administrativa.

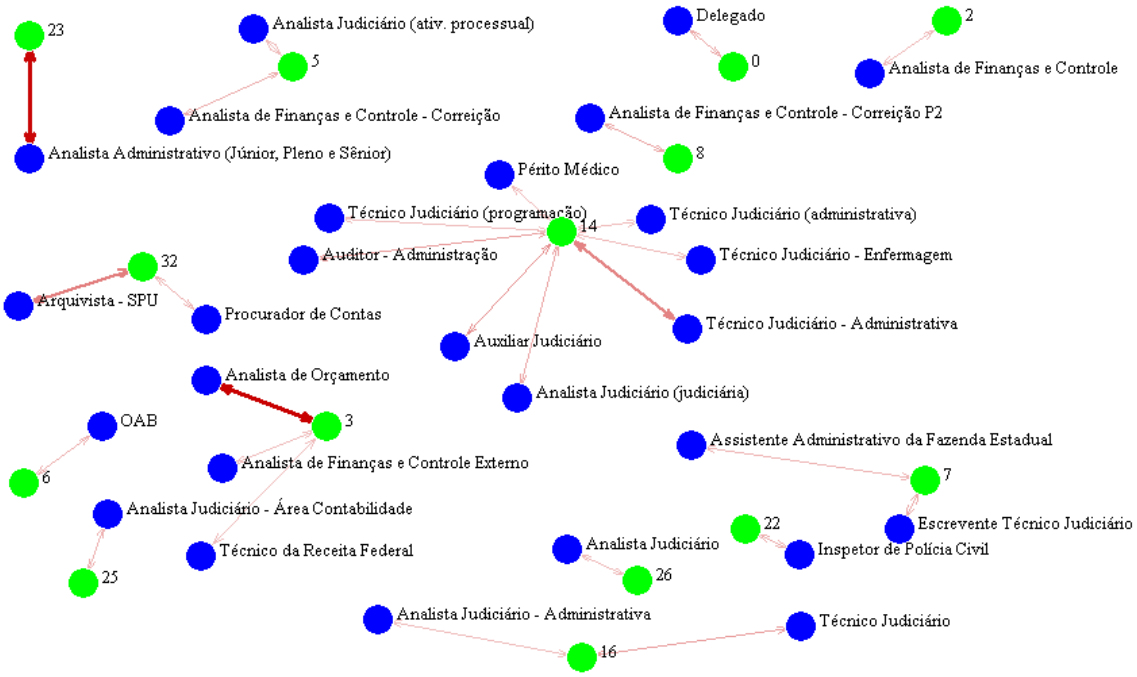


Figura 5-18 - Cargo e Grupos

6 Conclusão

No presente trabalho foi apresentado um conjunto de técnicas e algoritmos já existentes e consagrados, para a solução de problemas relacionados com descoberta de conhecimento em textos, mais precisamente provas extraídas da *web*, escritas na língua portuguesa do Brasil. Foram descritas as técnicas de classificação e de agrupamento (*clustering*) aplicadas a documentos, bem como a avaliação de links existentes entre os documentos.

Um dos principais problemas encontrado durante a fase de testes e tratado neste trabalho foi a dificuldade de se portar um arquivo no formato PDF, para uma base de dados textual limpa, sem os erros ocasionados por conversores automáticos de arquivos PDF para texto. A maioria das provas é encontrada na *Web* em formato PDF, o que dificulta a entrada de dados de origem para a pesquisa. O esforço inicial para conversão dos dados em texto pôde ser compensado com o resultado da pesquisa, que trouxe informações importantes e de grande valia referente às questões de concursos.

Para a etapa de pré-processamento desenvolveu-se uma ferramenta, com o objetivo de sistematizar todo o processo de preparação dos dados, desde o momento da coleta das provas até a saída do arquivo pronto para ser executado pelos algoritmos de mineração de textos. Neste processo os textos referentes às questões (enunciado e alternativa) foram tratados de forma adequada, com remoção de *stopwords*, retirada de acentos, integração de caixa e remoção de caracteres indesejados. Esse processo foi executado exaustivamente, modificando-se constantemente a *stoplist*, até se encontrar um texto limpo, sem a presença de palavras que pudessem confundir a interpretação pelos algoritmos de mineração. A técnica de *stemmer* foi implementada, mas não foi utilizada, pois os resultados não foram satisfatórios. Esta ferramenta teve o objetivo de preparar os dados para a fase seguinte da mineração: o processamento dos documentos.

Para processar os documentos, optou-se pelo uso do software PolyAnalyst da empresa Megaputer, pois se adequava perfeitamente para os objetivos propostos neste trabalho. Deste software foram utilizadas as ferramentas de classificação de textos, agrupamento de textos e análise de links, além das ferramentas de exploração de dados textuais, que foram muito úteis na descoberta de similaridade entre questões, através de análise estatística de co-ocorrências de palavras consecutivas dentro do texto.

A fim de demonstrar uma aplicação prática das tarefas de mineração, foram apresentados dois estudos de casos: questões de informática e questões de direito

administrativo. Tanto na base de informática quanto na base de direito administrativo observou-se as similaridades entre questões, associando-as às entidades organizadoras, aos cargos e aos assuntos cobrados nas provas.

Na etapa de exploração dos dados foram constatadas evidencias de questões repetidas em provas de concursos distintos além de destacar quais os termos mais freqüentes nas questões.

Na etapa de classificação, os classificadores SVM e Naive Bayes tiveram um bom desempenho, apresentando percentuais acima de 90% para a eficiência na classificação. O objetivo desta etapa foi constatar a distribuição das questões pelos assuntos correspondentes, de acordo com a definição do especialista. Nas questões de informática foi observado que as categorias “Conceitos Básicos”, “Arquivos” e “Softwares” apresentaram um percentual maior de erro nas classificações com o classificador Naive Bayes.

O processo de agrupamento revelou mais grupos do que originalmente havia sido definido. Buscaram-se várias configurações de grupos que pudessem retornar uma quantidade equivalente de categorias, mas isso não foi possível, pois nos testes realizados, quanto menor o número de grupos, mais documentos ficavam de fora dos grupos, sem características que o incluíssem em algum dos grupos formados. O algoritmo utilizado ajustou os grupos de acordo com a similaridade entre os termos e as frases encontradas nas questões.

As correlações através da técnica de Análise de Links evidenciaram a força das ligações existente entre os assuntos e os grupos, entre os organizadores e entre os cargos. Esta técnica foi a que revelou maiores descobertas para este tipo de base de dados. Através da análise feita nesta etapa, foram descobertos quais os assuntos mais cobrados nas questões por entidade organizadora, quais os assuntos mais cobrados por cargos e também quais os termos mais importantes para determinado assunto. Além disso, foram correlacionados os grupos aos cargos e às entidades organizadoras, sendo analisada a força da ligação através das medidas de suporte e significância.

A metodologia aplicada atendeu plenamente aos objetivos propostos. Os testes e avaliações apresentados mostraram a eficiência e aplicabilidade da solução proposta para o problema de mineração de textos para questões de concursos.

A visualização dos resultados se mostrou bastante informativa e de grande valia para a interpretação dos mesmos.

6.1 Futuros Trabalhos

Para trabalhos futuros, propõem-se experimentos com novas questões, contendo grandes volumes de documentos; Aumentar o número de estudos de casos, com a inclusão de outras disciplinas; Melhorar a interface do sistema para agilizar o processo de entrada dos documentos; aumentar a *Stoplist,* para abranger mais termos da língua portuguesa por disciplina; Adicionar suporte a algoritmos de classificação e agrupamento pela ferramenta proposta; Desenvolver um Thesaurus específico por disciplina.

Referências Bibliográficas

[1] TAN, A., “Text mining: The state of the art and the challenges”. In: Proceedings of the Pacific Asia Conf on Knowledge Discovery and Data Mining PAKDD’99 workshop on Knowledge Discovery from Advanced Databases, pp. 65–70, 1999.

[2] EBECKEN, N. F. F., LOPES, M. C. S., COSTA, M. C. A., “Mineração de textos”. In: *Sistemas inteligentes: fundamentos e aplicação. Barueri Manole. cap.*13, p. 337-370, São Paulo, 2003.

[3] FELDMAN, R. & H. HIRSH, *Exploiting background information in knowledge discovery from text*. Journal of Intelligence Information Systems, 1997.

[4] DÖRRE, J., P. GERSTL, & R. SEIFFERT. Text mining: finding nuggets inmountains of textual data. In: *KDD ’99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, 1999.

[5] MARTINS, Claudia Aparecida. Uma abordagem para pré-processamento de dados textuais em algoritmos de aprendizado. Tese de Doutorado (ICMC-USP), São Paulo, 2003.

[6] WIVES, Leandro Krug. *Tecnologias de Descoberta de Conhecimento em Textos Aplicadas à Inteligência Competitiva*. Programa de Pós-Graduação em Computação (UFRGS), Rio Grande do Sul, 2002.

[7] PORTER, M. F. *An Algorithm for Suffix Stripping*. Program, 14(3): 130-137, 1980.

[8] LOVINS, J. B. *Development of a stemming algorithm*. Mechanical Translation and Computational Linguistics 11(1-2), 22–31, 1968.

[9] ORENGO, V.; Huyck, C., “A Stemming Algorithm for Portuguese Language”. In: *Eigth Symposium on String Processing and Information Retrive(RIJ 79).* val (SPIRE 2001), Chile, pp. 186-193, 2001.

[10] TICON, Alexandre. *Aplicação das técnicas de mineração de textos em sistemas especialistas na liquidação de processos trabalhistas*. Dissertação de Mestrado (COOPE-UFRJ), Rio de Janeiro, 2007.

[11] YANG, Y; PEDERSON, J. “A Comparative Study on Feature Selection in Text Categorization”. In: *Proceedings of 14th International Conference on Machine Learning*, San Francisco, USA, p. 412-420, 1997.

[12] SEBASTIANI, F. “Machine learning in automated text categorization”. *ACM Computing Surveys*, vol. 34, no. 1, pp.1-47, 2002.

[13] JOACHIMS, T. “Learning to Classify Text Using Support Vector Machines”. *Kluwer Academic Publishers*, 2002.

[14] FAYYAD, U; SHAPIRO, G. SMYTH, P. “From Data Mining to Knowledge Discovery: An Overview”. In: *Advances in Knowledge Discovery and Data Mining*. Menlo Park: AAAI Press, 1996. 611 p. p.11-34.

[15] HABN, U; MANI, I. “The challenges of Summarization”. *IEEE Automatic Computer 33(11)*, p. 29-36, 2000.

[16] KORFHAGE, Robert. *Information Retrieval and Storage*. New York: John Wiley & Sons, 1997.

[17] KOWALSKI, Gerald. *Information Retrieval Systems: Theory and Implementation*. Boston: Kluwer Academic Publishers, 1997.

[18] SALTON, Gerard; MACGILL, Michael J. *Introdution to Modern Information Retrieval*. New York: McGRAW-Hill, 1983.

[19] RIJSBERGEN, C. *Information Retrieval*. 2ed. London: Butterworths, 1979.

[20] LANCASTER, F. Information Retrieval Systems: Characteristics, Testing and Evaluation. New York: John Wiley & Sons, 1968.

[21] YANG, Y. and LIU, X. “An evaluation of statistical approaches to text categorization”. *Journal of Information Retrieval*, v.1, n.1/2, p.67-88, 1999.

[22] REZENDE, S. O.; OLIVEIRA, R. B. T.; FELIZ,L. C. M.; ROCHA, C. A. J. “Visualization for Knowledge Discovery in database”. Em N. F. F. Ebecken (ed.), Data Mining, England, WT Press – Computational Mechanics Publications, p.81-95, 1998.

[23] BASTOS, V. M. *Ambiente de Descoberta de Conhecimento na Web para a Língua Portuguesa*. Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, 2006.

[24] DUMAIS, S., PLATT, J., HECKERMAN, D., and SAHAMI, M., “Inductive learning algorithms and representations for text categorization”. In: *Proceedings of the th1998 ACM 7 international conference on information and knowledge management*, p.148-155, 1998.

[25] AL-SHALABI, RIYAD; KANAAN. GHASSAN; GHARAIBEH, MANAF H., 2004, “Arabic Text Categorization Using kNN Algorithm”. Disponível em: < www.ijicis.net/Vol6_No1%20No_1.pdf >. Acesso em: 20 Out 2008.

[26] WIVES, Leandro Krug. *Um estudo sobre Agrupamento de Documentos Textuais em Processamento de Informações não Estruturadas Usando Técnicas de Clustering*. Dissertação de Mestrado. Programa de Pós-Graduação em Computação (UFRGS), Rio Grande do Sul, 1999.

[27] DANTAS, Marco Aurélio. *Implementação de metodologia de categorização de textos científicos*. Dissertação de Mestrado (COOPE-UFRJ), Rio de Janeiro, 2007.

[28] LOPES, M. C. S. *Mineração de dados textuais utilizando técnicas de clustering para o idioma português*. Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, 2004.

[29] PINHEIRO, José C.; SUN, Don X. “Methods for Linking and Mining Massive Heterogeneous Databases”. In: *KDD98*, 1998.

[30] GIBBS, M. Math to fight spam. Network World, Southborough. MA, p.30, Set.2003.

[31] DORNAN, A. “Lesson 188: Bayesian Spam Filtering”. *Network Magazine*, Manhasset, NY, p.64, Mar.2004.

[32] BILLSUS, D.; PAZZANI, M. A Hybrid User Model for News Story Classification. 1999. Proceedings of the Seventh International Conference on User Modeling (UM’99), Banff, Canada.

[33] THEODORIDAS, S. and KOUTROUMBAS, K. “Pattern Recognition”. *Academic Press*, p. 351-495, 1998.

[34] JAIN, A. K.; MURTY, M. N.; and FLYNN, P. J. “Data Clustering: A Review”. *ACM Computing Surveys 31 (3): 26423*, 1999.

[35] COLE, R. M. *Clustering with Genetic Algorithms*, M. Sc., Department of Computer Science, University of Western Australia, Australia, 1998.

[36] DUMAIS, S. T. and CHEN, H. “Hierarchical classification of web content”. In: *Proceedings of SIGIR'00*, p. 256-263, Ago. 2000.

[37] ZHANG, L., Zhu, J. Yao, T. “An evaluation of statistical spam filtering techniques”. In: *ACM Transcations on Asian Language Information Processing*. Vol. 3, nº 4, pp 243-269, 2004

[38] JSP. *Java Server Pages Technology*, Disponível em http://www.netbeans.org/ index_pt_BR.html. Acesso em 09 mar 2008.

[39] Campione, M.; Walrath, K., The Java Tutorial: Object-Oriented Programming for the Internet, SunSoft Press, 1996.

[40] JSP. *Java Server Pages Technology*, Disponível em: <http://java.sun.com/ products/jsp/whitepaper.html>, último acesso em 09 nov 2008.

[41] GRADY, Nancy W.; TUFANO, Daniel R.; FLANERY JUNIOR, Raymond E. “Immersive Visualization for Link Analysis”. In: *AAAI Fall Symposium on Artificial Intelligence and Link Analysis*, 1998.

[42] LYONS, Donal; TSEYTIN, Gregory S. “Phenomenal Data Mining and Link Analysis”. In: *AAAI Fall Symposium on Artificial Intelligence and Link Analysis*, 1998.

[43] MVC, “Model-View-Controller”. *Java BluePrints*, 2008. Disponível em < http://java.sun.com/blueprints/patterns/MVC-detailed.html>. Acesso em 07 nov 2008.

[44] MYSQL. *Banco de dados MySQL*. Disponível em <http://www.mysql.com/why-mysql>. Acesso em 09 mar 2008.

[45] SERPA, Antonio. *Aîuri: Um Portal para Mineração de Tex-tos Integrado a Grids Computacionais*. Dissertação de Mestrado (COOPE-UFRJ), Rio de Janeiro, 2007.

[46] OREN, Z., *Clustering Web Documents: A Phrase-Based Method for Grouping Search Engine Results.* Ph.D. dissertation, University of California at Berkeley, Berkeley, California, USA, 1999.

[47] POLY ANALYST, *Poly Analyst Help*, Poly Analyst 6 – Megaputer Intelligence Inc, 2008 – Tutorial do Software, 2008.

[48] KOENIG, M. E. D. "Information Driven Management: The New, but Little Perceived, Business Zeitgeist". *International Journal of Libraries and Information Services Vol 50*, No 3, p.137-221, 2000.

[49] GOLDSCHMIDT, R. e PASSOS, E. *Data Mining - Um Guia Prático*. Campus, 2005

[50] JENSEN, David. “Statistical Challenges of Inductive Inference in Linked Data”. In: *AAAI Fall Symposium on Artificial Intelligence and Link Analysis*, 1998.

[51] GOLDBERG, Henry G; SENATOR, Ted E. “Restructuring Databases for Knowledge Discovery by Consolidation and Link Formation”. In: *AAAI Fall Symposium on Artificial Intelligence and Link Analysis*, 1998.

Livros Grátis

(http://www.livrosgratis.com.br)

Milhares de Livros para Download:

Baixar livros de Literatura
Baixar livros de Literatura de Cordel
Baixar livros de Literatura Infantil
Baixar livros de Matemática
Baixar livros de Medicina
Baixar livros de Medicina Veterinária
Baixar livros de Meio Ambiente
Baixar livros de Meteorologia
Baixar Monografias e TCC
Baixar livros Multidisciplinar
Baixar livros de Música
Baixar livros de Psicologia
Baixar livros de Química
Baixar livros de Saúde Coletiva
Baixar livros de Serviço Social
Baixar livros de Sociologia
Baixar livros de Teologia
Baixar livros de Trabalho
Baixar livros de Turismo