2
Processo de Mineração de Textos

"O estudo em geral, a busca da verdade e da beleza são domínios em que nos é consentido ficar crianças toda a vida."

Albert Einstein

O processo de mineração de textos pode ser tão complexo quanto se deseje, porém está normalmente sujeito a uma subdivisão em outros macro-processos executados em determinada ordem, caracterizando fases distintas dentro do processo como um todo.

Um sistema completo para descoberta de conhecimento a partir de informações textuais se divide em etapas bem características nas quais se utilizam ramos variados de conhecimento. Esse fluxo procedimental (ARANHA, 2007) pode ser descrito, através de suas fases, pela figura 11:

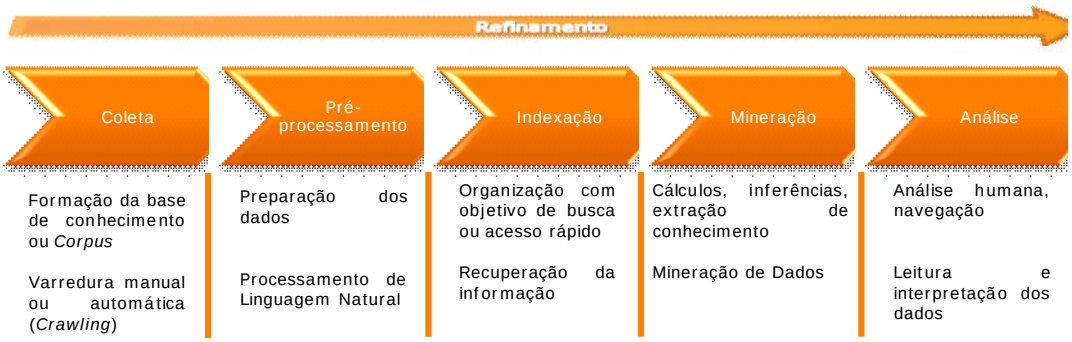

Figura 11 – Etapas do processo de extração de conhecimento em textos.

É importante, porém, destacar que cada fase, conforme a complexidade exigida, pode encerrar dentro, de si, todo ou parte do processo de mineração. Um *crawler* focado, por exemplo, possui toda uma inteligência de reconhecimento textual dos *links* mais interessantes e, por isso, apesar de ser uma ferramenta tipicamente de coleta, realiza também mineração (SOARES, 2008). Por esse motivo, cabe destacar que a possível aparência de uma desordem das fases preconizadas pelo *framework* pode ser explicada pela representação de um fluxo

secundário, constituído o processo inteiro ou apenas parte, dentro de cada etapa do processo em seu fluxo primário, este último apresentado na figura 11.

Esmiuçando cada uma das etapas, de forma geral, e considerando apenas o fluxo primário pode-se dizer que a fase de coleta é a responsável pela construção da base de textos a partir da qual o restante do processo irá se desdobrar. A fase seguinte, de pré-processamento, se destina a preparar os dados tornando-os aptos a serem organizados em formatos mais adequados às tarefas subseqüentes. A indexação é responsável pelo armazenamento na forma de uma estrutura previamente organizada de modo a facilitar a recuperação das informações. A etapa de mineração consiste na manipulação das informações armazenadas, com a utilização de várias técnicas, cálculos e inferências, com o intuito de extrair conhecimento útil e até então escondido. Finalmente, durante a fase de análise busca-se interpretar o conhecimento explicitado durante a fase de mineração visando a torná-lo utilizável como auxílio na tomada de decisão.

2.1
Coleta

A coleta é a tarefa responsável pela aquisição dos elementos sob os quais se apóiam o restante do trabalho. Normalmente se coletam documentos. Para essa finalidade podem ser empregadas várias técnicas, se diferenciando principalmente pelo grau de automatização com que são executadas. Fontes em formato digital auxiliam a automatização do processo. Coleções de textos podem advir de fontes de documentos pré-armazenados de forma organizada ou de fontes completamente distribuídas sem nenhuma estrutura organizada de armazenamento.

Na *Web*, a coleta pode ser realizada de forma automatizada através de *crawlers*. Um *crawler* é um robô que visita páginas na *Web* e repassa as informações coletadas para outro componente responsável pela indexação dessas páginas. A arquitetura mais atual de varredura utiliza vários desses robôs de forma distribuída trabalhando de maneira cooperativa. Executar um *Web crawler* é uma tarefa bastante desafiadora. Existem muitos truques de desempenho, aspectos de confiabilidade, além dos não menos importantes aspectos sociais envolvidos. Esse rastreamento é uma tarefa bastante frágil, pois envolve interação com centenas de

milhares de servidores *web* e vários servidores de nomes, que estão além do controle do sistema (BRIN, 1998). Uma forma variante, e que pode ser mais interessante de se coletar documentos na *Web*, é através do uso de *crawlers* focados, que dispensam a utilização de grandes recursos de *hardware*. Um *crawler* focado é altamente efetivo na construção de coleções de documentos de qualidade sobre tópicos específicos e oriundos da *web*, usando modestos computadores "caseiros" (DOM, 1999). Esses *crawlers* costumam implementar alguma forma de classificação durante a varredura na *Web* de maneira a armazenar apenas os documentos considerados relevantes (SOARES, 2008). Esses classificadores são construídos segundo alguma técnica de inteligência computacional que normalmente utiliza uma fase de treinamento supervisionado ou por reforço de forma que o usuário possa calibrar o que o *crawler* irá considerar relevante. Essa varredura, realizada de forma criteriosa, resulta em coleções de documentos bastante correlacionados, facilitando algumas das tarefas subseqüentes. Além disso, tende a economizar recursos durante a varredura, pois um *crawler* focado ideal deve recuperar o conjunto maximal de páginas relevantes enquanto simultaneamente atravessa o número mínimo de documentos na *Web* (DILIGENT, 2000).

2.2
Pré-processamento

Sistemas de mineração de texto não submetem aos seus algoritmos de descoberta de conhecimento coleções de textos despreparadas. Ênfase considerável em mineração de texto é dada ao que é comumente referenciado como operações de pré-processamento (FELDMAN, 2007). O pré-processamento é necessário para representar o texto numa forma mais estruturada, capaz de alimentar algoritmos de aprendizado de máquinas (GONÇALVES, 2006). Muitos dos tratamentos dados ao texto durante essa fase podem ser feitos tanto de forma automatizada como feito por humanos, porém o desempenho dos sistemas automáticos é extremamente superior (ARANHA, 2007). Nessa etapa palavras são

extraídas de documentos, desconsiderando algumas ***stopwords***[11] cuja utilização está mais relacionada com a organização estrutural das sentenças e não têm poder discriminatório, por exemplo, quanto à classe do texto. Essas *stopwords* são freqüentemente adicionadas de forma manual a uma ***stoplist***[12] e são desconsideradas no momento de extração das características. Esse modelo é conhecido como **saco de palavras**[13] e faz parte de uma abordagem com ampla utilização da estatística. É importante destacar que a utilização de *stopwords* não é obrigatória e pode inclusive não ser desejada, como no caso das máquinas de busca na Internet. Um exemplo para a não utilização de *stopwords*, é a busca da frase "*to be or not to be*", onde todas as palavras poderiam ser consideradas *stopwords* e nesse caso essa consulta não retornaria nenhum resultado, pois nenhuma das palavras seria indexada. Alguns modelos mais elaborados têm se tornado um grande desafio para os pesquisadores. Esses modelos incorporam técnicas de Processamento de Linguagem Natural que costumam torná-los mais complexos. Porém essas tarefas adicionais como análise de diferentes classes de palavras (substantivos, adjetivos, nomes próprios, verbos, etc.) tendem a aumentar a eficiência dos sistemas (KAO, 2004). Dentro das técnicas de pré-processamento baseadas em Processamento de Linguagem Natural podem ser citadas lematização e identificação de colocações dentre outras. Por outro lado a adição dessas técnicas tende também a tornar as ferramentas dependentes da língua usada nos textos. A figura 12 apresenta um possível fluxo para o pré-processamento e textos, onde a única etapa realmente obrigatória é a de atomização.

[11] Palavras muito freqüentes em todos os textos como artigos, pronomes e etc. Não costumam contribuir para o caráter discriminatório do texto que as contém.
[12] Lista de *stopwords*.
[13] Do inglês, *bag of words*.

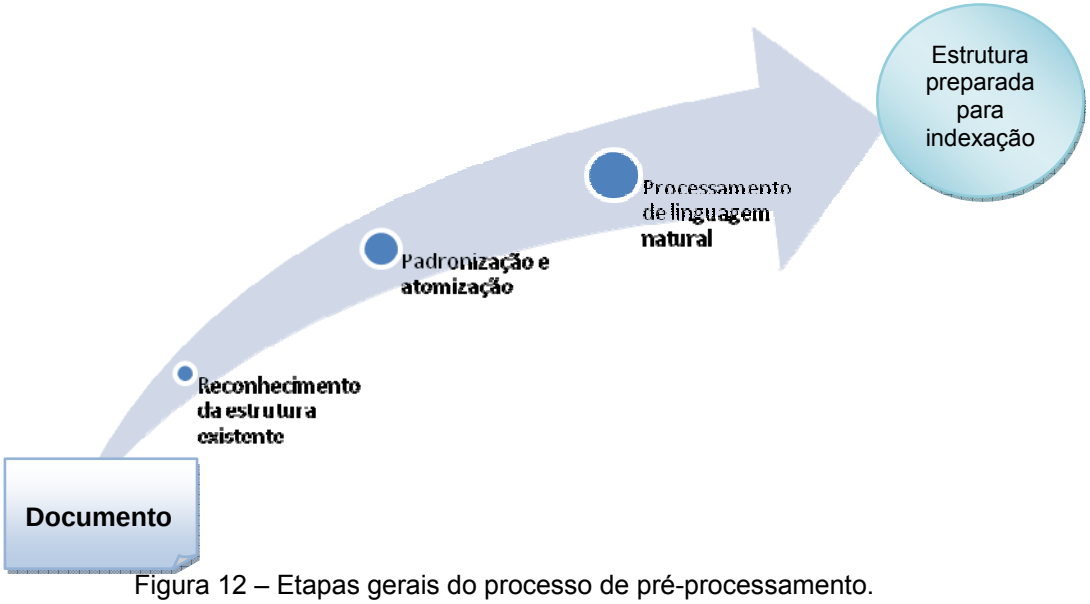

Figura 12 – Etapas gerais do processo de pré-processamento.

O fluxo apresentado tem caráter simplificado, pois cada um dos micro-processos pode se desdobrar em muitos. Por exemplo, supondo que o documento dado como entrada do processo seja um livro, primeiramente pode ser interessante se capturar algum tipo de estrutura como a divisão em capítulos. Durante a padronização podem ser eliminados os espaços vazios adicionais, a acentuação, os caracteres especiais não-indexáveis e etc. O processamento de linguagem natural pode ser desdobrado em várias etapas conforme a complexidade que se queira atribuir. Todas elas visam a utilizar conceitos lingüísticos com o objetivo de contribuir para melhoria do processo de estruturação posterior, seja reduzindo a dimensão dos dados, seja atribuindo maior caráter semântico a esses dados.

2.2.1
Tokenização

O primeiro passo para o processamento de texto escrito é a atomização, amplamente conhecida na literatura como *tokenização*[14]. *Token* é o nome que se dá aos termos extraídos dos textos, sejam eles palavras ou expressões compostas por mais de uma palavra. Um *token* pode também ser um n-grama, ou seja, um

[14] Alguns autoresde língua portuguesa utilizam o termo atomização para fazer referência à tarefa de *tokenização* e dessa forma utilizam o termo átomo para referenciar o termo *token*. Exemplos podem ser encontrados em "http://acdc.linguateca.pt/acesso/atomizacao.html", "http://www.linguateca.pt/Diana/download/SantosRochaAPL2002.rtf", ou "http://www6.ufrgs.br/textquim/relatoriotextquim0305.pdf".

conjunto de n caracteres consecutivos, porém a abordagem mais usual é que os *tokens* extraídos do texto sejam palavras, e nesse caso, o *tokenizador* é auxiliado pelo fato das palavras serem separadas por espaços ou sinais de pontuação, que em alguns casos podem ser considerados *tokens* delimitadores. A base formal para *tokenização* baseada em delimitadores é o uso não-ambíguo de certos caracteres como delimitadores. O alfabeto de letras, **α**, e o alfabeto de delimitadores, **δ**, são disjuntos. Um texto é uma seqüência de letras e delimitadores e, após o processo de *tokenização*, uma seqüência de *tokens*. *Tokenização* baseada em delimitadores não se aplica a algumas línguas como Árabe, Chinês ou Japonês, pois elas não usam sequer o espaço entre os caracteres e um único caractere traduz toda uma idéia tornando a tarefa extremamente mais complexa. Outra dificuldade é a ambigüidade dos delimitadores. Por exemplo, o ponto pode ser usado para abreviar palavras, o hífem para separar sílabas ou até mesmo usado em termos compostos. Por outro lado os *tokens* podem ser obtidos após a utilização de alguma técnica de processamento de linguagem natural, como por exemplo, lematização (*stemming*) que reduz uma palavra ao seu radical. Isso contribui para redução da dimensão do léxico.

2.2.2
Processamento de Linguagem Natural

Muitas técnicas de mineração de texto têm se valido, ultimamente, de inspiração lingüística. O texto é dividido sintaticamente usando informações de uma gramática formal e um léxico e a informação resultante é então interpretada semanticamente e usada para extrair informação sobre o que foi dito (KAO, 2004). Para isso o processamento de linguagem natural faz uso de conceitos lingüísticos como classes de palavras (substantivos verbos, adjetivos, etc.), estrutura das palavras (radical, vogal temática, desinências), formação das palavras (derivação prefixal, sufixal, parassintética, etc.), dentre outros.

Muitos conceitos lingüísticos podem ser utilizados para processamento da linguagem natural e, por conseguinte auxiliar as tarefas de pré-processamento (MOSCALEWSKI, 2002).

2.2.2.1
Identificação de colocações

Algumas vezes, podemos encontrar nos textos conjuntos de palavras que trazem um significado adicional em relação às palavras que o compõem, analisadas em separado. Em outros casos, um conjunto de palavras pode até mesmo ter um significado completamente novo e adaptado, como no caso de expressões idiomáticas. Esses conjuntos especiais de palavras são conhecidos como colocações[15]. Muitas vezes é interessante que o *token* seja composto por esse conjunto de palavras que traduzem uma idéia diferente.

2.2.2.2
Classes gramaticais

As palavras podem ser agrupadas em categorias gramaticais ou sintáticas, conhecidas como **classes de palavras**[16], segundo seus comportamentos sintáticos e semânticos similares. Na língua portuguesa temos três principais:

1. substantivo, usado para nomear entidades;
2. adjetivo, usado para qualificar o substantivo;
3. verbo, usado para expressar uma ação na sentença.

Para a tarefa de reconhecimento de classes de palavras (*Part of Speech* – POS, na literatura extrangeira) uma técnica de aprendizado de máquina que obtém bons resultados é a TBL (*Tranformation Based Learning*) (BRILL, 1995). Essa técnica é baseada em um aprendizado simbólico supervisionado e construção automática de regras lógicas simples do tipo "se-então". O processo é iterativo e as regras, baseadas em um *template* pré-apresentado vão sendo incorporada ou não às regras aprendidas segundo critérios de pontuação de melhoria das classificações a cada interação.

Conforme citado anteriormente, como possível contribuição do uso de processamento de linguagem natural, a redução de dimensão dos dados pode ser

[15] Do inglês c*ollocations*.
[16] Em inglês é conhecido como *Part of Speech* (POS).

exemplificada com o reconhecimento dos substantivos. Os substantivos possuem a capacidade de flexão, que pode ser aplicada com relação ao gênero, número ou grau, na língua portuguesa. O Processamento na Linguagem natural pode ser empregado para passar ao indexador um termo único referente ao substantivo, independente de flexão. Essa tarefa é realizada pelos algoritmos de lematização (*stemming*), que buscam uma forma neutra de representação dos termos baseados em um mesmo radical.

2.2.2.3
Lematização

Freqüentemente, encontram-se nos textos, palavras nas mais diversas formas flexionadas, seja em gênero número ou grau. Uma abordagem interessante é a redução dessas palavras ao mesmo radical e esse processo é conhecido como lematização ou *stemming*. Esse processo unifica em um único termo para indexação, um conjunto de palavras de mesma origem morfológica. Para efeito de entendimento do processo pode-se dizer que lema é a forma normalizada de uma palavra e, assim sendo, a lematização poderia ser explicada como a normalização de termos para um formato padrão da dimensão a qual os termos originais se aplicam. Pode-se fazer uma analogia com o processo de normalização usado em mineração de dados ou ainda da álgebra linear.

Um exemplo de redução ao radical pode ser dado por: caseiros, que pode ser reduzido para caseiro, ou até mesmo para casa. É importante destacar que essa redução a um radical muito simples é interessante no sentido de diminuir a dimensão do vetor que representa o documento e dessa forma facilitar e até mesmo melhorar algumas aplicações de mineração como o agrupamento, por exemplo, que tira proveito da redução de dimensão durante o cálculo de similaridades. Apesar de parecer, intuitivamente óbvia a melhoria do processo com a utilização da lematização, existem muitas controvérsias sobre os benefícios de seu uso (BAEZA-YATES, 1999). Por exemplo, no caso das máquinas de busca na *Web*, o uso da redução ao radical não é um aspecto importante para aumento da abrangência das consultas, visto ser muito provável de existir informação em abundância sem a execução da lematização dos termos fornecidos na consulta.

Outros problemas são os possíveis conflitos gerados por palavras com conceitos distintos mapeadas para uma mesma palavra básica, pois existem algumas exceções nas regras de formação das palavras, o que dificulta o trabalho dos algoritmos de lematização. Isso poderia ser resolvido por meio de um dicionário que associa todas as derivações possíveis de um mesmo radical a este, porém esse é um trabalho bastante extenso. A WordNet (MILLER, 1993) adota uma abordagem de buscar palavras com proximidade gráfica em relação aos radicais armazenados na sua base, o que também não é um método completamente infalível. Outro aspecto é dificuldade ainda maior de se lematizar colocações em comparação à lematização de termos simples, introduzindo um problema adicional para os algoritmos que visam a realizar esse trabalho. É preciso utilizar essa ferramenta com muita parcimônia e de forma dirigida aos objetivos a serem alcançados.

Os métodos de redução ao radical possuem variações em relação à abordagem utilizada para obtenção do radical e, de maneira geral, se organizam pela taxonomia apresentada na figura 13 (FRAKES, 1992).

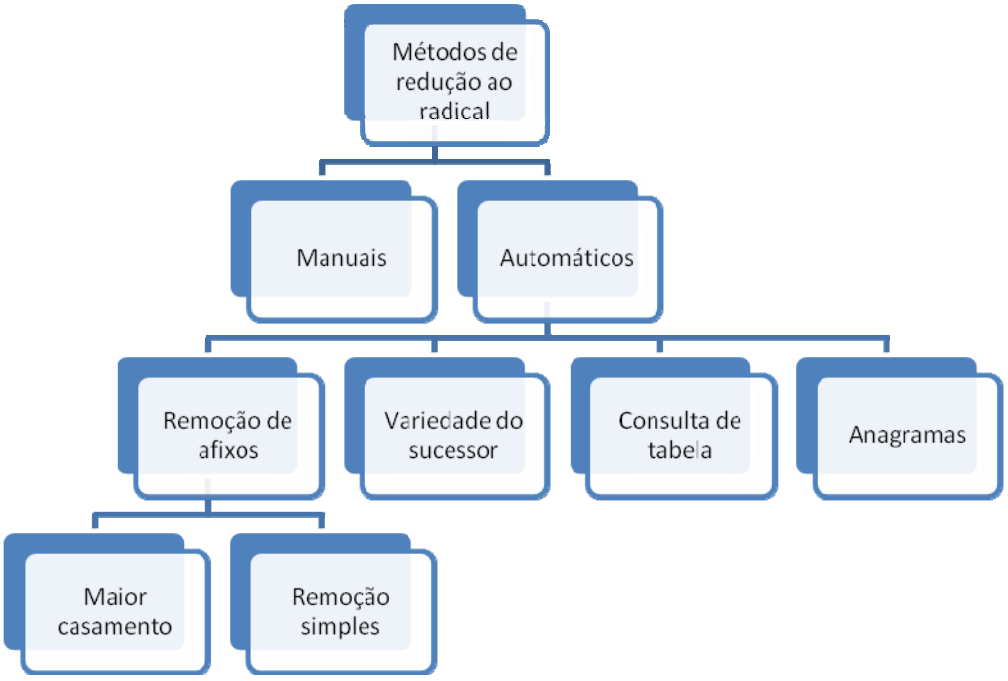

Figura 13 – Taxonomia para métodos de redução ao radical

Os métodos baseados em remoção de afixos se preocupam com a eliminação de prefixos e sufixos com o objetivo de obter somente o radical. Os métodos baseados na variedade do sucessor usam as freqüências de seqüência de letras no texto como base para a radicalização. Os lematizadores por anagramas reduzem os termos com base nos digramas e n-gramas que partilham. Os termos e os seus radicais correspondentes podem ser armazenados em tabelas ou bases de dados e o processo de radicalização ser feito com a utilização dessas tabelas de forma inclusive a compor outros métodos.

O Algoritmo mais conceituado para executar a tarefa de lematização é o Porter Stemming (PORTER, 1980), que está bastante consolidado para língua inglesa. Para aplicá-lo à língua portuguesa, várias modificações devem ser realizadas, pois cada língua possui suas próprias regras de redução de número, gênero e grau, além de outras regras para redução de aumentativos, diminutivos, formas verbais, dentre outras. Além do Porter também existe o Lovins (LOVINS, 1968), uma versão precursora, mas ainda hoje utilizada.

2.2.2.4
Análise de discurso

Análise de discurso consiste em elucidar relacionamentos entre sentenças em um texto (MOSCALEWSKI, 2002). Esse processo possui algumas dificuldades inerentes, sendo a principal a identificação de **anáforas**, que são termos diferentes se referindo à mesma entidade dentro de um texto. A resolução das relações anafóricas é um problema bastante difícil, pois envolve conhecimento semântico sobre os termos envolvidos. Por exemplo, no trecho:

> *Durante uma viagem, D. Pedro recebeu uma nova carta de Portugal que anulava a Assembléia Constituinte e exigia a volta imediata dele para a metrópole. Então, o Príncipe, próximo ao riacho do Ipiranga, levantou a espada e gritou: "Independência ou Morte!". Este fato ocorreu no dia 7 de setembro de 1822 e marcou a Independência do Brasil. No*

mês de dezembro de 1822, D. Pedro foi declarado imperador do Brasil.

existe uma relação anafórica entre D. Pedro e o termo Príncipe. Para essa descoberta o sistema deveria saber que D. Pedro era um Príncipe o que faz disso um problema bastante complexo. O texto ainda faz referência a D. Pedro como sendo imperador do Brasil demonstrando a possibilidade de ocorrência de vários termos diferentes referenciando uma mesma entidade. Esse conhecimento pode ser obtido com auxílio de uma base de dados de dicionário previamente construída ou ainda através de uma rede semântica como a WordNet[17], por exemplo, que será melhor abordada mais adiante.

2.3
Indexação

As técnicas de indexação são amplamente estudadas dentro do ramo de recuperação de informação, que advém da necessidade do homem de localizar e recuperar, de maneira conveniente, informações armazenadas. Para isso, é interessante que o armazenamento dessas informações também seja feito de uma forma conveniente, poupando assim o trabalho de análise de toda a base no momento da busca.

Alguns sistemas de recuperação de informações têm se baseado fortemente no modelo "saco de palavras" para a representação de documentos, negligenciando qualquer conhecimento lingüístico da língua em questão. Isso confere uma abordagem altamente estatística. Outra abordagem que pode ser empregada é a que busca auxílio em algum conhecimento semântico que os textos possam fornecer. Seja qual for a abordagem, a etapa de pré-processamento, utilizando processamento de linguagem natural ou não fornecerá à indexação os termos que alimentarão a estrutura de dados a ser utilizada.

Sendo a busca por informações realizada com base em consultas por palavras-chave é interessante uma estrutura de dados que indexe os termos

[17] Disponível em http://wordnet.princeton.edu/ (inglês) e http://www.instituto-camoes.pt/wordnet/wn.html (português)

existentes nos documentos, fornecidos pelo pré-processamento e ainda que a partir desses termos e através de um procedimento rápido seja possível a recuperação dos documentos associados à consulta. Dessa forma pode-se pensar em um índice remissivo de um livro, onde as palavras referenciam as páginas, dentro do livro, nas quais elas aparecem. A estrutura de dados conhecida como **índice invertido**[18] é hoje a mais utilizada para tal finalidade.

Alguns procedimentos anteriores são indicados para a construção de um índice invertido (MANNING, 2007) e (BAEZA-YATES, 1999). Inicia-se colecionando os documentos a serem indexados e estabelecendo-se as unidades dos documentos, isto é, as peças que compõe os documentos, que normalmente são palavras. Dependendo do tratamento lingüístico, essas peças, conhecidas como *tokens*, podem ser pedaços de palavras, reduzidos ao radical ou até mesmo grupos de mais de uma palavra (colocações). Essas peças são extraídas dos documentos por uma ferramenta conhecida como *tokenizador*. Cada *token* extraído é candidato a uma entrada no índice dependendo de algumas restrições utilizadas na fase de pré-processamento. Por ocasião da identificação dos *tokens* é realizada uma análise léxica dos documentos e durante esse processo são eliminados os caracteres inválidos, filtradas as seqüências de controle (ou de formatação de texto), podendo ainda ser feitas correções ortográficas ou validações dos termos, caso um dicionário seja utilizado. Em princípio, nem todas as palavras podem ser incluídas na estrutura de índice, pois algumas não são significativas, caracterizadas por pouco contribuírem semanticamente para o conteúdo dos textos, mas usadas apenas para fazer o encadeamento de idéias, como as preposições. Além disso, as palavras muito freqüentes não contribuírem para a discriminação de um texto com relação a outros de uma mesma coleção. Essas palavras, chamadas de *stopwords*, devem estar numa lista, a *stoplist*, para não serem indexadas. As *stopwords* aparecem em muitos documentos, e a indexação delas pode comprometer a precisão e a eficiência do sistema de recuperação das informações. Uma seleção criteriosa das *stopwords* pode contribuir de sobremaneira para a redução da dimensão do léxico.

[18] Do inglês *inverted index*.

2.4
Mineração

A etapa de mineração visa à obtenção de algum tipo de conhecimento útil oriundo da coleção de textos. É nessa etapa que são utilizadas as ferramentas para mineração de textos apresentadas (seção 1.3 – Ferramentas para Mineração de Textos) e, dependendo do objetivo específico, as diversas técnicas são empregadas. É importante destacar que com o uso de técnicas de extração de informação os dados deixam de possuir o caráter desestruturado, encontrado nos textos, e passam a ser organizados em estruturas tabulares, podendo ser armazenados em SGBDs (Sistemas de Gerenciamento de Banco de Dados). A partir de então a tarefa de mineração de textos se converte em mineração de dados (GOLDSCHMIDT, 2005). Esse formato de mineração de textos, apresentado na figura 14, é por vezes conhecido como mineração de dados em textos.

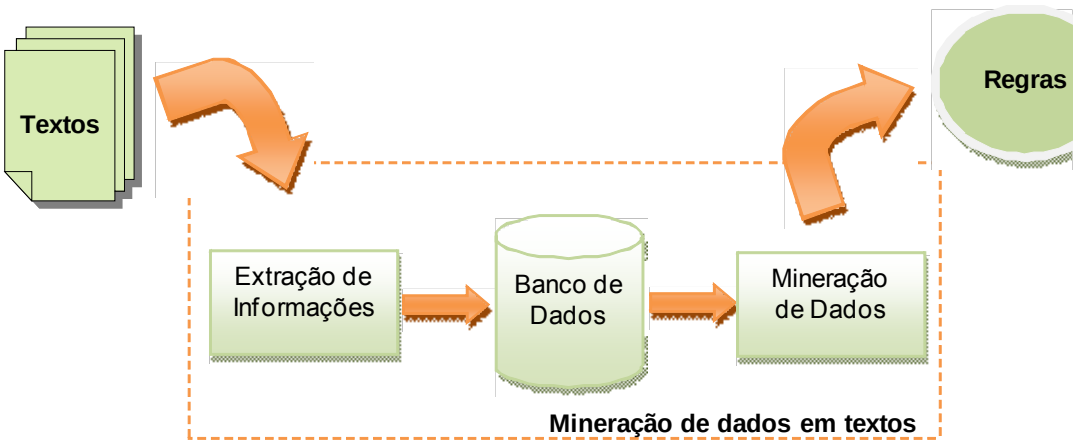

Figura 14 – Processo de mineração de dados em textos.

A estruturação da informação textual com o uso de SGBDs não é a única forma de se fazer mineração, pois outras estruturas de dados, construídas na fase de indexação, possibilitam que algumas das aplicações sejam empregadas. Conforme será apresentado mais adiante técnicas de agrupamento, por exemplo, podem ser utilizadas sobre estruturas mais simples, capazes de modelar os documentos em formatos apropriados para que elas sejam utilizadas.

Cabe ressaltar, com base na própria formulação da solução de mineração de textos através de estruturação dos mesmos com posterior mineração de dados, a dificuldade de se estabelecer claramente o fluxo de etapas isoladas de maneira a

compor o processo de mineração de textos como um todo. No caso específico do exemplo apresentado na figura 14, pode-se questionar se a própria extração de informações já não é uma forma de mineração, e como toda a razão, no caso dela estar sendo feita de forma automática, pois a própria estruturação por si só depende da obtenção de um conhecimento a cerca da estrutura do texto e dos termos que estão sendo utilizados na estruturação.

2.5
Análise

Análise é a etapa em que o ser humano interpreta as informações obtidas pela fase de mineração. Pode tirar proveito de alguma forma de pós-processamento que facilite a apresentação dos dados e visualização dos resultados com a possibilidade de navegação sobre as informações fornecidas. Uma representação visual pode comunicar uma informação muito mais rapidamente e efetivamente que outros métodos. Essa fase é mais bem fundamentada sobre interfaces gráficas amigáveis, ferramentas para geração de relatórios, gráficos e ferramentas configuráveis de consulta. Sistemas de mineração de texto precisam prover os usuários com um grande leque de ferramentas para interação com os dados e interpretação dos resultados. Alguns autores dedicaram capítulos inteiros de seus livros para tratar dos aspectos relacionados à visualização dos resultados, dentre eles, (BAEZA-YATES, 1999) e (FELDMAN, 2007).