Extração de Informação de Artigos Científicos: uma abordagem baseada em indução de regras de etiquetagem

Alberto Cáceres Álvarez

Extração de Informação de Artigos Científicos: uma abordagem baseada em indução de regras de etiquetagem

Alberto Cáceres Álvarez

Orientador: *Prof. Dr. Alneu de Andrade Lopes*

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências de Computação e Matemática Computacional.

USP - São Carlos
Fevereiro/2007

Este documento foi elaborado com o formatador de textos LaTeX, e as referências bibliográficas foram geradas automaticamente pelo BibTeX utilizando o formato padrão Chicago. O estilo de formatação deste trabalho foi elaborado por Ronaldo Cristiano Prati.

Dedicatória

À minha adorável mãe e
meu querido irmão Alain.

Agradecimentos

Agradeço primeiramente a minha família que sempre acreditou em mim e que apoiou em vir para São Carlos fazer o mestrado. Em especial, agradeço a minha amada mãe que eu tenho como um exemplo de vida e carinho, e que sempre me ajudou em todos os momentos, principalmente os difíceis. Mãe obrigado por me ajudar a alcançar mais esse objetivo. Ao meu irmão que sempre foi o modelo de dedicação a DEUS, sendo uma pessoa incrível. Obrigado pelo apoio nos momento finais da escrita.

Agradeço ao professor e meu orientador Alneu de Andrade Lopes, por ser além de mestre, um amigo. Obrigado por no decorrer dessa orientação, me mostrar a luz em situações aonde apenas prevalecia a escuridão.

Aos colegas do LABIC, em especial Rodrigo (calvo), Vinícius, Ronaldo, Augusto, Jean, Brunao, Edson. Obrigado Ronaldo pelas sugestões, ajudaram bastante. Em especial, agradeço ao Vinícius que desde o começo do meu mestrado me ajudou dando sugestões e também, acima de tudo, uma pessoa amiga e bastante divertida. Vinícius do fundo, obrigado.

Às pessoas que tiveram uma participação para a conclusão deste trabalho.

Ao CNPq pelo apoio financeiro para o desenvolvimento deste trabalho.

Resumo

Este trabalho faz parte do projeto de uma ferramenta denominada FIP (Ferramenta Inteligente de Apoio à Pesquisa) para recuperação, organização e mineração de grandes coleções de documentos. No contexto da ferramenta FIP, diversas técnicas de Recuperação de Informação, Mineração de Dados, Visualização de Informações e, em particular, técnicas de Extração de Informações, foco deste trabalho, são usadas. Sistemas de Extração de Informação atuam sobre um conjunto de dados não estruturados e objetivam localizar informações específicas em um documento ou coleção de documentos, extraí-las e estruturá-las com o intuito de facilitar o uso dessas informações. O objetivo específico desenvolvido nesta dissertação é induzir, de forma automática, um conjunto de regras para a extração de informações de artigos científicos. O sistema de extração proposto, inicialmente, analisa e extrai informações presentes no corpo dos artigos (título, autores, afiliação, resumo, palavras chaves) e, posteri ormente, foca na extração das informações de suas referências bibliográficas. A proposta para extração automática das informações das referências é uma abordagem nova, baseada no mapeamento do problema de *part-of-speech tagging* ao problema de extração de informação. Como produto final do processo de extração, tem-se uma base de dados com as informações extraídas e estruturadas no formato XML, disponível à ferramenta FIP ou a qualquer outra aplicação. Os resultados obtidos foram avaliados em termos das métricas precisão, cobertura e *F-measure*, alcançando bons resultados comparados com sistemas similares.

Abstract

This dissertation is part of a project of a tool named FIP (an Intelligent Tool for Research Supporting). FIP is a tool for retrieval, organization, and mining large document collections. In the context of FIP diverse techniques from Information Retrieval, Data Mining, Information Visualization, and particularly Information Extraction, focus of this work, are used. Information Extraction systems deal with unstructured data looking for specific information in a document or document collection, extracting and structuring them in order to facilitate their use. The specific objective presented in this dissertation is automatically to induce a set of rules for information extraction from scientific articles. The proposed extraction system initially analyzes and extracts information from the body of the articles (heading, authors, affiliation, abstract, and keywords) and then extracts information from each reference in its bibliographical references. The proposed approach for information extraction from references is a new technique based on the strategy of part-of-speech tagging. As the outcome of the extraction process, a database with extracted and structured information in XML format is made available for the FIP or any other application. The system has been evaluated using measures of Precision, Recall and F-measure, reaching good results compared to similar systems.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas

AM Aprendizado de Máquina

CRF *Conditional Random Fields*

MUC Conferências de Entendimento de Mensagens

DVHMM *Dual and Variable-length output Hidden Markov Model*

TBL Etiquetador Baseado em Transformação

EI Extração de Informação

FIP Ferramenta Inteligente de Apoio à Pesquisa

HMM *Hidden Markov Model*

KDD *Knowledge Discovery in Databases*

KDT *Knowledge Discovery from Text*

LABIC Laboratório de Inteligência Computacional

MD Mineração de Dados

MT Mineração de Textos

VTM Mineração Visual de Textos

POS *part-of-speech*

PLN Processamento de Língua Natural

ILP Programação Lógica Indutiva

PEx *Projection Explorer*

CBR Raciocínio Baseado em Casos

RI Recuperação de Informação

SVM *Support Vector Machines*

Capítulo 1

Introdução

Um dos problemas com que iniciantes, e mesmo pesquisadores mais experientes, se deparam atualmente é a quantidade de informações disponíveis, sobre qualquer área de conhecimento, principalmente na rede mundial de computadores. Cresce cada vez mais o número de artigos científicos, disponibilizados em páginas pessoais ou institucionais, tornando a busca e seleção de artigos fundamentais uma tarefa não trivial. As principais ferramentas de busca disponíveis selecionam um enorme conjunto de páginas, sendo que a maioria delas apresentam informações irrelevantes ao tema de interesse do pesquisador. Isso ocorre porque coexistem informações relevantes e uma considerável quantidade de informações irrelevantes. Portanto, a recuperação (busca e seleção) de informação relevante passa pelo conhecimento que o pesquisador tem daquele campo de conhecimento e é, conseqüentemente, difícil para iniciantes em uma área de pesquisa. Essa dificuldade em encontrar informações relevantes sobre um determinado assunto e, principalmente, as relações entre os documentos recuperados decorre das limitações da grande maioria das páginas (uso de código HTML sem informações semânticas sobre o conteúdo da página), e da própria diversidade e quantidade de material disponível; projetos de Web semântica (Berners-Lee, Hendler & Lassila, 2001) têm sido desenvolvidos com o objetivo de facilitar essa busca.

Uma forma de auxiliar um pesquisador a enfrentar o problema acima mencionado é disponibilizar ferramentas que o ajudem nessa mineração da Web por informações relevantes sobre a área ou tema de seu interesse. Neste contexto, é de significativo interesse uma ferramenta que, automaticamente, recupere a partir da Web artigos e publicações sobre uma de-

terminada área de pesquisa; estrutura um repositório com esses artigos; e organize uma representação gráfica dos principais conhecimentos na vizinhança desse tema, indicando de acordo com o objetivo do usuário quais são as publicações mais importantes, quais os temas relacionados, quais são as referências mais citadas, entre outras informações. Tal ferramenta pode apresentar um "mapa do terreno" a ser explorado pelo pesquisador. Essa ferramenta foi denominada FIP (Ferramenta Inteligente de Apoio à Pesquisa) e está em desenvolvimento neste programa. Uma descrição mais detalhada sobre a FIP é apresentada na próxima seção, bem como a relação deste trabalho com a FIP.

Para construir tal representação gráfica, entre diversas tarefas, existe a necessidade de extrair de forma automática informações dos artigos científicos. A FIP utiliza as informações extraídas para determinar, por exemplo, quais são os artigos científicos considerados fundamentais em uma determinada área de pesquisa.

A área de Extração de Informação (EI) objetiva localizar informações relevantes em um documento ou coleção de documentos expressos em língua natural, extraí-las e estruturá-las, por exemplo, em banco de dados, a fim de facilitar sua manipulação e análise. A informação extraída é determinada por um conjunto de padrões ou regras de extração específicas ao seu domínio. Técnicas de Aprendizado de Máquina são utilizadas para automatizar o processo de extração de informações de documentos.

A extração de informações a partir de artigos científicos depara-se com diversos desafios relacionados com a identificação, no texto, de cada uma das seguintes informações: título, autores, afiliação, resumo, palavras chaves e referências bibliográficas. No contexto da FIP, é interessante extrair também as informações contidas nas referências bibliográficas, pois tais informações, além de geralmente fornecer novos dados a respeito dos artigos, podem auxiliar a identificar possíveis relações entre artigos científicos. Técnicas de Processamento de Língua Natural podem ser utilizadas para extrair informações relevantes de documentos com pequeno ou nenhum grau de estruturação. O objetivo do uso dessas técnicas no contexto de EI é tentar compreender textos em alguma língua natural, a fim de encontrar informações relevantes a serem extraídas.

Na próxima seção é apresentado o contexto (ferramenta FIP) e o escopo deste trabalho.

1.1 Motivação e Escopo do Trabalho – A FIP

O projeto de uma Ferramenta Inteligente de Apoio à Pesquisa (FIP), em andamento no Laboratório de Inteligência Computacional (LABIC[1]) do ICMC-USP, tem como objetivo disponibilizar uma ferramenta que auxilie um estudante ou pesquisador iniciante em uma área a explorar uma coleção de documentos via uma representação gráfica que expresse características relevantes dos artigos e suas relações de similaridade. Essa representação deve permitir a visualização das principais áreas, sub-áreas, artigos e autores principais, além de outros aspectos importantes relacionados com o tema/tópico de pesquisa em questão.

Quando um usuário da FIP executar uma consulta, a ferramenta recuperará artigos relevantes de sua base de dados, realizará uma análise de similaridade, estabelecendo os agrupamentos (*clusters*) mais importantes, e construirá um mapa que represente graficamente uma visão geral do tema consultado, baseada nessa análise. A base de dados da FIP é atualizada a partir de um conjunto de artigos recuperados, automaticamente, da Web sobre áreas pré-definidas, sendo que o processo de extração de informações representa uma parte essencial no desenvolvimento dessa ferramenta.

A ferramenta auxiliará na pesquisa em áreas de conhecimento pré-determinadas, sendo projetada para indicar ao pesquisador as principais publicações e suas relações de similaridade, e os temas relacionados à área de consulta. Tal ferramenta pode apresentar um mapa, relacionado com a área de interesse, a ser explorado pelo pesquisador.

Um possível mapa a ser construído pela FIP é ilustrado na Figura 1.1. Esse mapa foi obtido a partir de experimentos com corpus de Raciocínio Baseado em Casos, Recuperação de Informação e Programação Lógica Indutiva, compreendendo um total de 600 artigos científicos, sendo que tais artigos são mapeados para uma superfície onde os relacionamentos podem ser observados e explorados minuciosamente conforme sua vizinhança (Lopes et al., 2006). Nessa representação gráfica é possível visualizar um artigo (ILP-1446Dze281-290.txt), pertencente a área de Programação Lógica Indutiva, mapeado em uma determinada região no mapa. A representação visual dos corpus envolve a determinação da similaridade entre documentos, levando em conta o conteúdo dos documentos. Nesse mapa os montes indicam os trabalhos relacionados ao tema de interesse, os vales represen-

[1] http://labic.icmc.usp.br/

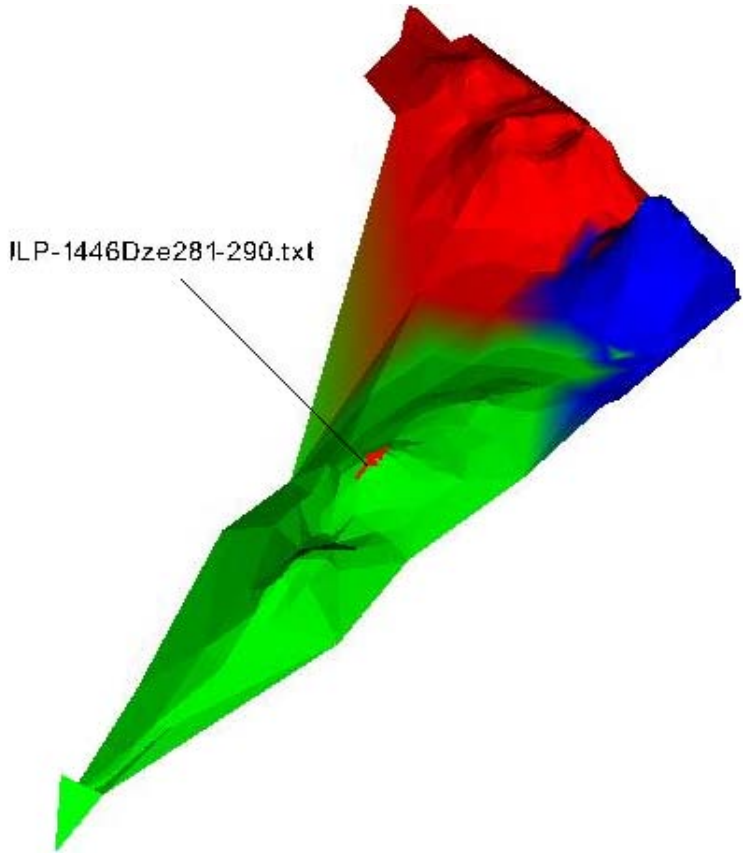


Figura 1.1. Mapa representando corpus de Raciocínio Baseado em Casos, Recuperação de Informação e Programação Lógica Indutiva

tam as fronteiras entre as sub-áreas e em cada cume estão concentrados os artigos mais similares entre si. Para a construção do mapa, utilizaram-se as informações dos artigos e publicações da área recuperada, tais como palavras chaves, resumo, título, autores, conjunto de referências citadas, data, publicação, etc. É importante ressaltar que outras características podem ser mapeadas, por exemplo, os cumes podem denotar publicações fundamentais. Uma forma de detectar as publicações fundamentais pode ser em função da quantidade de vezes que estas são referenciadas em outros artigos e a importância dos congressos, revistas ou periódicos nas quais foram publicadas.

Uma vantagem da representação gráfica é a facilidade de visualização dos pesos e relações entre as diversas abordagens e sub-áreas, além da visualização das fronteiras com outras áreas de pesquisa e das hierarquias existentes. Naturalmente, outras representações, que permitem distintas interpretações do mapa, já foram avaliadas em experimentos realizados

conjuntamente pelos pesquisadores[2] do ICMC-USP, especialistas das áreas relacionadas (Lopes et al., 2007). Alguns avanços relacionados com a mineração visual de coleções de documentos (artigos) são apresentados na Seção 1.2.

A FIP será composta por três módulos fundamentais relacionados com a recuperação de artigos científicos na Web, com a análise dos artigos (extração de informações e determinação da similaridade entre os artigos) e com a visualização (construção da representação visual da coleção de documentos recuperados) dos artigos recuperados - Figura 1.2.

Módulo 1: módulo responsável pela recuperação de artigos científicos disponíveis na Web por meio de técnicas de mineração na Web e mineração de texto. Dos artigos recuperados são extraídas informações como: autores, título, afiliação, resumo, palavras chave e conjunto de referências bibliográficas. Estas informações são armazenadas em um banco de dados ou em um *data warehouse*.

Módulo 2: módulo que trata da avaliação e análise da similaridade entre os artigos recuperados, além de encontrar relações de precedência, importância relativa, áreas e sub-áreas relacionadas com os documentos, por meio de técnicas como *clustering*, regras de associação e outras técnicas de Aprendizado de Máquina (AM) e de Mineração de Texto.

Módulo 3: módulo que realiza a representação gráfica do resultado do módulo 2, ou seja, a criação de um mapa que represente os níveis de similaridade entre os artigos científicos e permite a interação com o usuário. Pesquisas relacionadas com esse módulo têm sido realizadas em colaboração com o grupo de Computação Gráfica e Processamento de Imagens.

Na Seção 1.2 os trabalhos já realizados e em andamento relacionados com esses módulos são detalhados.

Sistemas semelhantes que recuperam e armazenam publicações disponíveis na Web têm sido construídos em instituições de pesquisa de outros países, tais como ISI Web of Knowledge[3], Citeseer[4] e MathSciNet[5]. O Cite-

[2] Alneu de Andrade Lopes, Rosane Minghim, Vinícius Veloso de Melo, Fernando Vieira Paulovich e Roberto Pinho.

[3] http://www.isinet.com/isi/

[4] http://www.citeseer.com/

[5] http://www.ams.org/mathscinet/

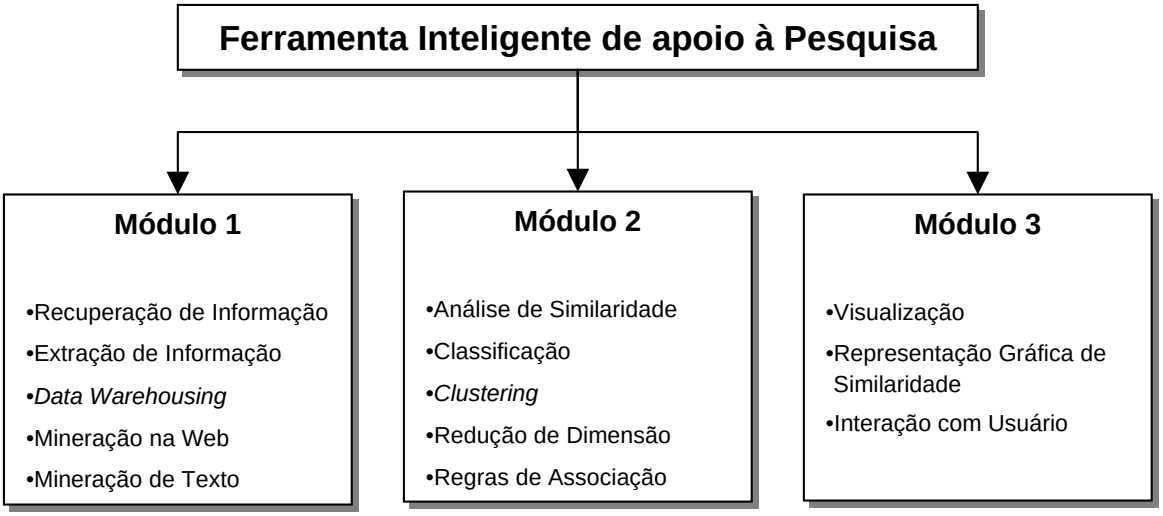


Figura 1.2. Módulos e técnicas utilizadas na FIP

seer, por exemplo, mantém uma coleção de documentos científicos da área de computação e ciência da informação (formatos PDF e PS) recuperados da Web e possibilita buscar documentos (por título, área, autor, etc.) ou citações em documentos (Lawrence, Bollacker & Giles, 1999). No entanto, esses sistemas não disponibilizam uma representação gráfica dos níveis de similaridade entre artigos científicos, da importância relativa de um documento e de áreas e sub-áreas relacionadas.

Algumas ferramentas de busca têm sido desenvolvidas visando a construção de um resultado gráfico a partir das páginas Web recuperadas por meio de uma consulta do usuário, como por exemplo: Kartoo[6] e Grokker[7]. A diferença dessas ferramentas de busca e a FIP encontra-se na análise da relevância de artigos científicos recuperados sobre um determinado tema, a qual será realizada pela segunda ferramenta citada (FIP) enquanto as outras limitam-se a analisar o conteúdo das páginas resultantes da consulta.

Na Figura 1.3 é representada a seqüência de procedimentos e técnicas utilizadas pela ferramenta FIP, o contexto deste projeto, e o foco do trabalho, realçado na figura pelo retângulo tracejado. A seguir são descritos os principais procedimentos e técnicas.

1. **Recuperação de Informação:** o sistema *WebMiner* (Brasil, 2006) é o responsável pela recuperação dos artigos da Web e atualizará de forma automática e periódica o conteúdo dos corpora de áreas previa-

[6] http://www.kartoo.com
[7] http://www.groxis.com/service/grok

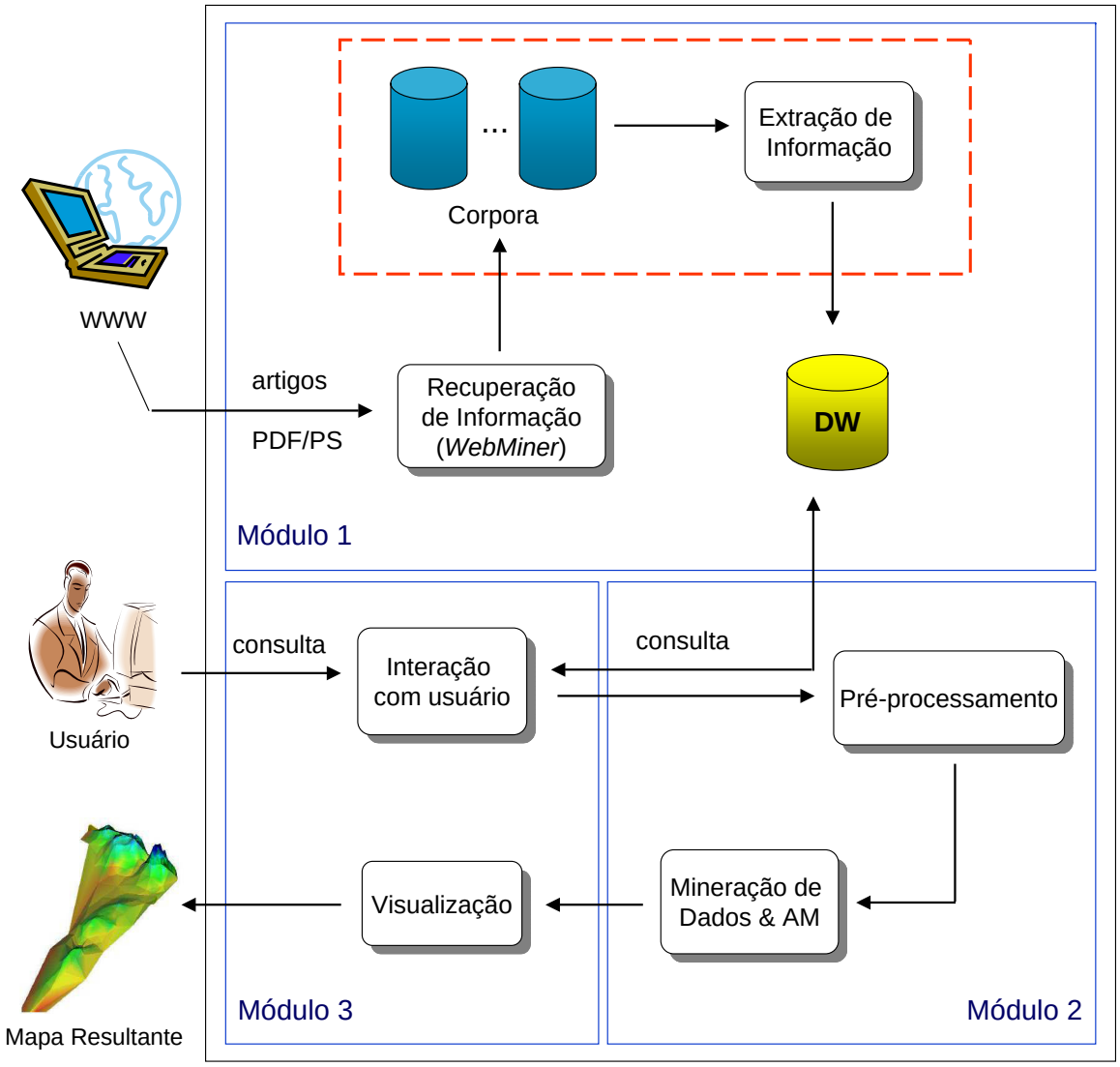


Figura 1.3. Módulos principais da Ferramenta Inteligente de Apoio à Pesquisa e o foco deste trabalho

mente selecionadas. Inicialmente, a base da FIP será composta apenas por artigos de algumas áreas de Inteligência Artificial pré-selecionadas, mas futuramente abrangerá outras áreas da Ciência da Computação. O processo é automatizado por técnicas de recuperação de informação (para recuperar os artigos PDF e PS) e mineração de texto (para determinar a relevância do artigo recuperado de acordo com a área escolhida). São armazenados os documentos originais e os documentos convertidos para o formato TXT.

2. **Extração de Informação:** através de abordagens baseadas em indução de regras de extração, informações como título, autores, resumo, referências, palavras-chave, serão extraídas dos artigos recuperados e disponibilizadas para serem armazenadas em um banco de dados ou em um *data warehouse*.

   Este trabalho está relacionado com o primeiro módulo da ferramenta e compreende dois objetivos: o primeiro consiste na indução de um conjunto de regras para a extração de informações presentes no corpo de artigos científicos; o segundo foca na extração das informações contidas na referências bibliográficas, e consiste na indução automática de um conjunto de regras de extração.

3. **Pré-processamento:** preparação dos textos e conversão para um formato vetorial após eliminação de palavras irrelevantes (*stopwords*), extração de radicais (*stemming*) e outros procedimentos, de tal forma que possam ser aplicados os algoritmos de Aprendizado de Máquina e Mineração de Dados.

4. **Mineração de Dados & AM:** técnicas de Mineração de Dados (MD) e Aprendizado de Máquina (AM) serão utilizadas a fim de encontrar relações de similaridade entre os artigos recuperados e organizá-los de maneira coerente.

5. **Visualização e Interação com o usuário:** técnicas de Mineração Visual de Textos serão empregadas para a criação e exploração de um mapa que represente graficamente os artigos relevantes, conforme o objetivo do usuário. No mapa, haverá *links* com os títulos dos artigos, sendo possível ao usuário obter informações do artigo referido a partir de cada *link*.

1.2 Trabalhos Realizados e em Andamento

Recentemente, no contexto da ferramenta FIP e em parceria com o grupo de Computação Gráfica e Processamento de Imagens, vários avanços relacionados com a pesquisa em mineração visual de coleções de documentos foram obtidos (Lopes et al., 2007; Lopes et al., 2006; Minghim, Paulovich & Lopes, 2006).

Lopes et al. (2007) propuseram um conjunto de ferramentas para a Mineração Visual de Textos (VTM) que integram técnicas de visualização e exploração multidimensionais com técnicas de mineração de textos (em particular regras de associação), para a construção de uma representação gráfica que possibilite ao usuário, compreender a estrutura geral representada em uma particular coleção de documentos, tal como examinar e identificar os principais tópicos/temas abordados em partes dessa coleção.

Com a finalidade de fornecer suporte ao processo de criação e exploração de mapas de documentos, foi desenvolvida uma ferramenta, denominada *Projection Explorer* (PEx) (Paulovich & Minghim, 2006). A PEx teve seu desenvolvimento inicial no trabalho de doutorado de Fernando Paulovich, sob orientação de Rosane Minghim, como um conjunto de técnicas de exploração visual de informações. Atualmente, com a colaboração do grupo da FIP, técnicas específicas para exploração visual de coleções de documentos foram desenvolvidas. A janela principal da ferramenta (aplicativo) PEx é apresentada na Figura 1.4.

As principais características da PEx para explorar mapas de documentos incluem (Lopes et al., 2007):

1. a possibilidade de encontrar os vizinhos mais próximos de um documento;
2. coloração de documentos (pontos) no mapa conforme a freqüência de uma palavra ou grupos de palavras;
3. criação de rótulos para identificar um grupo de documentos no mapa. Esses rótulos são baseados nas freqüências dos termos e, também, em regras de associação;
4. visualização do conteúdo de um documento representado no mapa;
5. coloração de um grupo de documentos considerando a distância dos mesmos a um documento pré-selecionado;

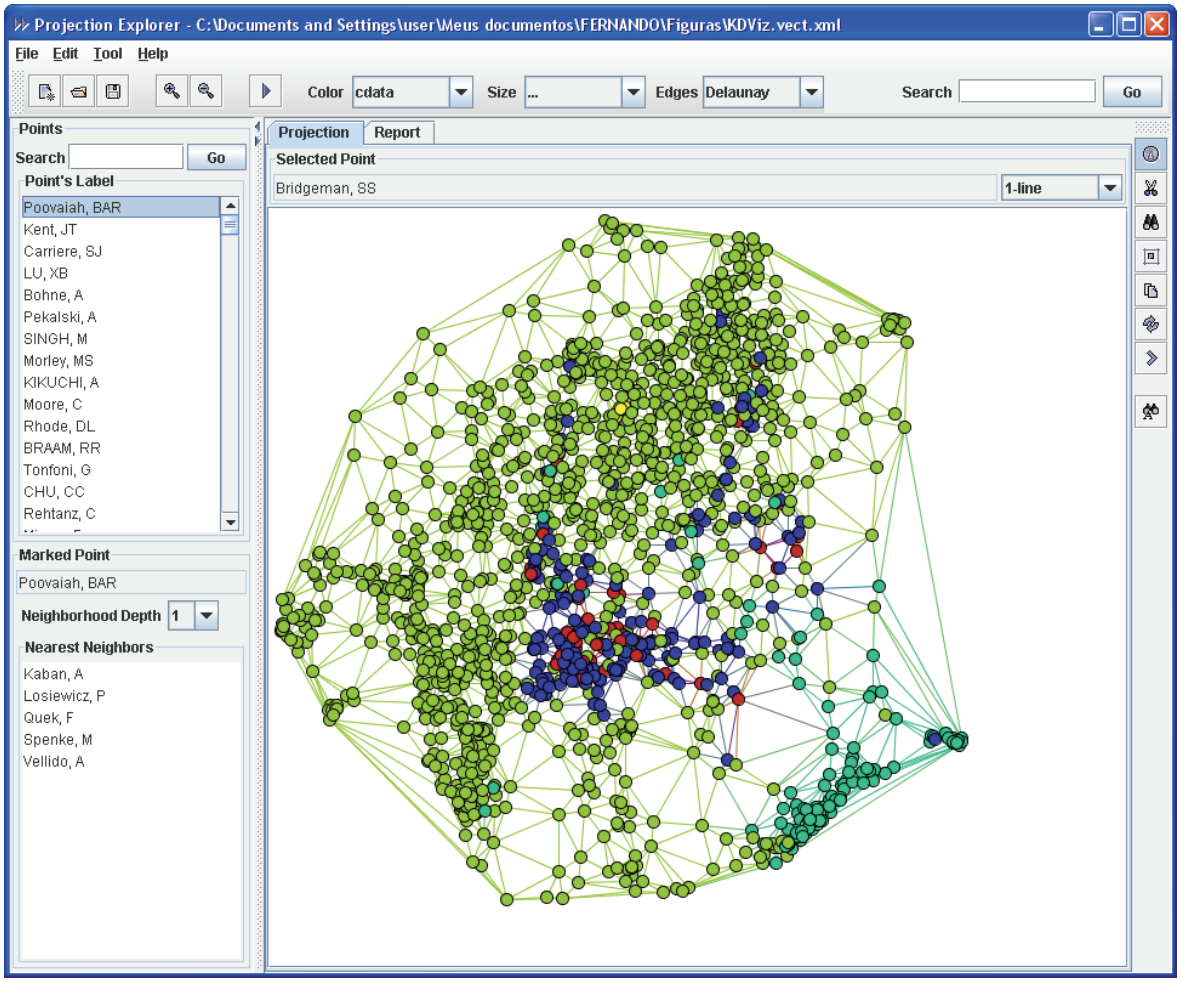


Figura 1.4. Janela principal da ferramenta *Projection Explorer* (PEx)

6. possibilidade de utilizar, além de uma coleção de documentos em formato ASCII, o resultado de um consulta na Web como entrada para a ferramenta.

A seguir é demonstrada uma exploração de um corpus com 574 documentos nas áreas de Programação Lógica Indutiva (ILP), Raciocínio Baseado em Casos (CBR) e Recuperação de Informação (RI).

Para fazer uso de várias funcionalidades implementadas na PEx, o usuário deve selecionar uma área de interesse no mapa.

Lopes et al. (2007) introduziram uma técnica (presente na PEx) para a detecção de tópicos ou temas através da geração e seleção (filtragem) adequada de regras de associação, as quais são baseadas no conteúdo de um grupo de documentos – pontos – em uma área pré-selecionada no mapa.

O mapa ilustrado na Figura 1.5, além de agrupar os documentos similares, apresenta um conjunto de regras de associação geradas para descrever

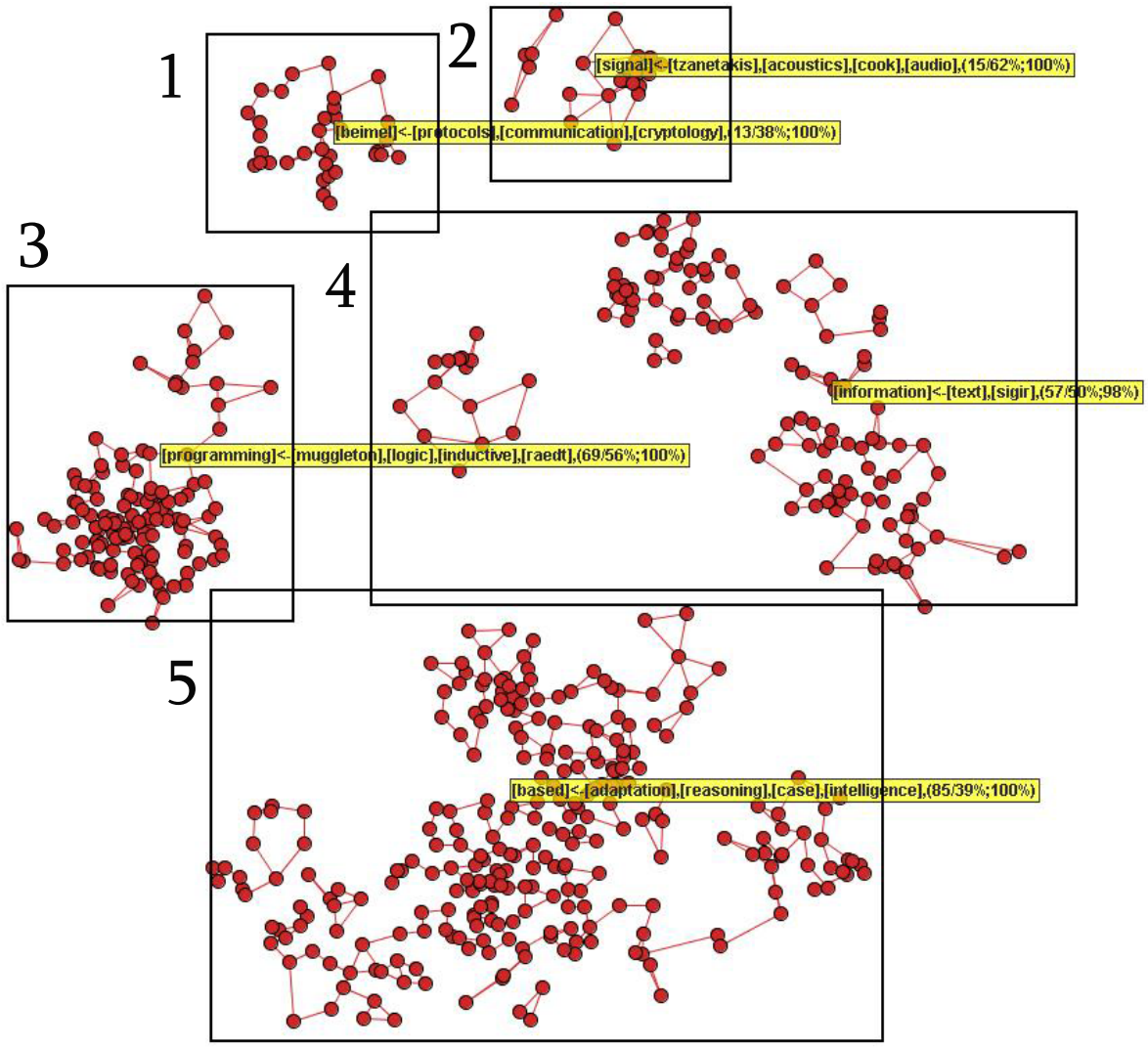


Figura 1.5. Resumo do mapa de CBR, ILP e RI com regras de associação

as (5) áreas selecionadas no mapa, determinadas visualmente (ou automaticamente). A regra de associação para a região 1 indica documentos relacionados com comunicação, protocolos e criptologia, com 100% de confiança, onde o termo "beimel" (um autor da área) aparece. As regras para as regiões 2 e 4 são relacionadas a área de RI. Para a região 3, a regra indica documentos da área de ILP. Finalmente, a regra para a região 5 relaciona documentos sobre CBR e Processo de Adaptação em CBR.

Uma vez detectadas as grandes (principais) áreas de pesquisa, o usuário pode desejar aprofundar o seu conhecimento sobre uma determinada área. A ferramenta PEx possibilita que o usuário selecione e explore uma área previamente detectada, com o objetivo de identificar as principais subáreas, os temas centrais abordados e os relacionamentos existentes.

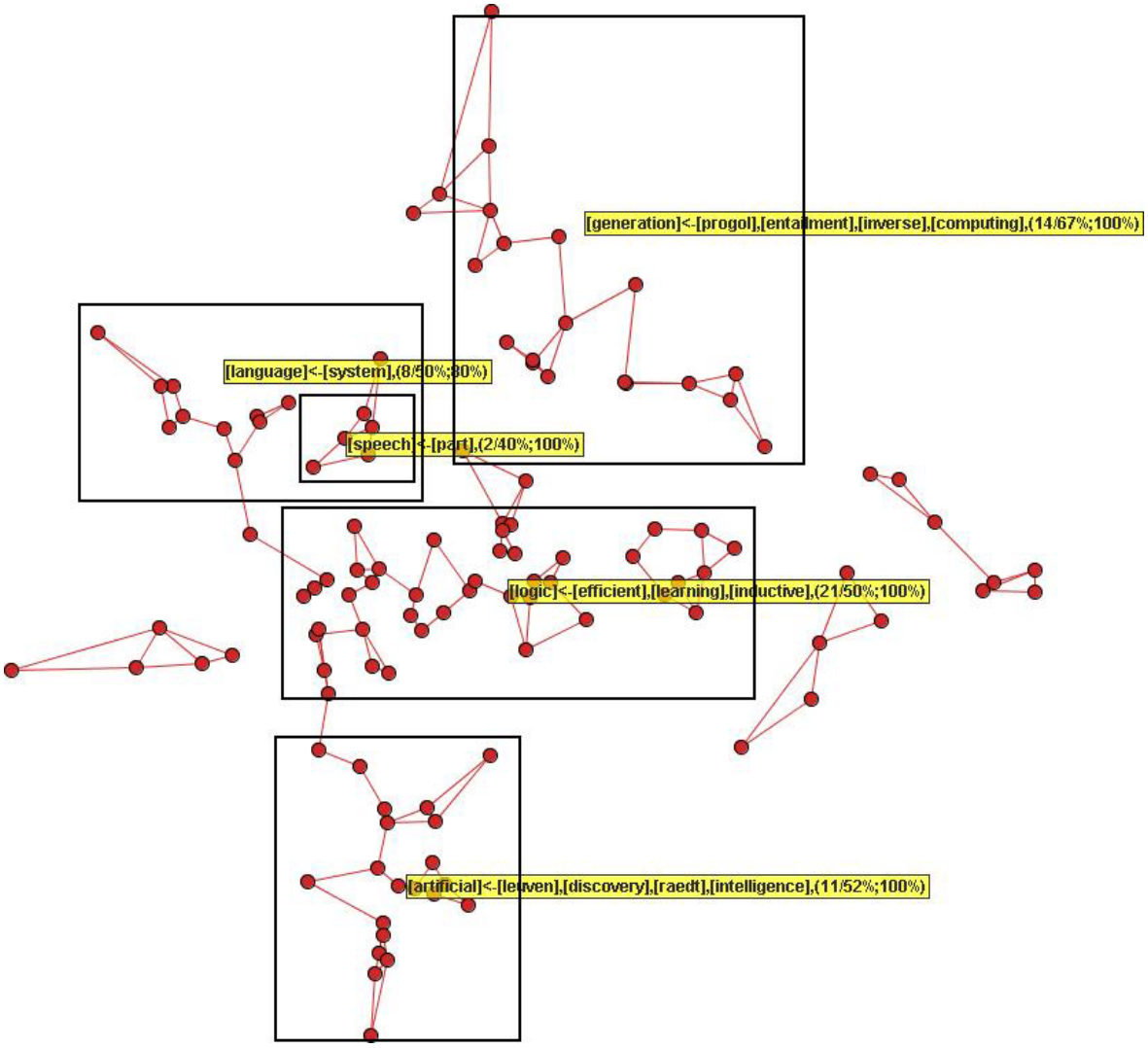


Figura 1.6. Uma visão aprofundada da área de ILP

O mapa apresentado na Figura 1.6 foi obtido como resultado da exploração da área de Programação Lógica Indutiva (região 3) do mapa anterior. Nesse novo mapa, os documentos, no caso artigos, do grupo representado pela regra [`language`] <– [`system`] estão relacionados com *inductive learning and language*, e especialmente com *part-of-speech* (POS) *tagging* usando abordagens de ILP ([`speech`] <– [`part`]).

No desenvolvimento da FIP, uma aluna de mestrado trabalhou na construção de um sistema de recuperação automática de artigos científicos da Web, relacionados com uma área pré-definida pelo usuário, denominado *WebMiner* (Brasil, 2006). Para recuperar artigos de uma área pré-definida, o sistema deve possuir algum conhecimento capaz de identificar termos e padrões particulares dessa área. Dessa forma é interessante para o *WebMiner*, e conseqüentemente para a FIP, classificar de forma automática artigos

em uma determinada área. Utilizando um conjunto relativamente pequeno de treino e um corpora abrangendo três áreas de pesquisa, a aluna obteve classificadores com alta taxa de precisão quando o número de classes diferentes na coleção é pequeno, isto é, com artigos pertencentes a poucas áreas ou sub-áreas (Brasil & Lopes, 2004). Um novo aluno de mestrado está se dedicando ao desenvolvimento de um *crawler* para a recuperação dos artigos.

Um aluno de mestrado investigou técnicas de *clustering* (Melo, 2005) para realizar o agrupamento de artigos. Neste trabalho foram avaliadas técnicas de *clustering* de documentos. No caso específico de artigos científicos, as referências bibliográficas dos artigos podem fornecer informações úteis para o cálculo da similaridade, pois artigos semelhantes ou que abordem temas semelhantes, normalmente, citam algumas referências em comum. O aluno utilizando como informação adicional para o *clustering* de artigos científicos as referências bibliográficas, obteve uma melhora significativa nos *clusters* construídos (Melo & Lopes, 2004b). Para utilizar as referências bibliográficas no *clustering*, o aluno realizou manualmente a extração, utilizando uma técnica baseada em regras, e a estruturação de informações das referências bibliográficas, capaz de identificar com uma chave única as referências similares (Melo & Lopes, 2005; Melo & Lopes, 2004a; Melo, Secato & Lopes, 2003).

Um outro aluno de mestrado está trabalhando na modelagem de um banco de dados dimensional (*data warehouse*) para armazenar dados de artigos científicos de áreas pré-selecionadas (Kanashiro, 2005). Outros alunos de mestrado e doutorado do LABIC também estão realizando experimentos com os corpora já produzidos, usando outras técnicas de Aprendizado de Máquina e Mineração de Dados.

1.3 Organização do Trabalho

Os próximos capítulos deste trabalho estão organizados da seguinte forma.

No Capítulo 2 são introduzidos alguns conceitos relacionados ao processo de mineração de textos. Como a técnica de extração de informação, proposta neste trabalho utiliza-se da etiquetagem prévia do texto, a definição do problema de *part-of-speech* (POS) *tagging*, assim como o funciona-

mento do etiquetador desenvolvido por Eric Brill são também apresentados.

No Capítulo 3 é apresentada uma introdução a área de Extração de Informação descrevendo os principais conceitos e técnicas utilizadas. São descritas também algumas métricas de avaliação utilizadas pelos sistemas de Extração de Informação.

No Capítulo 4 são descritos os procedimentos empregados neste trabalho para a identificação e extração das informações contidas no corpo e nas referências bibliográficas de artigos científicos. É comentado também a construção de um corpus de referências etiquetado que foi usado nos experimentos.

No Capítulo 5 são apresentados os resultados obtidos e a análise dos experimentos realizados com o objetivo de avaliar o sistema de extração de informação proposto.

Finalmente, no Capítulo 6 são apresentadas a conclusões e os trabalhos futuros.

Capítulo 2

Mineração de Textos

Mineração de textos, de maneira análoga à mineração de dados, 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 são apresentados os principais conceitos relacionados ao processo de mineração de textos, algumas áreas relacionadas e as etapas que compõem o processo. Como a abordagem proposta neste trabalho para a extração de informações de documentos textuais vale-se da etiquetagem prévia do texto, neste capítulo também é descrito o problema de *part-of-speech* (POS) *tagging*, bem como o funcionamento e aprendizado do etiquetador adotado nos experimentos, denominado TBL.

2.1 Descoberta de Conhecimento

Na literatura podem ser encontradas diversas definições para o processo de extração de conhecimento em base de dados, também conhecida como *Knowledge Discovery in Databases* (KDD) ou Mineração de Dados. Fayyad, Piatetsky-Shapiro & Smyth (1996a) definem KDD como o processo não trivial de identificação de padrões válidos, novos, potencialmente úteis e compreensíveis que estejam presentes nos dados. Nesta definição, *dados* são um conjunto de fatos, os quais podem ser, por exemplo, dados estruturados em uma base de dados, ou também dados não estruturados,

como documentos textuais; *validade* determina que os padrões descobertos devem possuir algum grau de certeza; os padrões devem conter *novas* informações sobre os dados; e, finalmente, um dos principais objetivos é que os padrões sejam *úteis* e *compreensíveis* a humanos para que o conhecimento final adquirido possa ser utilizado, por exemplo, em um processo de tomada de decisão (Rezende et al., 2003; Fayyad, Piatetsky-Shapiro & Smyth, 1996a).

Knowledge Discovery in Databases (KDD) é um processo iterativo e interativo, segmentado em algumas etapas que devem ser aplicadas ao conjunto de dados de interesse com o objetivo de extrair padrões úteis. Inicialmente, foi proposta uma divisão do processo em nove etapas (Fayyad, Piatetsky-Shapiro & Smyth, 1996b). Posteriormente, esse número foi reduzido para quatro (Weiss & Indurkhya, 1998). No entanto, considera-se, atualmente, uma abordagem que divide o processo num ciclo que pode ser repetido várias vezes de pré-processamento, extração de padrões e pós-processamento, antecedido por uma etapa de identificação do problema e que precede a etapa de utilização do conhecimento (Rezende et al., 2003). Esse processo é ilustrado na Figura 2.1, e suas etapas descritas, resumidamente, a seguir.

Figura 2.1. Etapas do processo de KDD (Rezende et al., 2003)

Identificação do problema A etapa de identificação do problema é responsável pela análise e compreensão do conhecimento do domínio da aplicação, bem como pela definição dos principais objetivos e metas a serem alcançados no processo de KDD.

Pré-processamento Na etapa de pré-processamento os dados, normalmente dispostos em formato inadequado, são preparados para que possam ser utilizados pelos algoritmos de extração de padrões. Diversas tarefas podem ser executadas na etapa de pré-processamento, tais como integração, transformação, limpeza e redução de dados.

- INTEGRAÇÃO: a integração tem por objetivo realizar a unificação dos dados, os quais podem ser provenientes de diferentes fontes, resultando em uma única fonte de dados, em geral no formato atributo-valor. Os dados de interesse podem ter origem, por exemplo, em arquivos texto, planilhas, banco de dados ou *data warehouse*.
- TRANSFORMAÇÃO: com a finalidade de superar quaisquer limitações presentes nos algoritmos de extração de padrões, pode ser necessário modificar a representação dos dados. Algumas transformações que podem ser aplicadas são: normalização, transformação de tipo, discretização de atributos quantitativos e conversão de atributos qualitativos em quantitativos (Batista, 2003).
- LIMPEZA: os dados coletados podem apresentar diversos problemas, entre eles atributos com valores desconhecidos, incorretos, inválidos ou imprecisos, sendo, dessa forma, imprescindível a realização de uma limpeza nos dados. Pode-se empregar o conhecimento relevante do domínio para a identificação e remoção desses erros.
- REDUÇÃO: em função de limitações de memória ou tempo de processamento pode ser necessário aplicar métodos para a redução dos dados. Pode-se reduzir o volume de dados por meio da redução do número de exemplos, atributos ou valores de um atributo (Weiss & Indurkhya, 1998).

Extração de Padrões A etapa de extração de padrões consiste na definição, configuração e execução de um ou mais algoritmos para a descoberta de padrões nos dados. Tarefas de mineração, tais como classificação,

clustering e regras de associação, estão relacionadas com a representação do modelo gerado, e, portanto, a escolha da tarefa a ser empregada deve ser realizada conforme os objetivos a serem alcançados (Rezende et al., 2003). Nesta etapa, pode-se utilizar algoritmos provenientes de diversas áreas de conhecimento, tais como Aprendizado de Máquina, Estatística, Redes Neurais e Banco de Dados.

Pós-processamento Na etapa de pós-processamento os padrões extraídos são avaliados e interpretados, constatando se o conhecimento obtido, de fato, atingiu as espectativas. Entretanto, caso o conhecimento obtido não seja de interesse do usuário final ou não atenda aos objetivos pré-estabelecidos, pode-se executar novamente algumas etapas ou todo o processo de KDD. Interessabilidade, compreensibilidade, precisão, taxa de erro e cobertura são algumas possíveis métricas utilizadas para avaliar a qualidade dos padrões (Monard & Baranauskas, 2003; Silberschatz & Tuzhilin, 1995).

Utilização do conhecimento Ao final, o conhecimento extraído pode ser integrado a um Sistema Inteligente, ou ainda disponibilizado ao usuário para ser utilizado em um processo decisório.

Na literatura, o termo Mineração de Dados (MD) normalmente é utilizado como sinônimo do processo de KDD, ou seja, MD representa todas as etapas do processo de descoberta de conhecimento. Entretanto, alguns autores consideram os termos KDD e MD referentes a processos distintos. Fayyad, Piatetsky-Shapiro & Smyth (1996a) descrevem MD como sendo uma parte do processo de KDD, mais especificamente, a aplicação de algoritmos de extração de padrões nos dados. Neste trabalho, os termos MD e KDD são tratados indistintamente referenciando o processo de extrair conhecimento a partir de dados.

Entretanto, quando os dados disponíveis ao processo de mineração estão em formato não estruturado, como textos ou documentos, o processo é denominado de Mineração de Textos, Mineração de Dados Textuais ou *Knowledge Discovery from Text* (KDT) (Ebecken, Lopes & Costa, 2003; Hearst, 1999; Feldman & Hirsh, 1997).

2.2 Uma Visão Geral de Mineração de Textos

Mineração de Textos (MT) refere-se ao processo não trivial de extração de padrões úteis e interessantes (conhecimento) a partir de um conjunto de documentos textuais não estruturados (Feldman & Hirsh, 1997). 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 mineração de dados, é que o conhecimento extraído seja compreensível a humanos. Mineração de textos utiliza técnicas das áreas de Extração de Informação, Processamento de Língua Natural (PLN) e Recuperação de Informação, juntamente com algoritmos e métodos de KDD, Aprendizado de Máquina e estatística.

Apesar de similar à MD, MT é, no entanto, um processo mais complexo, pois trabalha com dados textuais que são inerentemente não estruturados e que, eventualmente, possuem ambigüidade (Tan, 1999). Dörre, Gerstl & Seiffert (1999) consideram como o primeiro desafio da mineração de textos a manipulação e análise de informações textuais expressas em língua natural, que, inicialmente, não foram projetas para serem processadas por computadores, mas para serem interpretadas por humanos.

Pesquisas recentes na área de Mineração de Textos 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. Portanto, geralmente é necessária uma adaptação dos algoritmos de mineração de dados para tratar dados textuais (Hotho, Nürnberger & Paaß, 2005).

Mineração de textos freqüentemente vale-se de experiências e resultados de pesquisas em recuperação de informação e processamento de língua natural, as quais são descritas, sucintamente, a seguir.

Recuperação de Informação

Define-se Recuperação de Informação (RI) como o processo de busca de documentos relevantes em resposta a uma consulta, e que estes satisfaçam à necessidade de informação do usuário (Baeza-Yates & Ribeiro-Neto, 1999). Métodos estatísticos, por exemplo para o cálculo da similaridade entre dois documentos, podem ser usados para processamento automático

dos documentos textuais e comparação com a consulta dada (Weiss et al., 2005b).

A área de Recuperação de Informação, embora não seja relativamente recente, os primeiros trabalhos sobre indexação automática ocorreram em 1975, ganhou maior atenção da comunidade científica com a introdução da *World Wide Web* e da necessidade de motores de busca sofisticados (Baeza-Yates & Ribeiro-Neto, 1999). Embora a definição de recuperação de informação seja inicialmente baseada na idéia de consultas e respostas, sistemas que recuperam documentos baseados em palavras chaves, isto é, sistemas que recuperam documentos como a maioria dos motores de busca, são também chamados de sistemas de recuperação de informação.

Processamento de Língua Natural

Processamento de Língua Natural (PLN) consiste no desenvolvimento de métodos computacionais para a análise e manipulação ou codificação de informações expressas em língua natural (Allen, 1995). O objetivo geral do PLN é alcançar um melhor entendimento sobre a língua através do uso de computadores. Por exemplo, uma grande quantidade de textos em língua natural pode ser mais compreensível e útil quando sumarizada.

O processamento automático de textos pode empregar técnicas lingüísticas, estatísticas e, inclusive, técnicas consideradas simples como a manipulação de cadeias de caracteres (*strings*). Algumas das principais tarefas de PLN são reconhecimento de contexto, análise sintática, semântica, léxica e morfológica, sumarização e tradução de textos (Manning & Schütze, 2001).

2.3 Etapas do Processo de Descoberta de Conhecimento em Textos

O processo de Mineração de Textos (MT) ou *Knowledge Discovery from Text* (KDT) pode incluir etapas similares ao processo de KDD (Figura 2.1), nos quais não dados em geral, mas 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) – Figura 2.2.

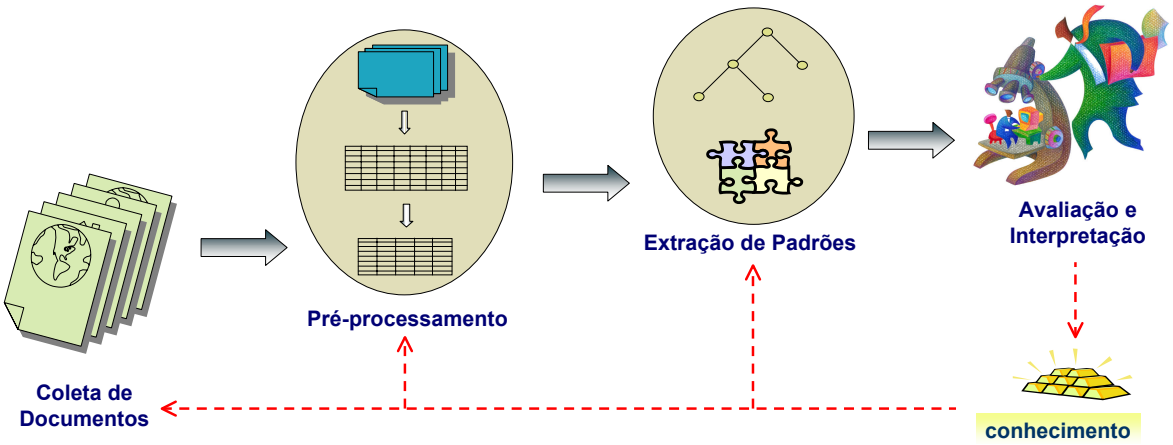


Figura 2.2. Etapas do processo de Mineração de Textos

2.3.1 Coleta de Documentos

A etapa de coleta de documentos é responsável por recuperar/coletar documentos com descrições textuais que sejam relevantes ao domínio de aplicação do conhecimento a ser extraído. 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.

Para facilitar a busca de documentos na internet, várias ferramentas de apoio têm sido construídas usando as seguintes abordagens: motores de busca baseados em robô (*robotic internet search engines*), motores de busca especializados baseados em robô (*robotic specialized search engines*), diretórios de assuntos (*subject directories*), mega-indexadores (*mega-indexes*) e mega-indexadores simultâneos (*simultaneous mega-indexes*) (Peterson, 1997). No entanto, essas abordagens podem ser classificadas em apenas dois grandes grupos, motor de busca e diretório WWW.

- **Motor de Busca** (*Search Engine*): neste grupo se enquadram as abordagens motores de busca baseados em robô, mega-indexadores, motores de busca especializados baseados em robô e mega-indexadores simultâneos. Pode-se citar como exemplos de motores de busca: Alta-

vista[1], Lycos[2], Excite[3], HotBot[4], Galaxy[5] e MetaCrawler[6];

- **Diretório WWW**: no qual se enquadra apenas a abordagem diretórios de assuntos. O exemplo mais conhecido para esta categoria é o Yahoo![7].

Na literatura podem ser encontrados diversos trabalhos relacionados à coleta de documentos provenientes da internet (Baeza-Yates, 1998; Joachims, Freitag & Mitchell, 1997). Técnicas de Aprendizado de Máquina podem ser empregadas para mapear o perfil de interesse do usuário com a finalidade de melhorar o resultado da coleta de documentos.

Entretanto, é possível que os documentos coletados estejam em uma grande variedade de formatos[8] e, dessa forma, pode ser útil converter tais documentos a um formato padrão, por exemplo o XML, antes de prosseguir para a etapa de pré-processamento.

2.3.2 Pré-processamento

Uma vez que os documentos estejam disponíveis, ou seja, a etapa de coleta de documentos tenha sido concluída, o próximo passo é realizar o pré-processamento desses documentos.

A etapa de pré-processamento é responsável por transformar uma coleção de documentos em uma representação estruturada adequada, normalmente no formato de uma tabela atributo-valor, a qual é mais apropriada para processamento do que simples arquivos textos. A etapa de pré-processamento geralmente possui um elevado custo computacional, e um organizado e cuidadoso planejamento nesta etapa é fundamental para obter um bom desempenho no processo de mineração de textos (Weiss et al., 2005a).

[1] `http://www.altavista.com`
[2] `http://www.lycos.com`
[3] `http://www.excite.com`
[4] `http://www.hotbot.com`
[5] `http://www.galaxy.com`
[6] `http://metacrawler.com/index.html`
[7] `http://www.yahoo.com`
[8] Alguns autores, por exemplo, podem criar seus documentos através de processadores de textos sofisticados; outros podem criar utilizando um simples editor de texto e salvar os documentos em formato ASCII; e outros podem escanear e armazenar os documentos como imagens.

Embora vários métodos empreguem técnicas que avaliam a estrutura sintática e semântica de textos, muitas abordagens de mineração de textos são baseadas na idéia que um documento pode ser representado por um conjunto de palavras, isto é, um documento pode ser descrito baseado em um conjunto de palavras presentes no texto (abordagem *bag-of-words*).

Essa representação, no entanto, pode resultar em uma tabela esparsa com alta dimensionalidade, portanto um objetivo da etapa de pré-processamento é reduzir a dimensionalidade dessa representação.

Representação de Documentos

Dada uma coleção de documentos, a abordagem *bag-of-words* consiste em representar cada documento da coleção, como um vetor de termos contidos no mesmo. Cada termo que ocorre no documento pode ser composto por apenas uma palavra (unigramas) ou várias palavras (bigramas, trigramas, ..., n-gramas). Com a finalidade de identificar todos os termos presentes em um documento, um procedimento de marcação (*tokenization*) é realizado, geralmente através do reconhecimento de espaços em branco, tabulações e sinais de pontuação que delimitam os termos. O conjunto de todos os diferentes termos que ocorrem em uma coleção de documentos é denominado de *dicionário* da coleção de documentos.

Considere uma coleção de documentos $D = \{d_1, d_2, ..., d_N\}$ e seu respectivo conjunto $T = \{t_1, t_2, ..., t_M\}$ de todos os diferentes termos presentes em D (*bag-of-words*). Essa coleção pode ser representada no formato de uma tabela atributo-valor, onde cada documento d_i corresponde a um exemplo (linha na tabela) e cada termo t_j corresponde a um atributo (coluna na tabela). A tabela atributo-valor, correspondente à representação de documentos, é apresentada na Tabela 2.1.

	t_1	$\dots$	t_k	$\dots$	t_M
d_1	v_{11}	$\dots$	v_{1k}	$\dots$	v_{1M}
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
d_j	v_{j1}	$\dots$	v_{jk}	$\dots$	v_{jM}
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
d_N	v_{N1}	$\dots$	v_{Nk}	$\dots$	v_{NM}

Tabela 2.1. Representação estruturada de documentos

O valor v_{jk} denota a importância relativa do termo t_k em relação ao do-

cumento d_j. Para quantificar esta importância, geralmente, utiliza-se uma representação vetorial, tal que para a atribuição de valores ao termos são empregadas medidas estatísticas baseadas na freqüência dos termos nos documentos (Weiss et al., 2005a; Ebecken, Lopes & Costa, 2003).

Dependendo da tarefa de mineração de textos, por exemplo categorização de documentos, é possível adicionar um atributo especial a tabela referente aos valores da classe/categoria de cada documento.

Redução da Representação

Em geral, uma das características do processo de mineração de textos é alta dimensionalidade do conjunto de atributos. Entretanto, em determinadas circunstâncias pode ser desejável aplicar métodos para a redução da representação, pois a alta dimensionalidade pode tornar o custo de processamento e armazenamento, em alguns casos, inviável.

Com a finalidade de reduzir a dimensionalidade da representação podem ser utilizados os métodos descritos, resumidamente, a seguir.

Filtragem A filtragem objetiva remover termos irrelevantes do *dicionário* e portanto da coleção de documentos. A remoção de termos, geralmente, baseia-se em um conjunto – *stoplist* – de palavras irrelevantes denominadas de *stopwords*. A idéia é remover termos (palavras) que tenham pouca ou nenhuma importância para análise de documentos, tais como preposições, artigos e conjunções.

Além do mais, termos que ocorrem com alta freqüência nos documentos geralmente não fornecem informação discriminativa suficiente e podem ser descartados, de forma análoga, termos que são pouco freqüentes provavelmente não possuem relevância estatística e, portanto, também podem ser descartados dos documentos (Yang & Pedersen, 1997; Frakes & Baeza-Yates, 1992).

Stemming Em inglês, como em muitas outras línguas, palavras ocorrem em textos em mais de uma forma. 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 prefixos e/ou 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.

Os algoritmos de *stemming* mais referenciados são os algoritmos do Porter (1980) e o algoritmo do Lovins (1968) que removem sufixos de termos. O algoritmo do Porter, proposto inicialmente para língua inglesa, utiliza um conjunto de regras pré-definidas para iterativamente transformar cada termo (palavra) ao seu provável *steam*.

Para a redução da dimensionalidade de atributos pode-se, inclusive, utilizar um *thesaurus* que organiza o valor semântico dos termos através do mapeamento de sinônimos, hierarquias e relacionamentos associativos entre termos (Ebecken, Lopes & Costa, 2003). No entanto, outras técnicas como a seleção de atributos relevantes também podem ser usadas (Forman, 2003; Yang & Pedersen, 1997).

Pré-processamento Lingüístico

Após a execução dos procedimentos citados anteriormente, em geral, o pré-processamento é concluído e pode-se avançar para a etapa de extração de padrões. No entanto, em casos onde o objetivo da mineração for, por exemplo, o reconhecimento de nomes próprios, lugares e organizações, pode ser necessário executar um (pré-)processamento lingüístico adicional e identificar/extrair atributos mais complexos (Weiss et al., 2005a; Manning & Schütze, 2001).

O processamento lingüístico, freqüentemente, consiste na execução das seguintes tarefas:

- **etiquetagem morfossintática** (*part-of-speech* (POS) *tagging*) determina a classe gramatical ou etiqueta morfossintática, por exemplo substantivo, verbo e adjetivo, para cada termo. Na Seção 2.4.1 são apresentados alguns dos principais conceitos relacionados a POS-*tagging*.
- **reconhecimento de frases**, também conhecido como *text chunking*, objetiva agrupar palavras adjacentes formando uma frase (sentença). Sistemas que desempenham esta tarefa, analisam um documento e identificam o início e fim de cada frase no documento (Weiss et al., 2005a);

- **desambiguação de sentido de palavras** busca resolver o problema da ambigüidade no significado de palavras ou frases. Um exemplo para língua inglesa é "*bank*", o qual pode ter, entre outros, os sentidos de "instituição financeira" ou "a margem de um rio ou lago";
- ***parsing*** produz a árvore de derivação sintática de uma sentença. A partir da árvore é possível encontrar a relação existente entre, uma palavra na sentença e todas as outras e, também, a sua função na sentença (sujeito, objeto, etc.).

Embora possam ser empregadas técnicas estatísticas, a justificativa para o processamento lingüístico citado anteriormente é para identificar atributos que possam ser úteis para a mineração de textos.

2.3.3 Extração de Padrões

Com a representação dos documentos em um formato adequado (tabela atributo-valor), é possível aplicar métodos para a extração de padrões novos, úteis e interessantes presentes nos documentos, de forma que o conhecimento extraído atenda aos objetivos e requisitos do usuário e/ou domínio da aplicação. Para a etapa de extração de padrões, geralmente, são utilizados os algoritmos de AM, sendo que grande parte deles são também utilizados em mineração de dados.

Algumas das principais tarefas relacionadas ao processo de Mineração de Textos (MT) são apresentadas a seguir.

Clustering de Documentos: consiste em identificar um conjunto finito de *clusters* (agrupamentos) a partir da coleção de documentos. Após o processo de *clustering*, os documentos são distribuídos entre um número de *clusters*, onde idealmente os documentos em um mesmo *cluster* são similares, e entre os *clusters*, bastante distintos. Os algoritmos de *clustering* de documentos são baseados em medidas de similaridade e consideram as palavras contidas nos documentos para determinar os agrupamentos (Zhong & Ghosh, 2003).

Categorização: dado um conjunto pré-definido de categorias ou classes, o objetivo da categorização é induzir um classificador que possa predizer se um (novo e desconhecido) documento pertence ou não a uma categoria (Yang & Pedersen, 1997). Como um documento pode pertencer a

várias categorias, a tarefa geralmente consiste em predizer se o documento pertence a cada uma das categorias em separado (classificação binária independente).

Extração de Informação: Extração de Informação (EI) é o processo de identificar e extrair informações específicas a partir de documentos textuais (Wilks, 1997). As informações extraídas podem ser estruturadas em banco de dados e, portanto, estarem disponíveis para uso posterior. No Capítulo 3 é dada uma descrição detalhada sobre o problema de extração de informação.

Sumarização: a sumarização consiste na criação de uma descrição compacta de um documento ou uma coleção de documentos, porém preservando seus significados-chave. Diversas abordagens concentram-se na idéia de extrair sentenças individuais informativas do documento original, numa tentativa de construir um texto menor que ainda conserve as idéias-chave originais (Radev, Hovy & McKeown, 2002).

A complexidade das tarefas de MT estão relacionadas a diversos fatores, tais como:

- o estilo no qual o documento está escrito. Por exemplo, documentos formais são mais fáceis de serem processados do que documentos informais;

- a língua adotada para elaborar o documento. Os algoritmos que manipulam textos são, geralmente, dependentes da língua;

- a natureza do conteúdo do documento. Documentos que contêm muita informação irrelevante são, freqüentemente, difíceis de serem processados, bem como documentos que contêm informação não textual, como figuras;

Assim, a escolha da tarefa a ser realizada no processo de MT depende desses fatores. Além disso, existem ainda os problemas com o tratamento de informações imperfeitas. Há cinco tipos de informações imperfeitas (Parsons, 1998), como pode ser visto na Tabela 2.2.

Tipos de informações	Descrição
INFORMAÇÕES INCOMPLETAS	Quando faltam valores para determinados atributos
INFORMAÇÕES IMPRECISAS	Quando há variações na granularidade dos valores (às vezes, os valores não podem ser medidos com precisão). Por exemplo, o tempo em que ocorreu um evento pode ser medido em anos, meses, dias, turnos, horas, minutos, etc.
INFORMAÇÕES INCERTAS	Ocorre porque as decisões são tomadas baseadas em fatos que não se conhece a veracidade, devido, por exemplo, às contradições nas fontes da informação ou às representações falhas
INFORMAÇÕES VAGAS	São decorrentes da imprecisão de termos
INFORMAÇÕES INCONSISTENTES	Quando há valores conflitantes

Tabela 2.2. Tipos de informações imperfeitas

2.3.4 Avaliação 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 (Ebecken, Lopes & Costa, 2003).

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, estatísticamente, avaliados em termos das métricas precisão, cobertura e *F-measure*. Essas medidas são apresentadas no Capítulo 3 na Seção 3.4.

A visualização gráfica de informações pode auxiliar a analisar e compreender o sentido de grandes e complexos conjuntos de documentos. Diversas ferramentas de visualização, capazes de ilustrar propriedades e relacionamentos intrínsecos de dados, podem ser utilizadas para a exploração interativa de conjuntos de dados (Lopes et al., 2007; Keim, 2002).

Como comentado anteriormente, um dos propósitos deste trabalho é extrair automaticamente informações de referências bibliográficas. As referências, normalmente, possuem as seguintes informações: autores, título, publicação (periódicos, revistas, etc.) ou evento (simpósios, congressos,

etc.), editora, páginas e ano (ver Seção 3.5).

A proposta desenvolvida nesta dissertação para extrair automaticamente informações de documentos textuais, baseia-se no mapeamento do problema de *part-of-speech* (POS) *tagging* ao problema de extração de informação. Ilustrado na Figura 2.3, o mapeamento para a extração de informações de um documento consiste em, inicialmente, etiquetar todos os termos do documento selecionando alguma etiqueta de um conjunto pré-definido de etiquetas e, posteriormente, combinar e extrair as informações de etiquetas correspondentes. Neste mapeamento, um termo com uma etiqueta ou combinação de termos consecutivos com a mesma etiqueta equivalem a uma unidade de informação a ser extraída do documento.

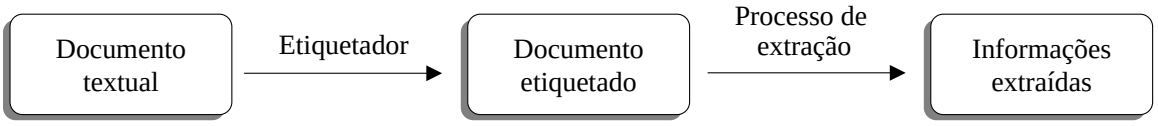


Figura 2.3. Mapeamento do problema de *part-of-speech* (POS) *tagging* ao problema de extração de informação

Este mapeamento para o problema de extração de informações de referências bibliográficas, consiste em etiquetar todos os termos das referências selecionando alguma etiqueta de um conjunto de possíveis etiquetas, por exemplo AUTOR, TÍTULO, REVISTA, EVENTO, LOCAL, PÁGINAS e ANO, e, posteriormente, combinar e extrair as informações (etiquetas correspondentes) das referências.

Os principais conceitos relacionados ao problema de POS-*tagging* são apresentados na próxima seção, especificamente na Seção 2.4.1. O etiquetador desenvolvido por Eric Brill, nomeado de TBL, utilizado em experimentos neste trabalho é descrito em detalhes na Seção 2.4.2.

2.4 Etiquetagem Automática de Corpus

A área de lingüística de corpus abrange uma grande variedade de níveis de análise. Um tipo de análise, por exemplo, poderia identificar todas as ocorrências de uma determinada cadeia de caracteres em um corpus, permitindo que estas sejam ao final manipuladas. No entanto, alguns tipos de análises necessitam de mais informações a respeito das palavras presentes no corpus, como por exemplo, informações de natureza gramatical.

Dessa forma, a partir de um corpus previamente etiquetado é possível induzir regras para extrair informações importantes contidas no corpus. Tais etiquetas (rótulos) são, principalmente, as categorias gramaticais (morfossintáticas) das palavras no corpus. Pode-se, por exemplo, a partir de um corpus etiquetado, recuperar o conjunto de palavras que pertençam a uma determinada categoria gramatical.

A etiquetagem de corpus, geralmente, não é uma tarefa simples. Existem diversas pesquisas em etiquetagem de corpus com informação lingüística, incluindo categorias gramaticais e estruturas sintáticas (Leech, Garside & Bryant, 2004; Marcus, Santorini & Marcinkiewicz, 1994). A etiquetagem automática é uma tarefa bem conhecida e bastante explorada em Processamento de Língua Natural e pode ser aplicada em várias áreas do processamento de informação, tais como pré-processamento para sumarização automática, pós-processamento para reconhecimento ótico de caracteres (OCR) e reconhecimento de fala, análise sintática (*parsing*), tradução automática e recuperação de informações, e mesmo para a etiquetagem de corpus, pois a etiquetagem manual de textos grandes é uma tarefa custosa.

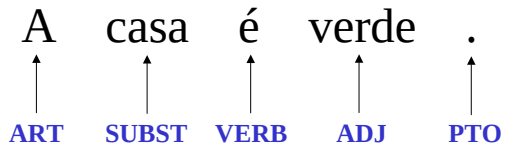
As ferramentas utilizadas na etiquetagem automática de corpus são os etiquetadores (*taggers*). Na próxima seção são apresentados os principais conceitos sobre etiquetagem morfossintática de textos, as etapas que compõem a tarefa de etiquetagem e as abordagens de etiquetagem existentes.

2.4.1 Etiquetagem Morfossintática de Textos

A etiquetagem morfossintática[9] de um texto em uma dada língua significa atribuir um rótulo ou etiqueta (*tag*), pertencente a um conjunto definido de etiquetas (*tagset*), a cada palavra, símbolo de pontuação, palavra estrangeira, ou fórmula matemática presente no texto, conforme o contexto em que essas informações aparecem. Para palavras da língua, utiliza-se uma etiqueta referente a sua categoria gramatical (adjetivo, advérbio, artigo, substantivo, etc.); para símbolos de pontuação (vírgula, ponto, ponto-e-vírgula, parênteses, aspas, etc.) freqüentemente utiliza-se o próprio símbolo; palavras estrangeiras, fórmulas matemáticas, ou alguma outra denominação no texto, geralmente são rotuladas com uma única etiqueta (EAGLES, 1996). Um exemplo simples da etiquetagem morfossintática de uma sentença para a língua portuguesa é apresentado na Figura 2.4. É ilus-

[9]Denominada, no inglês, de *part-of-speech* (POS) *tagging*.

trada também a possibilidade de se utilizar uma etiqueta diferente (PTO) para designar o ponto final de uma sentença.



ART	artigo
SUBST	substantivo
VERB	verbo
ADJ	adjetivo
PTO	ponto

Figura 2.4. Exemplo de *part-of-speech tagging*

As etiquetas para cada classe gramatical das palavras podem, por sua vez, ser refinadas com atributos referentes a cada classe. Para a língua portuguesa, por exemplo, o atributo substantivo pode ser refinado com relação ao tipo (comum, próprio), grau (aumentativo, diminutivo), gênero (masculino, feminino), e assim por diante. Pode-se inclusive criar uma etiqueta para uma única palavra específica no texto. O processo geral de etiquetagem de um texto é apresentado na Figura 2.5. Dado um conjunto de etiquetas e uma seqüência de termos do texto, o processo geral de etiquetagem consiste em associar a cada termo a sua respectiva etiqueta.

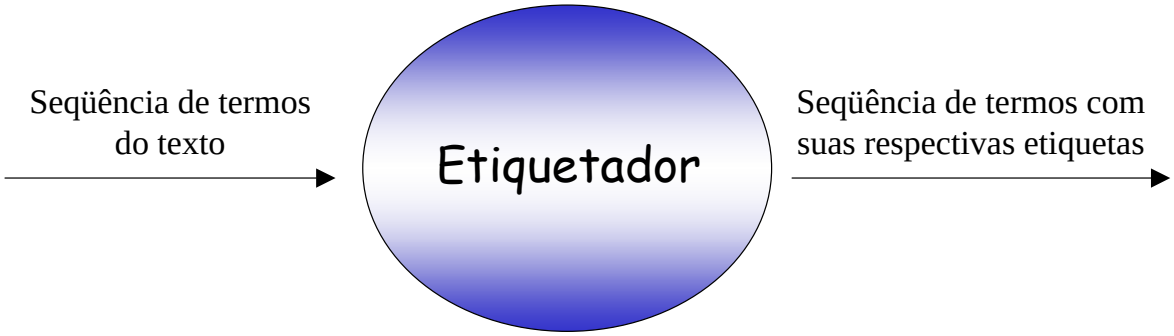


Figura 2.5. Processo geral de etiquetagem de texto

O etiquetador pode ser induzido automaticamente, constituído-se, em essência, de um conjunto de regras que consideram a palavra e/ou seu contexto (um conjunto de palavras ou etiquetas à esquerda e à direita dessa palavra) para determinar sua etiqueta.

A tarefa de etiquetagem automática é dividida em três módulos: *escrutinador léxico*, *classificador gramatical* e *desambigüizador*. O *escrutinador léxico* é responsável por separar (*tokenization*) e identificar as orações (sentenças) no texto. Cada termo (palavras, sinais de pontuação, etc.) no texto é separado, normalmente, através do reconhecimento de espaços em branco. A maioria dos etiquetadores consideram que os textos de entrada estão no

formato ideal e, dessa forma, não possuem um escrutinador léxico. O *classificador gramatical* designa classes gramaticais aos termos no texto com o auxílio, por exemplo, de um léxico e de um conjunto de informações para reconhecer termos (normalmente palavras) que não estão contidos no léxico. O *desambiguizador* utiliza informações relativas ao contexto para associar uma, e somente uma, classe gramatical (etiqueta) a cada termo do corpus. Este método objetiva resolver o problema de ambigüidade lexical, situação em que uma mesma palavra possua mais de uma categoria gramatical, e, nestes casos, o contexto deve ser considerado para a desambiguação. Tanto o léxico, quanto as informações utilizadas para avaliar o contexto integram o modelo da língua utilizado por cada etiquetador.

Os etiquetadores também podem ser construídos manualmente por lingüistas que, utilizando conhecimentos lingüísticos, desenvolvem um conjunto de regras de etiquetagem.

De acordo com o tipo de conhecimento utilizado para representar o modelo da língua, os etiquetadores podem ser classificados nas seguintes abordagens.

Simbólica: os etiquetadores da abordagem simbólica (ou lingüística) são baseados em regras (Lopes & Jorge, 2000; Voutilainen, 1995), casos (Daelemans, Zavrel & Berck, 1996), restrições (Chanod & Tapanainen, 1995) e árvores de decisão não probabilísticas.

Probabilística: etiquetadores probabilísticos fundamentam-se em técnicas como redes neurais (Ma et al., 1999), máxima entropia (Ratnaparkhi, 1996), Modelo de Markov (Wilkens & Kupiec, 1996) e árvores de decisão probabilísticas (Schmid, 1995).

Híbrida: os etiquetadores da abordagem híbrida empregam tanto conhecimento simbólico quanto probabilístico no processo de etiquetagem. Como um exemplo desta abordagem pode-se citar o etiquetador TBL (baseado em transformação dirigida por erro) (Brill, 1997; Brill, 1995; Brill, 1994).

Atualmente, os etiquetadores mais utilizados são os probabilísticos. Seu funcionamento basicamente consiste na construção de um modelo estatístico da língua que é utilizado para desambigüisar um conjunto de palavras. Este modelo, em geral, aparece como um conjunto de freqüências de

diferentes tipos de fenômenos lingüísticos e é construído através da observação de n-gramas, sendo que o mais comum é a modelagem na forma de unigramas, bigramas e trigramas. Os n-gramas são seqüências de n termos adjacentes.

De forma a maximizar as vantagens encontradas nas abordagens simbólica e probabilística e minimizar as desvantagens, Eric Brill desenvolveu um etiquetador híbrido que contém um componente estatístico e outro simbólico.

Neste trabalho optou-se pelo uso do etiquetador de Eric Brill por ser bastante conhecido, disponível e de uso relativamente fácil. Observa-se, entretanto, que a abordagem de Extração de Informação proposta não depende exclusivamente desse etiquetador.

2.4.2 Etiquetador Baseado em Transformação (TBL)

O etiquetador desenvolvido por Eric Brill, denominado TBL, baseia-se no algoritmo de Aprendizado Baseado em Transformação Dirigida por Erro (Brill, 1997; Brill, 1995; Brill, 1994). Tal algoritmo, ilustrado na Figura 2.6, pode ser aplicado a vários problemas, tais como a etiquetagem morfossintática e a análise sintática.

Aprendizado Baseado em Transformação Dirigida por Erro

Conforme apresentado na Figura 2.6, o corpus não-etiquetado passa inicialmente por uma etapa de etiquetagem. O sucesso do algoritmo depende em grande parte do sucesso desta etapa inicial, a qual pode ser baseada na freqüência de categorias gramaticais para cada palavra de um léxico. Após ter sido etiquetado, o corpus é comparado com o mesmo corpus manualmente etiquetado, e uma lista de transformações é induzida para ser aplicada à saída do etiquetador inicial. Cada transformação é composta por uma regra de reescrita e pelo contexto que irá desencadear esta regra[10]. A cada iteração de aprendizado, de acordo com alguma função objetivo, uma transformação que melhora o resultado da etiquetagem é encontrada e adicionada à lista ordenada de transformações, e o corpus etiquetado é

[10]Observar que algumas palavras têm apenas um papel morfossintático (*tag*), enquanto outras podem ter mais de um e, neste caso, o contexto tem que ser considerado para a desambiguação.

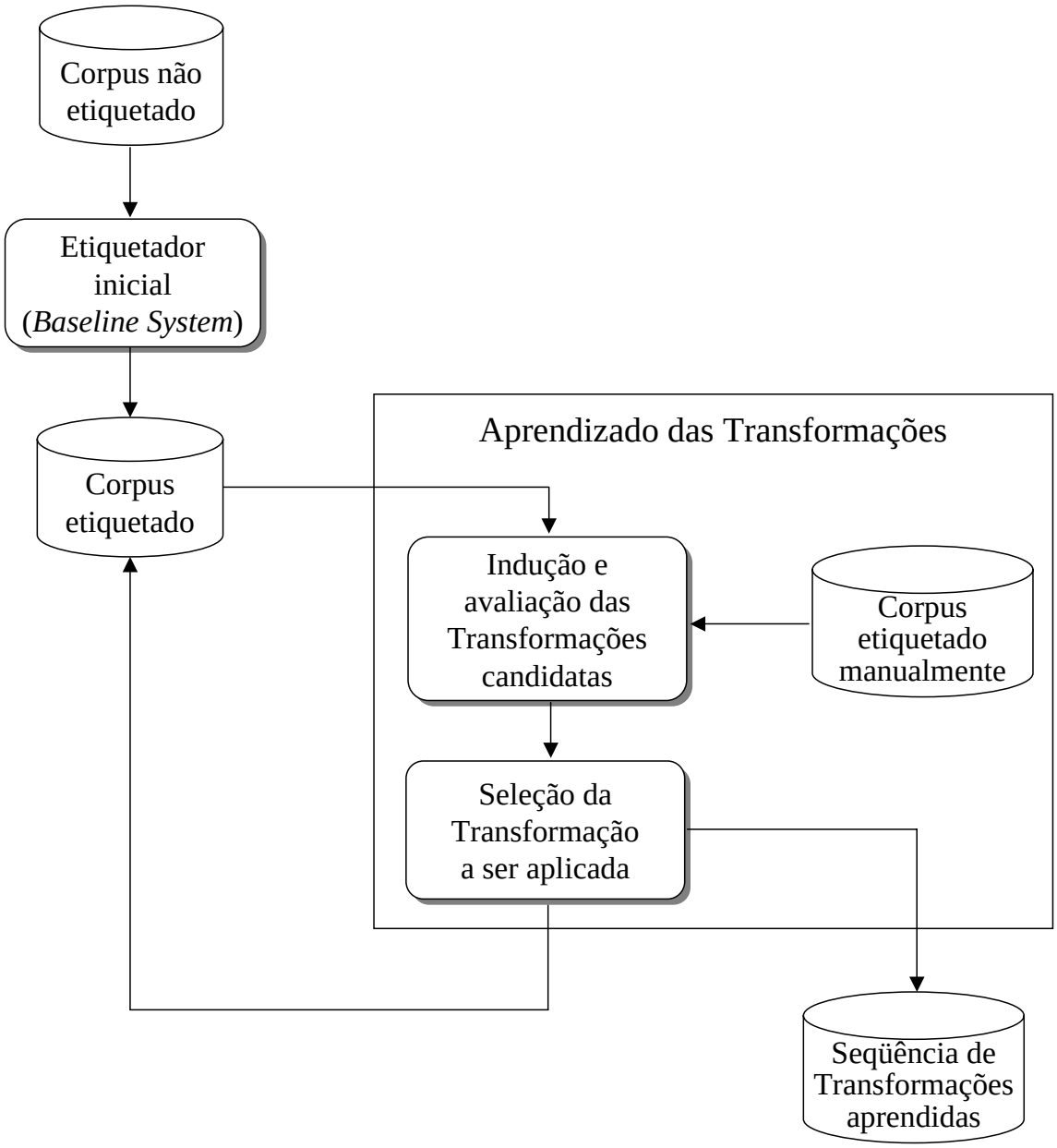


Figura 2.6. Algoritmo de Aprendizado Baseado em Transformação Dirigida por Erro

atualizado aplicando sobre ele a transformação aprendida. Esse processo se repete até que não seja mais encontrada uma transformação que melhore o corpus etiquetado.

Aprendizagem das Regras de Transformação

O etiquetador TBL possui dois módulos de etiquetagem: um módulo que gera regras, não contextuais, para determinar o conjunto de etiquetas mais prováveis de palavras (conhecidas e desconhecidas) e outro módulo que gera regras contextuais para melhorar a precisão da etiquetagem. O etiquetador apenas altera a etiqueta de uma palavra de X para Y se:

- a palavra não aparece no corpus de treinamento ou
- a palavra foi etiquetada com Y pelo menos uma vez no corpus de treinamento.

O primeiro módulo, responsável pela etapa de etiquetagem inicial, constrói um léxico a partir do corpus manualmente etiquetado com a finalidade de atribuir a etiqueta mais provável para cada palavra - Figura 2.7. O léxico é formado pela palavra seguida de sua etiqueta mais comum.

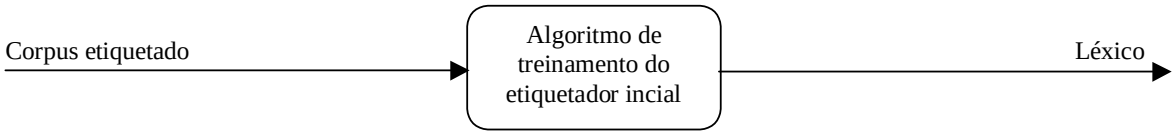


Figura 2.7. Etiquetador Baseado em Transformação Dirigida por Erro: treinamento do etiquetador inicial

Para etiquetar palavras desconhecidas (não presentes no léxico), o etiquetador inicial utiliza outro procedimento. Inicialmente, define que as palavras desconhecidas que iniciam por letra maiúscula tendem a ser substantivos próprios e as que iniciam por letras minúsculas tendem a ser substantivo comum. Em seguida o etiquetador gera regras léxicas para minimizar o erro desta etiquetagem inicial, utilizando apenas informações intrínsecas das palavras, isto é, sem considerar o contexto. Alguns modelos de regras para palavras desconhecidas podem ser vistos na Tabela 2.3.

O segundo módulo, etiquetador contextual, infere automaticamente as regras relativas ao contexto, a partir do corpus de treino (manualmente) etiquetado. Isto é feito ao realizar a etiquetagem do corpus de treino com o

Mude a etiqueta de uma palavra desconhecida de A para B se:
1. Removendo prefixo (sufixo) **l**, |**l**| <= 4, resulta em uma palavra no léxico (**l** é qualquer *string* de tamanho 1 a 4)
2. O primeiro (último) (1,2,3,4) caracteres da palavra é **c**
3. Adicionando a *string* **m** como prefixo (sufixo) resulta em uma palavra (|**m**|<=4)
4. A palavra **m** nunca aparece imediatamente a esquerda (direita) da palavra
5. O caractere **c** aparece na palavra

Tabela 2.3. Regras para palavras desconhecidas

etiquetador inicial, e em seguida comparar automaticamente os resultados e gerar a lista ordenada de transformações. A partir da aplicação dessa lista de transformações no corpus são geradas as regras de contexto. As aplicações que acarretarem o melhores resultados são utilizadas como regras de contexto - Figura 2.8. Portanto, obtém-se um conjunto de regras que analisa a atribuição das etiquetas feita pelo etiquetador inicial e as corrige conforme o contexto no qual as palavras aparecem.

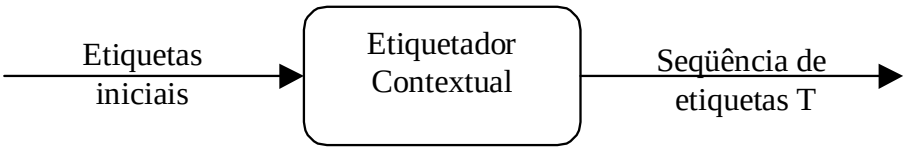


Figura 2.8. Etiquetador Baseado em Transformação Dirigida por Erro: etiquetador contextual

Como apresentado na Tabela 2.4, as regras contextuais podem fazer referência a etiquetas anteriores/posteriores (regras não lexicalizadas) ou a palavras anteriores/posteriores (regras lexicalizadas). Na Tabela 2.4 as variáveis **z**, **x** e **t** referem-se a todas as classes gramaticais (etiquetas) e as variáveis **w** e **y** abrangem todas as palavras do corpus de treinamento.

Um exemplo de aprendizado baseado em transformações é ilustrado na Figura 2.9. Neste exemplo assume-se que existam apenas quatro transformações possíveis (T1, T2, T3 e T4) e que a função objetivo utilizada seja o número total de erros. O corpus não-etiquetado passa pela etiquetagem inicial, e o resultado é um corpus etiquetado com 5100 erros. Em seguida, cada uma das transformações selecionadas são aplicadas *em ordem*. Neste exemplo, T2 foi a primeira transformação da lista, pois foi a transformação que possibilitou a maior redução de erros. É então aplicada a todo o corpus e o aprendizado continua. Em seguida a transformação que mais diminuiu o número de erros foi T3, logo T3 é aprendida como a segunda transfor-

Mude a etiqueta de uma palavra de A para B quando:
1. A palavra anterior (posterior) foi etiquetada como **z**
2. A palavra duas posições antes (depois) foi etiquetada como **z**
3. Uma das duas palavras anteriores (posteriores) foi etiquetada como **z**
4. Uma das três palavras anteriores (posteriores) foi etiquetada como **z**
5. A palavra anterior foi etiquetada como **z** a posterior como **x**
6. A palavra anterior (posterior) foi etiquetada como **z** e a palavra "duas posições antes" foi etiquetada como **x**
7. A palavra que vem antes (depois) é **w**
8. A segunda palavra que vem antes (depois) é **w**
9. Se uma das duas palavras que vem antes (depois) é **w**
10. Se a palavra atual é **w** e a anterior (posterior) é **y**
11. Se a palavra atual é **w** e a palavra anterior (posterior) é classificada como **t**
12. Se a palavra atual é **w**
13. Se a palavra anterior (posterior) é **w** e a classificação anterior (posterior) é **t**
14. Se a palavra atual é **w** e a palavra anterior (posterior) é **y** e a classificação anterior (posterior) é **t**

Tabela 2.4. Regras que utilizam informações relativas ao contexto

mação da lista. Partindo de T3 nota-se que não existem mais reduções no número de erros aplicando transformações, então o processo termina.

Para a etiquetagem de um corpus, supondo que o etiquetador TBL esteja treinado, consiste em submeter tal corpus ao etiquetador inicial para a etiquetagem de palavras conhecidas (léxico) e desconhecidas, e posteriormente aplicar em seqüência a lista de regras contextuais no corpus. As transformações que estão mais próximas do início da lista são as transformações que produzem um melhor resultado no corpus etiquetado. É importante ressaltar que uma transformação feita no início, pode ocasionar outras transformações posteriormente.

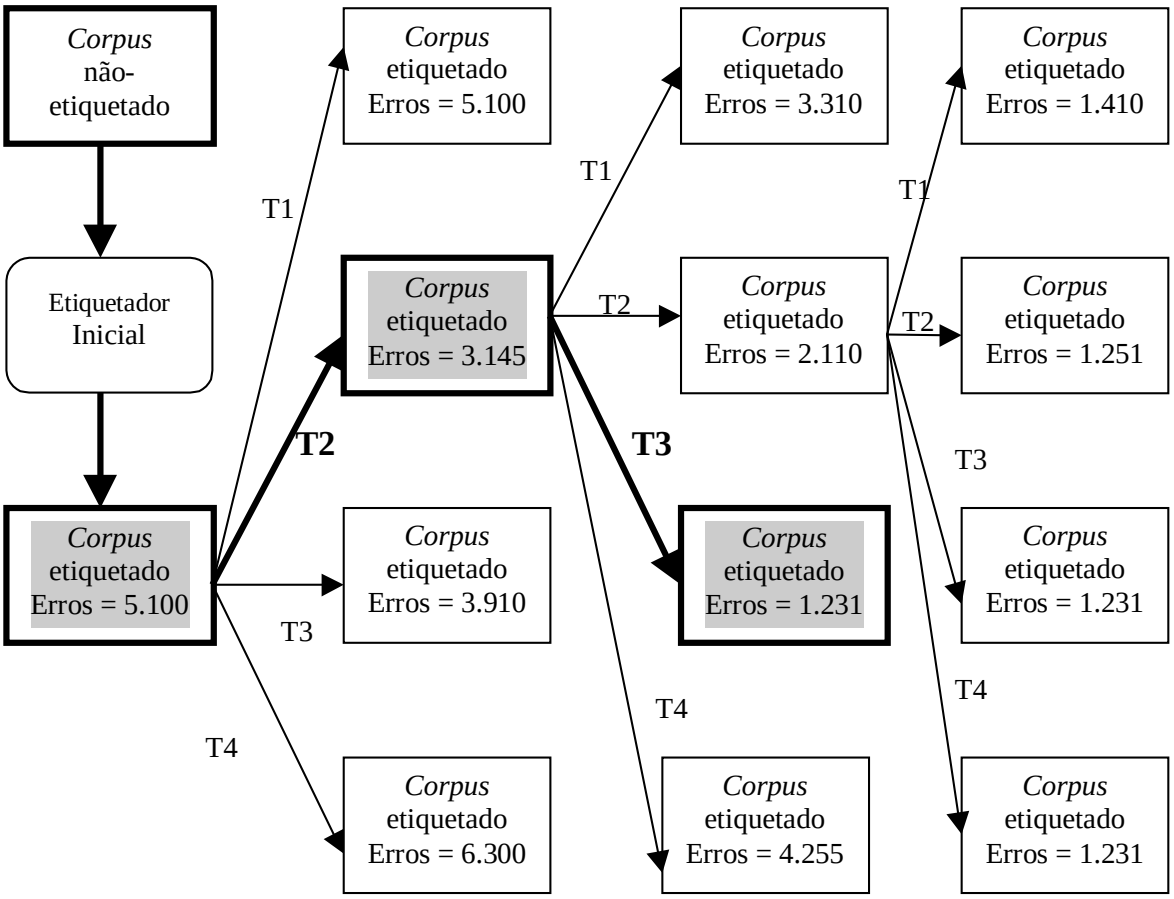


Figura 2.9. Exemplo de Aprendizado Baseado em Transformação Dirigida por Erro

Capítulo 3

Extração de Informação

Extração de Informação (EI) é originalmente a tarefa de encontrar informações específicas a partir de grandes volumes de documentos. Tais documentos podem apresentar algum nível de estruturação na apresentação das informações, como também podem ser totalmente livres. Neste capítulo são apresentados um breve histórico da área, a arquitetura típica de um sistema de EI, as medidas de avaliação utilizadas e os principais conceitos e técnicas utilizados por esses sistemas. É apresentado também o problema da extração de informações de artigos científicos, bem como alguns dos principais trabalhos relacionados com este tema.

3.1 Objetivo da Extração de Informação

A área de Extração de Informação visa localizar e extrair, de forma automática, informações relevantes em um documento ou coleção de documentos expressos em língua natural e estruturar tais informações para os padrões de saída, por exemplo em banco de dados ou textos em língua natural[1], a fim de facilitar sua manipulação e análise (Grishman, 1997). Sistemas de EI não realizam o entendimento completo dos documentos, pelo contrário, tais sistemas analisam partes dos documentos que possuam informações relevantes.

O objetivo da pesquisa em EI é construir sistemas que encontrem e combinem informações relevantes enquanto ignoram informações insignificantes e irrelevantes (Cowie & Lehnert, 1996). É importante ressaltar que a

[1]Essa forma de estruturação envolve, por exemplo, trabalhos relacionados com a sumarização automática de documentos.

informação extraída é determinada por um conjunto de padrões ou regras de extração específicas ao seu domínio. A definição de tais padrões pode ser feita manualmente, por algum especialista, ou com diferentes graus de automação.

Extração de Informação não deve ser confundida com a desenvolvida área de Recuperação de Informação (RI), a qual seleciona, de uma grande coleção, um subconjunto de documentos relevantes baseados em uma consulta do usuário. O contraste entre os objetivos dos sistemas de EI e RI pode ser declarado da seguinte forma: RI recupera documentos relevantes de uma coleção, ao passo que EI extrai informações relevantes dos documentos. Portanto, as duas técnicas são complementares e quando combinadas podem produzir ferramentas interessantes para processamento de texto (Gaizauskas & Wilks, 1998).

Ainda de acordo com Gaizauskas e Wilks (Gaizauskas & Wilks, 1998), a área de Extração de Informação, diferentemente de Recuperação de Informação, teve um crescimento acelerado nas duas últimas décadas. Dois fatores foram fundamentais para esse crescimento: o aumento exponencial da quantidade de dados textuais disponibilizados na forma digital, e a atenção dada a área através das Conferências de Entendimento de Mensagens (MUC) durante a década de 80.

Patrocinados pelo governo norte americano, o objetivo das MUCs foi avaliar e avançar as pesquisas em Extração de Informação. A MUC era formada por vários grupos de pesquisa acadêmicos e industriais de diferentes regiões. Cada grupo construía um sistema de EI para um determinado domínio, sendo que ao final todos os sistemas eram avaliados sobre um mesmo domínio e os resultados, computados utilizando um programa oficial de avaliação.

O objetivo principal das conferências foi obter um padrão de avaliação para os sistemas de EI, os quais anteriormente as essas conferências eram avaliados de forma esporádica e, freqüentemente, sobre o mesmo conjunto de dados em que foram treinados. Tais conferências forneceram um primeiro esforço em larga escala para avaliar sistemas de Processamento de Língua Natural (PLN).

3.2 Arquitetura de um Sistema de Extração de Informação

O processo de extração de informação possui duas etapas principais. Primeiro, o sistema extrai fatos (unidades de informação) do texto de um documento através da análise local do texto. Segundo, o sistema integra e combina esses fatos produzindo fatos maiores ou novos fatos (por alguma inferência). Ao final, os fatos considerados relevantes ao domínio são estruturados para o padrão de saída.

Para estruturar as informações ao padrão de saída, as técnicas de EI baseadas em PLN utilizam modelos (*templates*) que são estruturas com campos (*slots*) a serem preenchidos pelas informações que devem ser extraídas de um texto.

Com base na arquitetura definida por Grishman (Grishman, 1997) foram identificados seis módulos principais presentes em sistemas de EI baseados em PLN: processador léxico, reconhecimento de nomes, analisador sintático/semântico, padrões de extração, analisador do discurso e integração e preenchimento de *templates* - Figura 3.1.

Inicialmente o texto é dividido em sentenças e em termos. A separação dos termos (*tokenization*) é realizada, normalmente, pelo reconhecimento de espaços em branco e outros sinais de pontuação que delimitam os termos. Após a separação é feita uma análise léxica e morfológica dos termos para determinar a sua possível classe morfossintática (substantivo, verbo, artigo, etc.), e demais características (feminino, plural, etc.). Neste módulo (processador léxico) é comum a utilização de autômatos finitos para o reconhecimento das informações (Hobbs et al., 1997).

O próxima etapa do processamento identifica vários tipos de nomes próprios e outros itens que possam ter uma estrutura interna, tais como data e hora. Nomes são identificados por um conjunto de expressões regulares, expressos em função das classes morfossintáticas (*part-of-speech*) e características sintáticas e ortográficas (letras maiúsculas) presentes nos termos. Cowie e Lehnert citam como exemplos de reconhecimento de nomes próprios, palavras que iniciam com letra maiúscula e por, geralmente, virem próximas dos termos “Senhor” e “ltda” (Cowie & Lehnert, 1996).

O módulo de análise sintática e semântica é responsável por receber uma seqüência de itens léxicos e tentar construir uma estrutura sintá-

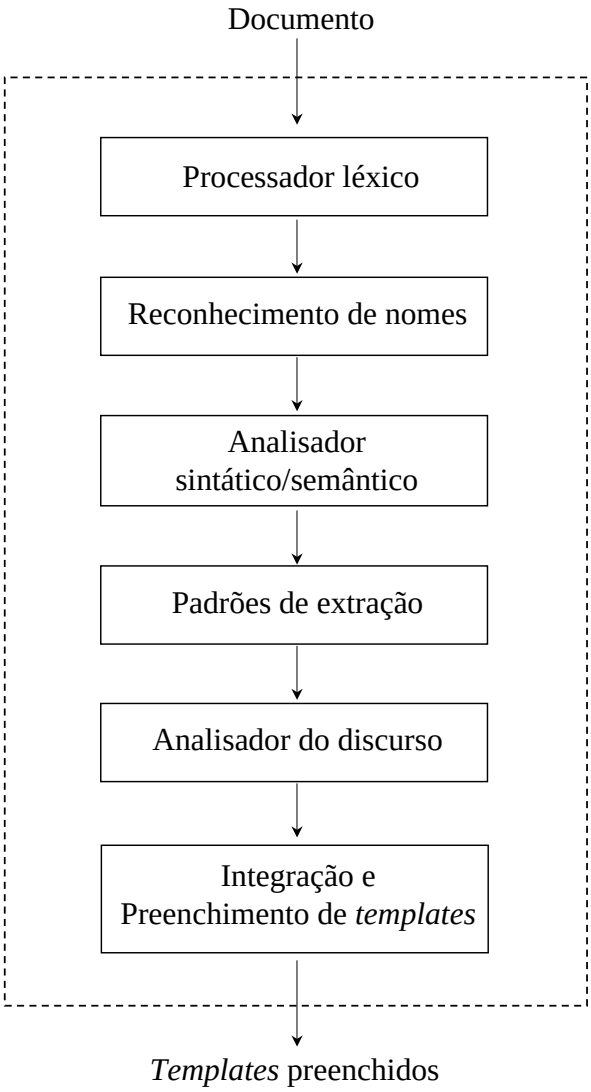


Figura 3.1. Estrutura de um sistema de Extração de Informação baseado em Processamento de Língua Natural (Grishman, 1997)

tica, juntamente com alguma informação semântica, para cada sentença do texto. Nas sentenças são identificados vários níveis de constituintes (segmentos de texto) e para cada constituinte dos grupos nominais e verbais conhecidos, são associadas algumas características que podem ser combinadas nas fases seguintes com os padrões de extração. Por exemplo, nos grupos nominais, pode-se incluir informações sobre a raiz do núcleo, se ele corresponde ou não a um nome próprio, assim como o seu papel semântico no contexto da frase. O papel semântico de um grupo nominal inclui informações (grupos nominais relacionados) que auxiliam a sua compressão no contexto da frase (Wiebe, Hirst & Horton, 1996).

A construção de regras ou padrões de extração consiste na indução

de um conjunto de regras de extração específico para o domínio tratado. Em geral, esses padrões baseiam-se em restrições sintáticas e semânticas, aplicadas aos constituintes das sentenças (Muslea, 1999). A etapa de análise do discurso tem como objetivo relacionar diferentes elementos do texto. Esta fase considera o relacionamento entre as sentenças, ao contrário das anteriores. Caso for necessário realizar algum processo de inferência sobre a informação, tornando-a explícita, pode ser realizado nesta etapa. Este módulo inclui a tratamento das seguintes tarefas:

- análise de frases nominais, que se refere à tarefa de reconhecer e interpretar apostos e outros grupos nominais complexos,
- resolução de correferência, que trata o problema de identificar quando uma nova frase nominal, normalmente um pronome, se refere a outra já citada anteriormente, e
- descoberta de relacionamento entre as partes do texto, que objetiva estruturar as palavras do texto em uma rede associativa, fornecendo suporte à tarefa de extração.

Finalmente, as informações parciais são combinadas e os *templates*, definidos pela aplicação, são preenchidos com as informações relevantes ao domínio.

3.3 Técnicas e Sistemas de Extração de Informação

Como já foi comentado, os textos ou documentos dos quais são extraídas as informações de interesse podem apresentar algum nível de estruturação na apresentação dos dados, como também podem ser totalmente livres. O tipo de texto de onde é feita a extração tem grande influência sobre a escolha da técnica a ser utilizada na construção de sistemas de EI, pois tal técnica pode se basear apenas na estrutura do texto, quando existente. A seguir é dada uma breve descrição dos possíveis tipos de textos.

Estruturado: um texto é considerado estruturado quando apresenta regularidade no formato de apresentação das informações. Essa regularidade, facilmente capturada por sistemas para EI, permite que cada

elemento de interesse seja identificado com base em regras uniformes, que consideram marcadores textuais tais como delimitadores, e/ou ordem de apresentação dos elementos. Como exemplo, pode-se citar um formulário preenchido.

Semi-estruturado: os textos semi-estruturados são aqueles que apresentam alguma regularidade na disposição dos dados. Alguns dados do texto podem apresentar uma formatação, enquanto outras informações aparecem de forma irregular. É o caso da primeira página de um artigo que, em geral, não segue um formato rígido, permitindo variações na ordem e na maneira com que as informações são apresentadas. Por exemplo, para mais de um autor, quando e-mails possuem o mesmo domínio, geralmente são informados de uma vez, separados por vírgula e entre chaves.

Não-estruturado: os textos não estruturados (livres) são aqueles que não exibem regularidade na apresentação dos dados. Neste caso, os dados a serem extraídos não são facilmente detectados, a menos que se tenha um conhecimento lingüístico sobre eles. Como exemplo deste tipo de texto, pode-se citar uma página Web.

Técnicas de Processamento de Língua Natural têm sido amplamente utilizadas no processo de extração de informações de documentos semi-estruturados e livres (Soderland, 1999; Cowie & Lehnert, 1996). O objetivo do uso dessas técnicas de PLN no contexto de EI é tentar compreender textos em alguma língua natural, a fim de encontrar informações relevantes a serem extraídas. Sistemas de extração baseados em PLN têm sido definidos para diferentes domínios, contando com etapas de processamento comuns aos sistemas de PLN em geral, e mais alguns módulos específicos para Extração de Informação (Rajman & Besançon, 1997).

Diversos trabalhos relacionados com a tarefa de EI são encontrados na literatura. Em geral esses métodos utilizam reconhecedores de estado-finito (Hobbs et al., 1997). O problema da adaptação a novos domínios, isto é, a criação de regras de acordo com o tipo de texto analisado, emprega técnicas de Aprendizado de Máquina para automatizar a aquisição das regras a serem usadas em um novo domínio, visando minimizar a participação humana (Glickman & Jones, 1999). Técnicas estatísticas de aprendizado de máquina, por exemplo *Hidden Markov Model* (HMM), estão sendo aplicadas em Extração de Informação, especialmente em tarefas como aprender

modelo de uma estrutura a partir de dados e como fazer o melhor uso de dados rotulados e não rotulados (Yin et al., 2004; Freitag & McCallum, 2000; Connan & Omlin, 2000; Seymore, McCallum & Rosenfeld, 1999).

Sistemas baseados em *Hidden Markov Model*s como o DATAMOLD (Borkar, Deshmukh & Sarawagi, 2001), DVHMM (Takasu, 2003) e AUTOBIB (Geng & Yang, 2003), são sistemas determinísticos de aprendizagem de regras que extraem informações de textos não estruturados e criam um registro estruturado. A precisão alcançada na extração de informações é geralmente alta. Contudo, sistemas que usam HMM costumam consumir muito tempo de processamento (Geng & Yang, 2003).

Outra abordagem utilizada para EI é a indução de *wrappers*. Sistemas *wrappers* exploram a regularidade apresentada por textos estruturados com o propósito de localizar informações relevantes. Em geral, um *wrapper* possui o objetivo principal de extrair informações relevantes presentes em documentos e exportar essas informações como parte de uma estrutura de dados, por exemplo em banco de dados (Freitag & Kushmerick, 2000). No contexto da Web, o propósito de um *wrapper* é converter informações implícitas armazenadas em páginas HTML em informações explícitas estruturadas, para posterior processamento (Eikvil, 1999). Normalmente, os *wrappers* são construídos de maneira ad-hoc, não existindo uma arquitetura consensual nesse tipo de sistema. Cada um possui modularidade, facilidade de manutenção e reuso do código de acordo com as suas necessidades individuais. Quanto à técnica de implementação, esses sistemas podem ser construídos de forma automática, semi-automática ou completamente manual (Freitag & Kushmerick, 2000).

Ciravegna (Ciravegna 2001) desenvolveu uma abordagem recente para a tarefa de extração. Seu sistema de aprendizado simbólico induz regras que inserem independentemente *tags* (marcas) em um texto, e possui quatro etapas: uma etapa inicial de inserção de *tags*; uma etapa de regras contextuais, no qual regras são aplicadas considerando a dependência entre *tags*; uma etapa de correção, no qual regras são aplicadas para ajustar a localização de *tags*; uma etapa de validação, onde *tags* incorretas são removidas. Ao final, as informações de *tags* correspondentes são combinadas e extraídas. Este sistema se baseia em evidências lingüísticas morfológicas.

Outros trabalhos utilizam conceitos da Programação Lógica Indutiva (ILP) para a resolução de problemas em PLN (Jorge & Lopes, 2000; Junker, Sintek & Rinck, 1999; Lopes & Brazdil, 1998). Algumas tarefas de Língua

Natural que empregam o aprendizado relacional incluem: o aprendizado de *part-of-speech tagging*, o aprendizado de relações semânticas e o aprendizado no contexto de tradução de máquina (Cussens & Džeroski, 2000). As principais vantagens da utilização de abordagens relacionais em PLN são:

1. as regras induzidas por sistemas relacionais são compreensíveis por linguistas,
2. sistemas relacionais permitem facilmente integrar algum conhecimento de fundo lingüístico na definição do problema, e
3. utilização de uma representação mais expressiva (baseada na linguagem de primeira ordem) para a linguagem de hipóteses e do conhecimento de fundo.

Diversos trabalhos utilizam o aprendizado relacional para auxiliar na construção de sistemas de extração de informação. Tais trabalhos utilizam o aprendizado somente como parte de um grande sistema de EI e necessitam mais interação humana do que simplesmente fornecer textos com *templates* preenchidos. CRISTAL cria um dicionário de padrões de extração generalizando padrões identificados no texto por um especialista (Soderland et al., 1995). AUTOSLOG cria um dicionário de padrões de extração especializando um conjunto de padrões sintáticos gerais e assume que posteriormente um especialista filtre os padrões produzidos (Riloff, 1993). PALKA aprende padrões de extração contando com um conceito hierárquico para guiar as generalizações e especializações (Kim & Moldovan, 1995). Esses sistemas contam anteriormente com uma etapa de análise de sentenças para identificar elementos sintáticos e seus relacionamentos, e necessitam de um processamento adicional para produzir ao final os *templates* preenchidos.

O sistema RAPIER (Califf & Mooney, 2003) (*Robust Automated Production of Information Extraction Rules*), diferentemente dos anteriores, aprende regras para a tarefa completa de extração de informação, tais regras extraem diretamente as informações relevantes dos documentos sem necessitar anteriormente de uma análise sintática das sentenças e também de realizar algum pós-processamento. O seu algoritmo de aprendizado incorpora técnicas vários sistemas de Programação Lógica Indutiva e basicamente consiste de uma busca (*bottom-up*) por padrões que caracterizam o texto. O RAPIER se inspirou nos seguintes sistemas: GOLEM (Muggleton & Feng, 1992), CHILLIN (Zelle & Mooney, 1994) e PROGOL (Muggleton, 1995).

Na Seção 3.5 são apresentados alguns dos principais trabalhos relacionados, especificamente, com a extração de informações (corpo e referências) de artigos científicos.

3.4 Métricas de Avaliação

Como já comentado, a necessidade por medidas de avaliação para o problema de extração de informação surgiu com as Conferências de Entendimento de Mensagens (MUC). Inicialmente essas medidas foram desenvolvidas baseando-se nas medidas de precisão e cobertura, da área de RI. No entanto, as definições das medidas de EI sofreram alterações em relação as usadas em RI, apesar dos nomes serem mantidos. Essas alterações permitiram considerar possíveis generalizações em EI onde, diferente de RI, dados não presentes na entrada podem ser erroneamente produzidos (Gaizauskas & Wilks, 1998).

O estudo realizado através das quatro primeiras MUC (Sundheim, 1992) forneceu a base para a definição das medidas de avaliação existentes.

Na tarefa de extração de informação, cobertura (*recall*) é definida como a quantidade de informações corretamente extraídas sobre todas as informações relevantes nos textos. Precisão (*precision*) é definida como a quantidade de informações corretamente extraídas sobre todas as informações extraídas. Portanto, cobertura refere-se a quanto de informação relevante foi corretamente extraída, enquanto precisão refere-se a confiança da informação extraída. Em função do *template* da extração, precisão (P) e cobertura (C) são definidos, respectivamente, na Equação 3.1 e Equação 3.2:

$$P = \frac{N_c}{N_p} \tag{3.1}$$

$$C = \frac{N_c}{N_t} \tag{3.2}$$

onde, N_c é o número de *slots* que foram corretamente preenchidos pelo sistema, N_p é o número total de *slots* que foram preenchidos pelo sistema e N_t é o número total de *slots* que deveriam ser preenchidos pelo sistema (as informações existem no texto). Estas medidas são inversamente relacionadas, isto é, quando ocorre um aumento na cobertura, a precisão tende a

diminuir e vice-versa.

Na tentativa de avaliar um sistema de EI levando em consideração a cobertura e a precisão, pode-se utilizar uma outra medida chamada *F-measure*, que combina as medidas anteriores, apresentada na Equação 3.3:

$$F - measure = \frac{(\beta^2 + 1) * C * P}{\beta^2 * (C + P)} \tag{3.3}$$

onde, o parâmetro β quantifica a preferência da cobertura sobre a precisão. Freqüentemente é usado $\beta = 1$ (Equação 3.4) com o propósito de avaliar sistemas de EI balanceando as duas medidas.

$$F_1 = \frac{2 * C * P}{C + P} \tag{3.4}$$

Neste trabalho são utilizadas as métricas de avaliação precisão, cobertura e *F-measure* (em particular F_1), como forma de estimar a qualidade da extração de informações de artigos científicos.

Para exemplificar o cálculo da precisão e cobertura, considere o exemplo de extração das informações de um documento representado na Figura 3.2. O documento, do qual são extraídos os autores e seus e-mails, corresponde a primeira página de um artigo científico. No exemplo, o sistema preenche corretamente dois *slots* do total de quatro *slots* relevantes, dessa forma a cobertura do sistema é igual a 2/4 (50%). Entretanto, o sistema preenche erroneamente o *slot* referente ao e-mail do primeiro autor ({belend, pedro}@sip.ucm.es) e a sua precisão é igual a 2/3 (0.667%).

Existem algumas considerações importantes com relação a avaliação em IE:

- o sistema pode atribuir algum crédito para uma informação parcialmente extraída, ao invés de considerá-la irrelevante. Neste trabalho, entretanto, adotou-se a abordagem mais conservadora, ou seja, uma informação apenas é considerada relevante quando for corretamente extraída. Por exemplo, se o *slot* título de um artigo é "*Integrated Case-Based Building Design*" e o sistema extrai apenas "*Integrated Case-Based Building*", isto contará como sendo uma extração incorreta;

- para a contagem de extrações corretas e erros, o sistema pode requerer que sejam extraídas todas as ocorrências (*all slot occurrences* - ASO)

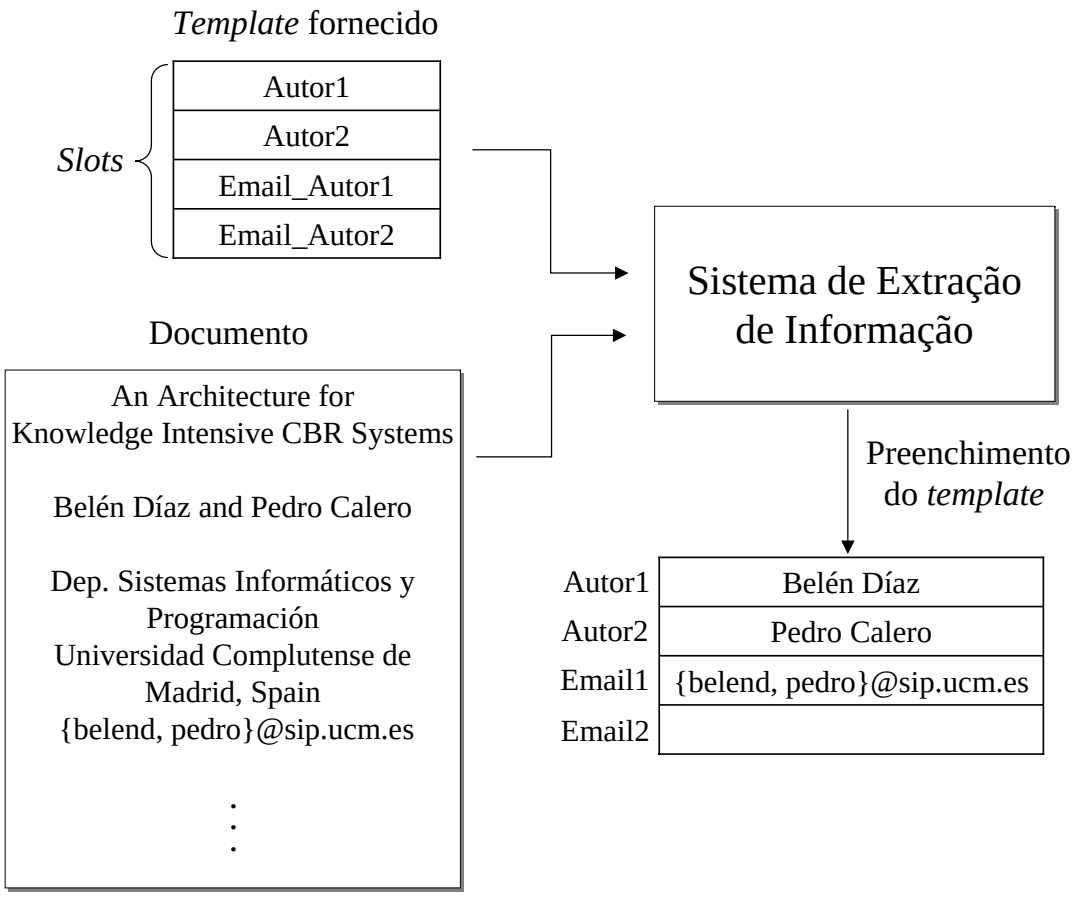


Figura 3.2. Exemplo de extração das informações de um documento

de um *slot* para obter crédito máximo. Assim, caso um documento contenha um *slot* com duas ocorrências, por exemplo 10am e 10:00, então o sistema deve extrair as duas;

- uma alternativa é considerar válido extrair 10am ou 10:00, visto que as duas referem a mesma entidade (*one slot occurrences* - OSO). A avaliação OSO foi empregada neste trabalho, pois sabe-se a priori que os *slots* de um artigo científico, na maioria dos casos, referem a um único valor.

3.5 Extração de Informações de Artigos Científicos

Por agir de acordo com padrões implícitos nos textos, um sistema de extração de informações geralmente é criado para atuar sobre um tipo específico de texto, por exemplo, documento Web ou publicações científicas como

é o caso deste projeto, visando desse modo tratar tanto aspectos gerais do texto como exceções. Entretanto, ainda não é possível obter resultados perfeitos (Cowie & Lehnert, 1996). Essa especificidade torna-se um problema, pois passa a ser impossível desenvolver um sistema genérico, que analise vários tipos de textos, automaticamente; outro problema enfrentado está relacionado com as várias formas de se representar uma mesma informação, além de possíveis erros.

A extração e estruturação dessas informações a partir de artigos científicos deparam-se com diversos desafios relacionados com a identificação, no texto, de cada uma das informações abaixo:

1. título;
2. autor(es);
3. afiliação (instituição, e-mail, etc);
4. resumo (*abstract*);
5. palavras chaves;
6. referências bibliográficas
   (título, autor(es), ano de publicação, evento, local, editores, etc).

Representing Knowledge for Case-Based Reasoning: The ROCADE System	<TÍTULO>
Béatrice Fuchs1 and Alain Mille2	<AUTOR1>, <AUTOR2>
1 Université Lyon III, IAE-Modeme 15 quai Claude Bernard, 69 007 Lyon, France 2 Université Lyon I, LISI Bât. 710, 43 bd du 11 novembre 1918, 69 100 Villeurbanne	<INSTITUIÇÃO-AUTOR1>, <INSTITUIÇÃO-AUTOR2>,
fuchs@univ-lyon3.fr amille@bat710.univ-lyonl.fr	<EMAIL-AUTOR1>, <EMAIL-AUTOR2>
CBR KNOWLEDGE REPRESENTATION . . .	<PALAVRA-CHAVE1> <PALAVRA-CHAVE2> . . .
Althoff et al., 1995b. Althoff k.-D., Auriol, E., Bergmann, R., Breen, S., Dittrich, S., Johnston, R., Manago, M., Traphoner, R., and Wess, S. (1995b). Case-Based Reasoning for Decision Support and Diagnostic Problem Solving: the INRECA Approach. In Proc. of the 3rd Workshop of the German Special Interest Group on CBR at the 3 German Expert System Conference	<REFERÊNCIA>
Armengol and Plaza, 1994. Armengol, E. and Plaza, E. (1994). A Knowledge Level Model of Case-Based Reasoning. In Richter, M. M., Wess, S., Althoff K.-D, and Maurer, F., editors, First European Workshop on Cased-Based Reasoning - EWGBR-93, pages 53-64, Kaiserslautern, Germany. LNAI, vol. 837, Springer, Berlin.	<REFERÊNCIA>
. . .	<REFERÊNCIA>[2]

Tabela 3.1. Estruturação de informações presentes em artigos científicos

[2]Essa referência, por sua vez, também pode ser estruturada (Seção 4.1 no Capítulo 4).

O processo de extração e estruturação das informações contidas em artigos, geralmente, não é trivial, um conjunto de regras deve ser criado de tal forma que se identifique, por exemplo, onde termina o título; onde começa e termina o nome do primeiro autor; tratamento de caracteres especiais, por exemplo, índice de nota de rodapé no nome dos autores; o fato de haver artigos escritos por um único autor e outros, por vários; etc. Deve-se ainda levar em consideração o fato de muitas vezes ser necessário, ao programa, uma adaptação da regra utilizada. Por exemplo, a maneira como extrair e-mails (compare os artigos da Tabela 3.1 e Figura 3.2), para mais de um autor, quando e-mails possuem o mesmo domínio, geralmente são informados deixando entre chaves os nomes dos usuários. Essas variações também podem ocorrer com as outras informações.

O tratamento das referências é um problema mais complexo, devido a uma grande variação que ocorre em sua estrutura. As informações nas referências podem ocorrer de várias formas e alguns desses itens não possuem posição fixa. As informações geralmente são delimitadas por " " ( ) [ ] , . ; : - ou algum outro caractere. Alguns formatos de referências são apresentados na Figura 3.3.

A. Aamodt. Knowledge intensive case-based reasoning and sustained learning. In *Proceedings of the ninth European Conference on Artificial Intelligence*, Stockholm, August, 1990.
Kolodner J.L : "An Introduction to Case-Based Reasoning", *Artificial Intelligence Review*, vol. 6, No. 1, 1992, pp 3-34.
[Koton, 1988] Phyllis Koton. *Using Experience in Learning and Problem Solving*. PhD thesis, MIT, 1988.

Figura 3.3. Diferentes formatos de referências

Atualmente na FIP, o processo de extração de informações dos artigos científicos é feito utilizando um conjunto de regras geradas manualmente. Tais regras foram geradas analisando os artigos (arquivos textos) com a finalidade de descobrir padrões sintáticos existentes. O corpus utilizado para a geração das regras foi construído digitalizando os artigos via scanner e aplicando um software de OCR, este procedimento acarretou o surgimento de alguns problemas tais como: presença de novos caracteres (erros), mudança de caracteres (onde era vírgula passou a ser ponto e vírgula) e surgimento de linhas em branco. Esses e outros fatores dificul-

tam a construção de um conjunto de regras de extração, por um processo manual, aumentando a taxa de erro na identificação das informações no conjunto de artigos.

Diversos trabalhos relacionados com a extração de informações de artigos foram propostos na literatura. Em geral esses métodos se enquadram em duas categorias: abordagem baseada em regras e abordagem baseada em Aprendizado de Máquina.

Baseada em Regras Giuffrida, Shek & Yang (2000), por exemplo, desenvolveram um sistema baseado em geras para extrair informações do cabeçalho[3] de artigos científicos, em formato PostScript. Foram utilizadas regras baseadas na análise da natureza espacial dos artigos, tal como "*títulos* são freqüentemente localizados na região superior da primeira página e possuem o maior tamanho de fonte". Ding, Chowdhury & Foo (1999) utilizaram uma técnica chamada mineração de *templates* para extrair importantes informações de artigos e, também, de suas referências. Ding, Chowdhury & Foo usaram um *template* para a extração de informações do corpo do artigos e outros três *templates* para a extração das referências bibliográficas, e obtiveram um resultado satisfatório. Contudo, as referências analisadas pertenciam a um único estilo. Day et al. (2005) propuseram um método baseado em conhecimento para extrair informações de referências; foi adotado uma ontologia denominada INFOMAP (estrutura de representação de conhecimento), capaz de extrair informações de seis estilos de referências bibliográficas com uma alta precisão. As principais diferenças entre a abordagem proposta neste trabalho e INFOMAP são: (1) INFOMAP trabalha apenas com um tipo de referência bibliográfica (artigos de revistas), enquanto que a abordagem proposta lida com vários (ver Seção 4.1.1 no Capítulo 4), (2) INFOMAP possui sete informações (*author, title, journal, volume, number* (*issue*), *year* e *pages*), que devem ser extraídas dos artigos, ao passo que a abordagem proposta trabalha com um conjunto bem maior de informações (ver Apêndice A) e (3) o conhecimento de interesse é representado em INFOMAP, manualmente, através de uma ferramenta de edição apropriada; a abordagem proposta, no entanto, vale-se de um processo semi-automático de aprendizagem de regras (conhecimento) para extrair informações de referências.

[3] O cabeçalho (Seymore, McCallum & Rosenfeld, 1999) é o conjunto todas as palavras do início do artigo até a primeira seção (normalmente a *introdução*) ou até o final da primeira página, a que ocorrer primeiro.

Mao, Kim & Thoma (2004) descreveram um sistema que utiliza um conjunto de características geométricas e contextuais de artigos médicos, para realizar a extração de informações presente no cabeçalho dos artigos. Melo, Secato & Lopes (2003) trabalharam na extração e principalmente na identificação automática das referências bibliográficas, o trabalho apresentou bons resultados na identificação, porém era inicialmente computacionalmente muito custoso. Algum tempo depois esse problema foi estudado e resolvido (Melo & Lopes, 2005; Melo & Lopes, 2004a).

Aprendizado de Máquina Connan e Omlin treinaram *Hidden Markov Models* (HMM) para reconhecer quando uma referência foi gerada por um dos seguintes estilos bibliográficos AAAI, NEWAPA, IEEE. A correta identificação do estilo possibilita que as informações (autores, título, editora, ano, etc.) sejam extraídas com precisão de até 97% (Connan & Omlin, 2000). Yin et al. (2004) aplicaram *bigram* HMM ao problema de extrair informações de referências com vários estilos, ou seja, sem o conhecimento a priori de qual estilo bibliográfico foi utilizado. A estrutura e os parâmetros do modelo HMM são automaticamente aprendidos a partir de exemplos de treinamento[4], e o sistema é capaz de alcançar uma precisão global acima de 90%. Takasu propôs um modelo estatístico chamado *Dual and Variable-length output Hidden Markov Model* (DVHMM) para a extração de atributos de referências na língua japonesa, capturadas utilizando um software OCR (Takasu, 2003). O modelo gerado tem a vantagem de representar a estrutura sintática das referências e os padrões de erros causados pelo OCR. O AUTOBIB é um outro trabalho de extração e integração de informações bibliográficas de artigos recuperados na Web (Geng & Yang, 2004). O AUTOBIB utiliza-se de um pequeno banco de dados de referências já estruturadas para prover um conjunto de treino para um *parser* baseado em HMM. O processo de extração é praticamente automático e utiliza referências contidas em páginas HTML. Seymore, McCallum & Rosenfeld (1999) também utilizaram HMM para extrair informações importantes do cabeçalho de artigos científicos em computação e alcançaram uma precisão global de 92,9%. Seymore, McCallum & Rosenfeld definiram 15 informações (classes) que podem ocorrer em um cabeçalho de um documento.

[4] O corpus de treino é um conjunto aleatório de 250 artigos, capturados da Web, em formato PDF.

Han et al. (2003) lidaram com o problema de extração de informações de artigos como sendo um problema de classificação. A abordagem proposta utiliza *Support Vector Machines* (SVM) para classificar cada linha do cabeçalho em uma ou mais de 15 classes. Eles utilizaram, principalmente, informações lingüísticas para os atributos e também algum conhecimento a priori do domínio. O método foi inicialmente proposto para melhorar a qualidade da extração dos repositórios Citeseer (Lawrence, Bollacker & Giles, 1999) e eBizSearch (Petinot et al., 2003). Peng & McCallum (2004) empregaram *Conditional Random Fields* (CRF) para extrair várias informações do cabeçalho e das referências de artigos científicos. O corpus de referências utilizado foi criado pelo projeto Cora (McCallum et al., 2000), o qual contêm 500 referências categorizadas em 13 elementos: *author, title, editor, booktitle, date, journal, volume, tech, institution, pages, location, publisher* e *note*.

Capítulo 4

Extração de Informação de Artigos Científicos

Neste capítulo são descritos os procedimentos empregados neste trabalho para a identificação e extração das informações dos artigos contidos em um corpus dado. Existem dois procedimentos principais, um para a extração das informações contidas em cada uma das referências de um artigo (Seção 4.1), e outro para a extração das informações de interesse presentes no corpo dos artigos (Seção 4.2). Antes desses procedimentos, é necessário uma etapa de aprendizagem, que no caso das referências, consiste principalmente no aprendizado das regras de etiquetagem (induzidas automaticamente) e, no caso do corpo, nas regras de extração de cada informação (induzidas manualmente). Essa estapa de aprendizagem também é apresentada neste capítulo. Também são comentadas a construção semi-automática de um corpus de referências etiquetado que foi usado no início dos experimentos de indução do etiquetador e de um corpus maior de referências etiquetadas gerado a partir de informações contidas em arquivos BibTEX(com cerca de um milhão de termos)[1].

4.1 Extração de Informação de Referências Bibliográficas

A indução automática de regras para a extração de informações de referências baseia-se no mapeamento descrito na Seção 2.3.4 na página 29.

[1] A ser disponibilizado em `http://icmc.usp.br/~alneu`

Nas próximas seções são brevemente apresentadas as definições de projeto, o pré-processamento realizado e o processo semi-automático de construção do corpus de treino para a indução de um etiquetador que etiquetará o conjunto de referências para o processo de extração.

4.1.1 Definições de Projeto

Inicialmente, foi necessário adotar algumas decisões de projeto para a realização dos experimentos. É importante ressaltar, que essas decisões foram tomadas de acordo com as necessidades da FIP, comentadas brevemente na introdução desta dissertação.

Para a definição do conjunto de etiquetas das referências bibliográficas foi necessário verificar quais são as informações que constituem uma referência. Um estudo sobre as principais normas (padrões) para a elaboração de referências bibliográficas foi realizado e optou-se por utilizar, neste trabalho, as normas técnicas definidas pela NBR6023 da ABNT (Associação Brasileira de Normas Técnicas, 2002), as normas do *Chicago Manual of Style Guide* baseadas no *Chicago Manual Style* (Chicago Editorial Staff, 1993) e as especificações técnicas da *American Psychological Association Publication Manual* (American Psychological Association, 1994). O conjunto de etiquetas também foi baseado no BibTEX, que é um formato para referências bibliográficas usado em combinação como o sistema de processamento de texto TEX/LATEX[2].

Esses padrões basicamente são compostos por tipos de referências (livros, artigos, revistas, eventos, etc.) e as informações que devem ser preenchidas em cada tipo, divididas em obrigatórias e opcionais. Neste trabalho não houve distinção entre tais informações, visto que a FIP deve ser capaz de lidar com todos os tipos de informações e, conseqüentemente, com todos os tipos de referências. Os tipos de referências bibliográficas, juntamente com a suas etiquetas, considerados neste projeto são descritos a seguir.

- ARTIGO
  - AUTHOR: *autor(es) do artigo;*
  - TITLE: *título do artigo;*
  - JOURNAL: *revista que publicou o artigo;*

[2] TEX/LATEX é, de fato, um padrão para publicações em várias áreas de pesquisa.

  - YEAR: *ano de publicação;*
  - VOLUME: *volume da revista;*
  - NUMBER: *número da revista;*
  - INITPAGE: *página inicial do artigo na revista;*
  - FINALPAGE: *página final do artigo na revista;*
  - MONTH: *mês de publicação;*
  - URL: *link para o arquivo (pdf, ps);*
  - URLACCESSDATE: *data em que a url foi acessada;*
  - NOTE: *alguma informação adicional;*
  - ISSN: *numeração internacional para publicações periódicas*

- LIVRO
  - AUTHOR: *autor(es) do livro;*
  - EDITOR: *editor(es) do livro;*
  - TITLE: *título do livro;*
  - PUBLISHER: *editora do livro;*
  - YEAR: *ano de publicação;*
  - VOLUME: *volume do livro;*
  - NUMBER: *número do livro;*
  - SERIES: *série que publicou o livro;*
  - ADDRESS: *endereço completo, ou apenas a cidade, da editora;*
  - EDITION: *edição do livro;*
  - MONTH: *mês de publicação;*
  - NOTE: *alguma informação adicional;*
  - ISBN: *numeração internacional para livros*

- CAPÍTULO DE LIVRO
  - AUTHOR: *autor(es) do livro;*
  - EDITOR: *editor(es) do livro;*
  - TITLE: *título do capítulo;*
  - BOOKTITLE: *título do livro;*
  - CHAPTER: *número do capítulo;*

  - INITPAGE: *página inicial;*
  - FINALPAGE: *página final;*
  - PUBLISHER: *editora do livro;*
  - YEAR: *ano de publicação;*
  - VOLUME: *volume do livro;*
  - NUMBER: *número do livro;*
  - SERIES: *série que publicou o livro;*
  - TYPE: *algo sobre o tipo do documento, por ex.: a seção;*
  - ADDRESS: *endereço completo, ou apenas a cidade, da editora;*
  - EDITION: *edição do livro;*
  - MONTH: *mês de publicação;*
  - NOTE: *alguma informação adicional;*
  - ISBN: *numeração internacional para livros*

- CONGRESSOS, CONFERÊNCIAS, SIMPÓSIOS, WORKSHOPS
  - EDITOR: *editor(es) do evento;*
  - TITLE: *título do evento;*
  - PUBLISHER: *editora do evento;*
  - YEAR: *ano de evento;*
  - SERIES: *série do evento;*
  - ORGANIZATION: *órgão patrocinador do evento;*
  - ADDRESS: *endereço completo, ou apenas a cidade, da editora;*
  - MONTH: *mês de publicação;*
  - NOTE: *alguma informação adicional*

- TRABALHOS APRESENTADOS EM EVENTOS
  - AUTHOR: *autor(es) do trabalho;*
  - TITLE: *título do trabalho;*
  - BOOKTITLE: *título do evento;*
  - YEAR: *ano de publicação;*
  - EDITOR: *editor(es) do trabalho;*
  - SERIES: *série do evento;*

  - INITPAGE: *página inicial;*
  - FINALPAGE: *página final;*
  - PUBLISHER: *editora;*
  - ORGANIZATION: *órgão patrocinador do evento;*
  - ADDRESS: *endereço completo, ou apenas a cidade, da editora;*
  - MONTH: *mês de publicação;*
  - URL: *link para o arquivo (pdf, ps);*
  - URLACCESSDATE: *data em que a url foi acessada;*
  - NOTE: *alguma informação adicional;*
  - ISSN: *numeração internacional para publicações periódicas*;
  - ISBN: *numeração internacional para livros*

- DISSERTAÇÕES E TESES
  - AUTHOR: *autor da monografia;*
  - TITLE: *título da monografia;*
  - SCHOOL: *universidade no qual a monografia foi escrita;*
  - YEAR: *ano de publicação;*
  - TYPE: *algo que indique se é uma dissertação ou tese;*
  - ADDRESS: *departamento no qual a monografia foi escrita;*
  - MONTH: *mês de publicação;*
  - URL: *link para o arquivo (pdf, ps);*
  - URLACCESSDATE: *data em que a url foi acessada;*
  - NOTE: *alguma informação adicional;*
  - ISBN: *numeração internacional para livros*

- RELATÓRIO TÉCNICO
  - AUTHOR: *autor do relatório;*
  - TITLE: *título do relatório;*
  - INSTITUTION: *instituição no qual o relatório foi escrito;*
  - YEAR: *ano de publicação;*
  - TYPE: *algo que indique se é um relatório técnico;*
  - NUMBER: *número do relatório;*

  - ADDRESS: *departamento no qual o relatório foi escrito;*
  - MONTH: *mês de publicação;*
  - NOTE: *alguma informação adicional*

- MÍDIA (CD-ROM)
  - AUTHOR: *autor da mídia;*
  - TITLE: *título da mídia;*
  - PUBLISHER: *editora;*
  - ADDRESS: *endereço completo, ou apenas a cidade, da editora;*
  - NOTE: *alguma informação adicional*

- PÁGINA DE INTERNET
  - AUTHOR: *autor da página;*
  - TITLE: *título da página;*
  - URL: *link para a página;*
  - URLACCESSDATE: *data em que a url foi acessada;*
  - NOTE: *alguma informação adicional*

Uma descrição do conjunto de etiquetas criado, denominado FIP tagset, é dado na Apêndice A. É importante destacar que mesmo elaborando-o de forma cuidadosa, o FIP tagset sofreu alterações durante o processo de etiquetagem manual do corpus. Uma das modificações foi a inclusão de uma etiqueta, nominada INDICATOR, que engloba todos os termos que indicam a presença de informações, por exemplo: pp, In, No, *pages*, etc.

4.1.2 O Pré-processamento

O corpus utilizado nos experimentos possui referências bibliográficas da área de Computação e abrange os estilos bibliográficos Plain, Alpha, Abbrv, Apalike, Chicago, entre outros. Na Seção 4.1.3, uma descrição completa do processo de construção e etiquetagem do corpus é apresentada. Inicialmente, para que o corpus pudesse ser utilizado, vários ajustes relacionados com a tarefa de pré-processamento foram realizados:

- correção de erros provenientes do software[3] de conversão de arquivos no formato PDF ou PS para o formato TXT;
- remoção do excesso de espaços em branco ou tabulações;
- padronização de um conjunto de caracteres, por ex.: todos os caracteres similares ao caractere hífen, foram substituídos pelo mesmo;
- tokenização do corpus;
- substituição da barra (/) pelo símbolo $b, pois o etiquetador TBL utiliza a barra para separar a palavra de sua etiqueta;
- formatação do corpus resultando em uma referência completa por linha

O software na maioria dos casos realiza perfeitamente a conversão, entretanto em algumas situações podem ocorrer alguns erros, por ex.: substituição do conjunto de caracteres "ffi" por Æ. A idéia de padronizar um conjunto de caracteres, para a tarefa de etiquetagem/extração, surgiu das seguintes observações:

- um corpus que contenha uma grande variedade[4] de sinais de pontuação poderia dificultar a tarefa do etiquetador, e
- a padronização de caracteres não afetaria o resultado do sistema de extração, ou seja, as informações que deveriam ser extraídas, com ou sem padronização, seriam as mesmas

Antes da etapa de pré-processamento, cada referência corresponde a várias linhas consecutivas no corpus. Para identificar uma referência completa, foram criados vários modelos (padrões), a partir da análise do corpus, que informam quando uma linha é início/fim de uma referência. A grande maioria das referências no corpus possue indicadores de início de referência, o que, conseqüentemente, facilita a criação dos modelos de identificação. Esses indicadores normalmente assumem os formatos apresentados na Tabela 4.1.

[3]Para a conversão dos arquivos foram utilizados os comandos *pdftotext* versão 3.0 (aplicativo linux) e *pstotext* versão 1.9 (aplicativo windows).

[4]O corpus usado nos experimentos, por exemplo, possui caracteres similares aos sinais de pontuação hífen, aspas simples e aspas duplas.

número. *corpo da referência*
[número]. *corpo da referência*
[conjunto de caracteres alfanuméricos, ano] *corpo da referência*
(conjunto de caracteres alfanuméricos, ano) *corpo da referência*

Tabela 4.1. Alguns formatos de início de referências

No entanto, para algumas referências que não foi possível criar modelos, foram criadas heurísticas com o intuito de identificá-las. Algumas das heurísticas utilizadas são apresentadas a seguir.

1. Uma linha do documento não é início de referência quando possui tamanho menor que 40 e não contém as palavras http e ftp. No entanto, uma referência muito raramente pode ser composta apenas por um *link* de tamanho pequeno.

2. Uma linha que começa por uma preposição não é início de referência.

3. Caso uma linha termine por uma preposição, a linha seguinte não é início de referência.

4. Uma linha que começa por uma letra minúscula não é início de referência.

5. Uma linha é inicio de referência quando contém um ano entre parênteses.

Ao observar as heurísticas acima é possível notar que existe uma maior facilidade em criar regras para determinar quando uma linha do corpus não é início de referência. Vale ressaltar também que a ordem na qual as heurísticas são utilizadas influenciam o resultado final, isto é, dependendo da ordem de execução destas, algumas referências, antes identificadas corretamente, podem ser erroneamente identificadas.

Ao final desta etapa constatou-se que alguns erros ainda não haviam sido corrigidos, isso se deve principalmente a grande quantidade de padrões e formatos, e também por existirem algumas referências que possuem estruturas raras, tal como uma tabela. Foi realizada mais uma inspeção no corpus e vários erros foram corrigidos manualmente. Acredita-se que ainda possam existir alguns erros, porém em uma porcentagem bem pequena.

4.1.3 A Construção dos Corpus de Referências Bibliográficas Etiquetados

Antes de realmente iniciar os experimentos com o etiquetador TBL é necessário ter um corpus etiquetado, razoavelmente grande[5], que contenha uma grande variedade de modelos. O corpus utilizado nos experimentos é composto por uma pequena parcela etiquetada manualmente e por outra automaticamente etiquetada utilizando informações contidas em arquivos BibTEX. Como comentado anteriormente, o corpus possui referências bibliográficas da área de Computação e abrange os estilos bibliográficos Plain, Alpha, Abbrv, Apalike, Chicago, principalmente. Algumas informações consideradas relevantes sobre o corpus são apresentadas na Tabela 4.2.

	Estilo bibliográfico	N. de palavras	N. de referências	Etiquetagem
1	Plain	215726	5000	Automática
2	Alpha	267679	5000	Automática
3	Abbrv	219061	4996	Automática
4	Chicago	220810	4993	Automática
5	Apalike	177326	3992	Automática
6	Aleatório	34384	947	Manual

Tabela 4.2. Informações sobre o corpus

A seguir é descrito em detalhes o processo de construção e etiquetagem automática de grande parte do corpus (linhas 2 a 6 na Tabela 4.2), assim como o processo de etiquetagem manual de uma pequena parcela do corpus.

Corpus Etiquetado a partir de Arquivos BibTEX

Um arquivo com extensão .BIB, chamado de base BibTEX, apresenta informações de referências bibliográficas separadas por campos, denominados campos BibTEX. A partir de um arquivo .BIB etiquetado (as informações de cada campo estão etiquetadas) é possível criar um conjunto de referências etiquetadas. Posteriormente, o procedimento utilizado para a etiquetagem de arquivos .BIB é apresentado.

Para a utilização do BibTEX é necessário especificar dois parâmetros: o estilo bibliográfico (arquivo .BST) e a base (arquivo .BIB). Atualmente, existe uma grande variedade de estilos bibliográficos para o BibTEX disponíveis na

[5]Na Figura 4.2 pode ser observada a relação do tamanho do corpus e a precisão do etiquetador.

WWW. Optou-se por utilizar neste trabalho os estilos citados anteriormente, disponibilizados pela CTAN[6], pois estão entre os estilos mais utilizados pela comunidade científica. Os estilos Plain, Alpha, Abbrv e Unsrt são os chamados *estilos bibliográficos padrão* que serviram de base para a criação de todos os outros estilos existentes. O estilo Unsrt não foi utilizado por ser como o Plain, exceto que as referências não são alfabeticamente ordenadas.

A base utilizada foi recuperada do repositório *The Collection of Computer Science Bibliographies*[7]. *The Collection of Computer Science Bibliographies* é uma coleção de literaturas científicas disponíveis, abrangendo os principais tópicos em Ciência da Computação. A base utilizada é composta por vários arquivos .BIB de diferentes tamanhos.

O procedimento para a construção de um conjunto de referências automaticamente etiquetadas é descrito a seguir. Até o presente momento, o BibTEX em sua implementação aceita no máximo 5000 citações, isto é, um arquivo .BIB deve possuir no máximo 5000 entradas. As entradas são os possíveis tipos de referências, por exemplo *article*, *book*, *inbook*, *inproceedings*, etc. Dentre os vários arquivos .BIB da base foram selecionados cinco[8], um para cada estilo, cada um com no máximo 5000 entradas. Em seguida é utilizado o programa JabRef[9] (versão 1.8.1) para constatar se existem erros sintáticos e inconsistências nos arquivos .BIB selecionados.

Após a definição e ajustes dos arquivos .BIB, o próximo passo é a tarefa de etiquetagem propriamente dita. Para uma melhor compreensão é ilustrado na Figura 4.1, um exemplo da etiquetagem da entrada INCOLLECTION contida em um arquivo chamado 00.bib da base.

Conforme observado neste exemplo, a etiquetagem consiste em etiquetar cada campo (*author*, *title*, *booktitle*, *publisher*, etc.) de acordo com as suas informações, delimitadas por chaves, com o rótulo correspondente ao campo. Algumas informações por terem um significado especial para o BibTEX não são etiquetadas, por exemplo a palavra 'and' localizada no campo author ou editor não deve ser etiquetada. Finalizando, para construir um conjunto de referências automaticamente etiquetadas basta apenas selecionar um arquivo BibTEX etiquetado e um estilo bibliográfico, e utilizar o BibTEX em combinação com o LATEX para gerar um documento que

[6] http://ctan.org/

[7] http://liinwww.ira.uka.de/bibliography/index.html

[8] Inicialmente um arquivo com mais de 5000 entradas é selecionado e, posteriormente, através de uma inspeção manual, seu número é reduzido para próximo de 5000.

[9] http://jabref.sourceforge.net/

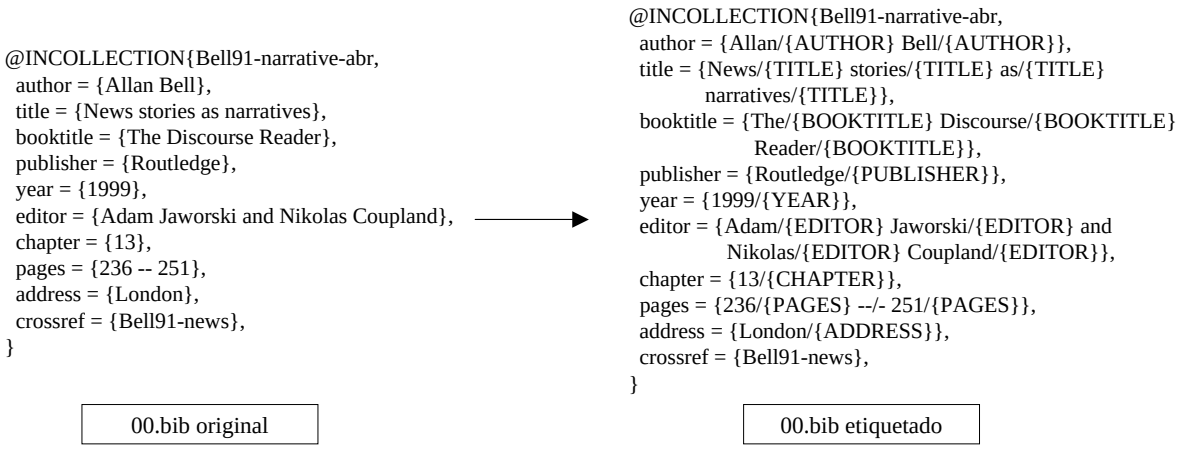


Figura 4.1. Exemplo da etiquetagem de um arquivo .BIB

contenha todas as suas referências corretamente etiquetadas. Utilizando o método acima descrito, foi possível construir grande parte do corpus etiquetado.

Corpus Etiquetado Semi-automáticamente

Como comentado anteriormente, uma pequena parcela do corpus foi etiquetada manualmente. Essa fração do corpus foi inicialmente recuperada pelo *WebMiner* (Brasil & Lopes 2004) e possui referências bibliográficas das áreas de Recuperação de Informação e Processamento de Língua Natural. Em trabalhos sobre etiquetagem morfossintática, nos quais inicialmente não exista um corpus etiquetado (*gold standard*), normalmente realiza-se um extenso trabalho manual. Existe também a possibilidade de utilizar uma ferramenta semi-automática de auxílio à etiquetagem. Infelizmente para este trabalho, no momento da etiquetagem dessa porção do corpus, não existia um corpus inicialmente etiquetado e nem uma ferramenta de auxílio à etiquetagem. No entanto, tal problema foi resolvido através de um processo iterativo e interativo de etiquetagem semi-automático[10], descrito sucintamente a seguir.

O processo para a construção de um corpus manualmente etiquetado consiste em: etiquetar manualmente uma pequena porção do corpus não-etiquetado; treinar o etiquetador TBL; utilizar o TBL treinado para etiquetar uma nova porção do corpus não-etiquetado; corrigir os erros do corpus previamente etiquetado (o qual deve ser consideravelmente mais rápido do que

[10]Este processo de construção de um corpus manualmente etiquetado foi sugerido por Eric Brill, na documentação do seu etiquetador.

etiquetar manualmente esta porção, desde o início); treinar o etiquetador sobre todo o corpus etiquetado disponível e repetir este processo até o etiquetador mostrar-se suficientemente preciso. Esse processo de etiquetagem é melhor descrito no Algoritmo 1.

Algoritmo 1: Criação do Corpus Etiquetado

Entrada: $Tr, \bar{C}$

`/* Corpus inicial de treino etiquetado (100 referências), Corpus completo não etiquetado */`

Saída: C (Corpus Completo Etiquetado)

Início

`// Inicializa Corpus Completo Etiquetado`

$C \leftarrow Tr$;

`// Inicializa conjunto de regras de etiquetagem`

$Regras \leftarrow \emptyset$;

`// Induz primeiro conjunto de regras`

$Regras \leftarrow Induzir\_Etiquetador(C)$;

Repita

$Novas\_Ref \leftarrow$ 100 novas referências $\in \bar{C}$;

$\bar{C} \leftarrow \bar{C} - Novas\_Ref$;

$Etiquetar(Novas\_Ref, Regras)$;

$Novas\_Ref\_Corrigidas \leftarrow Novas\_Ref$ corrigidas manualmente ;

`// Determina` $Erro$ `na etiquetagem`

$C \leftarrow C \cup Novas\_Ref\_Corrigidas$;

$Regras \leftarrow Induzir\_Etiquetador(C)$;

Até $\bar{C} = \emptyset$;

return C ;

Fim

Utilizando o processo descrito acima foram etiquetadas e corrigidas manualmente um conjunto de 947 referências bibliográficas, em um total de 34384 palavras. O gráfico comparativo da precisão do etiquetador com linha de tendência durante esse processo de etiquetagem manual é apresentado na Figura 4.2. O etiquetador geralmente aumenta a sua precisão quando um corpus maior de treino é usado, porém para os conjuntos de treino com 500 e 700 referências, o etiquetador obteve um decréscimo em sua precisão. A justificativa para este fato decorre da grande quantidade de novas palavras e, principalmente, de novos modelos de referências, presentes no

corpus de teste. Entretanto, com uma precisão do etiquetador de no mínimo 74%, este processo de construção de um corpus etiquetado torna-se mais viável do que etiquetar manualmente termo a termo no corpus.

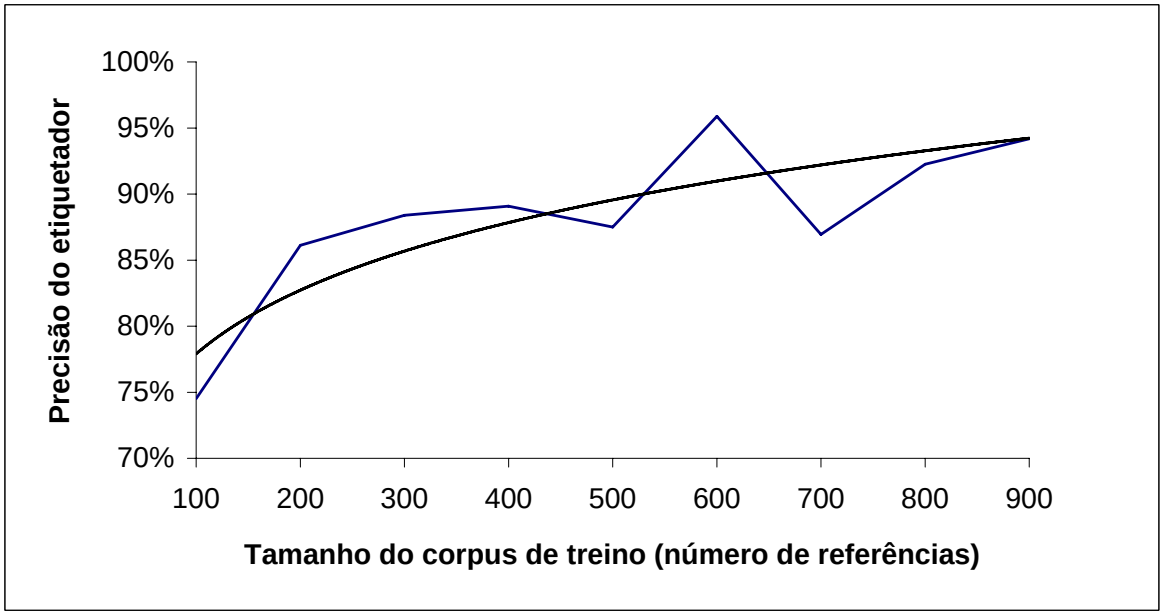


Figura 4.2. Gráfico comparativo de um processo interativo e iterativo de etiquetagem manual

4.1.4 A Extração das Informações de Referências Bibliográficas

Concluída a etapa de etiquetagem das referências, pode-se iniciar a extração de suas informações. A extração de elementos de referências bibliográficas, basicamente, consiste em concatenar informações de etiquetas correspondentes. Considera-se neste trabalho como etiquetas correspondentes, as etiquetas cujos os nomes são iguais ou sinais de pontuação que possuam algum significado para a informação a ser extraída, por exemplo o ponto que abrevia o nome de um autor ou o hífen que separa a página inicial da final de uma referência.

O produto final do processo de extração é um documento XML que contém as informações extraídas para cada referência. Para uma melhor compreensão é apresentado a seguir, um exemplo da extração de informações de uma referência etiquetada pelo TBL. Caso o programa receba como en-

trada a referência abaixo

Achermann/AUTHOR ,/, F/AUTHOR ./. and/AUTHOR Nierstrasz/AUTHOR ,/, O/AUTHOR ./. (/(2000c/YEAR)/) ./. Explicit/TITLE Namespaces/TITLE ./. In/INDICATOR Gutknecht/EDITOR ,/, J/EDITOR ./. and/EDITOR Weck/EDITOR ,/, W/EDITOR ./. ,/, editors/INDICATOR ,/, Modular/BOOKTITLE Programming/BOOKTITLE Languages/BOOKTITLE ,/, volume/INDICATOR 1897/
VOLUME of/SERIES LNCS/SERIES ,/, pages/INDICATOR 77/PAGES -/- 89/PAGES ,/, Zurich/ADDRESS ,/, Switzerland/ADDRESS ./. Springer/PUBLISHER -/- Verlag/PUBLISHER ./. URL/INDICATOR http/URL :/: $b/BARRA $b/BARRA www/URL ./. iam/URL ./. unibe/URL ./. ch/URL $b/BARRA ~/~ scg/URL $b/BARRA Archive/URL $b/BARRA Papers/URL $b/BARRA Ache00bExplicitNamespaces/URL ./. pdf/URL

irá combinar e extrair as suas informações, conforme descrito no seguinte código XML <ref>
<author>F. Achermann</author>
<author>O. Nierstrasz</author>
<year>2000</year>
<title>Explicit Namespaces</title>
<editor>J. Gutknecht</editor>
<editor>W. Weck</editor>
<booktitle>Modular Programming Languages</booktitle>
<volume>1897</volume>
<series>of LNCS</series>
<pages>77-89</pages>
<address>Zurich, Switzerland</address>
<publisher>Springer-Verlag</publisher>
<url>http://www.iam.unibe.ch/~scg/Archive/Papers/Ache00bExplicitNamespaces.pdf</url>
</ref>

A partir do exemplo acima, é possível observar algumas características do processo de extração: quando a referência possuir dois ou mais autores (ou editores), estes são extraídos em separado; os sinais de pontuação que estiverem entre o final de uma informação e início de outra são removidos; os termos da etiqueta INDICATOR são removidos, pois tais termos apenas indicam a presença de informações.

A adoção da XML (*eXtensible Markup Language*) neste trabalho, se deve pelos seguintes motivos:

1. a linguagem é um padrão aberto, flexível e facilmente expandível;
2. possibilita a transferência e manipulação de dados através da internet de modo fácil, rápido e consistente, de tal forma que qualquer tipo de aplicação, independentemente da plataforma, sistema operacional, ou linguagem em que foi construída consiga manuseá-los;
3. representa os dados de maneira hierárquica e organizada;
4. separa o conteúdo da apresentação, o que possibilita integrar e manipular dados de diversas fontes;
5. permite realizar buscas eficientes, pois os dados em um documento XML podem ser unicamente "etiquetados";
6. disponibilidade de várias ferramentas na Web que realizam a edição e análise (formatação/gramática) de documentos XML.

4.2 Extração de Informação do Corpo de Artigos Científicos

O corpus de artigos científicos usado nos experimentos é composto por 759 artigos em cinco diferentes áreas de pesquisa: Raciocínio Baseado em Casos (CBR), Programação Lógica Indutiva (ILP), Extração de Informação (EI), Recuperação de Informação (RI) e Processamento de Língua Natural (PLN). Os artigos das áreas de CBR e ILP foram manualmente obtidos das séries *Lecture Notes on Artificial Intelligence* (LNAI). Os outros artigos foram recuperados da Web através do WebMiner (Brasil & Lopes 2004) (áreas de RI e PLN) e da API do Google (área de EI).

Para a extração das informações presentes no corpo de artigos científicos foi adotada uma abordagem baseada em regras, a qual consiste em induzir um conjunto de regras de extração, com base na inspeção manual dos artigos. As regras induzidas consideram que as informações a serem extraídas (título, autores, afiliação, resumo, palavras chaves e bloco das referências bibliográficas) estão dispostas em uma determinada ordem nos artigos. Neste trabalho assume-se que as informações estão seqüencialmente organizadas na ordem[11]:

[11] As informações referentes ao conteúdo, propriamente dito, dos artigos (seções, tabelas, figuras, etc.) não são tratadas neste trabalho.

1. título
2. autor(res)
3. afiliação
4. resumo
5. palavras chaves
6. referências

Portanto, o sistema de extração não considera válido, por exemplo, extrair o resumo ou as palavras chaves, antes de extrair a afiliação, de um artigo. É importante ressaltar que a seqüência de informações apresentada é a mais freqüente no corpus.

As regras de extração foram induzidas, com base em uma análise detalhada dos padrões presentes nos artigos, de tal forma que se identifique corretamente onde começa e termina cada informação. Em outras palavras, o problema consiste em determinar quando uma linha no artigo (arquivo texto) corresponde ao início, ou final, de uma informação de interesse. Por exemplo, a seguir é apresentado um artigo, em formato textual, presente no corpus, cujo rótulo é AslamPaSa03.ps.txt.

```
1. A Unified Model for Metasearch and the Efficient
2. Evaluation of Retrieval Systems via the Hedge Algorithm
3. Javed A. Aslam, Virgiliu Pavlu, Robert Savell
4. Department of Computer Science
5. Dartmouth College
6. {jaa,virgilpavlu,rsavell}@cs.dartmouth.edu
7. ABSTRACT
8. We present a unified framework for simultaneously solving
9. both the pooling problem (the construction of efficient doc
10. ument pools for the evaluation of retrieval systems) and
11. metasearch (the fusion of ranked lists returned by retrieval
12. systems in order to increase performance). The implemen
13. tation is based on the Hedge algorithm for online learning,
14. which has the advantage of convergence to bounded error
15. rates approaching the performance of the best linear com
16. bination of the underlying systems. The choice of a loss
17. function closely related to the average precision measure of
```

```
18. system performance ensures that the judged document set
19. performs well, both in constructing a metasearch list and
20. as a pool for the accurate evaluation of retrieval systems.
21. Our experimental results on TREC data demonstrate excel
22. lent performance in all measures---evaluation of systems, re
23. trieval of relevant documents, and generation of metasearch
24. lists.
25. Categories and Subject Descriptors:
26. H.3.3 [Information Search and Retrieval]: Retrieval
27. Models.
28. General Terms: Algorithms, Theory.
29. Keywords: Metasearch, Pooling, Retrieval Systems.
30. 1. INTRODUCTION
31. In the annual TREC competition, participating systems
...
279. 4. REFERENCES
280. [1] G. V. Cormack, C. R. Palmer, and C. L. A. Clarke.
281. Efficient construction of large test collections. In Croft
282. et al. [2], pages 282--289.
283. [2] W. B. Croft, A. Moffat, C. J. van Rijsbergen,
284. R. Wilkinson, and J. Zobel, editors. Proceedings of the
285. 21th Annual International ACM SIGIR Conference on
286. Research and Development in Information Retrieval,
287. Melbourne, Australia, Aug. 1998. ACM Press, New
288. York.
289. [3] Y. Freund and R. E. Schapire. A decisiontheoretic
290. generalization of online learning and an application to
291. boosting. Journal of Computer and System Sciences,
292. 55(1):119--139, Aug. 1997.
```

As regras de extração, após serem aplicadas ao artigo AslamPaSa03.ps-.txt, são capazes de identificar e extrair corretamente o título (linha 1 a 2), os autores[12] (linha 3), a afiliação (linha 4 a 6), o resumo (linha 8 a 24), as palavras chaves (linha 29) e o bloco das referências (linha 280 a 292).

Alguns termos que possuem um significado especial ou mesmo linhas em branco, dependendo da localização no artigo, podem auxiliar para identificar cada informação. Na Tabela 4.3 são apresentadas algumas informações adicionais (conhecimento de fundo) que auxiliam na correta identifica-

[12]Neste trabalho os autores dos artigos são extraídos como sendo uma única unidade de informação, não ocorrendo, portanto, a identificação do primeiro autor, segundo autor, etc.

ção das informações a serem extraídas dos artigos.

Item de extração	Informações adicionais que auxiliam a sua identificação
título	a primeira linha, que não seja em branco, do artigo
autores	uma lista com os principais nomes e sobrenomes em inglês
afiliação	*university*, *institution*, *department*, *phone*, *school*
resumo	*abstract*, *motivation*, *summary*
palavras chaves	*keywords*, *key-words*, *key words*
referências	*references*, *bibliography*, *literature*

Tabela 4.3. Informações adicionais que auxiliam a identificar corretamente as informações dos artigos científicos

A identificação dos autores, diferentemente das outras informações, não é uma tarefa trivial. Considere o artigo AslamPaSa03.ps.txt, seus três autores (linha 3) estão separados por vírgula, sendo que o primeiro autor possui um sobrenome abreviado. Essas informações podem auxiliar a identificar os autores. Entretanto, considere um pequeno trecho de um outro artigo do corpus, rotulado como BD.ps.txt, apresentado a seguir.

```
1. Buses for Anonymous Message Delivery
2. Amos Beimel Shlomi Dolev
3. Department of Computer Science
4. Ben-Gurion University of the Negev
5. Beer-Sheva 84105, Israel
6. beimel,dolev@cs.bgu.ac.il
```

Para este caso, os autores (linha 2) aparecem sem outras informações adicionais, a não ser seus próprios nomes. Como forma de solucionar tal problema, optou-se por adicionar ao sistema de extração uma lista com os principais nomes e sobrenomes para a língua inglesa. Neste trabalho foram avaliadas três fontes para compor essa lista, as quais são apresentadas na Tabela 4.4.

Com relação a segunda fonte apresentada (Arquivos BibTEX), as referências bibliográficas são automaticamente geradas a partir de arquivos BibTEX(extensão .BIB), conforme o procedimento descrito na Seção 4.1.3. Foram realizados testes utilizando cada uma das fontes acima, frente a tarefa de extrair os autores de artigos científicos. Os resultados obtidos foram avaliados e conclui-se que a fonte "Referências Bibliográficas" foi que a obteve a maior redução na taxa de erro da extração. As outras duas

[13] http://www.census.gov/genealogy/names/

Fontes	Contexto	Quantidade de nomes
Referências Bibliográficas	Nomes de autores obtidos a partir das referências de artigos científicos, os quais estão presentes na base da FIP	2499
Arquivos BibTEX	Nomes de autores contidos em referências bibliográficas, geradas a partir de arquivos BibTEX	19208
U.S. Census Bureau	O Census Bureau disponibiliza uma lista com os nomes e sobrenomes freqüentemente usados nos EUA, incluídos no *1990 Census*[13]	94159

Tabela 4.4. Fontes avaliadas para a criação da lista de nomes e sobrenomes

fontes falharam por incluir, além de nomes, na lista de autores, outras informações que não são nomes de pessoas (por exemplo IEEE) ou nomes que possuem outro significado (por exemplo *English*). Portanto, optou-se por utilizar neste trabalho a lista de nomes provenientes das próprias referências bibliográficas da base da FIP, como forma de auxiliar na extração dos autores. Vale ressaltar que essa lista será atualizada periodicamente, a medida que a FIP julgar necessário.

Outra questão importante é o fato de que uma informação incorretamente extraída, possa influenciar na extração das informações seguintes. Por exemplo, caso o sistema de extração extraia incorretamente o título de um artigo, pode ser que os autores, também, sejam extraídos de forma incorreta. Com o intuito de estimar a propagação da influência no "erro" que uma informação exerce sobre as informações seguintes, foi realizado um estudo detalhado das regras de extração induzidas, e com estas lidam com a propagação de influência. Na Tabela 4.5 são apresentados os níveis de influência para todas as informação de interesse.

	título	autores	afiliação	resumo	palavras chaves	referências
título		forte	média	fraca	fraca	fraca
autores			média	fraca	fraca	fraca
afiliação				fraca	fraca	fraca
resumo					fraca	fraca
palavras chaves						fraca
referências						

Tabela 4.5. Análise da propagação de influência que uma informação exerce sobre as outras

A influência que uma informação (linha da tabela) exerce sobre as outras informações (colunas da tabela) são classificadas em três níveis: forte, média e fraca. As células em branco significam que os níveis de influên-

cia apenas podem existir em um único sentido, segundo a ordem de apresentação das informações, comentada anteriormente. Após uma análise da tabela pode-se observar que, apesar da influência se propagar do título para os autores, é muito improvável que a influência de uma informação incorretamente extraída se propague para as outras informações.

Capítulo 5

Resultados Obtidos

Neste capítulo são apresentados os resultados obtidos mediante a realização de diversos experimentos com o objetivo de extrair informações presentes no corpo e nas referências de artigos científicos. Na Seção 5.2 é apresentado o experimento que objetiva extrair as informações presentes no corpo dos artigos. Na Seção 5.1 são descritos os experimentos relacionados com a extração de informações contidas nas referências bibliográficas. Esses experimentos visam avaliar a qualidade da extração do sistema de EI de artigos científicos, proposto neste trabalho.

5.1 Os Experimentos com as Referências Bibliográficas

O corpus de referências bibliográficas semi-automaticamente etiquetadas, utilizado nos experimentos, após a etapa de pré-processamento, possui 24928 referências e 1134986 termos.

Para avaliar o comportamento do etiquetador TBL frente à tarefa de etiquetagem e, posteriormente, estimar a qualidade da extração das informações presentes nas referências, foram realizados dois experimentos: o primeiro, nomeado CME, utilizou a porção etiquetada manualmente do corpus (947 referências), de acordo com o processo de etiquetagem descrito na Seção 4.1.3; o segundo, denominado CSAE, usou o corpus completo (24928 referências).

Para o experimento CME (Seção 5.1.1), foi utilizado o método de ava-

liação *Cross-validation*[1] com dez partições, dessa forma foram criados dez corpus de treino e dez de teste, totalizando vinte novos corpus, a partir do corpus original (manualmente etiquetado). O número de termos e de referências para cada divisão do corpus são apresentados na Tabela 5.1.

Corpus	1	2	3	4	5	6	7	8	9	10
Treino (term)	31262	31465	30996	31087	30814	30115	31464	31101	30599	30553
Treino (refs)	853	866	855	853	849	835	865	859	847	841
Teste (term)	3122	2919	3388	3297	3570	4269	2920	3283	3785	3831
Teste (refs)	94	81	92	94	98	112	82	88	100	106

Tabela 5.1. Número de termos e de referências em cada divisão do corpus manualmente etiquetado

Com relação ao experimento CSAE (Seção 5.1.2), o corpus completo foi particionado em 70% para treinar o etiquetador e 30% para avaliar a sua precisão, pois em experimentos com corpus grande é comum treinar e avaliar o etiquetador dessa forma. Nos experimentos foi utilizado um PC com a configuração Athlon XP 2400MHz com 256Mb memória RAM e sistema operacional Linux 2.4.18.

O etiquetador TBL (software) foi modificado para que os experimentos com as referências pudessem ser realizados. As alterações feitas no TBL, bem como os detalhes de implementação para a realização dos experimentos podem ser vistos no trabalho de Álvarez & Lopes (2006).

Tanto no experimento CME, quanto no CSAE, são mostrados os resultados do processo de etiquetagem e do processo de extração das informações contidas nas referências.

5.1.1 O Experimento com as Referências do Corpus Manualmente Etiquetado

Os resultados obtidos com o experimento CME são descritos a seguir. No contexto de etiquetagem de informação, o número de termos, número de etiquetas erradas e porcentagem de etiquetas corretas e erradas, para cada corpus de treino e teste são apresentados na Tabela 5.2:

[1]Este método é também conhecido como *k-fold Cross-validation* sendo *k* o número de partições geradas aleatoriamente a partir da amostra de exemplos para treinar e testar o sistema, sendo que a amostra de exemplos é dividida em *k* partições mutuamente exclusivas. A cada iteração uma partição diferente é utilizada para testar o sistema e todas as outras *k*-1 partições são utilizadas para treinar o sistema. A taxa de erro é a média das taxas de erro calculadas dadas as diversas partições.

Corpus com 947 referências (34384 termos)										
Experimento	1	2	3	4	5	6	7	8	9	10
Cp. de treino (term)	31262	31465	30996	31087	30814	30115	31464	31101	30599	30553
Cp. de teste (term)	3122	2919	3388	3297	3570	4269	2920	3283	3785	3831
Etiq. erradas (term)	190	173	206	221	253	282	201	254	208	205
Etiq. erradas (%)	6,09%	5,93%	6,08%	6,70%	7,09%	6,61%	6,88%	7,74%	5,50%	5,35%
Etiq. corretas (%)	93,91%	94,07%	93,92%	93,30%	92,91%	93,39%	93,12%	92,26%	94,51%	94,65%

Tabela 5.2. Número de termos e a taxa de acerto para cada divisão do corpus manualmente etiquetado

O número médio de termos nos corpus de treino e teste, a precisão global (média das taxas de acerto) do etiquetador e o desvio padrão, calculado a partir da Fórmula 5.1, são apresentados na Tabela 5.3.

$$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \overline{x})^2} \tag{5.1}$$

Número médio de termos no corpus de treino	30946
Número médio de termos no corpus de teste	3438
Precisão global do etiquetador	93,604%
Desvio padrão da amostra	0,7431%

Tabela 5.3. Desvio padrão e precisão global do etiquetador

Além de determinar a precisão global do etiquetador, é importante conhecer também quais são as etiquetas problemáticas, isto é, quais etiquetas mais levam o etiquetador a cometer erros. A fórmula para medir o impacto do erro causado por uma etiqueta (*t*) na taxa global de erro do etiquetador é dada pela Equação 5.2, onde P_M e F_M são, respectivamente, a precisão média e freqüência média de uma etiqueta em porcentagem. Esta fórmula corresponde ao percentual de erros na etiquetagem do corpus, relacionado com a etiqueta *t*. Considera-se neste trabalho como problemáticas, as etiquetas para as quais o fator de impacto é superior a 0,4%.

$$impacto\_no\_erro_t = (1 - \frac{P_M t}{100})F_M t \tag{5.2}$$

A precisão média por etiquetas, juntamente com as suas freqüências médias, e o fator de impacto podem ser vistos na Tabela 5.4.

Em geral, quanto mais rara a etiqueta, maior a taxa de erro. Os etiquetadores normalmente não cometem erros quando a tarefa é etiquetar sinais de pontuação, pois em geral apresentam sempre a mesma representação. Entretanto, analisando a tabela nota-se que o etiquetador cometeu alguns erros com sinais de pontuação. Este fato ocorreu por dois motivos:

Etiqueta	Precisão Média	Freqüência Média	Impacto no Erro
BOOKTITLE	88,659%	11,024%	1,2503%
TITLE	95,006%	20,803%	1,0389%
AUTHOR	95,650%	13,917%	0,6054%
JOURNAL	85,090%	3,138%	0,4679%
INITPAGE	79,668%	1,559%	0,3169%
FINALPAGE	79,748%	1,559%	0,3157%
ADDRESS	84,911%	2,069%	0,3121%
NOTE	64,902%	0,876%	0,3075%
NUMBER	70,987%	0,762%	0,2211%
PUBLISHER	86,598%	1,611%	0,2159%
VOLUME	74,884%	0,821%	0,2063%
EDITOR	75,961%	0,754%	0,1814%
SERIES	60,389%	0,443%	0,1755%
INSTITUTION	41,859%	0,289%	0,1680%
CROSSREF	46,738%	0,208%	0,1107%
SCHOOL	70,316%	0,370%	0,1099%
URL	89,859%	1,023%	0,1037%
TYPE	86,630%	0,470%	0,0629%
INDICATOR	97,403%	2,333%	0,0606%
MONTH	96,925%	0,580%	0,0178%
CHAPTER	60,000%	0,030%	0,0119%
PAGES	50,000%	0,018%	0,0089%
URLACCESSDATE	86,667%	0,063%	0,0083%
DAYS	70,000%	0,018%	0,0054%
EDITION	90,000%	0,036%	0,0036%
’	80,000%	0,006%	0,0011%
!	90,000%	0,003%	0,0003%
=	90,000%	0,003%	0,0003%
‘	90,000%	0,003%	0,0003%
´	100,000%	0,185%	0,0000%
-	100,000%	4,101%	0,0000%
"	100,000%	0,358%	0,0000%
&	100,000%	0,062%	0,0000%
(	100,000%	1,132%	0,0000%
)	100,000%	1,135%	0,0000%
,	100,000%	8,903%	0,0000%
.	100,000%	14,137%	0,0000%
:	100,000%	1,364%	0,0000%
;	100,000%	0,195%	0,0000%
?	100,000%	0,033%	0,0000%
[	100,000%	0,085%	0,0000%
\	100,000%	0,003%	0,0000%
]	100,000%	0,085%	0,0000%
˜	100,000%	0,013%	0,0000%
<	100,000%	0,021%	0,0000%
>	100,000%	0,021%	0,0000%
BARRA	100,000%	0,740%	0,0000%
YEAR	100,000%	2,639%	0,0000%

Tabela 5.4. Freqüência média e precisão por etiquetas do etiquetador

- o sinal de pontuação apareceu uma única vez no corpus;
- alguns sinais de pontuação não foram corretamente padronizados

É importante observar que as etiquetas que possuem uma pequena taxa de acerto, etiquetas com precisão inferior a 80%, apresentam uma soma de freqüências de apenas 7,7%, ou seja, a possibilidade de alguma dessas etiquetas aparecerem em alguma referência é pequena. Isso obviamente contribui para um grande índice geral de acerto. Após uma análise dos erros cometidos pelo etiquetador verificou-se que o mesmo cometeu mais erros substituindo a etiqueta BOOKTITLE por TITLE e vice versa, sendo que as duas possuem um alto fator de impacto e uma precisão alta. Acredita-se que algumas etiquetas, tais como CROSSREF e INSTITUTION, apresentem uma pequena precisão, principalmente, por possuirem uma baixa freqüência, mas também por serem usadas em situações específicas.

No contexto de extração de informação, os resultados obtidos são mostrados variando o critério usado para a contagem de extrações corretas.

TERMO-A-TERMO: considera-se como uma extração correta (acrescenta uma unidade na contagem de acertos), um termo extraído corretamente. Neste nível de comparação, o objetivo é extrair o máximo possível de partes de uma informação. Por exemplo, extrair algumas ou todas as palavras que compõem o título de uma referência.

CAMPO-A-CAMPO: neste caso considera-se como uma extração correta, uma informação completa que foi extraída corretamente. Neste nível de comparação, a meta é extrair todas as ocorrências de uma informação de interesse. Por exemplo, extrair todas as palavras do título de uma referência.

Na Tabela 5.5 são apresentados os resultados obtidos usando o critério de comparação TERMO-A-TERMO. Como já mencionado, os resultados foram avaliados em termos da métricas precisão, cobertura e F-*measure* (em particular F_1), descritas no Capítulo 3 na Seção 3.4. Nesta tabela são apresentados os valores médios dessas métricas de avaliação, as quais foram computadas para cada uma das dez partições criadas a partir do corpus.

A média dos valores da métrica F-*measure*, apresentados na Tabela 5.5, é 83,8559%. As linhas em negrito correspondem as informações (etiquetas) consideradas mais relevantes das referências, e possuem valores para

Etiqueta	**Precisão**	**Cobertura**	**F-*measure***
sinais de pontuação[2]	100,0000%	100,0000%	100,0000%
BARRA	100,0000%	100,0000%	100,0000%
YEAR	**98,3267%**	**100,0000%**	**99,1503%**
INDICATOR	97,6706%	97,4027%	97,5130%
EDITION	94,4444%	100,0000%	96,6667%
AUTHOR	**96,4907%**	**95,6496%**	**96,0352%**
MONTH	94,7782%	96,9249%	95,7655%
TITLE	**91,7392%**	**95,0062%**	**93,3276%**
TYPE	93,5996%	86,6298%	89,6009%
BOOKTITLE	**89,9722%**	**88,6594%**	**89,2640%**
URL	90,4417%	87,2164%	88,5143%
PUBLISHER	**91,0071%**	**86,5982%**	**88,4758%**
ADDRESS	86,3548%	84,9106%	85,5211%
JOURNAL	**81,6828%**	**85,0897%**	**83,1617%**
FINALPAGE	**82,8268%**	**79,7480%**	**81,0835%**
INITPAGE	**79,5930%**	**79,6680%**	**79,4454%**
NUMBER	79,2726%	70,9870%	74,4206%
EDITOR	76,6941%	75,9609%	72,7881%
VOLUME	69,4547%	74,8841%	71,7265%
SCHOOL	71,1167%	70,3158%	67,1767%
NOTE	69,8138%	64,9015%	65,1721%
SERIES	70,5216%	60,3888%	64,3564%
URLACCESSDATE	56,2925%	52,3810%	51,6222%
CROSSREF	47,5382%	46,7379%	45,8503%
INSTITUTION	56,7460%	41,8591%	45,3106%
DAYS	0,0000%	0,0000%	0,0000%
PAGES	0,0000%	0,0000%	0,0000%

Tabela 5.5. Resultados obtidos usando o corpus das referências manualmente etiquetadas, comparação TERMO-A-TERMO

F-*measure*, em geral, alto. É importante destacar que a alta precisão na extração dos sinais de pontuação, apresentada na tabela, confirma que estes foram corretamente etiquetados e puderam contribuir para a correta etiquetagem e, posterior, extração das informações em sua vizinhança. A justificativa para que DAYS e PAGES possuam precisão nula, decorre, principalmente, da baixa freqüência com que tais informações aparecem no corpus.

Os resultados da extração utilizando o critério de comparação CAMPO-A-CAMPO são mostrados na Tabela 5.6.

Etiqueta	**Precisão**	**Cobertura**	**F-*measure***
YEAR	**98,731%**	**100,000%**	**99,358%**
MONTH	94,537%	97,396%	95,819%
AUTHOR	**91,646%**	**96,720%**	**94,085%**
EDITION	85,714%	85,714%	85,714%
URL	72,738%	100,000%	82,651%
PAGES	**69,103%**	**94,401%**	**79,648%**
PUBLISHER	**68,879%**	**93,371%**	**78,659%**
TYPE	68,143%	93,988%	78,426%
TITLE	**64,980%**	**98,586%**	**78,219%**
NUMBER	74,045%	77,734%	74,734%
ADDRESS	62,290%	89,333%	73,127%
VOLUME	68,948%	78,335%	72,949%
JOURNAL	**56,376%**	**88,629%**	**68,628%**
BOOKTITLE	**48,672%**	**95,211%**	**64,194%**
EDITOR	61,274%	72,778%	63,711%
URLACCESSDATE	52,778%	66,667%	57,778%
NOTE	31,442%	61,286%	39,156%
SCHOOL	27,040%	65,000%	34,501%
CROSSREF	24,429%	50,167%	31,335%
SERIES	20,119%	35,000%	21,667%
INSTITUTION	8,722%	25,000%	12,684%
CHAPTER	0,000%	0,000%	0,000%
DAYS	0,000%	0,000%	0,000%

Tabela 5.6. Resultados obtidos usando o corpus das referências manualmente etiquetadas, comparação CAMPO-A-CAMPO

A média dos valores da F-*measure* é 60,306%. Dentre as etiquetas mostradas nessa tabela, as que estão em negrito são as mais relevantes. Em geral, os resultados apresentados na Tabela 5.6 são piores se comparado aos resultados mostrados na Tabela 5.5, devido ao critério de contagem

[2]Os sinais de pontuação são: ’ - "( ) &) , . ; ? [ ] ˜ < >

de acertos adotado. No entanto, a etiqueta PAGES obteve uma melhora significativa devido ao pós-processamento realizado com as páginas inicial (INITPAGE) e final (FINALPAGE), e o total de páginas (PAGES) que podem ocorrer nas referências.

5.1.2 Experimento com as Referências do Corpus Semi-automaticamente Etiquetado

Os resultados obtidos com o experimento CSAE são descritos a seguir. Como mencionado anteriormente, o corpus utilizado neste experimento possui mais de 1 milhão de termos e foi particionado em 70% para treinamento e 30% para teste. O corpus de teste, gerado aleatoriamente, apresenta referências bibliográficas de vários estilos contidos no corpus completo.

No contexto de etiquetagem, o número de termos, etiquetas erradas e a porcentagem de etiquetas corretas e erradas, para o corpus de treino e teste são apresentados na Tabela 5.7.

Corpus com 24928 referências (1134986 termos)	
Número de termos no corpus de treino	797652
Número de termos no corpus de teste	337334
Número de etiquetas erradas	10362
Porcentagem de etiquetas erradas	3,1%
Porcentagem de etiquetas corretas	96,9%

Tabela 5.7. Número de termos e a taxa de acerto para o conjunto de treino e de teste do corpus

Com a disponibilidade de um corpus maior, o etiquetador neste experimento alcançou uma precisão global de 96,9%. Se comparado com o experimento CME, isso corresponde a um acréscimo de 3,3% em sua precisão.

Na Tabela 5.8 são apresentados os tempos de processamento do etiquetador TBL para o treinamento e a etiquetagem do corpus de teste. Embora o tempo de treinamento de regras contextuais tenha levado 38 horas, este treinamento apenas será realizado quando for necessário à FIP conhecer novos padrões de referências.

No contexto de EI, os resultados obtidos utilizando o critério de comparação TERMO-A-TERMO são apresentados na Tabela 5.9.

	Tempo de processamento
Regras para palavras desconhecidas	5 horas, 39 minutos e 25 segundos
Regras contextuais	38 horas, 36 minutos e 10 segundos
Tempo de etiquetagem	5 segundos

Tabela 5.8. Tempo de treinamento e etiquetagem - TBL

A média dos valores da métrica F-*measure*, apresentados na Tabela 5.9, é 92,2350%. As linhas em negrito correspondem as informações (etiquetas) consideradas de maior interesse, e possuem valores para F-*measure*, em geral, alto.

Observa-se que as etiquetas da Tabela 5.5 que possuem os menores valores para F-*measure* (PAGES, DAYS, INTITUTION, SERIES) aparecem na Tabela 5.9 com um acréscimo significativo nas três métricas utilizadas.

Os resultados da extração usando o critério de comparação CAMPO-A-CAMPO são apresentados na Tabela 5.10. O valor médio da medida F-*measure* é 79,4461%. Após uma análise dos resultados, pode-se verificar que a maioria das etiquetas são extraídas com F-*measure* superior aos resultados da Tabela 5.6. Embora nos resultados se utilize um corpus grande, apenas as etiquetas HOWPUBLISHED e URLACCESSDATE, que são muito pouco freqüentes, apresentam F-*measure* nulo. Além do mais, as etiquetas consideradas mais relevantes das referências (em negrito) possuem F-measure acima de 90%.

Diante de todos os resultados apresentados, conclui-se que a etiquetagem prévia das informações contidas referências, com alta precisão, eleva a confiança no processo de extração dessas informações.

5.2 Experimento com os Artigos Científicos

O corpus de artigos científicos usado no experimento é composto por 759 artigos em Computação, dividido em 574 para a indução manual de regras de extração (corpus de treino) e 218 para testar e avaliar as regras induzidas (corpus de teste). O corpus de treino possui artigos de quatro áreas de conhecimento: Raciocínio Baseado em Casos, Programação Lógica Indutiva, Recuperação de Informação e Processamento de Língua Natural. O corpus de teste possui artigos da área de Extração de Informação.

Na Tabela 5.11 são apresentados os resultados obtidos em termos das

métricas: precisão, cobertura e F-*measure*. O critério usado para a contagem de acertos foi o CAMPO-A-CAMPO, o qual é sensível ao menor erro dentro de uma informação extraída.

Observa-se que neste experimento, as medidas relativas às referências (última linha da tabela) foram computadas considerando uma extração correta se o bloco inteiro das referências bibliográficas foi extraído.

Para facilitar a realização desses experimentos, e disponibilizar um sistema de extração de informação de artigos científicos, foi desenvolvido um aplicativo cuja a interface é apresentada na Figura 5.1.

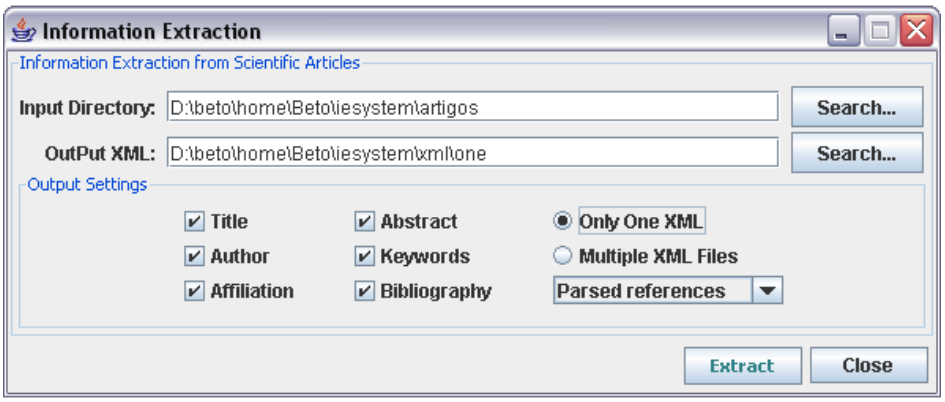


Figura 5.1. Interface do sistema de extração

O sistema de extração recebe como entrada dois diretórios, um que contém um conjunto de artigos no formato TXT, dos quais são extraídas as informações, e outro que armazena o arquivo XML contendo as informações.

Um conjunto de opções na interface permite que o usuário selecione as informações a serem extraídas. Tais opções permitem selecionar os itens a serem extraídos do corpo do artigo, e selecionar se deseja extrair o bloco de referências inteiro (como um único item de informação) ou extrair os elementos de cada uma das referências. Existe uma opção, *parsed references* ou *unparsed references*, com a qual o sistema o irá extrair o bloco inteiro ou os elementos de cada uma das referências, respectivamente.

Vale destacar que o sistema já incorpora o etiquetador TBL treinado, não exigindo do usuário a preocupação com toda a etapa de etiquetagem. No caso da escolha por *parsed references*, o sistema automaticamente chama o etiquetador.

Existem duas opções para o arquivo de saída, uma que agrupa todos os documentos em um único XML (*Only One XML*) e outra que gera o XML

para cada documento (*Multiple XML Files*).

[3]Os sinais de pontuação são: ! "# $ % & ( ) * , : ; ? @ [ ] _ ¡ ¿ ˜ + < = >

Etiqueta	Precisão	Cobertura	F-*measure*
sinais de pontuação[3]	100,0000%	100,0000%	100,0000%
BARRA	100,0000%	100,0000%	100,0000%
DAY,	100,0000%	100,0000%	100,0000%
.	99,9994%	99,9994%	99,9994%
-	99,9978%	99,9978%	99,9978%
’	99,9785%	100,0000%	99,9892%
ISSN	99,8617%	99,9862%	99,9239%
	99,5968%	100,0000%	99,7980%
AUTHOR	**99,4426%**	**99,7983%**	**99,6201%**
INDICATOR	99,2568%	99,5361%	99,3963%
TITLE	**99,1829%**	**99,3323%**	**99,2576%**
YEAR	**99,0673%**	**99,2779%**	**99,1725%**
PAGES	**98,1865%**	**99,5326%**	**98,8549%**
MONTH	98,5330%	99,0135%	98,7727%
EDITOR	98,9692%	98,1046%	98,5350%
VOLUME	98,2478%	98,5129%	98,3802%
JOURNAL	**98,1423%**	**97,7836%**	**97,9626%**
URL	96,7194%	98,6018%	97,6515%
ISBN	97,5717%	96,8237%	97,1963%
BOOKTITLE	**96,3119%**	**97,9726%**	**97,1352%**
NUMBER	96,6218%	96,8645%	96,7430%
ADDRESS	94,6286%	94,7804%	94,7044%
TYPE	92,7072%	90,1739%	91,4230%
PUBLISHER	**91,7258%**	**87,9477%**	**89,7970%**
EDITION	95,5357%	81,0606%	87,7049%
NOTE	90,8852%	81,0005%	85,6586%
SCHOOL	85,6798%	79,5078%	82,4785%
INSTITUTION	78,1482%	83,8088%	80,8796%
ORGANIZATION	78,5586%	83,0620%	80,7475%
INITPAGE	**87,7828%**	**72,6592%**	**79,5082%**
FINALPAGE	**87,4408%**	**69,1011%**	**77,1967%**
SERIES	82,2255%	72,4830%	77,0475%
KEY	94,1176%	61,5385%	74,4186%
DAYS	53,5354%	52,4752%	53,0000%
CROSSREF	91,8919%	30,0885%	45,3333%
CHAPTER	93,8462%	24,4000%	38,7302%
HOWPUBLISHED	73,3333%	4,9774%	9,3220%

Tabela 5.9. Resultados obtidos usando o corpus das referências semi-automaticamente etiquetadas, comparação TERMO-A-TERMO

Etiqueta	Precisão	Cobertura	F-*measure*
ISSN	99,6961%	100,0000%	99,8478%
YEAR	**99,7373%**	**99,5241%**	**99,6306%**
AUTHOR	**99,3480%**	**99,8901%**	**99,6183%**
MONTH	98,1207%	99,5382%	98,8244%
VOLUME	97,6225%	99,0702%	98,3410%
TITLE	**95,4744%**	**99,8237%**	**97,6006%**
NUMBER	95,8634%	98,8233%	97,3209%
EDITOR	95,1776%	98,8010%	96,9554%
PAGES	**94,6350%**	**99,3351%**	**96,9281%**
JOURNAL	**93,5112%**	**99,3768%**	**96,3548%**
URL	89,0071%	99,8409%	94,1132%
ADDRESS	90,7421%	97,4291%	93,9668%
BOOKTITLE	**87,5968%**	**99,6861%**	**93,2512%**
ISBN	84,3687%	100,0000%	91,5217%
PUBLISHER	**86,8580%**	**94,4609%**	**90,5000%**
TYPE	82,4985%	95,8333%	88,6673%
EDITION	88,5417%	84,1584%	86,2944%
DAYS	64,2045%	90,4000%	75,0831%
SERIES	61,2933%	91,5966%	73,4419%
KEY	93,7500%	60,0000%	73,1707%
NOTE	56,7788%	86,0787%	68,4241%
CHAPTER	72,5806%	62,5000%	67,1642%
SCHOOL	54,0835%	87,1345%	66,7413%
ORGANIZATION	53,0924%	84,1410%	65,1044%
INSTITUTION	52,0649%	85,8881%	64,8301%
CROSSREF	64,0000%	42,1053%	50,7937%
HOWPUBLISHED	0,0000%	0,0000%	0,0000%
URLACCESSDATE	0,0000%	0,0000%	0,0000%

Tabela 5.10. Resultados obtidos usando o corpus das referências semi-automaticamente etiquetadas, comparação CAMPO-A-CAMPO

Informação	Precisão	Cobertura	F-*measure*
título	91,0828%	92,2581%	91,6667%
autores	86,6242%	87,7419%	87,1795%
afiliação	90,0662%	91,2752%	90,6667%
resumo	94,9045%	96,1290%	95,5128%
palavras chaves	93,1034%	81,8182%	87,0968%
referências	95,5414%	96,7742%	96,1538%

Tabela 5.11. Resultados obtidos usando o corpus de artigos científicos

Capítulo 6

Conclusões e Trabalhos Futuros

A abordagem de extração proposta neste trabalho alcançou resultados em termo de Precisão e Cobertura similares aos sistemas de extração no estado da arte e, diferentemente de algumas abordagens que também se utilizam de uma classificação prévia de termos nos documentos, não se resume a extração de uma informação ou pequeno número de itens de informação. Nessas abordagem, via de regra são induzidos classificadores para, por exemplo, classificar uma sequência como *título*. Neste trabalho, desconsiderando sinais de pontuação, cerca de 30 *tags* são usadas apenas nas referências. Esse número de *tags* permite identificar e extrair praticamente todas as informações comuns em uma referência. Entretanto, conforme se pode verificar na Figura 6.1 a Precisão da extração de cada elemento das referências está altamente relacionada com a Precisão da etiquetagem.

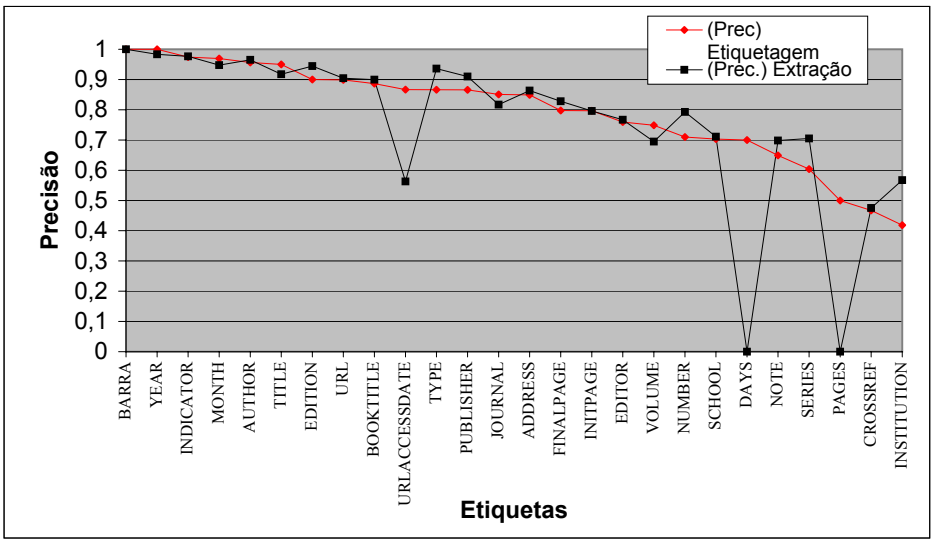


Figura 6.1. Precisão da etiquetagem e da extração por etiqueta.

Na Figura 6.1 observam-se apenas três casos nos quais as taxas de Precisão são significativamente diferentes (URLACCESSDATE, DAYS, PAGES). Porém, conforme se observar na Tabela 5.4, essas 3 etiquetas estão entre as 4 de menor impacto no erro, antes dos sinais de pontuação (que praticamente não têm influência). Ou seja, são etiquetas que raramente aparecem. Essa correlação entre as precisões leva a duas constatações, (i) que melhorar a qualidade, principalmente de etiquetas com alto impacto no erro, deve impactar positivamente a extração dos elementos correspondentes; (ii) que a abordagem proposta se justifica, pois, além da técnica de POS-tagging ser uma técnica conhecida e bastante usada, os etiquetadores são capazes de alcançar resultados muito bons.

Observa-se ainda, uma vez que a tarefa de extração de informação é ponto de partida de várias aplicações em PLN, que a saída do sistema de extração no formato XML facilita a transferência e manipulação dos dados via internet, independe de plataforma e permite fácil integração com qualquer aplicação.

A extração de informações do corpo e das referências do artigo ainda pode ser explorada. Neste trabalho, utilizou-se conhecimento de fundo, como lista de nomes próprios, que melhorou os resultados alcançados na extração de autores. Processo semelhante pode ser usado para outros elementos, como eventos, locais, etc. Sugestões e trabalhos futuros nesse sentido, são comentados na próxima seção.

Trabalhos Futuros

Nos dois procedimentos principais desenvolvidos para a extração de informações de artigos são possíveis melhorias e que podem não apenas superar os resultados do processo de extração atual, como facilitar o uso e diminuir o tempo da atividade. Sugestões relacionadas com esses dois procedimentos são apresentadas a seguir.

6.1 Extração de informação das Referências

- avaliar outros etiquetadores, principalmente os da abordagem probabilitística, para o problema de etiquetar referências bibliográficas. Pode-se citar como exemplos: MXPOST (Ratnaparkhi, 1996) e Treetagger

(Schmid, 1995). No caso específico do sistema de extração, o uso do etiquetador TBL apresenta algumas dificuldades, mesmo considerando a etapa de treinamento já realizada. O conjunto de regras não é facilmente editável, impõe que o sistema rode apenas sob o sistema operacional Linux, minimizando sua portabilidade e se houver necessidade de treinamento, a etapa de aprendizagem é bastante longa;

- realizar um pós-processamento após as referências terem sido etiquetadas, com o objetivo de elevar a qualidade da extração. Apenas as páginas (etiquetas INITPAGE, FINALPAGE, PAGES) das referências foram pós-processadas, restando, para trabalhos futuros, as outras informações (tags);

- realizar experimentos sobre etiquetagem e extração de referencias, utilizando um novo corpus provenientes de outras áreas de conhecimento e sobre outras línguas, por exemplo em português e espanhol, com forma de analisar a generalidade dessa abordagem;

- incluir outras bases BibTeX na geração do corpus semi-automaticamente etiquetado. Assim, o sistema de EI poderá "conhecer" novos modelos de referências e, consequentemente, extrair as suas informações.

6.2 Extração de Informação no Corpo dos Artigos

- induzir regras de extração "pesadas", isto é, as regras possuem um peso que corresponde a confiança na extração daquela regra. Isso poderá levar um item extraído a ser "categorizado" em mais de um tipo de informação. Um pós-processamento baseado no conhecimento do estilo da referência poderá auxiliar na categorização final;

- pré-classificar os documentos do corpus. Em um corpus recuperado automaticamente é comum a ocorrência de documentos que não são artigos científicos. Uma pré-classificação dos documentos do corpus evitaria esse problema e, eventualmente poderia identificar o formato do artigo.

- realizar um pós-processamento para quebrar os autores (que são extraídos como sendo uma única informação) em primeiro autor, se-

gundo, etc;

- melhorar e facilitar o uso de conhecimento de fundo no processo de extração;
- pesquisar abordagens de AM para a extração das informações do corpo de artigos científicos, tal como SVM;
- focar nas outras informações não tratadas neste trabalho, por ex.: notas de rodapé, evento ou jornal do artigo quando aparecem no mesmo, figuras, tabelas ou mesmo identificar as seções dos artigos, e extraí-las como sendo um "bloco" de informações;

Apêndice A

FIP tagset

Na Tabela A.1 são apresentadas as etiquetas do FIP tagset e o que cada uma abrange de informação, no contexto das referências bibliográficas. Nessa tabela, *documento* representa todos os tipos de referências vistos na Seção 4.1.1.

N	**Etiqueta**	**Informação abrangida**
1	ADDRESS	endereço completo, ou apenas a cidade, de uma editora
2	AUTHOR	autor(es) de um *documento*
3	BOOKTITLE	título de um livro ou evento
4	CHAPTER	número do capítulo de um livro
5	EDITION	edição de um livro
6	EDITOR	editor(es) de um livro ou evento
7	INSTITUTION	instituição no qual um relatório foi escrito
8	ISBN	numeração internacional para livros
9	ISSN	numeração internacional para publicações periódicas
10	JOURNAL	revista que publicou um artigo
11	MONTH	mês de publicação
12	NOTE	alguma informação adicional
13	NUMBER	número de uma revista, livro ou relatório
14	ORGANIZATION	órgão patrocinador de um evento
15	INDICATOR	pp, pages, In, Inc, Eds, Volume, Vol, No, editor(s), etc.
16	INITPAGE	página inicial do *documento*
17	FINALPAGE	página final do *documento*
18	PUBLISHER	editora
19	SCHOOL	universidade em que a monografia foi escrita
20	SERIES	série de um livro ou evento
21	TYPE	alguma informação que informe o tipo do *documento*
22	TITLE	título de um *documento* ou capítulo de um livro
23	URL	link para um *documento* (pdf, ps)
24	URLACCESSDATE	data em que a url foi acessada
25	VOLUME	volume de uma revista ou livro
26	YEAR	ano de publicação do *documento*
27	PAGES	número de páginas do *documento*
28	DAYS	dias em que aconteceu o evento
29	CROSSREF	referência cruzada
30	.	.
31	:	:
32	;	;
33	,	,
34	[	[
35	]	]
36	(	(
37	)	)
38	{	{
39	}	}
40	!	!
41	?	?
42	-	-
43	&	&
44	/	/
45	\	\
46	’	’
47	‘	‘
48	”	”
49	“	“

Tabela A.1. Conjunto de etiquetas do FIP tagset

Apêndice B

Manual do Etiquetador TBL

Neste apêndice são apresentadas as ferramentas utilizadas e os arquivos utilizados/gerados no treinamento e etiquetagem do etiquetador TBL.

O treinamento com o TBL consiste de duas etapas: aprender regras para etiquetagem de palavras desconhecidas e aprender regras contextuais. No treinamento para aprendizado de regras para palavras desconhecidas utiliza-se o comando:

```
unknown-lexical-learn.prl BIGWORDLIST SMALLWORDTAGLIST
BIGBIGRAMLIST N LEXRULEOUTFILE
```

Em que:
`BIGWORDLIST` é um arquivo com todas as palavras/símbolos que estão presentes no corpus em ordem decrescente de freqüência.

and
:
pages
IEEE
;
Signal
for

`SMALLWORDTAGLIST` é um arquivo no formato - palavra etiqueta freqüência - que lista o número de vezes que uma palavra aparece com uma dada etiqueta no corpus.

. . 89530
, , 52419
and AUTHOR 7689
In INDICATOR 5385
Conference BOOKTITLE 1420

BIGBIGRAMLIST é um arquivo com os bigramas que aparecem no corpus de treinamento.

: 33
pages .
image processing
Efficient computation
machine learning

N é um número que indica que deverão ser utilizados apenas os bigramas onde pelo menos uma das palavras é uma das n mais freqüentes no corpus.

LEXRULEOUTFILE é o arquivo onde serão armazenadas as regras aprendidas.

No treinamento para aprendizado de regras contextuais utiliza-se o comando:

```
contextual-rule-learn TAGGED-CORPUS DUMMY-TAGGED-CORPUS
CONTEXT-RULEFILE TRAINING.LEXICON
```

Em que:
TAGGED-CORPUS é o arquivo que contém o texto etiquetado manualmente, tokenizado e no formato de uma sentença por linha.

DUMMY-TAGGED-CORPUS é o arquivo formado pelo mesmo texto do arquivo TAGGED-CORPUS só que etiquetado pelo etiquetador inicial.

CONTEXT-RULEFILE é o arquivo onde serão armazenadas as regras contextuais.

TRAINING.LEXICON é o arquivo do léxico, que é formado pelas palavras e suas possíveis etiquetas que tenham aparecido no corpus de treinamento.

collection TITLE PUBLISHER NOTE

Tasaki AUTHOR
Based BOOKTITLE TITLE JOURNAL
Electronics JOURNAL BOOKTITLE INSTITUTION ORGANIZATION PUBLISHER
2000a YEAR

O etiquetador TBL tem ferramentas auxiliares para gerar este arquivos que são utilizados no treinamento e estão descritas no arquivo README que acompanha o etiquetador.

Na etiquetagem é utilizado o comando:

```
tagger LEXICON CORPUS BIGBIGRAMLISTS LEXRULEOUTFILE
CONTEXTUALRULEFILE [opções]
```

Em que:
`CORPUS` é o nome do arquivo que será etiquetado.

Depois do nome de todos os arquivos, podem ser colocadas opções - Tabela B.1.

Opções	**Descrição**
-h	Help
-w wordlist	Provê um conjunto extra de palavras além das que estão no léxico
-i filename	Grava o resultado intermediário do etiquetador inicial em um arquivo
-s number	Processa o corpus para ser etiquetado “number” linhas a cada iteração
-S	Utiliza apenas o etiquetador inicial
-F	Utiliza apenas o etiquetador final, ou seja o corpus deve estar etiquetado

Tabela B.1. Opções do comando `tagger`

Existe ainda a possibilidade de aumentar a lista de bigramas e o léxico. Há também a possibilidade de se alterar manualmente as regras.

Referências

Allen, J. (1995). *Natural Language Understanding* (2nd ed.). The Benjamin/Cummings Publishing Company. Citado na página 20.

American Psychological Association (1994). *Publication Manual of the American Psychological Association* (4th ed.). Washington, DC: Author. Citado na página 56.

Associação Brasileira de Normas Técnicas (2002). *NBR 6023: Informação e Documentação - Referências - Elaboração*. Rio de Janeiro: ABNT. Citado na página 56.

Baeza-Yates, R. & B. Ribeiro-Neto (1999). *Modern Information Retrieval*. Reading, MA: Addison-Wesley Longman Publishing Company. Citado nas páginas 19 and 20.

Baeza-Yates, R. A. (1998, October). Searching the World Wide Web: Challenges and partial solutions. In H. Coelho (Ed.), *Proceedings of the 6th Ibero-American Conference on AI on Progress in Artificial Intelligence - IBERAMIA 98*, Volume 1484 of *Lecture Notes in Computer Science*, Lisboa, Portugal, pp. 39–51. Springer Verlag. Citado na página 22.

Batista, G. E. (2003). *Pré-processamento de Dados em Aprendizado de Máquina Supervisionado*. Tese de Doutorado, ICMC-USP. São Carlos, SP. Citado na página 17.

Berners-Lee, T., J. Hendler, & O. Lassila (2001, May). The semantic web: A new form of web content that is meaningful to computers will unleash a revolution of new possibilities. *Scientific American 284*(5), 34–43. Citado na página 1.

Borkar, V., K. Deshmukh, & S. Sarawagi (2001). Automatic segmentation of text into structured records. In *Proceedings of the 2001 ACM SIGMOD International Conference on Management of Data*, California, pp. 175–186. Citado na página 45.

Brasil, C. & A. A. Lopes (2004, November). Mineração de artigos científicos usando aprendizado de máquina. In *Jornadas Chilenas de la Computa-*

ción - V Workshop de Inteligência Artificial, Arica-Chile, pp. 1–7. Citado nas páginas 13, 65, and 69.

Brasil, C. R. (2006). Abordagem simbólica de aprendizado de máquina na recuperação de artigos científicos a partir da web. Dissertação de Mestrado, ICMC - USP. Citado nas páginas 6 and 12.

Brill, E. (1994). Some advances in transformation-based part of speech tagging. In *AAAI '94: Proceedings of the twelfth national conference on Artificial intelligence (vol. 1)*, Seattle, Washington, United States, pp. 722–727. American Association for Artificial Intelligence. Citado nas páginas 32 and 33.

Brill, E. (1995). Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging. *Computational Linguistics 21*(4), 543–565. Citado nas páginas 32 and 33.

Brill, E. (1997). Unsupervised learning of disambiguation rules for part of speech tagging. In *Natural Language Processing Using Very Large Corpora*. Kluwer Academic Press. Citado nas páginas 32 and 33.

Califf, M. E. & R. J. Mooney (2003). Bottom-up relational learning of pattern matching rules for information extraction. *Journal of Machine Learning Research 4*, 177–210. Citado na página 46.

Chanod, J. P. & P. Tapanainen (1995). Creating a tagset, lexicon and guesser for a french tagger. Citado na página 32.

Chicago Editorial Staff (1993). *Chicago Manual of Style* (14th ed.). Chicago: University of Chicago Press. Citado na página 56.

Ciravegna, F. (2001). Adaptive information extraction from text by rule induction and generalization. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, Seattle, WA, pp. 1251–1256. Citado na página 45.

Connan, J. & C. W. Omlin (2000, March 24). Bibliography extraction with hidden markov models. Technical report, Department of Computer Science, University of Stellenbosch. Citado nas páginas 45 and 53.

Cowie, J. & W. Lehnert (1996). Information extraction. *Communications of the ACM 39*(1), 80–91. Citado nas páginas 39, 41, 44, and 50.

Cussens, J. & S. Džeroski (2000). *Learning language in logic*. Springer-Verlag New York, Inc. Citado na página 46.

Daelemans, W., J. Zavrel, & S. Berck (1996). MBT: A memory based part of speech tagger-generator. In E. Ejerhed & I. Dagan (Eds.), *Proceedings of the Fourth Workshop on Very Large Corpora*, pp. 14–27. Citado na página 32.

Day, M.-Y., T.-H. Tsai, C.-L. Sung, C.-W. Lee, S.-H. Wu, C.-S. Ong, & W.-L. Hsu (2005). A knowledge-based approach to citation extraction. In D. Zhang, T. M. Khoshgoftaar, & M.-L. Shyu (Eds.), *Proceedings of the 2005 IEEE International Conference on Information Reuse and Integration, IRI - 2005, August 15-17, 2005, Las Vegas Hilton, Las Vegas, NV, USA*, pp. 50–55. IEEE Systems, Man, and Cybernetics Society. Citado na página 52.

Ding, Y., G. Chowdhury, & S. Foo (1999). Template mining for the extraction of citation from digital documents. *Proceedings of the Second Asian Digital Library Conference, Taiwan*, 47–62. Citado na página 52.

Dörre, J., P. Gerstl, & R. Seiffert (1999). Text mining: finding nuggets in mountains of textual data. In *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, pp. 398–401. ACM Press. Citado na página 19.

EAGLES (1996). EAGLES - Expert Advisory Group on Language Engineering Standards. Recommendations for the Morphosyntactic Annotation of Corpora. `http://www.ilc.cnr.it/EAGLES96/pub/eagles/corpora/annotate.ps.gz`. Citado na página 30.

Ebecken, N. F. F., M. C. S. Lopes, & M. C. A. Costa (2003). Mineração de textos. In S. O. Rezende (Ed.), *Sistemas Inteligentes - Fundamentos e Aplicações*, Volume 1, Chapter 13, pp. 337–370. Manole. Citado nas páginas 18, 24, 25, and 28.

Eikvil, L. (1999). Information extraction from World Wide Web: A survey. Technical Report 945, Norweigan Computing Center. Citado na página 45.

Fayyad, U. M., G. Piatetsky-Shapiro, & P. Smyth (1996a). From data mining to knowledge discovery: An overview. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, & R. Uthurusamy (Eds.), *Advances in Knowledge Discovery and Data Mining*, pp. 1–34. Menlo Park, CA, USA: American Association for Artificial Intelligence/MIT Press. Citado nas páginas 15, 16, and 18.

Fayyad, U. M., G. Piatetsky-Shapiro, & P. Smyth (1996b). The KDD process for extracting useful knowledge from volumes of data. *Communications of the ACM 39*(11), 27–34. Citado na página 16.

Feldman, R. & H. Hirsh (1997). Exploiting background information in knowledge discovery from text. *Journal of Intelligence Information Systems 9*(1), 83–97. Citado nas páginas 18 and 19.

Forman, G. (2003, March). An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research 3*, 1289–1305. `http://jmlr.csail.mit.edu/papers/volume3/forman03a/forman03a.pdf`. Citado na página 25.

Frakes, W. B. & R. Baeza-Yates (1992). *Information Retrieval: data structures and algorithms*. Englewood Cliffs, New Jersey, US: Prentice Hall. ISBN 0-13-463837-9. Citado na página 24.

Freitag, D. & N. Kushmerick (2000). Boosted wrapper induction. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pp. 577–583. AAAI Press / The MIT Press. Citado na página 45.

Freitag, D. & A. McCallum (2000). Information extraction with HMM structures learned by stochastic optimization. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pp. 584–589. AAAI Press / The MIT Press. Citado na página 45.

Gaizauskas, R. & Y. Wilks (1998). Information extraction: Beyond document retrieval. *Journal of Documentation 54*(1), 70–105. Citado nas páginas 40 and 47.

Geng, J. & J. Yang (2003). Autobib: Automatic extraction and integration of bibliographic information on the web. In *29th VLDB Conference*, Berlin, Germany. Citado na página 45.

Geng, J. & J. Yang (2004). Autobib: Automatic extraction of bibliographic information on the web. In *8th International Database Engineering and Applications Symposium (IDEAS 2004), 7-9 July 2004, Coimbra, Portugal*, pp. 193–204. IEEE Computer Society. Citado na página 53.

Giuffrida, G., E. C. Shek, & J. Yang (2000). Knowledge-based metadata extraction from postscript files. In *DL '00: Proceedings of the fifth ACM conference on Digital libraries*, New York, NY, USA, pp. 77–84. ACM Press. Citado na página 52.

Glickman, O. & R. Jones (1999). Examining machine learning for adaptable end-to-end information extraction systems. In *AAAI 1999 Workshop on Machine Learning for Information Extraction*. Citado na página 44.

Grishman, R. (1997). Information extraction: Techniques and challenges. In *SCIE '97: International Summer School on Information Extraction*, pp. 10–27. Springer-Verlag. Citado nas páginas xvii, 39, 41, and 42.

Han, H., C. L. Giles, E. Manavoglu, H. Zha, Z. Zhang, & E. A. Fox (2003). Automatic document metadata extraction using support vector machines. In *JCDL '03: Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries*, Washington, DC, USA, pp. 37–48. IEEE Computer Society. Citado na página 54.

Hearst, M. A. (1999). Untangling text data mining. In *Proceedings of ACL'99 the 37th Annual Meeting of the Association for Computational Linguistics*. Citado na página 18.

Hobbs, J., D. Appelt, J. Bear, D. Israel, M. Kameyama, M. Stickel, & M. F. Tyson (1997). Fastus: A cascaded finite-state transducer for extracting information from natural-language text. In E. Roche & Y. Schabes (Eds.), *Finite-State Devices for Natural Language Processing*, pp. 383–406. Cambridge, MA: MIT Press. Citado nas páginas 41 and 44.

Hotho, A., A. Nürnberger, & G. Paaß (2005). A brief survey of text mining. *GLDV - Journal for Computational Linguistics and Language Technologie 20*(1), 19–62. Citado na página 19.

Joachims, T., D. Freitag, & T. M. Mitchell (1997, August). Web watcher: A tour guide for the World Wide Web. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI-97)*, Nagoya, Japan, pp. 770–777. Morgan Kaufmann Publishers. Citado na página 22.

Jorge, A. & A. A. Lopes (2000). Iterative part-of-speech tagging. In *Learning Language in Logic*, Volume 1925 of *Lecture Notes in Artificial Intelligence*, pp. 170–183. Berlim: Springer Verlag. Citado na página 45.

Junker, M., M. Sintek, & M. Rinck (1999). Learning for text categorization and information extraction with ILP. In J. Cussens (Ed.), *Proceedings of the 1st Workshop on Learning Language in Logic*, Bled, Slovenia, pp. 84–93. Citado na página 45.

Kanashiro, A. (2005, March). Um data warehouse para publicações científicas. Qualificação de Mestrado. ICMC/USP. Citado na página 13.

Keim, D. A. (2002, March). Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics 8*(1), 1–8. ISSN 1077-2626. Citado na página 28.

Kim, J. & D. I. Moldovan (1995). Acquisition of linguistic patterns for knowledge-based information extraction. *IEEE Transactions on Knowledge and Data Engineering 7*(5), 713–724. Citado na página 46.

Lawrence, S., K. Bollacker, & C. L. Giles (1999, November). Indexing and retrieval of scientific literature. In *Proceedings of the 8th International Conference on Information and Knowledge Management (CIKM-99)*, Kansas Cite, Missouri, USA, pp. 139–146. ACM Press. `http://citeseer.csail.mit.edu/lawrence99indexing.html`. Citado nas páginas 6 and 54.

Leech, G., R. Garside, & M. Bryant (2004). Claws4: The tagging of the british national corpus. In *Proceedings of the 15th International Conference on Computational Linguistics*, pp. 622–628. Citado na página 30.

Lopes, A. & P. Brazdil (1998). Redundant covering with global evaluation in the RC1 inductive learner. In *Advances in Artificial Intelligence, 14th Brazilian Symposium on Artificial Intelligence - SBIA 98*, LNAI 1515. Springer Verlag. Citado na página 45.

Lopes, A. A. & A. Jorge (2000). Combining rule-based and case-based learning for iterative part-of-speech tagging. In *EWCBR '00: Proceedings of the 5th European Workshop on Advances in Case-Based Reasoning*, London, UK, pp. 26–36. Springer-Verlag. Citado na página 32.

Lopes, A. A., R. Minghim, V. V. Melo, & F. V. Paulovich (2006, January). Mapping texts through dimensionality reduction and visualization techniques for interactive exploration of document collections. In R. F. Erbacher, J. C. Roberts, M. T. Gröhn, & K. Börner (Eds.), *IST/SPIE Workshop on Visualization and Data Analysis (VDA)*, Volume 6060, San Jose, California, USA, pp. 01–12. Citado nas páginas 3 and 9.

Lopes, A. A., R. Pinho, F. V. Paulovich, & R. Minghim (2007). Visual text mining using association rules. *Computers & Graphics: An International Journal of Systems & Applications in Computer Graphics 31*(6). Artigo aceito para publicação. Citado nas páginas 5, 9, 10, and 28.

Lovins, J. B. (1968). Development of a stemming algorithm. *Mechanical Translation and Computational Linguistics 11*(1-2), 22–31. Citado na página 25.

Álvarez, A. C. & A. A. Lopes (2006, September). Extração de informação de referências bibliográficas usando POS-*tagging*. Technical Report 281, ICMC-USP. ISSN 0103-2569. Citado na página 76.

Ma, Q., K. Uchimoto, M. Murata, & H. Isahara (1999, July). Elastic neural

networks for part of speech tagging. In *International Joint Conference on Neural Networks(IJCNN'99)*, Washington, DC. Citado na página 32.

Manning, C. & H. Schütze (2001). *Foundations of Statistical Natural Language Processing*. Cambridge, MA: MIT Press. Citado nas páginas 20 and 25.

Mao, S., J. W. Kim, & G. R. Thoma (2004). A dynamic feature generation system for automated metadata extraction in preservation of digital materials. In *1st International Workshop on Document Image Analysis for Libraries (DIAL 2004), 23-24 January 2004, Palo Alto, CA, USA*, pp. 225–232. IEEE Computer Society. Citado na página 53.

Marcus, M. P., B. Santorini, & M. A. Marcinkiewicz (1994). Building a large annotated corpus of english: The Penn Treebank. *Computational Linguistics 19*(2), 313–330. Citado na página 30.

McCallum, A., K. Nigam, J. Rennie, & K. Seymore (2000). Automating the Construction of Internet Portals with Machine Learning. *Information Retrieval 3*(2), 127–163. Citado na página 54.

Melo, V. & A. A. Lopes (2004a). Identificação eficiente de referências bibliográficas duplicadas em um corpora de artigos científicos. In *II Workshop de Teses e Dissertações em Inteligência Artificial*, São Luis - Maranhão, pp. 71–80. Citado nas páginas 13 and 53.

Melo, V. & A. A. Lopes (2004b). Usando as referências bibliográficas no clustering de artigos científicos. In *Jornadas Chilenas de la Computación - V Workshop de Inteligência Artificial*, Arica-Chile, pp. 1–7. Citado na página 13.

Melo, V. & A. A. Lopes (2005). Efficient identification of duplicate bibliographical references. In *Proceedings of the 5th Congress of Logic Applied to Technology - Laptec 2005*, Himeji - Japan, pp. 1–8. Citado nas páginas 13 and 53.

Melo, V., M. Secato, & A. A. Lopes (2003). Extração e identificação automáticas de informações bibliográficas de artigos científicos. In *Jornadas Chilenas de la Computación - IV Workshop de Inteligência Artificial*, Chillán, pp. 1–7. Citado nas páginas 13 and 53.

Melo, V. V. (2005). Clustering de artigos científicos. Dissertação de Mestrado, ICMC - USP. Citado na página 13.

Minghim, R., F. V. Paulovich, & A. A. Lopes (2006, January). Content-based text mapping using multidimensional projections for exploration of document collections. In R. F. Erbacher, J. C. Roberts, M. T.

Gröhn, & K. Börner (Eds.), *Proceedings of SPIE on Visualization and Data Analysis*, Volume 6060, San Jose, California, USA, pp. 1–12. Citado na página 9.

Monard, M. C. & J. A. Baranauskas (2003). Conceitos sobre aprendizado de máquina. In S. O. Rezende (Ed.), *Sistemas Inteligentes - Fundamentos e Aplicações* (1 ed.), Volume 1, Chapter 4, pp. 89–114. Manole. Citado na página 18.

Muggleton, S. (1995). Inverse entailment and progol. *New Generation Computing 13*, 245–286. Citado na página 46.

Muggleton, S. & C. Feng (1992). Efficient induction in logic programs. In S. Muggleton (Ed.), *Inductive Logic Programming*, pp. 281–298. New York: Academic Press. Citado na página 46.

Muslea, I. (1999, July). Extraction patterns for information extraction tasks: A survey. In *Proceedings of AAAI Workshop on Machine Learning for Information Extraction*, Orlando, Florida. Citado na página 43.

Parsons, S. (1998). Addendum to "current approaches to handling imperfect information in data and knowledge bases". *IEEE Transactions on Knowledge and Data Engineering 10*(5), 862. Citado na página 27.

Paulovich, F. V. & R. Minghim (2006). Text map explorer: a tool to create and explore document maps. In *10th International Conference on Information Visualisation (IV'06)*, Volume 1, Londres, pp. 245–251. IEEE Computer Society Press. Citado na página 9.

Peng, F. & A. McCallum (2004). Accurate information extraction from research papers using conditional random fields. In *Proceedings of Human Language Technology Conference and North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, pp. 329–336. Citado na página 54.

Peterson, R. E. (1997). Eight internet search engines compared. *First Monday 2*(2). `http://www.firstmonday.org/issues/issue2_2/peterson/index.html`. Citado na página 21.

Petinot, Y., P. B. Teregowda, H. Han, C. L. Giles, S. Lawrence, A. Rangaswamy, & N. Pal (2003). eBizSearch: an OAI-compliant digital library for eBusiness. In *JCDL '03: Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries*, Washington, DC, USA, pp. 199–209. IEEE Computer Society. Citado na página 54.

Porter, M. F. (1980, July). An algorithm for suffix stripping. *Program 14*(3), 130–137. Citado na página 25.

Radev, D. R., E. H. Hovy, & K. McKeown (2002). Introduction to the special issue on summarization. *Computational Linguistics 28*(4), 399–408. Citado na página 27.

Rajman, M. & R. Besançon (1997). *Text Mining: Natural Language techniques and Text Mining applications*. Chapman and Hall. Citado na página 44.

Ratnaparkhi, A. A. (1996). A maximum entropy part-of-speech tagger. In *Proceedings of the Empirical Methods in Natural Language Processing Conference*, Philadelphia, Pa. Citado nas páginas 32 and 90.

Rezende, S. O., J. B. Pugliesi, E. A. Melanda, & M. F. Paula (2003). Mineração de dados. In S. O. Rezende (Ed.), *Sistemas Inteligentes - Fundamentos e Aplicações*, Volume 1, Chapter 12, pp. 307–335. Manole. Citado nas páginas xvii, 16, and 18.

Riloff, E. (1993). Automatically constructing a dictionary for information extraction tasks. In *National Conference on Artificial Intelligence*, pp. 811–816. Citado na página 46.

Schmid, H. (1995). Probabilistic part–of–speech tagging using decision trees. In *Proceedings of the Conference on New Methods in Language Processing*, Manchester, UK. Citado nas páginas 32 and 91.

Seymore, K., A. McCallum, & R. Rosenfeld (1999, July). Learning hidden markov model structure for information extraction. In *AAAI'99 Workshop on Machine Learning for Information Extraction*, Orlando, Florida, USA. Citado nas páginas 45, 52, and 53.

Silberschatz, A. & A. Tuzhilin (1995, August). On subjective measures of interestingness in knowledge discovery. In U. M. Fayyad & R. Uthurusamy (Eds.), *Proceedings of the First International Conference on Knowledge Discovery and Data Mining (KDD-95)*, Montreal, Canada, pp. 275–281. AAAI Press. Citado na página 18.

Soderland, S. (1999). *Learning Information Extraction Rules for Semi-structured and Free Text*. Kluwer Academic Publishers. Citado na página 44.

Soderland, S., D. Fisher, J. Aseltine, & W. Lehnert (1995). Crystal: Inducing a conceptual dictionary. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, Montreal, Quebec, pp. 1314–1319. Citado na página 46.

Sundheim, B. (1992, June). Overview of the fourth message understanding evaluation and conference. In *Proceedings of Fourth Message Understanding Conference (MUC-4)*. Citado na página 47.

Takasu, A. (2003). Bibliographic attribute extraction from erroneous references based on a statistical model. In *JCDL '03: Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries*, Houston, Texas, pp. 49–60. IEEE Computer Society. Citado nas páginas 45 and 53.

Tan, A.-H. (1999, April). Text mining: The state of the art and the challenges. In N. Zhong & L. Zhou (Eds.), *Proceedings of the PAKDD 1999 Workshop on Knowledge Disocovery from Advanced Databases (PAKDD'99)*, Beijing, China, pp. 65–70. Citado na página 19.

Voutilainen, A. (1995). A syntax-based part-of-speech analyser. In *Proceedings of the seventh conference on European chapter of the Association for Computational Linguistics*, San Francisco, CA, USA, pp. 157–164. Morgan Kaufmann Publishers Inc. Citado na página 32.

Weiss, S. M. & N. Indurkhya (1998). *Predictive data mining: a practical guide*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. Citado nas páginas 16 and 17.

Weiss, S. M., N. Indurkhya, T. Zhang, & F. J. Damerau (2005a). From textual information to numerical vectors. In *Text Mining: Predictive Methods for Analysing Unstructured Information*, pp. 15–44. Springer Verlag. Citado nas páginas 22, 24, and 25.

Weiss, S. M., N. Indurkhya, T. Zhang, & F. J. Damerau (2005b). Information retrieval and text mining. In *Text Mining: Predictive Methods for Analysing Unstructured Information*, pp. 85–101. Springer Verlag. Citado na página 20.

Wiebe, J., G. Hirst, & D. Horton (1996). Language use in context. *Communications of the ACM 39*(1), 102–111. Citado na página 42.

Wilkens, M. & J. Kupiec (1996). Training hidden markov models for part of speech tagging. Revision 4. Citado na página 32.

Wilks, Y. (1997). Information extraction as a core language technology. In M. T. Pazienza (Ed.), *SCIE '97 International Summer School on Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology*, Volume 1299 of *Lecture Notes In Computer Science*, pp. 1–9. London, UK: Springer-Verlag. Citado na página 27.

Yang, Y. & J. O. Pedersen (1997, July). A comparative study on feature selection in text categorization. In D. H. Fisher (Ed.), *Proceedings of ICML-97, 14th International Conference on Machine Learning*, Nashville, Tennessee, USA, pp. 412–420. Morgan Kaufmann Publishers, San Francisco, US. `http://citeseer.ist.psu.edu/386891.html`. Citado nas páginas 24, 25, and 26.

Yin, P., M. Zhang, Z.-H. Deng, & D. Yang (2004). Metadata extraction from bibliographies using bigram HMM. In Z. Chen, H. Chen, Q. Miao, Y. Fu, E. A. Fox, & E.-P. Lim (Eds.), *Digital Libraries: International Collaboration and Cross-Fertilization, 7th International Conference on Asian Digital Libraries, ICADL 2004, Shanghai, China, December 13-17, 2004, Proceedings*, Volume 3334 of *Lecture Notes in Computer Science*, pp. 310–319. Springer. Citado nas páginas 45 and 53.

Zelle, J. M. & R. J. Mooney (1994). Combining top-down and bottom-up methods in inductive logic programming. In *Proceedings of the Eleventh International Conference on Machine Learning*, New Brunswick, NJ, pp. 343–351. Citado na página 46.

Zhong, S. & J. Ghosh (2003). A unified framework for model-based clustering. *Journal of Machine Learning Research 4*, 1001–1037. Citado na página 26.