UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

Metodologia para Tratamento e Manipulação de Informações de Comércio Eletrônico

por

RUI GUREGHIAN SCARINCI

Tese submetida à avaliação, como requisito
parcial para a obtenção do grau de doutor em
Ciência da Computação

Prof. Dr. José Palazzo M. de Oliveira
Orientador

Porto Alegre, julho de 2003.

CIP - CATALOGAÇÃO DA PUBLICAÇÃO

Scarinci, Rui Gureghian.

Metodologia para Tratamento e Manipulação de Informações de Comércio Eletrônico / por Rui Gureghian Scarinci — Porto Alegre: PPGC da UFRGS, 2003.

138 f.: il.

Tese (doutorado) — Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR-RS, 2003. Orientador: Oliveira, José Palazzo Moreira de.

1. Extração de Informações. 2. Recuperação de Informações. 3. Sistemas de Informação. I. Oliveira, José Palazzo Moreira de. II. Título.

Agradecimentos

Durante os sete anos transcorridos desde meu ingresso no Curso de Pós-Graduação em Ciência da Computação até o presente, conheci, trabalhei e mantive relações de amizade com diversas pessoas, que muito me auxiliaram nesta caminhada. Particularmente, gostaria de destacar e agradecer às seguintes pessoas:

- Professor Dr. José Palazzo Moreira de Oliveira, pela orientação, conselhos e ações práticas efetivadas com o intuito de oferecer as condições materiais e psicológicas para que eu pudesse realizar o meu trabalho.
- Armenuhy Gureghian de Scarinci, minha mãe e eterna orientadora, a quem dedico este trabalho, por ter me acompanhado em toda a minha trajetória, não apenas como estudante, mas, principalmente, como homem.
- Cristina Silveira Moraes Leite, pelo auxílio na redação deste trabalho e pela compreensão nos momentos em que estive distante mesmo sem desejar, mas, principalmente, por nunca deixar de estar ao meu lado e amar, mesmo nos momento difíceis.
- Bolsistas Christian Zambenedetti e Diego Schultz Trein, que me auxiliaram na busca da bibliografia que serviu de embasamento conceitual deste trabalho; bem como pela amizade demonstrada no transcorrer destes anos.
- Aos verdadeiros amigos que encontrei ao longo da minha vida, que, embora tenham sido poucos, certamente foram sinceros e dedicados.
- Finalmente, gostaria de agradecer àquelas pessoas que tentam obstaculisar-me, motivados por seu preconceito cego e pérfido, àqueles que sob o manto da solidariedade ocultam sentimentos vis, pois eles, com seus pensamentos e atitudes, fazem-me lutar com maior energia e convicção em busca de meus objetivos.

Sumário

Lista de Abreviaturas....07

Lista de Figuras....08

Lista de Tabelas....09

Resumo....10

Abstract....11

1 Introdução....12

1.1 Trabalho Desenvolvido....13

1.2 Estrutura da Tese....14

2 Visão Geral....15

2.1 Identificação do Problema....15
2.1.1 Quantidade....15
2.1.2 Diversidade de Estruturas....15
2.1.3 Validade....16
2.1.4 Caches....16
2.1.5 Diversidade de Usuários....17

2.2 Objetivo....17

2.3 Diferenciais e Benefícios....18
2.3.1 Nova Proposta de Integração de Processos: Bases de Dados Intermediárias....18
2.3.2 Nova Proposta de Configuração do Sistema de Extração: Sistema Baseado em Conhecimento....20

3 Extração de Informação....23

3.1 História....24

3.2 Definição do Domínio....25

3.3 Recuperação de Documentos e Extração de Informação....27
3.3.1 RD como Ponto de Partida para Extração....28
3.3.2 EI como Ponto de Partida para Recuperação....28
3.3.3 Recuperação e Extração em um Mesmo Nível....29

3.4 Extração e Descoberta de Informação em Textos....29

3.5 Método de Sistemas de Extração de Informações....31
3.5.1 Simplificação do Processo Sintático....32
3.5.2 Aprendizado de Linguagem Baseada em Textos Exemplo....34
3.5.3 Fases de Extração....35

3.6 Avaliação....40
3.6.1 Métricas....41

3.7 Performance....42

4 Extração de Informações em Múltiplos Níveis Baseada em Conhecimento.....44

4.1 Bases de Dados Intermediárias - BDI.....44
4.1.1 Relações Entre as Bases de Dados.....46
4.1.2 Múltiplos Níveis Conceituais.....47

4.2 Sistema Baseado em Conhecimento - SBC.....49
4.2.1 Máquina de Inferência.....51
4.2.2 Eventos de Entrada.....51
4.2.3 Conclusão.....51
4.2.4 Base de Conhecimento.....51
4.2.5 Representação e Construção do Conhecimento.....51

4.3 Processos de Extração.....53
4.3.1 Classificação Estrutural – P1.....54
4.3.2 Classificação por Domínio – P2.....56
4.3.3 Classificação.....60
4.3.4 Análise Superficial – P3.....61

5 Protótipo: Sistema de Extração de Informações em Múltiplos Níveis Baseado em Conhecimento – SE-MNBC.....67

5.1 Arquitetura.....67

5.2 Módulos do Sistema.....68
5.2.1 Classificador por Estrutura.....68
5.2.2 Classificador por Domínio.....70
5.2.3 Analisador Superficial.....71
5.2.4 Configurador.....72

5.3 Base de Conhecimento - BC.....74

5.4 Dicionários.....75

5.5 Bases de Dados.....75

5.6 Usuário.....75

5.7 Visão Geral do Processo de Extração Usando o SE-MNBC.....75

5.8 Implementação.....76
5.8.1 Ferramenta de Programação.....76
5.8.2 Requisitos de Execução.....77

6 Avaliação Experimental.....78

6.1 Economia Globalizada, Competição e Trade Points.....78

6.2 Rede Global de Trade Points – GTPN.....79

6.3 ETO - Eletronic Trade Opportunity.....79

6.4 Pequenas e Médias Empresas - PME.....81

6.5 Plano Experimental.....81

6.5.1 Variáveis 82
6.5.2 Métricas 82
6.5.3 Definições 82
6.5.4 Etapas.....83
6.6 Experimento A 83
6.7 Experimento B 88
7 Conclusão 92
7.1 Trabalhos Futuros 93
Anexo 1 Conferência de Entendimento de Mensagens 95
Anexo 2 Sistemas Relacionados.....103
Bibliografia.....126

Lista de Abreviaturas

ASCII	American Standard Code for Information Interchange
ATN	Augmented Transition Network
BC	Bases de Conhecimento
BD	Bases de Dados
BDI	Bases de Dados Intermediárias
BDNE/SE	Bases de Dados Não Estruturadas e/ou Semi-Estruturadas
CO	Coreference
CP	Condição de Pesquisa
CV	Condição de Verificação
EI	Extração de Informação
EI-MNBC	Extração de Informações em Múltiplos Níveis Baseada em Conhecimento
E-Mail	Eletronic Mail
ETO	Electronic Trading Opportunities
FTP	File Transfer Protocol
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IA	Inteligência Artificial
LN	Linguagem Natural
LSI	Latent Semantic Indexing
MD	Mineração de Dados
MDT	Mineração em Dados em Textos
MET	Multilingual Entity Task
MLDB	Multiple Layered Database
MUC	Message Understanding Conference
NC	Nodo Conceito
NE	Named Entity
PLN	Processamento da Linguagem Natural
PLN	Processamento da Linguagem Natural
PME	Pequenas e Médias Empresas
RAM	Random Acess Memory
RD	Recuperação de Documentos
RI	Recuperação de Informação
SBC	Sistema Baseado em Conhecimento
SE	Sistemas Especialistas
SE-MNBC	Sistema de Extração de Informações em Múltiplos Níveis Baseado em Conhecimento
SES	Sistema de Extração Semântica
SGBD	Sistema Gerenciador de Bancos de Dados
SGML	Standard Generalized Markup Language
ST	Scenario Template
TE	Template Elements
TI	Tecnologia da Informação
TR	Template Relations
URL	Uniform Resource Locator
WAIS	Wide Area Information Server

Lista de Figuras

FIGURA 3.1 - Modelo de Sistema de EI no Domínio de Desastres Naturais 24
FIGURA 3.2 - Regra Script do Tipo Produção 26
FIGURA 3.3 - Um Nodo Conceito Instanciado 27
FIGURA 3.4 - RD como Ponto de Partida para EI.....28
FIGURA 3.5 - EI como Ponto de Partida para RD.....28
FIGURA 3.6 - RD e EI Atuando em um Mesmo Nível.....29
FIGURA 3.7 - Metodologia Genérica de Sistemas de Extração 36
FIGURA 3.8 - Avaliação do Processo de Extração de Informação 41
FIGURA 3.9 - Recall e Precision.....42
FIGURA 4.1 - Bases de Dados Intermediárias (BDI) 45
FIGURA 4.2 - Relacionamentos na BDI.....47
FIGURA 4.3 - Hierarquia Conceitual.....49
FIGURA 4.4 - SBC na EI-MNBC 50
FIGURA 4.5 - Regra de Produção.....53
FIGURA 5.1 - Arquitetura do SE-MNBC 67
FIGURA 5.2 - Hierarquia Organizacional do SE-MNBC.....68
FIGURA 5.3 - Arquitetura do Classificador por Estrutura.....68
FIGURA 5.4 - Arquitetura do Classificador por Domínio 70
FIGURA 5.5 - Arquitetura do Analisador Superficial.....71
FIGURA 5.6 - Arquitetura do Configurador 72
FIGURA 6.1 - Exemplo de um ETO 81
FIGURA 6.2 - Tamanho das Bases de Dados Conforme por Processo.....89
FIGURA 6.3 - Velocidade de Extração de P1.....90
FIGURA 6.4 - Velocidade de Extração de P2.....90
FIGURA 6.5 - Velocidade de Extração de P3.....91

Lista de Tabelas

TABELA 3.1 - Duas Definições de Nodos Conceito 27
TABELA 4.1 - Conhecimento Relacional Simples 53
TABELA 4.2 - Base de Dados Nível 1 – N1 56
TABELA 4.3 - Base de Dados Nível 2 – N2 61
TABELA 4.4 - Base de Dados Nível 3 – N3 66
TABELA 6.1 - Resultado Final do Processo P2 – Experimento A 85
TABELA 6.3 - Resultados do Experimento B 88

Resumo

A grande disponibilidade de informações oferece um amplo potencial comercial. Contudo, o enorme volume e diversidade de oportunidades gera um problema: limitações comerciais criadas pela seleção e a manipulação manual dessa informação. O tratamento das grandes bases de dados não estruturadas e/ou semi-estruturadas (BDNE/SE), como as trazidas pela Internet, é uma fonte de conhecimento rica e confiável, permitindo a análise de mercados. O tratamento e a estruturação dessa informação permitirá seu melhor gerenciamento, a realização de consultas e a tomada de decisões, criando diferenciais competitivos de mercado.

Pesquisas em Recuperação de Informação (RI), as quais culminaram nesta tese, investem na melhoria da posição competitiva de pequenas e médias empresas, hoje inseridas, pelo comércio eletrônico, em um mercado globalizado, dinâmico e competitivo. O objetivo deste trabalho é o desenvolvimento de uma Metodologia de Extração de Informações para o Tratamento e Manipulação de Informações de Comércio Eletrônico. Chamaremos essa metodologia de EI-MNBC, ou seja, Extração de Informações em Múltiplos Níveis Baseada em Conhecimento. Os usuários da EI-MNBC podem rapidamente obter as informações desejadas, frente ao tempo de pesquisa e leitura manual dos dados, ou ao uso de ferramentas automáticas inadequadas. Os problemas de volume, diversidade de formatos de armazenamento, diferentes necessidades de pesquisa das informações, entre outros, serão solucionados.

A metodologia EI-MNBC utiliza conhecimentos de RI, combinando tecnologias de Recuperação de Documentos, Extração de Informações e Mineração de Dados em uma abordagem híbrida para o tratamento de BDNE/SE. Propõe-se uma nova forma de integração (múltiplos níveis) e configuração (sistema baseado em conhecimento - SBC) de processos de extração de informações, tratando de forma mais eficaz e eficiente as BDNE/SE usadas em comércio eletrônico. Esse tratamento viabilizará o uso de ferramentas de manipulação de dados estruturados, como Sistemas Gerenciadores de Bancos de Dados, sobre as informações anteriormente desestruturadas.

A busca do conhecimento existente em bases de dados textuais não estruturadas demanda a compreensão desses dados. O objetivo é enfatizar os aspectos cognitivos superficiais envolvidos na leitura de um texto, entendendo como as pessoas recuperam as informações e armazenando esse conhecimento em regras que guiarão o processo de extração. A estrutura da metodologia EI-MNBC é similar a de um SBC: os módulos de extração (*máquinas de inferência*) analisam os documentos (*eventos*) de acordo com o conteúdo das *bases de conhecimento*, interpretando as regras. O resultado é um arquivo estruturado com as informações extraídas (*conclusões*).

Usando a EI-MNBC, implementou-se o SE-MNBC (Sistema de Extração de Informações em Múltiplos Níveis Baseado em Conhecimento) que foi aplicado sobre o sistema ETO (*Electronic Trading Opportunities*). O sistema ETO permite que as empresas negociem através da troca de e-mails e o SE-MNBC extrai as informações relevantes nessas mensagens. A aplicação é estruturada em três fases: análise estrutural dos textos, identificação do assunto (domínio) de cada texto e extração, transformando a informação não estruturada em uma base de dados estruturada.

Palavras-Chave: Extração de Informações – EI, Recuperação de Informações – RI, Sistemas de Informação, Gerenciamento de Informações, Sistemas Baseados em Conhecimento, Comércio Eletrônico.

TITLE: "INFORMATION EXTRACTION METHODOLOGY FOR THE TREATMENT AND MANAGEMENT OF E-COMMERCE INFORMATION"

Abstract

The great availability of information offers an enormous potential for trade. However, the huge volume and diversity of opportunities generate a problem: commercial limitations created by the selection and the manual manipulation of the available information. The treatment of greatly unstructured and/or semi-structured databases (U/SSDB), as it happens on the Internet, is a rich and trustworthy source of knowledge, allowing market analysis. Treated and structured information will lead to better management and more successful decision-making, creating competitive differentials demanded by the market.

Researches in Information Retrieval (IR), the result of which is the present thesis, improve the competitive position of small and medium-sized companies, companies which e-commerce has inserted in a globalized, dynamic and very competitive market. The goal of this work is the development of Information Extraction Methodology for the Treatment and Management of E-Commerce Information. We will call this methodology KBIE-ML, or also, Knowledge Based Information Extraction on Multiple Layers. Compared to the manual search and reading time of the data, or to the use of inadequate automatic tools, the users of the KBIE-ML technology can get the desired information faster. So, problems of volume, diversity of storage formats, different search needs, among others, will be solved.

The KBIE-ML methodology uses a set of IR knowledge, combining Document Retrieval, Information Extraction and Data Mining technologies in a hybrid approach for the U/SSDB treatment. A new integration form (multiple levels) and configuration (Knowledge Based Systems - KBS) of information extraction processes is taken into account, creating a more effective and efficient treatment of U/SSDB used in e-commerce. This treatment will make it possible to use structured data manipulation tools, such as Data Base Management Systems, on the previously unstructured information.

The search of knowledge in unstructured textual databases demands the understanding of the stored data. The goal is to emphasize the superficial cognitive aspects involved in text reading, understanding how human beings retrieve information and code the reader knowledge as rules to guide the extraction process. The KBIE-ML process framework is similar to a KBS: the extraction modules (*Inference Machines*) analyze the input text files (*Events*) according to the content of the *Knowledge Bases*, interpreting the rules. As a result, an output file (*Conclusions*) is generated.

The KBIE-ML System was implemented using the developed methodology. It was applied on ETO System (Electronic Trading Opportunities). The ETO system enables companies to exchange trade information by e-mail, and KBIE-ML System extracts relevant information from the messages. The application is structured into three phases: structural text analyses, subject (domain) identification of each text, and extraction, transforming the unstructured information into a structured database.

Keywords: Information Extraction, Information Retrieval, Information Systems, Information Management, Knowledge Based Systems, Knowledge Database Discovery, E-Commerce.

1 Introdução

A análise do desenvolvimento da tecnologia de processamento de informações, segundo uma perspectiva histórica, mostra-nos que a Informática foi uma das áreas do conhecimento humano de maior crescimento. Como conseqüência direta dessa nova realidade, ocorreu uma rápida difusão da Informática em diversos setores da vida social e econômica, como universidades, escolas, empresas, etc. Verificou-se, assim, uma modificação no perfil das pessoas que utilizam o computador. Um público que anteriormente era constituído apenas por especialistas (engenheiros, físicos, matemáticos, etc.), passou a ser integrado por indivíduos não intimamente ligados à área de Informática [SCA 2000].

A informação é considerada um valioso recurso, e o uso da Informática apresenta muitos benefícios, entre eles: melhora na comunicação, maior precisão e respostas mais rápidas. Esta área provê processos formados por pessoas, programas e outros recursos que coletam, transformam e disseminam informação. Basicamente, um sistema de informações é responsável por transformar dados (fatos) em informação (dados colocados em um contexto significativo para um usuário final) de forma que essa possa ser utilizada. [WHA 2002].

Em princípio, um sistema de informações não necessita ser computadorizado, mas a utilização de computadores pode diminuir o custo da manipulação de dados. Os componentes que integram um sistema podem ser vistos como um espectro de disciplinas, desde as mais estruturadas, como engenharia e lógica, passando por aplicações comerciais (estatística e economia), até atingir a compreensão humana (psicologia cognitiva e comportamento). Algumas atividades são bastante mecânicas, como capturar dados, outras são lógicas, como programar um computador, e outras ainda estão num escopo de problemas pouco tratáveis, como “manter um projeto dentro do orçamento” e “manipular a resistência a mudanças”. [WHA 2002].

Dentro da área de Sistemas de Informação, uma sub-área que merece especial atenção, pois trata do problema de volume, diversidade e dinamismo das informações, é a de Recuperação de Informação (RI). RI engloba diversas sub-áreas menores, dentre as quais estão Recuperação de Documentos (RD), Extração de Informação (EI) e Mineração de Dados (MD) [SCA 2001]. Cabe salientar que alguns autores tratam os termos “Recuperação de Informação” e “Recuperação de Documentos” como sinônimos. Neste documento, trataremos RD como uma sub-área de RI, que abrange a recuperação de informações a partir de bases não estruturadas ou textos.

O crescimento das áreas de RD, EI e MD está relacionado a outro setor que nas últimas décadas, ao lado da Informática, experimentou um grande desenvolvimento e expansão: a tecnologia de redes de computadores (*hardware* e *software*) [SCA 2000]. Esse fato fez com que a visão de conceber os computadores como máquinas isoladas umas das outras, seja hoje um conceito superado [WYA 95]. Desta forma, os computadores são interligados por redes locais, metropolitanas ou de longa distância, constituindo-se em usuários e provedores de serviços, cada vez mais complexos e sofisticados.

Um ambiente real, onde podemos observar a aplicação prática dos conceitos e avanços acima descritos, é a rede Internet. Essa rede, criada no início da década de setenta, tem se expandido enormemente nos últimos anos, congregando atualmente milhões de usuários e cerca de 100 milhões de servidores em todo o mundo [MAT

2000]. Em média, a Internet dobra de tamanho aproximadamente a cada intervalo de doze a quinze meses [DVO 96].

Por outro lado, o mais importante aspecto a ser enfatizado a respeito da Internet não é o número de usuários, mas sim a enorme variedade e quantidade de informações e serviços, a eles disponibilizados por esta rede. Centenas de organizações, tais como universidades, institutos de pesquisa e empresas, entre outras, colocam em certos computadores, conectados à Internet, chamados de servidores, documentos, artigos e programas de diversos tipos e áreas. Desta maneira, o público em geral pode ter acesso a este conhecimento, antes mesmo que este seja apresentado em um congresso ou publicado em um periódico.

O uso deste meio de comunicação interativo que é a Internet para fins comerciais, o comércio eletrônico (*e-commerce*), tem crescido muito rapidamente. Estima-se que as vendas pela Internet possam atingir 327 bilhões de dólares por volta do ano 2002 [KOT 2000]. Esse crescimento está associado a um conjunto de características específicas deste mercado eletrônico que torna a compra em casa mais divertida, conveniente e prática. Economiza tempo e apresenta aos clientes uma variedade maior de produtos. Os clientes podem fazer comparações entre produtos, analisando catálogos eletrônicos e procurando os serviços de compras *on-line*. Os clientes empresariais também se beneficiam, conhecendo os produtos e os serviços disponíveis sem gastar tempo com visitas de vendedores.

Esta diversidade e quantidade de informações comerciais são manipuladas atualmente por um conjunto de ferramentas que ajudam as pessoas a procurar, localizar e recuperar informações, tais como Yahoo!, WAIS (*Wide Area Information Server*) e Altavista, que permitem acessar bancos de dados de todas as partes da Internet [WYA 95]. A manipulação é realizada a um nível macro, nível de rede, selecionando, dentre a grande variedade e volume de assuntos, um subconjunto dos dados oferecidos [SCA 97a]. No entanto, os dados selecionados e recuperados ainda se mantêm volumosos para tratamento a nível micro, nível local.

Existe muita dificuldade no manuseio das informações armazenadas em computadores locais (usuários domésticos ou empresas) [SCA 97]. Isso ocorre devido às ferramentas de seleção e do próprio volume de informações existente na rede global. Tais características não permitem uma seleção mais restritiva dos dados armazenados na Internet sem perda de informações importantes, acarretando na recuperação de grandes volumes de dados, muitos desses sem interesse para o usuário. Além disso, a análise exaustiva e empírica das informações recuperadas normalmente é impossível de ser realizada sobre megabytes ou até gigabytes de dados existentes nas bases de dados selecionadas [MAT 93]. Nesse ambiente, é praticamente impossível manter o controle da validade temporal dos dados disponíveis. Várias informações como preços, estatísticas, convites, etc. são atualizadas periodicamente, fazendo com que muitas vezes dados obsoletos sejam utilizados [SCA 97a]. O resultado é o desperdício de oportunidades, como a perda do prazo de envio de artigos para congressos ou de prazos para propostas comerciais.

1.1 Trabalho Desenvolvido

Uma solução tecnológica de apoio ao usuário é necessária para tratar as informações resultantes de processos de comércio eletrônico. Esta solução deve levar em conta o volume, a diversidade e o dinamismo das informações, bem como as necessidades específicas de cada usuário. Para atender esta demanda, propõe-se uma metodologia de extração de informações para auxiliar no tratamento e na manipulação de

informações de comércio eletrônico a partir de bases de dados não estruturadas e semi-estruturadas (BDNE/SE). Tais bases formam grande parte do conjunto de dados de transações comerciais contidos na Internet: ofertas, demandas, descrições de produtos e serviços, anúncios, etc. Para tal, serão utilizadas técnicas de RD, EI e MD. Muitos sistemas têm utilizado estas técnicas, porém, de forma isolada [SCA 2002a]. Propomos utilizá-las de forma unificada, combinando-as conforme as necessidades dos usuários. Sua integração ocorrerá de forma distribuída e equilibrada, visando uma boa relação entre o custo de processamento, a velocidade e a qualidade dos dados extraídos.

A metodologia desenvolvida é de ampla aplicação, podendo tratar diversos tipos de informações e necessidades, através da parametrização dos módulos de processamento de informação e sua combinação. Contudo, o enfoque em comércio eletrônico apresenta resultados práticos, com a definição de um conjunto de processos para atender esse domínio de aplicação e a avaliação da metodologia através de testes sobre dados comerciais. As BDNE/SE são o ponto de partida do processo de extração em comércio eletrônico:

(I) Um processo de classificação estrutural dos dados de entrada é realizado a fim de:

(a) filtrar os arquivos a serem tratados;

(b) dividir arquivos agrupados, tais como diversos arquivos armazenados em um mesmo arquivo físico compactado;

(c) converter os arquivos para um formato de armazenamento padrão (ASCII);

(d) agrupar os arquivos com estruturas internas de armazenamento de dados semelhantes.

(II) Após a classificação estrutural, ocorre a classificação por domínio, separando os arquivos de entrada em classes de assuntos definidas pelo usuário, indexando-os por domínio.

(III) Posteriormente, os segmentos de texto relevantes dos arquivos de entrada são identificados e extraídos, sendo armazenados de forma estruturada. Sistemas de mineração de dados ou de gerenciamento de bancos de dados podem ser utilizados para análise e manipulação das informações agora estruturadas e classificadas.

1.2 Estrutura da Tese

A tese está estruturada conforme as fases de uma metodologia científica. O primeiro capítulo apresenta e contextualiza dentro da Informática o assunto da tese, EI e áreas correlatas. A fim de focar o assunto, temos como tema: extração de informações em comércio eletrônico. O segundo capítulo apresenta a problemática que envolve o tema e a definição do objetivo com seus benefícios e vantagens. No terceiro capítulo é apresentado o embasamento conceitual que sustenta o desenvolvimento da hipótese em EI, descrita no quarto capítulo. O quinto capítulo descreve o sistema protótipo desenvolvido a partir da hipótese: metodologia para a extração de informações não estruturadas. Esse sistema serviu de ferramenta para o processo de avaliação aplicado a comércio eletrônico, apresentado, juntamente com o plano de pesquisa e os resultados quantitativos obtidos, no sexto capítulo. A análise qualitativa e as conclusões são colocadas no sétimo capítulo, juntamente com os trabalhos futuros propostos. O primeiro anexo apresenta as diversas edições do principal congresso da área de EI, e o segundo o estudo realizado de sistemas relacionados ao desenvolvimento da hipótese.

2 Visão Geral

Não obstante às vantagens da divulgação do conhecimento pelas redes de computadores de longa distância, depara-se com um conjunto de problemas criados pela quantidade, diversidade e dinamismo das informações. Em busca de um melhor aproveitamento dos recursos disponíveis, os usuários de computador anseiam por soluções eficazes para tais problemas.

2.1 Identificação do Problema

A seguir são descritos detalhadamente os problemas que motivaram o desenvolvimento deste trabalho.

2.1.1 Quantidade

Existe uma grande quantidade de informação livre na Internet, a qual está crescendo rapidamente [BOW 94]. Foram desenvolvidas ferramentas que auxiliam na localização e recuperação de informações, tais como GOPHER, WAIS (*Wide Area Information Server*), Altavista e Yahoo!, os quais acessam bancos de dados de todas as partes da Internet [DVO 96]. Tais ferramentas permitem ao usuário informar uma ou mais palavras-chave, procurando os documentos que contenham essas palavras [FER 94]. Mesmo com o auxílio dessas ferramentas, ao realizar pesquisas sobre assuntos específicos, o usuário ainda precisa manipular em seu computador pessoal uma grande quantidade de informação [SCA 2000]. Isso ocorre devido às ferramentas de seleção e ao próprio volume de informação existente na rede. Tais características não permitem uma seleção mais restritiva dos dados armazenados na Internet sem perda de informações importantes, acarretando a recuperação de grandes volumes de dados, muitos desses sem interesse para o usuário [SCA 97a]. Além disso, a quantidade de informação no nível local também é elevada por arquivos trazidos por ferramentas de FTP (*File Transfer Protocol*), ou de E-Mail (*Eletronic Mail*).

2.1.2 Diversidade de Estruturas

Analogamente ao enorme volume de dados, existe uma grande diversidade de assuntos e padrões de transferência e armazenamento da informação, criando os mais heterogêneos ambientes de pesquisa e consulta de dados [BOW 94]. Cada um desses ambientes visa oferecer um determinado conjunto de facilidades e vantagens ao usuário, como é claramente destacado no uso de ferramentas de consulta que utilizam o padrão HTTP (*Hypertext Transfer Protocol*), como, por exemplo, Internet Explorer.

Por outro lado, esta variedade de formatos de dados traz alguns problemas. O usuário deve conhecer um conjunto de padrões a fim de obter a informação desejada [WYA 95]. É necessário utilizar várias ferramentas de consulta, como, por exemplo, um visualizador como Netscape ou Internet Explorer, para arquivos em HTML (*Hypertext Markup Language*), e *Outlook* para arquivos de correio eletrônico. Esta variedade cria uma descentralização da consulta, pois não existe uma única ferramenta para manipular os diversos formatos de armazenamento dos dados sob análise [BOW 94]. Um usuário, ao utilizar ferramentas como WAIS ou Alta Vista para encontrar as informações desejadas obterá diferentes resultados de pesquisa. Assim, um estudo aprofundado e

completo precisa utilizar várias ferramentas de recuperação de informações na rede [SCA 97].

No entanto, a maior dificuldade de manipulação está ligada aos formatos de armazenamento não estruturados ou pouco estruturados, como, por exemplo, arquivos texto, *e-mails* (correspondência eletrônica) e artigos de *news* (jornais eletrônicos) [SCA 2002]. Nesses formatos, o entendimento do documento exige a leitura completa do mesmo pelo usuário, o que muitas vezes acarreta um gasto de tempo desnecessário, pois alguns podem não ser de interesse desse ou, então, ser de interesse, mas sua leitura completa só seria útil posteriormente. Além disso, a não existência de uma estrutura formal pré-definida, estimula a falta de padronização da informação dentro do documento, fazendo com que essas sejam organizadas de acordo com o capricho individual de cada autor [HAR 93]. Desta forma, o uso limitado de padrões estruturais acaba dificultando o entendimento e a seleção dos dados disponíveis.

2.1.3 Validade

Várias informações, como chamadas de artigos para congressos, preços de produtos e estatísticas econômicas, apresentam validade temporal (período de validade da informação) [SCA 97a]. Outras informações são atualizadas periodicamente.

Muitas dessas características temporais são explícitas, outras estão implícitas no meio de outros tipos de dados [EDE 94]. Isso torna difícil a recuperação de tal tipo de informação, gerando, várias vezes, a utilização de informações desatualizadas, ou a perda de oportunidades. Como exemplo, temos o caso de informações sobre chamadas de artigos para congressos, onde dados, como a data limite de envio de artigos, são de extrema importância. Nesse caso, a informação temporal está implícita no texto, sem uma localização padronizada, o que facilitaria sua recuperação.

No caso de um *e-mail*, por outro lado, a data da última modificação é colocada de forma explícita no cabeçalho do arquivo. No entanto, essa data só apresenta de forma consistente o tempo de transação, ou seja, quando foi atualizado o banco de dados [EDE 94]. O tempo de validade (período em que a informação armazenada é válida), contudo, pode ser inconsistente, pois não está explicitado e a informação contida no *e-mail* pode estar sem validade.

As características temporais são importantes para os dados que trafegam pela Internet, destacando-se a velocidade com que esta informação é atualizada [BEC 97a]. Há alguns anos atrás, por exemplo, a produção de artigos era menor, permitindo aos pesquisadores manterem arquivos com resumos e referências dos artigos de seu interesse. Atualmente, tal atividade é inviável, pois a velocidade de atualização dos dados cresceu proporcionalmente ao volume [SCA 2000]. O prazo de validade da informação encurtou, exigindo freqüentes atualizações das referências e um maior dinamismo por parte do pesquisador, o que, na maioria das vezes, é impossível.

2.1.4 Caches

Outra característica envolvida na problemática são os mecanismos de *cache* utilizados na Internet. *Caches* são estruturas locais que mantêm cópias de dados acessados repetidamente, evitando uma nova busca nos servidores e diminuindo o tempo de resposta e o tráfego na rede [BOW 94]. Realizou-se um intenso trabalho na área de servidores de arquivos e de sistemas operacionais para suportarem essa replicação de dados [HOW 88]. Um exemplo é a ferramenta WAIS, um servidor de

dados com índices temáticos sobre as informações disponíveis, armazenando as referências de forma centralizada [MAR 92]. Contudo, a sincronização entre o conteúdo dos *caches* e os dados originais gera um grande *overhead* de tráfego de comunicação [HOW 88]. Sendo assim, muitas referências ficam desatualizadas e sem validade. Desta forma, o usuário acaba, várias vezes, manipulando informações inválidas.

Atualmente, o volume de dados armazenado localmente é muito grande e também pode ser considerado uma *cache*, pois constitui cópias de arquivos remotos. Os dados locais correspondem a instantâneos (*snapshots*) de dados na rede, perdendo a validade por apresentarem assincronia em relação aos dados originais [SCA 97].

2.1.5 Diversidade de Usuários

Paralelamente ao crescente número de usuários de computadores, a diversidade de conhecimentos, experiências e interesses das pessoas é grande, ocasionando necessidades de dados e de ferramentas de pesquisa diversas [MAT 2000]. A maioria dessas pessoas tem pouco conhecimento sobre as estruturas de armazenamento e os métodos de recuperação e pesquisa de dados [TRE 2000]. Logo, o número de pessoas com dificuldade de encontrar informações que necessitam é cada vez maior.

2.2 Objetivo

Transmitir e gerenciar informações de forma precisa é importante para o desenvolvimento das empresas, ampliando sua eficiência competitiva. A grande disponibilidade de informações oferece um amplo potencial de oportunidades comerciais. Contudo, o enorme volume e diversidade de oportunidades gera um problema: limitações comerciais criadas pela seleção e a manipulação manual da informação disponível [SCA 2002a]. A maior dificuldade está na não estruturação da maioria dos dados disponíveis, como os trazidos pela Internet, por exemplo. O alto custo envolvido na manipulação e no tratamento das informações torna o completo acesso aos benefícios do comércio eletrônico proibitivo para as pequenas companhias [ELE 2002].

Através de pesquisas em RI, as quais culminaram nesta tese, esforços estão sendo investidos para melhorar a posição competitiva de pequenas e médias empresas (PME), hoje inseridas, através do comércio eletrônico, em um mercado globalizado, dinâmico e competitivo. O objetivo principal deste trabalho é o desenvolvimento de uma Metodologia de Extração de Informações para o Tratamento e Manipulação de Informações de Comércio Eletrônico. Chamaremos essa metodologia de EI-MNBC, ou seja, Extração de Informações em Múltiplos Níveis Baseada em Conhecimento.

O tratamento das grandes bases de dados não estruturadas é uma fonte de geração e verificação de conhecimento rica e confiável, permitindo a análise de mercados: produtos e serviços, concorrentes, fornecedores, público alvo, etc [TRA 2001]. A informação tratada e estruturada permitirá seu melhor gerenciamento, a realização de consultas e a tomada de decisões, criando importantes diferenciais competitivos exigidos pelo mercado. Os usuários da tecnologia EI-MNBC podem obter as informações desejadas de forma rápida, quando comparado ao tempo de pesquisa e leitura manual dos dados, ou com o uso de ferramentas automáticas inadequadas [SCA 2002a]. Dessa forma, problemas de volume, diversidade de formatos de armazenamento, diferentes necessidades de pesquisa, validade temporal das informações, entre outros, serão solucionados de forma fácil e eficaz.

A metodologia EI-MNBC utiliza um conjunto de conhecimentos de RI, combinando as tecnologias de RD, EI e MD, e gerando uma abordagem híbrida para o tratamento de informações. Propõe-se uma nova forma de integração (múltiplos níveis) e configuração (sistema baseado em conhecimento) de processos de extração de informações, criando uma metodologia de tratamento mais eficaz e eficiente de dados não estruturados usados em comércio eletrônico. Esse tratamento viabilizará o uso de ferramentas de manipulação de dados estruturados, como Sistemas Gerenciadores de Bancos de Dados (SGBD), sobre as informações anteriormente desestruturadas.

2.3 Diferenciais e Benefícios

Conforme estudos de diversos sistemas de extração de informações, verificou-se que a maioria deles apresenta uma estrutura funcional semelhante, descrita detalhadamente no capítulo 3 [HOB 2001] [WIL 97]. Essa estrutura é formada por um conjunto de módulos de tratamento de BDNE/SE que geram uma base de dados estruturada com os resultados da extração. A metodologia proposta apresenta os diferenciais acima descritos que agregam à área de EI uma melhor relação qualidade por custo de processamento [SCA 2002].

2.3.1 Nova Proposta de Integração de Processos: Bases de Dados Intermediárias

A EI-MNBC foi desenvolvida para manipular informações de comércio eletrônico através de múltiplos níveis conceituais, preocupando-se com as diferentes necessidades de informações dos usuários a partir de uma mesma base de dados inicial. Estudos anteriores em EI, no entanto, focaram-se no tratamento de informações em um único nível conceitual, o mais primitivo ou o mais alto, conforme evidenciado através dos sistemas participantes do MUC (*Message Understanding Conference*), apresentados no segundo anexo.

Extrair conhecimento em múltiplos níveis conceituais provê um maior espectro de compreensão dos dados, do mais genérico ao mais específico. Alguns usuários querem saber somente o assunto (domínio) de um arquivo não estruturado, enquanto outros desejam selecionar uma informação específica nesses arquivos, armazenando-a em uma base de dados estruturada. Com múltiplos níveis conceituais, descobrem-se e recuperam-se informações em diferentes graus de abstração [SCA 2002].

A EI-MNBC baseia-se em um conjunto de processos individuais e independentes. Muitos sistemas de EI são formados por conjuntos de processos com funções bem definidas e executados linearmente [HOB 2001] [WIL 97]. Contudo, a metodologia proposta usa bases de dados intermediárias (BDI) para conectar esses processos. Assim, o usuário obtém e analisa as informações em estágios intermediários do processamento de dados. O uso de BDI em EI melhora e modifica a tradicional metodologia dos sistemas dessa área [SCA 2002a]. Uma abordagem similar, conhecida como Base de Dados em Múltiplos Níveis, foi descrita para Descoberta de Conhecimento em Textos (*Knowledge Discovery in Text*) [HAN 95] [ZAI 99] [FU 96]. No entanto, tal abordagem generaliza a informação através dos diversos processos de descoberta (o nível mais alto apresenta a informação mais generalizada), enquanto a abordagem EI-MNBC, em EI, especializa a informação, focando-a em domínios específicos de interesse do usuário (os níveis mais altos têm informações mais especializadas) [SCA 2002].

Definiu-se, com base na metodologia EI-MNBC, quatro níveis conceituais para o tratamento e o gerenciamento de informações de comércio eletrônico. A progressiva especialização e transformação dessa informação, nível-a-nível, a partir da base de dados original (nível zero ou primitivo), ocorre até que toda a informação irrelevante seja descartada e a relevante extraída e estruturada. Tecnologias de Bancos de Dados podem ser aplicadas para recuperar e manipular os dados nos níveis mais altos e estruturados [ZAI 99]. É fácil adicionar outros processos a esta estrutura para atender diferentes necessidades da aplicação. Um sistema de Descoberta de Conhecimento, por exemplo, pode explorar os dados extraídos e armazenados no nível final, buscando informações implícitas.

A EI-MNBC define uma extração progressiva de informações, através de uma seqüência de processos que atuam sobre diferentes características dos arquivos de entrada, e usam informações de todos os níveis abaixo do mesmo [SCA 2002a]. Em contraste com outros sistemas de extração (inclusive os analisados no segundo anexo), esta abordagem é não-linear [SCA 2002]. Um processo P_n pode ser executado sem a completa execução de P_{n-1}, utilizando a informação parcial gerada pelos processos prévios e armazenada nas bases de dados inferiores. O processo P3 pode processar os arquivos existentes na base de dados N2, enquanto o processo P2 trata as informações existentes na base N1 e armazena os resultados em N2, por exemplo.

Nos sistemas de extração disponíveis, todos os documentos são processados e todo seu conteúdo é, pelo menos, superficialmente analisado [COW 96]. A metodologia EI-MNBC distribui o custo de processamento entre as diversas etapas (processos) de extração, de forma que processos mais complexos e custosos tratem um volume menor de informação [SCA 2002]. Com base nas necessidades do usuário e na complexidade de cada processo, as etapas iniciais, menos custosas, processam um volume maior de informações, as quais vão sendo especializadas, o que reduz o volume a ser tratado nas próximas etapas, mais custosas.

O uso de BDI apresenta vários benefícios, permitindo a divisão do processo de extração em fases independentes, além de um grande potencial de integração dos módulos de processamento. Cada um desses módulos extrai determinado tipo de informação desejada pelo usuário, ou necessária aos processos posteriores. A divisão do sistema de extração em processos bem definidos possibilita a combinação de diferentes técnicas de RI, EI e MD em um mesmo sistema [SCA 2002a]. Essas técnicas são utilizadas em diferentes momentos, na extração parcial dos dados requeridos. Nessa estrutura, as melhores características de cada técnica de extração são exploradas.

O uso de um sistema modularizado torna o trabalho de desenvolvimento e manutenção menos desgastante e facilita a reutilização [BOR 96]. Códigos fonte amplamente usados, documentados, modificados, testados e aprovados em outros sistemas, aumentam a qualidade e a confiabilidade do sistema perante reais condições de uso [GRA 97]. Além disso, a utilização de partes de programas anteriores é uma chave tecnológica na criação de sistemas com flexibilidade, qualidade e produtividade.

Uma das maiores dificuldades em EI é a portabilidade dos sistemas, visto a difícil adaptação desses aos novos domínios de aplicação [HOB 2001]. A metodologia EI-MNBC viabiliza o fácil encaixe de um novo módulo ao sistema de extração (ver 4.1) [SCA 2002]. Isso permite a rápida adaptação, agregação e/ou modificação do sistema de extração, portando-o para outro ambiente de aplicação.

Segundo a EI-MNBC, os módulos de sistemas de extração são independentes, sendo configurados e executados separadamente. Esta característica agrega uma grande

vantagem para a EI: possibilidade de execução passo-a-passo. Este tipo de execução, embora mais demorada, devido à constante interação com o usuário, permite a análise detalhada dos resultados de cada módulo. Assim, o usuário avalia e configura melhor cada processo. Supera-se, então, uma grande dificuldade dos sistemas de recuperação de informações [WIL 97]: a realimentação. Para a maioria desses sistemas qualquer alteração dos dados de entrada gera uma nova e completa execução do sistema.

As novas características de modularização, execução passo-a-passo e realimentação permitem uma maior redução de ruído, e conseqüente melhora dos resultados de extração. Ruído, em RI significa introdução de erros durante o processamento dos dados [HOB 2001]. Erros são informações irrelevantes ao usuário, que deveriam ser ignoradas pelo sistema, porém são extraídas. Na EI-MNBC, evita-se que um ruído se propague de um processo para os posteriores, alterando suas configurações e melhorando o resultado a ser transferido para os demais [SCA 2002]. Por outro lado, pode-se perder informações relevantes ao configurar o processo para evitar a inserção de ruído. Na metodologia EI-MNBC, caso o resultado de um processo seja insatisfatório, pode-se reconfigurá-lo, generalizando-o e recuperando informações perdidas [SCA 2002].

2.3.2 Nova Proposta de Configuração do Sistema de Extração: Sistema Baseado em Conhecimento

A busca do conhecimento existente em bases de dados textuais não estruturadas demanda a compreensão dos dados armazenados. Um leitor adquire conhecimento a partir de um texto, facilmente e naturalmente, identificando as informações relevantes e memorizando-as. Contudo, automatizar essa atividade é tão complexo quanto construir um sistema que entenda linguagem natural [MOU 92]. Para evitar essa complexidade, a metodologia EI-MNBC simula a atividade do leitor, compreendendo o documento, sem a manipulação de características semânticas profundas [SCA 97].

A EI-MNBC não visa o entendimento de linguagem natural. Ao contrário, através da convergência das técnicas de Sistemas Baseados em Conhecimento (SBC) e EI, apresenta uma maneira automatizada para auxiliar o usuário na extração de informações. O objetivo é enfatizar os aspectos cognitivos envolvidos na leitura de um texto, entendendo como as pessoas recuperam as informações e armazenando esse conhecimento em regras que guiarão o processo de extração [SCA 2002a]. As regras representam conhecimento empírico sobre as características textuais a serem exploradas para selecionar a informação relevante. Sistemas de recuperação e extração de informações freqüentemente requerem a adaptação de modelos de inferência de SBC para a análise de documentos [TUR 91]. A estrutura do processo EI-MNBC é similar a de um sistema baseado de conhecimento: os módulos de extração (*máquinas de inferência*) analisam os documentos (*eventos*) de acordo com o conteúdo das *bases de conhecimento* (BC) definidas pelo usuário, interpretando as regras [SCA 97a]. O resultado é um arquivo estruturado com as informações extraídas (*conclusões*). A representação do conhecimento é realizada através de regras de produção: *SE <condição> ENTÃO <ação>*. Essa forma de representação do conhecimento é genérica e pode ser aplicada na descrição de diferentes domínios de extração [RIC 93].

Cada aplicação de extração envolve uma diferente realidade, e é muito difícil portar e adaptar modelos de extração para novos domínios [WIL 97]. Esse processo freqüentemente requer meses de esforço por parte de especialistas no domínio e lingüistas computacionais familiarizados com o sistema de extração, limitando o

mercado de EI [GRI 97]. Parte do problema de adaptação está na natureza do domínio específico envolvido nessa tarefa: um sistema de extração de informações terá melhor desempenho se as fontes de conhecimento lingüístico estiverem ajustadas para um domínio particular, como, por exemplo, as estruturas léxicas [CAR 97]. Contudo, modificar e adicionar conhecimentos sobre um domínio específico em um sistema existente é custoso e propenso a erros.

A abordagem de SBC oferece flexibilidade e independência de configuração aos processos de extração. Ela facilita a integração e a configuração de diferentes processos. Assim, o sistema pode ser portado e adaptado às necessidades do usuário e ambiente de aplicação: formatos de armazenamento, assuntos, línguas e outras características de arquivos não estruturados [SCA 97a].

Outro problema é viabilizar a configuração dos sistemas de extração pelo usuário final [WIL 97]. É difícil desenvolver ferramentas de customização para esses sistemas, visto a forma e o nível de conhecimento a ser obtido do usuário. O uso de regras de produção torna a configuração dos processos de extração mais natural ao usuário [TRE 2000]. Além disso, é possível re-aproveitar e trocar as BC entre usuários com necessidades semelhantes [SCA 97]. Isso diminui o custo de configuração e viabiliza a permuta de experiência entre usuários.

A EI-MNBC, conforme descrito anteriormente, define a existência de um conjunto de processos individuais e independentes conectados por BDI. Assim, têm-se múltiplas máquinas de inferência associadas a uma mesma BC que guia todo o sistema de extração [SCA 2002a]. A divisão do sistema de extração em diversos processos facilita a criação da BC. Regras mais simples, associadas a processos primitivos, são usadas para tratar documentos em domínios amplos. Regras mais complicadas, para processos de mais alto nível, tratam informações mais complexas em domínios restritos, reduzindo o volume de definições na BC. Assim, o custo de criação e refinamento da BC é distribuído entre os níveis de processamento, na proporção inversa entre o volume de regras e a complexidade dessas [SCA 2002]. Além disso, é possível configurar os módulos de forma independente.

A abordagem de SBC facilita a adaptação de uma interface de configuração do processo de extração baseada em documentos de treino com a geração automática de regras, como para o tratamento de mensagens de correio eletrônico [HAL 2000]. O uso de documentos de treino tem sido usado por outros sistemas. Um dos primeiros locais a experimentar esta abordagem foi a Universidade de Massachusetts. Eles desenvolveram um sistema que usa técnicas de *Machine Learning* para gerar padrões de extração [LEH 94]. Posteriormente, o sistema generaliza e une os padrões derivados dos exemplos individuais, checando se o padrão resultante não *overgenerate* (combinou documentos exemplo que não foram marcados inicialmente como relevantes para o domínio).

A EI-MNBC extrai informações de um texto a partir de características superficiais, como palavras individuais (análise léxica), características contextuais e padrões de sentenças. O uso de técnicas superficiais simplifica o processo de extração e provê um mecanismo efetivo para sistemas em EI [COW 96].

Cada palavra ou termo (conjunto de palavras) extraído é processado como um "*token*" pelas regras na BC. Ele é analisado individualmente perante o contexto em que está inserido, seu padrão de escrita, ou conhecimento semântico armazenado em dicionários [SCA 2002]. Outros padrões contextuais e superficiais completam a análise inicial, aumentando a precisão do processo. O objetivo é encontrar propriedades estruturais de um termo individual no documento. O uso de padrões de contexto e

superficiais é também baseado em uma simples relação entre sentenças ou frases. Durante o processo de extração, um termo é definido com "cabeça" de pesquisa, sendo a partir dele extraídos outros termos relacionados. Isso amplifica o significado atribuído a esse termo, agregando informações estruturais/superficiais relacionadas. O usuário define o escopo das relações entre termos nas regras de extração.

Esta abordagem de extração superficial tem sua precisão melhorada visto a divisão da condição das regras de extração em duas. A primeira, condição de pesquisa (CP), esta associada ao termo "cabeça" e identifica uma informação relevante no documento. A segunda, condição de verificação (CV), confirma a relevância da informação identificada com base em características superficiais e contextuais: outros termos do documento, padrões de escrita, etc. Um exemplo de aplicação da CV seria a extração da informação de origem de uma mensagem de correio eletrônico, identificada pela palavra "*from*" seguida pelo caractere de dois-pontos. Poderia-se usar uma regra cuja CP seja o termo "*from*" e as CVs baseadas na existência do caractere ":", logo após esse termo, e do caractere "@" na mesma linha do documento, identificando um endereço de correio eletrônico. Caso seja encontrada a palavra "*from*", CP verdadeira, ainda devem ser confirmadas as duas CV associadas à pesquisa. Dessa forma, outras informações que poderiam ser identificadas pela palavra "*from*" no documento não serão extraídas por essa regra.

3 Extração de Informação

Várias pessoas e empresas necessitam acessar tipos específicos de informações em diferentes domínios de interesse. Muitas dessas informações, disponíveis em jornais, artigos e outros documentos, estão armazenadas de forma não estruturada [COS 97a]. Tais documentos incluem, normalmente, dados não essenciais à recuperação e compreensão da informação relevante ao usuário. Sistemas de RD são usados para "peneirar" grandes volumes de texto e encontrar os documentos de interesse. Porém, ainda é necessário lê-los para recuperar a informação relevante [LEH 94a].

A área de EI é dedicada ao processamento de dados associados a grandes volumes de texto com informações de um determinado domínio [LEW 96]. EI apresenta enfoque e processos bem definidos cuja tarefa é inerente a domínios específicos [CAR 98]. Um sistema de extração de informações tem como entrada texto irrestrito, "sumarizando-o" conforme um assunto pré-definido ou domínio de interesse. Assim, encontra informação útil sobre o domínio, codificando-a em uma forma estruturada. Segundo Cowie, EI é o nome dado a qualquer processo que seleciona, estrutura e combina dados encontrados em um ou mais textos [COW 96]. As tarefas de EI podem ser realizadas por analistas humanos que lêem a fonte documental e criam entradas para um banco de dados [RIL 94a]. Contudo, esse trabalho é demorado, tedioso e de difícil controle de qualidade, limitando sua aplicação.

Sistemas em EI extraem tipos específicos de informações a partir de bases de dados não estruturadas. Sua vantagem é a segmentação do texto de entrada, permitindo que partes não pertinentes ao domínio, como frases ou orações inteiras, sejam ignoradas. Isso simplifica consideravelmente o processamento, torna-o menos oneroso e reduz a ocorrência de problemas difíceis, como a resolução de ambigüidades [RIL 94]. Pode-se dizer que sistemas de extração são orientados para conteúdos de entrada e saída, o que torna possível administrar avaliações formais e comparações significativas entre sistemas [LEH 94]. Outras vezes, compara-se a produção de um determinado sistema com um banco de dados ideal, normalmente gerado manualmente, para determinar a qualidade dos resultados obtidos. O exemplo de sistema de extração de informações na Figura 3.1 resume histórias de desastres naturais e extrai, para cada evento, o tipo de desastre, quando aconteceu, e dados de qualquer dano material ou humano.

A transformação de informações não estruturadas em estruturadas permite seu processamento por outras aplicações, como pacotes de gerenciadores de bancos de dados tradicionais, aplicações comerciais e sistemas especialistas ou de redes neurais [COS 96]. Isso viabiliza a aplicação de outras áreas da Informática às bases de dados não estruturadas [SMI 97].

O objetivo das pesquisas em EI é construir sistemas que encontrem e recuperem informações relevantes enquanto ignoram as demais (irrelevantes), com precisão [OVE 96]. Cabe salientar que a informação extraída pode ser, muitas vezes, mais valiosa que o texto original [DAL 2000].

A área de EI é correlata à de Processamento da Linguagem Natural (PLN), sendo tratada por alguns autores como sub-área dessa. Contudo, PLN visa a compreensão de textos, analisando orações completas, estruturas gramaticais e outras regras baseadas em linguagem natural [CAR 98]. Por isso, apresenta altos custos

computacionais de *software* e *hardware*, e sua avaliação é altamente problemática. Além disso, ainda inexistem sistemas práticos que gerem análises detalhadas de textos não estruturados [CAR 97].

Com base no comparativo apresentado, um grande número de cientistas de PLN focou suas pesquisas em EI, encontrando resposta para muitas de suas necessidades de aplicação [LEH 94]. Já existem centros de tecnologia em EI com resultados promissores. Contudo, sistemas de extração comerciais ainda são raros. O atual nível tecnológico não permite a rápida construção desses sistemas para novas aplicações, bem como o desenvolvimento de sistemas completamente automáticos [SCA 2000]. No entanto, o fato de um sistema funcional ter uma aplicação imediata, encorajou os investidores a apoiarem pesquisas em EI. No momento, a consolidação dessa área continuará e provocará a existência de sistemas com resultados cada vez melhores.

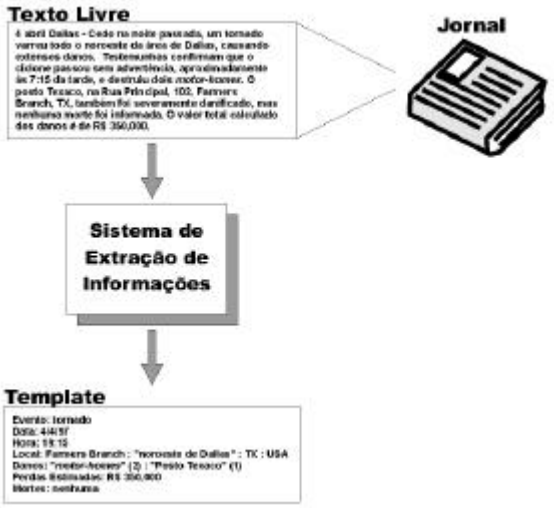

FIGURA 3.1 - Modelo de Sistema de EI no Domínio de Desastres Naturais

Assim, o uso de EI traz diversos benefícios: (**I**) Acarreta uma menor sobrecarga de informações sobre o usuário [COS 97a]. (**II**) Reduz o tempo de extração, se comparado com processos baseados em PLN [SCA 97a]. (**III**) Valoriza o tempo do usuário, utilizando-o fundamentalmente para o processo de tomada de decisão e não para o de extração de dados [COS 96]. (**IV**) Permite o gerenciamento automático das informações extraídas, pois estas, estruturadas, podem ser processadas diretamente por outros programas [LEH 94a]. (**V**) Apresenta qualidade semelhante à humana para a execução da mesma tarefa [OVE 96]. (**VI**) Facilita o controle da qualidade através de processos menos custosos e mais claros de avaliação [LEH 94]. (**VII**) Responde questões não solucionadas através RD [LEH 94a]. (**VIII**) Oferece uma tecnologia aplicável a textos reais, quando comparada com PLN [SCA 97].

3.1 História

Segundo Wilks, EI é uma nova tecnologia, mas não uma nova idéia. Já em 1964 eram encontrados artigos com títulos como "Pesquisa em Textos com *Templates*". Entretanto, o poder computacional da época não permitia que essas idéias fossem concretizadas [WIL 97]. Na década de 70, trabalhos efetivos em EI, como o de Naomi Sager, foram realizados dentro do domínio médico e constituíram um longo período de projeto, combinando análise superficial de sintaxe e o uso de *templates* (base de dados

estruturada formada por campos - *slots* – e usada para armazenar a informação extraída). O trabalho dependia de estruturas de extração criadas manualmente e de um conjunto restrito de técnicas, mas apresentou resultados altamente efetivos, convertendo sumários médicos de alta hospitalar para um formato acessível por ferramentas gerenciadoras de bancos de dados.

Ainda na mesma década, o trabalho FRUMP de Geral deJong, apresentou uma adaptação das estruturas *script* Schank de alto nível, o que nós chamaríamos hoje de cenários ou domínios, definidos por regras *script* de especificação do domínio de extração, preenchendo campos de arquivos estruturados com informação retirada de jornais. Este trabalho era, de muitas formas, presciente e foi qualificado como "brincadeira" em seu tempo, não sendo levado a sério [COW 96]. Ele nunca foi avaliado de forma quantitativa, frente a padrões de avaliação quanto a sua habilidade para recuperar textos e extrair informações desses. FRUMP, mais tarde, se tornou a base para um prematuro sistema comercial chamado ATRANS. Esse último foi desenvolvido para extrair informação de telexes gerados manualmente sobre transferências de dinheiro entre bancos internacionais [COW 96]. O ATRANS demonstrou que o uso de simples técnicas de PLN era adequado para aplicações em EI sob domínios restritos.

Na década de 80, DaSilva e Dwiggins desenvolveram um sistema para extração de informações sobre relatórios de vôos de satélites. Porém, este sistema era restrito a simples sentenças e carecia de uma metodologia para extração de eventos completos. Um trabalho pioneiro em EI foi o de Cowie em 1983, Extração de Estruturas Canônicas a partir de descrições de classificações de plantas e animais.

O primeiro sistema de extração resultante de um complexo problema comercial foi o de Hayes et al., o sistema JASPER do grupo de Carnegie em 1986 [COW 96]. Como os sistemas anteriores, esse apresentava um alto grau de tarefas manuais, e não obteve acesso aos principais recursos lingüísticos externos: textos exemplo/padrões, informações léxicas, dicionários, como o Gazetteers (dicionário geográfico), nem incorporou qualquer algoritmo de aprendizagem. Porém, o sistema foi avaliado e comparado seriamente, inclusive dentro dos regimes do MUC e do TIPSTER, programa de pesquisa do governo dos Estados Unidos em RD e EI [GRI 95].

3.2 Definição do Domínio

Um sistema de extração recupera informações em um universo de dados**,** formado pelos documentos sob análise, conforme o domínio de interesse especificado. Essa especificação delimita o escopo de atuação do sistema e normalmente é baseada em padrões léxicos, sintáticos, contextuais e estruturais, entre outros. Uma boa especificação é ampla o suficiente para extrair a informação relevante, mas restrita o bastante para não se aplicar a domínios impróprios e extrair informações irrelevantes [CAR 98].

Existem diversas formas de especificação de domínio, as mais usadas são linguagens do tipo *script*, expressões regulares e nodos conceito [SCA 97a]. A especificação também pode ser híbrida, valendo-se de distintas qualidades das diversas formas. Técnicas de representação do conhecimento usadas na área de Inteligência Artificial (IA) influenciam na EI. A especificação do domínio pode ser considerada uma BC usada pelo processo de extração, facilitando sua configuração e aumentando o potencial de reutilização do conhecimento [SWA 89]. Ajustes para melhoria da qualidade de extração ou adaptação do domínio são facilitados.

Linguagens *script* não são compiladas, sendo utilizadas diretamente pelo módulo de extração. O conjunto de comandos da linguagem varia conforme o sistema de extração. Dois exemplos importantes de linguagens *script* são as baseadas em sintaxes SQL [HAN 96] e as em regras de produção do tipo condição-ação [SCA 97]. A Figura 3.2 apresenta uma regra produção que, ao encontrar o termo "*from*", extrai a linha onde encontra-se esse termo, armazenando-a no arquivo de saída.

```
SE Palavra('from')
ENTÃO Copia(linha)
```

FIGURA 3.2 - Regra Script do Tipo Produção

O uso de expressões regulares para a especificação de domínio também é muito comum, como é o caso do sistema InfoExtractor [SMI 97]. Nesse sistema, a expressão regular especifica *tokens* gatilho ou palavras-chave para ativação de um conceito de extração. O conceito de extração especifica como extrair a informação quando a expressão é verificada. Quando o *token* especificado é encontrado, extrai-se a informação e abre-se um novo campo no arquivo de saída (*slot*) que é preenchido essa informação. Datas, por exemplo, são normalmente descritas desse modo: "NN/NN/NN | NN/NN/NNNN". O objetivo é reconhecer a expressão regular especificada, para a instanciar o conceito de extração.

A terceira metodologia de especificação é baseada em nodos conceito. Esses nodos são estruturas que extraem informação relevante de uma oração a partir de uma palavra "gatilho", a qual indica a possível presença de uma informação relevante. Contudo, cada nodo tem um conjunto de condições de ativação que especifica o contexto lingüístico que deve existir para sua ativação. Assim, ele só é ativado quando a palavra gatilho aparece e as condições são verificadas.

A extração seletiva de conceitos foi implementada em um analisador conceitual de orações chamado CIRCUS [LEH 91]. O núcleo do sistema CIRCUS é um dicionário de nodos conceito para um domínio específico. O exemplo da Tabela 3.1 apresenta dois nodos conceito usados nesse sistema no domínio de terrorismo, ambos ativados pela palavra "assassinado". O primeiro, $assassino-ativo$, é ativado se o verbo "assassinar" aparecer em uma construção ativa, como: "os terroristas assassinaram o prefeito". O segundo, $assassino-passivo$, só é ativado quando esse verbo aparece na forma passiva, como: "três camponeses foram assassinados por guerrilhas". Um nodo pode ser ativado por diferentes palavras. Por exemplo, $assassino-passivo$ também é ativado pela palavra "morta", sendo ativado também pela frase "três camponeses foram mortos por guerrilhas". Se uma oração contém múltiplas palavras gatilho, o sistema CIRCUS instancia vários nodos conceito para a oração. Se uma oração não contém nenhuma palavra gatilho, não nenhuma informação será extraída dela. Nodos conceito instanciados são a única produção gerada pelo CIRCUS.

Um nodo conceito especifica campos em um arquivo de saída estruturado (*slots*) para armazenar a informação extraída [LEH 91]. Cada *slot* esta associado a um tipo de informação e contém uma expectativa sintática que define onde a informação será encontrada em uma oração. Por exemplo, $assassino-passivo$ contém dois *slots*: um de vítima e outro de predador. Neste sentido, $assassino-passivo$ só é ativado em uma construção passiva. O conceito define vítima como o sujeito do verbo "assassinar" e predador como o objeto da preposição "por".

Cada *slot* também tem um conjunto de obrigações fortes e fracas. Elas especificam preferências semânticas para os tipos de informações que podem corretamente preencher o *slot*. As obrigações fortes devem ser obrigatoriamente

satisfeitas para que o *slot* seja preenchido. Já as obrigações fracas necessariamente não, servindo para indicar preferência em situações onde informações diferentes possam ser extraídas a partir da mesma oração.

TABELA 3.1 - Duas Definições de Nodos Conceito

Nome:	$assassino-ativo$	$assassino-passivo$
Palavra de gatilho:	Assassinado	Assassinado
***Slots* variáveis:**	((predador (*SUBJECT * 1)) (a vítima (*DOBJ * 1)))	((a vítima (*SUBJECT * 1)) (predador (*PP * (ser-prep? ' (por)))))
Obrigações do *slot*:	((predador de classe *SUBJECT *) (a vítima de classe *DOBJ *))	((a vítima de classe *SUBJECT *) (predador de classe *PP *))
***Slots* constantes:**	(assassinato do tipo)	(assassinato do tipo)
Condições de Ativação:	((ativo))	((passivo))

A Figura 3.3 mostra uma oração e o conceito instanciado resultante.

Oração: Três camponeses foram assassinados por guerrilhas.
$ASSASSINO-PASSIVO $
vítima = “três camponeses”
predador = “guerrilhas”

FIGURA 3.3 - Um Nodo Conceito Instanciado

3.3 Recuperação de Documentos e Extração de Informação

RD e EI são áreas complementares, pois as duas podem ser combinadas [SME 97]. Contudo, elas têm objetivos distintos. RD recupera documentos relevantes de uma coleção, mas não extrai informação desses. Um sistema de extração tem a habilidade de extrair a informação relevante dos documentos, de acordo com critérios específicos, e representa-la em estruturas [COS 97a]. Neste sentido, EI trabalha em domínios de aplicação mais específicos que RD [SCA 97a]. Processos de extração reconhecem as entidades utilizadas em textos com precisão, tais como: organizações, pessoas, locais, tempo (horários), valores (dinheiro), etc.

EI estabelece relações entre partes de um documento, gerando informações mais completas, concisas e coerentes [LEH 94a]. A informação é extraída das relações entre fatos individuais. Ao contrário de RD, não é o bastante identificar fatos isolados, o que pode ser feito por simples pesquisa de palavras-chave [LEW 95]. EI interessa-se pela estrutura dos textos, enquanto, do ponto de vista de RD, textos são arquivos de palavras desordenadas. Dessa forma, EI é mais precisa que RD e potencialmente mais eficiente, pois reduz a chance do usuário ler textos irrelevantes [SME 97]. Por outro lado, sistemas de EI são mais difíceis e necessitam de maior conhecimento para serem construídos [SME 97]. Além de serem mais custosos computacionalmente do que sistemas de RD [RIL 94a]. Talvez por isso, a tecnologia de RD tenha precedido a de EI, sendo mais madura.

EI e RD apresentam muitos fundamentos semelhantes, principalmente quanto ao processamento automático de linguagem natural. As técnicas de RD estão fundamentadas na linguagem contida nos documentos pertencentes ao universo de pesquisa e nas consultas do usuário. EI extrai informações de documentos em linguagem natural. Assim, é de interesse para a comunidade de EI ver como uma tarefa relacionada a ambas as áreas, como linguagem natural, tem sido gerenciada por RD.

3.3.1 RD como Ponto de Partida para Extração

EI extrai informação de documentos, normalmente de fontes eletrônicas publicamente disponíveis, como redes de notícias, com o uso do computador. Muitas das aplicações dessa tecnologia são precedidas por uma fase de RD, que seleciona um conjunto de documentos relevantes a partir de alguma consulta do tipo pesquisa de palavras ou termos em documentos [SME 97]. Segundo Cowie, pode-se entender sistemas de recuperação de documentos como uma combinação de ceifadeiras que trazem material útil a partir do vasto campo de material cru [COW 96]. Com o grande potencial de informação útil em mãos, um sistema de extração pode então transformar o material "cru", refinando-o e reduzindo-o, a uma pequena parte desse (Figura 3.4). Dessa forma, EI auxilia RD e permite que a informação seja mais facilmente absorvida e analisada.

Por outro lado, cabe salientar que muitos textos providos por sistemas de recuperação podem ser irrelevantes, gerando resultados de extração inválidos [COW 96]. Por exemplo, em uma pesquisa, sete de mil textos sobre chips de microeletrônica tratavam de batatas fritas ("*potato chips*"). Nesse sentido, alguns textos recuperados necessitam ser excluídos antes das informações serem extraídas.

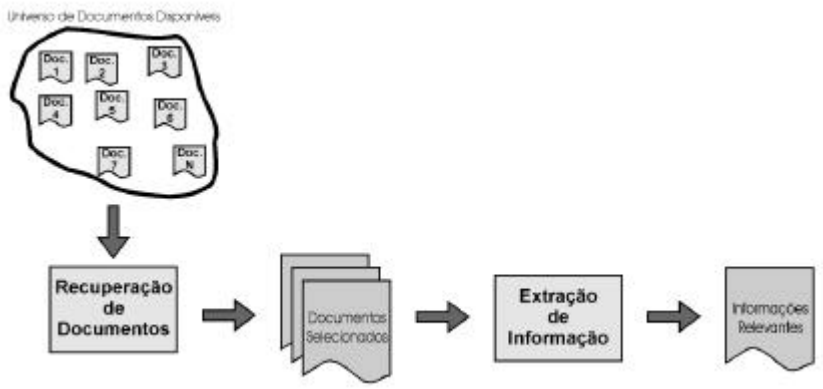

FIGURA 3.4 - RD como Ponto de Partida para EI

3.3.2 EI como Ponto de Partida para Recuperação

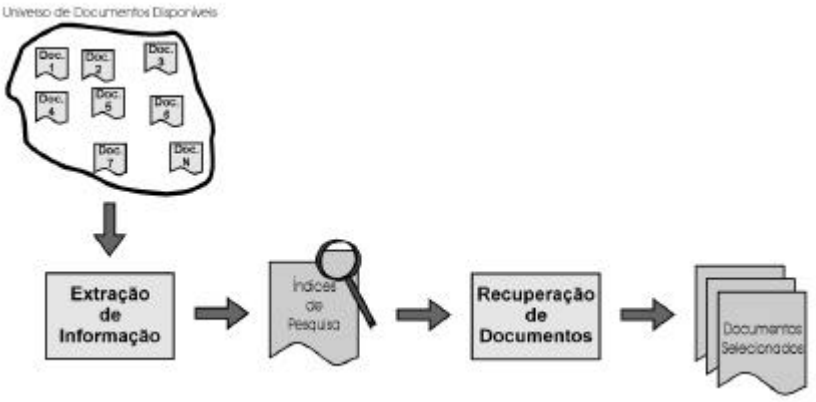

FIGURA 3.5 - EI como Ponto de Partida para RD

EI pode processar documentos extraindo informações relevantes para a melhor indexação desses, como parte componente de um processo de RD [SME 97]. Dessa forma, o processo de extração auxilia na classificação de documentos através da seleção

das palavras-chave utilizadas nos índices de pesquisa, servindo de base para a aplicação das técnicas de RD (Figura 3.5). Usando EI, pode-se superar muitas das limitações impostas por técnicas baseadas em palavras-chave e termos. Em particular, no estudo realizado por Riloff, a alta precisão de recuperação é alcançada devido à classificação sensível ao contexto [RIL 94a]. As frases e os contextos lingüísticos são reconhecidos e manipulados pelo sistema. Além disso, esse sistema pode classificar textos que seriam inacessíveis com o uso de técnicas baseadas em palavras, visto que podem não conter qualquer palavra-chave associada ao domínio de recuperação. Por outro lado, o uso de técnicas de EI é útil ao processo de recuperação, mas ainda apresenta limitações quanto à amplitude do domínio de extração e o volume de documentos [SCA 2000].

3.3.3 Recuperação e Extração em um Mesmo Nível

Processos de EI podem existir no mesmo nível que processos de RD em uma aplicação [SME 97]. A combinação de tais processos pode ser controlada pelo próprio usuário ou automaticamente, a fim de melhor usar as características de cada técnica (Figura 3.6). Um exemplo de aplicação seria um *browser* de pesquisa na Internet, onde o processo de recuperação utiliza recursos de EI para melhor selecionar os documentos, e processos de EI são aplicados sobre os documentos recuperados, extraindo de cada um a informação desejada. Essa informação será armazenada e apresentada ao usuário de forma precisa e estruturada, podendo ser facilmente exportada para outras ferramentas de manipulação de dados. Além disso, refinamentos da consulta podem ser realizados valendo-se das características inerentes a cada área.

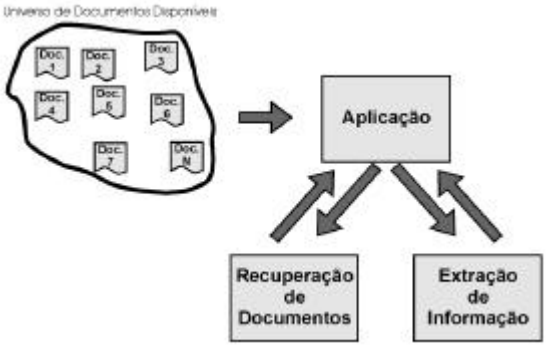

FIGURA 3.6 - RD e EI Atuando em um Mesmo Nível

3.4 Extração e Descoberta de Informação em Textos

Mineração de Dados (MD) ou Descoberta de Conhecimento em Bases de Dados (*Knowledge Discovery in Databases*) enfoca a exploração computadorizada de grandes volumes de dados, visando à descoberta de padrões relevantes e interessantes existentes nesses dados [FEL 99]. Métodos de MD têm sido aplicados a uma grande variedade de domínios, desde dados de vendas em supermercados, até análise de fotos de satélite.

Muito do trabalho de MD esta concentrado em bancos de dados estruturados, mas também existe uma demanda para a mineração de dados na forma de textos não estruturados. Surpreendentemente, somente um pequeno conjunto de exemplos de sistemas em Mineração em Dados em Textos (MDT), ou Descoberta de Conhecimento em Textos (*Knowledge Discovery in Text*), está disponível. MD tende a ser muito matemática, e a maioria das pesquisas nesse campo se preocupa somente com extração de conhecimento de bancos de dados [DIX 97]. Por isto, muitas das técnicas e teorias

não se prestam livremente para MDT. Por outro lado, considerando o volume de coleções de documentos existente, MDT é uma área de pesquisa muito útil.

MDT é o processo de encontrar padrões e informações implícitas interessantes ou úteis em um conjunto de informações textuais não estruturadas [LOH 99]. MDT combina técnicas de EI, RD e PLN com os métodos de MD. O principal uso de MDT é extrair conhecimento previamente desconhecido armazenado em um volume de texto [FEL 95]. Muito da literatura sobre MDT ainda está associada com EI e outras áreas relacionadas e seu caráter inovador pode ser atribuído ao nível de poder computacional requerido pelas aplicações nessa área [WIL 97]. Coleções de dados interessantes estão tipicamente na ordem de gigabytes de dados. Isto torna o processamento desses dados muito caro. Contudo, muita pesquisa tem sido dirigida atualmente para melhorar a eficiência dos algoritmos de mineração em textos.

Os dados estruturados, armazenados em Bancos de Dados (BD), são mais fáceis de serem tratados por meios computacionais, porque existem linguagens formais, como SQL e QBE, que permitem sua manipulação e consulta de forma mais concisa e precisa [LOH 99]. Já os dados não estruturados necessitam de mecanismos computacionais diferentes dos tradicionalmente usados, para que possam ser manipulados e consultados.

Para aplicar métodos tradicionais de MD sobre textos, é necessário impor alguma estrutura para os dados [DIX 97]. Ou seja, algum ser humano deve definir a estrutura desses dados, coletá-los e armazená-los num BD convencional. Esse processo necessita de apoio automatizado, pois é difícil, tedioso e sujeito a erros se feito por pessoas. Isso porque o volume de dados é enorme e existe dificuldade em se representar digitalmente estas formas não convencionais de dados [LOH 99]. O primeiro passo nesse processo é decidir qual estrutura impor aos dados. Para fazer isso, deve-se considerar muito cuidadosamente os processos posteriores de descoberta a serem usados. Dada as fortes limitações da tecnologia atual no processamento de textos, nós necessitamos definir normalmente estruturas simples que possam ser extraídas a partir dos textos de forma automática e com baixos custos [FEL 95]. Por outro lado, a estrutura deve ser boa o suficiente para permitir a execução das operações de MD de forma interessante.

Dixon destaca a fase de pré-processamento como uma fase crucial para MDT, influenciando a mineração de dados conforme o processamento inicial do texto [DIX 97]. MDT está bastante relacionada com as áreas de RD e EI, e pode-se considerar que sistemas de mineração usam componentes que executam tarefas dessas áreas [FEL 99]. Segundo Dixon, a melhor visão de um sistema de mineração seria como apresentado na sucessão de passos abaixo [DIX 97]. Alguns pesquisadores apresentam as fases de recuperação e extração combinadas em uma única fase de pré-processamento.

1. Recuperação de Documentos: localiza e recupera documentos considerados relevantes ao usuário.

2. Extração de Informação: extrai dos documentos selecionados a informação que o usuário especificou como relevantes, armazenando-as de forma estruturada, normalmente um BD.

3. Mineração de Dados: descobre padrões de dados no BD gerado anteriormente com o uso de técnicas MD padrão.

4. Interpretação: encontra uma interpretação para os padrões descobertos. Idealmente, a interpretação deveria ser em formato de linguagem natural.

O resultado estruturado gerado pela aplicação de um sistema de extração sobre um conjunto de textos, viabiliza o uso de técnicas de MD sobre a saída desse

processamento [COS 97]. Além disso, o volume de informação a ser processado pelo sistema de MD é menor, concentrando-se nos dados relevantes. Segundo Cowie, minerar *templates* refere-se a qualquer esforço que sistematicamente explora *templates* fonte gerados durante uma das fases do processamento de um sistema MDT [COW 96]. Um certo número de sistemas apresentados no MUC tem prestado a atenção nessa tarefa. Por exemplo, extrair de textos todas as frases que descrevam atentados a bomba e organizações terroristas, e usar estatísticas de descoberta de conhecimento para estabelecer correlações e distribuições de probabilidade dos dados. Um dos grandes benefícios de EI esta na grande quantidade de sistemas que podem valer-se de seus resultados, e a importância dessa área reside na redução do esforço para aquisição de conhecimento.

Um problema que pode ocorrer em processos de extração é a falta de qualidade das informações extraídas, atingindo conseqüentemente o processo de descoberta. Muitas vezes, são encontradas informações redundantes ou contraditórias no universo dos documentos recuperados e dos quais são extraídas informações. Loh aconselha que não usar informações baseadas em apenas um documento, pois esta informação pode estar desatualizada [LOH 99]. Entretanto, deve-se também ter cuidado com informações extraídas de vários documentos, pois podem existir contradições. McKeown trata parcialmente desse problema [RAD 98]. Seu trabalho apresenta técnicas para analisar diversos artigos sobre um mesmo evento e criar um resumo em linguagem natural. A aplicação dessa técnica servirá para encontrar padrões de informações, informações implícitas e discrepâncias entre documentos textuais [LOH 99].

Feldman mostra como sistemas em MDT estão utilizando uma técnica simples de EI, denominada de categorização de segmentos de texto a partir suas características principais [FEL 95]. Tal método se mostra simples, robusto e fácil de produzir. Nesse processo, é estabelecido um conjunto de conceitos centrais (categorias) para um texto, sendo permitida uma visualização hierarquicamente ordenada dos conceitos, e a mineração por relações entre documentos e entre conceitos. O sistema de mineração de dados em textos Document Explorer constrói um banco de dados a partir de uma coleção de documentos, e aplica técnicas de mineração baseadas em categorias [FEL 95].

EI é o componente mais importante do processo de MDT [FEL 99]. Alguns artigos tratam os dois processos como a mesma tecnologia, entretanto eles não são equivalentes. MDT trata alguns problemas não contemplados por EI. Sistemas de descoberta buscam deduzir um conjunto de regras ou um modelo de domínio com base no texto. Isto prevê um forte uso de técnicas de aprendizado de máquina, além dos componentes de extração existentes em EI [COW 96]. A BC que se espera extrair, normalmente é designada para um sistema especialista ou sistemas baseados em casos. MDT é mais ambiciosa quanto ao entendimento do texto que EI, cujo objetivo se restringe a extrair informações existentes no texto, sem deduzir novas ou encontrar padrões e informações implícitas em um conjunto de informações textuais não estruturadas [LOH 99].

3.5 Método de Sistemas de Extração de Informações

No início da pesquisa em EI, os sistemas variavam bastante em suas abordagens de extração de informações. Conforme Cardie, de um lado estavam os sistemas que processavam os textos usando técnicas tradicionais de PLN: primeiramente, a análise sintática completa de cada sentença, então a análise semântica das estruturas sintáticas

resultantes, e finalmente a análise de discurso das representações sintáticas e semânticas [CAR 98]. No outro extremo, ficavam os sistemas que usavam técnicas de identificação de padrões de léxicos, estruturais e contextuais, com raro uso de análise lingüística dos textos de entrada (ver 3.5.1). No entanto, com o passar dos anos, os pesquisadores começaram a convergir para uma metodologia padrão de sistemas em EI.

A área de EI, além da base de PLN, cresceu explorando e unindo-se a uma abordagem de análise lingüística mais empírica e baseada em textos. Assim, dá-se menos ênfase à teoria lingüística tradicional e derivam-se estruturas e vários níveis de generalização lingüística dos documentos [WIL 97]. Esse movimento tem muitas tendências, como o uso de dicionários semânticos (com informações semânticas associadas aos termos), criados a partir de textos exemplo, e recursos como o WordNet, um dicionário semântico em forma de rede [CAR 97]. O movimento empírico juntou forças com a revitalização da área de *Machine Learning*. Grandes quantidades de padrões lingüísticos passaram a ser gerados e, posteriormente, generalizados automaticamente para novos e maiores volumes de textos por algoritmos de extração com técnicas de aprendizado [WIL 97] (ver 3.5.2). As recentes pesquisas em EI enfatizam a importância de várias áreas negligenciadas em PLN, enquanto demonstram que métodos simples são adequados para diversas tarefas envolvidas na análise de textos.

Segundo Wilks, duas características estruturais apresentam importantes papéis no crescimento surpreendente de EI [WIL 97]. A primeira foi a modularização, transformando tarefas lingüísticas computacionais monolíticas em módulos menores e mais simples de utilizar e avaliar. Essa separação pode não corresponder somente às divisões clássicas de níveis lingüísticos, como sintaxe e semântica. A segunda foi uma metodologia de PLN mais flexível. Os módulos anteriormente referidos podem ser combinados de várias maneiras para executar tarefas diferentes, como, por exemplo, extração de informações, mas também tradução computadorizada de textos. Além disso, diferentes módulos para uma mesma tarefa podem ser sistematicamente comparados, ou o mesmo módulo pode ter seu desempenho comparado com base em tipos de textos diferentes.

3.5.1 Simplificação do Processo Sintático

Parsers parciais realizam a análise sintática parcial de sentenças visando a análise semântica e enfocando somente o que é necessário para essa última [HOB 2001]. Proponentes dos analisadores de sentenças semanticamente orientados argumentam que *parsers* parciais são viáveis, mas relativamente poucas pesquisas têm sido conduzidas nesse sentido. Pesquisadores tradicionais da comunidade lingüística tem favorecido às análises sintáticas completas de sentenças, uma visão que tem dominado a área de lingüística computacional nas últimas três décadas.

A análise sintática completa de cada sentença de um texto normalmente apresenta um custo computacional grande para o resultado desejado. O tempo de processamento inviabiliza seu uso em bases textuais reais. Analisadores sintáticos tipicamente operam em tempo polinomial e tendem a “atolar” com sentenças contendo mais de 20 ou 30 palavras [COW 96]. Essas condições têm motivado os pesquisadores a reverem suas exigências por analisadores sintáticos completos.

Existem importantes diferenças entre a necessidade de análise de sentenças de um sistema de extração e os *parsers* tradicionais. O objetivo da análise sintática em um sistema de extração não é produzir uma árvore gramatical completa, detalhada para cada

sentença no texto. O sistema necessita somente realizar um *parsing* parcial, construindo somente a estrutura necessária para o entendimento semântico da sentença [CAR 98].

Segundo Cowie, várias saídas tem sido avaliadas a fim de simplificar o processamento sintático para análise semântica [COW 96]. Alguns pesquisadores têm preservado seus analisadores sintáticos completos enquanto estão trabalhando em etapas anteriores a esse processamento para separar sentenças relevantes de irrelevantes, o que reduz o volume de informação a ser tratada pelo analisador. Outros pesquisadores simplificam seus processos sintáticos para produzirem fragmentos de árvores de análise sintática ou árvores incompletas com as características mais relevantes para uma análise semântica. Por outro lado, existem pesquisadores que abandonaram totalmente seus analisadores sintáticos completos para investir em *parsers* parciais que nunca tiveram a intenção de prover uma análise completa. Um *parser* parcial procura por fragmentos de texto que possam ser reconhecidos como relevantes ao entendimento semântico da sentença, isto é, normalmente grupos de sujeitos e de verbos [WIL 97]. Devido a essa cobertura limitada, um *parser* parcial pode contar unicamente com técnicas de busca de padrões, freqüentemente através de processos de estado-finito, para identificar esses fragmentos baseado em características sintáticas locais [CAR 98]. A partir desse conjunto reduzido de informações sintáticas, normalmente sujeitos e verbos, pode-se aplicar generalizações estatísticas usando conjuntos maiores de textos [WIL 97].

A natureza dessas unidades de texto, ou *tokens*, utilizadas para análises baseadas em padrões, requer que elas não estejam contidas em um dicionário léxico, e que um sistema automático as reconheça, seja através de contexto, ou através de padrões de segmentos de texto [COW 96]. Sistemas de extração devem reconhecer tais construções precisamente. Para algumas aplicações, uma base de dados de segmentos de textos e dicionários léxicos obtidos a partir desses textos provê alcance adequado de informação para o processo de identificação. No MUC são oferecidos aos participantes grandes volumes de informação de nomes próprios e outros recursos padrão, como o Gazzetteer, que contem 250.000 nomes de localidades, e outras listas de nomes de companhias [CAR 97].

Essa riqueza de recursos léxicos, por outro lado, pode causar seus próprios problemas, como ambigüidades léxicas aplicadas a nomes próprios [COW 96]. Para textos em inglês ou português, por exemplo, informações com letras maiúsculas ou minúsculas podem ajudar na identificação de nomes próprios, mas muitos artigos de jornais eletrônicos continuam sendo transmitidos somente com caracteres maiúsculos. Além disso, alguns delimitadores como títulos (exemplo: Sr. e Dr.) e designações para companhias (exemplo: S.A. e LTDA.) podem permitir que palavras desconhecidas, em conjunção com outros indicadores, sejam parte de um padrão de nome próprio. A performance dos processos de identificação de padrões aparece na faixa de 40% a 90%, dependendo do domínio e das técnicas usadas [COW 96].

Cabe salientar também que *parsers* parciais são bem apropriados para aplicações de extração de informações por uma razão adicional: as decisões de resolução de ambigüidades, que fazem o desenvolvimento do *parser* difícil, podem ser postergadas até os últimos estágios do processamento, onde uma perspectiva *top-down*, a partir da tarefa de extração, pode guiar as ações do sistema. Nesse sentido, características superficiais, mais simples de serem extraídas e processadas, têm sido usadas para prover um efetivo mecanismo para o desenvolvimento de sistemas com *parsers* parciais [COW 96].

3.5.2 Aprendizado de Linguagem Baseada em Textos Exemplo

Um assunto crítico para a área de EI é o trabalho manual exigido para definir o domínio de extração e/ou treinar o sistema em um conjunto apropriado de documentos. O tempo exigido para construir manualmente um dicionário de padrões e/ou regras para a análise lingüística de um domínio é significativo, bem como para alterar esse domínio.

Vários pesquisadores têm investigado o uso de métodos automáticos baseados em conjuntos de textos exemplo para o aprendizado de padrões de extração de informações [WIL 97]. Os métodos de aprendizagem variam em várias dimensões: a classe de padrões a ser definida, o conjunto de textos de treinamento requerido, a quantidade e o tipo de realimentação humana requerida, o grau de pré-processamento necessário, o conhecimento inicial requerido, e os preconceitos inerentes no próprio algoritmo de aprendizagem.

Em geral, algoritmos de aprendizagem de linguagem baseados em textos exemplo têm sido utilizados para prover dados de configuração para componentes individuais dos sistemas de extração e, como resultado, para melhorar a performance de extração de ponta-a-ponta. Em teoria, esses métodos podem ser usados para cada módulo dos sistemas de extração: classificação gramatical, classificação semântica, extração de ambigüidades, *parser* parciais, etc. O segredo é ter muitos dados de treino, ou seja, textos exemplo. Algoritmos de aprendizado de linguagem adquirem uma habilidade particular (conhecimento) no processo de extração utilizando exemplos de como realizar a tarefa de forma correta e generalizando-a a partir de outros exemplos, para posteriormente tratar novos casos [WIL 97]. O algoritmo, portanto, depende criticamente da existência de conjuntos de textos gerados com a apropriada supervisão de informação.

Tratando-se de módulos de sistemas de extração, que são primariamente independentes de domínio, podem ser usados algoritmos de aprendizagem genéricos, que sirvam para a configuração de módulos em diferentes sistemas de extração. Um dos sistemas iniciais a adquirir padrões de extração automaticamente foi o AutoSlog [RIL 93]. Esse sistema aprende padrões de extração em domínios específicos, definindo nodos conceito para uso com o sistema CIRCUS (ver 3.2) e outros. Contudo, essa portabilidade dos sistemas de aprendizagem depende do domínio em que será aplicado o sistema de extração e dos textos usado para treinamento. Quando o primeiro é alterado, mesmo que para domínios correlatos, novos textos de treinamento devem ser criados e usados para retreinar, ou atualizar os conhecimentos do algoritmo de aprendizagem e conseqüentemente do processo de extração [CAR 98]. Definir contextos padrões é difícil e coleções de textos têm sido geradas somente para contextos predefinidos e para um pequeno número de características selecionadas.

Técnicas de aprendizado de linguagem natural são mais difíceis de serem aplicadas aos estágios finais de extração, como aprendizado de padrões de extração, resolução de coreferências, e geração de *templates* [CAR 97]. Primeiro porque não existem usualmente coleções de textos geradas com a apropriada semântica e domínio específico. A típica coleção para as tarefas de EI é formada de textos e suas associadas chaves de resposta, isto é, os *templates* de saída que devem ser produzidos para cada texto. Isso significa que uma nova coleção deve ser criada para cada nova tarefa de extração. Segundo, porque as coleções simplesmente não contem a informação de controle necessária para treinar os muitos componentes de um sistema de extração, incluindo os módulos de classificação léxica, resolução de coreferências, e geração de *templates*. Os *templates* de saída são freqüentemente inadequados, mesmo para

aprendizado de padrões de extração: eles indicam quais *strings* devem ser extraídos e como eles devem ser classificados, mas não dizem nada sobre qual ocorrência da *string* é responsável pela extração quando múltiplas ocorrências aparecem em um documento. Dessa forma, os pesquisadores criaram seus próprios textos de treinamento, mesmo que esse processo de criação seja muito lento.

Outro problema está relacionado à habilidade necessária para o processamento de linguagem a nível semântico e em domínios específicos, que freqüentemente já obtém resultados semânticos a partir dos níveis iniciais do processo de extração, isto é, classificação léxica e *parser* parcial. Essa característica complica a generalização de exemplos de treinamento para algoritmos de aprendizagem, pois não podem existir textos padrão a partir dos quais exemplos de treinamento completos possam ser independentes de domínio, como é o caso para classificação em categorias gramaticais e *parsing*. As características que descrevem o problema de aprendizado dependem da informação disponível ao sistema de extração no qual o algoritmo de aprendizado é componente, e estas características tornam-se disponíveis somente após os textos de treinamento terem passado através dos estágios iniciais de análise lingüística. Qualquer mudança no comportamento dos módulos iniciais necessitará de novos exemplos de treinamento, que devem ser gerados, e os algoritmos para os estágios posteriores de extração retreinados. Além disso, o algoritmo de aprendizagem deve ajustar-se efetivamente ao ruído causado pelos erros dos componentes iniciais dos processos de extração. O efeito cumulativo das complicações anteriores faz com que os algoritmos de aprendizagem usados para os níveis iniciais não possam ser prontamente aplicados para a aquisição de informações com altos níveis de qualidade e precisão, e novos algoritmos freqüentemente necessitam ser desenvolvidos [HOB 2001].

Em despeito às dificuldades de aplicação dos métodos de aprendizagem para EI, a abordagem baseada em textos exemplo permite simultaneamente auxiliar na solução dos dois maiores problemas em extração: precisão e portabilidade. Quando os dados de treinamento são derivados a partir do mesmo tipo de texto que o sistema de extração irá processar, a habilidade de aquisição da linguagem é automaticamente ajustada para a coleção de textos, incrementando a precisão do sistema. Adicionalmente, visto que cada habilidade de entendimento da linguagem natural é aprendida automaticamente, ao invés de codificada manualmente dentro do sistema, essa habilidade pode ser movida rapidamente de um sistema para outro através do retreinamento dos componentes apropriados [HOB 2001].

3.5.3 Fases de Extração

A semelhança estrutural dos principais sistemas de extração de informações é tal que Hobbs pôde descrever um sistema de extração genérico, que, em linhas gerais, engloba a maioria dos principais sistemas atuais [HOB 2001]. Esse sistema genérico está representado através dos diversos sistemas que competem no MUC, onde muitos dos módulos listados (Figura 3.7) são compartilhados entre eles. Isso demonstra que os pesquisadores de EI não só compartilham módulos, mas também suposições sobre como a tarefa de EI deve ser realizada [WIL 97].

Contudo, existem diversas variações de sistema para sistema: combinação dos módulos, detalhes de implementação, estruturas de controle e dados, etc. [CAR 98]. Nesse sentido, qualquer sistema será caracterizado por seu próprio conjunto e combinação de módulos, mas geralmente eles virão do conjunto a seguir apresentado, e a maioria dos sistemas executará as funções desses módulos em algum lugar [HOB 2001].

Segundo Hobbs, um sistema de extração é uma cascata de fases ou módulos interligados, que a cada passo somam estrutura e freqüentemente perdem informação, esperançosamente irrelevante, aplicando regras que são adquiridas manualmente e/ou automaticamente [HOB 2001]. O resultado de uma fase é os dados de entrada da fase seguinte, acarretando numa interdependência entre as fases.

A seguir é apresentada a metodologia genérica definida por Hobbs, muito semelhante às apresentadas por Costantino, Cardie e Cowie, referências complementares ao texto que se segue [HOB 2001] [COS 97a] [CAR 97] [COW 96].

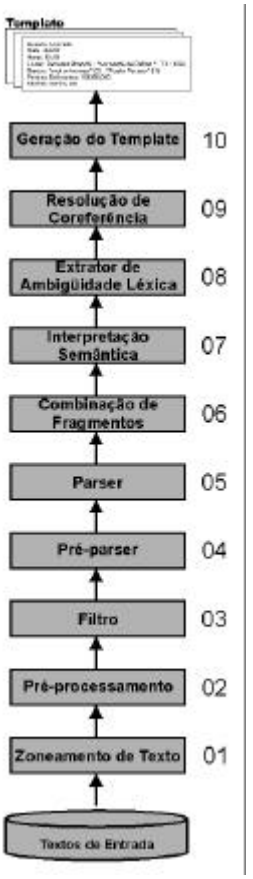

FIGURA 3.7 - Metodologia Genérica de Sistemas de Extração

3.5.3.1 Zoneamento de Texto

Este módulo transforma um documento em um conjunto de segmentos de texto separando, no mínimo, as regiões formatadas do documento das não formatadas. Alguns sistemas podem ir mais adiante, segmentando o texto não formatado por categoria, procurando por partículas de texto dispersas no todo, ou através de meios estatísticos. Esse módulo, nos sistemas usados no MUC-5 [MUC 96], serviu para armazenar a data e informação de autoria ou fonte existente no cabeçalho dos documentos para inclusão no *template.* A data, por exemplo, foi usada para interpretar diretivas temporais do tipo “no último mês” nas fases subseqüentes. Parte da informação do cabeçalho é freqüentemente descartada como irrelevante. Poucos sistemas têm um tratamento sistemático de zoneamento de texto. Isso ocorre somente quando um código fonte apropriado é desenvolvido manualmente.

3.5.3.2 Pré-processamento

Esta fase toma o texto, ou os segmentos de texto, como uma sucessão de caracteres, localizando os limites das orações e produzindo para cada oração uma sucessão de itens léxicos (*tokens*). Os itens léxicos geralmente são as palavras junto com os atributos léxicos correspondentes contidos em um descritor léxico (dicionário). Esse módulo, no mínimo, determina as possíveis categorias gramaticais para cada palavra, e pode selecionar uma única categoria. Com base nesses dados, a fase de pré-processamento pode realizar as seguintes tarefas:

- Disponibilizar os atributos léxicos no dicionário às fases subseqüentes.
- Reconhecer termos com mais de uma palavra.
- Reconhecer e normalizar certos tipos básicos de informação que ocorrem no texto: datas, tempos, nomes de pessoas e companhias, localizações, moedas, e outros.

- Tratar palavras desconhecidas, no mínimo ignorando-as, ou normalmente tentando definir sua morfologia ou o seu contexto aplicado no texto, bem como toda a informação possível de ser obtida.
- Correções ortográficas podem ser realizadas nesse módulo.

Os métodos usados nessa fase são o levantamento das características léxicas, algumas vezes juntamente com análise morfológica, outras com a classificação em categorias gramaticais; pesquisa de padrões para reconhecer e normalizar entidades básicas; técnicas de correção ortográfica; e uma variedade de heurísticas para manipulação de palavras desconhecidas.

O dicionário léxico pode ser desenvolvido manualmente ou obtido a partir de outro sistema, mas cada vez mais são adaptações de dicionários já existentes e são ampliados automaticamente por técnicas estatísticas que operam sobre os *templates* e/ou textos exemplo (ver 3.5.2).

3.5.3.3 Filtro

Este módulo usa técnicas superficiais para filtrar as orações provavelmente irrelevantes e, assim, reduzir o tamanho do texto, o qual será processado mais rapidamente. Em qualquer aplicação particular, os módulos subseqüentes procuram padrões de palavras que indiquem eventos relevantes. Se uma oração não tem nenhum desses padrões, então não há razão para processá-la nas fases posteriores. Esse módulo pode varrer a oração a procura dessas palavras-chave. O conjunto de palavras-chave pode ser criado manualmente ou gerado automaticamente a partir de padrões. Alternativamente, um perfil estatístico pode ser gerado automaticamente das palavras ou N-gramas que caracterizam as orações pertinentes. A oração corrente é avaliada por essa medida e somente será processada se excede algum patamar probabilístico definido.

3.5.3.4 Pré-Parser

Cada vez mais, os sistemas não analisam gramaticalmente uma oração diretamente das palavras que a compõem para uma árvore de análise gramatical completa. Certas estruturas simples são muito comuns e podem ser reconhecidas facilmente, mantendo uma alta confiabilidade. O módulo de pré-*parsing* reconhece essas estruturas, simplificando a tarefa do módulo de *parser* de orações. Alguns sistemas reconhecem nessa fase, grupos de substantivos, ou seja, frases nominais baseadas em um substantivo principal (“cabeça”); como também grupos de verbos, ou verbos juntamente com seus auxiliares. Apostos podem ser anexados aos seus substantivos cabeça com alta confiabilidade. Geralmente são reconhecidas estruturas simples, de pequena escala, ou frases com base em padrões, às vezes definidos com o auxílio de heurísticas específicas, as quais são geradas manualmente. A informação originada nesse nível somente é encapsulada e às vezes descartada. Por exemplo, apositivos de idade podem ser desconsiderados em muitas aplicações.

3.5.3.5 *Parser*

O *parser* é um módulo com entradas e saídas. A entrada é a sucessão de palavras ou itens léxicos (*tokens*) que constituem a oração. A saída é uma árvore de derivação gramatical da oração (*parse tree*). Geralmente, nenhuma informação é perdida entre a entrada e a saída.

Esse módulo usa uma seqüência de itens léxicos e, às vezes, frases, tentando produzir uma árvore gramatical (*parse tree*) para a oração inteira. Sistemas que fazem

análise gramatical completa usualmente representam suas regras como uma estrutura gramatical da oração acrescida de restrições na aplicação das regras (*Augmented Transition Network*, ou *ATNs*), ou como gramáticas de unificação, nas quais as restrições são representadas declarativamente. O algoritmo de análise gramatical mais freqüentemente usado é o *chart parsing*: a oração é analisada de traz para diante (*bottom-up*), com a aplicação de restrições do início para o fim (*top-down*). Estruturas similares, que se estendem sobre as mesmas palavras (*tokens*), são fundidas para trazer este processo de tempo exponencial para tempo polinomial.

Cada vez mais sistemas estão abandonando a análise gramatical completa em aplicações de EI. Alguns desses sistemas reconhecem somente fragmentos, pois, embora estejam usando os métodos padrão para análise completa, sua gramática tem cobertura limitada. Em outros sistemas, o *parser* aplicado é dependente do domínio, utilizando técnicas de busca de padrões em lugar de processos mais complexos, tentando somente localizar, dentro da oração, vários padrões que são de interesse na aplicação (ver 3.5.1).

Gramáticas para o módulo de *parsing* são desenvolvidas manualmente durante um longo período de tempo, ou são obtidas a partir de outros sistemas.

3.5.3.6 Combinação de Fragmentos

Nenhum *parser* existente pode realizar uma análise gramatical completa para 75% ou mais das orações existentes em jornais, ou seja, orações reais e complexas. Então, os sistemas precisam combinar árvores gramaticais obtidas a partir da analise de fragmentos de orações. Esse módulo combina essas árvores. Uma metodologia de combinação simples é unir uma oração inteira com os fragmentos de outras orações. Uma técnica mais apurada é tentar encaixar fragmentos de diferentes orações. Dessa forma, transformam-se fragmentos de árvores gramaticais em uma árvore completa. Cabe salientar que ainda não existe uma teoria concreta sobre a tarefa de combinação de fragmentos, e os métodos utilizados foram desenvolvidos manualmente.

3.5.3.7 Interpretação Semântica

Este módulo transforma fragmentos de árvores gramaticais em uma estrutura semântica, ou forma lógica, ou estrutura de eventos. Todas essas são basicamente representações explícitas do tipo predicado-argumento e relações de modificação que estão implícitas na oração. Freqüentemente, o extrator de ambigüidade léxica é ativado também neste nível. Alguns sistemas têm dois níveis de forma lógica, um geral, do tipo independente de domínio, que pretende codificar toda a informação da oração, e outro tipo para representações mais específicas, dependentes de domínio, que freqüentemente omite qualquer informação não pertinente à aplicação. Um processo de simplificação da forma lógica traduz de uma forma (geral) para a outra (específica).

A interpretação semântica é realizada por uma função ou processo equivalente que combina os predicados com seus argumentos. Normalmente as regras de combinação são especificadas manualmente. Existem diversas variações de como este processo é ativado através dos módulos de quatro a sete. Uma forma é a seguinte: o sistema agrupa palavras em frases, e essas frases em orações analisadas gramaticalmente, e, por seguinte, traduz as orações analisadas para uma forma lógica. O usual, contudo, é passar diretamente das palavras para as orações gramaticalmente analisadas e, então, para as formas lógicas. Recentemente, muitos sistemas não têm realizado análises gramaticais completas das orações. Eles agrupam palavras em frases e traduzem as frases em formas lógicas, e dali em diante é tudo processamento de

discurso. Em uma estrutura de gramática de categorias, passa-se diretamente de palavras para formas lógicas.

3.5.3.8 Extrator de Ambigüidade Léxica

Este módulo transforma uma estrutura semântica com predicados gerais ou ambíguos em uma estrutura semântica com predicados específicos e sem ambigüidades. O extrator de ambigüidade léxica é utilizado freqüentemente em outros níveis, e, às vezes, somente desta maneira. Por exemplo, a ambigüidade gerada pela palavra inglesa "*types*" em "*He types...*" e "*The types...*" pode ser solucionada durante o processo sintático ou durante a classificação em categorias gramaticais. A ambigüidade de "*... rob a bank...*" ou "*... form a joint venture with a bank...*" pode ser solucionada quando um padrão dependente do domínio é encontrado. O fato do padrão ter ocorrido soluciona a ambigüidade. Dessa forma, o extrator de ambigüidade léxica apresenta-se restringindo a interpretação de uma palavra conforme o contexto no qual esta palavra ambígua aparece.

As regras usadas para extrair a ambigüidade são, na maioria dos casos, desenvolvidas manualmente, embora esta seja a área onde os métodos estatísticos contribuíram mais para a lingüística computacional, especialmente na classificação em categorias gramaticais.

3.5.3.9 Resolução de Coreferências ou Processamento de Discurso

Esta fase transforma uma estrutura semântica em árvore, na qual podem existir nodos separados para uma mesma entidade, em uma estrutura de rede, na qual esses nodos são fundidos. Este módulo soluciona coreferências para entidades básicas, como pronomes, frases nominais definidas, e anáforas. Também soluciona coreferências para entidades mais complexas, como eventos. Ou seja, um evento parcialmente descrito em uma parte do texto pode ter sua descrição complementada, caso identificado que esse já foi previamente encontrado no texto; ou é uma conseqüência de outro evento previamente encontrado, como uma morte causada por um ataque. Além disso, a descrição de um evento encontrado no início do texto pode ser associada a um evento relacionado posterior.

Três critérios principais são usados para determinar se duas entidades podem ser fundidas: (**I**) O primeiro, consistência semântica, é normalmente especificado por uma hierarquia de tipos. Assim, "*the japonese automaker*" pode ser fundido com "*Toyota Motor Corp.*". Para pronomes, a consistência semântica ocorre de acordo com o número e o gênero, e talvez sobre qualquer propriedade que poça ser determinada do contexto do pronome; por exemplo, em "*its sales*", "*it*" provavelmente se refere a uma companhia. (**II**) O segundo critério é baseado nas várias medidas de compatibilidade entre entidades do texto; por exemplo, a fusão de dois eventos pode ser condicionada a alcance da sobreposição entre seus conjuntos de argumentos conhecidos, como também à compatibilidade de seus tipos. (**III**) O terceiro critério é de proximidade, conforme determinado por alguma métrica. Por exemplo, pode-se querer fundir dois eventos somente se eles acontecerem em um espaço de N orações de distância entre si. A medida de proximidade pode ser simplesmente o número de palavras entre os eventos no texto. Por exemplo, solucionando casos de pronomes, pode-se provavelmente melhor definir o sujeito da oração atual com base no objeto da oração prévia; esta também é uma simples medida de proximidade.

As regras envolvidas neste módulo são desenvolvidas manualmente, com muito esforço cognitivo. A hierarquia de tipo usada para coreferência, por exemplo,

normalmente é desenvolvida manualmente, contudo alguns pesquisadores começaram a usar WordNet e outros *thesauri* para o desenvolvimento de hierarquias de tipo, ou têm tentado usar meios estatísticos para inferir uma hierarquia de tipo.

O termo "processamento de discurso", nesse contexto, significa resolução de coreferências entre entidades e eventos relevantes. Não houve nenhuma tentativa com bons resultados para reconhecer ou usar a estrutura do texto, além de simples segmentação com base em partículas superficiais do discurso para uso em métricas de proximidade para resolução de coreferências.

3.5.3.10 Geração do Template

Este módulo transforma as estruturas semânticas geradas pelos módulos anteriores em *templates* estruturados conforme requerido para a avaliação do processo de extração. Eventos sem interesse para o usuário são descartados. Não há nenhum método automático para desenvolver as regras de formatação do *template.* Este módulo é o ponto de ligação entre o sistema extrator e os demais sistemas que irão valer-se dos dados extraídos.

3.6 Avaliação

EI utiliza métricas relativamente claras de avaliação comparativa de performance. O resultado de um sistema de extração é uma base de dados estruturada, sendo possível comparar essa saída com uma base de dados ideal a fim de determinar quão bem o sistema executou suas tarefas. Isso distingue EI de muitas outras áreas associadas à lingüística, onde avaliações são muito problemáticas. O requisito crítico para qualquer análise quantitativa é a capacidade de avaliar precisamente os dados de entrada e de saída [LEH 94]. É difícil, por exemplo, especificar e avaliar dados de entrada e de saída para um sistema de tradução ou um sistema de sumarização do tipo texto-para-texto. Nessas tarefas, os conjuntos de dados não podem ser diretamente usados para avaliações quantitativas, pois inexiste correspondência entre os dados de entradas e de saída, ou seja, para uma determinada entrada não existe um único resultado de saída.

Por outro lado, métodos quantitativos prestam-se a intensas análises e são, portanto, metodologicamente complexos. Sem o complemento de uma interpretação inteligente de resultados, tais métodos são limitados quanto ao seu significado [CAR 97]. EI requer um alto nível de compreensão das avaliações, o qual não é capturado diretamente em um gráfico de dispersão. Dessa forma, os sistemas de extração, mesmo prestando-se muito bem a avaliações quantitativas, devem ser analisados segundo outras metodologias às vezes mais subjetivas.

Os métodos de avaliação em EI dependem dos seguintes fatores [DAL 2000]: (**I**) especificação detalhada do que deve ser extraído, visando alta qualidade e consistência entre os sistemas sendo comparados, bem como do processo de avaliação; (**II**) preparação por especialistas humanos de uma base de dados ideal, com os resultados de extração desejados a partir dos documentos teste; (**III**) geração automática de *templates* (base de dados estruturada) pelos sistemas sob avaliação a partir dos documentos teste; e (**IV**) comparação automática ou manual da performance de extração automáticas frente à manual, utilizando um protocolo predeterminado de classificação.

No MUC, os sistemas dos participantes do congresso são avaliados com o auxílio de um programa de classificação, ou programa de escore, que analisa a performance dos sistemas, comparando o *template* gerado por um sistema com o

template chave disponível, gerado manualmente (Figura 3.8) [GRI 95]. O programa alinha os objetos do *template* chave com os objetos no *template* de resposta e, então, calcula os escores baseado em quão bem as respostas combinam com as chaves. A informação extraída deve estar apropriadamente posicionada nos *slots* e no *template* de resposta certo para ser corretamente contabilizada pelo programa.

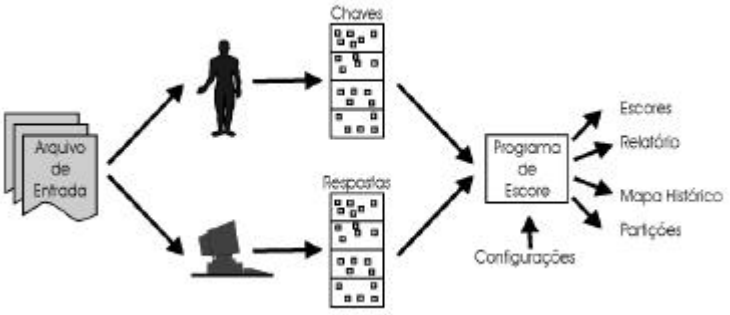

FIGURA 3.8 - Avaliação do Processo de Extração de Informação

Os escores são calculados por dois caminhos. No primeiro, os escores são baseados na contagem de quantos *slots* foram preenchidos corretamente. No segundo, usado para a tarefa de coreferência, os objetos no *template* chave são agrupados em classes de equivalência, como são também os objetos no *template* resposta [INT 2001]. Escores para coreferência são baseados em quão bem as classes de equivalência estão em acordo. São usadas métricas de avaliação de *recall* e de *precision* [LEH 94] (ver 3.6.1).

3.6.1 Métricas

Um sistema de extração pode ser avaliado por sua eficiência em selecionar e recuperar as informações desejadas pelo usuário, em outras palavras, a relevância para o usuário dos dados extraídos. Uma recuperação eficiente depende de dois fatores principais, conforme Salton [SAL 87]: **(I)** os itens relevantes para as necessidades do usuário devem ser recuperados, e **(II)** os itens estranhos, ou irrelevantes, devem ser rejeitados.

Duas medidas são normalmente utilizadas para descrever a habilidade de um sistema na recuperação de itens relevantes e na rejeição de itens irrelevantes de uma base de dados: *recall* e *precision* (Figura 3.9). *Recall* é a proporção de itens relevantes recuperados, sendo calculada pela divisão do número de itens relevantes recuperados pelo número total de itens relevantes na base de dados. Essa medida define quão completo ou compreensivo o sistema é na extração de informações relevantes [HOB 96]. *Precision* é a proporção de itens recuperados que são relevantes, ou seja, capacidade de não recuperar itens irrelevantes. Ela é calculada pela divisão do número de itens relevantes recuperados pelo número total de itens recuperados. Ela define a precisão do sistema, ou seja, sua capacidade de extrair informações corretas [HOB 96].

Em princípio, deseja-se que um sistema produza um alto índice de *recall* e de *precision*. No entanto, tal condição é várias vezes conflitante, pois, quando se recuperam muitos itens relevantes (*recall*), aumenta a probabilidade de também se recuperar dados estranhos às necessidades do usuário [RIL 94a]. Por outro lado, quando a maioria dos itens recuperados é relevante (*precision*), outras informações relevantes podem não ser recuperadas.

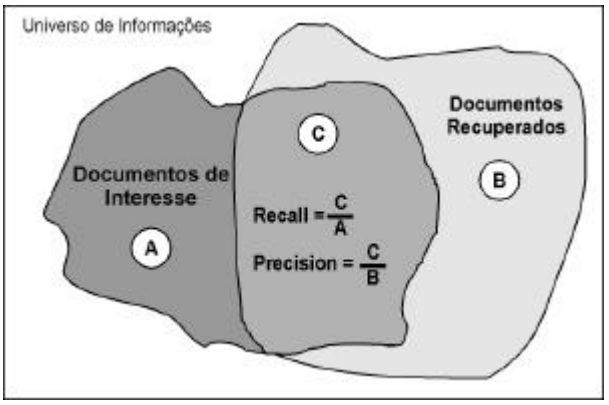

FIGURA 3.9 - Recall e Precision

A função de *recall* em EI é mais bem satisfeita por especificações amplas do domínio de extração, atribuindo maior flexibilidade de seleção ao sistema. Já o fator *precision* é melhor satisfeito pelo uso de domínios específicos, que são capazes de isolar as informações relevantes dentro do universo de aplicação. Contudo, nessa restrição podem ser excluídos vários dados relevantes que deveriam ser recuperados.

Para avaliar um sistema, é importante interpretar os resultados de *precision* frente ao número de informações relevantes em cada conjunto de documentos de teste [RIL 94a]. Se usarmos como exemplo dois conjuntos de textos, cada um contendo 100 documentos, sendo o primeiro com 69 informações relevantes, e o segundo com somente 55, um algoritmo de extração constante, que classifica todas as informações como relevantes, irá apresentar 69% de *precision* no primeiro conjunto e 55% no segundo. Neste sentido, a quantidade de informações relevantes disponíveis é fundamental para calcular-se a medida de *precision*.

A medida F-*score* é um padrão de métrica para EI, combinando as medidas de *recall* e de *precision* em um único valor [HOB 96]. A F-*score* usa um valor *B* para ajuste do peso relativo (importância) de *recall* e de *precision* para o calculo do valor desta medida [RIL 94]. Se $B = 1$, os pesos de *recall* e de *precision* são iguais. Se $B > 1$, *precision* é mais significativo; e se $B < 1$, *recall* tem um peso maior. Dessa forma, o fator peso pode variar conforme o objetivo do sistema a ser avaliado. Caso este tenha sido desenvolvido para atuar sobre domínios específicos com alta precisão, pode-se usar $B = 2$, aumentando a importância da medida de *precision* na avaliação do sistema. A formula do F-*score* usa as variáveis *P* (*precision*), *R* (*recall*), e *B* (importância relativa entre *R* e *P*), sendo a seguinte [HOB 96]:

$$F = \frac{(B^2 + 1)PR}{B^2P + R}$$

3.7 Performance

Uma grande barreira à larga utilização de sistemas de extração é a limitação de performance. A precisão e a robustez dos sistemas deve continuar sendo melhorada. Erros humanos de extração são geralmente causados pela falta de atenção, enquanto erros nos processos de extração automáticos são devidos principalmente ao conhecimento relativamente superficial do domínio de aplicação. As avaliações realizadas no MUC demonstram que é possível avaliar rigorosamente alguns aspectos dos sistemas de extração e que a performance desses sistemas depende (**I**) da

complexidade associada à tarefa de extração, (**II**) da qualidade da especificação do domínio, (**III**) da complexidade sintática e semântica dos documentos e (**IV**) da regularidade da linguagem nesses documentos [CAR 97].

Em geral, os melhores sistemas de extração atingem níveis em torno de 50% de *recall* e 70% de *precision* em tarefas de extração bastante complexas, e podem alcançar níveis mais altos de performance para tarefas mais fáceis: aproximadamente 90% de *recall* e *precision* [GRI 97]. No MUC-6, pode-se notar a relativa similaridade do nível de performance dos sistemas mais bem classificados no *ranking* da conferência. Cinco dos nove sistemas obtiveram F-*score* na faixa de 51 a 56, refletindo níveis de *recall* na ordem de 43% a 50% e de *precision* na ordem de 59% a 70% [GRI 95]. Embora tais níveis de performance não devam ser vistos como impressionantes, deve-se lembrar que extração de informações é difícil tanto para pessoas como para máquinas. O estudo de Wilks, por exemplo, demonstrou que o melhor sistema automático de extração tem uma medida de erro somente duas vezes mais alta que a de análises de um perito especialmente treinado em extração de informações [WIL 97].

Segundo Grishman, a similaridade de performance entre diversos sistemas de extração reflete, em parte, a convergência de tecnologia: os melhores sistemas apresentam muitas semelhanças metodológicas e técnicas [GRI 97]. Adicionalmente, isso provavelmente também reflete características da própria tarefa. Essa semelhança é razoável se comparada a outros fenômenos lingüísticos, como em estruturas sintáticas, onde uma grande parte dos fatos relevantes são lingüisticamente codificados por um pequeno número de formas (itens léxicos, estruturas sintáticas, etc.). Como resultado, é relativamente fácil, se um razoável conjunto de formas for definido, determinar algum nível médio de performance. Isso demonstra que, sendo o processo de extração resultado da combinação de várias tarefas lingüísticas, uma nova e adicional melhoria é custosa [WIL 97].

Quanto à dificuldade de especificação com qualidade de domínios de extração, o maior problema, neste caso, está no constante investimento em melhorias de performance para cada novo cenário de aplicação dos sistemas [GRI 97]. Normalmente tais dificuldades são mais difíceis de serem identificadas e solucionadas por sua especificidade. Contudo, ferramentas baseadas em textos exemplo têm auxiliado muito na aquisição e generalização de padrões de extração melhores em domínios específicos [CAR 97].

4 Extração de Informações em Múltiplos Níveis Baseada em Conhecimento

Este capítulo descreve o método (hipótese da tese) de Extração de Informações em Múltiplos Níveis Baseada em Conhecimento (EI-MNBC), embasado nos conceitos e teorias apresentados nos capítulos anteriores e anexos. A EI-MNBC auxilia o usuário, de forma automatizada, na extração de informações a partir de BDNE/SE, através da convergência de técnicas de EI e SBC. O método não tem por objetivo a construção de um sistema que entenda a linguagem natural, mas sim a manipulação e o tratamento de informações não estruturadas. Ele apresenta ao usuário um conjunto de dados menor que o armazenado, permitindo-lhe avaliar e entender o conteúdo das bases de dados.

Segundo Hedberg, um conjunto de ferramentas computacionais, como agentes de software, sistemas baseados em regras e programas estatísticos, podem ser aplicadas na solução do problema de descoberta do conhecimento [HED 95]. Nesse trabalho, são usadas técnicas de SBC, enfatizando os aspectos cognitivos envolvidos na leitura de um texto, ou seja, como as pessoas recuperam a informação não estruturada.

A EI-MNBC é formada por múltiplas etapas, fornecendo ao usuário as informações relevantes, conforme seu interesse, existentes nas BDNE/SE pela filtragem, seleção e processamento progressivo dos textos. O usuário auxilia no processo de extração, pois a análise e o entendimento dos dados resultantes desse processamento ficam a seu cargo. No entanto, esse auxílio somente é válido caso os resultados da extração forem de boa qualidade. Assim, o conhecimento do usuário, como leitor, é usado para extrair as informações relevantes a partir dos textos: "conhecimento para extrair conhecimento".

4.1 Bases de Dados Intermediárias - BDI

Diversas técnicas de RD, EI e MD emergiram como uma solução para o problema da análise de dados enfrentado por muitas organizações. Estudos anteriores focaram-se no tratamento de informações em um único nível conceitual, o mais primitivo, ou o mais alto [FU 96]. Entretanto, muitas vezes é desejável descobrir conhecimento em múltiplos níveis conceituais, provendo um espectro de compreensão dos dados, do mais genérico, ao mais específico.

A EI-MNBC baseia-se em um conjunto de processos, divididos em três etapas de processamento, que implementados formarão os módulos extratores de um sistema. A grande maioria dos sistemas de extração divide-se em processos encapsulados executados seqüencialmente [HOB 2001]. A EI-MNBC, no entanto, não cria uma ligação direta entre esses processos, usa BDI para interliga-los (Figura 4.1). A vantagem do uso de BDI são os diversos níveis de abstração das informações, supondo-se que a maioria dos usuários não quer ler em detalhes um grande volume de informação, como documentos na íntegra, preferindo uma descrição geral da informação. Diferente dos sistemas de extração tradicionais, o usuário pode escolher entre vários níveis de abstração (múltiplos níveis conceituais – 4.1.2). O uso de BDI em EI aprimora e modifica a tradicional metodologia de sistemas de extração.

FIGURA 4.1 - Bases de Dados Intermediárias (BDI)

Han, Zaiane e Fu usaram bases de dados intermediárias em MD: *Multiple Layered Database* – MLDB [HAN 95] [ZAI 99] [FU 96]. Eles aplicaram MLDB para transformar informações desestruturadas em bases de dados relativamente estruturadas, permitindo o uso de SGBDs para manusear e recuperar os dados nos níveis mais altos. É possível generalizar e transformar a primitiva e diversa informação original em informação razoavelmente estruturada, classificada, descritiva e de alto nível [ZAI 99].

A EI-MNBC também usa BDI, mas em EI. Assim, ao invés de generalizar as informações, especializa e transforma a primitiva e diversa informação original em informação razoavelmente estruturada, classificada, descritiva e de alto nível [SCA 2002]. As informações são progressivamente especializadas e transformadas pelos processos de extração até que toda a informação irrelevante seja descartada e a relevante extraída e estruturada. A informação extraída é armazenada nas diversas bases de dados intermediária, conforme o nível de estruturação e abstração, iniciando pela base de nível 1, e, assim, sucessivamente até o nível final, ou mais alto. O nível mais baixo é o nível zero ou primitivo, correspondendo à informação bruta, armazenada na base de dados original, sem nenhum tratamento. Os níveis altos, nível 1 e demais, armazenam informações extraídas dos níveis mais baixos. A EI-MNBC transforma uma volumosa e desestruturada base de dados primitiva em bases progressivamente menores e melhores estruturadas. SGBDs podem manusear e recuperar os dados nos níveis mais altos, conforme descrito por Zaiane [ZAI 99].

Ao contrário de outros sistemas de extração, a metodologia proposta não é seqüencial. O processo de extração utilizado para gerar o nível N pode valer-se dos dados armazenas no nível diretamente abaixo ao seu (N-1) e todos os anteriores, inclusive no nível primitivo. Além disso, pode-se executar um processo N sem a completa execução do processo N-1, utilizando as informações parciais geradas por esse processo armazenadas em bases inferiores ao processo N. De forma semelhante, processos podem ser executados em paralelo. Em 4.3 serão apresentados exemplos.

O uso de BDI divide um sistema de extração em processos mais simples, concisos e independentes, além de facilitar sua integração. Cada um desses processos extrai determinado tipo de informação desejada pelo usuário e/ou necessária para o processo seguinte. Essa divisão facilita a combinação de diferentes técnicas de RD, EI e MD no mesmo sistema de extração. Tais técnicas serão utilizadas em momentos diferentes, cada uma extraindo parcialmente os dados desejados pelo usuário. Dessa forma, emprega-se no processo de extração a melhor vantagem de cada técnica, obtendo um tratamento de informações mais completo e atendendo melhor as necessidades dos usuários dessa metodologia.

A divisão de um sistema de extração em processos interligados com BDI também aumenta a produtividade e a qualidade desses sistemas, visto a maior modularização e facilidade de reutilização de código/módulos [BOR 96]. Os desenvolvedores reduzem custos e tempo de manutenção, e aumentam os níveis de desempenho, confiabilidade e segurança dos sistemas, etc.

Sendo os processos bastante independentes, eles podem ser executados separadamente. Isso agrega uma grande vantagem ao sistema de extração: possibilidade de execução passo-a-passo. Esse tipo de execução, embora mais demorada, devido à constante interação com o usuário, permite uma análise dos resultados de cada etapa de processamento. A análise é realizada sobre as BDI. Avaliar os resultados de cada processo evita a propagação de erros para as etapas posteriores: maior precisão de extração. Para isto, basta alterar as configurações do processo corrente, melhorando os resultados antes de usá-los nos demais processos. Por outro lado, caso um processo não extraia muitas informações relevantes, o usuário também pode reconfigurar esse módulo, generalizando-o e recuperando as informações descartadas: mais informações relevantes extraídas.

Dessa forma, a metodologia EI-MNBC permite, após o término da execução de um processo, modificar suas características, configurando-o, e executá-lo novamente tornando a extração mais eficiente. A maioria dos sistemas em RI não possuem esta característica de realimentação parcial. Assim, qualquer alteração dos dados de entrada gera uma nova e completa execução do processo de extração. Na EI-MNBC é possível alterar os dados de entrada de um dos processos do sistema, sem a obrigatoriedade de se refazer toda a execução do sistema: qualidade de extração com velocidade e baixo custo de processamento.

4.1.1 Relações Entre as Bases de Dados

Devido à diversidade de informação armazenada na base de dados primitiva, é difícil criar estruturas relacionais de bancos de dados para esse nível. Entretanto, pode-se criar estruturas relacionais para armazenar informações especializadas e razoavelmente estruturadas acima do nível de informação primitivo. Para facilitar, assume-se que as bases de dados de níveis não primitivos (nível 1 e superiores) são construídas conforme o modelo entidade-relacionamento estendido, com capacidade de armazenar e gerenciar tipos de dados complexos, incluindo conjuntos ou listas de dados, dados estruturados, hipertexto, dados multimídia, etc. Assim, as BDI, com exceção do nível 0, apresentam uma estrutura bem definida, conforme o modelo entidade-relacionamento. Zaiane e Fu também adotaram esse modelo em seu trabalho com sucesso [HAN 95a] [FU 95].

Os relacionamentos entre as bases de dados podem ser construídos tanto explicitamente, criando-se relacionamentos, conforme o modelo entidade-relacionamento, como implicitamente, adicionando-se enlaces nas tuplas de cada entidade [HAN 95]. Essa relação é criada desde a formação do nível 1, adicionando *Uniform Resource Locators* (URLs) para os correspondentes arquivos de entrada no nível primitivo. Assim, os demais níveis mantêm relação com a base de dados não estruturada que forma o nível zero. Segundo a EI-MNBC, todas as BDI estão relacionadas entre si, permitindo o mapeamento das informações existentes. O nível 1 é o de mais baixo nível de informação manipulável por um SGBD tradicional.

O relacionamento entre as BDI fornece uma visão geral dos dados nos diversos níveis de abstração e permite sua consulta e pesquisa. Por isso é importante uma estrutura de relacionamento que inclua o nível zero. Além disso, esse relacionamento mostra quais informações de baixo nível foram especializadas ou transformadas para gerar uma informação de mais alto nível. A Figura 4.2 apresenta a base definida para atender o objetivo desta tese, auxiliando no tratamento e manipulação de informações de comércio eletrônico.

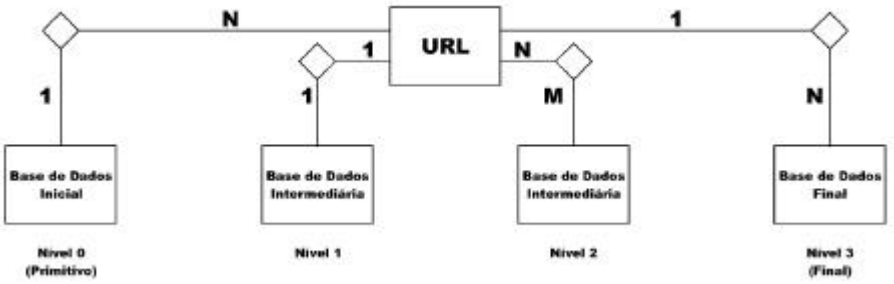

FIGURA 4.2 - Relacionamentos na BDI

4.1.2 Múltiplos Níveis Conceituais

A EI-MNBC trata as informações através de múltiplos níveis conceituais, atendendo as diferentes necessidades de informações dos usuários a partir de uma mesma base de dados bruta (nível 0). Alguns usuários, por exemplo, desejam saber somente o assunto de um arquivo texto; enquanto outros querem selecionar determinadas informações desses arquivos, colocando-as em uma base de dados estruturada e aumentando a complexidade de extração. Os múltiplos níveis conceituais são decorrentes do alto grau de modularização, do uso de BDI e de um tratamento progressivo das informações. A EI-MNBC permite ao usuário obter informações das etapas intermediárias do processamento de dados, analisando os resultados dos diferentes níveis de tratamento e especialização, maiores que o da base inicial, mas não tão específicos, como na final.

A definição de conceito é importante para a metodologia EI-MNBC. Embora vários pesquisadores usem o termo “conceito”, é difícil defini-lo formalmente. Ao procurar no dicionário, encontra-se que “conceito” é uma idéia, opinião, pensamento. Isso confirma a genérica e intuitiva noção de que conceitos são usados para explorar e examinar o conteúdo de conversas, textos, documentos, livros, mensagens, etc. Conceitos, segundo Loh, representam atributos do mundo real (eventos, objetos, sentimentos, ações, etc.) e auxiliam a compreender idéias e ideologias presentes nos textos [LOH 2000]. Segundo Fu, conceitos em bases de dados são normalmente organizados em ordens parciais chamadas hierarquias conceituais, as quais têm um importante papel no processo de tratamento de dados [FU 95] [FU 96]. Elas especificam o domínio dos dados, afetando seu processamento e os resultados obtidos.

Conceitos e dados podem seguidamente ser organizados em diferentes níveis de abstração, baseados na informação das hierarquias conceituais [ZAI 99]. Por exemplo, o local de nascimento de uma pessoa pode ser organizado em uma hierarquia, como cidade, estado, país, etc. Uma hierarquia conceitual define certas relações de generalização ou especialização para os conceitos em si ou um conjunto de atributos [HAN 95b]. Uma hierarquia conceitual pode ser implicitamente armazenada em uma base de dados, como endereços, ou explicitamente definida por especialistas, como “físicos são cientistas”. Também pode ser formada escolhendo conjuntos e subconjuntos dos atributos das relações da base de dados. Por exemplo, os atributos no esquema da relação comida (categoria; marca; especificações do conteúdo; tamanho do pacote; preço) podem formar hierarquias conceituais baseadas na relação das diferentes combinações do conjunto de atributos. Uma hierarquia conceitual também pode ser gerada automaticamente, através da análise de distribuição dos dados na base de dados,

ou do conjunto de dados relevantes, o que normalmente pode ser realizado em dados numéricos, como distribuições de preço [HAN 95a].

Em qualquer tarefa de tratamento de informações não estruturadas, os usuários estão interessados apenas em um subconjunto dos dados e atributos em uma base de dados. Em um sistema baseado em múltiplos níveis conceituais, pode-se descobrir e recuperar informações de um único nível conceitual, ou de diferentes níveis [HAN 95b]. Contudo, é necessário, muitas vezes, utilizar uma linguagem de consulta ou interface gráfica para especificar os dados de interesse, o conjunto de atributos relevantes e as hierarquias conceituais desejadas.

A EI-MNBC define quatro níveis conceituais para o tratamento de informações de comércio eletrônico, objetivo desta tese. Contudo, facilmente podem ser utilizados outros processos que, a partir das BDI, atendam outras necessidades de aplicação. Um processo de MD, por exemplo, pode explorar os dados em N3. Processos anteriores ao N0 (pré-processos) também podem existir para recuperar documentos em servidores distribuídos. Os níveis conceituais definidos para comércio eletrônico são os seguintes:

Nível 0: Informações originais, sem tratamento.

Nível 1: Informação estrutural (formato de armazenamento dos dados e nível de estruturação dos mesmos). Esse nível muitas vezes também define o conteúdo semântico de um texto (ver 4.3.1).

Nível 2: Informação sobre o assunto ou domínio dos dados.

Nível 3: Informação detalhada dos domínios de interesse do usuário.

A introdução das hierarquias conceituais para o tratamento de informações pode ser motivada e justificada pelas seguintes características [FU 96]:

- Uma hierarquia conceitual fornece conhecimento sobre os dados, o qual é necessário e útil no tratamento desses. O uso de conhecimento hierárquico dos dados baseia-se no processo de descoberta científico, no qual pesquisadores aprendem mais sobre algo conduzindo experimentos baseados em crenças ou conhecimentos anteriores. Na EI-MNBC, as informações em N2 são baseadas em N1, pois N1 fornece o conhecimento necessário para a geração de N2. Ou seja, N1 tem conhecimento sobre a estrutura de armazenamento dos dados a serem processados para gerar N2. Na seqüência, N3 utiliza a classificação de informações por assunto, existente em N2, para melhor executar a extração que resultará em N3. Assim, cria-se uma hierarquia conceitual entre os níveis.
- Hierarquias conceituais organizam conceitos de forma hierárquica ou de árvore. Organizações hierárquicas são familiares para os humanos e fáceis de compreender, simplificando o entendimento dos resultados gerados pelo tratamento das informações (Figura 4.3).
- Hierarquias conceituais definem níveis para os conceitos de forma simples e concisa. Isso é necessário e útil para o tratamento de informações através de múltiplos níveis.
- Hierarquias conceituais estão seguidamente disponíveis e podem ser ajustadas e geradas automaticamente. Entre os dados de uma base de dados existem algumas ordens parciais, e as hierarquias conceituais são usadas para capturar tais ordens.

Uma hierarquia conceitual pode ser representada por nodos organizados em uma árvore, valores de um atributo, chamados conceitos. Um nodo especial, “RAIZ”, é

reservado para a raiz da árvore [FU 96]. Uma hierarquia conceitual exemplo para arquivos de dados de comércio eletrônico é mostrada na Figura 4.3.

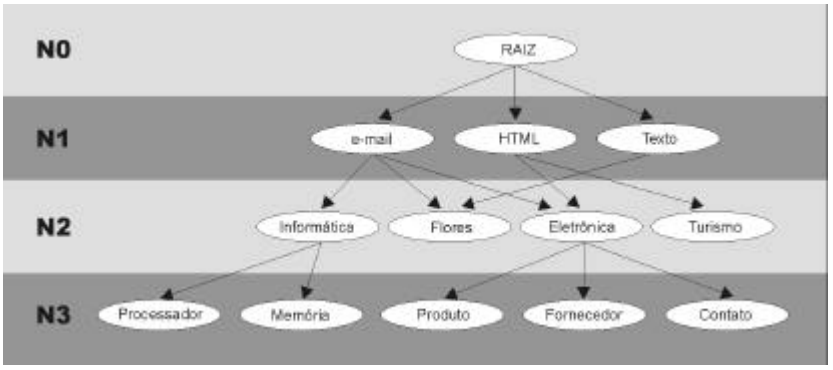

FIGURA 4.3 - Hierarquia Conceitual

4.2 Sistema Baseado em Conhecimento - SBC

Sistemas de extração de informações devem ser personalizáveis, flexíveis e com grande capacidade de evolução, servindo aos interesses individuais dos usuários e às diferentes fontes de informação [SCA 2000]. Essas necessidades levaram EI a apresentar importantes características de Sistemas Especialistas (SE). Um SE representa conhecimento suficiente sobre uma área especifica de um especialista humano, permitindo consultas com habilidade e eficiência [MIC 82]. Diferente dos programas convencionais, SE resolve problemas de julgamento do mesmo tipo encontrado pelas pessoas em seus trabalhos. Alguns SE fornecem respostas em termos de porcentagem de certeza, propagando dentro do sistema graus de confiança associados às diversas informações [WEI 88]. Alguns demonstram como as respostas são encontradas, aumentando a credibilidade do usuário no SE.

SE e SBC diferem profundamente dos sistemas convencionais quanto as suas capacidades, projetos e operações [GEN 86]. Provavelmente, a maior diferença resida na habilidade dos primeiros simularem raciocínio humano, inferirem e julgarem, freqüentemente com informações incompletas; enquanto os últimos efetuam tarefas puramente mecânicas e processam dados, ainda que em alta velocidade. Outra diferença é que SE e SBC derivam conclusões e soluções de heurísticas baseadas em um domínio específico de conhecimento; enquanto os convencionais geram resultados através de algoritmos. Além disso, sistemas convencionais processam exclusivamente com números e caracteres; enquanto SE e SBC trabalham com símbolos e conceitos.

Não é necessário compreender como a mente humana soluciona problemas específicos para produzir um SE que auxilie seu usuário, ampliando sua capacidade e produtividade [GEN 86]. É suficiente extrair o conhecimento de um ou mais especialistas e estruturar esse conhecimento em uma representação computacional uniforme, que permita a aplicação de métodos consistentes de processamento em computador. O desenvolvimento de um típico SE inicia pelo engenheiro de conhecimento, que exaustivamente entrevista uma reconhecida autoridade em um campo particular (domínio específico) e codifica a perícia obtida em regras e fatos [GEN 86]. Depois de representado simbolicamente, o conhecimento extraído é transportado para um computador, que eletronicamente repete análises peritas e estratégias de solução de problemas naquele domínio.

SE e SBC, apesar de áreas semelhantes, diferem em alguns aspectos [SCA 97]. SBC organiza o conhecimento do domínio do problema separadamente de outros tipos de conhecimento usados pelo sistema, como os procedimentos de resolução de problemas ou de interação com o usuário [WEI 88]. Essa coleção de conhecimento especializado é chamada BC, e os procedimentos gerais de solução de problema, de máquina de inferência. Dessa forma, uma mesma máquina de inferência pode ser utilizada com diferentes BC, as quais pertencem a domínios específicos distintos. A BC de um SBC contém fatos (dados) ou regras (fatos condicionais) usados como base para tomada de decisão. A máquina de inferência decide como e em que ordem aplicar as regras a fim de deduzir novos conhecimentos.

A solução de um problema por um SBC pode ser vista como um fluxo de raciocínio, que vai das evidências (fatos e regras) para as conclusões [SCA 97a]. Grande parte da habilidade dos especialistas humanos na solução de problemas reside na capacidade de estreitar o campo das soluções possíveis a partir de cada informação adicional recebida. Da mesma forma, um SBC, dotado de um modelo de inferência bem projetado, fará uso das diversas evidências, combinando-as em hipóteses intermediárias sobre conclusões parciais, a partir das quais os resultados serão, finalmente, deduzidos.

Conforme descrito, observa-se que a EI-MNBC pode ser classificada como um SBC, pois sua metodologia de extração é baseada na estrutura de um SBC (Figura 4.4). Os processos de extração funcionam como máquinas de inferência guiadas pelo conhecimento representado na BC configurada pelo usuário do sistema. Segundo essa visão, temos um conjunto de *eventos de entrada* que, processados por uma *máquina de inferência*, geram uma *conclusão*, a qual é extraída com base no conhecimento armazenado na *BC*. Uma mesma *máquina de inferência* pode utilizar BC diferentes que definem múltiplos domínios de extração, conforme as necessidades do usuário e as características das bases de dados a serem analisadas. Além disso, diferentes BC podem enfatizar características distintas de um mesmo conjunto de textos [SCA 97a].

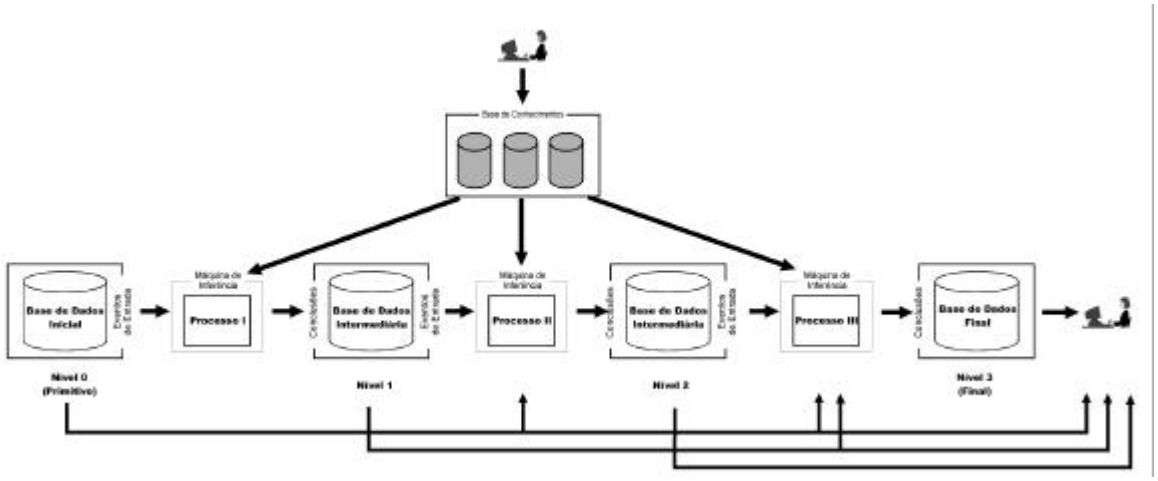

FIGURA 4.4 - SBC na EI-MNBC

Estruturas baseadas em conhecimento normalmente são aplicadas a domínios específicos. A EI-MNBC viabilizou seu uso através do conjunto e organização dos processos definidos nessa metodologia. Os processos de classificação de arquivos por estrutura e domínio, P1 e P2, respectivamente, restringem e definem os domínios tratados por P3, onde as regras de extração são mais complexas. A especialização progressiva da informação permite BC mais específicas e completas em cada estrutura e

domínio de extração. Regras mais simples, associadas aos processos iniciais, são usadas para tratar documentos em domínios amplos. Regras mais complicadas, para os processos finais, tratam informações mais complexas em domínios restritos, reduzindo o volume de definições na BC. O custo de criação e refinamento da BC é distribuído entre os níveis de processamento, na proporção inversa entre o volume de regras e a complexidade dessas [SCA 2000]. Caso contrário, a complexidade de configuração das BC e o custo de extração de alguns processos tornariam a EI-MNBC inviável.

4.2.1 Máquina de Inferência

Seguindo a estrutura de um SBC, a máquina de inferência é representada na EI-MNBC pelos processos de extração de informações: P1, P2 e P3. Esses processos analisam as BDNE/SE conforme o conteúdo da BC definida pelo usuário, gerando um arquivo de saída. Assim, a máquina de inferência interpreta as regras que representam o conhecimento necessário à extração: características estruturais, léxicas, sintáticas e cognitivas (como o usuário extrai a informação) que definem as informações a serem selecionadas e extraídas a partir dos arquivos de entrada.

4.2.2 Eventos de Entrada

Os eventos de entrada na EI-MNBC são representados pelo conjunto de arquivos de entrada que compõe a base de dados a ser analisada pela máquina de inferência, podendo ser a base de dados primitiva, ou as intermediárias N1 ou N2. Cada arquivo contém um conjunto de características e informações que serão analisadas conforme as definições do usuário, contidas na BC, as quais guiam o processo de inferência para tratamento da informação.

4.2.3 Conclusão

A conclusão é caracterizada pelos arquivos de saída, intermediários (N1 ou N2), ou final N3. Esses arquivos contêm, de forma estruturada, as informações extraídas por um processo dos arquivos da base de dados de entrada. O uso de BDI permite classificar determinadas bases de dados (N1 e N2) como eventos de entrada e conclusões, conforme a máquina de inferência sob análise.

4.2.4 Base de Conhecimento

A BC é composta por um conjunto de arquivos de regras com o conhecimento de como extrair informações a partir de BDNE/SE. Elas representam conhecimentos empíricos, gerados a partir do conhecimento do usuário. Esse conhecimento é representado de forma relacional e procedimental, usando regras com a estrutura situação/ação. Essa representação do conhecimento é bastante comum e muito natural para expressar ações [SCA 97a]. Dessa forma, o sistema analisa e processa os textos levando em conta conhecimentos cognitivos existentes, ou seja, o conhecimento envolvido na leitura de um texto por um ser humano.

4.2.5 Representação e Construção do Conhecimento

A incorporação de BC a sistemas de análise automática de textos em domínios particulares recebe muita atenção dos pesquisadores em RI. Uma BC é uma representação abstrata de um domínio de conhecimento ou de um ambiente particular, incluindo seus principais conceitos e relações entre eles. Diversos modelos são usados

para representar conhecimento: redes semânticas, coleções de estruturas lógicas, *script*, etc [COW 96]. Esses modelos apresentam uma estrutura formal de representação e operações de manipulação [WEI 88]. Pesquisadores procuram métodos de representação poderosos, aplicáveis na descrição de uma ampla classe de problemas.

A principal dificuldade nesta estrutura é a própria representação. É raro isolar porções particulares de conhecimento de forma totalmente contida e fechada em si própria. Isto é, a interpretação de parcelas particulares requer não somente o conhecimento de um assunto local, mas também um amplo contexto geralmente não contido em BC particulares [RIC 93]. Segundo esta visão, um texto pode não ser corretamente comparado com uma BC incompleta, sem conhecimento sobre todo o contexto do documento.

Outros problemas da representação do conhecimento são a dificuldade de análise e retirada de ambigüidades de textos e a transformação desses textos em uma estrutura padrão para comparação com a BC. Normalmente, quando os termos de um texto são corretamente analisados, seu vocabulário pode ser muito diferente das especificações da BC. Isso complica a aplicação das regras de inferência e pode então ser necessária a relação entre diferentes estruturas de conhecimento armazenadas. Uma boa estrutura de representação de conhecimento deve possuir as seguintes propriedades [RIC 93]:

- Adequação representacional: capacidade de representar todos os tipos de conhecimento necessários a um domínio.
- Adequação inferencial: capacidade de manipular as estruturas representacionais de modo a derivar novas estruturas que correspondam a novos conhecimentos, inferidos a partir de conhecimentos antigos.
- Eficácia inferencial: capacidade de incorporar à estrutura de conhecimento informações adicionais, usadas para focalizar a atenção dos mecanismos de inferência nas direções mais promissoras.
- Eficácia aquisitiva: capacidade de adquirir novas informações facilmente. O caso mais simples envolve a inserção direta, manual, de novos conhecimentos. Idealmente, o próprio programa seria capaz de controlar a aquisição de conhecimentos.

Infelizmente, ainda não foi encontrado um único sistema que otimize todas as capacidades para todos os tipos de conhecimento [RIC 93]. Como resultado, existem várias técnicas de representação do conhecimento, e muitos programas baseiam-se em mais de uma técnica. A seguir serão apresentadas duas técnicas bastante comuns que servem de embasamento conceitual da EI-MNBC [RIC 93] [WEI 88] [VIC 90].

4.2.5.1 Conhecimento Relacional Simples

O modo mais simples de representar fatos declarativos é através de relações de mesmo tipo, utilizando sistemas de banco de dados. A Tabela 4.1 mostra um exemplo de relação sobre jogadores de Beisebol usando banco de dados relacional. Contudo, essa estrutura de representação fornece capacidades inferenciais muito fracas, pois está limitada às operações do SGBD e aos procedimentos pré-programados. Ela é muito usada por ser independente de domínio de aplicação e o conhecimento representado pode servir de entrada para mecanismos de inferência mais poderosos. Dados os fatos na Tabela 4.1, por exemplo, não é possível nem mesmo responder a pergunta: "Quem é o jogador mais pesado?". Por outro lado, se for fornecido um procedimento para descobrir o jogador mais pesado, então esses dados permitirão a geração da resposta. Se, no entanto, tivermos regras para decidir qual rebatedor deve enfrentar um determinado lançador, com base no fato deles serem canhotos ou destros, então, essa mesma relação

pode proporcionar pelo menos parte das informações exigidas pelas regras. Muitos produtos comerciais já solucionaram as questões práticas envolvidas na relação entre um sistema de banco de dados que fornece suporte a um sistema de representação de conhecimento que implementa recursos de inferência.

TABELA 4.1 - Conhecimento Relacional Simples

Jogador	Altura	Peso	Rebate-Lança
João	1,80	90	Direita-Direita
Carlos	1,75	85	Direita-Direita
Mário	1,85	105	Esquerda-Esquerda
Pedro	1,87	100	Esquerda-Direita

4.2.5.2 Conhecimento Procedimental

O conhecimento procedimental, também chamado de operacional, pode ser representado em programas de várias maneiras. A mais comum é como código de programação através de linguagens, como LISP, descrevendo como fazer algo. O conhecimento é utilizado quando se executa o programa para a realização de uma tarefa. Contudo, essa forma de representar o conhecimento procedimental não é bem qualificada quanto às propriedades de adequação inferencial e eficiência aquisitiva. A primeira porque é difícil escrever um programa que raciocine sobre o comportamento de outro programa; e a segunda porque é complicado atualizar e depurar grandes programas.

Devido a essas dificuldades de raciocínio com linguagens de programação comuns, tentou-se encontrar outras maneiras de representar conhecimentos procedimentais, visando facilitar a manipulação desse conhecimento por pessoas e por outros programas. Assim, a técnica usada em programas de IA é a das regras de produção. Essa representação provê um conjunto modular de regras na forma situação/ação. Um exemplo de representação comum e natural de expressar conhecimento são as regras de produção do tipo "SE [condição] ENTÃO [ação]" [WEI 88]. Na Figura 4.5, uma regra de produção representa conhecimentos operacionais jogadores de beisebol normalmente possuem.

SE
nona rodada, e
vitória está próxima, e
menos de dois fora do jogo, e
primeira base está livre, e
rebatedor for melhor batedor que o próximo rebatedor
ENTÃO
avance o rebatedor

FIGURA 4.5 - Regra de Produção

4.3 Processos de Extração

As pesquisas em EI usam diferentes embasamentos conceituais e metodologias de extração, tais como, metodologias matemáticas e estatísticas, metodologias sintáticas e metodologias de *Machine Learning* [COW 96]. Cada uma dessas linhas de pesquisa tem suas características, vantagens e desvantagens. Em uma estrutura em múltiplos níveis pode-se facilmente combiná-las em um mesmo processo de extração, utilizando o melhor de cada metodologia e obtendo resultados de maior qualidade. A EI-MNBC facilita a integração de diferentes tecnologias de EI, visando um tratamento de informações mais completo, atendendo melhor as necessidades dos usuários.

A extração progressiva de informações de comércio eletrônico esta dividida em três processos: classificação estrutural, classificação por domínio e análise superficial. Cada processo, ou módulo de sistema, pesquisa determinados tipos de informações, utilizando as metodologias mais qualificadas de forma independente. A informação extraída é complementada e refinada ao passar por cada um dos processos. Diferentes características dos textos são exploradas com o objetivo de extrair seu conteúdo central (tema). A combinação das técnicas permite o correto descarte das informações irrelevantes, sem interesse para o usuário, e o destaque dos aspectos centrais dos arquivos de entrada. Isto viabiliza o entendimento das informações de forma ágil e fácil, usando a técnica mais adequada para uma determinada situação momentânea a fim de obter melhores resultados.

4.3.1 Classificação Estrutural – P1

O grande volume e diversidade dos dados, juntamente com a necessidade de processos de extração eficazes e eficientes, sugere o uso de diferentes meios de manipulação dos dados armazenados conforme o arquivo original. EI é um procedimento semântico dependente da identificação do tipo de arquivo de entrada. Neste sentido, o processo de classificação estrutural (P1) explora a extensão dos nomes de arquivos para determinar sua estrutura e forma de extração de outras informações mais apropriadas. Essa técnica é utilizada no sistema de recuperação de informações Essence [HAR 93], onde sumarizadores e procedimentos diferentes são utilizados para cada tipo de arquivo, gerando os índices de pesquisa. O sistema de pesquisa de arquivos ARCHIE também forma seu índice de pesquisa baseado no nome do arquivo [EMT 92]. Observando cada convenção de nome de arquivo, podem ser determinados os tipos de arquivos com um alto grau de acerto [HAR 95]. Como exemplo, pode-se observar a extensão dos nomes de arquivos fontes na linguagem de programação C, ou seja, a extensão ".c"; nomes de arquivos com a extensão ".ps", tipicamente arquivos em *PostScript*; e arquivos com a extensão ".txt", identificando arquivos texto em ASCII.

Esta classificação gera grupos de arquivos baseados nas diferentes estruturas de armazenamento de dados. Arquivos semi-estruturados são agrupados conforme suas diferentes estruturas internas, e arquivos sem uma estrutura padrão, como normalmente são os arquivos com a extensão TXT, formam um outro grupo. A geração dos grupos viabiliza o uso de regras de extração específicas para cada estrutura de arquivo, aumentando a eficiência de processamento e levando em conta características estruturais dos tipos, a fim de destacar as informações relevantes em cada arquivo [GIF 91]. Como exemplo, em arquivos ".c", podemos destacar uma característica comum quanto aos delimitadores de comentários: "/*" e "*/". Além disso, arquivos completos, sem interesse para o usuário, podem ser descartados, reduzindo processamento nas etapas seguintes.

P1 é executado sobre o nível de dados primitivo. Neste processo existem operações para tratamento dos arquivos de entrada, modificando-os para melhor extrair as informações. Estas operações (I) identificam e (II) selecionam os arquivos a serem tratados, (III) dividem arquivos agrupados, tais como diversos arquivos armazenados em um mesmo arquivo físico compactado; (IV) convertem os arquivos para um formato de armazenamento padrão (ASCII) e (V) classificam os arquivos conforme sua estrutura interna de armazenamento de informações. Cada uma das operações define uma fase do processo P1.

4.3.1.1 Identificação

Identificação é a primeira fase de P1. Ela extrai três características dos arquivos em N0: extensão do nome do arquivo, tamanho e data de modificação ou criação. A BC fornece a informação de localização (diretórios) dos arquivos de entrada primitivos. Como a BC é configurada pelo usuário, ele define os locais a serem explorados pelo sistema.

Os dados referentes às características físicas dos arquivos são um importante subsídio para as fases posteriores de extração. Eles apresentam conteúdo semântico sobre o arquivo, como é o caso da extensão do nome, que identifica sua estrutura interna (formato de armazenamento).

O relacionamento entre os arquivos primitivos e as demais BDI é construído de forma explícita, pela adição de ponteiros URL para cada arquivo em N0 (ver 4.1.1). Assim, essa primeira fase cria o arquivo de relacionamento entre as BDI.

4.3.1.2 Seleção

A segunda fase seleciona os arquivos a serem processados nas próximas fases conforme suas características físicas, extraídas na fase anterior. O usuário, configurando a BC, pode selecionar, por exemplo, somente os arquivos mais atuais, poupando tempo de processamento nas próximas fases e processos com dados provavelmente desatualizados. Da mesma forma, arquivos com tamanho excessivo podem ser desprezados nos passos seguintes de extração, onde o processamento é mais custoso.

4.3.1.3 Desdobramento

Após a seleção de arquivos a serem manipulados pelo sistema, dá-se início a fase de desdobramento dos arquivos de informação filtrados. O desdobramento é aplicado somente sobre arquivos agrupados, isto é, arquivos que possuem em sua estrutura física vários arquivos lógicos, desdobrando-os. Assim, a fase de desdobramento modifica os arquivos de entrada. Exemplos de arquivos agrupados são arquivos compactados dentro de um único arquivo físico, ou arquivos de armazenamento de mensagens de correio eletrônico.

A fase de desdobramento consulta à BC, onde estão definidos os tipos de arquivos a serem desdobrados, conforme sua extensão. Isso permite processá-los de maneira diferenciada, conforme sua estrutura interna. As regras da BC, quando executadas, acarretam na ativação de programas externos, usados no desdobramento. No final dessa fase, todos os arquivos processados estão fisicamente separados.

Arquivos agrupados podem conter em sua estrutura outros arquivos agrupados. Assim, o desdobramento de tais arquivos requer o uso de um processo recursivo. Além disso, a base intermediária N1 conterá a URL para o arquivo agrupado (original) e para os arquivos resultantes do desdobramento. Desta forma, a partir dos arquivos desdobrados, oriundos de arquivos agrupados, chega-se aos “pais” desses arquivos. Contudo, o relacionamento entre N1 e as bases superiores é realizado pela URL do arquivo desdobrado, o qual será processado por P2 e P3. Consultas às informações em N0 devem passar por N1, a fim de relacionar os arquivos de dados anteriores e posteriores a fase de desdobramento.

Os arquivos resultantes do desdobramento são reavaliados pelas duas fases anteriores. Assim, um arquivo desdobrado tem suas características externas extraídas e, posteriormente, são filtrados, podendo ser desprezados. Exemplos típicos são arquivos

compactados, que resultam, através do processo de desdobramento, em arquivos rejeitados pelas regras associadas à segunda fase.

4.3.1.4 Conversão

A fase de conversão, conforme a anterior, modifica os arquivos de entrada, convertendo-os para o formato ASCII, requisito imprescindível para os processos posteriores. Além disso, realiza a "limpeza" dos arquivos, retirando caracteres de controle, como o de retorno de carro (*carriage return*). As regras na BC definem quais arquivos não estão no formato ASCII e devem ser convertidos, ou quais devem ser limpos, conforme sua extensão. Exemplos são arquivos *PostScript* e HTML, muito comumente encontrados na Internet.

As regras também definem quais programas externos usar para que esta fase atinja seu objetivo. Sendo assim, na BC devem existir regras para cada tipo de arquivo a ser convertido, conforme sua extensão. Isso permite um processamento diferenciado de arquivos com estruturas internas distintas. Ao final desta fase, todos os arquivos estão no formato texto.

Os arquivos convertidos podem ter seu nome e/ou extensão alterados pelo conversor, devendo ser mantido um relacionamento entre o novo arquivo gerado e o que o originou. A base N1 contém uma URL para o arquivo original e outra, relacionada, para o arquivo convertido, usando a mesma estrutura da fase anterior.

4.3.1.5 Classificação

Nesta fase, cada um dos arquivos é classificado explorando convenções de extensões de nomes já identificadas nas fases anteriores. A extensão contém atributos semânticos, pois identifica o tipo de estrutura interna de um arquivo. A partir dessa identificação, pode-se utilizar regras de extração específicas para cada estrutura, aumentando a eficiência de processamento nas demais fases e melhor selecionando as informações relevantes conforme as características estruturais dos arquivos [GIF 91].

A classificação resulta em um índice de arquivos conforme suas estruturas internas. Arquivos com diferentes extensões podem pertencer a uma mesma categoria, desde que o usuário defina essa categoria é representada por mais de uma extensão na BC. Assim, determinadas regras de extração serão usadas somente para arquivos com uma estrutura específica nos processos posteriores.

4.3.1.6 Base de Dados Intermediária N1

A Tabela 4.2 apresenta a base de dados N1.

TABELA 4.2 - Base de Dados Nível 1 – N1

Arquivo de Dados Sobre os Arquivos Classificados por Estrutura	
Campo	**Tipo de Dado**
URL Original	Cadeia de Caracteres
URL Pós-Processo	Cadeia de Caracteres
Classe	Cadeia de Caracteres
Tamanho	Valor Numérico
Data	Valor Numérico
Extensão	Cadeia de Caracteres

4.3.2 Classificação por Domínio – P2

Classificação é uma técnica empregada para identificar a que categoria determinado documento pertence conforme seu conteúdo [LEW 91] [NOR 96] [YAN

99]. Para tanto, as categorias devem ter sido previamente modeladas ou descritas através de suas características, atributos ou fórmula matemática [RIJ 97].

O processo P2 utiliza classificação para filtrar arquivos de dados e extrair informações de domínio (neste caso considerado como um assunto ou tema) envolvendo os arquivos em N1. Aqueles que não se encaixarem em alguma das categorias (domínios) predefinidas são ignorados pelo processo P3 (análise superficial) ou colocados em uma categoria separada, para que sejam analisados futuramente ou quando houver necessidade. Os demais documentos são armazenados em N2. Cada categoria de domínio relevante para o usuário terá associado ao processo P3 um conjunto de regras de análise superficial específico, determinado por ele, conforme sua necessidade de informação. Ignorando os domínios irrelevantes, minimiza-se o custo de processamento e problemas de sobrecarga de informações em P3.

Os sistemas de classificação de documentos geralmente utilizam uma das seguintes técnicas [WIV 2000]: (**I**) Regras de Inferência: um conjunto de características determinado nas regras deve ser encontrado nos documentos para que esses sejam identificados como pertencentes a uma determinada categoria. Necessita-se de muito tempo para elaborar as regras (esse processo é geralmente manual) e elas devem ser adaptadas caso o domínio mude. Geralmente são desenvolvidas para uma tarefa e domínio específico. O conhecimento modelado é facilmente compreendido (por estar na forma de regras) e seus resultados são, na maioria dos casos, melhores do que os apresentados pelos outros métodos (maiores informações em [APT 94] [LEW 94]). (**II**) Modelos Conexionistas (redes neurais artificiais): esses sistemas induzem automaticamente um modelo matemático ou um conjunto de regras a partir de um conjunto de documentos de treinamento. Podem ser colocados em prática rapidamente e são capazes de se adaptar as mudanças do ambiente de dados. Eles não necessitam de um especialista ou pessoa na análise do domínio. Por outro lado, necessitam do conjunto de treinamento e seu modelo ou regras não é tão compreensível (maiores informações em [WIE 95]). (**III**) Método de Similaridade de Vetores ou de Centróides: as categorias e os documentos são representados por vetores (conjuntos) de palavras (denominados centróides). Cada documento é comparado com o vetor descritivo de cada categoria. A categoria mais similar ao documento é definida como sua categoria [LOH 2000]. (**IV**) Árvores de Decisão: é uma abordagem parecida com a primeira, porém utiliza técnicas de aprendizado de máquina para induzir as regras. Para cada categoria uma árvore de decisão é criada [APT 94] [LEW 94] [YAN 99]. (**V**) Classificadores de Bayes: parecido com o conexionista, porém fundamenta-se em teoria probabilística, definindo a probabilidade de determinado documento pertencer a uma determinada classe [KOW 97].

A EI-MNBC usa a técnica de similaridade de vetores, pois é a mais adequada à estrutura de processos definida na metodologia. Muitas pesquisas e sistemas de recuperação e extração de informações representam o conteúdo de textos somente usando termos simples, consistindo normalmente na avaliação de palavras individuais [LOH 99a]. Nesse tipo de avaliação, é usado um enfoque estatístico; ou seja, a exploração das propriedades estatísticas do texto, como a observação da freqüência de ocorrência das palavras no mesmo. A partir dessa freqüência, atribuí-se um peso a cada palavra, comparando-as. As palavras com maior peso são as palavras-chave na descrição do conteúdo global do texto no momento de uma consulta. Os resultados de recuperação dependem crucialmente da escolha de pesos efetivos para os termos.

As categorias usadas na classificação dos arquivos são predefinidas pelo usuário através da BC da EI-MNBC. Essas categorias são caracterizadas por palavras-chave

(atributos), as quais devem ser encontradas nos arquivos sob classificação, conforme o algoritmo de similaridade, determinando a categoria de um arquivo.

Os documentos em N1, previamente processados por P1, estão no formato ASCII e classificados por sua estrutura de armazenamento. Isso permite o uso da técnica de *stopwords* adaptada para cada estrutura, conforme descrito abaixo, aumentando a qualidade do processo de classificação. Identificadores HTML, por exemplo, podem ser descartados durante o processo P2. Não existe a necessidade de um processo de classificação extremamente criterioso, onde existiriam diversas categorias, muitas vezes semelhantes entre si, exigindo uma alta qualidade de classificação. Isso geraria um aumento significativo do custo de processamento, indo contra os objetivos da metodologia. P3 realizará, posteriormente, um processo de extração mais criterioso e custoso, somente sobre as categorias relevantes ao usuário. Dessa forma, ocorre uma distribuição equilibrada entre o volume de arquivos a serem processados e o custo de processamento.

O processo de classificação de domínio pode ser dividido em diversas fases ou sub-processos, conforme descrito a seguir.

4.3.2.1 Stopwords

Normalmente, de 40% a 50% de um texto é formado por palavras, chamadas de *stopwords*, que contribuem pouco no significado geral do texto, sendo menos utilizadas para recuperação de informações [LAN 68]. Essas palavras são ditas de conexão ou reiteração como, por exemplo, pronomes, artigos, conjunções, verbos auxiliares, adjetivos quantitativos, etc. Elas podem ser eliminadas, pois não servem para a caracterização do conteúdo dos arquivos, já que aparecem tão comumente em todos eles [BEC 97]. Outras palavras são inerentes à linguagem utilizada ou ao contexto dos documentos processados [WIV 99]. Portanto, é comum excluir as *stopwords*, pois elas não influenciam no processo de classificação e o tornam mais demorado.

Na EI-MNBC, o usuário indica, através da BC, quais palavras devem ser ignoradas pelo processo. Documentos com estruturas internas diferentes podem exigir listas de *stopwords* específicas. Assim, a classificação estrutural do arquivo é usada para selecionar a lista mais adequada, tornando P2 adaptável às diferentes estruturas. As *stopwords* definidas nas listas são excluídas do conjunto de atributos de cada arquivo, reduzindo o número de características utilizadas na fase de cálculo de similaridade.

4.3.2.2 Discriminação de Atributos

Não é fácil identificar atributos (características) de BDNE/SE, pois não há sinal ou local predeterminado que os indique. É necessário um método que identifique as características marcantes de cada arquivo. Tratando-se de informação não estruturada, as palavras podem servir para caracterizar os arquivos, sendo utilizadas como atributos [NG 97].

Em um texto, nem todas as palavras aparecem com a mesma freqüência, podendo-se selecionar as palavras mais usadas [KOR 97]. Caso contrário, o uso de todas as palavras do texto tornaria o cálculo de similaridade muito demorado. Em aplicações reais de extração os usuários necessitam de um tempo de resposta curto. Portanto, o maior “gargalo” das técnicas de classificação por domínio está na etapa de seleção de atributos, que objetiva diminuir o número de características a serem processadas. Para melhorar a eficiência de seleção, é comum utilizar um método de pesquisa estatístico [RIJ 2000]. As técnicas mais simples de identificação de atributos relevantes são as de cálculo de freqüência: freqüência absoluta, freqüência relativa e

freqüência inversa de documentos [SAL 83]. Essas técnicas são utilizadas pela EI-MNBC, exceto a última, pois, sendo as categorias previamente definidas pelo usuário, a quantidade de documentos em que um termo aparece não é considerada. O resultado das técnicas de classificação não é muito influenciado pela função de discriminação escolhida, mas sim, pelo algoritmo de cálculo de similaridade [WIL 88].

a) A freqüência absoluta (F_{abs}) consiste em contar o número de vezes que determinada palavra aparece em um documento. Essa é uma medida de peso muito simples, não distinguindo termos que aparecem em poucos documentos de termos que aparecem em muitos [BEC 97]. Esse tipo de análise pode ser, em alguns casos, extremamente importante, pois os termos que aparecem em muitos documentos não são capazes de discriminar um documento de outro [RIJ 2000]. Além disso, a freqüência absoluta não leva em conta a quantidade de palavras existentes no documento. Uma palavra pouco freqüente em um documento pequeno tem a mesma importância de uma palavra muito freqüente em um documento grande. Contudo, a freqüência absoluta serve de base para o cálculo da freqüência relativa.

b) A freqüência relativa (F_{rel}) busca solucionar esse último problema da F_{abs}, levando em conta o tamanho do documento (quantidade de palavras que esse possui) e normalizando os pesos conforme essa informação [BEC 97]. Sem essa normalização, os documentos grandes e pequenos seriam representados por valores em escalas diferentes. Os documentos maiores possuiriam melhores chances de serem recuperados, já que receberiam valores maiores no cálculo de similaridades [SAL 87]. É calculada a freqüência (porcentagem) que um termo ocorre no texto em relação aos outros termos [RIJ 2000]. Essa avaliação apresenta melhores resultados semânticos, sendo complementada pela anterior. A freqüência relativa (F_{rel}) de uma palavra x em um documento é calculada dividindo-se sua freqüência absoluta (F_{abs}) pelo total de palavras nesse documento (N): $F_{rel}(x) = F_{abs}(x) / N$.

4.3.2.3 Seleção de Atributos

Discriminados os atributos dos arquivos, aplica-se uma técnica de seleção de atributos relevantes. Essas técnicas, na maioria das vezes, são independentes da função de discriminação (cálculo de freqüência) escolhida. Uma boa seleção consegue reduzir ao máximo o conjunto de atributos sem sacrificar a identificação do conteúdo dos documentos, o que implica no desempenho da fase de classificação.

A EI-MNBC usa a técnica de seleção de termos por truncagem, pois é mais facilmente implementada e não influi negativamente nos resultados, conforme testes realizados por Schütze [SCH 97], além de oferecer um ganho de performance no algoritmo. Existem outras técnicas de seleção, como a Indexação Semântica Latente (*Latent Semantic Indexing – LSI*) [LEW 91]. A LSI, por exemplo, mostra-se mais efetiva para o agrupamento de termos onde as categorias não estão previamente definidas (não sendo o caso da EI-MNBC), pois usa a freqüência inversa de documentos. Outras técnicas levam em conta a estrutura dos documentos ou propõem a análise sintática desses a fim de identificar semanticamente (ou morfologicamente) os atributos mais importantes [WIV 2000]. Porém, o custo de processamento é demasiadamente alto.

Na truncagem, estabelece-se um número máximo de atributos por documento. Para tanto, ordena-se o vetor (lista) de atributos de cada documento pelo grau de relevância (freqüência relativa). Assim, somente os primeiros n atributos são utilizados. A truncagem define que palavras com pouca freqüência não caracterizam fortemente o documento, podendo ser ignoradas [WIV 2000]. Estabelecer o grau mínimo ou o

número mínimo de atributos relevantes é um processo complicado e difícil. Porém, os experimentos de Schütze indicam que, na grande maioria dos casos, o uso de 50 atributos oferece resultados satisfatórios [SCH 97]. Na EI-MNBC, o usuário define o número de atributos relevantes, configurando a BC.

4.3.3 Classificação

Identificados os atributos relevantes de cada arquivo, parte-se para a análise de similaridade entre esses arquivos e as categorias definidas pelo usuário. Esta fase determina a categoria de cada documento, usando a técnica de similaridade de vetores, conforme justificado anteriormente. Sua eficiência depende dos atributos selecionados. Caso as fases anteriores não tenham identificado atributos realmente relevantes, o processo P2 será comprometido, pois a classificação pode não representar a coleção de documentos processada.

Quanto maior o número de atributos utilizado nesta fase, mais confiável será o grau de similaridade entre os arquivos e as categorias [WIV 2000]. Contudo, a maior dificuldade na classificação dos textos é a grande quantidade de termos a serem processados: a alta dimensionalidade do espaço de atributos [SAL 83]. O espaço de atributos constitui-se dos atributos extraídos dos textos nas etapas anteriores. Podem existir centenas ou milhares de atributos diferentes representando os documentos de uma coleção, mesmo se ela for pequena.

No modelo vetor-espacial, cada documento é representado por um vetor de termos, e cada termo possui um valor associado indicando seu grau de importância (*peso*) no documento. Portanto, cada documento possui um vetor associado constituído por pares de elementos na forma {(palavra_1, peso_1), (palavra_2, peso_2)...(palavra_n, peso_n)} [SAL 97]. Nesse vetor são representadas todas as palavras da coleção e não somente aquelas presentes em um documento. Os termos não contidos em um documento recebem *peso* zero e os demais tem seu peso calculados através de uma fórmula de identificação de importância. Pesos próximos de um (1) indicam termos extremamente importantes e próximos de zero (0) caracterizam termos irrelevantes. Cada elemento do vetor é considerado uma coordenada dimensional. Assim, os documentos podem ser colocados em um espaço euclidiano de *n* dimensões (onde *n* é o número de termos) e a posição de um documento em cada dimensão é dada por seu peso. Na EI-MNBC, o termo com maior freqüência apresenta peso 1 e o com menor freqüência diferente de 0, após a etapa de seleção, 0,01. Os demais termos, intermediários, tem seu peso calculado entre 1 e 0,01, conforme a distribuição proporcional do número de ocorrências do termo no documento (freqüência relativa).

As categorias também são representadas por vetores, cujos termos e pesos são atribuídos pelo usuário. Dessa forma, vetores de documentos são comparados com vetores de categorias, identificando o grau de similaridade entre eles. A categoria mais similar (mais próxima no espaço) a um determinado documento classifica-o.

O modelo vetor-espacial é baseado em um processo de combinação do tipo *matching*. Esse modelo encontra os documentos mais similares a uma categoria, sem a necessidade de uma combinação exata entre os termos que definem a categoria e os que definem o documento (*exact-matching*) [SPA 97]. Como resultado, para cada categoria os documentos são ordenados de forma decrescente, conforme sua similaridade com essa. Essa ordenação identifica os documentos relevantes de uma categoria.

O conjunto completo de valores em um vetor descreve a posição de um documento ou categoria no espaço; ou seja, as distâncias entre um documento e outro ou uma categoria indicam seu grau de similaridade. Documentos e/ou categorias com os mesmos termos são colocados em uma mesma região do espaço. A similaridade entre um documento e uma categoria é calculada pela comparação de seus vetores; isto é, sua distância no espaço, usando uma medida de similaridade como o coeficiente do coseno, utilizada por Salton em seu modelo original [LOH 99a]. Nessa formula, C representa o vetor de termos da categoria; D o vetor de termos do documento; *Ptc* são os pesos dos termos da categoria e *Ptd* os pesos dos termos do documento; e *t* é o número de termos. Um documento, por definição da EI-MNBC, pertence somente a uma categoria, a de maior similaridade. Isso simplifica a escolha das regras de extração associadas a P3 e selecionadas conforme a classificação estrutural (P1) e de domínio (P2) do documento.

$$\text{Similaridade}(C, D) = \frac{\sum_{t=1}^{n} Ptc.Ptd}{\sqrt{\sum_{t=1}^{n} (Ptc)^2 . \sum_{t=1}^{n} (Ptd)^2}}$$

4.3.3.1 Base de Dados N2

A Tabela 4.3 apresenta a base de dados N2.

TABELA 4.3 - Base de Dados Nível 2 – N2

Arquivo de Dados Sobre os Arquivos Analisados Superficialmente	
Campo	**Tipo de Dado**
URL Pós-Processo	Cadeia de Caracteres
Categoria	Cadeia de Caracteres
Similaridade	Numérico

4.3.4 Análise Superficial – P3

Sistemas de extração de informações baseados em análise sintática são freqüentemente utilizados para extrair unidades de identificação completas, como nomes próprios ou preposições, a partir de documentos. Isto destaca as relações entre as palavras, ultrapassando os limites do conhecimento contido puramente em termos simples. Obtém-se um maior conjunto de informações sobre cada palavra de um texto, pois a classificação sintática amplia e até altera o significado individual de uma palavra [LEV 88]. Por exemplo, na frase "João joga bola", a palavra "João" ganha o atributo de nome próprio através da análise sintática, não detectado em simples análises de termos. A análise sintática, ou *parsing*, converte uma frase “plana” em uma estrutura hierárquica, que corresponde às unidades de significado sintático da frase [CAR 97].

Programas de análise sintática são normalmente grandes e demandam o armazenamento de uma grande quantidade de termos e estruturas, bem como a necessidade de equipamentos com maior capacidade de processamento, impedindo a análise de grandes quantidades de textos [HOB 2001]. Na prática, métodos sintáticos são freqüentemente aplicados em um modo “seguro de falhas”, analisando textos particulares, onde a informação completa sobre cada palavra pode não ser avaliada, e certas regras gramaticais podem ser violadas [SAL 88]. Normalmente, quando a quantidade de documentos é grande e analisadores sintáticos são usados, o resultado é pobre. Nesses casos, métodos estatísticos são preferidos. Alternativamente, um analisador mais simples pode ser usado, concentrando-se em certas passagens do documento em preferência a outras [HOB 2001].

Textos contêm componentes sintáticos, bem como semânticos. Um mesmo conjunto de palavras pode ser utilizado para descrever diferentes situações, dependendo do conteúdo do documento, ou seu contexto. Além disso, a estrutura das frases e o vocabulário, baseados no conhecimento individual do autor, exigem a inclusão desse conhecimento no sistema que pretende analisar o documento. A sintaxe por si só não resolve muitas ambigüidades que complicam a análise de textos, fazendo com que métodos puramente sintáticos não sejam suficientemente poderosos para produzir uma análise apropriada dos documentos [COS 97]. Características contextuais, ou dependentes do discurso, e conhecimento são necessários para esta proposta. Assim, novos sistemas têm incluído informações adicionais sobre palavras a partir de dicionários, ampliando seu vocabulário e informações semânticas de áreas particulares [COW 96].

Várias opções têm sido avaliadas a fim de simplificar o processamento sintático (*parsers* parciais) com vistas à análise semântica: recuperação de fragmentos de textos relevantes usando técnicas de busca de padrões, freqüentemente através de processos de estado-finito, para identificar os fragmentos com base em características sintáticas locais [CAR 98]. A partir desse conjunto reduzido de informações sintáticas, pode-se aplicar generalizações estatísticas através de conjuntos maiores de documentos [WIL 97]. Características superficiais, mais simples de serem extraídas e processadas, têm sido usadas para prover um surpreendentemente e efetivo mecanismo para o desenvolvimento de sistemas com *parsers* parciais [COW 96].

A metodologia EI-MNBC prevê o uso único de técnicas de análise superficial, como conhecimento associado a termos, identificação de padrões sintáticos e análise baseada em contexto para extrair informações e gerar as bases de dados estruturadas a partir das BDNE/SE. Assim, a EI-MNBC trata grandes volumes de dados não estruturados, pois a análise superficial dos documentos apresenta menor custo de processamento. O conhecimento armazenado na BC, e os processos de análise léxica e sintática parcial (superficial) provêem o alcance adequado de extração de informações para a EI-MNBC. Elas reconhecem as construções de texto relevantes para o usuário, extraindo-as.

O uso de técnicas superficiais pela EI-MNBC somente é possível devido aos processos de classificação anteriores (P1 e P2), os quais definem previamente a estrutura e o domínio dos documentos sobre os quais o processo P3 será executado. A importância dessa classificação é provada através de outros trabalhos de EI que utilizam parcialmente a análise superficial. Esses sistemas, como o de Naomi Sager, tratam informações em um domínio bem definido, no caso exemplificado, no domínio médico, apresentando resultados altamente efetivos [COW 96].

As regras na BC referentes ao processo P3 permitem sua configuração conforme as características léxicas e sintáticas exploradas pelo usuário a fim de extrair as informações desejadas. Utilizando as BDIs anteriores, associa-se a cada classificação de estrutura e/ou domínio a BC para a classe de documentos a serem tratada em P3. Dessa forma, o domínio de extração torna-se mais restrito, facilitando a configuração das regras associadas a P3 e aumentando a qualidade dos dados extraídos.

A EI-MNBC prevê situações de análise semântica levando em conta o contexto dos documentos. Isso impede que o resultado semântico seja meramente baseado nas características léxicas e sintáticas (superficiais). As informações de contexto encontram-se na BC especificada pelo usuário para o domínio, na identificação estatística do domínio, na estrutura da base de dados resultante e através da análise de termos

relacionados. Adicionalmente, o significado semântico das palavras muitas vezes é determinado pelas informações semânticas associadas com a palavra sob análise, contido na BC e em dicionários. Esse conhecimento semântico estende-se para as palavras parentes na forma de uma estrutura relacional, a qual indica a função de cada termo parente dentro da estrutura semântica como um todo.

Ambigüidades léxicas e anáforas são resolvidas usando heurísticas de preferência, definidas pelas regras na BC. Uma vez que os termos são identificados, o conhecimento disponível na BC soluciona improváveis entendimentos do termo sob análise, bem como auxilia a retirar ambigüidades de termos relacionados.

O contexto do documento sob análise, juntamente com seu domínio, definido anteriormente (P2), influencia na escolha do significado semântico a ser associado a cada termo analisado: são selecionados os significados mais associados ao presente contexto. Outro fator importante é o montante de conhecimento disponível na BC sobre um dado domínio.

4.3.4.1 Análise Léxica

A primeira fase do processo de análise superficial é a análise léxica. Os documentos são percorridos e analisados palavra-a-palavra (*tokens*) com o objetivo de obter conhecimento. A analise fundamenta-se nas regras da BC definida pelo usuário, verificando se o *token* apresenta informações ou identifica segmentos de texto relevantes. A ocorrência de um *token* pode auxiliar na extração das informações do documento, trazendo consigo um significado semântico importante no contexto do documento. A palavra "*deadline*", por exemplo, pode ser um *token* importante na análise de documentos do domínio “Chamadas de Artigos para Congressos”. A partir da informação extraída, o usuário conhecerá detalhes importantes do documento analisado. Além disso, a existência de um *token* em um documento pode desencadear um processo de análise sintática, conforme descrito no item a seguir.

Um *token* pode conter informações adicionais que alteram ou complementam seu significado semântico, identificando-as como relevantes. Várias delas são extraídas através da análise de padrões, comparando a palavra com padrões pré-definidos na BC [SCA 97a]. Para textos em inglês ou em português, por exemplo, letras maiúsculas ou minúsculas ajudam na identificação de nomes próprios. Alguns delimitadores, como títulos (exemplo: Sr. e Dr.) e designações para companhias (exemplo: S.A. e Ltda.), podem reconhecer palavras como nomes próprios. A cadeia de caracteres “10/10/95” é outro exemplo onde a formatação dos números identifica o *token* como uma data, atribuindo-lhe conhecimento semântico. A performance de processos de identificação de padrões normalmente varia de 40% a 90%, dependendo do domínio e das técnicas usadas [COW 96].

Um *token* pode ser composto por uma ou mais palavras. Muitas vezes o significado semântico correto de uma palavra somente é extraído quando essa é combinada com palavras posteriores ou anteriores. Um exemplo é o termo em inglês “*Call for Paper*”. Individualmente, essas palavras podem não apresentar um significado relevante ao usuário, ou até podem gerar uma análise errônea. Contudo, combinadas, apresentam um conteúdo semântico único. Dessa forma, as regras de extração permitem a pesquisa de termos; ou seja, a pesquisa de um conjunto de palavras em uma determinada ordem.

A localização de um *token* dentro de um documento pode complementar seu significado semântico. O uso de regras de extração conforme a classificação estrutural

do documento, definida em P1, auxilia na análise léxica. Estruturas de correios eletrônicos, por exemplo, contem a data de envio da mensagem na terceira linha do arquivo, o que determina o significado de um *token* com formato de data nessa localização.

4.3.4.2 Análise Sintática e Relações de Termos

Pessoas claramente adquirem informações de um documento tanto a partir do conteúdo das palavras individuais, quanto da estrutura na qual essas palavras estão inseridas [MET 89]. A sintaxe reflete essa estrutura e esta fase do processo P3 complementa as informações extraídas dos termos individuais (análise léxica) com informações sintáticas. Isto aumenta a precisão do processo de extração proposto.

No entanto, nesta metodologia, a informação sintática é obtida a partir da análise da relação entre os termos de um documento. Não são identificados elementos sintáticos como sujeitos ou objetos. A análise sintática proposta encontra as propriedades estruturais associadas a um termo identificando onde ele está inserido no documento, conforme outros termos próximos a ele que complementam seu significado. Um termo é colocado como o "cabeça de pesquisa" a partir do qual os demais termos relacionados são extraídos. O termo cabeça é resultante da prévia análise léxica, destacando-o como relevante dentro do contexto do documento, conforme as definições do usuário. O objetivo é ampliar o significado atribuído ao termo cabeça na análise léxica através da extração de informações sintáticas superficiais associadas a ele. Os demais termos envolvidos na análise sintática são chamados termos dependentes; ou seja, são os termos relacionados com o termo cabeça. O escopo das relações entre termos é definido pelo usuário na BC. Essa forma de pesquisa sintática tem motivação e base nos trabalhos de [MET 89] e [SAL 68], cujos objetivos são semelhantes ao deste processo de extração: obter o conhecimento armazenado em múltiplos termos.

Palavras iguais podem ocorrer em documentos diferentes, combinadas de formas distintas. Por exemplo, dois textos, ou duas sentenças, contêm os mesmos termos, mas não se referem ao mesmo conceito. O significado de cada ocorrência desses termos depende muito da relação dentre os mesmos em cada ocasião, ou, até mesmo, da relação com outros termos próximos. No exemplo do item anterior, definiu-se a cadeia de caracteres "10/10/95" como uma data. No entanto, essa data deve ter um significado no contexto global do documento sob análise. Com o uso das características sintáticas superficiais, podemos associar essa data com a palavra que a antecede, por exemplo: "*deadline*". Isso atribui, a partir do contexto, um significado semântico maior à cadeia de caracteres "10/10/95". Assim, conforme o domínio de extração, definido no processo P2, por exemplo, uma chamada para o envio de artigos a um congresso, pode-se saber, com um certo grau de certeza, a data final de envio dos artigos.

A descoberta do conhecimento contido em um documento tende a funcionar melhor com termos não tão raros, nem tão freqüentes nos documentos sob análise [MET 89]. Com termos muito comuns, a análise sintática superficial identifica as relações entre termos, especificando o significado semântico dos mesmos. A combinação dos termos ocorre menos freqüentemente nos documentos do que os termos individuais, um exemplo é a palavra "Data". Essa palavra pode aparecer muitas vezes em um texto, mas seu significado passa a ser melhor definido no contexto do documento quando associada a outras palavras dependentes, como: "Data de Inscrição", "Data de Entrega", ou "Data de Avaliação".

Os procedimentos de análise léxica e sintática são interdependentes. O analisador sintático é requisitado pelo léxico sempre que esse identifica um termo

relevante ao usuário conforme definido na BC. No exemplo anterior, o analisador léxico destacou o *token* "10/10/95", com base em um padrão de formato, como possível identificador de uma informação relevante e realizou sua interpretação semântica com sendo uma data. Com base nessa interpretação, requisitou uma análise sintática, ampliando o entendimento semântico do termo inicialmente identificado e de outros termos relacionados a esse, concluindo que era uma data de "*deadline*".

4.3.4.3 Dicionários de Termos

A natureza das unidades de texto, ou *tokens*, utilizadas para análises baseadas em padrões, requer que elas muitas vezes estejam contidas em um dicionário léxico [CAR 97]. Dicionários, ou *thesauruses*, são ferramentas usadas para detectar relações entre palavras na linguagem natural [SAL 68]. Tais relações podem ser utilizadas para criar um sistema de análise da linguagem, retirando a ambigüidade de significado de termos e, além disso, gerando grupos de similaridade entre palavras pela identificação de relações em um contexto de várias entradas do dicionário [SAL 68].

A EI-MNBC usa dicionários para expandir o conhecimento semântico sobre uma palavra, encontrando outras palavras relacionadas a essa e utilizando esse conjunto maior para a extração de informações. A adição de termos relacionados ou mais amplos é uma estratégia de expansão semântica para o melhoramento do *recall* [SCA 2000]. Aumentando o número de termos, mais documentos podem ter seu domínio identificado em um processo de classificação, havendo, portanto, uma maior chance de recuperar informações relevantes [SPA 97]. Além disso, dicionários armazenam conhecimento semântico sobre seus termos, os quais podem auxiliar na extração das informações, por exemplo, identificando uma palavra como um nome próprio. No MUC, os participantes valem-se de grandes volumes de informação disponíveis em dicionários, como o Gazzetteer, que contem 250.000 nomes de localidades [CAR 97].

Durante a análise léxica, segundo a metodologia EI-MNBC, as palavras relevantes em um domínio podem estar armazenadas diretamente em regras na BC ou em dicionários de termos. Tais dicionários são formados por um conjunto de arquivos com as palavras a serem identificadas pelo analisador léxico. Esses arquivos são definidos através das regras de extração.

4.3.4.4 Granularidade de Pesquisa

A granularidade de pesquisa é um tópico importante para o processo de análise superficial. Ela determina a unidade na qual a informação contida no texto será pesquisada. A EI-MNBC percorre o texto com granularidade de palavra; ou seja, o texto é "lido" pelo sistema palavra-a-palavra, e essas se tornam os *tokens* de entrada para o processo de extração. Essa granularidade permite uma boa análise do documento pelo processo de extração P3. As características léxicas são avaliadas palavra-a-palavra e, relacionando-as, são extraídos os componentes sintáticos dos documentos. Além disso, essa granularidade proporciona uma boa flexibilidade de configuração do processo de extração pelo usuário. Configura-se o sistema utilizando as palavras como unidades de atômicas de informação a serem extraídas. Pode-se ter, por exemplo, a palavra *from* na classificação estrutural de correspondências eletrônicas. O usuário pode configurar o sistema para extrair todas as palavras subseqüentes ao *from*, utilizando uma palavra como referência para a extração do endereço de remetente.

Granularidades menores, como caracteres, exigiriam um tempo maior de processamento e uma maior complexidade das regras de extração, pois teriam que permitir a combinação de caracteres na busca de *tokens* maiores: palavras, por exemplo.

Além disso, o documento seria pesquisado caractere-a-caractere. Contudo, na EI-MNBC, o uso de uma granularidade de pesquisa de palavras não impede que essas palavras tenham o tamanho de um caractere, como o artigo "a", ou, então, que o sistema extraia do documento um simples caractere. Por exemplo: pode ocorrer que, ao encontrar-se a palavra "*from*" em um documento, o caractere seguinte deva ser extraído, o caractere ":".

No entanto, a fim de ampliar a flexibilidade de extração e a precisão, sem um aumento prejudicial do tempo de extração e da complexidade de configuração das regras da BC, a EI-MNBC também se permite uma granularidade de pesquisa de caractere. Para tal, dividiu-se a parte condicional das regras relacionadas à análise superficial. Uma primeira parte usa granularidade de palavra e a outra de caractere. A primeira parte da condição chama-se de condição de pesquisa e a segunda de condição de verificação. Sempre que a condição de pesquisa é verdadeira a condição de verificação é testada. Pode-se ter uma condição de pesquisa com a palavra "*from*" combinada com uma condição de verificação com o caractere ":", por exemplo. Sempre que a palavra "*from*" for encontrada, será verificado se o caractere subseqüente é o ":", e, somente se a condição de verificação for verdadeira, a ação definida na regra será executada.

4.3.4.5 Grupos de Informações

As informações extraídas pelo processo de análise superficial formarão a BDI de nível 3 (N3). Conforme as BDIs geradas anteriormente, as informações extraídas por P3 estarão relacionadas com o documento original (nível primitivo) e as demais BDIs geradas por P1 e P2. O relacionamento ocorre através de uma URL para o documento original.

A informações armazenadas em N3 são classificadas em grupos, permitindo sua organização e facilitando o entendimento da mesma pelo usuário. Esses grupos são definidos previamente pelo usuário através das regras de extração. A partir de N3 pode-se gerar um banco de dados relacional. Cada grupo da BDI definirá um atributo (coluna) da relação (tabela), e as informações formarão as tuplas (linhas).

4.3.4.6 Base de Dados N3

A Tabela 4.4 apresenta a base de dados N3.

TABELA 4.4 - Base de Dados Nível 3 – N3

Arquivo de Dados Sobre os Arquivos Analisados Superficialmente	
Campo	**Tipo de Dado**
URL Pós-Processo	Cadeia de Caracteres
Grupo	Cadeia de Caracteres
Informação	Cadeia de Caracteres

5 Protótipo: Sistema de Extração de Informações em Múltiplos Níveis Baseado em Conhecimento – SE-MNBC

A fim de comprovar a metodologia EI-MNBC, hipótese da tese, e atingir os objetivos desejados definiu-se a arquitetura do protótipo do sistema **SE-MNBC - Sistema de Extração de Informações em Múltiplos Níveis Baseado em Conhecimento.** Esse protótipo é uma versão atualizada do Sistema de Extração Semântica de Informações (SES) [SCA 97a]. A linguagem de extração utilizada no protótipo foi definida em outros trabalhos do grupo de pesquisas em EI do Instituto de Informática da UFRGS [SCA 97] [SCA 97a] [ZAM 2001].

5.1 Arquitetura

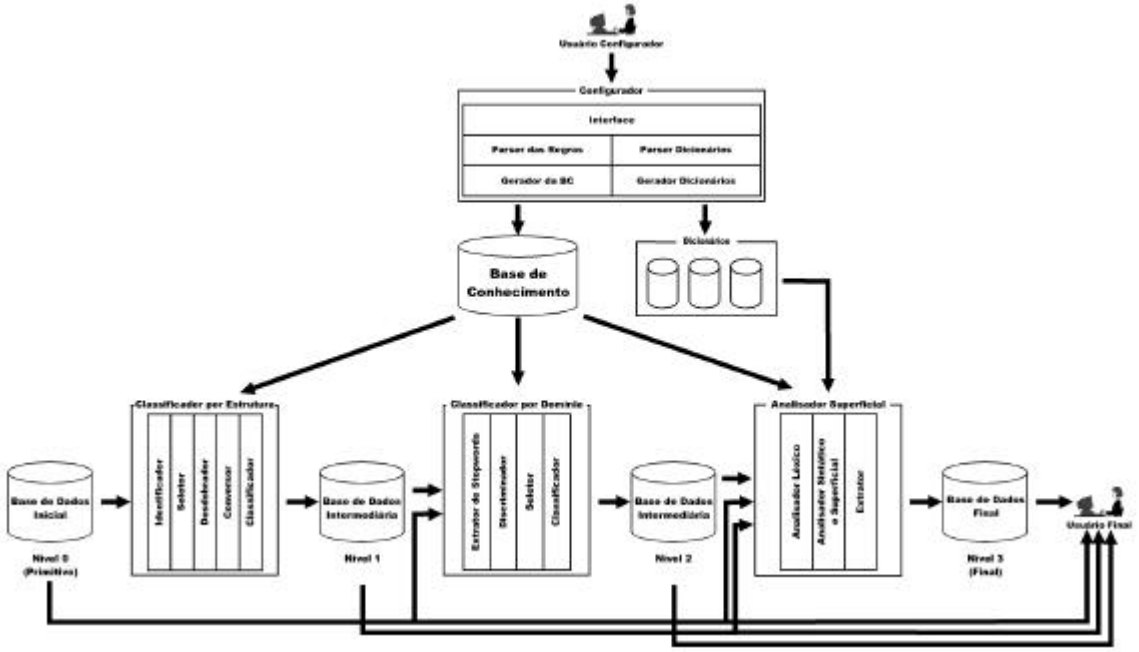

FIGURA 5.1 - Arquitetura do SE-MNBC

Extração de informações a partir de BDNE/SE é uma tarefa difícil e complexa. Um sistema que execute essa tarefa precisa ser flexível para atender às necessidades do usuário e o dinamismo do ambiente de aplicação. O SE-MNBC não é um sistema monolítico, mas sim um sistema orientado à expansão. Ele apresenta um conjunto de módulos para a extração de informações de comércio eletrônico ao qual é possível, sem grande esforço de programação, agregar outros módulos ou então adicionar novas funcionalidades as já existentes (Figura 5.1). O bom nível de modularização, com tarefas de extração independentes, facilita o entendimento da arquitetura e a implementação do sistema. Essas características são princípios da metodologia EI-MNBC, tendo sido constatados pelos analistas/programadores do protótipo durante o período de desenvolvimento e posterior avaliação do código fonte produzido [ZAM 2001]:

- Facilidade para projetar, implementar e testar individualmente cada um dos módulos do sistema, diminuindo o tempo e a complexidade de integração dos módulos do sistema.
- Facilidade de manutenção e documentação.

- Facilidade de evolução e integração de novas características ao processo de extração inicialmente projetado.
- Facilidade de reutilização de código a partir de funções já desenvolvidas pelos analistas/programadores, inclusive funções utilizadas no sistema SES [SCA 97a].
- Redução de custos de desenvolvimento, com a ampliação da produtividade e maior qualidade e confiabilidade final do sistema resultante.

5.2 Módulos do Sistema

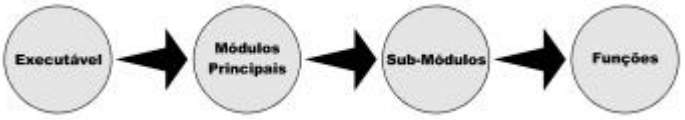

FIGURA 5.2 - Hierarquia Organizacional do SE-MNBC

O SE-MNBC está dividido em um conjunto de quatro módulos principais que se dividem em sub-módulos menores, visando uma melhor organização e clareza do código fonte gerado. Cada um desses sub-módulos contém uma ou mais funções que juntas executam uma tarefa independente encapsulada nele. Existe uma hierarquia organizacional usada no desenvolvimento do sistema. A Figura 5.2 mostra cada um dos níveis hierárquicos, desde o mais amplo (executável) até o mais específico (funções).

5.2.1 Classificador por Estrutura

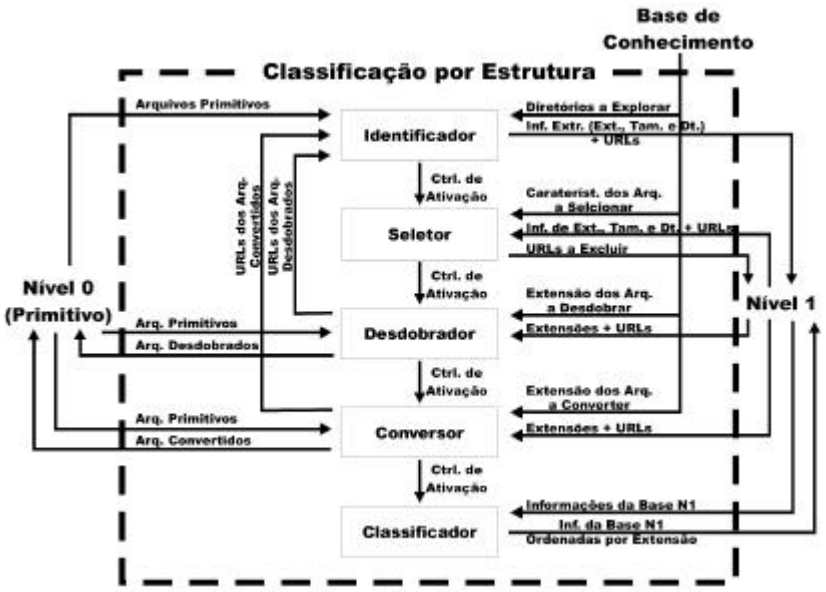

FIGURA 5.3 - Arquitetura do Classificador por Estrutura

O módulo *classificador por estrutura* analisa como as informações estão armazenadas dentro dos arquivos de dados, extraindo conhecimento semântico a partir dessa informação, conforme definido pela EI-MNBC (Figura 5.3). Este módulo atua diretamente sobre a base de dados inicial (*N0*), tratando os arquivos primitivos. Todos os arquivos da *N0* são analisados pelo *classificador*, podendo alguns serem desprezados e não fazerem parte da base *N1*, resultante deste processo. O objetivo do *classificador* é atingido através da combinação de cinco sub-módulos, conforme descrito a seguir.

a) Identificador: verifica junto à *BC* quais diretórios contêm os arquivos primitivos (*N0*), extraindo deles as informações de extensão, tamanho e data de modificação ou criação. Como a *BC* é configurada pelo *usuário*, ele define os locais explorados pelo SE-MNBC. Os dados extraídos pelo *identificador* são armazenados, de forma estruturada, na BDI *N1*. Conforme definido na EI-MNBC, os relacionamentos entre os arquivos primitivos e a BDI *N1* são construídos de forma implícita, pela adição de ponteiros URL entre o arquivo original das informações e as tuplas de *N1*.

b) Seletor: seleciona as informações armazenadas em *N1* e, conseqüentemente, os arquivos a serem processados nas próximas etapas da *classificação por estrutura*, levando em conta as características estruturais extraídas. A seleção é realizada conforme as necessidades do *usuário* armazenadas na *BC*. Este sub-módulo atua somente sobre a BDI *N1*, alterando-a, sem acessar a *N0*.

c) Desdobrador: divide os arquivos agrupados existentes na *N0*, isto é, arquivos que possuem em sua estrutura física vários arquivos lógicos. O desdobramento é realizado fazendo-se consultas à *BC*, onde existem regras para cada tipo de arquivo a ser desdobrado, conforme sua extensão. Tais regras, quando executadas pelo *desdobrador*, ativam programas externos usados no desdobramento, tais como descompaquitadores de arquivos. Cada arquivo agrupado é copiado para um diretório auxiliar, onde o programa externo é executado, desdobrando o arquivo original em um ou mais arquivos, que, após o término do processamento, são retornados ao diretório onde inicialmente estava o arquivo agrupado. Os arquivos resultantes dessa etapa são analisados pelo *identificador* e pelo *seletor* e, caso não selecionados, suas URLs não são incluídas em *N1*. Os arquivos agrupados não são processados pelas demais etapas do SE-MNBC, mas sim os arquivos resultantes do desdobramento e selecionados.

A BDI *N1* contêm URLs para o arquivo agrupado e para os arquivos resultantes do desdobramento. A partir dos arquivos desdobrados, chega-se aos seus "pais". O relacionamento entre *N1* e as bases superiores é realizado pela URL do arquivo desdobrado. Consultas às informações em *N0* passam por *N1* a fim de relacionar os arquivos de informações anteriores e posteriores ao processo de desdobramento.

Arquivos agrupados podem conter em sua estrutura outros arquivos agrupados. Assim, o desdobramento de tais arquivos requer a utilização de um processo recursivo. O trabalho do *desdobrador* somente termina quando não existem mais arquivos agrupados a serem tratados.

d) Conversor: converte os arquivos primitivos em *N0* para o formato ASCII, requisito imprescindível para algumas etapas posteriores, e "limpa-os", retirando caracteres de controle, como o de retorno de carro (*carriage return*). Esse processo ocorre conforme as regras na *BC*, onde se determinam os arquivos a serem convertidos através de sua extensão. Programas externos são usados nesta tarefa, tais como conversores de arquivos gravados no padrão Microsoft Word, extensão DOC, para texto. A BDI *N1* contêm URLs para o arquivo original e para os arquivos resultantes da conversão. A partir do arquivo convertido, chega-se ao seu "pai". Os arquivos originais não são processados pelas demais etapas do SE-MNBC, somente os convertidos.

e) Classificador: reordena as informações em *N1* pelo atributo de extensão, gerando grupos de arquivos baseados nos diferentes tipos de estruturas de armazenamento de informações. Essa classificação permite o uso de regras de extração mais adequadas pelos demais processos de extração, conforme a estrutura interna dos arquivos.

Os arquivos apontados pela *N1*, através de URLs, não foram fisicamente duplicados durante o processo de *classificação por estrutura*, exceto os desdobrados e os convertidos, duplicados para que as alterações necessárias fossem feitas sem a perda do arquivo original. Assim, a base de dados primitiva tem seu tamanho ampliado o mínimo necessário.

5.2.2 Classificador por Domínio

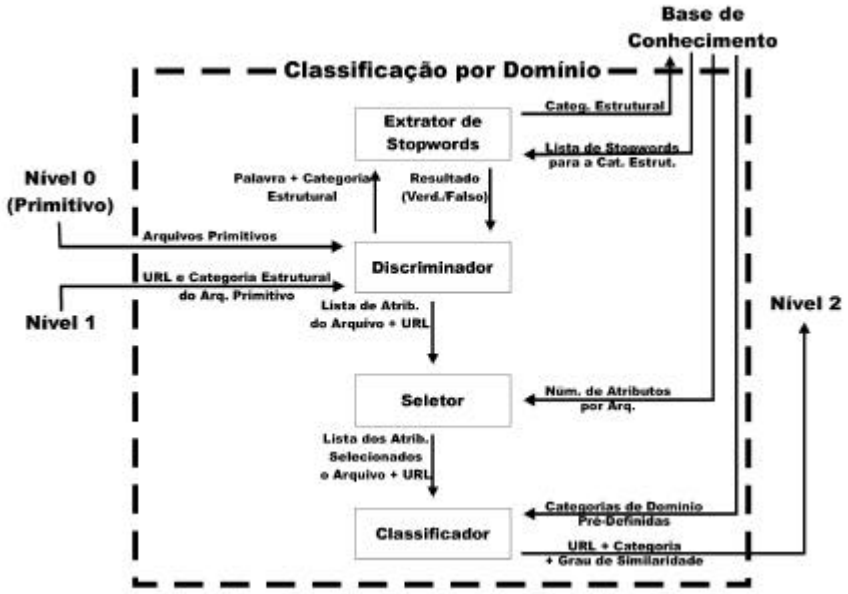

FIGURA 5.4 - Arquitetura do Classificador por Domínio

O *classificador por domínio* extrai informações sobre o domínio dos documentos; ou seja, o assunto ou tema, que os envolve (Figura 5.4). Todos os arquivos cujas URLs existem na BDI *N1* são analisados e classificados. Arquivos que não se enquadram em uma das categorias de domínio definidas pelo *usuário*, não sendo de interesse dele, são descartados da base *N2*, resultante deste processo. A seguir são apresentados os sub-módulos do *classificador por domínio*.

a) Extrator de *Stopwors*: retira dos arquivos apontados por *N1* palavras que contribuem pouco no significado geral do documento, conforme as regras existentes na *BC* para uma determinada categoria estrutural de arquivo. O *extrator de stopwords* não altera o arquivo original. Esse módulo é ativado pelo *discriminador*, durante a discriminação dos atributos, filtrando somente aquelas palavras que devem fazer parte do cálculo de freqüência. Assim, o arquivo original é "lido" apenas uma vez.

b) Discriminador: percorre o conteúdo de todos os arquivos apontados por *N1*, palavra-a-palavra, para extrair os atributos que os identificam. Na metodologia EI-MNBC, as palavras mais significativas de um documento são seus atributos. Cada atributo recebe um peso, conforme sua freqüência relativa no documento. Para tal, durante a "leitura" do arquivo, cria-se uma lista com suas palavras e o

número de usos dela nesse documento (freqüência absoluta). Antes de incluir as palavras na lista, o *extrator de stopwors* é ativado, e somente as palavras que contribuem na identificação do domínio do documento são contabilizadas. Além disso, um contador armazena o total de palavras do arquivo sob análise. Com tais informações, após a "leitura" do arquivo, calcula-se a freqüência relativa de cada palavra; ou seja, a freqüência percentual de ocorrência de um termo em relação aos outros termos de um documento. Atribui-se também, às palavras da lista, seu peso relativo ao tamanho do documento.

c) Seletor: ordena de forma crescente a lista de atributos de um arquivo conforme as freqüências relativas dos atributos e aplica a técnica de truncagem, definida pela EI-MNBC, para a seleção dos atributos mais importantes. O número máximo de atributos utilizados para caracterizar cada documento é definido na *BC*. Assim, a lista é truncada mantendo-se somente os primeiros *n* atributos de maior freqüência relativa em um documento.

d) Classificador: determina a categoria de cada documento, aplicando a técnica de similaridade de vetores, conforme definido na EI-MNBC. A partir da lista de atributos de um documento e das listas de atributos que definem as categorias de domínios, previamente declaradas pelo usuário na *BC*, calcula-se a similaridade entre um documento e as categorias. Esse cálculo usa a comparação de vetores (listas) de atributos e determina a distância entre eles no espaço, usando a medida de similaridade do coeficiente do co-seno. Associa-se a cada documento a categoria mais similar a ele, armazenando essa informação na BDI *N2*, juntamente com a URL do arquivo e o grau de similaridade entre eles.

5.2.3 Analisador Superficial

FIGURA 5.5 - Arquitetura do Analisador Superficial

As regras armazenadas nas *BC* referentes ao *analisador superficial* configuram-no conforme o conjunto de características léxicas e sintáticas exploradas pelo usuário para extrair as informações relevantes (Figura 5.5). Utilizando a classificação de cada arquivo por domínio em *N2* e por estrutura em *N1*, associa-se a cada combinação dessas classificações as regras de extração superficial. Dessa forma, o domínio de extração é mais restrito, facilitando a configuração da *BC* e aumentando a qualidade dos dados extraídos. Arquivos de classes não interessantes ao usuário são desprezados por este processo. Abaixo são descritos os sub-módulos do *analisador superficial*.

a) Analisador Léxico: analisa os arquivos apontados pela BDI *N2* em busca de conhecimento a partir de termos, ou seja, das palavras que os compõem. O arquivo é percorrido palavra-a-palavra, verificando se alguma delas identifica conteúdo relevante no documento conforme as regras da *BC*. As palavras pesquisadas também podem estar armazenadas em *dicionários* usados pelo *analisador léxico.* As informações relevantes são extraídas e armazenadas na BDI *N3*, divididas em grupos definidos pelo usuário na *BC*. Os grupos relacionam informações do mesmo tipo. Também é armazenada a URL do arquivo de origem da informação. Algumas palavras identificadas como relevantes ativam o *analisador sintático e superficial.*

b) Analisador Sintático e Superficial: complementa a informação léxica extraindo informações sintáticas com o uso de técnicas de análise superficial: conhecimento associado a termos, identificação de padrões sintáticos e análise baseada em contexto. O *analisador sintático e superficial* é acionado quando o *analisador léxico* envia um termo identificado como relevante, chamado de "cabeça de pesquisa". São extraídos os termos relacionados ("dependentes") ao cabeça, ampliando o conhecimento obtido sobre ele na análise léxica. O escopo das relações entre termos é definido nas regras da *BC*. As informações extraídas são armazenadas na BDI *N3*, organizadas em grupos de informações, como ocorre com o *analisador léxico*.

5.2.4 Configurador

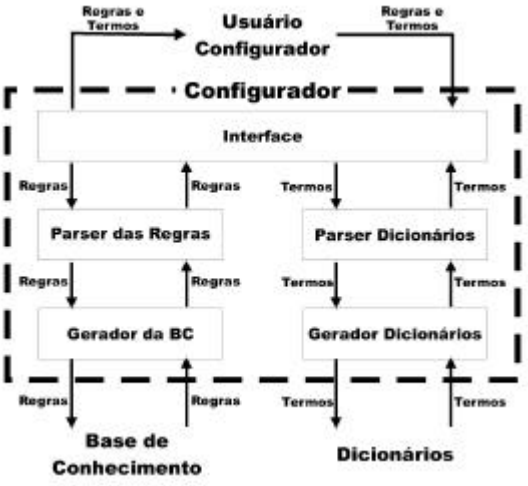

FIGURA 5.6 - Arquitetura do Configurador

O SE-MNBC é um SBC. Os conhecimentos armazenados na *BC* são independentes da máquina de inferência (processos de extração) e diferentes *BC* podem ser usadas com a mesma máquina. No entanto, essa independência não tem valor sem o módulo *configurador* (Figura 5.6). Apesar de não estar diretamente relacionado com os demais módulos, o *configurador* atribui ao SE-MNBC uma qualidade de "sistema aberto", permitindo a geração e a alteração de diversas *BC* e *dicionários*, conforme as necessidades do usuário e o ambiente de extração. O *configurador* apresenta características importantes a fim de cumprir seus objetivos, estando esta dividido em cinco sub-módulos.

5.2.4.1 Interface

O SE-MNBC será utilizado principalmente por usuários não especialistas em Informática. O projetista que constrói uma aplicação para esse público alvo deve ter atenção com o desenho e a funcionalidade da interface utilizada pelos usuários do sistema [FER 94]. A interface do SE-MNBC possibilita o bom aproveitamento do potencial do sistema, facilitando o uso e a compreensão desse e a exploração das facilidades de extração. Assim, a interface responde às necessidades do usuário, satisfazendo-o. Essa qualidade foi atingida incorporando-se ao *configurador* as principais características desejáveis em uma interface [FRA 93]:

- **Diversidade:** A interface deve ser utilizada, de maneira conveniente, por todos os tipos de usuários, conforme suas características.
- **Complacência:** A interface deve ser tolerante com o usuário, permitindo que ele se refaça de situações de erro, e deve levar em consideração o esquecimento por ele das informações já apresentadas.
- **Eficiência:** A execução de uma tarefa pela interface deve despender pouco esforço do usuário.
- **Conveniência:** O acesso e manuseio das operações da interface devem ser fáceis.
- **Flexibilidade:** A interface deve oferecer várias maneiras de efetuar-se uma mesma operação.
- **Consistência:** A interface deve apresentar-se e comportar-se segundo regras fixas, bem definidas e conhecidas do usuário.
- **Prestimosidade:** A interface deve ajudar o usuário, bem como alertá-lo de situações de erro que eventualmente ocorram.
- **Imitação:** O uso de exemplos, explanações, analogias, comparações e descrições familiariza o usuário com a interface através da imitação do diálogo humano.
- **Naturalidade:** A comunicação com o usuário deve ser a mais natural possível, sem a exigência de conhecimento de termos específicos, fora de seu cotidiano.
- **Satisfação:** O usuário deve ficar, ao usar a interface, sem frustrações quanto à espera, dificuldade, falta de assistência, etc.
- **Passividade:** Preferencialmente, o controle da interação deve ficar por conta do usuário, isto implica na interface assumir um papel passivo nesse diálogo.

A escolha de um sistema operacional com uma biblioteca gráfica para o desenvolvimento de uma interface amigável, intuitiva e padronizada foi fundamental para a boa qualidade da interface do SE-MNBC. O sistema operacional Windows, cada dia mais difundido no mercado, foi o mais adequado para a implementação do SE-MNBC, pois disponibiliza diversos recursos gráficos aos seus programadores. Seu ambiente gráfico simula uma mesa de trabalho, fazendo uma melhor utilização da tela e transmitindo informações de uma maneira visual rica [PET 92]. A interface gráfica do Windows além de mais rica visualmente, também transmite um grande número de informações ao usuário. Os programas desenvolvidos para esse ambiente têm uma mesma aparência e modos de operação semelhantes, sendo, portanto mais fáceis de aprender e usar [PER 96]. Além disso, o uso de características multimídia fortalece ainda mais o desenvolvimento de interfaces amigáveis, auxiliadas pelo uso de som, imagens e animações.

5.2.4.2 Parser de Entrada de Regras de Extração

O SE-MNBC tem um *parser de entrada* para as regras de extração, conforme a linguagem de extração definida [SCA 97a] [ZAM 2001]. Tais regras apresentam formato *script* e são digitadas pelo *usuário* como um texto ou, de forma mais visual, usando recursos da *interface*, como painéis de botões, com o auxílio do *mouse*. Além disso, o *parser* verifica a sintaxe das regras na busca de possíveis erros de digitação ou de uso indevido da linguagem.

5.2.4.3 Gerador da Base de Conhecimento

O *gerador da base de conhecimento* grava e lê as regras na *BC*, estando diretamente relacionado com o *parser de entrada de regras de extração*. As regras são armazenadas na *BC* no formato texto.

5.2.4.4 Parser de Entrada de Termos para Dicionários

O SE-MNBC disponibiliza um *parser de entrada* para os termos que compõem os diversos *dicionários* que auxiliam o processo de extração. Diferentemente do *parser de regras*, usa-se neste sub-módulo uma *interface* no formato de formulário. O *usuário* preenche os campos de informação existentes no formulário para a gerar e manter os *dicionários*, conforme um conjunto de restrições de entrada de dados pré-definidas. Sempre que uma entrada não prevista for realizada, o sistema impede o avanço para o próximo campo e emite uma mensagem. Este formato foi escolhido devido à simplicidade das informações manipuladas: listas de termos.

5.2.4.5 Gerador de Dicionários

O *gerador de dicionários* grava e lê os arquivos de dados estruturados do tipo relacional com os termos definidos pelo *usuário* para cada *dicionário* através do *parser de entrada de termos para dicionários*.

5.3 Base de Conhecimento - BC

Extrair informações com precisão depende de *BCs* que representem bem o conhecimento de especialistas humanos (*usuário*) em EI. O sistema está sujeito à habilidade do especialista em criar boas regras de extração. Tais regras podem ser inicialmente fundamentadas em um subconjunto dos documentos a serem analisados pelo sistema que também tenha sido manualmente analisado. Estudando os documentos e os resultados manuais, o especialista gera a primeira versão das regras, normalmente sem obter boa qualidade de extração automática. Contudo, o SE-MNBC permite que o *usuário* refine as regras, fazendo experimentos com cada uma delas até obter resultados satisfatórios. O *usuário* pode criar diversas *BCs* e utilizá-las conforme sua necessidade e caso de aplicação.

O SE-MNBC armazena em uma única *BC* as regras para todos os seus processos de extração, *classificação por estrutura*, *classificação por domínio* e *análise sintática e superficial*, valendo-se do módulo *configurador* para gerá-la. Esse último módulo disponibiliza visões individuais das regras de extração para cada processo, ou uma visão global da *BC*. Fisicamente, cada *BC* é formada por um arquivo texto que armazena as regras de extração.

5.4 Dicionários

Independentes da *BC*, os *dicionários* contem as palavras pesquisadas pelo processo de extração. Um *dicionário* é um arquivo estruturado relacional contendo um grupo de palavras relacionadas entre si, como, por exemplo, nomes de países. Um mesmo *dicionário* pode ser usado em regras de diferentes *BC*.

5.5 Bases de Dados

Conforme a Figura 5.1, cada módulo de extração tem uma ou mais *bases de dados de entrada* que são processadas gerando a *base de dados de saída.* A *base de dados inicial* (*N0*), ou primitiva, é composta por um conjunto de arquivos com diferentes conteúdos e estruturas de armazenamento de dados, sendo o ponto de partida do processo de extração. As demais *bases* são estruturadas conforme definido na metodologia EI-MNBC. Um processo pode valer-se de dados em todas as BDI anteriormente processadas ou da *N0*. Em um escopo mais amplo do que o do SE-MNBC, *N0* pode ser resultante de um processo de recuperação, por exemplo, e as demais *bases* utilizadas por outros sistemas de informações. As *bases de dados*, exceto *N0*, são arquivos texto estruturados (campos de dados separados por ponto-e-vírgula): um padrão de armazenamento simples e amplamente difundido para a fácil integração com processos de outros sistemas. As bases *N1*, *N2* e *N3* estão relacionadas entre si e com *N0* através de ponteiros URL, armazenados em suas tuplas, apontando para os arquivos não estruturados em *N0*.

5.6 Usuário

O *usuário* é caracterizado pela pessoa que configura o sistema ou usufrui os resultados do mesmo, conforme demonstra a Figura 5.1. O uso de *BCs* permite que o SE-MNBC seja manipulado por dois tipos de *usuário*, correspondendo a papéis distintos que podem ser assumidos pela mesma pessoa. Isto possibilita uma maior flexibilidade de uso do sistema frente aos conhecimentos exigidos do *usuário*.

a) Usuário Configurador: pessoa responsável pela geração da *BC*, utilizando o módulo *configurador* e possuindo conhecimento técnico do SE-MNBC e da forma de representação do conhecimento usada nele, bem como um raciocínio lógico formal com relação à estruturação desse conhecimento. Deve analisar manualmente os arquivos primitivos e os termos e padrões que identificam as informações relevantes ao *usuário* final. Essa análise é necessária para a geração de uma *BC* de boa qualidade para uma extração precisa. Cabe salientar que diferentes *usuários* podem trocar *BCs* aplicadas a diversos tipos de arquivos primitivos, facilitando o processo de configuração, já que as *BC* são independentes do sistema.

b) Usuário Final: pessoa que ativa o SE-MNBC e indica os arquivos a serem tratados, sem gerar *BC*, apenas utilizando as já existentes, criadas por *usuários* configuradores. O *usuário* final deve conhecer o ambiente gráfico, padrão Windows, para instalar e executar o SE-MNBC e acessar as BDI resultantes da extração.

5.7 Visão Geral do Processo de Extração Usando o SE-MNBC

O processo de extração do SE-MNBC pode ser dividido em quatro fases básicas:

a) Primeira: o *usuário*, através do *configurador*, gera a *BC* necessária para a configuração dos módulos de extração, viabilizando a adaptação do sistema aos diversos tipos de arquivos de entrada. Uma *BC* pode ser continuamente refinada a fim de melhorar a qualidade dos resultados e portar o sistema para aplicações.
b) Segunda: o *usuário*, através do *configurador*, gera os *dicionários* de termo usados no processo de extração. Contudo, esta fase pode ser esporádica, pois os *dicionários* podem ser previamente configurados e reaproveitados.
c) Terceira: *usuário final* ativa os processos de extração, analisando os arquivos primitivos. Os módulos do SE-MNBC podem ser executados individualmente, inclusive em paralelo, conforme as necessidades do *usuário* e as informações já disponíveis em cada BDI. As regras da *BC* orientam os processos de extração.
d) Quarta: o *usuário final* analisa as informações extraídas e armazenadas em *N1*, *N2* e *N3*. Nesta fase também pode ocorrer a integração com outros processos de extração que utilizem os resultados do SE-MNBC.

5.8 Implementação

A implementação do SE-MNBC baseou-se na arquitetura apresentada e, conseqüentemente, na metodologia EI-MNBC. A seguir encontram-se algumas informações técnicas relativas ao seu desenvolvimento e uso.

5.8.1 Ferramenta de Programação

O SE-MNBC foi desenvolvido com a ferramenta de programação Delphi 3 [BOR 97], [BOR 97a], [BOR 97b], [CAN 96]. O uso dessa ferramenta facilitou o processo de programação e aumentou a qualidade do sistema gerado. O Delphi permitiu que o SE-MNBC usasse os recursos do ambiente Windows, proporcionando ao *usuário* final os benefícios desse sistema operacional, principalmente quanto à *interface*. A escolha do Delphi para a implementação do SE-MNBC resume-se a duas palavras: velocidade e facilidade de uso, conforme as seguintes características da ferramenta:

a) Possui um ambiente de desenvolvimento integrado para Windows, com características de programação visual. A maior parte do programa é desenvolvida em um ambiente gráfico com o uso do *mouse*. Muitos resultados de programação são conhecidos sem a necessidade de compilação, ganhando-se tempo e economizando-se esforços.

b) A linguagem utilizada, Pascal, foi ampliada e melhorada, tendo sido adapta-la à programação com objetos e sendo mais intuitiva e simples com o uso de componentes visuais.

c) Assistentes de codificação (*wizards*) aceleram a digitação de comandos e funções, além de mostrarem sua sintaxe correta ou os parâmetros necessários. Com isso, há uma redução de erros e conseqüente otimização do tempo de desenvolvimento.

d) Aumento de produtividade com o uso de um editor de código profissional integrado ao ambiente Windows, com indicações de erros de sintaxe e *debugger* para a execução passo-a-passo do código fonte.

e) O compilador gera rápida e eficientemente código executável nativo 32 *bits*, sem a necessidade de bibliotecas de *run-time.*

f) Boa performance do código executável gerado, auxiliando na boa velocidade dos processos de extração.

5.8.2 Requisitos de Execução

Os requisitos mínimos de equipamento para o uso do SE-MNBC são os seguintes:

a) Sistema operacional Windows 95 ou superior;
b) Processador Pentium™ ou compatível;
c) 32 Mb de memória principal;
d) 12 Mb disponíveis no disco rígido.

Porém, com o objetivo de reduzir o tempo de processamento, recomenda-se 128Mb de memória principal. O espaço em disco depende do volume de documentos a ser processado.

6 Avaliação Experimental

Neste capítulo, apresenta-se um estudo aplicado do sistema SE-MNBC, baseado na metodologia EI-MNBC, avaliando a hipótese e os conceitos abordados, bem como os conhecimentos adquiridos em oito anos de pesquisa em RI. A validação de uma hipótese através de experiências, passíveis de serem repetidas por outros pesquisadores, é um dos principais meios de desenvolvimento do conhecimento científico [CAM 69].

6.1 Economia Globalizada, Competição e Trade Points

A crescente necessidade de transmitir e controlar informações comerciais em torno do mundo é de grande importância para a eficiência comercial das competitivas economias de hoje. As rápidas mudanças ocorridas para a remoção das barreiras comerciais de muitos países e as oportunidades oferecidas pelas novas tecnologias de comunicação estão mudando o cenário econômico mundial [KOT 2000]. Em conseqüência disso, há uma fonte contínua de novos produtos, serviços, e conhecimento. Companhias devem tratar de ciclos de vida mais curtos e de uma obsolescência mais rápida, de reduzir os custos unitários, de crescentes investimentos, de riscos mais elevados, dos critérios de fixação de novos preços, e de novos campos de competição. Esta rápida mudança organizacional das companhias e dos processos de produção gera a necessidade de serviços comerciais novos, que coletam, processam e disseminam a informação comercial eletrônica de forma rápida e precisa, usando as tecnologias de comércio eletrônico [KOT 2000].

O programa Trade Point é uma rede internacional de negócios, criada pela UNCTAD (*United Nations Conference on Trade and Development*), organismo da ONU responsável pelo comércio internacional. Esse programa começou em fevereiro de 1992, por ocasião da VIII Sessão da UNCTAD em Cartagena, Índia, com o objetivo inicial de estabelecer 16 pontos de comércio (Trade Points) pilotos [ELE 2002]. Essa primeira fase culminou no Simpósio Internacional das Nações Unidas em Eficiência Comercial, em Columbus. Nesse simpósio, mais de 2000 líderes de setores públicos e privados, incluindo primeiros ministros, vice-primeiros ministros e cerca de oitenta ministros de 136 países, lançaram a Iniciativa de Eficiência Comercial (*Trade Efficiency Iniciative* - TEI), criando oficialmente a Rede Global de Trade Points (*Global Trade Point Network* - GTPN). O principal objetivo da TEI é abrir o comércio internacional a novos participantes, especialmente PME, por meio da simplificação e da padronização de procedimentos de comércio de todo o mundo, acompanhadas de acesso a tecnologias avançadas e a redes de informação [ELE 2002]. Essa foi uma primeira apresentação a nível mundial do importante papel que as estradas de informação globais podem exercer no campo do comércio e do desenvolvimento [TRA 2001]. Os Trade Points apresentam as seguintes atividades [ELE 2002]:

- Facilitar negócios, agrupando fisicamente ou de maneira virtual os participantes de transações de comércio exterior (despachantes aduaneiros, câmaras de comércio, transportadoras, bancos, seguradoras, etc.), fornecendo os serviços requeridos por essas transações a um custo razoável.
- Prover informações sobre comércio internacional, com dados sobre oportunidades de negócio e mercado, clientes e fornecedores potenciais, regulamentos e exigências comerciais internacionais, etc.

- Fornecer portas de acesso a redes globais de comércio, operando de maneira interligada, via Internet.

A participação brasileira no programa de Trade Points é coordenada pelo Departamento de Promoção Comercial (DPR) do Ministério das Relações Exteriores (MRE) [TRA 2001]. Dentro do DPR, sua coordenação incumbe à Divisão de Programas de Promoção Comercial (DPG), que atua como Secretaria Executiva do Fórum Brasileiro de Trade Points.

O Trade Point Porto Alegre proporciona oportunidades de negócios no estado do Rio Grande do Sul. Ele é um centro de negócios formado por iniciativa do SEBRAE (Serviço Nacional de Apoio à Micro e Pequena Empresa), da Prefeitura Municipal de Porto Alegre, do BRDE (Banco Regional de Desenvolvimento do Extremo Sul), da FIERGS (Federação das Indústrias do Estado do Rio Grande do Sul) e da FEDERASUL (Federação das Associações Comerciais do Rio Grande do Sul). O centro segue o modelo ditado pela UNCTAD e tem apoio da UFRGS (Universidade Federal do Rio Grande do Sul), do SEDAI (Secretaria Extraordinária para Assuntos Internacionais do Governo do Estado) e do SECAR (Secretaria de Captação de Recursos) [TRA 2001].

6.2 Rede Global de Trade Points – GTPN

A GTPN baseia-se nas mais avançadas tecnologias de redes de comunicação e multimídia. Uma de suas características mais usadas é seu Web Site na Internet. Graças aos Trade Points, todos os países e empresas conectados podem trocar ETOs (*Electronic Trading Opportunities*), bem como outros tipos de informações a respeito de regulamentos comerciais, práticas e conhecimentos de mercado [ELE 2002].

Visto o aumento do número de Trade Points conectados, esta rede de comércio global emergirá rapidamente como uma das principais redes mundiais para a troca de informações relacionadas a negócios, principalmente via Internet [ELE 2002]. Tornar-se-á cada vez mais difícil aos negociantes ficarem fora dela, e o aumento subseqüente no tráfego permitirá reduções substantivas no custo de coletar, formatar, transmitir e processar tal informação. Isso permitirá um contínuo aumento do número de beneficiários dos Trade Point, especialmente as PME, e um aumento do avanço tecnológico envolvendo a GTPN, com o reinvestimento em pesquisa e em desenvolvimento.

A conexão ponto-a-ponto, via Internet, permite aos Trade Points conectarem-se a lugares tradicionalmente excluídos das chamadas estradas de informação. Uma característica importante da GTPN é sua descentralização. Os Trade Points armazenam dados a nível nacional e fornecem acesso a dados de outros Trade Points.

6.3 ETO - Eletronic Trade Opportunity

Há alguns anos, o elevado custo dos sistemas de TI (Tecnologia da Informação) e de serviços de telecomunicações avançados eram recursos inacessíveis para PME. O sistema de ETOs fornece tais recursos em escala internacional, oferecendo benefícios previamente apreciados somente por grandes organizações. Ele foi desenvolvido pelo UNTPDC (*United Nations Trade Point Development Centre*), uma organização de suporte ao programa TEI [ELE 2002]. O objetivo do UNTPDC é ser um centro para ampliar a sustentação tecnológica ao comércio, provendo suporte técnico e conceitual para as avançadas GTPNs (*in state-of-the-art*). Esse sistema responde a três importantes problemas:

- Necessidade de aumentar a consciência internacional das possibilidades de uso da TI moderna para auxiliar na solução dos problemas de comércio.
- Permitir a aplicação eficaz da TI ao comércio.
- Promover o uso de modelos capazes de reduzirem o custo processual no comércio internacional.

O sistema de ETOs junta assinantes de todo o mundo em um único ponto de contato para negociações, investimentos e oportunidades de negócios. Atualmente estão conectados 155 Trade Points e 10.000 corporações em 75 países desenvolvidos, 60 em desenvolvimento e 20 sub-desenvolvidos [TRA 2001]. O sistema gera mais de 130.000 registros mensais, resultando em quase 13 GB de informação para comércio cada mês. Atualmente, o roteador de ETOs transfere mais de 2 milhão de correspondências eletrônicas por dia com oportunidades comerciais para 10.000 organizações em diversos países. Esses cálculos baseiam-se em transferências diretas de ETOs do roteador à caixa de correio eletrônico dos usuários. Cada ETO é emitido mais de 10.000 vezes aos usuários individuais, livres de taxas. Isso permite que os Trade Points e diversas organizações troquem informações comerciais usando texto semi-estruturado transmitido por correspondências eletrônicas [ELE 2002]. As mensagens estão disponíveis em uma base de dados global para a distribuição eletrônica de informações comerciais.

Um ETO é um arquivo de correspondência eletrônica, apresentando um cabeçalho com informações semi-estruturadas em três campos de dados, o que facilita o processo de extração. O primeiro campo contém o endereço de correio eletrônico do emitente do ETO; o segundo, a data de envio; e o terceiro é o campo de assunto. Apesar desse último campo apresentar nas correspondências eletrônicas conteúdo e formatação livre, no caso dos ETOs, existe uma formatação padronizada, estando limitado a 40 caracteres e resumindo o resto do conteúdo do arquivo ETO [ELE 2002]. O início desse resumo apresenta a classificação do ETO em caracteres maiúsculos, nas seguintes categorias: OFFER para ofertas, DEMAND para demandas, e MISC para outros assuntos comerciais. Após essa categoria, encontra-se o caractere dois-pontos (“:”) e, entre colchetes, duas letras maiúsculas identificando o país de origem do ETO, conforme especificação ISO para nomes de países [ELE 2002]. Depois da informação de país, deve-se colocar somente o nome do produto ou serviço oferecido ou procurado, no caso de OFFERs ou DEMANDs, ou, se for da categoria MISC, uma descrição sucinta do conteúdo do ETO. Maiores informações são colocadas no corpo do ETO.

O corpo do ETO não apresenta uma estrutura padronizada, contendo texto livre com informações organizadas conforme a livre decisão do autor. Contudo, conforme orientação da UNTPDC, alguns dados sobre o emissor do ETO devem estar presentes no corpo desse: Empresa/Companhia, Endereço, País, Telefone, Fax, E-Mail e Endereço da Página na Internet (*Home Page*). Tais informações são normalmente apresentadas na forma de tópicos cuja localização no corpo do ETO pode variar. Apesar dos Trade Points usarem formulários para gerarem os ETOs através de páginas HTML, esses formulários podem armazenar os tópicos acima em diferentes posições do arquivo texto, dificultando sua extração. O restante do corpo do ETO é formado por texto livre, onde o autor coloca outras informações comerciais sem nenhuma padronização prevista. Orienta-se colocar uma completa descrição do produto ou serviço, forma de pagamento, custo, embalagem, etc. É proibida a comercialização de narcóticos, pornografia, ou qualquer produto ou serviço ilegal. Arquivos anexados não são aceitos. A maioria dos ETOs é escrita em inglês, conforme mostra a Figura 6.1.

From: dinesh@acer.co.ae [SMTP:dinesh@acer.co.ae]
Sent Date: Thursday, 14 de December de 2000 11:39
To: offer-mail@heuristic.untpdc.org
Subject: OFFER: [AE] ACER NOTEBOOKS SPECIAL OFFER

Dear Sir/Madam,

We currently have a special offer for the notebook behind described, we are pleased to submit our offers for your perusal:

TravelMate 737TLV
PIII 700 / 128 MB / 18 GB / 6x DVD / FDD / 15" TFT / Integ.
Fax / Integ. LAN / Win 2000 (English)
Special Unit Price : US$ 2349/-

Terms & Conditions:
Prices are ex-works Jebel Ali Free Zone, Dubai, United Arab Emirates
Validity of offer : 3 days
Payment terms : 100% T/T advance or Irrevocable LC as per format (to be supplied)
Delivery : Within 3 working days

Please visit our website for any detailed information regarding the product or our company. We look forward to your consi

Contact: dinesh@acer.co.ae

Dinesh P. (Sales Manager)
Acer Computer M. E. Ltd.
Jebel Ali Free Zone
Dubai -, United Arab Emirates (-)
Tel: +971-4-8813111
Fax: +971-4-8812200
URL: http://www.acer.co.ae

This ETO has been posted to UNTPDC by:
****
OPTIMA EUROPEAN TRADE BOARD
since 1995
Official European bulletin board for posting offers, business
opportunities, and responses to them. Posting is free.
http://www.trade-board.com/
****

FIGURA 6.1 - Exemplo de um ETO

6.4 Pequenas e Médias Empresas - PME

O comércio eletrônico é provavelmente a única ferramenta de *marketing* com a qual PMEs podem atingir um mesmo potencial de negócios que as grandes, com um baixo investimento [RYN 2001]. O sistema SE-MNBC pode auxiliar, a baixo custo, na seleção e manipulação do enorme volume de informações comerciais disponível pelo sistema ETO. O SE-MNBC pode estruturar as informações existentes nos ETOs, permitindo sua consulta através de SGBDs. Isso melhora ainda mais a posição competitiva das PMEs com seus restritos orçamentos para investimentos em tecnologias de TI visando o processamento dos ETOs, ou no inviável processamento manual. Dessa forma, o experimento de avaliação da metodologia EI-MNBC utiliza os ETOs como fonte de dados. Executaremos os processos de extração definidos para o tratamento de documentos de comércio eletrônico sobre as mensagens distribuídas pelos Trade Points pela Internet. Assim, iremos cumprir o objetivo da tese criando uma Metodologia de Extração de Informações para o Tratamento e Manipulação de Informações de Comércio Eletrônico armazenadas nos ETOs.

6.5 Plano Experimental

Um plano experimental fornece os passos para a execução de um experimento, de forma a torná-lo válido [CAM 69]. Formulada a hipótese (metodologia EI-MNBC), definem-se as informações a serem extraídas, as variáveis envolvidas no experimento, as métricas de avaliação desse e suas etapas.

6.5.1 Variáveis

O plano experimental usa como ferramentas de execução os sistemas de extração selecionados. Conforme os experimentos do MUC, dois sistemas de extração serão utilizados: um manual e outro automático, o SE-MNBC, permitindo a comparação de seus resultados [LEH 94]. Toma-se como ideal de qualidade os resultados manuais, obtidos por especialistas humanos. Assim, mede-se quanto os resultados automáticos aproximam-se desse ideal, determinando a qualidade do SE-MNBC e da metodologia que esse representa.

A realização dos experimentos prevê a participação de dois usuários. Devido à objetividade do experimento definido e de sua avaliação, os resultados não podem sofrer efeitos ou distorções por parte dos participantes. Um dos usuários executará a extração manual e o outro gerará a BC para o sistema SE-MNBC e o executará.

A base de dados utilizada no experimento é formadas por arquivos dos ETOs disponíveis nos primeiros 20 dias do mês de dezembro de 2000, outros arquivos com informações de comércio eletrônico e arquivos diversos. Isso caracteriza um caso de aplicação real: grande quantidade e variedade de informações dinâmicas que podem atender diferentes necessidades de extração.

O ambiente de execução do processo experimental é caracterizado pelo computador e sistema operacional usados:

- Sistema Operacional: Windows 98SE.
- Número de processos executando no computador estranhos ao sistema SE-MNBC ou ao Windows 98SE: zero.
- Computador: processador AMD Duron 950 Mhz, 128 Mb de memória RAM e disco rígido de 40 Gb.

6.5.2 Métricas

A avaliação dos resultados do processo experimental baseia-se métricas de *recall* e *precision*, utilizadas também no MUC. Além dessas, a velocidade de extração será avaliada pela razão entre o volume de dados processados e o tempo de processamento.

6.5.3 Definições

As informações a serem extraídas, tanto pelo processo manual, quanto pelo automático, representam a necessidade do usuário. A definição dessas informações margeia a extração manual pelos especialistas humanos e a criação das regras na BC para configuração do SE-MNBC.

Categorias estruturais: arquivos texto (TXT), como é o caso dos ETOs, documentos no padrão MSWord (DOC) e documentos compactados no padrão ZIP. Arquivos muito antigos, anteriores a 10 de outubro de 2000, ou muito grandes, acima de 20Kb, devem ser descartados. Assim, não são processadas pelo processo P2 e posteriores, informações desatualizadas ou com grande custo de processamento.

Categorias de assunto/domínio: documentos de comércio eletrônico sobre celulares e computação.

Dados a extrair dos documentos: deve-se extrair dos documentos de comércio eletrônico as seguintes informações: *remetente, assunto, tipo (demanda ou oferta), país,*

produto, endereço, telefone, website, data de envio e *validade da proposta.* As informações de *website* e *validade* são extraídas somente de documentos da categoria computação, diferenciando as regras de extração para os dois domínios definidos.

6.5.4 Etapas

Serão realizados dois experimentos divididos nas etapas a seguir:

Experimento A: avalia a qualidade do processo de extração.

1. Realizar o processo de extração manual (usuário A) sobre parte da base de dados de avaliação, pois essa tarefa seria inviável sobre toda a base.
2. Criar a BC para o SE-MNBC (usuário B) a fim de extrair dos mesmos arquivos analisados manualmente as informações definidas como relevantes.
3. Realizar o processo de extração automatizado sobre os arquivos analisados manualmente.
4. Comparar os resultados manuais ("ideais") com os obtidos pelo SE-MNBC usando métricas de *recall* e *precision*.

Experimento B: avalia a velocidade e a capacidade de extração do sistema desenvolvido conforme a metodologia EI-MNBC.

1. Realizar o processo de automatizado sobre toda a base de dados de teste usando a BC do experimento A.
2. Medir o tempo de extração para cada 2000 arquivos de dados e o volume de arquivos processados em cada fase de extração.
3. Analisar os resultados buscando uma constante que descreva a relação volume por tempo e volume por processo.

6.6 Experimento A

O experimento *A* valida a qualidade de extração da metodologia EI-MNBC, implementada no sistema SE-MNBC, processando 300 arquivos (1,3 Mb) distribuídos em 4 diretórios no computador de teste, formando a base de dados *N0*. Utilizou-se um sub-conjunto da base de testes, pois seria inviável o processamento manual de toda essa base para a geração dos resultados "ideais", que servem de parâmetro para a avaliação. O objetivo deste experimento é comprovar a eficácia da metodologia na (**I**) extração e classificação dos arquivos por estrutura de armazenamento; (**II**) seleção de arquivos com características estruturais específicas: estrutura interna, tamanho e atualidade; (**III**) tratamento de arquivos com diferentes estruturas internas; (**IV**) extração e classificação dos documentos por domínio; (**V**) extração das informações relevantes para o usuário; (**VI**) criação de uma base de dados estruturada que atenda as necessidades de informação do usuário e facilite a análise dos dados extraídos.

Inicialmente, executou-se o processo de extração manual dos 300 arquivos. Os documentos foram lidos e classificados, e as informações relevantes destacadas, conforme definido anteriormente. Esse processo foi realizado por um único usuário e conferido por um segundo, garantindo sua isenção e validade. Um terceiro usuário teve acesso a uma amostra de 15 arquivos de cada domínio definido como relevante dos 300 que formam a bases de dados deste experimento. Juntamente como as definições anteriores, esse usuário criou a BC para a extração automática. Essas regras puderam ser refinadas através de execuções passso-a-passo do SE-MNBC utilizando os arquivos de dados de amostras. A seguir apresenta-se uma descrição detalhada do processo de

extração automático realizado e uma análise quantitativa dos resultados, comparando os processos manual e automático:

Processo *P1* - Classificação por Estrutura: A partir da base de dados *N0*, *P1* (**I**) extraiu informações sobre as características externas dos arquivos (extensão de nome, tamanho, data de criação/modificação); (**II**) descartou 41 (0,2 Mb) por não estarem nos diretórios de dados definidos pelo usuário, selecionando 259 arquivos (1.1Mb) em 2 diretórios. Dentre esses arquivos, 11 (684Kb) foram descartados por serem maiores que 20Kb, 14 (81,8Kb) por serem anteriores a 10 de outubro de 2000, e 3 arquivos (14,2Kb) por terem extensão diferente das definidas nas regras de extração (DOC, TXT e ZIP). Sobraram 231 arquivos (357Kb - 3 arquivos compactados com a extensão ZIP, 2 documentos no padrão MSWord, extensão DOC, e 226 arquivos texto, extensão TXT). (**III**) Os arquivos com a extensão ZIP foram desdobrados, ou seja, descompactados, por um programa externo ativado pelo SE-MNBC, resultando em 28 novos arquivos (1 executável, extensão EXE; 1 com a extensão DIZ e 26 arquivos texto, com a extensão TXT), substituindo-se, portanto, os 3 arquivos compactados por 28 outros arquivos (55Kb) na base de dados *N1*, que passou a ter 256 arquivos (391Kb). Contudo, os arquivos desdobrados também passaram pela fase de seleção, onde se descartou 1 arquivo texto, cujo tamanho era maior que 20Kb e 5 arquivos (com as extensões EXE e DIZ), pois eram anteriores a 10 de outubro de 2000. Assim, *N1* ficou com 250 arquivos (362Kb), sendo formada por 2 documentos no padrão MSWord, extensão DOC, e 248 arquivos texto, extensão TXT. (**IV**) A última fase de *P1* converteu os 2 arquivos DOC para TXT. Ao final, a base de dados *N1* continha 250 arquivos texto (362Kb).

Somente os arquivos que atendiam as restrições definidas nas regras de extração foram recuperados. Não se descartou nenhum arquivo que devesse passar para o processo *P2*. Além disso, todos os arquivos selecionados foram corretamente classificados estruturalmente, conforme comparativo com os resultados manuais. Assim, os índices de *recall* e de *precision* foram de 100%. O tempo de execução de *P1* foi de 9 segundos.

Processo *P2* – Classificação por Domínio: dos documentos selecionados por *P1*, (**I**) retirou-se as *stop words*, conforme grupos de palavras por categoria estrutural definidos nas regras de extração: pronomes, artigos, palavras que ocorrem freqüentemente em todos os documentos, entre outras. (**II**) Os documentos tiveram seus atributos (palavras) discriminados e ordenados, conforme sua freqüência absoluta e relativa, selecionando-se para representar cada documento os 15 atributos que melhor identificavam seu conteúdo. (**III**) Classificou-se cada documento através da análise de similaridade entre eles e as categorias definidas pelo usuário. Um documento foi classificado em somente uma categoria, aquela com a qual obteve maior grau de similaridade. Os documentos com grau de similaridade zero com todas as categorias foram descartados de *N2*, sendo classificados como *sem categoria*. A categoria *computação* foi definida pelas palavras “motherboard”, “motherboards”, “byte”, “bits”, “computer”, “computers”, “software”, “hardware”, “notebook”, “notebooks”, “laptop” e “laptops”; e a *celular* pelas palavras “mobile”, “nokia”, “celular”, “batt” (e suas variações, como “battery”) e “accessories”. Essas definições basearam-se na análise manual de 15 documentos de cada categoria. Os pesos de cada atributo foram gerados pela freqüência relativa em que eles apareceram nesses 30 documentos.

A tabela 6.1 apresenta os resultados de *P2*. Selecionou-se em *N2* 89 documentos das categorias de interesse do usuário: 35 em *celular* e 54 em *computação*. Dentre os 35 documentos classificados em *celular*, 3 eram sobre *computação*, estando entre os 86

documentos não classificados nessa última categoria. Um documento de *celular* foi incorretamente colocado em *sem categoria* e os demais 32 esperados nessa categoria foram corretamente classificados. Dos 140 documentos esperados na categoria *computação*, 54 foram corretamente classificados, 3 incorretamente colocados em *celular* e 83 em *sem categoria*. Nenhum documento foi incorretamente classificado como *computação*. Quanto aos 77 documentos esperados em *sem categoria*, todos foram corretamente classificados. Contudo, 84 documentos, 83 de *computação* e 1 de *celular*, foram colocados nessa categoria. Dois documentos em *N2* não são sobre comércio eletrônico. Um deles é um arquivo texto, contendo um Contrato de Licenciamento de Software, cujo grau de similaridade com a categoria *computação* foi de 14%, pois ele apresenta alguns termos, como "software" e "hardware". O outro é um arquivo texto sobre aparelhos celulares com grau de similaridade de 13% com a categoria *celular*.

TABELA 6.1 - Resultado Final do Processo P2 – Experimento A

	Número de documentos esperados em cada categoria (A)	**Corretamente classificados (B)**	**Incorretamente classificados (C)**	***Recall*** **R=B/A**	***Precision*** **P=B/(B+C)**
Celular	33	32	3	96,97	91,43
Computação	140	54	0	38,57	100,00
Sem Categoria	77	77	84	100,00	47,83
Totais	250	163	87		
Média				78,51	79,75

A maior qualidade de classificação associada à categoria *celular* deve-se a melhor seleção das palavras que a definem pelo sistema de clustering. Pelo mesmo motivo, o índice de *recall* da categoria *computação* foi baixo, visto o restrito conjunto de palavras, manualmente definidas, para representá-la. No geral, a média de *recall* e *precision* foi razoável, podendo ser melhorada através de categorias melhor definidas. O tempo de execução do processo P2 foi de 5 minutos e 9 segundos.

A categoria *celular* apresentou melhor *recall* do que a *computação*, pois seu conjunto de atributos foi refinado. Fez-se uma primeira execução de *P2*, e, com base nos iniciais resultados obtidos, ajustou-se os atributos de *celular*. A categoria *computação* não foi refinada, mantendo-se os atributos manualmente selecionados, sem nenhum teste de aplicação e decorrente melhoria dos atributos. Isso gerou um baixo índice de *recall*, pois os atributos não representavam todos os documentos esperados nesse domínio. Contudo, esses atributos não permitiram a classificação de documentos de outras categorias, conforme demonstra o índice de *precision*. Essa comparação mostra uma vantagem da metodologia EI-MNBC, que prevê a avaliação parcial do processo e extração antes de ativar o seguinte. No geral, a média de *recall* e *precision* foi boa, podendo ser melhorada através de novos refinamentos. O F-*score* de *P2* é de 79,13, com pesos iguais de *recall* e *precision*.

Processo 3 – Análise Superficial: Foram processados em *P3* os 89 documentos existentes em *N2*, sendo que 2 desses não eram ETOs. Três dicionários de termos associados aos domínios de interesse auxiliaram em *P3*: endereços, produtos e países. Os termos nos dicionários foram escolhidos com base na análise manual de uma amostra de 15 documentos de cada categoria relevante, onde esses apareciam freqüentemente como indicativo de um possível endereço, produto ou país.

Para a extração de informações de *produto* foram criadas seis diferentes regras com o objetivo de recuperar características distintas referentes a um mesmo produto. Além disso, as diversas regras podem melhorar a qualidade dos resultados, pois a informação desejada pode ser identificada no documento de diversas formas.

Conseqüentemente, *N3* tem seis campos para armazenar os resultados obtidos a partir de cada regra. Isso explica o valor de 308 informações sobre *produto* existente na tabela 6.2, embora tenham sido processados em *P3* apenas 89 documentos. Essa abordagem de uso de regras diferentes para a extração de uma mesma informação também permitiu através de refinamentos selecionar a regra mais eficiente, ou combiná-las na busca de melhores resultados.

A extração de *endereço* usou duas regras e dois campos de resultados em N3: *end_base* e *end_simples*. Ambas as regras procuram por um termo indicativo da informação de *endereço*, como, por exemplo: "Rua", "Avenida", "Bloco", etc. Contudo, a primeira regra (campo *end_base*) extrai todo o segmento de texto a partir do termo indicativo até encontrar, nas próximas quatro linhas, um termo que indique um país. Essa característica ocorre freqüentemente em informações de *endereço*, permitindo uma extração de melhor qualidade (com menos informação irrelevante). A segunda regra (campo *end_simples*), simplesmente extrai todo o segmento de texto a partir do termo indicativo até três linhas depois desse, o que facilita a extração, em alguns casos, de conteúdo irrelevante. Caso a primeira regra extraia o *endereço*, a segunda não é executada, conforme definido na BC. Assim, as duas regras se combinam, visando uma melhor qualidade de extração:

```
IF Pesquisa_Dic(endereços, CxLivre, 8, 100) AND Verifica_Dic(países, CxLivre, 1, 4, LINHA, DEPOIS)
    THEN BEGIN
        Move(1, LINHA, ANTES)
        Copia_Até(dic, países, 1, CxLivre, 4, LINHA, DEPOIS, NALT, end_base)
        Move(1, LINHA, DEPOIS)
    END
    ELSE BEGIN
        Copia(3, LINHA, NALT, end_simples)
        Move(1, LINHA, DEPOIS)
    END;
```

A avaliação do processo P3 é apresentada na tabela 6.2, conforme as variáveis a seguir:

Extr. Exata (A): a informação extraída é exatamente igual à definida pelo processo de extração manual.

Extr. Corr. + (B): a informação extraída contem a informação definida manualmente como relevante e outras irrelevantes.

Extr. Corr. - (C): a informação extraída representa parte da informação definida manualmente como relevante.

Extr. Incorr. (D): a informação extraída não foi definida como relevante pelo processo manual.

Não Extr. Corr (E): a informação desejada não foi extraída, pois não existe no arquivo.

Não Extr. Incorr. (F): a informação desejada não foi extraída, mas existe no arquivo, conforme definido pelo processo manual.

Itens Relev. Extr. (G): total de itens relevantes extraídos (A+B+C).

Itens Relev. (H): total de itens relevantes existentes em *N2*, conforme definido na extração manual (A+B+C+F).

Itens Extr. (I): total de itens extraídos, independentemente de sua relevância, definida na extração manual (A+B+C+D).

***Recall* Amplo (RA):** índice percentual de *recall* (G / H).

***Precision* Amplo (PA):** índice percentual de *precision* (G / I).

***Recall* Exato (RE):** índice percentual de *recall* (A / H).

***Precision* Exato (PE):** índice percentual de *precision* (A / I).

A quantidade total de informações apresentadas na tabela 6.2 pode ser maior ou igual ao número de documentos processados em P3: 89. Esse total é igual quando os documentos contem somente uma vez o tipo de informação definida como relevante ou não a contem; e maior, quando contem essa informação mais de uma vez, como, por exemplo, dois telefones de contado.

As informações de *remetente, data de envio, assunto, tipo* e *país*, por estarem no cabeçalho dos ETOs, que segue a estrutura de um e-mail, tiveram um índice de *recall* e *precision* de 100%. Isso confirma a ótima adaptação do SE-MNBC a dados semi-estruturados. Dos 89 arquivos processados, 46 eram do *tipo* oferta, 41 do *tipo* demanda e 2 não apresentavam essa informação (não são ETOs).

TABELA 6.2 - Resultado Final do Processo P3 – Experimento A

		Extração Exata (A)	Extr. Corr.. + (B)	Extr. Corr. - (C)	Extr. Incorr.eta (D)	Não Extr. Corr. (E)	Não Extr. Incorr. (F)	TOTAISS	Itens Relev. Extr. (G=A+B+C)	Itens Relev. Esperados (H=A+B+C+D+F)	Itens Extr. (I=A+B+C+D)	*Recall* Amplo (RA=G/H)	*Precision* Amplo (PA=G/I)	*Recall* Exato (RE=A/H)	*Precision* Exato (PE=A/I)
Informação Estruturada															
Remetente	Valor	87	0	0	0	2	0	89	87	87	87				
	%	97,75	0,00	0,00	0,00	2,25	0,00	100,00				100,00	100,00	100,00	100,00
Data de Envio	Valor	87	0	0	0	2		89	87	87	87				
	%	97,75	0,00	0,00	0,00	2,25	0,00	100,00				100,00	100,00	100,00	100,00
Assunto	Valor	87	0	0	0	2		89	87	87	87				
	%	97,75	0,00	0,00	0,00	2,25	0,00	100,00				100,00	100,00	100,00	100,00
Tipo	Valor	87	0	0	0	2		89	87	87	87				
	%	97,75	0,00	0,00	0,00	2,25	0,00	100,00				100,00	100,00	100,00	100,00
País	Valor	87	0	0	0	2		89	87	87	87				
	%	97,75	0,00	0,00	0,00	2,25	0,00	100,00				100,00	100,00	100,00	100,00
Informação Não Estruturada															
Endereço	Valor	65	50	12	1	2	5	135	127	132	128				
	%	48,15	37,04	8,89	0,74	1,48	3,70	100,00				96,21	99,22	49,24	50,78
Telefone	Valor	62	5	1	0	30	0	98	68	68	68				
	%	63,27	5,10	1,02	0,00	30,61	0,00	100,00				100,00	100,00	91,18	91,18
Website	Valor	18	0	0	1	31	4	54	18	22	19				
	%	33,33	0,00	0,00	1,85	57,41	7,41	100,00				81,82	94,24	81,82	94,74
Validade	Valor	3	0	0	0	51	0	308	3	3	3				
	%	5,56	0,00	0,00	0,00	94,44	0,00	100,00				100,00	100,00	100,00	100,00
Produto	Valor	189	47	8	14	3	47	308	244	291	258				
	%	61,36	15,26	2,60	4,55	0,97	15,26	100,00				83,85	94,57	64,95	73,26
Médias															
Informações Não Estrutur.												92.38	97.71	77.44	81.99
Geral												96.19	98.85	88.72	91.00
Formação da Informação de Endereço															
End_Base	Valor	42	7	9	0	2	15	75	58	73	58				
	%	56,00	9,33	12,33	0,00	2,67	20,00	100,00				79,45	100,00	57,53	72,41
End_Simples	Valor	23	43	3	1	2	5	77	69	74	70				
	%	29,87	55,84	3,90	1,30	2,60	6,49	100,00				93,24	98,57	31,08	32,86

O resultado de extração da informação *endereço* é gerado a partir da combinação das informações armazenadas nos campos *end_base* e *end_simples* da base de dados N3. No entanto, para apresentar como se chegou a esse resultado, combinando duas regras, as linhas finais da tabela 6.2 apresentam os resultados de cada regra em separado. Comprovou-se que a informação armazenada em *end_base* é mais precisa que a armazenada em *end_simples* (9,33% contra 55,85% na variável B). Por outro lado, o índice de *recall* baixou, pois existiam *endereços* sem a indicação do país, não sendo extraídos pela primeira regra (20,00% contra 6,49% na variável F). Na pesquisa por *endereços* sem o uso do dicionário de países ocorreu uma extração incorreta, o que não existiu na extração com dicionário, porque ambas as regras identificaram a possível existência da informação de *endereço*, a qual, neste caso, não existia. Contudo, a regra que utiliza dicionário não extraiu a informação incorreta, pois não existia o nome de um país até o limite de quatro linhas. A outra regra, que não usa dicionário, extraiu do documento o segmento de texto a partir do termo indicativo do possível *endereço* até três linhas após. Isso reduziu o índice de *precision* no resultado geral de *endereço*. Além disso, a regra de extração para *end_base* deixou de extrair 15 *endereços* relevantes, dos quais 10 foram extraídos pela regra subseqüente, geradora de *end_simples* e que não usa dicionário. Essas 15 falhas ocorreram porque a informação de país não se encontrava até a quarta linha após o indicador de *endereço* ou o país existente no arquivo não fazia parte do dicionário usado pela regra.

As informações de *telefone*, *website* e *validade*, embora estejam no corpo dos documentos, onde não há uma estrutura definida, obtiveram ótimos índices de *recall* e *precision*, pois normalmente havia termos que indicam a ocorrência delas. O resultado alcançado para a informação *produto* foi bom, com um alto índice de *precision* amplo (94,57%). O índice de *recall* amplo, mesmo ficando em 83,85%, poderia ser melhorado,

utilizando-se um dicionário de termos de produtos mais completo. O índice da variável D foi de 4,55%, demonstrando que poucas informações foram extraídas incorretamente.

A médias de *recall* e *precision* de *P3*, tanto amplas como exatas, quando se tratando apenas da extração de informações não estruturadas, são boas: 92,38% (RA), 97,71% (PA), 77,44% (RE), 81,99% (PE). O F-*score* amplo de *P3* é de 97,50 e o exato 89,85, ambos com pesos iguais de *recall* e *precision*. Isso confirma a boa aplicação do SE-MNBC para o tratamento de BDNE/SE, conforme a Tabela 7.2 com os resultados dos MUC. O tempo de execução do processo P3 foi de 10 minutos e 15 segundos.

6.7 Experimento B

O experimento *B* avalia a capacidade e velocidade de extração da metodologia EI-MNBC, implementada no sistema SE-MNBC, processando 10.024 arquivos (13,90 Mb) distribuídos em 4 diretórios no computador de teste, formando a base de dados *N0*. Dentre os arquivos utilizados no experimento *B*, estavam os 300 utilizados no experimento *A* e outros 9.724 ETOs no formato texto. O objetivo deste experimento é comprovar a eficácia da metodologia na (**I**) extração de informações a partir de grandes volumes de BDNE/SE; (**II**) distribuição do custo de processamento na proporção inversa entre complexidade de processamento e volume de dados; e (**III**) re-aproveitamento e adaptação de BC.

Neste experimento, o SE-MNBC utilizou a mesma BC do experimento anterior e as regras não foram alteradas ou refinadas durante o experimento. Isso demonstrou a viabilidade de re-aproveitamento da BC. O sistema foi executado em rodadas de 2.024 arquivos até os 10.024 disponíveis, somando-se 2.000 arquivos e medindo-se o tempo de processamento e o volume de dados processados em cada rodada. A TABELA 6.3 apresenta a descrição detalhada do processo de extração automático realizado.

TABELA 6.3 - Resultados do Experimento B

		N0	P1			N1	Redução N1/N0	P2			N2	Redução N2/N1	P3			N3	Redução N3/N2
			Tempo	Segundos	Médias			Tempo	Segundos	Médias			Tempo	Segundos	Médias		
Arqs.	(Qt.)	**10024**	0 : 9 : 9	549	18,2587	**9974**	0,50%	1 : 38 : 57	5937	1,6800	**367**	96,32%	1 : 0 : 22	3622	0,1013	**1**	99,73%
	(Mb)	**13,8979**			0,0253	**13,4583**	3,16%			0,0023	**0,4777**	96,45%			0,0001	**0,0864**	81,92%
Arqs.	(Qt.)	**8024**	0 : 1 : 35	95	84,4632	**7974**	0,62%	1 : 0 : 10	3610	2,2089	**312**	96,09%	0 : 4 : 27	267	1,1685	**1**	99,68%
	(Mb)	**11,1113**			0,1170	**10,5460**	5,09%			0,0029	**0,3929**	96,27%			0,0015	**0,0707**	82,02%
Arqs.	(Qt.)	**6024**	0 : 0 : 47	47	128,1702	**5974**	0,83%	0 : 39 : 54	2394	2,4954	**232**	96,12%	0 : 1 : 58	118	1,9661	**1**	99,57%
	(Mb)	**8,5402**			0,1817	**7,9936**	6,40%			0,0033	**0,2896**	96,38%			0,0025	**0,0577**	80,09%
Arqs.	(Qt.)	**4024**	0 : 0 : 32	32	125,7500	**3974**	1,24%	0 : 24 : 17	1457	2,7275	**153**	96,15%	0 : 1 : 10	70	2,1857	**1**	99,35%
	(Mb)	**5,7726**			0,1804	**5,2457**	9,13%			0,0036	**0,1934**	96,31%			0,0028	**0,0370**	80,87%
Arqs.	(Qt.)	**2024**	0 : 0 : 16	16	126,5000	**1974**	2,47%	0 : 11 : 51	711	2,7764	**84**	95,74%	0 : 0 : 35	35	2,4000	**1**	98,81%
	(Mb)	**2,8353**			0,1772	**2,6922**	5,05%			0,0038	**0,1037**	96,15%			0,0030	**0,0212**	79,54%
Médias	(Qt.)				96,6284		1,13%			2,3776		96,08%			1,5643		99,43%
	(Mb)				0,1363		5,76%			0,0032		96,31%			0,0020		80,89%
Desvio Padrão	(Qt.)				47,5002		0,0080			0,4501		0,0021			0,9411		0,0037
	(Mb)				0,0678		0,0221			0,0006		0,0011			0,0012		0,0109

O processo P1 no experimento *B* eliminou os mesmos 50 arquivos não selecionados pelo P1 no experimento *A*, pois a base de dados de teste contem os 300 arquivos usados no experimento anterior, acrescida de outros arquivos ETOs no formato texto, os quais são selecionados para N1. Portanto, a redução de tamanho de N0 para N1 foi pequena: aproximadamente 1,13% dos arquivos foram descartados (Figura FIGURA 6.2). Caso houvesse em N0 outros arquivos com estruturas não definidas na BC, reduções maiores teriam ocorrido. P2, por outro lado, gerou uma redução média de 96%, tanto em volume quanto em quantidade de arquivos, da BDI N1 para a N2. Isso demonstra que muitos dos documentos existentes em N1 não pertencem aos domínios de interesse: celular ou computação. A redução em quantidade dos documentos em N2

para o arquivo resultante em N3 não pode ser considerada em quantidade, pois N3 sempre resulta em um arquivo, independentemente do tamanho de N2. Contudo, o tamanho desse arquivo varia conforme o volume de informações em N2, pois mais dados serão extraídos e armazenados em N3. Obteve-se uma redução média de 80,89% do volume (Mb) de informação existentes em N2 para N3, extraindo dos documentos em N2 somente as informações identificadas como relevantes. O volume de informações a ser tratado pelo usuário do sistema em N2 é cerca de 30 vezes menor que o original (N0) e em N3 160 vezes.

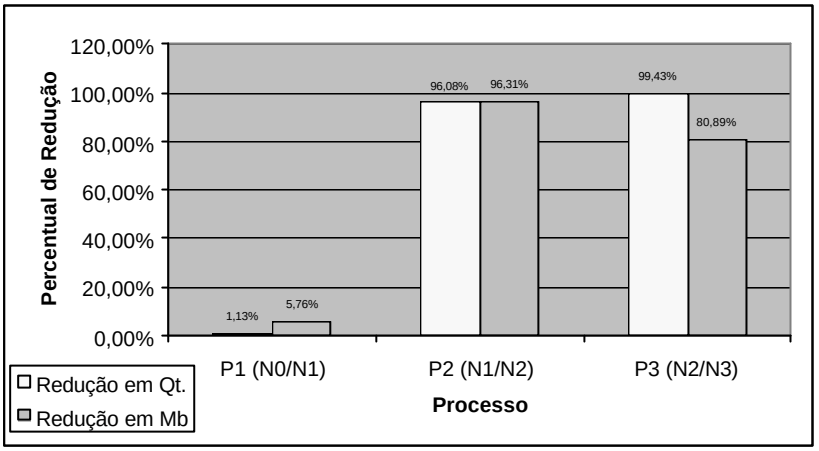

FIGURA 6.2 - Tamanho das Bases de Dados Conforme por Processo

O custo de processamento foi distribuído entre os processos na proporção aproximadamente inversa entre sua complexidade e o volume de dados tratado. Comparando-se os processos P2 e P3, esse último, apresentou tempos de processamento por documento até 10 vezes maiores que o primeiro. Contudo, P3 tratou um volume muito menor de dados.

Em teoria, a velocidade individual dos processos deveria ser constante, independentemente do volume de dados (Mb) de entrada, pois não existe relação direta entre o processamento de um documento e outro em nenhum dos processos. Contudo, observa-se uma perda de velocidade em todos os processos conforme o aumento do volume de dados tratados. Essa queda de performance deve-se à implementação do SE-MNBC e a capacidade do computador utilizado (pouca memória RAM). O sistema utiliza tabelas em memória com dados sobre os documentos sendo tratados, as quais tem volume considerável para quantidades maiores de documentos, exigindo *swap* de memória e ocasionando a perda de performance.

A velocidade de extração de P1 foi muito boa: em média 96,63 arquivos por segundo, conforme a FIGURA 6.3 e a TABELA 6.3. Contudo, a velocidade cai bastante quando a quantidade de 6.024 arquivos é ultrapassada, visto os motivos anteriormente destacados. Até essa quantidade, a velocidade de extração era praticamente constante. A análise é realizada sobre a *quantidade* (*Qt*.) de arquivos e não sobre o volume em Mb desses, pois P1 extrai as características externas dos mesmos, não sofrendo influência do tamanho dos arquivos. Os processos P2 e P3 também apresentaram uma curva decrescente de velocidade conforme o aumento do volume (Mb) de dados tratados pelo mesmo motivo (FIGURA 6.4 e FIGURA 6.5). Essa característica foi mais acentuada em P3, visto a maior complexidade desse processo. No entanto, suas velocidades de processamento foram em média boas frente à complexidade de processamento, viabilizando o uso do sistema no tratamento de grandes volumes de BDNE/SE.

Na prática, essa perda de performance não impacta sobre a aplicação do sistema, pois os usuários normalmente processam semanalmente os ETOs a fim de não perderem as novas oportunidades disponibilizadas. O volume semanal médio é de 3.500 ETOs. De qualquer forma, melhorias na implementação do sistema, visando otimizar o uso da memória, podem aumentar a velocidade de processamento para grandes volumes de BDNE/SE.

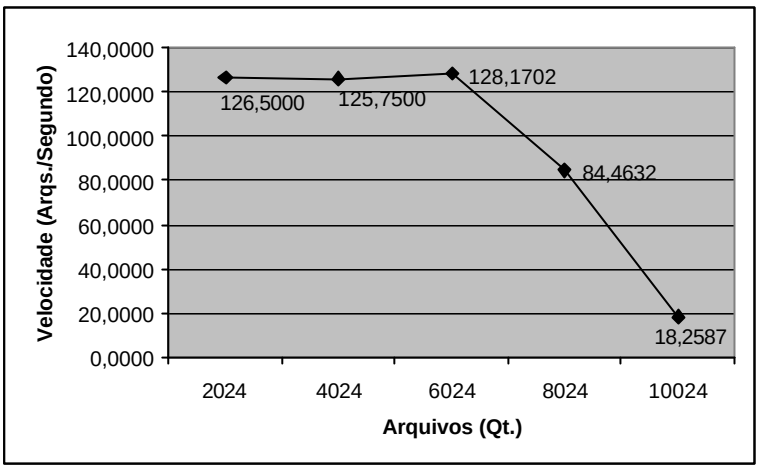

FIGURA 6.3 - Velocidade de Extração de P1

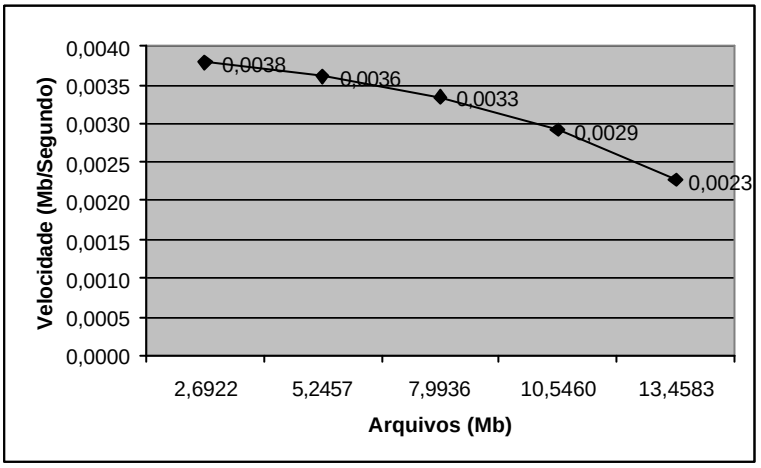

FIGURA 6.4 - Velocidade de Extração de P2

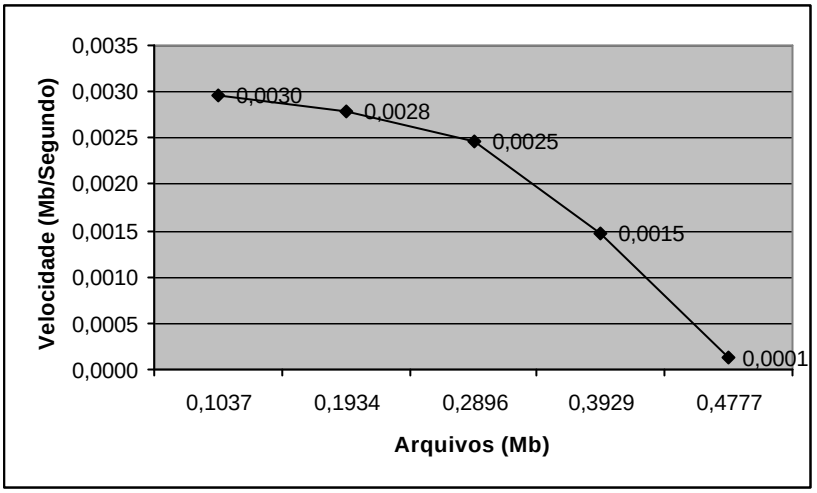

FIGURA 6.5 - Velocidade de Extração de P3

7 Conclusão

Este trabalho apresentou os conceitos da metodologia EI-MNBC, os quais foram implementados no sistema de extração SE-MNBC e aplicados no tratamento das informações não estruturadas de comércio eletrônico existentes no sistema ETO. O processo da extração desenvolvido é auxiliado pelo uso de uma BC que especifica seu funcionamento conforme o ambiente de extração e as necessidades do usuário. Técnicas de RD, EI e MD são combinadas de acordo com as informações requeridas e seu nível de detalhamento. A EI-MNBC agrega qualidade ao processo de extração, permitindo ao usuário encontrar informações relevantes em múltiplos níveis conceituais, de forma simples e direcionada às suas necessidades através de BDI. O sistema implementado é muito útil em análises comerciais de mercado, inteligência competitiva e assim por diante [CEI 2001].

As vantagens da EI-MNBC dependem da qualidade da BC. Uma BC bem definida é obtida se existirem especialistas no ambiente de extração, dedicação e dicionários predefinidos. Refinamentos são importantes e devem ser feitos por pessoas habilitadas a analisar erros de extração. O nível de qualidade depende de quanto esforço e recursos o usuário quer ou tem para investir na tarefa de engenharia de conhecimento. A descrição do conhecimento evolui para um modelo mais exato conforme os usuários se tornam mais familiarizados com a linguagem usada nos documentos. A criação da BC é facilitada pela execução e avaliação parcial das etapas de extração, que podem ser configuradas separadamente através dos diversos níveis conceituais de extração, distribuindo o custo de configuração. Além disso, as BC são independentes do sistema e podem ser reutilizadas, algumas vezes com pequenos ajustes, diminuindo ainda mais o custo de configuração do sistema quando trata de processos de extração semelhantes.

O uso de BDI permitiu a avaliação dos resultados de cada etapa de extração e da eficiência de sua configuração. Observou-se nos experimentos que ruídos são mais facilmente identificados em cada etapa da extração, permitindo seu ajuste e evitando sua propagação, o que amplia a precisão de extração. Por outro lado, a perda de informação relevante também é reduzida através de refinamentos das regras associadas com cada processo intermediário, generalizando-as e recuperando a informação perdida. Assim, a metodologia EI-MNBC possibilita uma elevada qualidade de extração com a avaliação parcial desse processo e refinamentos da BC de acordo com os resultados parciais (realimentação).

O SE-MNBC extraiu a informação relevante gradualmente, com o uso de processos independentes e integrados de extração, os quais agiram sobre características diferentes dos documentos, combinando diferentes técnicas de RD, IE e MD. O custo de processamento foi distribuído entre as etapas de extração, permitindo rejeitar a informação irrelevante e extrair e estruturar a relevante.

O uso de técnicas de análise superficial no processo *P3* revelou-se eficiente. Entretanto, o uso dessa abordagem somente foi possível devido aos processos anteriores (*P1* e *P2*), que definem previamente a estrutura e o domínio dos documentos sobre os quais as regras de extração de P3 serão executadas.

A metodologia apresenta portabilidade quanto a sua aplicação. Ela é independente da estrutura de armazenamento dos documentos ou do domínio de extração. É possível criar muitas BC, definindo múltiplos domínios, de acordo com as

necessidades do usuário, sem restringir-se à extração de informações de um determinado assunto, língua ou tipo de dado. Essa flexibilidade reduz o esforço para obter-se bons resultados de extração, identificando e recuperando informações dos documentos, conforme suas características, e estruturando-as. A adoção de técnicas de SBC em EI satisfaz diferentes necessidades de extração.

O desenvolvimento do SE-MNBC foi facilitado pela organização modular definida pela metodologia EI-MNBC, bem como a documentação e posteriores manutenções. Além disso, módulos de outros sistemas e de versões anteriores puderam ser reutilizados.

Embora os bons resultados obtidos com o sistema SE-MNBC, algumas dificuldades inerentes à manipulação de BDNE/SE podem aparecer. A informação desejada pode estar incompleta, por exemplo, falsas indicações de dados relevantes. No caso de um ETO pode ser um número de telefone esperado, mas inexistente, após a palavra "*phone*". Informação imprecisa ou incompleta também ocorre, por exemplo, datas sem o dia ou somente referenciando o ano. Outro problema prejudicial ao processo de extração é os erros ortográficos, resultando em ruído. A metodologia não atinge 100% de *recall* e *precision*, cabendo ao usuário ler os documentos identificados como relevantes pelo sistema na busca de detalhes não extraídos. Ele deve decidir se o documento merece ou não uma leitura aprofundada conforme as informações extraídas.

Analisando o presente trabalho e seus resultados, confirma-se sua capacidade de auxiliar os usuários no tratamento e na manipulação de informações de comércio eletrônico a partir de BDNE/SE. A aplicação da EI-MNBC simplificou o uso do sistema de extração através das técnicas de SBC e de BDI. Usuários leigos, utilizando BC já existentes e processos de extração independentes e adaptados as suas necessidades, são capazes de extrair e manipular BDNE/SE sem ter grandes conhecimentos de computação. O SE-MNBC tratou grandes volumes de dados, diminuindo o tempo de absorção da informação não estruturada pelo usuário através de sua estruturação, distribuição do custo de processamento na proporção inversa entre seu volume de dados e a complexidade de processamento, e com a alta qualidade de extração obtida pela flexibilidade de adaptação do sistema às necessidades do usuário. O uso das BDI simplificou a análise de complexos conjuntos de informações, disponibilizando dados com diferentes graus de especialização (níveis conceituais).

É possível obter bons resultados para o crescimento da posição competitiva das PME, permitindo seu acesso ao relevante conteúdo distribuído nos ETOs. Se o processo de extração ocorrer conforme as considerações apontadas aqui, qualquer empresa, mesmo em países sub-desenvolvidos, pode obter vantagens do sistema de ETOs, encontrando novas oportunidades, mercados, parceiros, processos e métodos, ou mesmo idéias para novos produtos. Isso permite à empresa evoluir seu negócio a fim de obter vantagens frente ao seu competidor, mantendo-se no mercado e talvez ultrapassando outras (aparentemente) mais capazes e experientes.

7.1 Trabalhos Futuros

Esta tese fundamentou-se em um conjunto de trabalhos desenvolvidos: duas dissertações de mestrado (Scarinci [SCA 97a] e Zambenedetti [ZAM 2001]); três trabalhos de conclusão de graduação (Becker [BEC 97], Trein [TRE 2000] e Halmenschlager [HAL 2000]); cinco artigos (Scarinci [SCA 97], [SCA 2001], [SCA 2002], [SCA 2002a] e Becker [BEC 97a]); e duas monografias (Scarinci [SCA 95], [SCA 2000]). Contudo, este trabalho não se constitui em uma linha de chegada, mas

sim, em um ponto de partida do qual muitos outros trabalhos ou estudos poderão ser desenvolvidos.

Um importante parâmetro utilizado na avaliação qualitativa de um trabalho de pesquisa, tendo como referência à motivação de seu desenvolvimento, são as perspectivas e idéias por ele inspiradas. Tendo em vista o objetivo do presente trabalho e a possibilidade de aprofundar os estudos realizados, ampliando os resultados até agora obtidos, cria-se um estímulo a criatividade deste e de outros pesquisadores, ensejando-lhes a concepção de projetos que podem ser propostos e desenvolvidos futuramente. Destacam-se as seguintes propostas de pesquisa, as quais visam complementar ou aproveitar os resultados obtidos e os conhecimentos já adquiridos:

Integração de um módulo de MD: as informações extraídas até o processo *P3* poderiam ser analisadas por um módulo de MD integrado a atual arquitetura do SE-MNBC. Isso permitiria a descoberta de conhecimentos implícitos nas informações extraídas.

Melhorias do SE-MNBC: novas versões do sistema poderiam otimizar o uso da memória, aumentando sua performance sobre grandes volumes de dados.

Novos testes: gerar novas avaliações do SE-MNBC em outros ambientes de aplicação, bem como realizar o desenvolvimento de novas versões, visando a correção de falhas e a implementação de melhorias.

Atualização automática da BC: estudar técnicas e metodologias de aquisição automática do conhecimento, tornando possível a construção, pelo próprio sistema, de novas regras de extração, melhorando os resultados obtidos e facilitando sua configuração.

ANEXO 1 Conferência de Entendimento de Mensagens

A.1 Message Understanding Conference - MUC

A Conferência de Entendimento de Mensagens foi criada pelo NOSC (*Naval Ocean Systems Center*) para avaliar e fomentar pesquisas em análise automática de dados textuais, inicialmente contidos em mensagens militares. Embora chamada de "conferência", a característica que distingue os MUCs não são as conferências propriamente ditas, mas sim as avaliações às quais os sistemas dos participantes são submetidos [GRI 95].

A avaliação de performance usando métricas claras tem importante destaque dentro da área de EI e, conseqüentemente, no MUC. No entanto, o aspecto de maior destaque e exclusivo deste congresso são as avaliações conduzidas como uma conferência aberta [LEH 94]. Qualquer laboratório de pesquisa no mundo pode participar de um MUC, e os resultados das avaliações dos sistemas são abertamente publicados. Esses diferenciais endossaram os testes realizados com credibilidade e influência, dissociando o MUC de avaliações restritas de performance, realizadas por seletos grupos de organizadores [LEH 94]. Nenhum grupo de pesquisa, acadêmico ou comercial, é excluído das avaliações, e a participação na conferência atribui um grau de legitimidade aos pesquisadores que sujeitam seus sistemas e projetos de pesquisa ao exame público.

Outra contribuição importante do MUC para PLN e EI é o fato de uma grande variedade de grupos de pesquisa examinarem a mesma tarefa em profundidade, durante as avaliações, para, posteriormente, se reunirem e discutirem seus avanços e falhas [INT 2001]. Isso aproxima e facilita a comunicação entre os grupos de pesquisa. Diferenças teóricas, experiências diversas e paradigmas distintos são distâncias difíceis de se cruzar, mas excelentes resultados podem ser obtidos quando todos estão tentando construir processos para a mesma tarefa.

Os MUCs têm auxiliado a definir um programa de pesquisa e desenvolvimento em EI. A DARPA (*Defense Advanced Research Projects Agency*), através dessas conferências, tem gerado um grande volume de informações científicas e programas de tecnologia, os quais são fortemente guiados pelas avaliações regulares dos sistemas. Os MUCs são notáveis quanto ao seu papel de incentivo à pesquisa. Eles formatam substancialmente o programa de pesquisa em EI e trazem essa área para o corrente estado. Analisando a evolução dos MUCs e dos sistemas participantes, podemos analisar a evolução da área de EI e conhecer o "estado-da-arte" nessa tecnologia.

A.1.1 MUC-1 e MUC-2

EI tem figurado de forma promissora no campo de PLN empírico. As primeiras tentativas de avaliação comparativa da performance dos sistemas de extração de informações ocorreram no MUC-1 e no MUC-2, respectivamente em 1987 e 1989. Embora essas primeiras conferências tenham sido menos ambiciosas em seu escopo que o MUC-3, elas serviram de base para o sucesso das conferências subseqüentes. Os documentos analisados foram providos pelo *Foreign Broadcast Information Service*, e as conferências foram patrocinadas pelo escritório de Tecnologia de Informação da DARPA [LEH 94].

O MUC-1 foi fundamentalmente exploratório, cada grupo participante projetou seu próprio formato de arquivo de saída para armazenar as informações extraídas, e não

existiam avaliações formais [GRI 95]. No MUC-2, consolidou-se a tarefa de extração como preenchimento de *templates*. Os participantes recebiam a descrição de uma classe de eventos a ser identificada nos documentos, e para cada evento devia-se preencher o *template* com informações sobre ele [GRI 95]. O *template* tinha 10 *slots* para informações, tais como: tipo de evento, agente, momento (tempo), local e efeito, etc. Além disso, o MUC-2 definiu as primeiras medidas de avaliação: *recall* e *precision* [GRI 95]. Ambos, MUC-1 e MUC-2, envolveram extração de informações a partir de mensagens sobre alvos e operações militares [INT 2001].

A.1.2 MUC-3

Em 1991, a DARPA expôs uma ambiciosa avaliação das tecnologias em EI através do MUC-3. Essa foi a primeira tentativa de avaliação de porte da performance de sistemas de extração usando documentos desconhecidos e rigorosos procedimentos de avaliação [MUC 96]. Quinze laboratórios de pesquisa participaram deste congresso. Coordenar a avaliação de complicados sistemas de extração de todos esses laboratórios foi difícil, mas o MUC-3 foi conduzido com considerável planejamento [LEH 94]. A descrição dos sistemas avaliados e seus resultados foram publicados.

Disponibilizou-se 1300 documentos providos pelo *Foreign Broadcast Information Service* descrevendo ações terroristas na América Latina para uso no desenvolvimento dos sistemas [INT 2001]. Cada documento foi agrupado com o *template* manualmente criado para extrair as informações relevantes. A estrutura do *template*, mais complexa que a do MUC anterior, tinha 18 *slots*, conforme o exemplo da Figura A.1 [GRI 95]. Os *slots* não aplicáveis a um tipo de incidente eram marcados com "*". Além disso, podiam existir respostas "corretas" alternativas para o mesmo *slot*.

TST1-MUC3-0080

BOGOTA, 3 APR 90 (INTRAVISION TELEVISION CADENA 1) – [REPORT] [JORGE ALONSO SIERRA VALENCIA] [TEXT] LIBERAL SENATOR FEDERICO ESTRADA VELEZ WAS KIDNAPPED ON 3 APRIL AT THE CORNER OF 60TH AND 48TH STREETS IN WESTERN MEDELIN, ONLY 100 METERS FROM A METROPOLITAN POLICE CAI [IMMEDIATE ATTENTION CENTER]. THE ANTIOQUIA DEPARTMENT LIBERAL PARTY LEADER HAD LEFT HIS HOUSE WITHOUT ANY BODYGUARDS ONLY MINUTES EARLIER. AS HE WAITED FOR THE TRAFFIC LIGHT TO CHANGE, THREE HEAVILLY ARMEND MEN FORCED HIM TO GET OUT OF HIS CAR AND INTO A BLUE RENAULT.

HOURS LATER, THROUGH ANONYMOUS TELEPHONE CALLS TO THE METROPOLITAN POLICE AND TO THE MEDIA, THE EXTRADITABLES CLAIMED RESPONSIBILITY FOR THE KIDNAPING. IN THE CALLS, THEY ANNOUNCED THAT THEY WOULD RELEASE THE SENATOR WITH A NEW MESSAGE FOR THE NACIONAL GOVERNMENT.

LAST WEEK, FEDERICO ESTRADA VELEZ HAD REJECTED TALKS BETWEEN THE GOVERNMENT AND THE DRUG TRAFFICKERS.

00. MESSAGE ID	TST1-MUC3-0080
01. *TEMPLATE* ID	1
02. DATE OF INCIDENT	03 APR 90
03. TYPE OF INCIDENT	KIDNAPPING
04. CATEGORY OF INDENT	TERRORIST ACT
05. PERPETRATOR: ID OF INDIV (S)	"THREE HEAVILY ARMED MEN"
06. PERPETRATOR: ID OF ORG (S)	"THE EXTRADITABLES"
07. PERPETRATOR: CONFIDENCE	CLAEMED OR ADMITTED: "THE EXTRADITABLES"
08. PHYSICAL TARGET: ID (S)	*
09. PHYSICAL TARGET: TOTAL NUM	*
10. PHYSICAL TARGET: TYPE (S)	*
11. HUMAN TARGET: ID (S)	"FEDERICO ESTRADA VELEZ" ("LIBERAL SENATOR")
12. HUMAN TARGET: TOTAL NUM	1
13. HUMAN TARGET: TYPE (S)	GOVERNMENT OFFICIAL: "FEDERICO ESTRADA VELEZ"
14. TARGET: FOREIGN NATION (S)	-
15. INSTRUMENT: TYPE (S)	*
16. LOCATION OF INCIDENT	COLOMBIA: MEDELLIN (CITY)
17. EFFECT ON PHYSICAL TARGET (S)	*
18. EFFECT ON HUMAN TARGET (S)	-

FIGURA A.1 - Mensagem e Template Associado Usado no MUC-3

O desafio dos participantes do MUC-3 foi criar um sistema que gerasse as instâncias de *templates* automaticamente, sem assistência humana. Cada sistema foi avaliado com o uso de 100 documentos [LEH 94]. Nenhuma alteração nos sistemas foi permitida após a entrega dos documentos de avaliação. Um programa de classificação avaliou a performance total dos sistemas, analisando cada *template* resposta frente aos *templates* chave (modelos manuais). A informação devia estar apropriadamente armazenada no *template* e nos *slots* de resposta corretos para ser considerada certa. Utilizou-se medidas de *recall* e *precision* na avaliação.

A.1.3 MUC-4

O MUC-4, realizado em 1992, foi muito similar ao MUC-3. Dezessete organizações participaram, acadêmicas e comerciais, incluindo doze já participantes do MUC-3 e cinco novas universidades [LEH 94]. A DARPA proveu 1500 documentos com respostas para uso no desenvolvimento dos sistemas participantes [RIL 94]. As respostas eram instâncias de *templates* manualmente gerados com a informação a ser extraída. Também foram disponibilizados dois conjuntos de 100 novos documentos e respostas para a avaliação final. Os documentos sobre terrorismo na América Latina, usados no MUC-3, foram novamente aplicados no MUC-4. Do MUC-1 ao MUC-4 todos os documentos eram com caracteres em caixa alta [GRI 95].

Cada participante desenvolveu um sistema para extrair informações sobre terrorismo na América Latina a partir de artigos de jornal. Um extenso conjunto de regras definia o domínio "terrorismo". Em geral, um documento era relevante se mencionava um incidente terrorista ocorrido em um país da América Latina [RIL 94a]. Descrições gerais de eventos terroristas (por exemplo, "têm existido muitas bombas..."), eventos acontecidos a mais de dois meses antes da data do jornal onde o artigo encontrava-se, e eventos terroristas envolvendo militares não eram considerados relevantes. O uso do mesmo domínio do MUC-3 permitiu aos participantes veteranos corrigirem problemas enfrentados na conferência anterior.

Os sistemas geravam um ou mais *templates* para cada documento, os quais tinham, por exemplo, *slots* para nomes de grupos terroristas, vítimas, alvos físico, armamentos, datas, locais, etc [LEH 94]. Contudo, apesar da semelhança com o MUC-3, ocorreu um pequeno crescimento da complexidade do *template* nesta conferência: usou-se 24 *slots*. Para cada incidente terrorista relevante, o sistema instanciava um *template* com informações sobre ele. Cada *template* instanciado continha informação sobre um único incidente terrorista. Se um documento descrevesse múltiplos eventos relevantes, então o sistema gerava múltiplos *templates*. Eventos terroristas irrelevantes resultavam em *templates* sem informações.

No MUC-4, escolheu-se uma métrica de avaliação dos sistemas diferente, visto que essa nova medida é mais sensível a *templates* inválidos do que a usada no MUC-3 (*recall*/*precision*): F-*score* [LEH 94].

A.1.4 MUC-5

O MUC-5, realizado em 1993, fez parte do TIPSTER (programa do governo americano de pesquisa e desenvolvimento em RD e EI), tendo a participação de dezessete centros de pesquisa [LEH 94a]. Essa edição do congresso representou um substancial amadurecimento e salto na complexidade de extração. Usaram-se dois tipos de documentos, um sobre *joint ventures* internacionais (empreendimentos conjuntos) e

outro sobre fabricação de circuitos eletrônicos [INT 2001]. Existiam documentos em Inglês e Japonês. Em lugar de um único *template*, usou-se para os documentos de *joint venture* uma estrutura com 11 tipos de objetos diferentes que formavam os *templates* [GRI 95]. Cada objeto aplicava-se a um determinado tipo de informação, e juntos somavam 47 *slots*, quase o dobro de *slots* usados no MUC-4. A definição dos domínios de extração, com as informações a serem extraídas, também dobrou de tamanho, tendo mais de 40 páginas. Outra inovação foi o uso de estruturas *nexted* entre os objetos. Nos MUCs anteriores, cada evento era representado em um único *template*, gerando um único registro na base de dados, com um grande número de atributos [GRI 95]. Esse formato mostrou-se inadequado quando o evento tinha vários participantes, por exemplo, várias vítimas de um ataque terrorista. Além disso, alguns pesquisadores queriam gravar um conjunto de fatos sobre cada participante. Esse tipo de informação podia ser facilmente gravado na estrutura hierárquica usada no MUC-5. Nela existe um único objeto "pai" para um evento, o qual aponta para uma lista de objetos "filhos", um para cada participante do evento.

A.1.5 MUC-6

A sexta edição do MUC foi realizada em novembro de 1995 em Columbia, Maryland. As conferências anteriores eram focadas em uma única tarefa de extração de informações: analisar documentos livremente, identificando eventos de tipos específicos e preenchendo *templates* com informações extraídas de cada evento. Durante o curso dos cinco MUCs anteriores, as tarefas e os *templates* ficaram cada vez mais complicados. Em vista disso, a DARPA reuniu os participantes do TIPSTER e representantes do governo em dezembro de 1993 para definir os objetivos e as tarefas do MUC-6. Essa reunião, dirigida por Ralph Grishman, definiu os objetivos das conferências posteriores, a fim de tornar os sistemas de extração mais portáveis para novos domínios e encorajar trabalhos básicos em EI, através da avaliação de tecnologias bem focadas [MUC 96]. Entre os objetivos identificados estão os seguintes [GRI 95]: (I) demonstrar tecnologias de EI independentes de domínio e de aplicação ampla e imediata; (II) encorajar pesquisas para tornar os sistemas de extração mais portáveis; e (III) incentivar trabalhos de "compreensão profunda" da linguagem natural. A definição desses objetivos era, em parte, uma reação ao direcionamento dos MUCs anteriores. As tarefas do MUC-5, em particular, eram muito complexas e um grande esforço havia sido investido pelo governo na preparação dos dados para o desenvolvimento e teste dos sistemas, e pelos participantes em adaptar esses sistemas para as tarefas [GRI 95].

Para atingir os objetivos, formulou-se quatro tipos de tarefas para o MUC-6 [MUC 96]. Os participantes podiam escolher um subconjunto delas. As tarefas especificadas foram posteriormente detalhadas em 1994 e no começo de 1995, através de exemplos aplicados e extensivas discussões por e-mail. Em abril de 1995, após esse detalhamento, realizou-se um conjunto de avaliações chamado "*dry run*". A avaliação formal do aconteceu em setembro de 1995, meses antes da conferência, em novembro do mesmo ano [MUC 96]. O domínio especificado foi o seguinte: alterações no corpo executivo de corporações [GRI 95]. As tarefas do MUC-6, ordenadas da mais simples para a mais difícil, estão abaixo apresentadas:

(I) Reconhecimento de entidades (*Named Entity* - NE): reconhecer nomes de entidades: pessoas, organizações, locais, expressões temporais, expressões numéricas, etc [INT 2001]. Os programas usados na extração de NEs são normalmente independentes de domínio e de uso prático em EI e PLN [GRI 95]. O resultado da tarefa

era a inserção de marcas SGML (*Standard Generalized Markup Language*) nos documentos, identificando as entidades. SGML é uma meta-linguagem utilizada para descrever outras linguagens, baseando-se na teoria de que qualquer documento pode ser dividido em três partes: dados, estrutura e formato [LIL 96]. A expressão (marca) ENAMEX (*ENtity NAme EXpression*) é usada tanto para nomes de organizações como de pessoas; a expressão NUMEX (*NUMeric EXpression*) é usada para moedas e percentagens, conforme exemplo [GRI 95]: "O <ENAMEX TYPE="LOCATION">Brasil</ENAMEX> lançou um satélite para a transmissão de sinais de televisão, ampliando a capacidade de recepção de canais internacionais em <NUMEX TYPE="PERCENT">15 por cento</NUMEX> no decorrer <TIMEX TYPE="DATE">do próximo ano</TIMEX>."

(II) Coreferência (*Coreference* – CO): associar informações relacionadas. Esta tarefa usa tecnologias de EI facilmente independentes de domínio, e serve de ligação entre as tarefas de NE e de TE (a seguir descrita) [GRI 95]. Na avaliação do MUC, somente coreferências de identidade foram marcadas nos documentos, utilizando a SGML [LIL 96]. O exemplo a seguir ilustra coreferência de identidade entre "*its*" e "*The U.K. satellite television broadcaster*", bem como entre a função "*its subscriber base*" e o valor '*5.35 million.*": **The U.K. satellite television broadcaster* said **its* subscriber base* grew 17.5 percent during the past year to *5.35 million**. Existe somente um identificador SGML por *string*. Outras ligações são inferidas a partir de ligações explicitas. Assume-se que a relação de coreferência é simétrica e transitiva; então, se em uma frase A é marcada como referenciada por B (indicada por um ponteiro REF de B para A), inferi-se que B é referenciada por A. Se A é referenciada por B, B é referenciada por C, inferi-se que A é referenciada por C.

(III) Elementos de *template* (*Template Elements* – TE): para atingir o segundo objetivo, a tarefa de extração do MUC-6 deveria usar *templates* relativamente simples, mais parecidos com os do MUC-2 do que com os do MUC-5 [GRI 95]. Contudo, em harmonia com a estrutura de objetos hierárquicos do MUC-5, foi previsto que o *template* teria um objeto "pai" (evento) apontando para outros objetos "filhos": participantes do evento (pessoas, organizações, produtos, etc.) [INT 2001]. Por outro lado, para aumentar a portabilidade dos sistemas, padronizou-se os objetos "filhos", visto estarem envolvidos em uma grande variedade de ações. Desta forma, os participantes do MUC desenvolveram funções de extração somente para os objetos "filho" e usaram-nas em diferentes eventos. Esses objetos "filhotes" eram chamados de *Template Elements* e o seu preenchimento era o foco desta tarefa, apelidada de "mini-MUC", visto a maior simplicidade dos *templates* resultantes.

(IV) *Templates* de Cenário (*Scenario Template* – ST): representa o processo de extração de informações tradicional, resultando em *templates* complexos que ligam os *templates* elementares, gerados pela tarefa de TE [DAL 2000]. A avaliação da tarefa de ST é realizada em um determinado domínio [MUC 96]. Isto reduz o tempo desperdiçado pelos participantes tornando os sistemas especialistas em um determinado domínio; além de encorajar o desenvolvimento de ferramentas para porta-los para novos domínios. O domínio da avaliação do MUC-6 foi revelado somente semanas antes dela ocorrer. O objetivo era dotar os sistemas de mecanismos de extração para a compreensão mais profunda de documentos, pois a maioria dos participantes dos MUCs estava trabalhando com técnicas de extração relativamente superficiais, baseadas principalmente em busca de padrões [MUC 96]. Sem conhecer previamente o domínio de extração, os sistemas não podem se valer totalmente das características superficiais

de um determinado domínio e devem apresentar funções de extração mais desenvolvidas quanto às análises léxica, sintática e semântica.

A.1.6 MUC-7

O MUC-7, realizado em 1997, seguiu as mesmas diretrizes da conferência anterior. Contudo, destacou pela diversidade de idiomas no processo de extração. Juntamente com o MUC-7, foi realizado o *Second Multilingual Entity Task* (MET-2) [INT 2001]. A avaliação da tarefa de NE usou documentos de desenvolvimento e avaliação de diferentes línguas [CHI 97]. O domínio de desenvolvimento foi colisões aeronáuticas e o de avaliação eventos de lançamento. Essa mudança de domínio causou efeitos similares em todas as línguas. Os participantes ficaram desapontados com os seus escores na avaliação quando comparados com os escores da etapa de desenvolvimento. No entanto, os escores finais ainda estavam acima do limite operacional de 80% (F-*score*), sem quaisquer alterações nos sistemas para o novo domínio [CHI 97].

No MUC-7, estavam presentes mais participantes internacionais do que nas conferências anteriores. O principal interesse dos participantes eram os palestrantes não nativos na língua de seus sistemas, ou por sistemas desenvolvidos em países de língua diferente. Nessa edição do MUC, incluiu-se a tarefa de *Template Relations* (TR), que identifica a relação entre elementos de *template*, fazendo a ligação entre as tarefas de TE e ST [CHI 97]. Contudo, limitou-se TR às relações em organizações: funcionário_de, produto_de, localização_de, mesmo sendo essa tarefa facilmente expansível para todas as combinações lógicas e relações entre tipos de entidades. Um exemplo de TR é apresentado a seguir:

```
<EMPLOYEE_OF-9602040136-5> :=
PERSON:              <ENTITY-9602040136-11>
ORGANIZATION:   <ENTITY-9602040136-1>

<ENTITY-9602040136-11> :=
ENT_NAME:                  "Dennis Gillespie"
ENT_TYPE:                  PERSON
ENT_DESCRIPTOR:            "Capt."  / "the commander of Carrier Air Wing 11"
ENT_CATEGORY:  PER_MIL

<ENTITY-9602040136-1> :=
ENT_NAME:                  "NAVY"
ENT_TYPE:                  ORGANIZATION
ENT_CATEGORY:  ORG_GOVT
```

A.1.2 Resultados

TABELA A.1 - Tarefas Avaliadas do MUC-3 ao MUC-7

Avaliação/ Tarefa	Named Entity (NE)	Coreference (CO)	*Template* Element (TE)	*Template* Relation (TR)	Scenario *Template* (ST)	Multilingual
MUC-3					SIM	
MUC-4					SIM	
MUC-5					SIM	SIM
MUC-6	SIM	SIM	SIM		SIM	
MUC-7	SIM	SIM	SIM	SIM	SIM	
MET-1	SIM					SIM
MET-2	SIM					SIM

TABELA A.2 - Resultados Máximos Relatados do MUC-3 até o MUC-7 por Tarefa

Avaliação/ Tarefa	**Named Entity (NE)**	**Coreference (CO)**	***Template* Element (TE)**	***Template* Relation (TR)**	**Scenario *Template* (ST)**	**Multilingual**
MUC-3					R < 50% P < 70%	
MUC-4					F < 56%	
MUC-5					I JV F < 53% I ME F < 50%	J JV F < 64% J ME F < 57%
MUC-6	F < 97%	R < 63% P < 72%	F < 80%		F < 57%	
MUC-7	F < 94%	F < 62%	F < 87%	F < 76%	F < 51%	
Multilingual						
MET-1	C F < 85% J F < 93% E F < 94%					
MET-2	C F < 91% J F < 87%					

Legenda: Língua: **I** = Inglês **C** = Chinês **J** = Japonês **E** = Espanhol
Assunto: **JV** = Joint Venture **ME** = Micro Eletrônica
Métrica: **R** = *Recall* **P** = *Precision* **F** = F-s*core* (*Recall* e *Precision* com pesos iguais)

ANEXO 2 Sistemas Relacionados

A.2 Sistemas Relacionados

O estudo dos sistemas de extração de informações que mais se destacaram no MUC permite a análise detalhada das técnicas utilizadas, da situação atual da pesquisa e seus resultados práticos. Apesar de todos os sistemas a seguir analisados terem participado do MUC e apresentarem semelhanças de arquitetura e técnicas empregadas, conforme destacado por [HOB 2001], seus objetivos de extração variam quanto à precisão, velocidade de processamento e volume de dados a ser tratado. Neste sentido, este estudo dá acesso ao que há de mais atual e variado em EI.

A.2.1 LOLITA

O sistema LOLITA (*Large-scale, Object-based, Linguistic Interactor, Translator and Analyser*) está sob desenvolvimento na Universidade de Durham desde 1986. Seguindo os princípios de PLN, o sistema projetou-se como uma plataforma de proposta geral para prover capacidades de PLN a fim de suportar várias aplicações em múltiplos domínios [GAR 99]. Diferentes tipos de aplicações podem ser construídas em torno do núcleo original do LOLITA, o qual oferece duas facilidades principais: análise, convertendo textos para uma representação lógica de seu significado, e geração, apresentando estas informações lógicas na forma de texto [MOR 95]. Uma das vantagens do uso de um núcleo de PLN de propósito genérico é a imediata reflexão nas aplicações das melhorias realizadas neste núcleo. Contudo, ao contrário de muitos outros sistemas de PLN, o LOLITA não é uma estrutura que pode ser modelada para diferentes domínios, ele armazena conhecimento de domínios específicos para sustentar aplicações de forma apropriada [COS 97].

Atualmente, cerca de 20 pesquisadores trabalham no LOLITA no Laboratório de Engenharia de Linguagem Natural de Durham. Diversos tipos de aplicações foram construídas em torno do núcleo do sistema e incluem, entre outras [GAR 99]:

- Extração de informações: produção de sumários e *templates*;
- Tradução simples, baseada em significado: atualmente, de Italiano para Inglês;
- Consultas em linguagem natural: fornece-se informações para o LOLITA e depois se faz perguntas sobre essas informações.
- Tutor na língua chinesa: uma pseudogramática, mesclando características gramaticais Chinesas e Inglesas, permite a detecção de erros de estudantes de Chinês usando construções Inglesas.

Uma das melhores aplicações desenvolvidas sobre LOLITA é um sistema de extração de informações. Esse trabalho foi financiado pela Siemens Plessey Systems e por uma agência do governo inglês, e envolve extração de informações de acordo com as especificações do usuário a partir de artigos de jornal, artigos de *teletext* e relatórios da polícia [COS 97]. A importância do sistema LOLITA em EI foi demonstrada por sua performance de sucesso nas competições do MUC-6 e do MUC-7, atendendo todas as quatro tarefas de extração definidas.

O LOLITA apresenta uma característica de “larga escala”. Muitos resultados bons em PLN são demonstrados com base unicamente em pequenos exemplos, falhando em ambientes reais, onde o volume e a complexidade dos dados de entrada é muito maior. Essa é uma das principais causas para a lenta introdução da tecnologia de PLN a

nível comercial. O LOLITA foi montado desde o início como base para a construção de aplicações de larga escala. Alguns detalhes desse sistema mostram que ele é um dos maiores do seu tipo no mundo [COS 97]:

- Analisa textos reais.
- Realiza análise morfológica, gramatical, semântica, pragmática, análise de discurso e funções de geração de *templates*.
- Baseia-se em uma rede semântica de 100k nodos, compatível com a WordNet.
- Contém 1500 regras gramaticais.
- Desenvolvido em Haskell com mais de 50k linhas, correspondendo a aproximadamente 500k linhas em uma linguagem imperativa, como C.
- Extensivo tratamento de linguagem natural em Inglês, e tratamento parcial de Chinês e Espanhol.
- Ampla utilização: provê aplicações em áreas que variam de PLN a Interfaces para Bancos de Dados, EI, Tutoriais de Línguas, Traduções, etc.
- Boa velocidade de processamento.
- Apresenta avançadas capacidades de inferência sobre a rede semântica utilizada em seu núcleo.

A.2.1.1 Arquitetura

O LOLITA é um sistema núcleo suplementado com um conjunto de aplicações que usam as facilidades de PLN fornecidas por ele [GAR 99]. A Figura A.1 mostra a arquitetura utilizada no MUC; ou seja, o núcleo do LOLITA e as aplicações específicas para atender as tarefas de extração definidas na conferência. A parte mais importante do núcleo é a grande BC chamada de rede semântica. Essa rede é usada em muitos estágios da análise de documentos, e os resultados da análise são adicionados nela novamente, conforme sua representação lógica de documentos, sem ambigüidade. Os estágios de análise são bastante padronizados e arranjados em um *pipeline* (linha de processamento de dados). Cada estágio é implementado através de regras que definem seu funcionamento. O sistema não utiliza qualquer forma estocástica ou técnicas adaptativas em seu núcleo. As aplicações podem ler os resultados das análises a partir da rede semântica, e geralmente interrogar o conteúdo da rede. Algumas facilidades de "suporte" são providas pelo núcleo para auxiliar a construção das aplicações, como o mecanismo gerador de *templates* e o gerador de linguagem natural (LN), o qual traduz partes da rede semântica para o inglês.

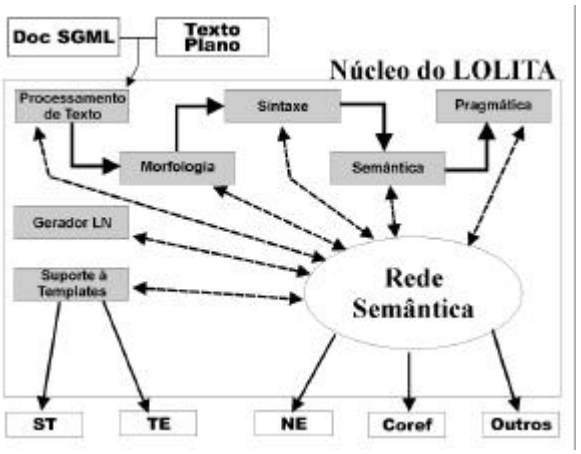

FIGURA A.1 - Arquitetura do Sistema LOLITA e Aplicações Usadas no MUC

A.2.1.1.1 Rede Semântica

A parte mais importante do núcleo do LOLITA é a BC, representada através de uma rede semântica. Essa rede é formada por um grafo direcionado com mais de 100.000 nodos contendo resultados da análise de documentos; ou seja, do processamento dos documentos para uma representação lógica [GAR 99]. Ela pode ser acessada por todas as aplicações construídas sobre o núcleo do sistema. Um nodo da rede semântica corresponde a uma entidade ou um evento, e cada nodo possui as seguintes estruturas [MOR 95]:

- **Um conjunto de *"links"*:** um *link* é uma relação entre nodos e contém um arco e um conjunto de alvos. Os alvos são outros nodos e o arco também é um nodo. Existem cerca de sessenta tipos de arco, os quais representam diferentes tipos de relações entre nodos. No exemplo da Figura A.2, temos os arcos: *subject_*, *action_* e *object_*. O significado de um nodo é dado por suas conexões; ou seja, sua posição relativa na rede.
- **Um conjunto de "controles":** um controle representa as informações básicas de um nodo, como tipo (evento, entidade, relação), família (humano, inanimado, organização), tipo léxico (nome, preposição, advérbio), entre outros. Um importante controle é o *rank*, que codifica informação de quantificação. Conceitos de conjuntos genéricos tem um *Universal Rank* (**U**), objetos especificadamente definidos tem um *Named Individual Rank* (**NI**), e objetos individuais em geral, ou seja, não definidos, um *Individual Rank* (**I**).
- **Um nome associado:** palavra que caracteriza o significado do nodo.

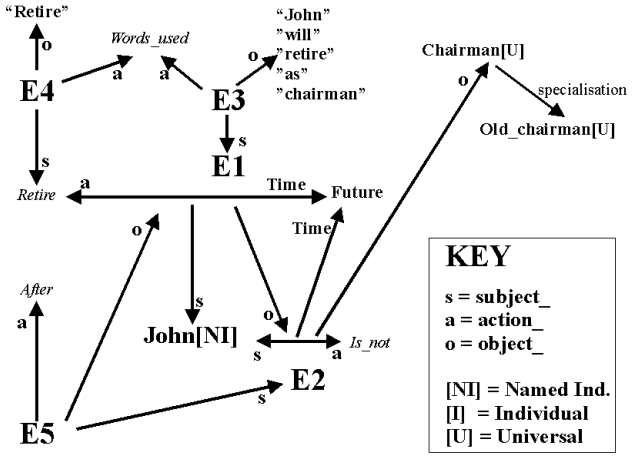

FIGURA A.2 - Rede Semântica para a Sentença "John will retire as chairman"

A rede semântica do LOLITA é utilizada para armazenar diversos tipos de informação, como hierarquias (homem é_um mamífero é_um vertebrado), informações léxicas, eventos prototípicos (que definem restrições a outros eventos) e outros tipos de eventos. Entretanto, a maior parte da rede, em torno de 70%, é formada pela WordNet, uma base de dados em forma de rede com informações léxicas e semânticas sobre palavras na língua inglesa [MOR 95].

Antes de participar do MUC, o documento original era descartado pelo LOLITA após a criação de sua representação na rede semântica. Contudo, esta habilidade de referenciar o documento original pode servir para vários propósitos, não somente para as tarefas do MUC. Neste sentido, um mecanismo de referência foi criado e adicionado ao núcleo do sistema. Esse mecanismo permite que a estrutura do texto seja toda representada na rede [GAR 99].

A.2.1.1.2 Pré-processamento

A análise de um documento inicia com uma representação SGML específica do LOLITA, chamada SGML *tree*. Aplicações individuais devem converter de seus formatos (*PostScript*, LaTeX, HTML, etc.) para este formato interno [GAR 99]. Na aplicação em extração de informações para o MUC, o conversor é somente um simples *parser* SGML, pois, dependendo da tarefa de extração, o documento já apresentava uma estrutura SGML.

O pré-processador inclui estruturas adicionais à árvore SGML quando necessário. Em particular, as seguintes estruturas são tratadas: citações, parágrafos, sentenças e palavras [MOR 95]. Além disso, para citações são inseridos marcadores sobre todas as sentenças entre aspas. Finalmente, para cada palavra é alocada uma TextRef; ou seja, um apontador que referencia a sua ocorrência no documento original.

A.2.1.1.3 Análise Morfológica

A análise morfológica é realizada sobre a árvore SGML. Cada folha é um *token* e os nodos dessa árvore representam a estrutura do documento. Algumas transformações são feitas, como expandir contrações do tipo *"I'll"* para *"I will"* e expressões monetárias e numéricas, tais como, *"$10 million"* para *"10 million dollars"*, bem como transformar certos tipos de expressões idiomáticas superficiais, como, por exemplo, a expressão *"in charge of"* [GAR 99]. Além disso, algumas divisões de palavras por hifenação também são tratadas.

Após este primeiro tratamento, mais superficial, o processamento morfológico é executado sobre todas as folhas da árvore SGML. Pesquisas ao dicionário do LOLITA são realizas com base no afixo extraído das palavras. Se bem sucedida a pesquisa, a palavra é ligada ao nodo léxico e ao semântico indicados no dicionário, permitindo acesso a essas informações durante o resto desta e das demais fases [MOR 95].

A retirada do afixo das palavras tem como prejuízo a perda de informações como número e gênero. Para que isso não ocorra, essa informação é representada usando-se um sistema de "características", as quais são associadas ao afixo [COS 97]. Essas características são utilizadas pela fase posterior, análise sintática. Outra característica associada inclui as classes gramaticais de palavras (nomes, verbos, etc.) e algumas características baseadas na semântica. Finalmente, possíveis categorias sintáticas para uma palavra são determinadas a partir de informações do nodo léxico e, algumas vezes, do semântico. Assim, cada folha é mapeada para um conjunto de alternativas, variando em categoria e característica, as quais representam as possíveis interpretações dessa folha.

A.2.1.1.4 Análise Sintática

Esta fase é formada pelas seguintes etapas [GAR 99]:

- **Pré-parser:** identifica e suporta expressões monetárias.
- ***Parsing* de sentenças inteiras:** utiliza o algoritmo de Tomita [TOM 86] para gerar uma "floresta gramatical", ou seja, um conjunto de árvores gramaticais na forma de um grafo acíclico direcionado. Devido à complexidade da gramática, essa floresta é freqüentemente muito grande, implicando em várias análises gramaticais possíveis, as quais são indicadas no grafo.
- **Decodificar a "floresta gramatical":** a floresta é seletivamente explorada a partir do nodo superior, utilizando um conjunto de heurísticas baseadas, por exemplo, nas características extraídas na fase de

morfologia e em probabilidades manualmente definidas de certas construções gramaticais aparecerem. Erros de representação gramatical na floresta e árvores indesejadas envolvem um aumento de custo de processamento dessa e das fases posteriores, bem como do funcionamento do sistema como um todo. Logo, o objetivo desta etapa é encontrar um conjunto de árvores de menor custo; ou seja, que melhor represente as sentenças e com o menor tamanho.

- **Seleção da melhor árvore gramatical:** a partir do conjunto de árvores gerado na etapa anterior e com base em várias heurísticas sobre a forma (distribuição dos nodos) das árvores, é escolhida a melhor árvore. Por exemplo, o usuário pode preferir uma árvore mais profunda que ampla.
- **Normalização:** transformações baseadas em sintaxe, as quais preservam o significado das sentenças, são aplicadas às árvores para reduzir o número de casos possíveis de serem analisados na fase semântica, simplificando, assim, o processamento posterior. Um caso comum é transformar construções na forma passiva para a ativa, como, por exemplo: "Eu fui mordido por um cachorro" para "Um cachorro me mordeu". Outra classe de transformações envolve ocorrências do tipo "Você foi surpreendido" para "*ALGO* surpreendeu você", tornando explícito o objeto que realizou a ação.

A.2.1.1.5 Análise Semântica e Pragmática

Nesta fase, a árvore gramatical é transformada em parte da rede semântica. O processo de conversão é dividido em dois estágios: semântico e pragmático [GAR 99].

A análise semântica é geralmente composicional; ou seja, o significado de uma árvore é construído a partir do significado de suas subárvores. Um mecanismo caminha através da árvore gramatical, aplicando regras semânticas com base, principalmente, no tipo sintático da frase do corrente nodo da árvore.

Contudo, o sistema prevê situações de análise semântica levando em conta o contexto. Isso impede que o resultado semântico seja meramente composicional. As informações de contexto são geradas a partir das árvores gramaticais analisadas anteriormente à que está no momento sob análise, as quais já apresentam informações semânticas associadas. Assim, o LOLITA controla as possíveis informações que dizem respeito ao assunto sob análise em ordem de ocorrência, sendo este processo também usado para resolver anáforas.

O significado semântico da maioria das folhas muitas vezes é determinado pelo nodo semântico associado com a palavra sob análise na fase de morfologia. Esse é estendido para as folhas parentes na forma de uma "função estrutural", indicando a função de cada nodo parente dentro da estrutura semântica da árvore.

O estágio pragmático retira ambigüidades e verifica tipos. Ambigüidades léxicas e anáforas são resolvidas usando heurísticas de preferência, as quais são primeiramente aplicadas para tirar a ambigüidade da ação do evento. Uma vez que a ação é identificada, qualquer conhecimento disponível a partir do protótipo de evento associado com essa ação dentro da rede semântica pode ser utilizado para margear pragmaticamente improváveis entendimentos da árvore sob análise, bem como para auxiliar a retirar ambigüidades dos elementos restantes relacionados ao evento.

O conhecimento do contexto sob análise, juntamente com o assunto do documento, informado ao sistema antecipadamente, influencia na escolha do sentido a

ser associado a cada palavra analisada: são selecionados aqueles significados semanticamente semelhantes com o significado presente no contexto ou no assunto do documento, onde a semelhança semântica é computada com base na distância entre os nodos na rede semântica. Outro fator que pesa na escolha de um significado a ser associado a uma palavra ou a uma árvore, é o montante de conhecimento que o sistema já tem sobre um dado assunto ou evento, ou a freqüência de uso desse significado.

Uma vez que um evento não contém ambigüidades, o sistema tenta estabelecer conexões plausíveis entre ele e o documento processado previamente [MOR 95].

A.2.1.1.6 Suporte a Templates

O núcleo do LOLITA contém um processo de produção de *templates* independente do domínio de aplicação [GAR 99]. Esse mecanismo pesquisa informações na rede semântica e usa regras de inferência para derivar fatos implícitos e formatos de saída genéricos. Um *template* contém um conjunto predefinido de *slots* com regras de preenchimento associadas, as quais direcionam a pesquisa para a informação apropriada na rede semântica.

A.2.1.1.7 Gerador de Linguagem Natural

Conforme explicado anteriormente, o gerador LN, cria, a partir da representação formal utilizada na rede semântica, uma saída do conhecimento armazenado na forma de texto em inglês.

A.2.2 LaSIE

O sistema LaSIE (*Large Scale Information Extraction*) foi desenvolvido na Universidade de Sheffield como parte de uma pesquisa em EI e, mais genericamente, em PLN [WIL 2000]. Esse sistema, simples e integrado, constrói um modelo unificado do documento original (diagrama), o qual é utilizado para vários propósitos, mas, especialmente, para produzir resultados para as tarefas do MUC. A construção de uma única e rica representação do documento exige que o LaSIE execute análises léxicas, sintáticas e semânticas.

O LaSIE foi projetado para ser um sistema de extração de informações de alta precisão; ou seja, altos níveis de *precision* [WIL 2000]. Contudo, apesar de priorizar o *precision*, o aumento dessa qualidade não afetou significativamente os níveis de *recall*. O LaSIE obteve níveis de *recall* condizentes com a maioria dos sistemas que apresentaram altos escores no MUC. Todavia, esta priorização fez com que seus desenvolvedores não se comprometessem com a eficiência do tempo de extração e de desenvolvimento do sistema. Entre as características que distinguem o sistema estão as seguintes [GAI 95]:

- Uma abordagem integrada, permitindo que o conhecimento dos vários níveis lingüísticos seja aplicado para cada tarefa do MUC-6, como, por exemplo: usar conhecimentos de reconhecimento de entidades (NE) para a tarefa de CO.
- A ausência de qualquer informação léxica necessária para o processo de análise gramatical é suprida dinamicamente através de uma análise morfológica e léxica.
- O uso de uma gramática derivada semi-automaticamente da estrutura Penn TreeBank que armazena informações semânticas e sintáticas através de uma “floresta” de árvores gramaticais [PEN 99].

- O uso e a aquisição de um modelo global para extração, em particular para as tarefas de CO e ST.
- Um módulo de sumarização que produz um breve resumo em linguagem natural dos eventos de um domínio (*scenario events*).

A.2.2.1 Arquitetura

Altamente modularizado, o LaSIE é formado essencialmente por um *pipeline* de módulos que processam todo o documento antes que o próximo módulo seja invocado [HUM 99]. O método de extração utilizado envolve a construção composicional de representações semânticas de sentenças individuais de um documento de acordo com regras semânticas, juntamente com as partes que constituem a estrutura da frase, obtidas pelo analisador sintático usando regras derivados de uma gramática livre de contexto [GAI 95]. As representações semânticas das sucessivas sentenças são então integradas em um "modelo de discurso" que, após o processamento de todo o documento, pode ser visto como uma especialização de um modelo global, ajustada pelo sistema para processar cada documento.

O alto nível de estruturação do LaSIE é apresentado na Figura A.3. A arquitetura *pipeline* desse sistema consiste em três estágios principais: pré-processamento léxico, análise gramatical e interpretação semântica, e interpretação de discurso. Nenhum desses estágios corresponde diretamente a uma das tarefas de extração definidas a partir do MUC-6. Isso porque o LaSIE foi projetado como um sistema de extração de propósitos gerais, inicialmente direcionado para as tarefas do MUC-6, mas não restrito às mesmas. Além disso, todos os resultados para as tarefas são gerados somente após a construção completa da representação do documento. Isso reflete a decisão de usar informações derivadas a partir de todos os níveis de processamento lingüístico para a execução de cada uma das tarefas do MUC-6. Primeiramente realizam-se todos os estágios de extração para depois, com base na informação extraída, gerar os resultados para as tarefas do MUC [HUM 99]. Os estágios de extração do LaSIE são apresentados na Figura A.3.

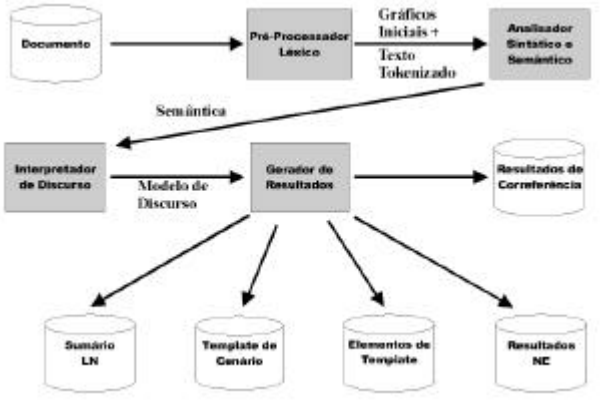

FIGURA A.3 - Arquitetura do Sistema LaSIE

A.2.2.1.1 Pré-processador Léxico

A entrada do estágio de pré-processamento léxico é um arquivo ASCII padrão contendo marcações SGML nas separações de parágrafo. A saída é formada por duas partes: uma seqüência de diagramas lexicalmente organizados para uso no analisador gramatical e uma representação de *tokens* do documento original, para posterior

reconstrução do mesmo com o incremento de marcações SGML pelo módulo de geração de resultados [GAI 95].

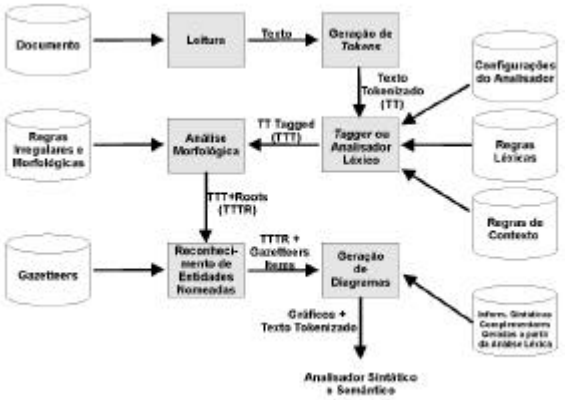

FIGURA A.4 - Pré-processamento Léxico

O pré-processamento léxico consiste na geração de *tokens* e separação das sentenças de entrada, na análise léxica dos *tokens*, na análise morfológica para obter as formas primitivas das palavras, na busca de padrões com o uso de listas pré-compiladas de nomes de entidades e, finalmente, na criação de estruturas léxicas com as características das sentenças, as quais entrarão no analisador gramatical, próximo estágio. Um processamento léxico convencional de PLN não é utilizado. Análises léxicas e morfológicas dinâmicas provêem toda a informação requerida pelo analisador gramatical. Esse estágio divide-se nas seguintes etapas apresentadas na Figura A.4 [GAI 95]:

- **Geração de *Tokens*:** nesta etapa, os *tokens* são separados e associados a um identificador (número seqüencial gerado pelo sistema). Esses identificadores são preservados durante todos os estágios de extração para facilitar a inclusão de marcações SGML para as tarefas de NE e CO pelo gerador de resultados. Além disso, o documento é formatado para uma sentença por linha e *tokens* separados por espaços em branco, conforme exigido pelo analisador léxico ou *tagger*.
- **Tagger ou Analise Léxica:** o analisador léxico é configurado através de regras e faz uso do Penn TreeBank [PEN 99], que fornece uma estrutura de armazenamento de informações léxicas, sintáticas e semânticas. Dentre as possíveis configurações do *tagger* está a inclusão de novos *tags* para análise de datas, marcações SGML e símbolos de pontuação.
- **Análise Morfológica:** todos os verbos e substantivos devem passar pelo analisador morfológico, que retorna a forma primitiva dos mesmos para uso pelo analisador sintático. Essa análise é baseada em um conjunto de 34 expressões regulares, conjuntamente com cerca de 3000 expressões irregulares.
- **Reconhecimento de Entidades Nomeadas:** Antes de dar início a analise sintática, tenta-se identificar e marcar as entidades relacionadas às sentenças. Comparam-se os termos de entrada com uma lista pré-carregada de nomes próprios, formatos de datas, etc. Além disso, pesquisam-se nomes comuns que podem indicar entidades. Essas listas contém nomes de organizações, designadores de companhias (“Co.”, “Ltda”, “PLC”, etc.); títulos pessoais (“*President*”, “*Mr.*”, etc.); nomes de pessoas; unidades (“*dollars*”, “*pounds*”, etc.); nomes de lugares, como, países, estados, províncias e cidades,

derivados do Gazetteer; e expressões de tempo, como, *"first quarter of"*. Além dessas listas, existem listas de *triggers* ("gatilhos"), que são palavras que indicam que um *token* próximo é uma entidade. Elas contêm *triggers* de localização, como, *"Gulf"* e *"Mountain"*, e de organizações, como, por exemplo: *Agency*, *Ministry*, *Airline* e *Association*.

- **Geração de Diagramas:** Para cada sentença, constrói-se um diagrama com uma entrada para cada item léxico e entidade encontrada na sentença. Cada entrada contém um valor com informações sobre os itens gerados nas etapas acima.

A.2.2.1.2 Analisador Sintático e Semântico

O *parser* do LaSIE é do tipo *bottom-up* e processa uma gramática livre de contexto; ou seja, processa a gramática a partir dos componentes não terminais das regras de produção até os terminais. Foi implementado em Prolog, usando facilidades características dessa linguagem de programação [GAI 95]. Durante o processo de análise sintática, representações semânticas das sentenças presentes no documento são construídas, utilizando inteiramente a unificação de termos do Prolog. Logo, a análise semântica é composicional, pois as informações semânticas são agregadas à estrutura sintática durante o processo de análise, construindo o resultado semântico durante o caminhamento pelas regras de produção [WIL 2000]. Portanto, as informações semânticas são baseadas na estrutura sintática da sentença. Este processo de *parsing* ocorre em duas etapas, usando gramáticas diferentes [HUM 99] (Figura A.5):

- ***Parser* NE:** a primeira etapa do *parsing* vale-se de uma gramática especial para nomear entidades. Seu objetivo é encontrar termos relevantes à tarefa de NE. Essa gramática é baseada em regras de produção criadas manualmente que usam as informações geradas pelo analisador léxico (*tokens*, informações morfológicas e semânticas derivadas do Gazetteer). A criação das regras de produção é simples: padrões são detectados no documento e manualmente adicionados à gramática.
- ***Parser* de Sentença:** a gramática do *parser* de sentença, livre de contexto, é formada por um conjunto de regras de produção com o objetivo de realizar a análise gramatical de conteúdos variados, sendo essas regras de aplicação genérica. Essa gramática foi derivada a partir da Penn TreeBank-II (PTB-II) [PEN 99].

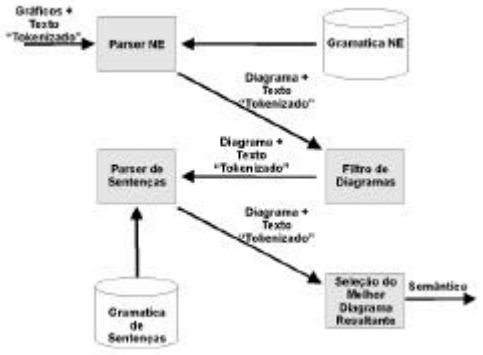

FIGURA A.5 - Analisador Sintático e Semântico

Quando o processo de análise termina; ou seja, quando o *parser* chega ao topo da estrutura de análise (regras terminais), um algoritmo de seleção da melhor estrutura com as informações sintáticas e semânticas (diagrama) escolhe uma única análise [GAI 95]. Isso ocorre porque o sistema pode apresentar mais de uma interpretação sintática

e/ou semântica de uma sentença, conforme as regras de produção existentes. O processo de seleção utiliza uma heurística que escolhe a menor seqüência de regras de análise que forneça o máximo conteúdo semântico [HUM 99]. Caso existam várias alternativas equivalentes, a última análise gerada é selecionada. Essa abordagem elimina qualquer ambigüidade detectada na análise e garante que uma única análise será passada para o interpretador de discurso.

A.2.2.1.3 Interpretador de Discurso

Este estágio traduz a representação semântica produzida pelo analisador sintático e semântico em uma representação de instâncias, suas classes ontológicas e suas propriedades, utilizando a linguagem de representação de conhecimento XI (Figura A.6) [WIL 2000]. A XI permite a definição de uma hierarquia de classificação cruzada (*cross-classification*) e a associação de propriedades arbitrárias com classes e instâncias. Como resultado tem-se uma representação do domínio de extração (modelo) através dessa rede semântica.

FIGURA A.6 - Interpretador de Discurso

Além de algumas informações adquiridas das listas Gazetteer, e as correspondentes regras gramaticais, todo o conhecimento específico do domínio no LaSIE esta concentrado no modelo de universo (domínio) do interpretador de discurso. Esse modelo é expresso utilizando uma rede semântica, com base na linguagem XI, onde os nodos representam "conceitos" (classes ou instâncias), com atributos de valor associados as suas propriedades, e os arcos, as relações entre os nodos, as quais modelam uma hierarquia conceitual e suportam heranças de propriedades [HUM 99]. O modelo usado para as tarefas do MUC-6, por exemplo, foi extremamente simples, consistindo em 40 classes de objetos pré-definidas e 46 tipos de atributos.

Durante o processamento de um documento, novas classes (nodos) de objetos e eventos são automaticamente adicionadas ao modelo corrente para enriquecer a hierarquia (estágio de adição de instâncias). Os novos nodos são adicionados como sub-classes diretas de seus objetos ou eventos originais. Assim, as propriedades de uma classe são herdadas pelas suas sub-classes (estágio de expansão de pressuposição). Esse mecanismo hierárquico com herança de propriedades permite a coreferência entre instâncias de uma classe no modelo (estágio de resolução de coreferência) [GAI 95].

Os atributos de um nodo podem ter valores atômicos ou valores especificados por regras de inferência associadas com esse nodo. Tais regras são baseadas na linguagem XI. A adição de uma instância ou propriedade de uma certa classe ao modelo, enquanto ocorre o processamento do documento, conduzirá à avaliação de qualquer regra de inferência hereditária, podendo causar a adição de mais uma instância

ou propriedade ao modelo, ou a reclassificação das instâncias existentes (estágio de expansão do modelo) [GAI 95].

A.2.2.1.4 Gerador de Resultados

Os resultados de extração são produzidos por pesquisas sobre o modelo de domínio criado no estágio de interpretação de discurso. A maioria das classes semânticas e tipos de propriedades pré-definidos no modelo já são motivados pelas características de extração requeridas pelo usuário [HUM 99]. Neste sentido, a geração dos resultados somente envolve a recuperação, a partir do modelo de discurso, daquelas instancias com as propriedades requisitadas e o correto formato dos valores dessas propriedades para o preenchimento dos *slots* dos *templates* [GAI 95].

A.2.3 FASTUS

O FASTUS (*Finite State Automata-based Text Understanding System*) extrai informações de documentos em linguagem natural para bases de dados estruturadas e outras aplicações usando autômatos de estados finitos. Sua performance no MUC demonstrou a viabilidade dessa tecnologia para tarefas de EI [APP 95]. Contudo, seus autores destacam que o sistema realiza extração de informações, não compreensão de textos, sendo mais eficiente quando apenas uma parte do documento contém informações relevantes, e existe uma representação relativamente simples e rígida, na qual a informação é mapeada [HOB 99]. EI é uma tarefa muito mais simples e tratável, caracterizada por informações específicas a serem extraídas dos documentos. Compreensão de textos, em contraste, é difícil, e apresenta um grande número de problemas ainda não resolvidos. O FASTUS foi desenvolvido sem usar profundas técnicas de PLN. Entre suas principais características, destacam-se as seguintes: velocidade de processamento, uso de regras robustas de extração, uso de padrões de extração para domínios específicos e escalabilidade horizontal entre seus processos [HOB 99].

O sistema passou por significativas revisões desde sua primeira versão, atualmente não compartilha mais nenhuma linha de código com a original. Contudo, as idéias fundamentais por traz do FASTUS estão contidas na versão atual: uma arquitetura em *pipeline* de processos, cada um provendo um nível complementar de análise dos documentos e unificação de seus resultados [HOB 99]. Existem versões em inglês e japonês desse sistema.

A.2.3.1 Arquitetura

O FASTUS é formado por uma série de processos, chamados *transducers*, organizados em um *pipeline*, que transformam os documentos nos *templates* de saída [APP 95]. Essa arquitetura é muito flexível, e tem sido aplicada com sucesso em um grande número de diferentes tarefas e domínios de EI [HOB 99]. Cada *transducer* (ou fase) do *pipeline* pode utilizar os resultados da fase anterior através da estrutura que transfere os dados entre as fases, forma seqüencial; ou de forma não determinística, acessando todas as informações já disponibilizadas pelos *transducers* anteriores em seus *templates* de saída [APP 95]. Tipicamente, contudo, todas as fases do FASTUS, exceto a fase final, seguem o primeiro regime, os *templates* com os fragmentos de texto extraídos por cada *transducer* são unidos para formar a análise final pelo módulo *merger* [HOB 99]. Os *transducers* podem passar segmentos não analisados de textos para a fase seguinte, ou eliminar trechos dos documentos de entrada. É possível variar o

número de *transducers* para cada caso de aplicação (ver *preprocessor* abaixo), bem como controlar precisamente o que cada fase aceita como entrada e produz na saída. A última versão do FASTUS possui a seguinte seqüência de *transducers* [APP 95] [HOB 99]:

- ***Tokenizer*:** recebe o conjunto de caracteres ASCII que forma os documentos a serem analisados, agrupa-os em palavras e transforma-as numa seqüência de *tokens*. Além disso, essa fase detecta abreviações, determina limites de sentenças e normaliza prefixos e sufixos.
- ***Multiword Analyzer*:** reconhece seqüências de *tokens*, combinando-as para formar um item léxico único (termos), como, por exemplo: *"because of"*.
- ***Preprocessor*:** neste ponto, o projetista da aplicação pode inserir módulos adicionais ao FASTUS para o manuseio de construções de palavras complexas ou termos baseados em mais de uma palavra, os quais serão tratados automaticamente pelo sistema. Um exemplo é transformar a seqüência de palavras "vinte e três" no número "23".
- ***Name Recognizer*:** reconhece seqüências de palavras que podem ser identificadas, sem ambigüidade: nomes próprios, por exemplo. Além disso, encontra palavras desconhecidas e seqüências de palavras capitalizadas que não combinaram com os padrões de nomes conhecidos, assinalando-as para que as fases subseqüentes possam determinar seu tipo a partir do conhecimento do contexto de extração.
- ***Parser*:** recebe a lista de *tokens*, produzida nas etapas anteriores, e forma uma lista de frases a partir das quais constrói as estruturas básicas da sintaxe inglesa, realizando uma análise gramatical simples. São geradas somente aquelas estruturas sintáticas que podem ser construídas quase que completamente sem ambigüidade, utilizando regras de estados finitos. A saída dessa fase é composta por grupos de substantivos, os quais "encabeçam" as frases nominais, e grupos de verbos. Pontuações, preposições, pronomes relativos e conjunções são passadas como "partículas" para as próximas fases.
- ***Combiner*:** consiste em combinar as frases fornecidas pelo *parser* a fim de obter, quando possível, estruturas maiores, unindo informações adjacentes no documento. O exemplo típico de informação tratada por esta fase são os apositivos, tais como: "João Carlos, 26, diretor da RJR Informática". Outras informações normalmente unificadas são entidades de um mesmo tipo e preposições de tempo e local.
- ***Domain*:** reconhece combinações particulares de assuntos, verbos e objetos necessários para o preenchimento correto dos *templates* para uma dada tarefa de extração de informações. Enquanto as fases iniciais do FASTUS podem ter uma menor dependência do domínio de extração, essa fase é fortemente dependente dele. Contudo, as últimas versões do FASTUS permitem facilmente a customização dessa etapa para um novo domínio de aplicação.

O FASTUS ainda inclui o *merger* para unir os *templates* gerados pela fase *domain* [APP 95]. Essa operação de unificação é executada a partir de especificações precisas dadas pelo projetista do sistema na definição do domínio. Ele determina cada *slot* e o tipo de dado que irá preenchê-lo. Para cada tipo, o FASTUS provê procedimentos que comparam dois itens do mesmo tipo e decidem se eles são idênticos ou distintos, ou se um é mais ou menos genérico que o outro, ou ainda quando eles são incomparáveis. Dependendo desse resultado, as instruções de unificação definem se os

objetos podem ser unificados ou não. Caso sim, esses objetos devem ser combinados como itens distintos de um conjunto. O *merger* parte da hipótese que a comparação e decisão de unificação são independentes de contexto; isto é, não é necessário conhecer nada além dos valores dos *slots* para determinar se eles podem ser unificados.

A.2.4 UMASS

A pesquisa em EI na Universidade de Massachussetts tem ponto culminante no sistema de extração UMASS, sigla que identifica a própria universidade. Esse sistema é formado por componentes de PLN portáveis e treináveis com o objetivo de eliminar as tarefas manuais de engenharia de conhecimento [FIS 95].

O UMASS apresenta duas principais versões. A primeira, utilizada até sua participação no MUC-5, e a segunda, desenvolvida para o MUC-6. As principais mudanças entre as versões encontram-se nos módulos de reconhecimento de *strings*, analisador léxico, analisador de sentenças, algoritmo de construção automatizada de dicionário, analisador de discurso e analisador de coreferência [FIS 95]. Em sua última versão o UMASS está dividido em quatro principais componentes que atendem as tarefas do MUC (Tabela 7.1).

TABELA A.1 - Módulos do UMASS

Especialistas\Tarefas	**NE**	**CO**	**TE**	**ST**
BADGER: analisa sentenças.	X	X	X	X
CRYSTAL: gera um dicionário conceitual de nodos.		X	X	X
RESOLVE: analisa coreferências.		X	X	X
WRAP-UP: estabelece ligações relacionais entre entidades.			X	X

A.2.4.1 Arquitetura

A estrutura do UMASS é formada por quatro módulos principais que extraem as informações dos documentos através de uma seqüência de atividades (Figura A.7). O primeiro passo de extração é realizado por algoritmos de tratamento de *strings* para o reconhecimento e manipulação de nomes próprios (lugares, pessoas e organizações), datas, moedas e porcentagens, sendo independentes entre si e dos demais módulos. As rotinas especializadas em nomes próprios utilizam dicionários de termos para o reconhecimento. No caso de nomes de organizações e pessoas, dicionários do Laboratório de Recuperação de Informações da UMASS foram usados. Para o tratamento de lugares, usou-se um subconjunto do conteúdo do dicionário de termos geográficos Gazetteer [FIS 95].

O segundo passo da extração é realizado pelo analisador de sentenças BADGER. Ele substituiu o módulo CIRCUS, não sendo significativamente diferente desse, descrito anteriormente e utilizado até o MUC-5. Esse analisador consiste de uma coleção de processos para a análise léxica, uma árvore de decisão treinável para localizar construções apositivas, analise sintática local e instanciação semântica de informações relevantes extraídas dos documentos, usando estruturas de informação chamadas de nodos conceito (NC) [LEH 91]. O conjunto de definições de NC utilizado pelo BADGER forma um dicionário representando as regras de extração [SOD 95a]. Como resultado do processamento, são disponibilizadas instâncias dos NC, representando as informações extraídas do documento. Um mesmo trecho de texto pode ser instanciado por mais de um NC, pois múltiplas definições de NC podem ser aplicadas ao mesmo fragmento de texto. Diferentes dicionários de NC podem ser conectados ao BADGER, dependendo da tarefa e do domínio de aplicação. Isto demonstra a completa portabilidade desses dicionários. O BADGER também conta com

um dicionário de estruturas gramaticais, bem como informações semânticas associadas às características léxicas e sintáticas, organizadas hierarquicamente e aplicáveis a domínios específicos [FIS 95].

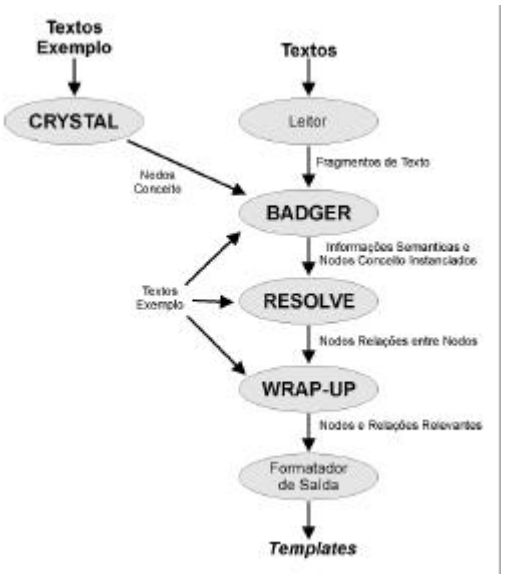

FIGURA A.7 - Arquitetura do Sistema UMASS

No MUC-6, o BADGER reconhecia 27 estruturas gramaticais e levava 24 horas para criar um dicionário de estruturas para a análise gramatical. Para isso, começa-se com um conjunto de expressões léxicas inicial, independentes de domínio, que incluía preposições, determinantes e um grande número de verbos regulares. Depois, adicionavam-se os termos que ocorreram 500 ou mais vezes nos seis últimos anos de artigos do Wall Street Journal, armazenados na coleção do TIPSTER, baseado em uma inspeção manual. O dicionário gramatical tinha 2048 entradas de características léxicas. Características semânticas foram determinadas para estes termos e também para os termos de interesse para a tarefa de ST que aparecia no conjunto formal de treino: 100 textos. A associação de informações semânticas às características léxicas tinha 5453 entradas. São utilizadas 45 características semânticas e a criação desse dicionário de características semânticas levou 36 horas para ser construídos no caso do MUC-6 [FIS 95]. A criação do dicionário de definições de NC foi totalmente automatizada pelo CRYSTAL, um sistema indutivo para a construção dos dicionários que apóia o BADGER. Nos primeiros MUCs, utilizava-se o sistema AutoSlog, descrito anteriormente, para gerar os dicionários de NC. Todavia, o AutoSlog requeria interação humana para estimar a qualidade dos NC definidos. O CRYSTAL não requer muita revisão humana, criando os dicionários com técnicas de *Machine Learning* [CAR 98] [SOD 95].

Transformar os resultados do BADGER nos *templates* do MUC-6 exige identificar tipos semânticos de informações, preencher os atributos dos NC e reconhecer coreferências. Essas tarefas são executadas primariamente por rotinas de consolidação dentro do módulo WRAP-UP (terceiro passo), um analisador de discurso treinável que estabelece ligações relacionais entre informações coreferentes, e do módulo RESOLVE, um analisador de coreferência treinável [FIS 95].

Quando o WRAP-UP determina o papel de um substantivo numa frase, ou possível relação entre duas frases nominais, ele o faz com base em várias evidências. Tais evidências são extraídas por um mecanismo independente de domínio, o qual codifica as características de informações nos atributos de NC, a posição relativa de

informações coreferentes na rede de nodos, e os padrões de verbos nos quais a frase nominal aparece. Muitas vezes, essas evidências fornecem uma interpretação consistente, mas outras vezes também discordam uma da outra. As decisões são gerenciadas por árvores de decisão treinadas; logo, discrepâncias nos dados de entrada são manipuladas com base em situações similares, encontradas durante o treinamento. O WRAP-UP é mais efetivo quando seu treinamento lhe fornece a experiência necessária para lidar com diversos tipos de inconformidades [FIS 95].

O RESOLVE gerencia fusões de informações coreferenciadas pelo WRAP-UP. Ele determina quando duas frases nominais se referem à mesma entidade e devem ser fundidas com o objetivo de consolidar suas características em uma única descrição de entidade. Se o RESOLVE faz fusões em demasia terá um baixo índice de *recall*; enquanto que, se este for muito passivo, o índice de *precision* será baixo. As decisões do RESOLVE são baseadas em uma árvore de decisão induzida a partir de representações vetoriais de características de frases nominais. Algumas características utilizadas nessas representações são independentes de domínio, e outras são de domínios específicos [FIS 95].

A.2.4.1.1 CRYSTAL e BADGER

Uma das principais fontes de conhecimento de um sistema de extração de informações é o dicionário de padrões lingüísticos utilizado para identificar informações relevantes no documento. A criação automática desses dicionários conceituais auxilia na portabilidade e escalabilidade do sistema [SOD 95a]. O CRYSTAL é um sistema que automaticamente induz um dicionário de definições de NC para identificar informações relevantes a partir de documentos de treino. Cada uma dessas definições é generalizada o máximo possível sem produzir erros, de modo que um número mínimo de entradas do dicionário cubra todas as possíveis instâncias relevantes de treino. Testando a precisão de cada definição proposta, o CRYSTAL freqüentemente obtém resultados melhores que os gerados manualmente ao criar regras de extração confiáveis [SOD 95].

O dicionário de definições de NC descreve a sintaxe local e o contexto semântico onde a informação relevante provavelmente se encontra. A informação extraída dos documentos pelo analisador de sentenças BADGER é representada como instâncias dos NC definidos. Antes das definições de NC serem aplicadas, o BADGER segmenta o documento para identificar componentes sintáticos: sujeitos, verbos, objetos diretos e indiretos e frases preposicionais. Além disso, procura a classe semântica de cada palavra em um dicionário léxico de domínio específico. A definição de um NC especifica um conjunto de restrições sintáticas e semânticas a serem satisfeitas para que ela seja aplicada a um segmento do documento. Por exemplo, quando a tarefa é analisar relatórios hospitalares e identificar referências a “diagnósticos” e “sinal ou sintoma”, as palavras a seguir também são consideradas importantes indícios para a obtenção da informação desejada, sendo classificadas como subtipos das definições iniciais:

- Diagnóstico: confirmado, suspeito, pré-existente e passado.
- Sinal ou Sintoma: presente, ausente, presumido, desconhecido e histórico.

O exemplo abaixo, Figura A.8, define um NC para o domínio médico que identifica referências a sintomas ausentes. Essa definição extrai o objeto direto de uma sentença quando o sujeito dela apresenta a palavra “paciente”, ou seja, uma palavra que pertença à classe semântica <Paciente ou Grupo Desabilitado>. Além disso, o verbo da sentença deve ser o “negar” em voz ativa, e o objeto direto deve possuir uma palavra da classe semântica <Sinal ou Sintoma>. Essa definição de NC extrairia o trecho de texto “qualquer episódio de náusea” da sentença “O paciente nega qualquer episódio de

náusea". Não seria possível aplicar essa definição para a sentença "O paciente nega histórico de asma", uma vez que "asma" pertence à classe semântica <Doença ou Síndrome>, que não é uma subclasse de <Sinal ou Sintoma> [SOD 95a].

Tipo NC: Sinal ou Sintoma
Subtipo: Ausente
Extraído do Objeto Direto
Verbo em Voz Ativa

Restrições do Sujeito:
Palavras incluem "PACIENTE"
Classe Semântica: <Paciente ou Grupo Desabilitado>

Restrições do Verbo:
Palavras incluem "NEGAR"

Restrições do Objeto Direto:
Classe Semântica: <Sinal ou Sintoma>

FIGURA A.8 - NC para Identificar "Sinal ou Sintoma Ausente"

Um dicionário de definições de NC para o domínio médico é específico para a semântica e estilo de relatórios hospitalares e não poderia ser transferido para outras aplicações. Um novo dicionário deve ser construído para cada aplicação. As definições devem ser gerais o suficiente para cobrir instâncias não vistas anteriormente, mas, ao mesmo tempo, restritas o suficiente para evitar uma generalização demasiada de instâncias que não contém informações relevantes [SOD 95a]. Uma ferramenta que automaticamente gera o dicionário é necessária para assegurar que o BADGER possa ser facilmente transportado para novos domínios de aplicação. Neste sentido, observa-se a importância do sistema CRYSTAL, uma ferramenta de indução de dicionários, criando automaticamente um dicionário de NC a partir de um conjunto de documentos para treinamento do sistema [SOD 95a].

A.2.4.1.2 Nodos Conceito - NC

Um NC é uma estrutura instanciada pelo analisador de sentenças BADGER para representar informação relevante identificada em documentos. Também pode ser visto como uma regra, que possui várias restrições [SOD 95a]. O NC tem dois campos fixos, tipo e subtipo, bem como campos para armazenar a informação extraída, que normalmente são frases nominais do documento. Um NC é instanciado de um segmento de texto quando as restrições da definição do NC são satisfeitas. Essas restrições operam nos principais constituintes sintáticos: sujeitos, verbos, objetos direto ou indireto e frases preposicionais. Qualquer um desses constituintes pode ser testado: uma seqüência de palavras específica e a existência de determinadas classes semânticas no sujeito principal da frase, ou nos objetos (direto ou indireto). O verbo pode ser selecionado quanto à voz ativa ou passiva [SOD 95a].

A Figura A.9 mostra uma definição de NC que identifica diagnósticos pré-existentes, conforme restrições do seguinte tipo: "...foi diagnosticado com recorrência de <Doença ou Síndrome> no <Parte do Corpo ou Órgão>". Nesse caso, a informação a ser extraída deve encontrar-se em uma frase preposicional com a preposição "com" e as palavras "recorrência de". Além disso, a frase deve ter um substantivo principal cuja classe semântica seja <Doença ou Síndrome> e um termo modificado cuja classe seja <Parte do Corpo ou Órgão>. Essa definição de NC aplica-se com sucesso a seguinte sentença: "O paciente foi diagnosticado com recorrência de câncer na laringe". Desde que não existam restrições quanto ao sujeito, o segmento de texto pode ter qualquer sujeito, incluindo um pronome relativo ou sujeito oculto [SOD 95a].

Tipo de NC: Diagnóstico
Subtipo: Pré-existente
Extraído de Frase Preposicional “COM”
Verbo em Voz Passiva

Restrições do Verbo:
Palavras incluem "DIAGNOSTICADO"

Restrições da Frase Preposicional:
Preposição = "COM"
Palavras incluem "RECORRÊNCIA DE"
Classe Semântica <Doença ou Síndrome>
Classe Semântica <Parte do Corpo ou Órgão>

FIGURA A.9 - Uma Definição de NC para “Diagnóstico Pré-Existente”

Em alguns documentos a recorrência de uma doença seja o principal diagnóstico da hospitalização corrente de um paciente e deva ser identificada como “diagnóstico confirmado”; ou talvez seja a condição de que não existe mais doença, e essa deva ser identificada como “diagnóstico passado”. Nesse último caso, a definição de NC produzirá um erro de extração. Por outro lado, essa definição pode ser confiável, mas perde alguns exemplos válidos que cobriria caso as restrições fossem ligeiramente alteradas. No entanto, julgar o quanto alterar as restrições de definição de um NC é difícil de ser feito *a priori.* Isto requer uma cuidadosa consideração por alguém que, ao mesmo tempo, seja especialista no domínio e tenha profundo conhecimento do analisador de sentenças BADGER. Um modo alternativo de manualmente tratar as definições NC é induzi-las automaticamente a partir documentos de treino escolhidos por um especialista no domínio [SOD 95a].

A.2.4.1.3 Criando Dicionários no CRYSTAL

O CRYSTAL deriva dicionários de definições de NC a partir de um conjunto de documentos de treino. Inicialmente, ele cria o novo dicionário com uma definição de NC para cada segmento de texto manualmente marcado como relevante [SOD 95a]. Após, as restrições das definições iniciais são gradualmente relaxadas, ampliando sua cobertura enquanto são combinadas as definições similares para formar um dicionário mais compacto. As definições finais de NC são as mais genéricas possíveis, mas de forma a não produzirem erros de extração quando aplicadas sobre o próprio corpo de documentos de treino [SOD 95a].

O primeiro passo na criação de um dicionário é a escolha do conjunto de documentos de treino por um especialista no domínio. Após, cada frase com informação a ser extraída é marcada usando SGML para definir o tipo e subtipo de NC. Conforme o exemplo anterior, no domínio médico, três enfermeiras, sob supervisão de um médico, escolheram e marcaram os documentos de treino. Esses documentos foram então segmentados pelo analisador de sentenças BADGER, conforme a marcação, criando um conjunto de instâncias de NC de treino [SOD 95a]. A partir de cada instância, o CRYSTAL iniciou a indução do dicionário de definições de NC. Se uma instância de treino é do tipo <Diagnóstico> e subtipo <Pré-existente>, uma definição inicial de NC é criada para extrair o seguimento de texto que originou a instância como um diagnóstico pré-existente. As restrições na definição inicial dos NC são derivadas das palavras e classes encontradas na instância exemplo [SOD 95a].

Antes do processo de indução iniciar, o CRYSTAL não pode prever quais características de uma instância são essenciais para uma definição de NC e quais características são meramente acidentais. São codificados todos os detalhes do segmento de texto como restrições na definição inicial de NC, requerendo a exata seqüência de palavras e classes semânticas em cada campo sintático do NC [SOD 95a].

Posteriormente, o CRYSTAL irá induzir quais restrições devem ser relaxadas. A Figura A.10 mostra a definição inicial de NC derivada a partir do segmento de texto “Sem problemas com a exceção de moderada deficiência de respiração e crônico inchaço nos tornozelos.” O especialista no domínio marcou “deficiência de respiração” e “inchaço nos tornozelos” como do tipo de NC <Sinal ou Sintoma> e subtipo <Presente>. Quando o BADGER analisa a sentença marcada, relaciona a frase nominal “a exceção de moderada deficiência de respiração e crônico inchaço nos tornozelos.” no campo de frase preposicional. Quando uma frase nominal possui múltiplos substantivos principais ou múltiplos modificadores, a restrição de classe torna-se uma restrição conjuntiva. Restrições de classes em palavras como “sem problemas”, da classe <Classe Raiz>, são abandonadas como desprovidas de conteúdo semântico [SOD 95a].

Tipo de NC: Sinal ou Sintoma
Subtipo: Presente
Extraído da Frase Preposicional “COM”
Verbo = <NULL>

Restrições do Sujeito:
Palavras incluem “Sem problemas”

Restrições da Frase Preposicional:
Preposição = “COM”
Palavras incluem “a exceção de moderada deficiência de respiração e crônico inchaço nos tornozelos.”
Classe Semântica <Sinal ou Sintoma>
Classe Semântica <Sinal ou Sintoma>, <Parte ou Região do Corpo>

FIGURA A.10 - Definição Inicial de NC com as Palavras Exatas da Instância de Treino

É improvável que essa definição de NC seja aplicada para um fragmento de texto diferente, mas é garantido que funcionará para a sentença original. As definições iniciais são muito restritas para serem úteis até que o CRYSTAL relaxe algumas restrições. Restrições de palavras exatas são relaxadas mantendo-se apenas as palavras comuns a ambas restrições originais, ou abandonando a restrição. Restrições semânticas são relaxadas através de um movimento de ascensão na hierarquia semântica para encontrar uma classe ancestral comum às restrições originais, ou, caso atinjam a raiz da hierarquia semântica, abandonando a restrição [SOD 95a]. O número de combinações para relaxar as restrições de uma definição pode ser muito grande, existem mais de 57.000 generalizações possíveis da definição inicial da Figura A.10 [SOD 95a]. O algoritmo de generalização do CRYSTAL é descrito a seguir:

```
Inicializar Dicionário e Banco de Instâncias de Treino
DO UNTIL não existir mais definições iniciais de NC no dicionário
        D = uma definição inicial de NC removida do dicionário
        LOOP
                D' = definição de NC mais similar a D
                IF D' = NULL
                        EXIT LOOP
                U = a unificação de D e D'
                Testa a cobertura de U nas instâncias de treino
                IF indício de erro de U > tolerância
                        EXIT LOOP
                Apagar todas definições de NC cobertas por U
                SET D = U
                Adicionar D para o dicionário
                RETURN dicionário
        END LOOP
END UNTIL
```

O CRYSTAL induz generalizações úteis de definições iniciais de NC comparando e localizando definições similares. Seja D a definição sendo generalizada, e D’ uma segunda definição similar a D, conforme uma métrica de similaridade baseada no número de “relaxamentos” requeridos para unificar duas definições, uma nova definição U é criada com restrições suficientemente relaxadas para unificar D e D’

[SOD 95a]. A nova definição é então testada contra os documentos de treino para ter-se certeza de que não extrai frases não marcadas com o tipo e subtipo de NC sendo induzido. Confirmando U como uma definição de NC válida, apagam-se as definições cobertas por ela, reduzindo o tamanho do dicionário: D e D'. A definição U torna-se a corrente, e o processo é repetido, utilizando definições similares a ela para guiar o futuro relaxamento de restrições. Eventualmente, atinge-se um ponto de relaxamento que produzirá uma definição que excede a tolerância pré-especificada de erro de extração. Nesse ponto, o CRYSTAL começa o processo de generalização a partir de outra definição inicial de NC, até que todas as definições iniciais tenham sido analisadas.

Ao invés de reprocessar os documentos de treino cada vez que testa uma definição de NC proposta, o CRYSTAL usa o analisador de sentenças BADGER para segmentar os documentos e criar uma base de segmentos de treino. Essa base inclui todos os segmentos dos documentos, não apenas os marcados como relevantes. Se uma definição de NC satisfaz todas as restrições das instâncias que a originaram, mas extrai uma frase que não foi marcada com o tipo e subtipo definido nesse NC, é contabilizado um erro para essa definição [SOD 95a]. O CRYSTAL aceita uma limitada quantidade de erro nos resultados de treino, conforme um parâmetro de tolerância de erro. Não se considera ruim uma definição com um único erro de extração. Isso aumenta a robustez do dicionário, o que é necessário quando se trata de texto irrestrito. O CRYSTAL é um aperfeiçoamento das tentativas anteriores, como o AutoSlog, de derivar regras de análise de textos a partir de documentos de treino. O objetivo é encontrar um conjunto mínimo de definições de NC que cubra todas as instâncias de treino e testar cada definição proposta contra os documentos de teste para assegurar que o índice de erro esteja dentro da tolerância pré-definida. O parâmetro tolerância de erro do CRYSTAL permite ao usuário manipular a oscilação de *recall-precision* [SOD 95a].

A.2.4.1.4 WRAP-UP

O módulo WRAP-UP realiza a análise de discurso no UMASS, relacionando as informações individuais anteriormente extraídas pelo BADGER. A definição do domínio de extração, além de determinar quais objetos são relevantes, define quais relações entre esses objetos são de interesse do usuário. Cada objeto relevante encontrado em um documento é representado por uma instância de NC, e as relações entre eles indicadas por ponteiros de referência entre as instâncias [SOD 95b]. O WRAP-UP usa técnicas de *Machine Learning* para construir automaticamente o conjunto de classificadores necessários para a análise de discurso em um domínio específico. Um classificador é uma estrutura de informação que representa uma relação entre NC, contendo atributos com informações sobre ela e ponteiros para os NC relacionados. Essas estruturas são criadas a partir de árvores de decisão que analisam os documentos e confirmam ou não a existência de relações entre objetos. Caso sim, uma estrutura classificadora é criada para representar a relação. O usuário fornece a definição dos NC, as relações de seu interesse e os documentos de treino com as informações e relações relevantes marcadas manualmente, e o WRAP-UP gera as árvores de decisão sem nenhuma outra codificação manual, ajustando o sistema para um novo domínio.

Para entender o processamento envolvido na análise de discurso, considere o documento sobre microeletrônica da Figura A.11. As informações relevantes nesse domínio são os processos de fabricação, bem como companhias, equipamentos e dispositivos associados a esses processos. O classificador referente a processos de fabricação pode apontar para os de equipamentos utilizados no processo e para os de

dispositivos de manufatura. O classificador de equipamento pode apontar para os de fabricantes e para os de módulos dos equipamentos. Existem quatro possíveis relações entre processo de fabricação e companhia fabricante: desenvolvedor, fabricante, distribuidor e comprador/usuário, com múltiplos papéis possíveis para cada companhia. Os classificadores do documento exemplo são o processo *"x-ray lithography"* que usa o equipamento *"stepper"* para fazer dois tipos de dispositivos: *"microprocessors"* e *"memory devices"*. O *"stepper"* possui um submódulo de *"x-ray source"*. Embora duas companhias e uma agência do governo sejam mencionadas no documento, apenas a "Hampshire Instruments Inc." tem um papel importante no processo de *"x-ray lithography"*. Portanto, a "IBM" e o *"Defense Advanced Research Projects Agency's National"* não são relevantes e devem ser ignorados pelo WRAP-UP. O resultado desejado da análise de discurso do documento exemplo esta na Figura A.12. O classificador *"microelectronics-capability"* é criado para mostrar as relações entre companhias e processos, tendo a "Hampshire Instruments Inc." como desenvolvedora do processo *"x-ray lithography"*. Esse processo aponta para *"stepper"*, que aponta para o equipamento *"x-ray source"* e de volta para "Hampshire Instruments Inc.". O *"x-ray lithography"* também aponta para outros dois dispositivos.

IBM's Systems Integration division has awarded Hampshire Instruments Inc. a subcontract for x-ray mask making and wafer exposure under the Defense Advanced Research Projects Agency's National: X-ray Lithography Program. Under the contract, Hampshire will produce gold-on-silicon photo masks from data provided by IBM, print test wafers from the masks in its Series 5000 wafer stepper, using a laser based soft x-ray source. Test patterns for the contract include 0.5-micron features from microprocessor and memory devices.

FIGURA A.11 - Exemplo de um Texto do Domínio da Microeletrônica

FIGURA A.12 - Resultado da Análise do Documento de Microeletrônica

O WRAP-UP é independente de domínio, sendo treinado para gerenciar a análise de discurso em um domínio específico. Durante a sua fase de treinamento, o analisador de discurso deriva várias árvores de decisão ID3 [QUI 2000] usando documentos de treino e os correspondentes resultados desejados gerados manualmente. O ID3 é um algoritmo de aprendizado supervisionado e requer um conjunto de classificadores de treino, corretamente gerados para cada relação extraída dos documentos de treino. Esse módulo inicia a etapa de construção de árvores, passando cada documentos para o analisador de sentença BADGER, que cria um conjunto NC representando a informação extraída. O WRAP-UP processa os NC e os resultados

desejados gerados manualmente e constrói as árvores para a primeira etapa (filtragem), então utiliza tais árvores para construir árvores para a segunda etapa, e assim por diante, até as árvores terem sido construídas para todas as seis etapas.

Para iniciar seu processamento, o WRAP-UP recebe as instâncias de NC geradas pelo analisador de sentenças BADGER. Ele deve descartar informações erroneamente identificadas durante a análise de sentenças, combinar NC coreferenciais e determinar relações entre objetos. O algoritmo central do WRAP-UP consiste de seis etapas, cada uma com o seu conjunto de classificadores para guiar o processamento [SOD 95b]:

1. Filtrar a informação extraída erroneamente.
2. Combinar atributos coreferenciais de objetos.
3. Relacionar objetos.
4. Dividir objetos com múltiplos relacionamentos (ponteiros).
5. Inferir objetos ausentes.
6. Adicionar valores *default* para os campos sem valor.

O WRAP-UP automaticamente determina as árvores de decisão e, conseqüentemente, os classificadores resultantes para cada etapa a partir da especificação de saída do domínio, a qual consiste em uma lista de objetos e possíveis relações entre eles. A primeira etapa, filtragem, possui um classificador para cada campo de cada NC do domínio. Esses classificadores julgam a validade das informações armazenadas nesses campos, descartando as irrelevantes, se a árvore de decisão retorna "negativo". A segunda, combinação, cria um classificador para cada par de NC do mesmo tipo. Se a árvore de decisão confirma que ambos NC podem ser mapeados para o mesmo classificador, ele é mantido. A etapa de relacionamento possui um classificador para cada relação possível conforme as especificações de saída definidas pelo usuário. Por exemplo, o BADGER identifica o objeto "*x-ray lithography*", bem como dois equipamentos: o "*stepper*" e o "*x-ray source*". Uma das decisões na etapa de relacionamento é quando adicionar um ponteiro do processo "*x-ray lithography*" para um ou ambos equipamentos. O WRAP-UP cria um classificador separado para cada possível relacionamento, um para "*x-ray lithography*" e "*x-ray source*", e outro para "*x-ray lithography*" e "*stepper*". Cada classificador é passado para uma árvore de decisão "*Lithography-Equipment-Link*", e um ponteiro é adicionado se a árvore retorna "positivo". Após as etapas de combinação e relacionamento, o WRAP-UP possui uma etapa que considera classificadores com múltiplos ponteiros. Em alguns casos, o classificador é dividido em múltiplas cópias, cada uma com um ponteiro simples. A próxima etapa considera classificadores "órfãos"; ou seja, não apontados por nenhum outro objeto, podendo inferir um relacionamento (que antes não existia) de um objeto para um órfão. Árvores para essa etapa retornam uma classificação especificando o tipo de objeto a inferir. A última etapa adiciona valores *default*, sensíveis ao contexto, para atributos vazios de classificadores, como, por exemplo, um *status* "em uso" ou "em desenvolvimento" para os classificadores do tipo equipamento.

Quando o WRAP-UP cria um classificador para um objeto ou par de objetos, ele processa o máximo de informação passada para ele pelo analisador de sentenças. Cada NC extraído pelo BADGER possui um classificador com um campo para cada atributo do NC transferido para o WRAP-UP, tal como tipo-equipamento ou nome-equipamento. Esse analisador também passa a posição de cada referência ao objeto no documento e as definições de NC utilizadas para identificar aquela referência. Os classificadores apresentam campos de informações, alguns indicando valores de atributos de objetos, alguns expressando a posição relativa no documento das referências mais próximas para

os objetos relacionados, e outros mostrando os padrões de definição de NC para cada referência.

A.2.4.1.5 RESOLVE

Um dos desafios em EI é determinar qual referência diz respeito a qual objeto. Esse problema pode ser definido como de classificação: dadas duas referências, elas aludem ao mesmo objeto ou a objetos diferentes? O módulo RESOLVE classifica pares de frases como coreferentes ou não. Os erros gerados pelo analisador de sentenças BADGER na extração de informações coreferenciais, criando instâncias distintas de NC para o mesmo objeto, são eliminados pelo *Coreference Marking Interface* (CMI), sub-módulo do RESOLVE, que extrai coreferências de documentos. Para minimizar as dificuldades de criação e manutenção do conjunto complexo de regras utilizado pelo CMI, o RESOLVE vale-se de uma árvore de decisão para determinar a ordem e o peso relativo de diferentes evidências de coreferências especificadas nas regras [MCC 95]. Resultados mostram que o uso de árvores apresenta melhor performance que somente o de regras para a tarefa de CO do MUC [MCC 95].

A estrutura de dados utilizada no processamento de coreferências é o *token*, que converte os NCs de saída do analisador de sentença BADGER em uma representação mais independente do sistema [FIS 95]. Antes do processamento de coreferência, cada *token* contém uma frase nominal, um ou mais padrões léxicos circunstanciais, identificadores gramaticais, características semânticas e informação deduzida da frase ou contexto no qual essa foi encontrada. A informação deduzida inclui o tipo de objeto referenciado pela frase, qualquer parte de palavra de nome ou lugar contida na frase, e alguma informação de domínio específico.

O módulo de coreferência foi projetado para minimizar falsos positivos, isto é, declarar duas frases como coreferenciais quando elas não são. Essa decisão de projeto baseou-se no fato que falsos positivos aparentam ser mais maléficos à performance do sistema do que falsos negativos. Essa abordagem conservadora é compartilhada pela maioria dos participantes do MUC [MCC 95]. As regras utilizadas para determinar quando duas frases são coreferenciais são mostradas na Figura A.13. Visando minimizar falsos positivos, sempre que nenhuma das regras é disparada, o sistema classifica o par de frases como não coreferente. Uma das muitas dificuldades no desenvolvimento do conjunto de regras estava em ordenar as regras. Essa dificuldade motivou o uso da abordagem de *Machine Learning*. O objetivo era um sistema que pudesse aprender como combinar as evidências positivas e negativas descritas nas regras usando árvores de decisão.

IF both tokens come from the same trigger family
THEN they are not coreferent.
IF each token comes from a different partition
THEN they are not coreferent.
IF both tokens contain a common phrase
THEN they are coreferent.
IF both tokens refer to joint ventures
THEN they are coreferent.
IF both tokens contain the same company name
THEN they are coreferent.
IF one token contains an alias of the other
THEN they are coreferent.
IF only one token refers to a joint venture
THEN they are not coreferent.
IF each token conta ins different company names
THEN they are not coreferent.

FIGURA A.13 - Regras de Correferência

Bibliografia

[APP 95] APPELT, D. et al. SRI International FASTUS System MUC-6 Test Results and Analysis. In: MESSAGE UNDERSTANDING CONFERENCE, MUC-6, 1995, Maryland, US. **Message Understanding Conference:** proceedings. San Francisco: Morgan Kaufmann, [1995].

[APT 94] APTÉ, C. et al. Automated Learning of Decision Rules for Text Categorization. **ACM Transactions on Information Systems**, New York, v.12, n.3, p. 233-251, 1994.

[BEC 97] BECKER, C.; DEITOS, H. R. **SIRI – Sistema Inteligente de Recuperação de Informações**, 1997. 79p. Trabalho de Conclusão (Bacharelado em Informática) - Instituto de Informática, PUCRS, Porto Alegre.

[BEC 97a] BECKER, C.; DEITOS, H. R. SIRI – Sistema Inteligente de Recuperação de Informações. In: JORNADA DE INICIAÇÃO CIENTÍFICA DA PUCRS, 1., 1997, Porto Alegre, Rio Grande do Sul. **Anais...** Porto Alegre: Instituto de Informática da PUCRS, 1997.

[BOR 96] BORBA, P. **Critérios para Modularidade.** 1996. Material de Aula, Departamento de Informática, UFPE, Recife. Disponível em: <http://www.di.ufpe.br/~java/graduacao961/aulas/aula4/criterios.html>. Acesso em: jan. 2001.

[BOR 97] BORLAND INTERNATIONAL. **Borland Delphi 3 Object Pacal Language Guide**. Scotts Valley, CA, USA, 1997.

[BOR 97a] BORLAND INTERNATIONAL. **Borland Delphi 3 User's Guide**. Scotts Valley, CA, USA, 1997.

[BOR 97b] BORLAND INTERNATIONAL. **Borland Delphi 3 Visual Component Library Reference**. Scotts Valley, CA, USA, 1997. 2v.

[BOW 94] BOWMAN, C. M. et al. Scalable Internet Resource Discovery: Research Problems and Approaches. **Communications of the ACM**, New York, v. 37, n. 8, p. 98-107, 1994. Disponível em: <citeseer.nj.nec.com/bowman94scalable.html>. Acesso em: fev. 2000.

[CAN 96] CANTU, M. **Dominando o Delphi**. São Paulo: Makron Books, 1996.

[CAR 97] CARDIE, C. Empirical Methods in Information Extraction. **AI Magazine**, Menlo Park, v. 18, n. 4, p. 65-79, 1997. Disponível em: <http://citeseer.nj.nec.com/cardie97empirical.html>. Acesso em: mar. 2000.

[CAR 98] CARDIE, C.; PIERCE, D. **Proposal for an Interactive Environment for Information Extraction.** Ithaca, NY: Department of Computer Science, Cornell University, 1998. Disponível em: <citeseer.nj.nec.com/528993.html>. Acesso em: maio 2000.

[CEI 2001] CURSO DE ESPECIALIZAÇÃO EM INTELIGÊNCIA COMPETITIVA, 5., 2001. **[Anais...]**. Disponível em: <http://www.ceic.com.br>. Acesso em: jan. 2001.

[CHI 97] CHINCHOR, N. A. Overview of MUC-7/MET-2. In: MESSAGE UNDERSTANDING CONFERENCE, MUC-7, 1997, San Diego, US. **Message Understanding Conference:** proceedings. San Diego: Science Applications International Corporation, [1998]. Disponível em: <http://www.itl.nist.gov/iaui/894.02/related_projects/muc/index.html>. Acesso em: ago. 1999.

[COS 96] COSTANTINO, M.; MORGAN, R. G.; COLLINGHAM, R. J. Financial Information Extraction Using Pre-defined and User-definable Templates in the LOLITA. **CIT - Journal of Computing and Information Technology**, [S.l.], v. 4, n. 4, p. 241-255, 1996. Disponível em: <citeseer.nj.nec.com/costantino97financial.html>. Acesso em: nov. 1999.

[COS 97] COSTANTINO, M. **Financial Information Extraction Using Pre-defined and User-definable Templates in the LOLITA**, 1997. Ph. D. Thesis, Department of Computer Science, University of Durham. Disponível em: <http://www.advanced-finance.co.uk>. Acesso em: 1998.

[COS 97a] COSTANTINO, M. et al. Natural Language Processing and Information Extraction: Qualitative Analysis of Financial News Articles In: CONFERENCE ON COMPUTATIONAL INTELLIGENCE FOR FINANCIAL ENGINEERING, CIFEr, 1997, New York, USA. **Proceedings…** Disponível em: <http://www.advanced-finance.co.uk>. Acesso em: 1998.

[COW 96] COWIE, J.; LEHNERT, W. Information Extraction. **Communications of the ACM**, New York, v. 39. n. 1, p. 80-91, Jan. 1996.

[DAL 2000] DALE, R.; MOISL, H.; SOMERS, H. **A Handbook of Natural Language Processing**. New York: Marcel Dekker Inc, 2000.

[DIX 97] DIXON, M. **An Overview of Document Mining Technology.** 1997. Disponível em: <citeseer.nj.nec.com/dixon97overview.html>. Acesso em: out. 1997.

[DVO 96] DVORAK, J. C. Uma Explosão Chamada Internet. **Informática Exame**, São Paulo, v. 11, n. 126, p. 13, set. 1996.

[EDE 94] EDELWEISS, N.; OLIVEIRA, J. M. P.; PERNICI, B. An Object-Oriented Approach to a Temporal Query Language. In: DEXA CONFERENCE, 1994, Greece. **Proceedings...** Berlin: Springer-Verlag, 1994. p. 223-235. (Lecture Notes in Computer Science, n. 856).

[ELE 2002] ELECTRONIC Trading Opportunities (ETO) System. [S.l.]: United Nations Trade Point Development Center, UNTPDC. Disponível em: <http://www.wtpfed.org>. Acesso em: 2002.

[EMT 92] EMTAGE, A.; DEUTSCH, P. Archie - An Eletronic Directory Service for the Internet. In: USENIX WINTER CONFERENCE, 1992, San Fracisco, California. **Proceedings...** Montréal: McGill, 1992. p. 93-110.

[FEL 95] FELDMAN, R.; DAGAN, I. Knowledge Discovery in Textual Databases (KDT). In: INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY, KDD, 1., 1995, Montreal, Canada. **Proceedings…** [S.l.:s.n.], 1995. p. 112-117. Disponível em: < citeseer.nj.nec.com/feldman95knowledge.html>. Acesso em: set. 1999.

[FEL 99] FELDENS, M. A. **Knowledge Discovery in Databases.** Disponível em: <http://www.inf.ufrgs.br/~feldens/ucpel.zip> Acesso em: nov. 1999.

[FER 94] FERNANDEZ, L. F. N. **SDIP:** um Ambiente Inteligente para a Localização de Informações na Internet. 1994. 238 p. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

[FIS 95] FISHER, D.; SODERLAND S.; MCCARTHY, J. et al. Description of the UMass System as Used for MUC-6. In: MESSAGE UNDERSTANDING CONFERENCE, MUC-6, 1995, Maryland, US. **Message Understanding Conference**: proceedings... San Francisco: Morgan Kaufmann, [1995].

[FRA 93] FRAINER, A. S. **Planos na Interação Homem-Máquina.** 1993. Dissertação Mestrado. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

[FU 95] FU, Y.; HAN, J. Meta-Rule-Guided Mining of Association Rules in Relational Databases. In: INT. WORKSHOP ON INTEGRATION OF KNOWLEDGE DISCOVERY WITH DEDUCTIVE AND OBJECT-ORIENTED DATABASES, KDOOD, 1995, Singapore. **Proceedings…** [S.l.:s.n.], 1995. p. 39-46. Disponível em: < http://citeseer.nj.nec.com/fu95metaruleguided.html>. Acesso em: 1997.

[FU 96] FU, Y. **Discovery of Multiple-Level Rules From Large Databases**. 1996. 197 p. Thesis (Ph. D. Thesis in Computer Science), Simon Fraser University, B.C. Canada. Disponível em: <http://db.cs.sfu.ca/sections/publication/theses/theses.html>. Acesso em: 1997.

[GAI 95] GAIZAUSKAS, R. et al. University of Sheffield: Description of the LaSIE system as used for MUC-6. In: MESSAGE UNDERSTANDING CONFERENCE, MUC-6, 1995, Maryland, US. **Message Understanding Conference**: proceedings... San Francisco: Morgan Kaufmann, [1995].

[GAR 99] GARIGLIANO, R.; URBANOWICZ, A.; NETTLETON, D. J. University of Durham: Description of the LOLITA system as used for MUC-7. In: MESSAGE UNDERSTANDING CONFERENCE, MUC-7, 1997, San Diego, US. **Message Understanding Conference:** proceedings. San Diego: Science Applications International Corporation, [1998]. Disponível em: < http://www.itl.nist.gov/iaui/894.02/related_projects/muc/index.html>. Acesso em: 1999.

[GEN 86] GENARO, S. **Sistemas Especialistas:** o Conhecimento Artificial. Rio de Janeiro: LTC, 1986. 192 p.

[GIF 91] GIFFORD, D. R. et al. Semantic File System. In: ACM SYMPOSIUM ON OPERATING SYSTEMS PRINCIPLES, 30., 1991. **Proceedings...** [S.l.:s.n.], 1991. p. 16-25, Disponível em: <http://citeseer.nj.nec.com/gifford91semantic.html>. Acesso em 1996.

[GRA 97] GRAHL, E. A. **Página da Disciplina de Engenharia de Software - Reutilização de Software**. Disponível em: <http://www.furb.rct-sc.br/~egrahl>. Acesso em: 1997.

[GRI 95] GRISHMAN, R. Design of the MUC-6 Evaluation. In: MESSAGE UNDERSTANDING CONFERENCE, MUC-6, 1995, Maryland, US. **Message Understanding Conference**: proceedings... San Francisco: Morgan Kaufmann, [1995].

[GRI 97] GRISHMAN, R. Information Extraction: Techniques and Challenges. In: INTERNATIONAL SUMMER SCHOOL ON INFORMATION EXTRACTION, 1997, Frascati, IT. **Information Extraction:** a Multidisciplinary Approach to an Emerging Information Technology. Berlin: Springer-Verlag, 1997. p. 10-27

[HAL 2000] HALMENSCHLAGER, D. **SEIE – Sistema de Extração de Informações de E-Mails**, 2000. 44 p. Trabalho de Conclusão (Bacharelado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

[HAN 95] HAN, J.; FU, Y. Discobery of Multiple-Level Association Rules from Large Databases. In: INT. CONFERENCE ON VERY LARGE DATABASES, 21., Zurich, Switzerland, 1995. **Proceedings...** Disponível em: <http://citeseer.nj.nec.com/han95discovery.html>. Acesso em: 1997.

[HAN 95a] HAN, J. Mining Knowledge at Multiple Concept Levels. In: INT. CONF. ON INFORMATION AND KNOWLEDGE MANAGEMENT, CIKM, 4., 1995, Baltimore, Maryland. **Proceedings…** [S.l.:s.n.], 1995. p. 19-24. Disponível em: <http://db.cs.sfu.ca/sections/publication/kdd/kdd.html>. Acesso em: 1996.

[HAN 95b] HAN, J.; ZAÏANE, O. R; FU, Y. Resource and Knowledge Discovery in Global Information Systems: a Multiple Layered Database Approach. In: CONFERENCE ON ADVANCES IN DIGITAL LIBRARIES, 1995, Washington, DC. **Proceedings...** Disponível em: <http://citeseer.nj.nec.com/han95resource.html>. Acesso em: 1996.

[HAN 96] HAN, J. et al. DMQL: A Data Mining Query Language for Relational Databases. In: SIGMOD WORKSHOP ON RESEARCH ISSUES ON DATA MINING AND KNOWLEDGE DISCOVERY, DMKD, 1996, Montreal, Canada. **Proceedings...** Disponível em: <http://db.cs.sfu.ca/sections/publication/kdd/kdd.html>. Acesso em: 1996.

[HAR 93] HARDY, D. R.; SCHWARTZ, M. F. Essence: a Resource Discovery System Based on Semantic File Indexing. In: USENIX WINTER CONFERENCE, 1993, San Diego, California. **Proceedings...** Boulder: University of Colorado, 1993. p. 361-374. Disponível em: <http://citeseer.nj.nec.com/hardy93essence.html>. Acesso em: 1996.

[HAR 95] HARDY, D. R.; SCHWARTZ, M. F. Customized Information Extraction as Basis for Resource Discovery. **ACM Transactions on Computer Systems**, New York, v. 14, n. 2, p. 171-199, 1996. Disponível em: <http://citeseer.nj.nec.com/hardy94customized.html>. Acesso em: 1996.

[HED 95] HEDBERG, S. R. The Data Gold Rush. **Byte**, Peterborough, N. H., v. 20, n. 10, p. 83-88, Oct. 1995.

[HOB 96] HOBBS, J. R. et al. FASTUS: a Cascaded Finite-State Transducer for Extracting Information from Natural-Language Text. In: FINITE STATE DEVICES FOR NATURAL LANGUAGE PROCESSING, 1996. **Proceedings...** Cambridge, MA: MIT Press, 1996. Disponível em: <http://citeseer.nj.nec.com/hobbs96fastus.html>. Acesso em: 1996.

[HOB 99] HOBBS, J. R. et al. **FASTUS - Extracting Information from Real-World Texts.** Disponível em: <http://www.ai.sri.com/natural-language/projects/fastus.html>. Acesso em: 2000.

[HOB 2001] HOBBS, J. R. **Generic Information Extraction System**. 2001. Página do Artificial Intelligence Center SRI International. Disponível em: <http://www.itl.nist.gov/iaui/894.02/related_projects/tipster/gen_ie.htm >. Acesso em: 2001.

[HOW 88] HOWARD, J. et al. Scale and Performance in Distribute File Systems. **ACM Transactions on Computer Systems**, New York, v. 6. n. 1, p. 51-81, Feb. 1988.

[HUM 99] HUMPHREYS, K. et al. University of Sheffield: Description of the LaSIE-II system as used for MUC-7. In: MESSAGE UNDERSTANDING CONFERENCE, MUC-7, 1997, San Diego, US. **Message Understanding Conference:** proceedings. San Diego: Science Applications International Corporation, [1998]. Disponível em: <http://www.itl.nist.gov/iaui/894.02/related_projects/muc/index.html>. Acesso em: 1999.

[INT 2001] INTRODUCTION to Information Extraction. Disponível em: <http://www.itl.nist.gov/iaui/894.02/related_projects/muc/index.html>. Acesso em: 2001.

[KOR 97] KORFHAGE, R. R. **Information Storage and Retrieval.** EUA: John Wiley & Sons, 1997.

[KOT 2000] KOTLER, P. **Administração de Marketing**. São Paulo: Prentice Hall, 2000. 765 p.

[KOW 97] KOWALSKI, G. **Information Retrieval Systems:** Theory and Implementation. Boston: Kluwer Academic Publishers, 1997. 282 p.

[LAN 68] LANCASTER, F. W. **Information Retrivel Systems - Characteristics, Testing end Evaluation**. New York : John Wiley & Sons, 1968.

[LEH 91] LEHNERT, W. Symbolic/Subsimbolic sentence analysis: Exploiting the best of two worlds. **Advances in Connectionist and Neural Computing Theory**, Norwood, v. 1, 1991.

[LEH 94] LEHNERT, W. et al. Evaluating an Information Extraction System. **Journal of Integrated Computer-Aided Engineering**, [S.l.], v. 1, n. 6, 1994. Disponível em: <http://citeseer.nj.nec.com/lehnert94evaluating.html>. Acesso em: 1998.

[LEH 94a] LEHNERT, W.; FISHER, D. **Information Extraction**. [S.l.]: Natural Language Processing Laboratory, Computer Science Department, University of Massachusetts, 1994. Disponível em: <http://www-nlp.cs.umass.edu>. Acesso em: 1998.

[LEV 88] LEVINE, R. I.; DRANG, D. E.; EDELSON B. **Inteligência Artificial e Sistemas Especialistas**. São Paulo: McGraw-Hill, 1988.

[LEW 91] LEWIS, D. D. **Representation and Learning in Information Retrieval**. 1991. Thesis (Ph. D. thesis in Computer and Information Science) - Department of Computer and Information Science, University of Massachusetts, Amherst.

[LEW 94] LEWIS, D. D.; RINGUETTE, M. Comparison of Two Learning Algorithms for Text Categorization. In: ANNUAL SYMPOSIUM ON DOCUMENT ANALYSIS AND INFORMATION RETRIEVAL, 3., 1994, Las Vegas, US. **Proceedings...** Las Vegas: University of Nevada, 1994. p. 81-93. Disponível em: <http://www.research.att.com/~lewis/papers/lewis94b.ps>. Acesso em: 1999.

[LEW 95] LEWIS, D. D. A Brief Overview of Information Retrieval. In: IEEE AUTOMATIC SPEECH RECOGNITION WORKSHOP, 1995, Snowbird, Utah, US. **Proceedings...** Disponível em: <http://www.research.att.com/~lewis/chronobib.html>. Acesso em: 1995.

[LEW 96] LEWIS, D. D.; JONES, K. S. Natural Language Processing for Information Retrieval. **Communications of the ACM**, New York, v. 39, n. 1, p. 92-101, Jan. 1996.

[LIL 96] LILL, T. **All You Want to Know about SGML.** Waterloo, Ont., CA: A. J. Lill Consultants, 1996. Disponível em: <http://www.ajlc.waterloo.on.ca/courses/sgml.html>. Acesso em: jun. 2001.

[LOH 99] LOH, S. **Descoberta de Conhecimento em Bases de Dados Textuais.** 1999. Disponível em: <http://atlas.ucpel.tche.br/~loh/kdt_comp.htm>. Acesso em: 2000.

[LOH 99a] LOH, S. **Descoberta de Conhecimento em Textos.** 1999. 143 p. Exame de Qualificação (Doutorado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

[LOH 2000] LOH, S. et al. Concept-Based Knowledge Discovery in Texts Extracted From the WEB. **ACM SIGKDD Explorations**, Newsletter of the ACM Special Interest Group on Knowledge Discovery and Data Mining, [S.l.], v. 2, n. 1, p. 29-39, 2000. Disponível em: <http://www.acm.org/sigs/sigkdd/explorations/issue2-1/loh.pdf>. Acesso em: 2000.

[MAR 92] MARSHALL, P. WAIS: The Wide Area Information Server or Anonymous What? In: CANADIAN NET, 1992, St John's NFLD, Memorial University of Newfoundland. **Proceedings…** Ontario: University of Western Ontario, Academic Networking Computing and Communications, [1992]. Disponível em: <http://www.canarie.ca/~marshall/papers.html>. Acesso em: 1996.

[MAT 93] MATHEUS, C. J.; CHAN, P. K.; SHAPIRO, G. P. Systems for Knowledge Discovery in Database. **IEEE Transactions on Knowledge and Data Engineering**, New York, v. 5, p. 903-913, 1993. Disponível em: <http://citeseer.nj.nec.com/177052.html>. Acesso em: 1995.

[MAT 2000] MATRIX NET. **Beyond Mere Latency.** Disponível em: <http://www.matrix.net>. Acesso em: 2000.

[MCC 95] MCCARTHY, J. F.; LEHNERT, W. Using Decision Trees for Coreference Resolution. In: INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE, 40., 1995. **Proceedings...** [S.l.:s.n., 1995] p. 1050-1055. Disponível em: <http://citeseer.nj.nec.com/mccarthy95using.html>. Acesso em: 1996.

[MET 89] METZLER, D. P.; HAAS, S. W. The Constituent Object Parser: Syntactic Structure Matching for Information Retrieval. **ACM Transactions on Information Systems**, New York, v. 7, n. 3, p. 292-316, July 1989.

[MIC 82] MICHIE, D. **Introdutory Readings in Expert Systems**. New York: Gordon and Breach Science, 1982.

[MOR 95] MORGAN, R.; GARIGLIANO, R. et al. University of Durham: Description of the LOLITA System as Used in MUC-6. In: MESSAGE UNDERSTANDING CONFERENCE, MUC-6, 1995, Maryland, US. **Message Understanding Conference**: proceedings... San Francisco: Morgan Kaufmann, [1995].

[MOU 92] MOULIN, B.; ROUSSEAU, D. Automated Knowledge Acquisition from Regulatory Texts. **IEEE Expert**, Los Alamitos, v. 7, n. 2, p. 27-35, Oct. 1992.

[MUC 96] MUC-6: The Sixth in a Series of Message Understanding Conferences, 1996. Disponível em: <http://cs.nyu.edu/cs/faculty/grishman/muc6.html>. Acesso em: 2001.

[NG 97] NG, T. H.; GOH, W. B.; LOW, K. L. Feature selection, perceptron learning, and a usability case study for text categorization. In: ACM INTERNATIONAL CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL, SIGIR, 20., 1997, Philadelphia, US. **Proceedings…** New York: ACM Press, 1997. p. 67-73. Disponível em: <citeseer.nj.nec.com/ng97feature.html>. Acesso em: 2000.

[NOR 96] NORMAN, B. et al. **A Learning Subject Field Code**r. Syracuse, NY: NPAC, Syracuse University, 1996. 6 p. (Project REU'96 Report). Disponível em: <http://www.npac.syr.edu/REU/reu96/project_reu.html>. Acesso em: 2000.

[OVE 96] OVERVIEW of Information Extraction Task. Disponível em: <http://cs.nyu.edu/cs/faculty/grishman/IEtask15.book_2.html>. Acesso em: 2000.

[PEN 99] PENN Treebank Project. [S.l.]: LINC Laboratory, Computer and Information Science Department, University of Pennsylvania, 1999. Disponível em: <http://www.cis.upenn.edu/~treebank/home.html>. Acesso em: 2000.

[PER 96] PERSON, R. et al. **Usando Windows 95**. Rio de Janeiro: Campus, 1996. 1388 p.

[PET 92] PETZOLD, C. **Programando para Windows.** São Paulo: Makron Books, 1993. 1034 p.

[QUI 2000] QUINLAN, J. R. **MiniBoosting Decision Trees.** Submitted to JAIR. Disponível em: <http://citeseer.nj.nec.com/quinlan99miniboosting.html>. Acesso em: 2000.

[RAD 98] RADEV, D.; McKEOWN, K. Generating Natural Language Summaries from Multiple Online Sources. **Journal of Computational Linguistics**, [S.l.], v. 24, n. 3, p. 469-500, 1998. Disponível em: <citeseer.nj.nec.com/article/radev99generating.html>. Acesso em: 2000.

[RIC 93] RICH, E.; KNIGHT, K. **Inteligência Artificial**. São Paulo: Makron Books, 1993.

[RIJ 97] RIJSBERGEN, C. J. V. A Non-classical Logic for Information Retrieval. In: SPARCK-JONES, K.; WILLET, P. (Ed.). **Readings in Information Retrieval.** San Francisco: Morgan Kaufmann, 1997.

[RIJ 2000] RIJSBERGEN, C. J. V. **Information Retrieval**. Disponível em: < http://www.dcs.gla.ac.uk/Keith/Preface.html>. Acesso em: 2000.

[RIL 93] RILOFF, E. Automatically Constructing a Dictionary for Information Extraction Tasks. In: NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE, AAAI, 11., 1993. **Proceedigs...** Cambridge: MIT Press, 1993. p. 811-816, Disponível em: <http://citeseer.nj.nec.com/riloff93automatically.html >. Acesso em: 1996.

[RIL 94] RILOFF, E.; LEHNERT, W. Information Extraction as a Basis for High-Precision Text Classification. **ACM Transactions on Information Systems**, New York, v. 12, n. 3, p. 296-333, 1994. Disponível em: <http://citeseer.nj.nec.com/riloff94information.html>. Acesso em: 1996.

[RIL 94a] RILOFF, E. **Information Extraction as a Basis for Portable Text Classification System.** 1994. 166 p. Thesis (Ph. D. in Computer Science) - Department of Computer Science, University of Massachusetts, Amherst, MA, USA. Disponível em: < http://www.cs.utah.edu/~riloff>. Acesso em: 1999.

[RYN 2001] RYNECKI, S. **Diretor do Trade Point San Diego**. Disponível em: <http://www.tppoa.com.br/tradenews/sandiego.htm>. Acesso em: abr. 2001.

[SAL 68] SALTON, G. **Automatic Information Organization and Retrieval**. New York: McGraw-Hill, 1968.

[SAL 83] SALTON, G. **Introduction to Modern Information Retrieval**. New York: McGraw-Hill, 1983.

[SAL 87] SALTON, G.; BUCKLEY, C. **Term Weighting Approaches in Automatic Text Retrieval.** New York: Cornell University, 1987. 21 p. (Techinical Report 87-881).

[SAL 88] SALTON, G.; BUCKLEY, C. **Improving Retrieval Performance by Relevance Feedback.** New York: Cornell University, 1988. 24 p. (Techinical Report 88-898).

[SCA 95] SCARINCI, R. G. **Sistema de Descoberta de Conhecimento em Bases de Dados Não Estruturadas e Semi-Estruturadas para Manipulação de Informações.** 1995. 56 p. Trabalho Individual (Mestrado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

[SCA 97] SCARINCI, R. G; OLIVEIRA, J. P. M. SES – Sistema de Extração Semântica de Informações. In: SIMPÓSIO BRASILEIRO DE BANCO DE DADOS, 1997, Fortaleza, Ceara. **Anais...** Fortaleza: Universidade Federal do Ceará, 1997. p. 65-79.

[SCA 97a] SCARINCI, R. G. **SES – Sistema de Extração Semântica de Informações**. 1997. 170 p. Dissertação (Mestrado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

[SCA 2000] SCARINCI, R. G. **Extração de Informação**. 2000. 150 p. Exame de Qualificação (Doutorado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

[SCA 2001] SCARINCI, R. G. Extração de Informação como Base para Descoberta de Conhecimento em Dados Não Estruturados. In: WORKSHOP INTERNO SOBRE DESCOBERTA DE CONHECIMENTO EM BASES DE DADOS, 1., 2000, Porto Alegre. **Proceedings...** Porto Alegre: Instituto de Informática da UFRGS, 2001. v. 1, p. 15-20.

[SCA 2002] SCARINCI, R. G. et al. Managing Unstructured E-Commerce Information. In: INTERNATIONAL JOINT WORKSHOP ON CONCEPTUAL MODELING APPROACHES FOR E-BUSINESS - E-COMO, 3., 2002, Tampere. **Proceedings...** [S.l.: s.n.], 2002. v. 1, p. 50-62.

[SCA 2002a] SCARINCI, R. G. et al. E-Business Knowledge Based Information Retrieval. In: INTERNATIONAL SEMINAR ON ADVANCED RESEARCH IN ELECTRONIC BUSINESS, 1., 2002, Rio de Janeiro. **Proceedings...** Rio de Janeiro: Pontifícia Universidade Católica do Rio de Janeiro, 2002. v. 1, p. 106-113.

[SCH 97] SCHÜTZE, H.; SILVERSTEIN, C. Projections for Efficient Document Clustering. In: ACM INTERNATIONAL CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL, SIGIR, 20., Philadelphia, USA, 1997. **Proceedings...** New York: ACM Press, 1997. p. 74-81. Disponível em: <http://citeseer.nj.nec.com/76529.html>. Acesso em: 2000.

[SME 97] SMEATON, A. F. Information Retrieval: Still Butting Heads with Natural Language Processing. In: INTERNATIONAL SUMMER SCHOOL ON INFORMATION EXTRACTION, SCIE, 1997, Frascati, Italy. **Information Extraction: a** Multidisciplinary Approach to an Emerging Information Technology. Berlin: Springer-Verlag, 1997. p. 115-138

[SMI 97] SMITH, D.; LOPEZ, M. Information Extraction for Semi-structured Documents. In: WORKSHOP ON MANAGEMENT OF SEMI-STRUCTURED DATA, 1997, Tucson, Arizona. **Proceedings…** Disponível em: <http://citeseer.nj.nec.com/smith97information.html>. Acesso em: 1999.

[SOD 95] SODERLAND, S. et al. Issues in Inductive Learning of Domain-Specific Text Extraction Rules. In: WORKSHOP ON NEW APPROACHES TO LEARNING FOR NATURAL LANGUAGE PROCESSING AT THE INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE, 14., 1995, Montreal, CA. **Workshop Notes.** Montreal: AAAI, 1995. p. 290-301 Disponível em: <http://citeseer.nj.nec.com/article/soderland95issues.html>. Acesso em: 1996.

[SOD 95a] SODERLAND, S. et al. Crystal: Inducing a Conceptual Dictionary. In: INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE, 14., 1995, Montreal, CA. **Proceedings...** San Francisco: Morgan Kaufmann, 1997. p. 1314-1319. Disponível em: <http://citeseer.nj.nec.com/soderland95crystal.html>. Acesso em: 1996.

[SOD 95b] SODERLAND, S.; LEHNERT, W. Learning Domain-Specific Discourse Rules for Information Extraction. In: SPRING SYMPOSIUM ON EMPIRICAL METHODS IN DISCOURSE INTERPRETATION AND GENERATION, AAAI, 1995, Stanford, CA, USA. **Proceedings...** p. 143-148. Disponível em: <http://citeseer.nj.nec.com/article/soderland95learning.html>. Acesso em: 1996.

[SOT 98] SOTO, Patrícia. Text Mining: Beyond Search Technology. **DB2 Magazine**, [S.l.], v. 3, n. 3, 1998. Disponível em: <http://www.db2mag.com/db_area/archives/1998/q3/98fsoto.shtml>. Acesso em: 2000.

[SPA 97] SPARCK, J. K.; WILLET, P. **Readings in Information Retrieval.** San Francisco: Morgan Kaufmannm, 1997.

[SWA 89] SWARTOUT, W. R.; SMOLIAR, S. W. Explanation: a Source of Guidance for Knowledge Representation. In: MORIK, K. (Ed.). **Knowledge Representation and Organization in Machine Learning**. Berlin: Springer-Verlag, 1989. p. 1-16

[TOM 86] TOMITA, M. **Efficient Parsing of NL**: a Fast Algorithm for Practical Systems. Boston: Kluwer Academic, 1986.

[TRA 2001] TRADE Point de Porto Alegre. Disponível em: <http://www.tppoa.com.br>. Acesso em: abr. 2001.

[TRE 2000] TREIN, D. S. **Uma Arquitetura em Múltiplos Níveis para Extração de Informações**. 2000. 59 p. Trabalho de Conclusão (Bacharelado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

[TUR 91] TURTLE, H.; CROFT, B. W. Evaluation of an Inference Network Based Retrieval Model. **ACM Transactions on Information Systems**, New York, v. 9, n. 3, p. 187-222, July 1991.

[VIC 90] VICCARI, R. M. **Inteligência Artificial:** representação do conhecimento. Vitória: SBC, 1990. 71 p.

[WEI 88] WEISS, S. M. **Guia Prático para Projetar Sistemas Especialistas**. Rio de Janeiro: LTC, 1988. 169 p.

[WHA 2002] WHAT is an Information System? 2002. Department of Information Systems, Massey University, Palmerston North, New Zealand. Disponível em: <http://fims-www.massey.ac.nz/~is/is_whatwhy.html>. Acesso em: 2002.

[WIE 95] WIENER, E. D. et al. A Neural Network Approach to Topic Spotting. In: ANNUAL SYMPOSIUM ON DOCUMENT ANALYSIS AND INFORMATION RETRIEVAL, SDAIR, 4., 1995, Las Vegas. **Proceedings...** [S.l. : s.n.], 1995. p.317-332. Disponível em: <http://www.stern.nyu.edu/~aweigend/Research/Papers/TextCategorization>. Acesso em: 2000.

[WIL 88] WILLET, P. Recent Trends In Hierarchic Document Clustering: A Critical Review. **Information Processing & Management,** [S.l.], v.24, n.5, p.577-597, 1988.

[WIL 97] WILKS, Y. Information Extraction as a Core Language Technology. In: INTERNATIONAL SUMMER SCHOOL ON INFORMATION EXTRACTION, SCIE, 1997, Frascati, IT. **Information Extraction:** a Multidisciplinary Approach to an Emerging Information Technology. Berlin: Springer-Verlag, 1997. p. 14-18.

[WIL 2000] WILKS, Y; GAIZAUSKAS, R. **LaSIE - Large Scale Information Extraction**. Sheffield, UK: Department of Computer Science, University of Sheffield. Disponível em: <http://www.dcs.shef.ac.uk/research/groups/nlp/funded/lasie.html>. Acesso em: 2000.

[WIV 99] WIVES, L. K. **Um estudo sobre Agrupamento de Documentos Textuais em Processamento de Informações não Estruturadas Usando Técnicas de "Clustering"**. 1999. 84 p. Dissertação (Mestrado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

[WIV 2000] WIVES, L. K. **Tecnologias de Descoberta de Conhecimento em Textos Aplicadas à Inteligência Competitiva**. 2000. 100 p. Exame de Qualificação (Doutorado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

[WYA 95] WYATT, A. L. **Sucesso com Internet**. São Paulo: Érica, 1995. 404 p.

[YAN 99] YANG, Y.; LIU, X. An Evaluation of Statistical Approaches to Text Categorization. **Journal of Information Retrieval**, [S.l.], v.1, n.1/2, p.67-88, 1999.

[ZAI 99] ZAÏANE, O. R. **Resource and Knowledge Discovery from the Internet and Multimedia Repositories.** 1999. 304 p. Thesis (Ph. D. Thesis in Computer Science) - Simon Fraser University, B.C. Canada. Disponível em: <http://db.cs.sfu.ca/sections/publication/theses/theses.html>. Acesso em: 2000.

[ZAM 2001] ZAMBENEDETTI, Christian. **Uma Linguagem para Extração de Informações**. 2001. 150 p. Dissertação (Mestrado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.