Universidade de São Paulo
Instituto de Ciências Matemáticas e de Computação
Departamento de Computação e Estatística

Projeto e Desenvolvimento de uma Base de Dados Lexicais do Português

Juliana Galvani Greghi

Orientadora: Profª Drª Maria das Graças Volpe Nunes

Fevereiro 2002

"Gastei uma hora pensando em um verso
que a pena não quer escrever.
No entanto ele está cá dentro
inquieto, vivo.

Ele está cá dentro
e não quer sair.
Mas a poesia deste momento
inunda minha vida inteira."

(Carlos Drummond de Andrade)

... à minha família, razão da minha vida.

Agradecimentos

À Fapesp, pelo apoio financeiro;
à minha orientadora, Graça, pela confiança e dedicação;
aos colegas do NILC, pela valiosa contribuição para o meu trabalho e pelos momentos de diversão;
aos colegas do ICMC pela ajuda dispensada;
aos amigos, pelas palavras de incentivo;
à minha família, pelo apoio incondicional;
à Deus, por me dar forças para chegar até aqui.

Sumário

Lista de Figuras e Tabelas

Resumo

O desenvolvimento de recursos computacionais para PLN é, na maioria das vezes, uma tarefa árdua e demorada, principalmente na fase inicial, de aquisição de conhecimento. Essa tarefa pode ser simplificada com a centralização dos dados em um repositório que armazene todas as informações lexicais disponíveis de uma determinada língua.

Este trabalho documenta o desenvolvimento da DIADORIM, uma base relacional de dados lexicais para a língua portuguesa, com cerca de 1.5 milhão de entradas. Os desafios lingüísticos e computacionais encontrados durante todo o processo são discutidos. Várias interfaces de acesso e edição foram implementadas e avaliadas.

Abstract

The development of computational resources for NLP is usually an expensive task in terms of time complexity, mainly in knowledge acquisition phase. This task can be simplified by keeping the data in a central repository, which stores all the available lexical information of a particular language.

This work describes the development process of DIADORIM, a relational lexical database for Brazilian Portuguese with around 1.5 million entries. Linguistic and computational development problems are deeply discussed. Interfaces for accessing and editing data were implemented and evaluated.

Capítulo 1

Introdução

Os trabalhos desenvolvidos em Processamento de Língua Natural (PLN) podem, de um modo geral, ser divididos em duas classes: a) a classe dos recursos e b) a classe das aplicações. Fazem parte de *b* os tradutores automáticos, as interfaces em língua natural para bases de dados ou de conhecimento, os revisores ortográficos e gramaticais, os dicionários eletrônicos, os *thesauri*, os interpretadores e os geradores de língua natural. Por outro lado, em *a*, estão presentes os programas e as bases de informações lingüísticas que dão suporte ao desenvolvimento e implementação de aplicações de PLN. No primeiro grupo, de programas, temos os analisadores léxicos, sintáticos (*parsers*) e semânticos e os etiquetadores de texto (*taggers*), entre outros. No grupo de bases de informações lingüísticas encontram-se os *corpora* (bancos de textos escritos ou orais), os dicionários ou léxicos e as bases lexicais.

Os analisadores sintáticos ou *parsers* são responsáveis pelo processo de análise sintática de uma sentença, ou seja, pela combinação sucessiva de símbolos a ela pertencentes, com o objetivo de reconhecer sua estrutura sintática, validada por uma gramática. Para realizar essa tarefa, o *parser* necessita da ajuda de outros dois componentes: um léxico e uma gramática.

Os etiquetadores ou *taggers* são responsáveis pela marcação de um texto com etiquetas especiais, que podem ser morfossintáticas, semânticas, sintáticas ou prosódicas[1]. Esses etiquetadores podem ser construídos manualmente, por lingüistas, ou automaticamente, abstraídos de *corpus*.

Um *corpus* (plural *corpora*) é uma coleção de textos representativos, de tamanho finito, disponível em formato eletrônico e que sirva como referência da língua. Um *corpus* pode ser formado por textos da língua escrita e/ou falada, de gêneros distintos como jornalístico, literário, etc.

Os dicionários ou léxicos são uma importante fonte de informação lexical e semântica, muito usados em aplicações de PLN. Neste trabalho, um léxico é entendido como um

[1] Outras informações sobre *taggers* podem ser obtidas em http://www.ling.lancs.ac.uk/monkey/ihe/linguistics/corpus2/2types.htm (visitado em 02/02/2002)

conjunto de informações lexicais de um vocabulário, em forma digital, que é parte de uma aplicação de PLN. Por exemplo, em um revisor gramatical de português, o léxico contém informações morfossintáticas das palavras da língua portuguesa.

As bases lexicais são entendidas, aqui, como bases volumosas e abrangentes, compreendendo vários atributos lingüísticos para cada item lexical, e não necessariamente servindo a uma aplicação específica, mas à centralização e organização das informações lexicais, a fim de apoiar a pesquisa e o desenvolvimento de aplicações de PLN para uma dada língua. Neste trabalho, a língua em questão é o português do Brasil.

Para o uso em outros aplicativos, como tradução, interpretação ou verificação gramatical, a existência de uma base lexical abrangente, que contenha informações sobre a classificação de seus itens quanto a suas características sintáticas, morfológicas, e até mesmo semânticas, é muito importante. Pelo seu alto custo no que diz respeito a tempo de desenvolvimento, tamanho e especialização da equipe de desenvolvimento, o processo de construção desse tipo de recurso tem sido considerado uma das etapas mais difíceis no processo de construção de aplicativos abrangentes e robustos para PLN.

Até o início dos anos 80, o processo de desenvolvimento de léxicos e bases de informação lexical era realizado sem grandes preocupações com a padronização na elaboração e organização dos dados utilizados ou mesmo na construção do recurso propriamente dito, o que tornava a modificação e a reutilização dos dados duas tarefas praticamente impossíveis de serem executadas. A partir de então, vários pesquisadores passaram a se preocupar com a reutilização dos dados e, conseqüentemente, com a diminuição do esforço inicial para o desenvolvimento de novas aplicações (EVANS & KILGARRIFF, 1995). A grande maioria desses trabalhos aponta para a necessidade de construir bases lexicais extensíveis, interoperacionais e bem estruturadas, pois, além de exigir recursos humanos especializados, o trabalho de criação de uma base lexical é muito longo (WITTMAN & RIBEIRO, 1998).

O *Núcleo Interinstitucional de Lingüística Computacional de São Carlos* (*NILC*) tem desenvolvido, desde sua criação em 1991, vários aplicativos e recursos lingüísticos para o português brasileiro. Dentre os projetos desenvolvidos, destaca-se o Revisor Gramatical ReGra (MARTINS ET AL., 1998a), desenvolvido com o apoio da Itautec-Philco, da Fapesp, do CNPq e da Finep, que está comercialmente disponível no produto Redação da Língua Portuguesa, da Itautec-Philco, e também como parte integrante do Microsoft Office 2000, versão português.

Esse revisor conta com um léxico com mais de 1,5 milhão de entradas (incluindo flexões e derivações), cada uma podendo pertencer a uma ou mais categorias sintáticas, com atributos específicos e distintos (NUNES ET AL., 1996). Esse léxico é, originalmente, mantido em arquivos textuais volumosos e ineficientes para manipulação.

Além do léxico, o NILC possui outros recursos lexicais que podem, ao lado do léxico, compor uma base lexical significativamente mais abrangente, rica e informativa, tornando-se um recurso lingüístico valioso para o português brasileiro. Destacamos, entre esses recursos, os dados resultantes do projeto, em desenvolvimento, Universal Network Language (UNL), patrocinado pela Universidade das Nações Unidas (UNU), para o qual o NILC desenvolve ferramentas de tradução multilíngüe correspondentes ao português (OLIVEIRA JR., 2001), e os recursos oriundos do projeto PADCT-Finep/1999-2001, que abriga o subprojeto de um *thesaurus* para o português brasileiro (NUNES ET AL., 2001).

Com o intuito de centralizar os recursos lexicais do NILC foi desenvolvida uma base de dados que armazena as informações disponíveis nos diversos recursos desenvolvidos pelo grupo. Essa base lexical pode ser considerada um recurso de muita utilidade tanto para usuários comuns, por meio de consultas, como para a comunidade de processamento computacional do português, que poderá derivar recursos necessários para o desenvolvimento de diferentes aplicações. Em particular, a própria equipe do NILC se beneficiará com a centralização e reorganização dessas informações, tanto do ponto de vista do acesso quanto da sua manutenção e segurança. Para tanto, foi desenvolvida uma interface de aceso, dividida em quatro módulos: "Consulta aos dados morfossintáticos", "Consulta aos dados do *Thesaurus*", "Consulta aos dados UNL" e "Edição", além de uma ferramenta de extração de listas de dados especializada. Esse tipo de lista especializada pode ser bastante útil para a realização de estudos para o desenvolvimento de aplicações específicas.

Os objetivos deste trabalho foram: a) desenvolver a base de dados lexicais, a partir de agora referenciada como DIADORIM, incorporando as informações presentes no léxico do NILC, no dicionário UNL-Português utilizado no projeto UNL/Brasil, e as informações presentes no *thesaurus* da língua portuguesa; b) criar uma interface de acesso que possibilite a consulta e a edição dos dados, via Web, e uma ferramenta para extração de listas especializadas para aplicações particulares; c) disponibilizar tal recurso não só para o NILC e instituições vinculadas a ele, como também para outros grupos de pesquisa que, porventura, se interessem em utilizá-la para consulta, análise ou compilação de dados da língua portuguesa. Essa base de

dados pode servir como suporte aos projetos já desenvolvidos pelo grupo e como fonte de informação genérica para a construção de futuras ferramentas para o processamento automático do português.

Além das contribuições já explicitadas nos objetivos acima, vale ressaltar a importância dessa experiência quanto ao trabalho de projeto, implementação e avaliação de uma base de dados com tais características. A utilização de um Sistema de Gerenciamento de Banco de Dados (SGBD) para tal fim impõe desafios importantes, discutidos em detalhes neste trabalho.

Vale, neste momento, uma breve explicação a respeito da escolha do nome "DIADORIM": a obra "Grande Sertão: Veredas", de João Guimarães Rosa, narra a saga de Riobaldo e Diadorim. Autor brasileiro de indiscutível talento, Guimarães Rosa apresenta nesta obra, e em obras subseqüentes, a criação de neologismos formados por construções típicas da língua portuguesa. A escolha do nome "Diadorim" para a base de dados lexicais é uma homenagem ao autor, considerado uma figura de grande destaque no panorama da literatura brasileira.

No Capítulo 2 serão apresentadas algumas possibilidades de armazenamento para dados lexicais com suas vantagens e desvantagens; no Capítulo 3 serão apresentados trabalhos relacionados ao que foi desenvolvido; nos Capítulo 4 e 5 são relatados o processo de modelagem e desenvolvimento da DIADORIM, com detalhes sobre a implementação realizada, e o processo de projeto e desenvolvimento da interface de acesso, dividida nos módulos "Consulta aos dados morfossintáticos", "Consulta aos dados do *Thesaurus*", "Consulta aos dados UNL" e "Edição", e a ferramenta "Geração de Listas Especializadas". O Capítulo 6 apresenta os resultados obtidos na avaliação do sistema como um todo.

Capítulo 2

Bases de Dados Lexicais

As bases de dados lexicais são, em geral, utilizadas como repositório central de informações lexicais de uma determinada língua. As informações armazenadas podem ser de natureza sintática, semântica, pragmático-discursiva, fonético-fonológica ou morfológica, ou ainda podem expressar relações entre itens lexicais de uma mesma língua (como em um *thesaurus*) ou entre itens lexicais de línguas distintas (como em um dicionário bilíngüe). Além de facilitar a manipulação e manutenção das informações, essas bases são uma forma estruturada de armazenar os dados. Inicialmente os computadores eram utilizados apenas para acessar exemplos de usos das palavras. A partir da década de 80, esse cenário foi sendo alterado e, aos poucos, os computadores passaram a ter maior participação no processamento de palavras e no processo de criação de dicionários, principalmente com o desenvolvimento de bases de dados de informações lexicais (IDE ET AL., 1993).

As informações armazenadas em uma base de dados lexicais podem ser simplesmente consultadas e visualizadas ou podem ser utilizadas por aplicações de PLN, por exemplo, revisores gramaticais ou tradutores automáticos. A existência de uma base dessa natureza possibilita a atualização e manutenção de dicionários, a verificação da coerência das informações em um único dicionário ou entre vários, a troca de informações entre vários projetos e também a geração de diferentes versões de um dicionário com base em uma mesma fonte de dados.

Assim como os dicionários, as bases de dados lexicais foram, inicialmente, desenvolvidas em diferentes formatos, sem que qualquer padrão fosse estabelecido. E, novamente, a reutilização dos dados tornou-se uma tarefa difícil de ser realizada. Entretanto, a troca de informações e a integração de dados são de extrema importância para evitar a duplicação de esforços e possibilitar o desenvolvimento de grandes bases de informação lingüística, uma vez que dados lexicais apresentam um alto grau de complexidade em relação aos dados normalmente usados em pesquisas na área de banco de dados (IDE & VÉRONIS, 1992).

É necessário realizar um levantamento sobre os modelos de representação dos dados e também sobre os meios de armazenamento disponíveis, para que um modelo adequado para determinada aplicação seja implementado.

Durante os anos que se passaram, vários modelos foram usados, sendo os mais conhecidos os modelos Textual, Relacional e o baseado em *features* (IDE ET AL., *op.cit.*).

Modelo Textual

O modelo textual distingue as informações sobre os verbetes de uma língua por meio de *marcações,* que podem ser tipográficas, descritivas ou baseadas em gramáticas. As marcações tipográficas são bastante utilizadas em dicionários impressos e diferenciam as informações pelo uso de caracteres em *itálico*, sublinhados, em **negrito**, etc (Figura 1). Esse tipo de marcação, entretanto, apresenta certas desvantagens quando utilizada em dicionários eletrônicos, pois informações diferentes podem ter a mesma marca. As marcações descritivas diferenciam as informações através de indicações sobre o conteúdo do campo que está sendo delimitado (Figura 2). Esse tipo de marcação não apresenta os mesmos problemas da marcação tipográfica, mas exige que sejam desenvolvidos programas eficientes de recuperação da informação e que os textos marcados sejam quase estáticos, ou seja, que não sofram alterações ou atualizações constantemente, além terem uma estrutura rígida. As marcações baseadas em gramática, embora consigam descrever a estrutura hierárquica do documento por meio de uma gramática livre de contexto e não apresentem as mesmas limitações dos modelos puramente lineares, ainda são limitadas a textos fixos, que não podem ser facilmente atualizados ou modificados. Esses arquivos, em geral, são processados automaticamente para que a gramática utilizada possa agir sobre a estrutura do documento. As alterações exigem que o texto seja re-processado e, em alguns casos, que a gramática empregada seja alterada. Dessa forma, os textos devem sofrer o menor número de atualizações possível. Um exemplo de como poderia ser representado um arquivo com marcações baseadas em gramática é apresentado na Figura 3.

conectado *adj.* que se conectou; ligado, interligado ⟨*muitos fios c.*⟩ ⊙ ETIM part. de *conectar*; ver *nex-* ⊙ ANT desconectado

Figura 1 – Exemplo de uma entrada de um dicionário impresso usando marcações tipográficas[2]

[2] As informações contidas no exemplo foram retiradas do *Dicionário Houaiss da Língua Portuguesa (2001)*

```
<ent h=disuse ><hdw>disuse</hdw><pr><ph>dɪ'sjuːs</ph></pr>
<hps ps=n cu=U><hsn><def>the condition of not being used (any
longer)</def></hsn></hps></ent>
```

Figura 2 – Exemplo de uma entrada usando marcações descritivas[3]

Na Figura 2 as etiquetas utilizadas para descrever as informações indicam *entry, headword, part of speech*, etc.

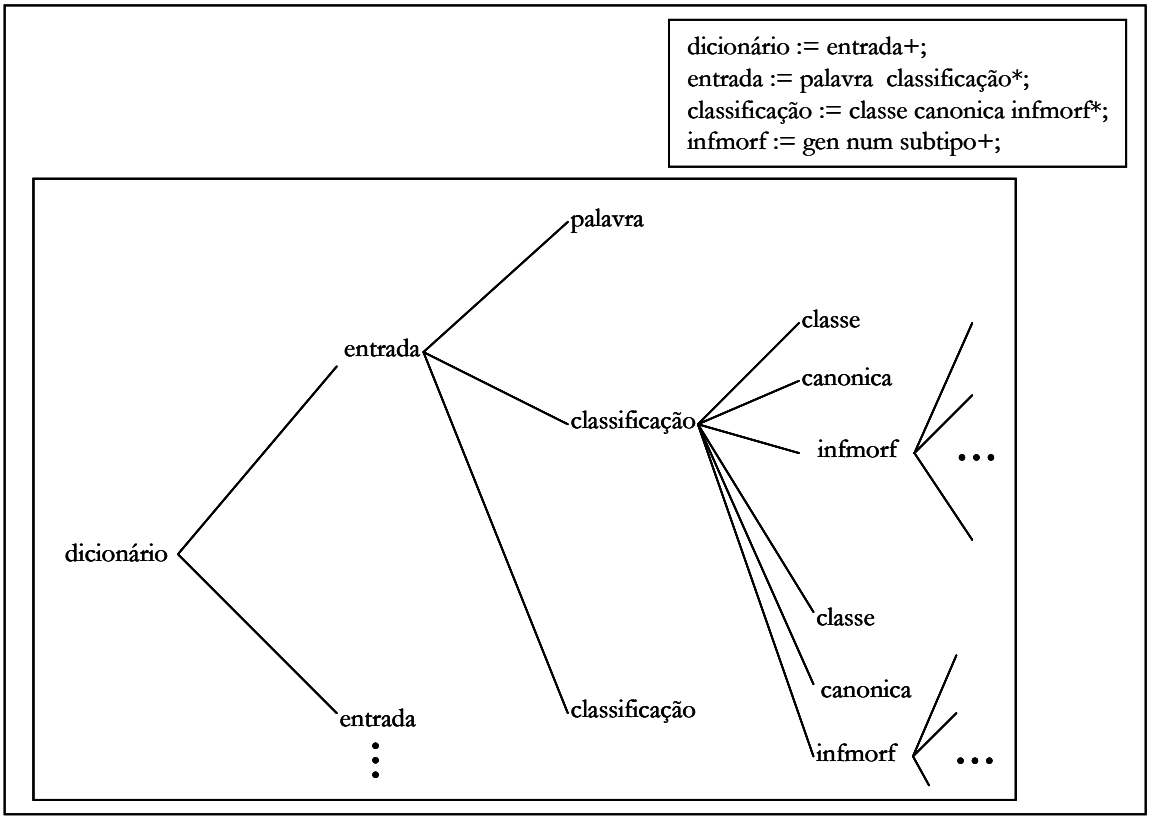

Figura 3 – Possível representação de parte de uma gramática e a árvore de decomposição correspondente

Modelo Relacional

O modelo Relacional, introduzido por Codd em 1970, tem sido um dos modelos mais conhecidos para a construção de bases de dados e baseia-se em uma estrutura de dados simples e uniforme: a relação (ELMASRI & NAVATHE, 2000). A base de dados é representada como

[3] O modelo de marcação descritiva foi baseado em exemplo apresentado em (IDE ET AL., 1993) e as informações presentes no exemplo apresentado foram extraídas do *Cambridge International Dictionary of English* (1995).

uma coleção de relações, que por sua vez, são representadas por tabelas. Cada linha na tabela representa uma coleção de valores de dados relacionados, que podem ser interpretados como entidades do mundo real ou relacionamentos entre elas. Cada linha da tabela é denominada *tupla* e cada coluna é um *atributo*. Para evitar redundância de informações, as relações devem ser normalizadas, ou seja, devem ser analisadas a fim de eliminar dados repetidos (ou que podem ser derivados a partir de outros dados) segundo as regras estabelecidas pelo método e a recuperação das informações é realizada por meio de consultas às relações definidas (Figura 4).

As vantagens na utilização do Modelo Relacional podem ser percebidas em: a) a tecnologia desenvolvida para manipular bases de dados relacionais já foi bastante aprimorada, apresentando, em vários casos, desempenho melhor que outros tipos de modelos; b) a organização e estruturação dos dados, conseguida com a aplicação do modelo, é bastante relevante e c) a recuperação dos dados por meio de consultas realizadas sobre as tabelas e os relacionamentos expressos pelas mesmas é bastante facilitada. Uma desvantagem na utilização desse modelo é que, para evitar redundância nos dados armazenados, as informações a respeito de uma mesma entidade ficam armazenadas em tabelas diferentes. Isso faz com que seja necessária uma busca cuidadosa, que una as tabelas em que dados de uma mesma entidade estão presentes, o que pode ocasionar certa perda de desempenho. Há também certa perda da semântica natural da aplicação, ou seja, a modelagem relacional é pouco intuitiva e, para que seja entendido o que está sendo representado, é necessário um estudo cuidadoso da representação. Outro problema encontrado diz respeito à perda da estrutura hierárquica de uma entrada do dicionário em virtude da fragmentação dos dados. Para tentar solucionar esse problema, é necessário simular essa hierarquia usando informações adicionais.

PALAVRA

ITEM	ID
Casa	12
Casaca	14
⋮	⋮
Relato	45
⋮	⋮

CLASSIFICAÇÃO

ID	CATEGORIA	CANÔNICA
12	substantivo	Casa
12	verbo	Casar
⋮	⋮	⋮
45	verbo	Relatar
⋮	⋮	⋮

Figura 4 – Exemplos de tabelas para o armazenamento de dados lexicais

Uma maneira encontrada para tentar solucionar o problema de perda de desempenho com a divisão dos dados em tabelas distintas é a utilização do Modelo Relacional sem que sejam realizadas as normalizações exigidas. Dessa forma, é possível recuperar a estrutura hierárquica das informações e a união de tabelas para recuperação de dados não é mais necessária (Figura 5). Apesar disso, esse modelo também tem problemas: não permite o aninhamento recursivo das relações, proibindo recursão e fazendo com que seja necessário que as consultas elaboradas procurem todas as posições possíveis em que uma informação sobre uma certa entidade possa aparecer. As duas formas são implementadas por SGBDs Relacionais.

PALAVRAS

ITEM	ID	CATEGORIA	CANÔNICA
casa	12	Substantivo	casa
casa	12	Verbo	casar
⋮	⋮	⋮	⋮
relato	45	Verbo	relatar
⋮	⋮	⋮	⋮

Figura 5 – Exemplo de tabela não normalizada

Modelo baseado em *features*

O modelo baseado em *features* representa as entradas baseadas em *feature structures*. Esse tipo de estrutura é usado há muito tempo no processamento de língua natural para codificar informações lingüísticas, especialmente em formalismos gramaticais. Esse modelo é baseado na Teoria Léxico-Funcional (KAPLAN & BRESNAN, 1982) (SHIEBER, 1986) e representa os itens lexicais como "objetos" distintos e as informações como pares (Atributo,Valor) (Figura 6). Foi originalmente descrito para ser utilizado em *parsers* para processamento de línguas naturais e, portanto, não oferece operações típicas de bases de dados. A principal vantagem encontrada neste modelo é a clareza com que as informações são representadas: os relacionamentos estabelecidos entre os itens lexicais, bem como o conjunto de informações referente a cada um deles podem ser facilmente compreendidos e recuperados. Entretanto, há certa dificuldade em implementar tal modelo, já que não existem SGBDs baseados em *features*. Uma possível solução para tal problema é a utilização de SGBDs orientados a objeto para

simular o comportamento do que seria uma base de dados desse tipo. Esses sistemas podem prover a flexibilidade e expressividade necessárias.

As dificuldades para a implementação desse modelo provêm de: a) a necessidade de se definir os dados de forma bastante específica, o que praticamente impossibilita a utilização de dados lexicais já especificados em outro formato, e b) da deficiência dos SGBD-OO em manipular grandes massas de dados. Como o Modelo Relacional é usado há bastante tempo, a tecnologia empregada em SGBDs Relacionais já está bastante aprimorada e, em alguns casos, pode ser mais eficiente.

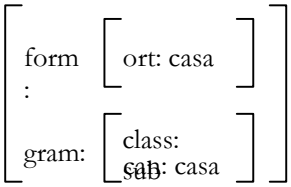

Figura 6 – Possível representação de dados em pares (Atributo,Valor)

Como pôde ser visto, existem diversas formas de se construir uma base de dados lexicais. Algumas mais eficientes, outras nem tanto. A forma como esta base deve ser implementada varia de acordo com as decisões particulares de cada projeto.

No próximo capítulo serão apresentados alguns trabalhos relacionados ao armazenamento e manipulação de dados lexicais que, apesar de tratarem do mesmo assunto, foram implementados de maneiras distintas.

Capítulo 3

Trabalhos Relacionados

Como foi citado anteriormente, existem várias iniciativas para a construção de bases de informações lexicais para várias línguas, em especial o inglês. Entretanto, nem todos os projetos que estão sendo desenvolvidos pela comunidade de PLN usam SGBDs para gerenciar os dados, mas todos têm tido a preocupação de adotar um padrão para a representação dos dados, a fim de que possam ser reutilizados.

A seguir, serão apresentados alguns exemplos de projetos que tratam do assunto.

3.1 WordNet

WordNet é um sistema de referência lexical do inglês, *on-line,* desenvolvido pelo *Cognitive Science Laboratory,* na Universidade de Princeton (MILLER ET AL., 1993).[4]

O sistema foi desenvolvido para o tratamento da língua inglesa e a principal característica desse projeto é a divisão do léxico em cinco categorias: substantivos, verbos, adjetivos, advérbios e palavras funcionais. Essa classificação, apesar de causar certa redundância das informações armazenadas (algumas palavras podem ser classificadas em mais de uma categoria), traz a vantagem de que diferenças fundamentais na organização semântica dessas categorias sintáticas podem ser claramente observadas e facilmente exploradas.

A base lexical da WordNet foi construída manualmente, apesar de ser um método custoso e demorado. As vantagens de tal desenvolvimento são que as entradas podem ser criadas com o conteúdo específico, que deverá ser útil em diversas aplicações, e o formato em que as informações são representadas pode ser controlado, exigindo menor esforço no processamento das informações. Ela é um dicionário semântico modelado como uma rede devido, em parte, à representação de palavras e conceitos como um sistema inter-relacionado (FELLBAUM, 1999).

[4] Outras informações podem ser obtidas em (http://www.cogsci.princeton.edu/~wn)

A característica mais ambiciosa da WordNet é a tentativa de organizar as informações lexicais em termos de significado da palavra, mais do que por sua forma. Isso faz com que muitos comparem a WordNet a um *thesaurus* online. Entretanto, essa base de dados inclui muito mais informação do que simplesmente um conjunto de sinônimos: ela armazena informações detalhadas sobre as relações entre as palavras e os conjuntos de sinônimos.

A WordNet trabalha sobre os conceitos de "forma" e "significado" de uma palavra. "Forma" refere-se à maneira como ela é representada, ou seja, a escrita correspondente; e "significado" refere-se ao conceito atribuído à palavra em determinada acepção. A associação entre uma forma e um conceito é realizada através de uma matriz lexical, em que o mapeamento entre formas e significados é N:N, ou seja, algumas formas podem ter diferentes significados, e alguns significados podem ser expressos por várias formas diferentes. O significado S1 de uma palavra pode ser representado por uma lista de palavras que podem ser usadas para expressá-lo { P1,P2,...}. Uma matriz lexical pode ser vista como um mapeamento entre palavras e conjuntos de sinônimos (*synsets*). Os sinônimos são relações lexicais entre palavras e têm papel central na Wordnet. Dessa forma, são usadas notações diferentes para representar relações de sinonímia, representadas por '{' e '}', e outras relações lexicais, representadas por '[' e ']'. As relações semânticas são representadas por ponteiros.

O modelo proposto para a Wordnet tem sido amplamente utilizado no desenvolvimento de projetos de mesma natureza para outras línguas. Um exemplo é o desenvolvimento do projeto EuroWordNet, que segue as diretrizes de desenvolvimento da WordNet para desenvolver *wordnets* para oito línguas européias: inglês britânico, alemão, francês, holandês, espanhol, italiano, tcheco e estônio. O projeto terminou no final de 1999 e as *wordnets* estão disponíveis em arquivos texto e em formato de bases de dados.[5]

3.2 Habanera

Habanera é uma base de conhecimento lexical multilíngüe, desenvolvido pelo *Computing Research Laboratory* (*CRL*), Universidade do Estado do Novo México, EUA. Essa base contém informações referentes a dicionários, estabelece relações entre os itens de um mesmo dicionário (por exemplo, sinônimos) ou entre dicionários distintos e deverá servir como repositório central de informações (ZAJAC, 1998). A língua inglesa é o núcleo da base, mas

[5] Todas as informações e documentação estão disponíveis no site do projeto http://www.hum.uva.nl/~ewn

foram incorporados recursos lexicais para outras línguas, tais como árabe, chinês, japonês, coreano, russo, espanhol e sérvio-croata.

Um dos objetivos desse projeto é aumentar a taxa de reutilização dos recursos desenvolvidos no CRL pela padronização da estrutura das entradas lexicais. Outro objetivo é diminuir os custos com aquisição de conhecimento, em particular, de conhecimento sintático e semântico usados em aplicações de PLN, e garantir a coerência e consistência dos dados.

Algumas das decisões de projeto para o desenvolvimento da Habanera foram: o uso das definições para marcação, em SGML, de textos eletrônicos (*Humanities Text Initiative – HTI*) como fonte para a definição de uma estrutura de entrada padrão; a arquitetura adotada segue as diferenciações entre meta-esquema, esquema e instância, definidas em pelo *Expert Advisory Group on Language Engineering Standards - EAGLES* (EAGLES, 1993), o que dá certa flexibilidade à modelagem da estrutura da base de conhecimento lexical; a organização dos dicionários como dicionários monolingües, estabelecendo relações entre as entradas; o uso de *typed features structures* como dispositivo primário de descrição; e o uso de um SGBD-OO comercial para armazenar os recursos lexicais do CRL e suportar usuários concorrentes. Essa base lexical faz parte de um projeto maior, denominado Corelli[6], cujo objetivo é desenvolver uma arquitetura e um conjunto de ferramentas apropriadas para o rápido desenvolvimento de sistemas de tradução multilíngüe.

A *Humanities Text Initiative* - HTI[7] - é uma unidade da Universidade de Michigan que, em colaboração com outras unidades da Biblioteca da Universidade, é responsável pela seleção de textos para a conversão para o formato eletrônico, criação de metadados para descrição dos textos e documentos fonte, e distribuição desse material na WWW. Um dos trabalhos desenvolvidos pelo HTI foi o desenvolvimento de diretrizes para a codificação, em SGML, de textos eletrônicos (*Text Encoding Initiative* – TEI). Essas diretrizes devem ser usadas para o intercâmbio de documentos entre grupos diferentes, usando produtos e sistemas em um grande conjunto de aplicações.

O *Expert Advisory Group on Language Engineering Standards* – EAGLES[8] é uma iniciativa da Comissão Européia, coordenada pelo *Consorzio Pisa Ricerche*, na Itália, que procura estabelecer padrões para: a) desenvolvimento de recursos em grande escala, por exemplo, *corpora* de textos

[6] Outras informações sobre o Projeto Corelli podem ser obtidas em http://crl.nmsu.edu/Research/Projects/corelli/index.html (visitado em 30/01/2002)
[7] Outras informações podem ser obtidas em http://www.hti.umich.edu e as diretrizes para codificação de textos (*TEI Guidelines*)podem ser obtidas em http://www.hti.umich.edu/t/tei/ (visitado em 30/01/2002)
[8] Outras informações sobre o grupo e trabalhos desenvolvidos podem ser obtidos em http://www.ilc.pi.cnr.it/EAGLES96/home.html

escritos, léxicos "computacionais" e *corpora* de textos falados; b) meios para manipulação de tais recursos através de formalismos utilizados em lingüística computacional e c) meios para acessar e avaliar diferentes recursos e ferramentas.

3.3 Projeto Parole

O Projeto Parole (*Preparatory Action for Linguistic Resources Organisation for Language Engineering*) está sendo desenvolvido sob a coordenação do *Consorzio Pisa Ricerche* (*CPR*), no *Istituto di Linguistica Computazionale* em Pisa, Itália, teve início em abril de 1996 e conta com a colaboração de várias instituições européias.

Os esforços desse projeto estão concentrados na produção de recursos lingüísticos de qualidade (corpora e léxicos para quatorze línguas européias), necessários para o desenvolvimento de aplicações e pesquisas, produzidos sob um formato uniforme, podendo ser reutilizados em outras aplicações.

Os corpora, produzidos em onze línguas da UE, são comparáveis no sentido de que têm tamanhos e composição relativa semelhantes, ou seja, textos de gêneros diferentes aparecem em proporções iguais nos *corpora* desenvolvidos.

Os léxicos são constituídos por lemas, seus tamanhos variam entre 20.000 e 30.000 entradas, e estão disponíveis em formato SGML. Os *corpora* serão desenvolvidos de acordo com as diretrizes do HTI, EAGLES e as diretrizes preparatórias do projeto PAROLE[9].

3.4 ACQUILEX

O projeto Acquilex, fundado pela Comissão Européia, foi realizado com a colaboração de vários grupos de pesquisa na área. O projeto Acquilex I desenvolveu técnicas e metodologias para a utilização de MRDs (*Machine-Readable Dictionaries*) na construção de componentes lexicais para sistemas de PLN. O principal objetivo foi estender as técnicas existentes para o processamento de MRDs monolingues para a extração de informações lexicais de vários MRDs multilíngües e, a partir disso, construir uma base multilíngüe de conhecimento lexical.

O projeto Acquilex II teve como objetivo estender os resultados obtidos no primeiro projeto e dar continuidade às pesquisas em modelagem de léxicos e construção de bases de conhecimento multilíngüe. O projeto também fez uso de *corpora* como fonte de informação para construção semi-automática de recursos lexicais. O trabalho foi dividido em duas áreas: o

(visitado em 30/01/2002)

[9] As diretrizes do projeto, os léxicos e *corpora* já concluídos e informações mais detalhadas podem ser obtidas em

desenvolvimento de uma metodologia e a construção de ferramentas para criar bases de dados lexicais a partir do MRDs, e a subseqüente construção de bases de conhecimento lexical a partir de fragmentos dessa base usando os softwares desenvolvidos para integrar, enriquecer e formalizar a base de informações.

Os resultados obtidos com os projetos tiveram grande influência no desenvolvimento de outras iniciativas internacionais e projetos, por exemplo, o *TEI*, *COMLEX* e *EAGLES*.

Como visto anteriormente, TEI é um conjunto de diretrizes para codificação, em SGML, de textos em formato eletrônico, criado pelo *Human Text Initiative*. EAGLES é um grupo é um grupo, fundado pela Comissão Européia, com o objetivo de estabelecer padrões para o desenvolvimento e avaliação de aplicações de diversos tipos.

COMLEX é um léxico para o inglês, com cerca de 38000 lemas, desenvolvido pela Universidade de Nova York, sob a supervisão do *Linguistic Data Consortium* (LDC). Esse léxico contém informações detalhas sobre as características sintáticas de cada item lexical e é particularmente detalhado no tratamento das subcategorizações dentro de cada categoria gramatical. O LDC é um consórcio entre universidades, companhias e laboratórios de pesquisa governamentais responsável por criar, coletar e distribuir bases de textos escritos e falados, léxicos e outros recursos para pesquisa e desenvolvimento e diversas áreas.

Os vários projetos relacionados ao trabalho apresentado não seguem uma mesma linha de desenvolvimento, demonstrando mais uma vez que, dependendo das condições iniciais de desenvolvimento e dos objetivos que se deseja alcançar, os métodos empregados para o projeto e desenvolvimento de uma base de dados lexicais podem variar. O ponto em comum entre todos é a percepção da necessidade de se ter um meio centralizador de informações, que facilite o desenvolvimento de novas aplicações para PLN e permita o intercâmbio de informações.

No próximo capítulo serão apresentadas as condições inicias e as decisões que guiaram o projeto e desenvolvimento da DIADORIM.

---

http://www.hltcentral.org/parole (visitado em 10/01/2002).

Capítulo 4

DIADORIM: da Modelagem à Implementação

A construção da base de dados lexicais foi dividida em 4 etapas:

- Modelagem e Implementação da Base de Dados
- Migração dos dados do léxico do ReGra
- Migração dos dados do Thesaurus
- Migração dos dados do dicionário UNL

4.1 Modelagem e Implementação da Base de Dados

O processo de desenvolvimento de um sistema pode ser dividido, basicamente, em duas etapas: a modelagem e a implementação. Na modelagem são definidos os objetivos a serem cumpridos e as diretrizes que irão guiar a próxima etapa, de implementação. Nesta fase, todas as diretrizes estabelecidas na fase anterior devem ser cumpridas a fim de que o sistema possa atender aos requisitos e necessidades que levaram ao seu desenvolvimento.

4.1.1 Modelagem

A DIADORIM foi criada a partir da reorganização das informações disponíveis em outras bases existentes no NILC, em especial, o léxico do ReGra - ferramenta de revisão gramatical automática[10]; e o dicionário Português-UNL, que servia a um projeto de tradução automática multilingual baseada em interlíngua[11]. A estrutura de cada uma dessas bases é bastante diferente, já que cada um é extremamente dependente da aplicação a que está relacionada (GREGHI ET AL., 2001).

[10] O Projeto ReGra vem sendo desenvolvido desde 1996 com auxílio de várias agências de fomento brasileiras (FINEP, FAPESP), em parceria com a iniciativa privada. Seus resultados podem ser observados no conjunto de ferramentas de auxílio à escrita Redação Língua Portuguesa (RLP) e nos aplicativos de revisão ortográfica e gramatical incorporados ao editor de textos Word, da MicroSoft, versão 2000 em diante. Para maiores informações sobre o ReGra, consulte-se (NUNES ET AL., 1996)

[11] Universal Networking Language (UNL) é uma interlíngua eletrônica desenvolvida para comportar a representação de informação de origem multilíngüe. Conforma a base do Projeto UNL, coordenado pelo IAS/UNU e pela UNDL Foundation, que subsidiaram a construção da base lexical correspondente para o português brasileiro. Para outras informações sobre a representação UNL e o Projeto UNL consulte-se (UCHIDA ET AL, 1999).

Essa diferença entre as estruturas implicava em duplicação de informações morfossintáticas e de esforço na manutenção das bases - as alterações no conjunto de informações dos verbetes devem ser realizadas duas vezes, o que não parece ser razoável, principalmente considerando o altíssimo custo dessas intervenções, quase sempre feitas manualmente por pessoal especializado. Dessa forma, surgiu a necessidade de fazer com que as bases de dados convergissem para um repositório central, sem que isso implicasse a perda das especificidades inerentes a cada uma das aplicações. O desafio se revelava tanto mais insuperável porque a simples junção das entradas compiladas em uma e outra base de dados lexicais afetaria o desempenho de ambas as ferramentas, na medida em que instalaria contradições (sob a forma de novos casos de homografia) para as quais não poderiam ser previstas estratégias de desambigüização (GREGHI ET AL., *op cit.*).

4.1.1.1 Modelagem Lingüística

Na tentativa de unir as bases em um repositório único, decidiu-se por um modelo híbrido, caracterizado pela convivência de estruturas de dados distintas, cujo ponto de interseção seria os verbetes do dicionário. A estrutura toda seria representada por duas componentes distintas: uma estrutura do tipo rede, denominada componente gnosiológica, usada para a representação das relações entre os verbetes e sua referência no mundo e na cultura, requeridas pelo Projeto UNL; e uma estrutura do tipo árvore, denominada componente lingüística, usada para a representação das relações entre os verbetes e a língua, requeridas por ambos os projetos (UNL e ReGra). O verbete ocuparia, simultaneamente, a função de nó da estrutura gnosiológica e a posição de raiz da estrutura lingüística, exercendo, dessa forma, a condição de unidade-ponte entre os dois conjuntos de informação.

A componente gnosiológica

A componente gnosiológica compreende um conjunto de nós, hipernós e arcos que corresponderiam à estrutura de conhecimentos (do mundo) recortada e definida pela cultura. Os verbetes estariam relacionados a este repositório de informações na medida em que apontariam para subcomponentes desta estrutura, que constituiriam, em última instância, seus

significados. Trata-se, em última instância, daquilo que na tradição da teoria lingüística tem sido referido como "designatum", "sentido" ou "intensão"[12].

É importante salientar que, no modelo proposto, o sentido de um verbete não corresponde a um conceito isolado, mas a estruturas conceituais complexas[13], indicadas pela natureza reticulada da componente gnosiológica. O verbete não está associado apenas a um nó isolado, mas também a um conjunto de arcos que chegam ao nó referido ou partem dele em direção a outros nós, de forma a estabelecer uma ativação em cadeia de sentidos co-relacionados, que possam ser utilizados no processo de desambigüização ou de validação dos enunciados lingüísticos[14].

A componente gnosiológica seria composta por dois conjuntos de dados: um conjunto de primitivos conceptuais (os nós) e um conjunto de relações entre esses primitivos conceptuais (os arcos). Os primitivos conceptuais, embora pressuponham uma "teoria do mundo" muitas vezes ainda não definida, conformariam definições de natureza antes espontânea, intuitiva, pré-teórica[15] e corresponderiam a categorias axiomáticas definidas pela cultura. No modelo adotado, seriam representadas como entidades indivisíveis, inanalisáveis em conjuntos de traços e irredutíveis a uma instância prototípica ou a várias instâncias exemplares[16]. Para efeito de citação, são rotuladas por palavras da língua inglesa.

As relações que se estabelecem entre os primitivos conceituais podem ser divididas em: relações ontológicas e relações psicológicas. As relações ontológicas corresponderiam a arcos que definiriam uma das quatro relações lógicas disponíveis:

a) sinonímia (equ): para a identidade de conceitos (face/cara/rosto)
b) antonímia (ant): para a oposição de conceitos por
   polarização (verdadeiro/falso);
   gradação (fervente/quente/morno/frio/gelado);

[12] Os conceitos de designatum (em oposição a denotatum), de sentido (em oposição a referência) e de intensão (em oposição a extensão) vêm sendo utilizados principalmente a partir do clássico Sobre o sentido e a referência, de Gotlob Frege.
[13] Reproduz-se, neste sentido, a distinção estabelecida em (JACKENDOFF, 1983), entre "linguistic expressions" e "conceptual structures".
[14] A estrutura da componente gnosiológica recupera, em linhas gerais, a idéia de acesso lexical prevista pelo modelo de Cohort (MARSLEN-WILSON & WELSH 1978); (MARSLEN-WILSON & TYLER, 1980), embora este último tenha sido utilizado como um modelo de ativação interativa para o reconhecimento de palavras.
[15] Cf. (MEDIN & ORTONY, 1989); (MURPHY & MEDIN, 1985).
[16] Diferentemente do que postulam, portanto, os modelos de categorização clássicos, que apostam na possibilidade de serem definidos conjuntos necessários e suficientes de traços diferenciadores (KATZ & FODOR, 1963); o modelo proposto por (ROSCH 1973, 1975), que se apóia na premissa de que algumas instâncias (os protótipos) de uma categoria são mais representativas do que outras, e que passariam a conduzir, por isso, o processo de categorização; e o modelo dos exemplares (MEDIN & SCHAFFER, 1978), que postula que a categorização é empreendida a partir de várias instâncias (exemplares), em vez de apenas uma.

inversão (pai/filho); e

exclusão (sábado/domingo);

c) hiponímia (icl): para a inclusão de conceitos (carro/veículo/objeto); e

d) partonímia (pof): para a partição de conceitos (flor/jardim).

Essas inter-relações caracterizariam uma ontologia navegável que instrumentalizaria vários dos recursos previstos para desambigüização e para o aconselhamento lexical, e serviria à operação do *thesaurus*.

As relações psicológicas materializariam as situações de co-ocorrência dos conceitos, caracterizando uma base de conhecimento, também navegável, constituída a partir das 38 relações binárias definidas pela interlíngua UNL (como agente, objeto, instrumento, beneficiário, etc.)[17]. Esses novos arcos formariam um conjunto de procedimentos de validação e revisão semântica das ferramentas do projeto UNL e do revisor gramatical.

A Figura 7 traz uma amostra da estrutura gnosiológica prevista para a base de dados lexical.

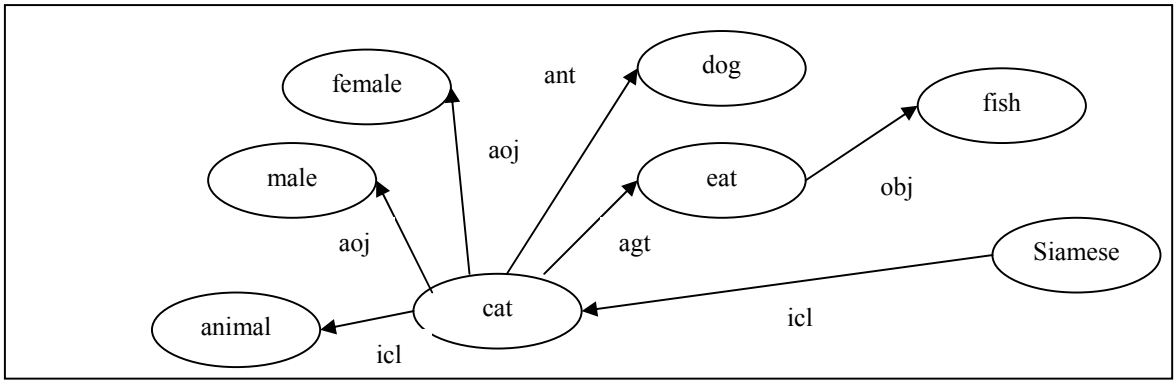

Figura 7 - Amostra de parte da estrutura conceptual prevista para o conceito representado por "cat"[18]

A componente lingüística

O objetivo da estrutura da componente lingüística é representar o conjunto de relações existentes entre os verbetes que compõem o sistema lingüístico. Para permitir essa associação, optou-se por representar o verbete em cada um dos níveis de análise lingüística,

[17] Para a lista exaustiva das relações semânticas previstas pela especificação UNL consulte-se (UCHIDA ET AL, 1999).

[18] Na figura apresentada, os arcos com a etiqueta "icl" correspondem a relações ontológicas de hiponímia, e o com a etiqueta "ant", a relações de antonímia; os arcos com as etiquetas "agt" (agente), "obj" (objeto) e "aoj" (atributo) correspondem a relações psicológicas definidas pela representação UNL.

acompanhando a idéia de que a língua é um sistema multiestratificado, comportando diferentes níveis de descrição, cada um dos quais elegendo problemas e utilizando modelos teóricos diferentes[19]. Foram propostos 5 diferentes níveis de descrição lingüística, cada um dos quais envolvendo categorias que lhes seriam específicas (Figura 8):

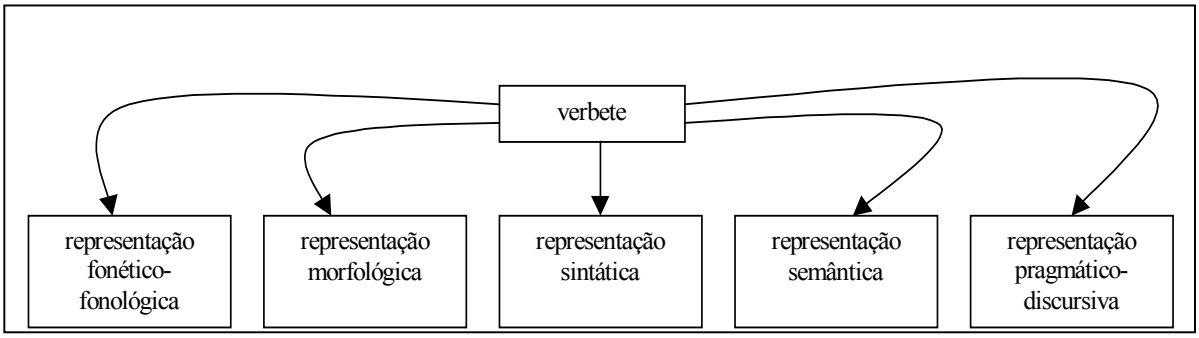

Figura 8 - Níveis de representação lingüística do verbete

Representação fonético-fonológica

Em um primeiro momento, essa representação corresponderia somente à oposição [±tônico], válida para a diferenciação dos pronomes pessoais oblíquos. Como a função principal da base, neste momento, é unir os dados usados pelos aplicativos disponíveis, não foram incorporados traços que não estejam sendo usados. Entretanto, em um futuro próximo, poderão ser incorporadas informações sobre a transcrição fonética da palavra, com a indicação da sílaba tônica, para que as estratégias de aconselhamento ortográfico possam ser aprimoradas.

Representação morfológica

A representação morfológica compreenderia a classificação do verbete em uma de quatro categorias: morfemas, lexias simples, lexias compostas ou lexias complexas. Os morfemas corresponderiam às unidades mínimas de significação da língua. Seriam as raízes e os afixos (prefixos ou sufixos), e não constituiriam, isoladamente, itens lexicais da língua. As lexias simples envolveriam combinações de raízes e afixos previstas pelos dicionários da língua ou consagradas pelo uso. Seriam cadeias de caracteres isoladas por espaços em branco. As lexias compostas envolveriam combinações de mais de uma raiz, e seriam classificadas segundo

[19] Para a pertinência da concepção de diferentes níveis de análise lingüística, consulte-se (BENVENISTE, 1966.)

a forma da composição (por justaposição ou por aglutinação). As lexias complexas corresponderiam a expressões fixas da língua, ou seja, a cadeias de caracteres que incluem espaços em branco. Em função da ausência de informação sobre a complexidade das lexias nas fontes de dados usadas, o modelo foi provisoriamente simplificado e a representação usada para a implementação pode ser vista na Figura 9.

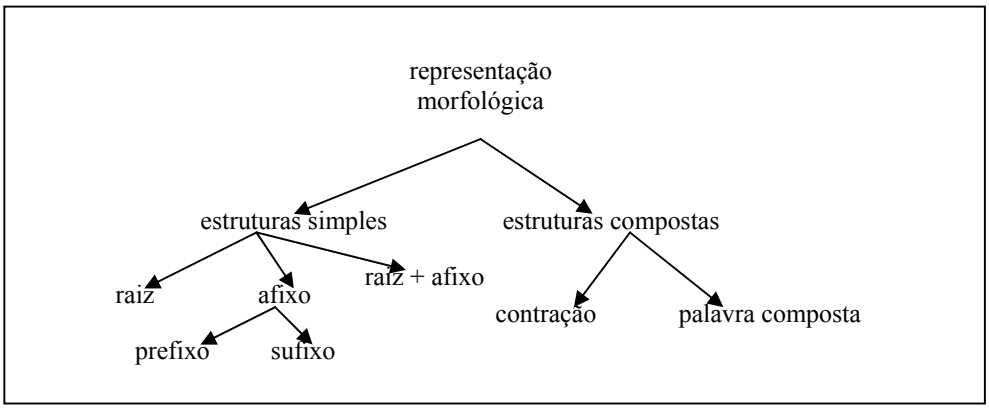

Figura 9 - Representação simplificada da estrutura morfológica do verbete

Representação sintática

Do ponto de vista de sua representação sintática, os verbetes seriam classificados segundo 1) seu comportamento gramatical e 2) seu comportamento sintático. Por comportamento gramatical, deve-se entender, aqui, a capacidade de um determinado verbete assumir um conjunto específico de flexões que o caracterizam como pertencente a uma determinada classe gramatical da língua portuguesa. Por comportamento sintático, presume-se o conjunto de relações de dependência que o verbete assume na sentença.

Seriam quatro as possibilidades de classificação morfossintática do verbete, derivadas da combinação dos primitivos sintáticos nome [N] e verbo [V][20]:

a) [+N,-V], que caracterizaria os verbetes cujo comportamento sintático coincidiria com o preenchimento da posição de núcleo do sintagma nominal: os substantivos

[20] Para a utilização de [nome] e [verbo] como primitivos gramaticais consulte-se (CHOMSKY, 1970) e (CHOMSKY & LASNIK, 1977).

propriamente ditos, os nomes próprios, as abreviaturas, as siglas, os pronomes pessoais do caso reto e do caso oblíquo, os pronomes de tratamento, alguns pronomes demonstrativos (como "isto"), alguns pronomes indefinidos (como "alguém), alguns pronomes interrogativos (como "quem"), alguns pronomes relativos (como "que"), os numerais coletivos (como "década") e os numerais multiplicativos (como "dobro"), nomeadamente;

b) [-N,+V], que caracterizaria os verbetes cujo comportamento sintático coincidiria com o preenchimento da posição de núcleo do sintagma verbal, ou seja, os verbos propriamente ditos;

c) [+N,+V], que caracterizaria os verbetes cujo comportamento sintático coincidiria com o preenchimento da posição de núcleo do sintagma modificador do sintagma nominal: os adjetivos, os pronomes possessivos, alguns pronomes demonstrativos (como "este"), alguns pronomes indefinidos (como "algum"), alguns pronomes interrogativos (como "qual"), alguns pronomes relativos (como "cujo"), os numerais cardinais (como "dois"), os numerais ordinais (como "primeiro"), os numerais fracionários (como "terço") e os artigos;

d) [-N,-V], que caracterizaria os verbetes cujo comportamento sintático coincidiria com o preenchimento da posição de núcleo do sintagma modificador do sintagma verbal ou de núcleo do sintagma modificador de outros sintagmas modificadores: os advérbios, as preposições, as conjunções, as interjeições.

Para cada um dos ramos derivados dessa quadripartição, estariam associadas categorias morfossintáticas específicas, acompanhando a estrutura da língua portuguesa[21]. O Grupo [+N,-V] traria, desta forma, informação relativa:

a) ao gênero gramatical do verbete, indicado pela combinação dos primitivos [masculino] e [feminino], para a formação do masculino (*sapato*) [+masculino,-feminino], do feminino (*saia*) [-masculino,+feminino], do comum-de-dois ou

[21] Na definição das classes e subclasses morfossintáticas, optou-se pela representação das categorias definidas pela Nomenclatura Gramatical Brasileira, reproduzidas em praticamente todas as gramáticas normativas da língua portuguesa, como (CUNHA & CINTRA, 1985), (BECHARA, 1976) e (ROCHA LIMA,1972).

uniforme (*pianista*) [+masculino,+feminino], do neutro, invariável ou não representado (*isso, bonit*) [-masculino,-feminino];

b) ao número gramatical do verbete, indicado pela combinação dos primitivos [singular] e [plural], para a formação do singular (*livro*) [+singular,-plural], do plural (*livros*) [-singular,+plural], do número uniforme (*lápis*) [+singular,+plural], do invariável ou não representado (*três, bonit*) [-singular,-plural];

c) à classe gramatical do verbete, representada por uma entre as seguintes possibilidades:

c1) substantivo comum (*mesa, cadeira, livro, saia, sapato*)
c2) substantivo próprio (*João, Rio de Janeiro*)
c3) sigla (ABNT, OAB)
c4) abreviatura (*dr., prof., p.*)
c5) pronome
c5a) demonstrativo (*isto*)
c5b) interrogativo (*quem*)
c5c) tratamento (*Vossa Senhoria*)
c5d) relativo (*o qual*)
c5e) indefinido (*alguém*)
c5f) pessoal (*eu, me, mim, comigo*)
c6) numeral (*dobro*)
c7) verbo (*fazer*)

A Figura 10 apresenta a estrutura do Grupo [+N,-V]

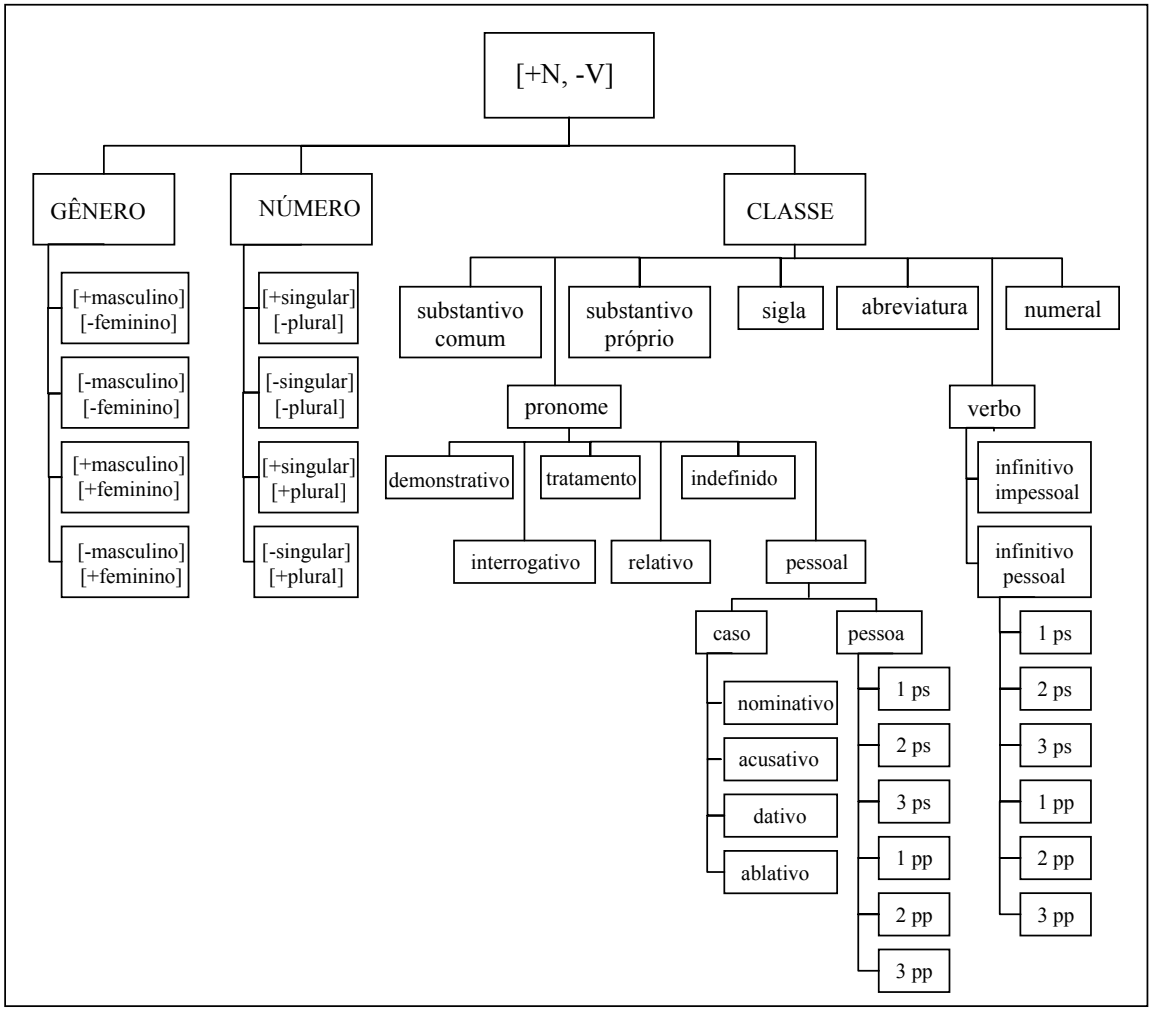

Figura 10 – Estrutura do Grupo [+N,-V]

O grupo [-N,+V] representaria as categorias pertinentes às formas verbais:

a) tempo: subdividido em tempo da referência e tempo do evento, acompanhando sugestão de (REICHENBACH, 1947);

b) modo: indicativo, subjuntivo e imperativo;

c) aspecto: perfeito e imperfeito.

d) pessoa: variando da primeira à terceira pessoa, do singular e do plural;

e) tipo de verbo: auxiliar (*estar*, em *Ela está fazendo isso*), de ligação (*estar*, em *Ela está doente*) ou nocional (*fazer*, em *Ela fez isso*)

A Figura 11 reproduz a estrutura do Grupo [-N,+V].

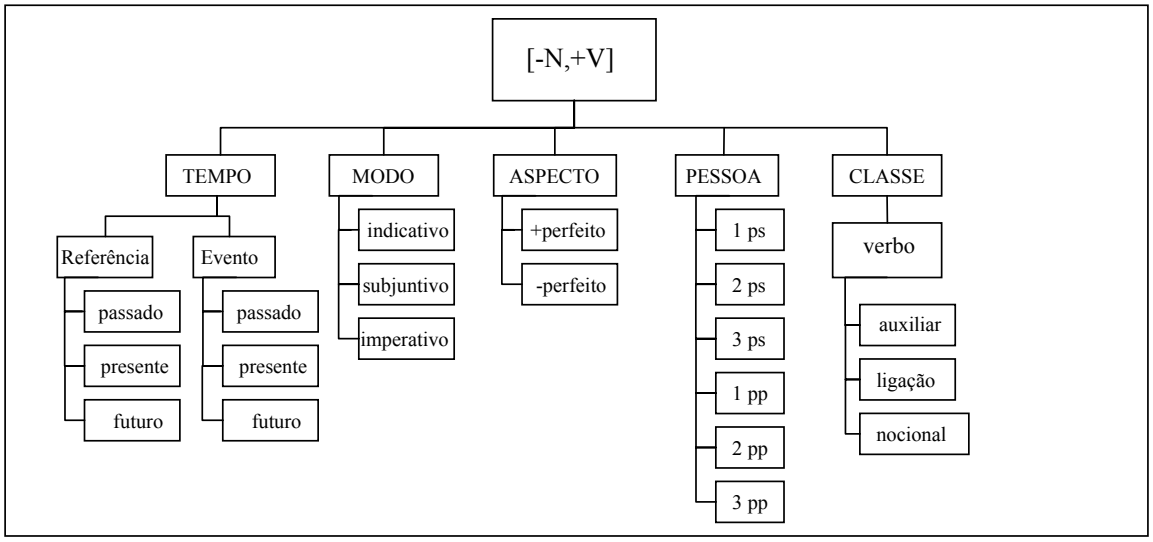

Figura 11 – Estrutura do grupo [-N,+V]

O grupo [+N,+V] reproduziria as informações de gênero e número já referidas no grupo [+N,-V], às quais acrescentaria a subclassificação prevista para as classes gramaticais, conforme indicado na figura 12.

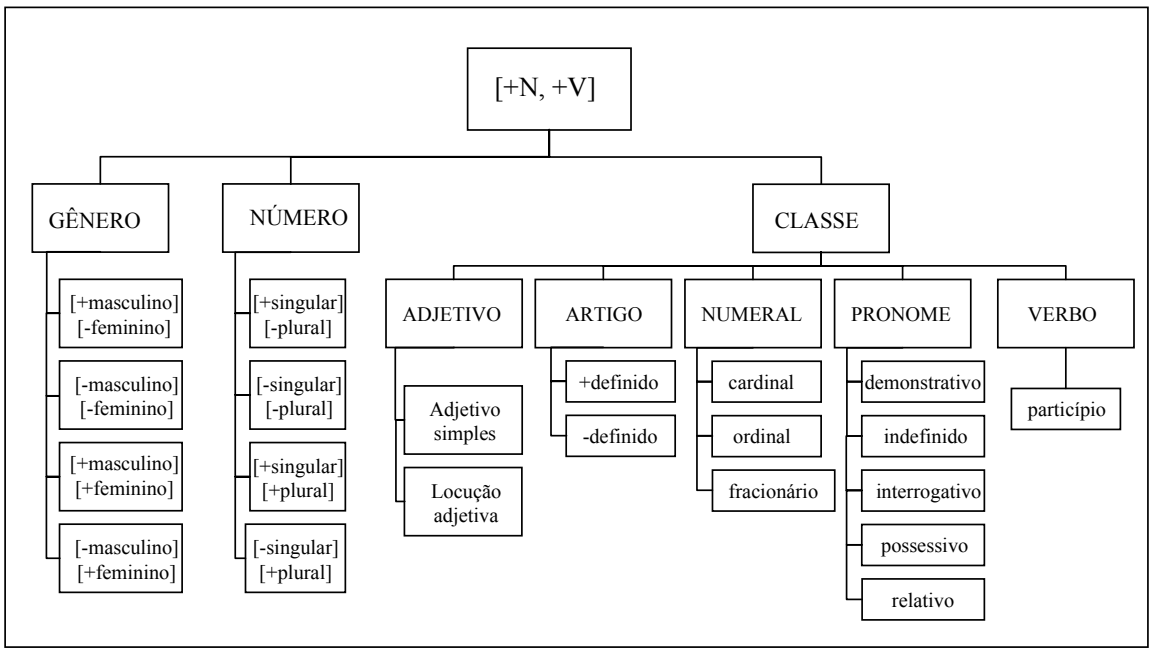

Figura 12 – Estrutura do grupo [+N,+V]

O grupo [-N,-V] representaria, por fim, as informações pertinentes às preposições, advérbios, interjeições, conjunções e as formas do gerúndio dos verbos, como indicado na figura 13.

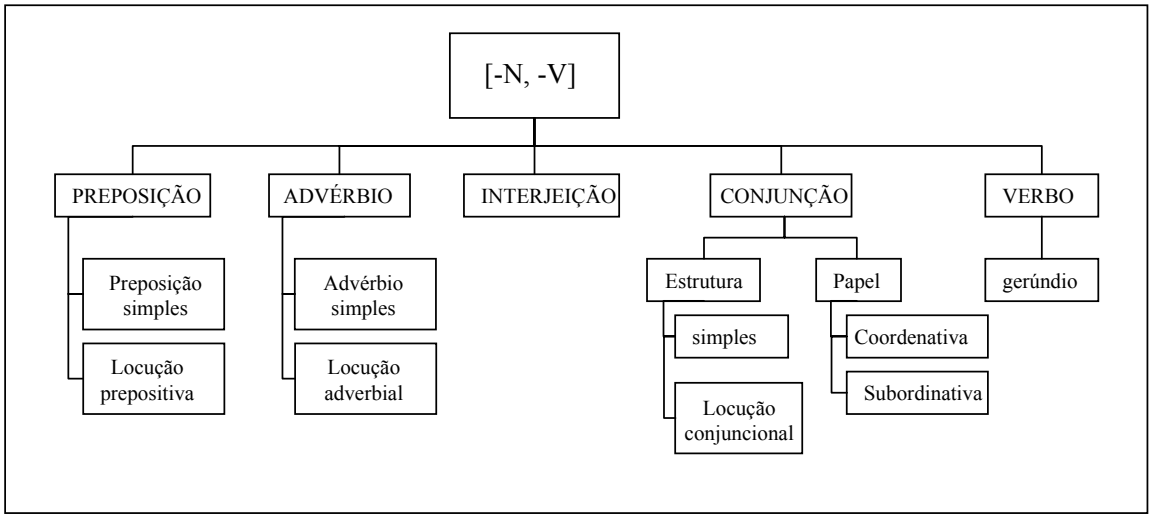

Figura 13 – Estrutura do grupo [-N,-V]

Em relação ao comportamento sintático, os verbetes representariam a) sua estrutura argumental e b) suas relações de regência. A estrutura argumental do verbete indicaria o número de argumentos por ele selecionado: verbetes impessoais (como os verbos impessoais e os substantivos não deverbais) não selecionariam nenhum argumento (0 argumento); verbetes intransitivos (com os verbos intransitivos, os adjetivos que não admitem complemento nominal, e substantivos deverbais) selecionariam um argumento; verbetes transitivos (como os verbos transitivos diretos ou indiretos, adjetivos e advérbios que requerem complemento nominal, e preposições) selecionariam dois argumentos; verbos bitransitivos (como os verbos transitivos diretos e indiretos e algumas preposições, como "entre") selecionariam três argumentos. A regência dos verbetes poderia ser b1) direta (se os argumentos selecionados não vêm precedidos por preposição), b2) indireta (se pelo menos um argumento selecionado vem precedido por preposição) e b3) pronominal (no caso dos verbos que selecionam obrigatoriamente como complemento o pronome pessoal oblíquo átono). Esses dois conjuntos de informação instrumentalizariam a análise sintática automática.

A figura 14 ilustra o nível sintático de representação dos verbetes.

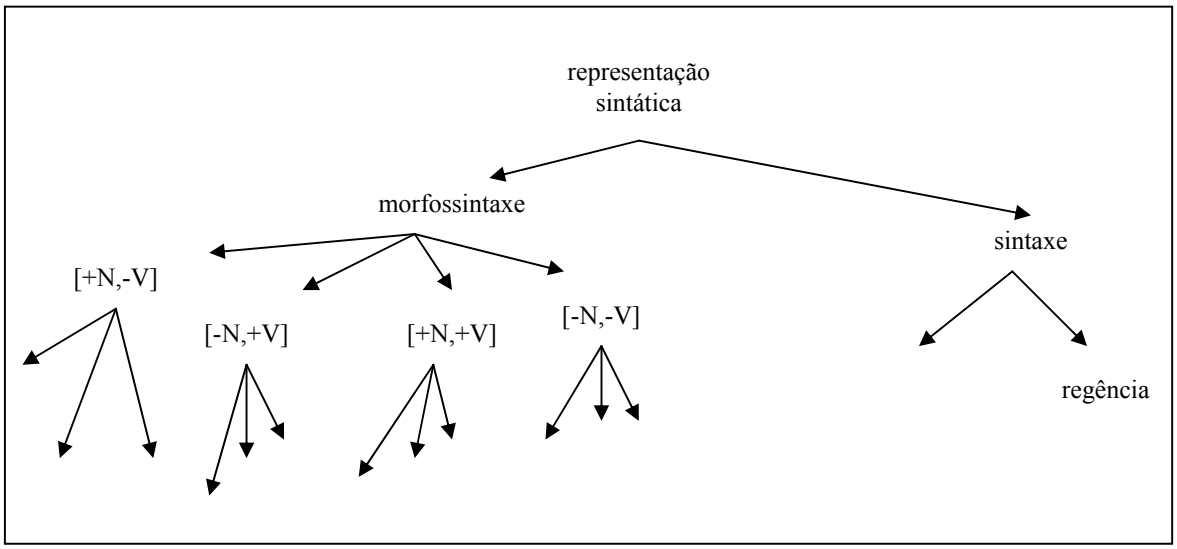

Figura 14 - Estrutura da representação sintática do verbete

Representação semântica e pragmático-discursiva

Os níveis de representação semântico e pragmático-discursivo do verbete estão ainda por serem desenvolvidos, a partir dos resultados obtidos pelo projeto TraSem (RINO ET AL, 2001), para a definição dos traços a serem incorporados às entradas lexicais.

Além dos dados do léxico do ReGra e do Dicionário Português-UNL, a DIADORIM também armazena dados do projeto Thesaurus Eletrônico para o Português do Brasil (TeP). Esses dados são conjuntos de sinônimos e antônimos, presentes na componente gnosiológica da estrutura proposta. A Figura 15 mostra uma representação total dessa estrutura.

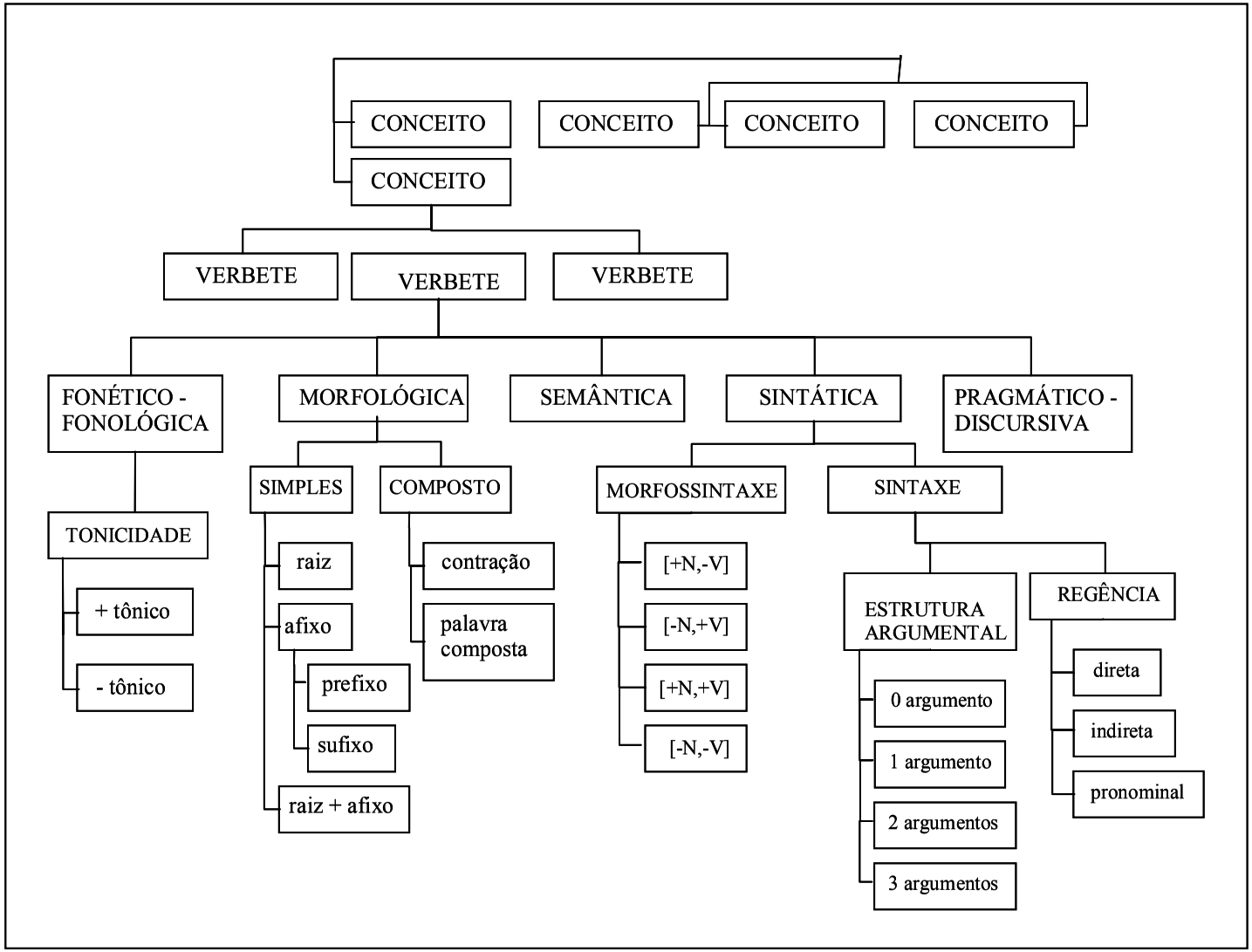

Figura 15 - Estrutura Proposta

4.1.1.2 Modelagem Computacional

A modelagem computacional pode ser dividia em duas etapas:

- Escolha do Modelo de Dados a ser usado
- Modelagem Computacional de acordo com o Modelo escolhido

Escolha do Modelo de Dados a ser usado

O armazenamento de dados lexicais desperta várias discussões a respeito da forma como os dados devem ser armazenados. Enquanto alguns trabalhos optam pelo armazenamento dos dados em arquivos com marcações apropriadas, por exemplo, as diretrizes definidas pelo grupo HTI (JANSZ, 1998); outras iniciativas apontam para a utilização de Sistemas de Gerenciamento de Bancos de Dados (ZAJAC, *op cit.*); (IDE ET AL., *op cit.*).

O armazenamento dos dados em arquivos especiais torna-os mais portáveis, já que os usuários potenciais não precisam ter acesso ao SGBD em questão. Tais arquivos devem ser processados por ferramentas específicas e a atualização dos dados deve ser minimizada, já que a cada alteração o arquivo deve ser novamente processado. Entretanto, quando se pensa em um volume de dados considerável como o da DIADORIM, que deverá servir como um ambiente centralizador, seguro e de fácil manipulação, as vantagens inerentes à utilização de um SGBD (consultas rápidas e elaboradas e maior segurança e consistência dos dados) parecem se sobrepor a qualquer argumento apresentado anteriormente.

Dessa forma, os critérios que levaram à conclusão de que a DIADORIM deveria ser armazenada em um sistema próprio para manipulação de bancos de dados podem ser assim resumidos:

- necessidade de organizar as informações disponíveis nos vários aplicativos desenvolvidos pelo NILC, padronizando a representação das informações, tentando evitar a inconsistência dos dados e aumentando a possibilidade de reutilização desses;
- garantia de segurança dos dados, já que os SGBDs têm dispositivos próprios para evitar danos acidentais aos dados armazenados e para restringir a forma de acesso que cada usuário pode ter, por exemplo, concedendo permissão de alteração dos dados somente a pessoas autorizadas.

Depois de definir que a DIADORIM seria armazenada em um SGBD, passou-se à escolha do Modelo de Dados a ser usado. Os dois modelos viáveis para o desenvolvimento de

uma base desse tipo são o Modelo Relacional e o Modelo baseado em *features* (IDE ET AL., *op.cit*). Neste trabalho optou-se pelo Modelo Relacional e essa escolha pode ser justificada pelos seguintes fatos:

- a utilização do outro modelo, implementado em um SGBD-OO, exigiria um esforço inicial muito grande, já que os dados do léxico deveriam ser completamente remodelados para serem utilizados em tal modelo;
- a perda de desempenho devido à junção de tabelas não é exatamente um problema, já que o número de aplicações existentes, que utilizam sistemas relacionais, é muito grande e, por isso, os algoritmos de junção de tabelas presentes nos SGBDs são bastante otimizados;
- a perda de informação sobre a hierarquia das informações pode ser solucionada através do uso de campos especiais, que simulam a hierarquia das informações, armazenando dados adicionais. Esses dados, por sua vez, podem ser extraídos do próprio conjunto de informações.

Modelagem Computacional segundo o Modelo de Dados escolhido

Como citado acima, o Modelo de Dados escolhido foi o Relacional. Para melhor visualização da estrutura da base, foi elaborado um Diagrama de Dados Entidade-Relacionamento, um modelo de dados conceitual de alto nível muito utilizado para o projeto conceitual de bases de dados. Esse diagrama é apresentado na Figura 16 e foi elaborado com base no modelo lingüístico apresentado anteriormente.

Cabem, neste momento, algumas explicações a respeito do diagrama apresentado: o relacionamento e a entidade marcados em azul indicam que cada palavra expressa um conceito, representado em uma ontologia da língua portuguesa. Essa ontologia, entretanto, ainda não foi criada e, por isso, tal relacionamento, não foi devidamente modelado. O mesmo acontece com os módulos de representação "Pragmatico-Discursiva" e "Semântica", que não foram modelados por não haver dados que possam ser utilizados para preenchê-los. Outro esclarecimento pertinente é sobre a entidade PALAVRA e seu atributo único *Lexema:* por que não coloca-lo como atributo da entidade classificação? A resposta é simples: esse modelo está em contínua atualização e deve ser incrementado com novas relações semânticas e pragmático-discursivas entre os itens lexicais. Dessa forma, é provável que novas entidades venham se

relacionar com PALAVRA, sendo importante manter as entradas independentes de qualquer classificação até que esses novos dados sejam implementados.

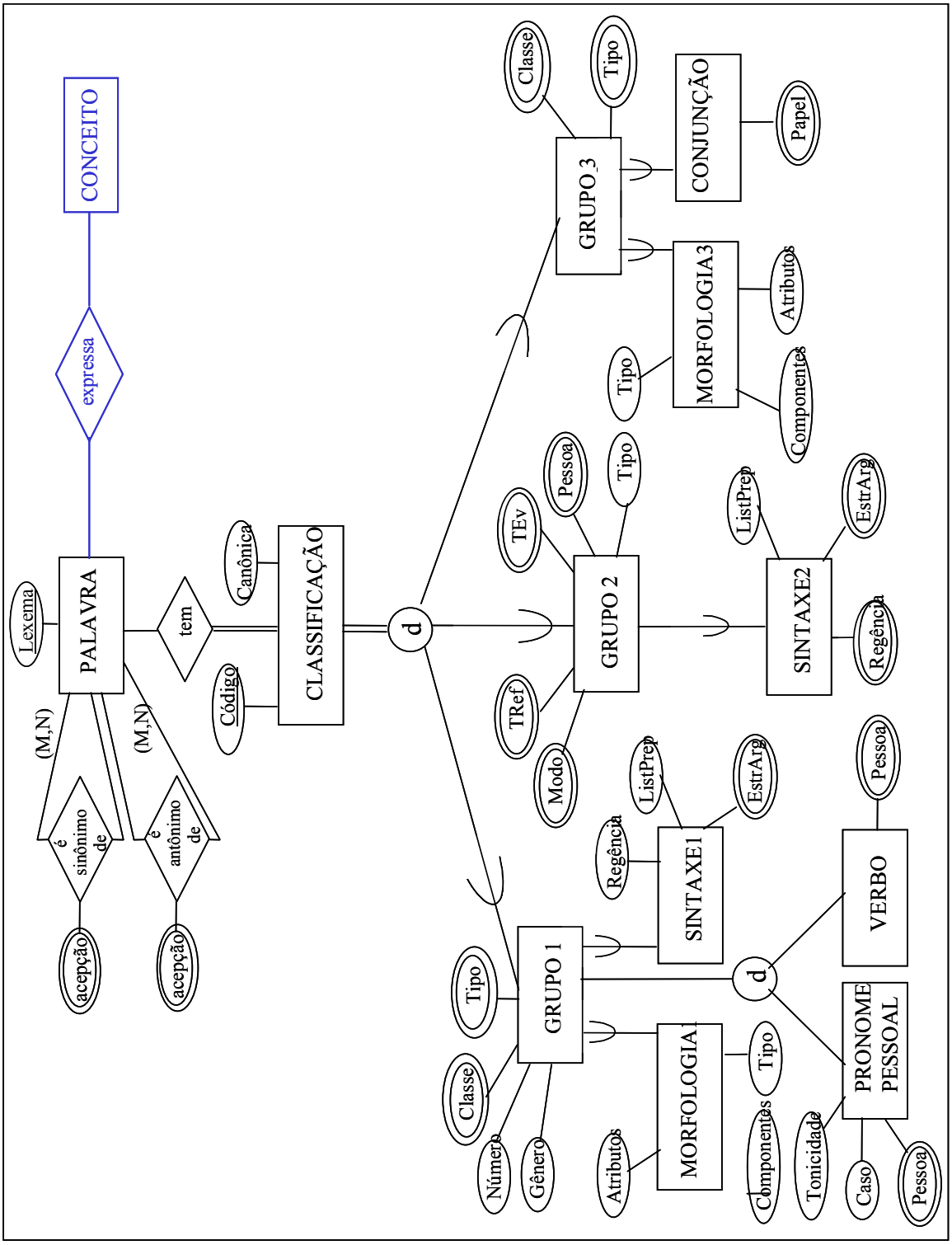

Figura 16 - Diagrama Entidade-Relacionamento

4.1.2 Implementação da Base de Dados

O processo de Implementação da base de dados pode ser dividido em três etapas:

- Escolha do Sistema de Gerenciamento de Banco de Dados
- Mapeamento das tabelas e criação da base de dados

4.1.2.1 Escolha do Sistema de Gerenciamento de Banco de Dados

A escolha do SGBD deve levar em consideração as consultas e operações de gerenciamento que serão realizadas com maior freqüência, as características e variáveis do SGBD que podem ser personalizadas de acordo com a aplicação, e quais as conseqüências resultantes dessas alterações para o sistema como um todo.

No caso da DIADORIM, o SGBD utilizado é o Microsoft SQL Server, 6.5. Ele tem demonstrado ser um SGBD confiável, com várias operações de gerenciamento que previnem erros, falhas ou perdas involuntárias de dados. Todas as informações necessárias para a manipulação do sistema estão disponíveis nos manuais que acompanham o software.

Um outro ponto a ser considerado é que o laboratório já possuía o SGBD em questão, não sendo necessário adquirir um sistema com função semelhante.

4.1.2.2 Mapeamento das tabelas e criação da base de dados

O primeiro passo realizado para a implementação da DIADORIM foi o mapeamento das tabelas. Este, por sua vez, seguiu todas diretrizes de normalização e mapeamento de tabelas apresentado em (ELMASRI & NAVATHE, *op cit.*). O mapeamento realizado para a criação da base de dados pode ser visto no Apêndice A - Detalhes sobre a implementação da DIADORIM.

O próximo passo realizado em direção à construção da DIADORIM foi calcular o tamanho provável de cada tabela e, dessa forma, estimar um tamanho máximo para a base de dados. O modo como esse cálculo foi realizado (MICROSOFT, 1995) pode ser visto, em detalhes, no Apêndice A - Detalhes sobre a implementação da DIADORIM.

Depois de realizar todos os cálculos necessários, foi definido o arquivo sobre o qual a base de dados é criada e os dados são armazenados. Esse arquivo é referenciado, pelo SQL Server, como *DatabaseDevice*. Ele tem extensão .dat e o seu tamanho deve ser o tamanho máximo previsto para a base de dados. No caso da DIADORIM, apesar do valor obtido com a estimativa de tamanho ter sido de 396 Mb, o valor usado para definição do arquivo foi de 600Mb. Essa definição de tamanho foi uma conseqüência direta do fato de que novos dados

deveriam ser inseridos futuramente e a base de dados teria seu tamanho alterado. Para evitar que futuramente a base de dados tivesse que ser estendida e um novo *device* tivesse que ser criado, optou-se por definir uma base de dados de tamanho superior ao estimado e, quando todos os dados já tiverem sido inseridos na base, efetuar uma diminuição do tamanho da base, se necessário.

O próximo passo foi a criação de um arquivo utilizado para armazenar o log de todas as transações efetuadas na base de dados (*LogDevice*). O tamanho desse arquivo de log deve ser definido com um valor entre 10 e 25% do tamanho total da base de dados. No caso da DIADORIM o arquivo de log foi definido com 25% do tamanho do arquivo de dados – 150 Mb.

Foram criados mais dois arquivos que serão usados para estender os segmentos de sistema e padrão (*system segment* e *default segment*), com tamanhos de 40 e 20 Mb, respectivamente. A determinação do tamanho desses segmentos foi empírica. Conforme a DIADORIM ia sendo usada e o tamanho das consultas aumentava, o sistema falhava e indicava que tais segmentos estavam cheios, sendo necessário aumentá-los para que as consultas especificadas pudessem ser realizadas. Como não foi encontrado qualquer tipo de recomendação a respeito do tamanho destes segmentos, eles foram sendo aumentados aos poucos, até que se atingisse um tamanho que permitisse a realização de tais operações sem que ocorressem falhas no sistema.

A DIADORIM foi criada e todas as tabelas foram definidas de acordo com o mapeamento realizado, mas os índices (chave primária, chave estrangeira, índice secundário) não foram definidos até que todos os dados tivessem sido inseridos na base.

Depois de analisar cuidadosamente um conjunto de variáveis do SGBD e as conseqüências de suas alterações para o sistema como um todo, algumas foram alteradas de acordo com a aplicação. Somente aquelas variáveis que não iriam causar perdas ao sistema foram alteradas. As variáveis que tiveram seus valores alterados foram (Tabela 1):

Tabela 1 – Conjunto de variáveis do SQL Server alteradas para a aplicação

Variável	**Valor padrão**	**Valor atual**
Memory	4096	8192
User connections	20	50
Remote Conn Timeout	10	10000

4.2 Migração dos dados do léxico do ReGra

Para que os dados do léxico do ReGra pudessem ser inseridos na DIADORIM, a tarefa de transferência foi dividida em 2 etapas: a) Análise dos dados do léxico e b) Inserção dos dados na base

a) Análise dos dados do léxico: O léxico originalmente usado no revisor gramatical ReGra é armazenado em um arquivo tipo texto e os dados referentes a um item lexical são expressos em uma cadeia única, com as informações separadas por caracteres especiais. Cada entrada é constituída de uma palavra ou, no máximo, palavras compostas hifenizadas. É importante ressaltar o papel da forma canônica, que tem a função de ligar toda palavra à forma básica que lhe deu origem. Com isso, possibilita-se recuperar as várias flexões e derivações de uma mesma forma básica. Assim, *bonita, bonitas, bonitos*, por exemplo, estão todas ligadas à forma canônica *bonito* e, conseqüentemente, todas ligadas entre si com seus atributos. Dois exemplos de entradas presentes no léxico são apresentados a seguir.

mundo=<S.M.SI.N.[]?.?.[mundo]0.>

No exemplo acima, a palavra *mundo* é classificada como substantivo, com gênero masculino(M), número singular (SI), grau nulo (N) e canônica "mundo".

dado=<ADJ.M.SI.N.[a.com.contra.em.por.]?.?.[dado]4.#V.[][PARTIC.M.SI.]N.[][dar]4.>

Neste exemplo, a palavra *dado* é classificada como adjetivo e verbo. A ordem em que as classificações aparecem é relevante, pois indica qual classificação deve ser mais facilmente encontrada em textos escritos. As classificações são separadas pelo caractere "#" e, dentro de cada classificação, as informações são separadas pelo caractere " **.** ". A classificação ADJ traz como informações o gênero (M.), o número (SI.), o grau (N), uma lista de preposições que são regidas pela palavra ([a.com.contra.em.por.]), e a canônica ([dado]). A classificação V traz como informações a conjugação, o gênero e o número ([PARTIC.M.SI.]), e a canônica ([dar]). As outras informações são para controle do revisor gramatical.

Cada classe gramatical presente no léxico foi analisada separadamente, verificando qual o conjunto de dados expresso para cada uma e como esses dados foram ordenados na cadeia de

informações. Esse estudo foi realizado com a análise do léxico e com a descrição detalhada do léxico apresentada em (NUNES ET AL, *op cit.*).

b) Inserção dos dados na base: para que a transferência dos dados do léxico para a DIADORIM pudesse ser realizada automaticamente, foi necessário o desenvolvimento de uma ferramenta que fizesse a conversão e a formatação dos dados existentes para o novo formato exigido pela modelagem realizada. Essa ferramenta processa o arquivo de entradas do léxico analisando, linha a linha, todas as informações disponíveis sobre cada item lexical. As informações são recuperadas e inseridas em arquivos com extensão .txt específicos, que representam as tabela implementada no SGBD. Para que as informações recuperadas possam ser inseridas corretamente nas tabelas, o caractere " \ " indica o fim de cada campo, permitindo, assim, que esses arquivos sejam copiados para as respectivas tabelas. Essa operação é realizada com a execução do comando bcp (*bulk copy*), disponível entre as funções e aplicações do SQL Server (MICROSOFT, *op cit.*).

4.3 A Migração do Thesaurus

O projeto do *Thesaurus* Eletrônico para Português do Brasil[22] (TeP), teve como objetivo a construção de uma base de *thesaurus* eletrônico que permitisse ao usuário da língua portuguesa substituir uma determinada palavra, durante a composição de um texto, por razões de estilo e/ou precisão, baseado em um repertório de sinônimos e antônimos. Cada conjunto de sinônimos e antônimos é associado a uma determinada acepção de uma forma lexical e é construído com base nas relações semânticas - sinonímia e antonímia - que se estabelecem entre essas formas, não entre os conceitos por elas atualizados. Por essa razão, essas relações semânticas são denominadas "relações de sentido" (DIAS-DA-SILVA ET AL, 2000). A base de informações do thesaurus conta com cerca de 40.000 conjuntos de sinônimos e antônimos entre verbos, substantivos, adjetivos e advérbios. Vale, neste momento, explicar que, para que a base de informações possa, realmente, ser usada como um thesaurus, é necessário que, para cada entrada, exemplos de uso sejam fornecidos para todas as acepções (sinônimos) associados.

Essa base de informações, assim como o léxico do ReGra, é armazenada em um arquivo com extensão .txt, com as informações separadas por caracteres específicos. Para que esses

[22] Mais informações podem ser obtidas em http://www.nilc.icmc.sc.usp.br, no link "Tools & Resources".

dados pudessem ser inseridos na base de dados automaticamente, foi desenvolvida uma ferramenta que formata os dados do *Thesaurus* e insere-os, através do comando bcp, nas tabelas da DIADORIM. Um exemplo de entrada usada na base de informações do *Thesaurus* é mostrado a seguir.

> abotoadura = S={abotoadura, abotoamento} A={desabotoamento, desabotuadura, desafivelamento, desaperto} # R={}.

Neste exemplo é apresentado o substantivo *abotoadura* com seus conjuntos de sinônimos (S={abotoadura, abotoamento}) antônimos separados por (A={desabotoamento, desabotuadura, desafivelamento, desaperto}). O conjunto "R={}" ainda deve ser preenchido e deverá trazer palavras relacionadas àquela sendo analisada. Caso a palavra possa ser classificada em mais de uma acepção, as informações referentes a cada acepção são separadas pelo caractere "&". Caso uma palavra não tenha sinônimos ou antônimos, o conjunto é representado pelo conjunto vazio "{}", como no exemplo abaixo, em que a palavra *pingo* apresenta três acepções distintas e, em nenhuma delas, o conjunto de antônimos é preenchido.

> pingo = S={pingo, respingo, ressalte, ressalto, salpicadura, salpicamento, salpico} A={} & S={banha, chorume, gordura, pingo, pingue} A={} & S={gota, pinga, pingo} A={} # R={}.

4.4 Migração dos dados do Dicionário UNL-Português

O projeto UNL (*Universal Networking Language*)[23] propõe a construção de uma linguagem intermediária que possa auxiliar na comunicação multilíngüe por meio de sistemas computacionais de PLN (UCHIDA, 1999). O objetivo final é que um usuário da Internet possa ter acesso aos softwares de codificação e decodificação UNL e possa, dessa forma, elaborar documentos e disponibilizá-los em UNL e possa ter acesso a outros documentos UNL e codificá-los para a sua própria língua (OLIVEIRA JR. ET AL., 2001).

Ambos os softwares UNL trabalham sobre um dicionário no qual palavras de uma determinada língua são associadas a uma representação simples, que indica o significado genérico de uma palavra em inglês. Esses conceitos são denominados "Palavras Universais" ou *Universal Words* (UWs).

[23] Mais informações podem ser obtidas em http://www.nilc.icmc.sc.usp.br, no link "Projects".

O dicionário UNL-Português conta com cerca de 60000 entradas e a sintaxe utilizada no dicionário segue ao seguinte formato (MARTINS ET AL., 1998b):

[headword] canônica "UW" (traços gramaticais) <P,f,p>;

onde: *headword* é a palavra do português, correspondente ao significado expresso pela UW;

'canônica" é a forma canônica da palavra em português;

(traços gramaticais) é o conjunto de traços gramaticais e semânticos da *headword*

(informações de natureza morfológica, por exemplo)

P: indica português;

f e **p** são valores que exprimem a freqüência e a prioridade de uso da *headword*

A Figura 17 mostra dois exemplos de entradas para o dicionário UNL-Português.

```
[ambiente] {} ambiente "environment" (s,stem,masc,rege(de))<P,0,0>;

[favore] {} favorecer "encourage(icl>event)" (16, v,stem, vtd, ação) <P,0,0>;
[favore] {} favorecer "further(icl>event)" (16, v,stem, vtd, ação) <P,0,0>;
[favore] {} favorecer "promote(icl>event)" (16, v,stem, vtd, ação) <P,0,0>;
```

Figura 17 – Exemplos de entradas para o dicionário UNL-Português

No primeiro exemplo apresentado na Figura 17, vê-se a *headword* "ambiente" acompanhada da informação sobre a canônica, em português, a *universal word* correspondente e o conjunto de informações semânticas e gramaticais. No caso de uma palavra apresentar mais de uma acepção, como mostra o segundo exemplo apresentado, são representadas entradas distintas para cada UW definida.

Os dados desse dicionário são armazenados em tabelas, gerenciadas pelo MSAccess. Para transferir os dados para a DIADORIM, foi realizado um breve estudo para definir quais informações deveriam permanecer armazenadas e qual seria a melhor forma de relacioná-las aos dados já existentes na base.

Definidos tais pontos, a transferência de dados ocorreu de forma direta, já que tanto o MSAccess quanto o SQL Server têm funções de exportação e importação de dados de outros sistemas de gerenciamento e esse processo pôde ser realizado automaticamente.

Capítulo 5

Interfaces e Ferramentas de Acesso

O acesso aos dados armazenados na DIADORIM pode ser realizado através de uma interface, via Web, dividida em quatro módulos, que dão acesso aos conjuntos de dados disponíveis ou permitem a edição dos mesmos. Além da interface via Web, foi desenvolvida uma ferramenta que permite a extração de listas contendo um conjunto de informações específicas, de acordo com a necessidade do usuário.

O acesso à DIADORIM ficou, dessa forma, dividido em:

- Módulo de consulta aos dados morfossintáticos
- Módulo de consulta aos dados do *Thesaurus*
- Módulo de consulta aos dados UNL
- Módulo de edição dos dados
- Ferramenta de Geração de Listas Especializadas

As interfaces de consulta aos dados morfossintáticos e do *Thesaurus* foram avaliadas junto ao usuário e os resultados da avaliação serão apresentados no Capítulo 6. As interfaces de consulta aos dados UNL e de edição dos dados têm acesso restrito a pessoas autorizadas.

5.1 Módulo de consulta aos dados morfossintáticos

A DIADORIM, além de repositório central de dados, também deve servir como fonte de consulta para usuários que queiram obter informações detalhadas a respeito dos itens lexicais da língua portuguesa. Para tornar tal funcionalidade viável, foi necessário desenvolver um módulo que permitisse a qualquer usuário, leigo ou especialista, consultar as diversas informações disponíveis na base de dados.

Antes que essa interface fosse criada, foi feito um breve levantamento junto ao usuário potencial do sistema - pesquisadores e estudantes da área de computação, lingüística e lingüística computacional - sobre as informações que deveriam estar disponíveis e a forma como essa consulta poderia ser realizada. A partir desse levantamento foi elaborado um

conjunto de requisitos gerais que deveriam ser atendidos pelo módulo de consulta aos dados morfossintáticos, apresentado a seguir:

Conjunto de Requisitos para o Desenvolvimento do Módulo de Consulta aos dados Morfossintáticos:

1. A busca de informações deve ser disparada com o uso de uma palavra da língua portuguesa
2. O sistema deve permitir que o usuário faça buscas indiretas, ou seja, possa consultar uma lista de palavras que obedeçam a uma condição de busca especificada.
3. O sistema deve permitir que o usuário selecione uma das palavras listadas e use-a como gatilho para uma nova busca
4. O sistema deve permitir que o usuário restrinja a busca de informações por categorias gramaticais.
5. As informações devem ser apresentadas ao usuário de forma clara e completa, de acordo com as restrições pré-estabelecidas pelo mesmo.
6. O sistema deve permitir que o usuário tenha acesso aos outros módulos de consulta aos dados da DIADORIM.

De posse desse conjunto de requisitos, alguns protótipos foram desenvolvidos até chegar a versão representada pelas figuras 18, 19, 20 e 21.

Figura 18 – Tela de entrada do módulo de consulta aos dados morfossintáticos

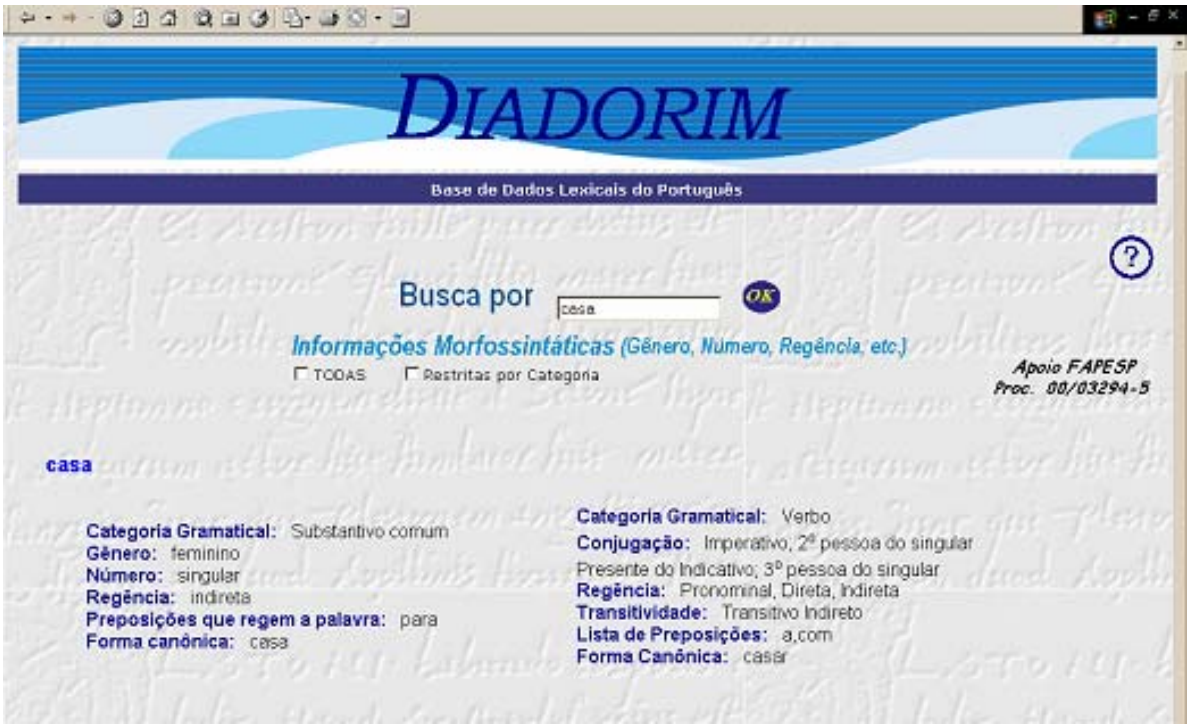

Figura 19 – Tela de apresentação de resultados de uma consulta realizada

Figura 20 – Consulta usando máscara

Figura 21 – Tela de Ajuda do módulo de consulta aos dados morfossintáticos

5.2 Módulo de consulta aos dados do *Thesaurus*

O módulo de consulta aos dados do *Thesaurus* permite que o usuário visualize o conjunto de sinônimos e antônimos de uma determinada palavra. Assim como ocorreu para o

desenvolvimento do módulo de acesso aos dados morfossintáticos, alguns requisitos foram definidos para o desenvolvimento do módulo de acesso aos dados do *Thesaurus*.

Conjunto de Requisitos para o Desenvolvimento do Módulo de Consulta aos dados do *Thesaurus*:

1. A busca de informações deve sempre ser disparada com uma palavra da língua portuguesa.
2. O sistema deve apresentar ao usuário todas as acepções a que uma palavra possa estar relacionada e deve permitir que o usuário selecione a acepção sobre a qual deseja obter mais informações
3. O sistema deve apresentar o conjunto de sinônimos e antônimos da palavra pesquisada a cada acepção selecionada.
4. O sistema deve permitir que o usuário tenha acesso aos outros módulos de consulta aos dados da DIADORIM

De posse desse conjunto de requisitos, o módulo de consulta foi desenvolvido (figuras 22, 23, 24 e 25).

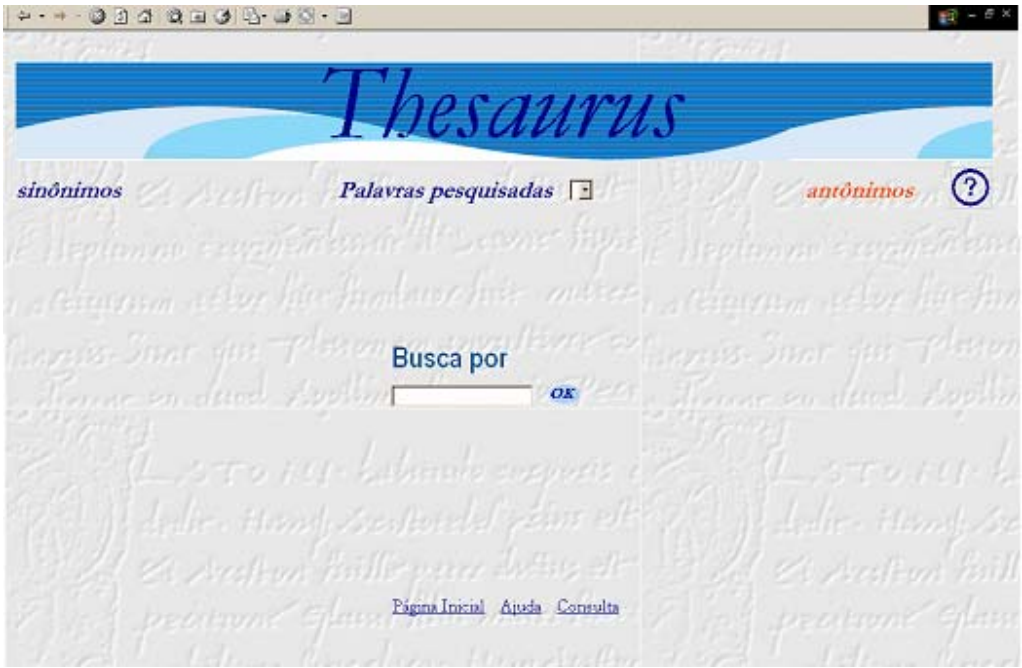

Figura 22 – Tela inicial do módulo de consulta aos dados do *Thesaurus*

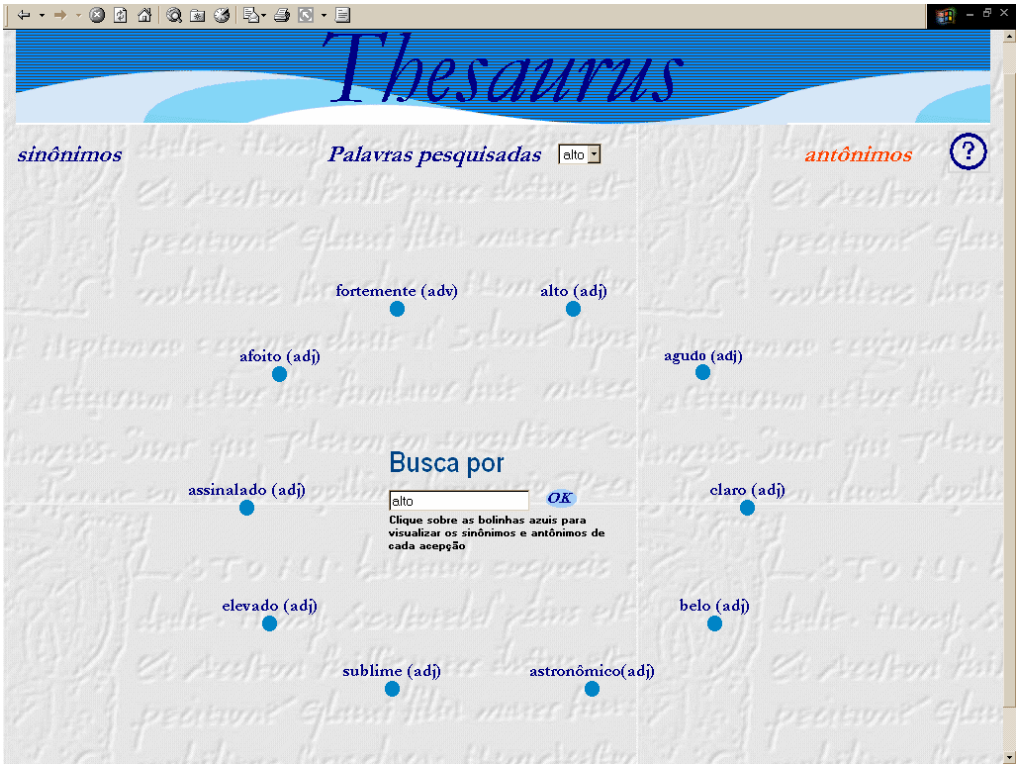

Figura 23 – Tela de apresentação dos resultados de uma consulta realizada

Figura 24 – Tela de apresentação do conjunto de sinônimos e antônimos de uma acepção

Figura 25 – Tela de Ajuda do módulo de consulta aos dados do *Thesaurus*

5.3 Módulo de consulta aos dados UNL

O módulo de consulta aos dados do dicionário UNL permite que o usuário visualize o conjunto de informações disponível para cada *headword* apresentada. Esse módulo tem acesso controlado, podendo ser acessado somente por pessoas autorizadas. Os dados não receberão nenhum tratamento especial, sendo apresentados ao usuário da forma como foram originalmente elaborados (Figuras 26 e 27).

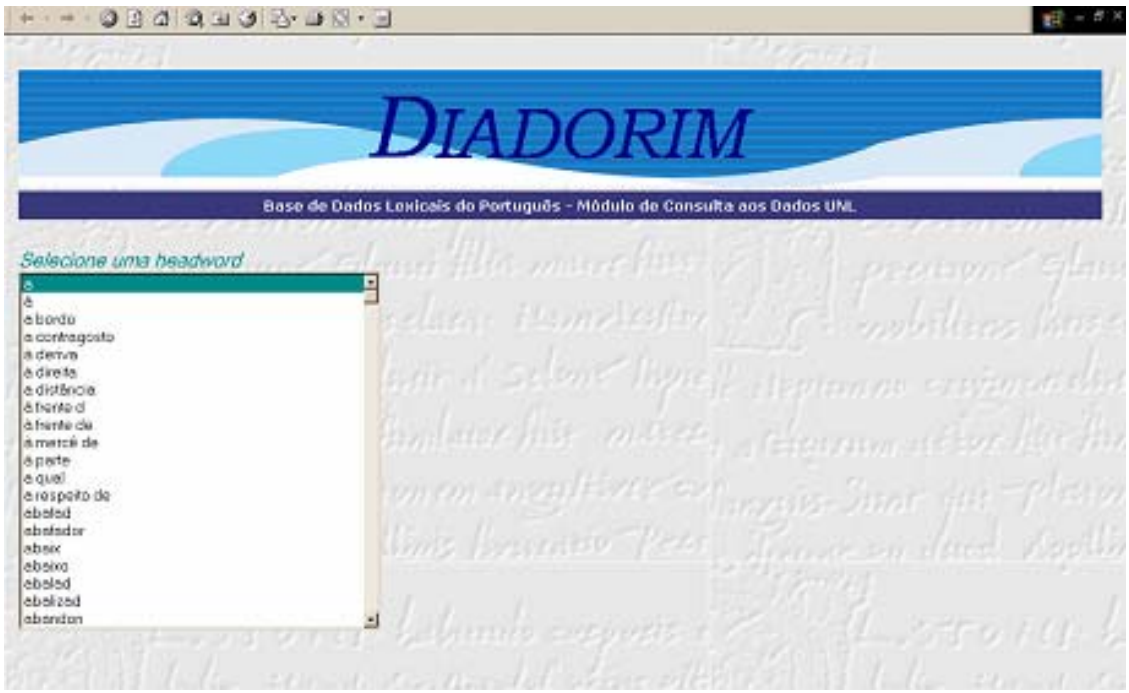

Figura 26 – Tela inicial de consulta aos dados da UNL

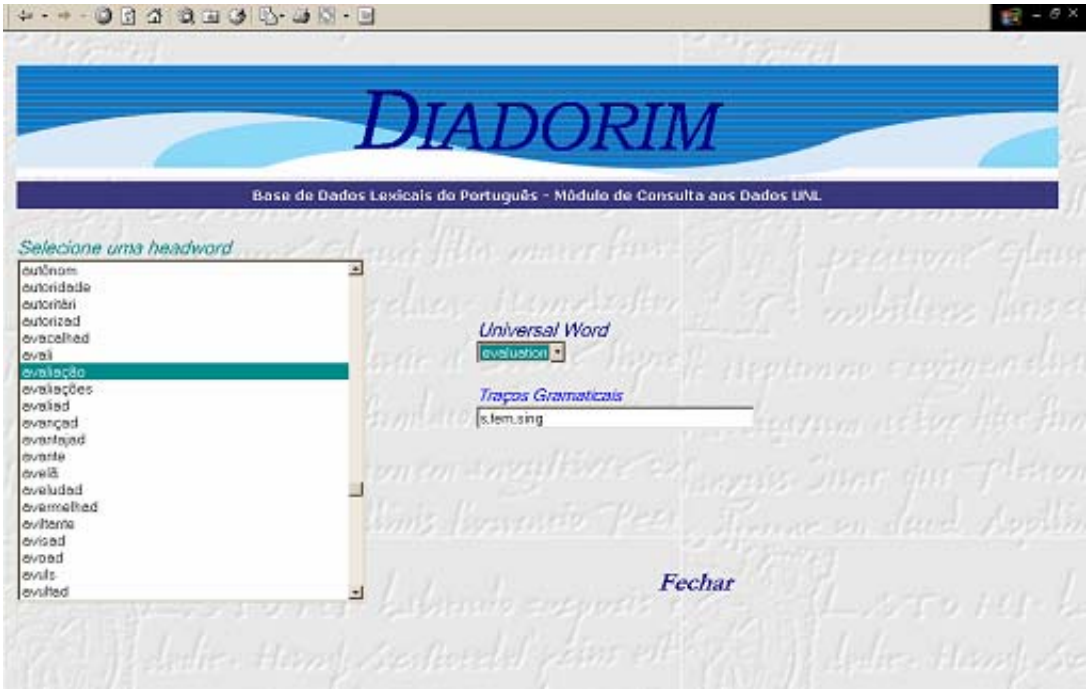

Figura 27 – Tela de apresentação dos resultados

5.4 Módulo de edição dos dados

Outra preocupação a respeito do acesso aos dados da DIADORIM foi desenvolver uma interface que permita a fácil alteração dos dados armazenados e, também, a inserção de novos dados, sem que fosse necessário que o usuário conhecesse a estrutura da base ou a forma como as informações devem ser distribuídas nas tabelas.

Para evitar que a base de dados seja acessada inúmeras vezes, a cada edição a ser realizada, todas as alterações são armazenadas em um arquivo e processadas no final de toda a tarefa. Esse módulo também tem acesso controlado para evitar a realização de alterações equivocadas, que comprometam a consistência dos dados (Figuras 28, 29 e 30).

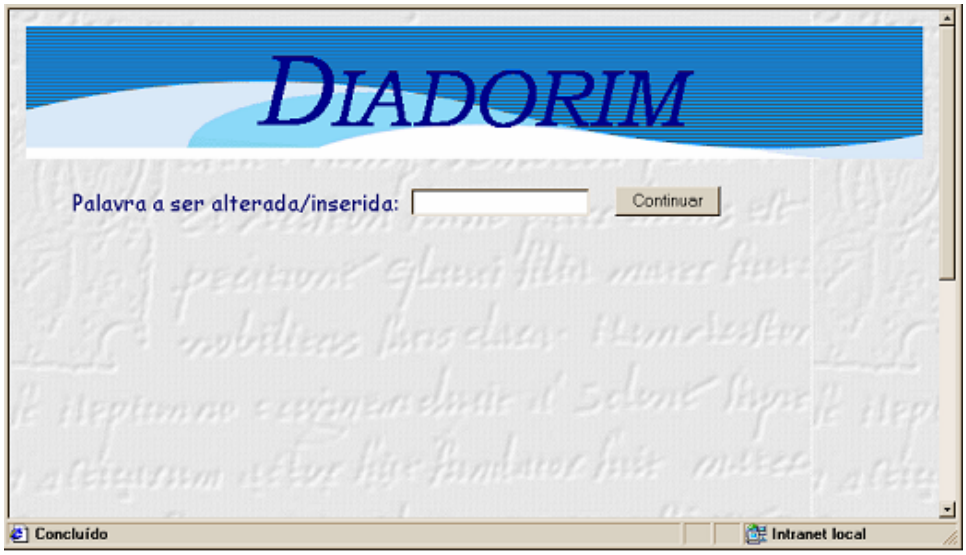

Figura 28 – Tela inicial

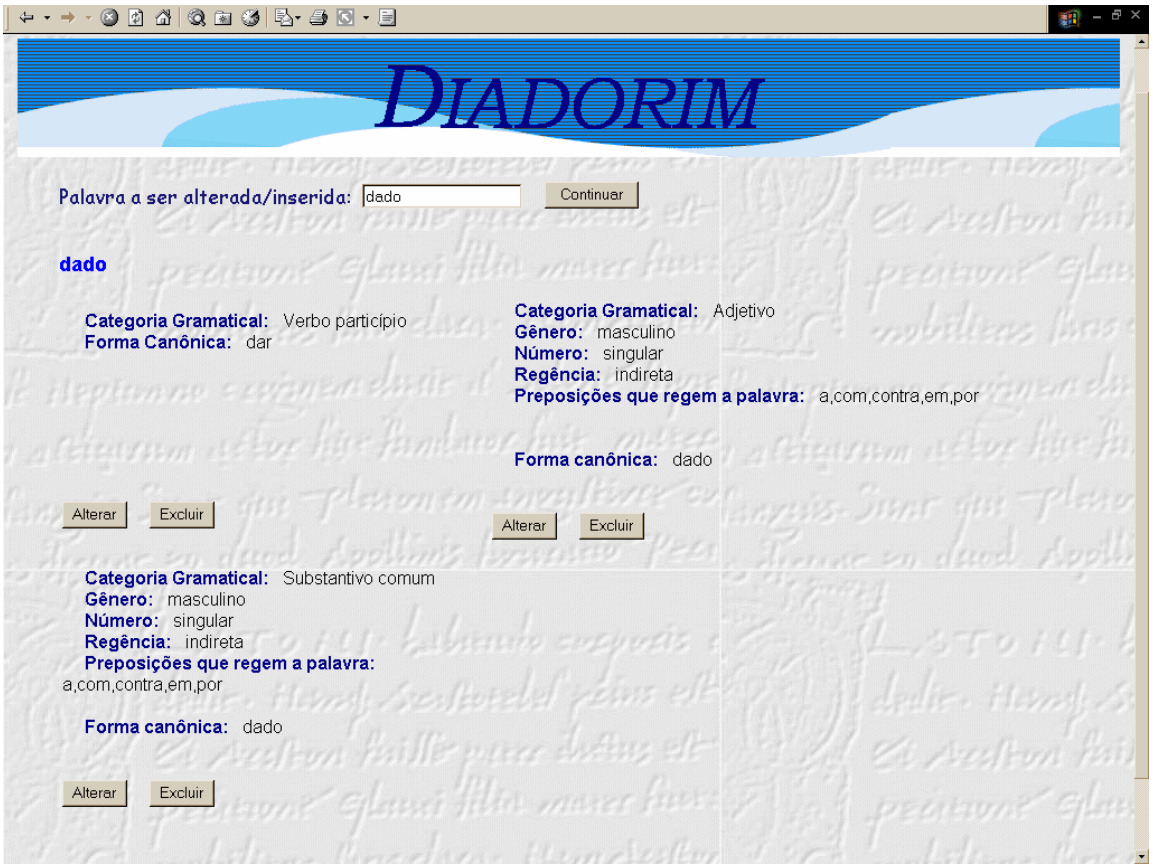

Figura 29 – Tela de apresentação de resultados

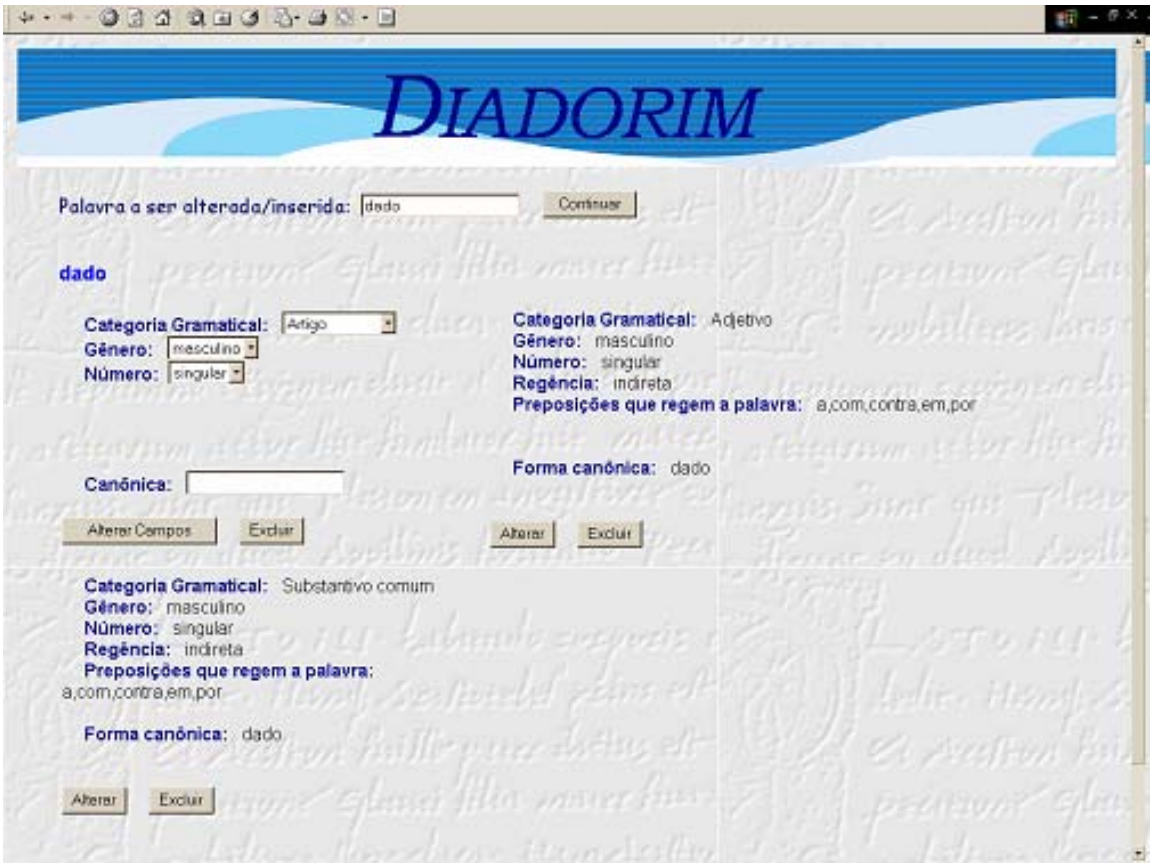

Figura 30 – Tela com campos possíveis de serem alterados

A Figura 29 apresenta todos os conjuntos de informações disponíveis para a palavra selecionada. O usuário deve escolher qual grupo de informações deseja alterar e, de acordo com a classe gramatical selecionada (Figura 30), será apresentado um conjunto de informações que pode ser alterado. Depois de alterar os campos desejados, o usuário pressiona o botão "Alterar Campos" e todas as informações que foram alteradas serão armazenadas.

5.5 Ferramenta de geração de listas especializadas

A DIADORIM foi projetada para servir como fonte de consulta e também como fonte de recursos para o desenvolvimento de novos aplicativos para PLN. Dessa forma, foi projetada e construída uma interface que permite ao usuário gerar listas de informações de acordo com sua necessidade.

A interface permite que o usuário escolha uma entre três tipos de listas possíveis:

a) lista de palavras da classe gramatical X, ou seja, todas as palavras classificadas em X, independentemente de pertencerem à outra classe gramatical.

Exemplo: caso o usuário deseje gerar uma lista de palavras pertencentes à classe "Substantivo", uma das palavras que iriam figurar nesta lista seria "canto", apesar de também estar classificada como flexão do verbo cantar e como adjetivo;

b) lista de palavras que pertencem SOMENTE à classe gramatical Y, ou seja, todas as palavras que só estão classificadas em Y. Neste caso, a palavra "casa", por exemplo, não estaria presente na lista, pois está classificada também como flexão do verbo casar.

c) lista de palavras que possam estar classificadas AO MESMO TEMPO nas classes gramaticais X, Y e Z, ou seja, palavras que pertencem à todas as classes especificadas.

Exemplo: se as classes gramaticais X, Y e Z fossem as classes Adjetivo, Substantivo e Verbo, esta lista seria composta por palavras que, necessariamente, pertencem às classes em questão, como é o caso de "machucado" e "partido".

Além do tipo de lista a ser gerada, o usuário pode, também, fazer restrições quanto ao gênero, número, e ao intervalo a que os itens lexicais devem pertencer, por exemplo, listar palavras maiores que "ar" e menores que "fri".

A seqüência de figuras 31, 32, 33 e 34 mostra um exemplo de interação em que deverá ser gerada uma lista com itens classificados como substantivos, do gênero feminino, sem qualquer restrição a respeito do número, e que estejam compreendidos no intervalo entre as letras h e p.

Figura 31 – Seleção do tipo de lista a ser gerada

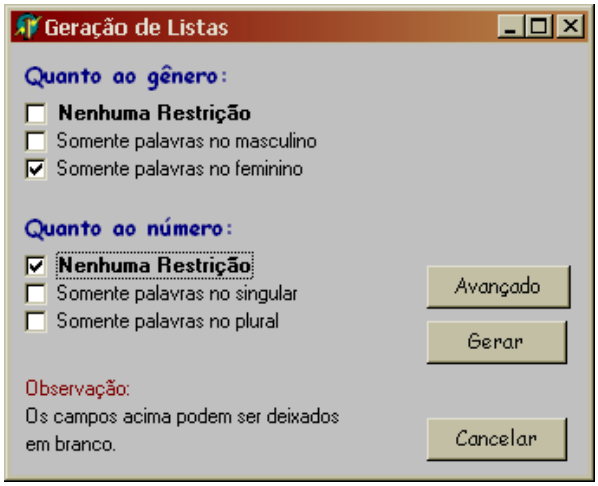

Figura 32 – Conjunto de Restrições Possíveis

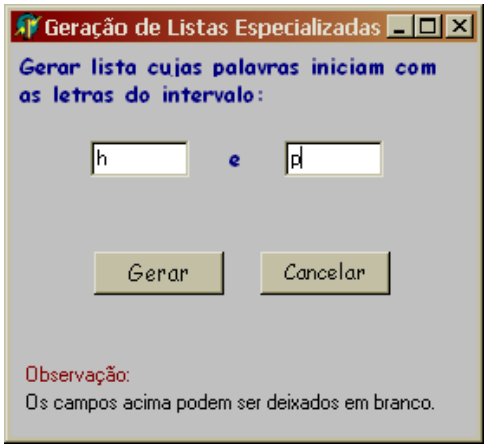

Figura 33 – Intervalo de Pertinência

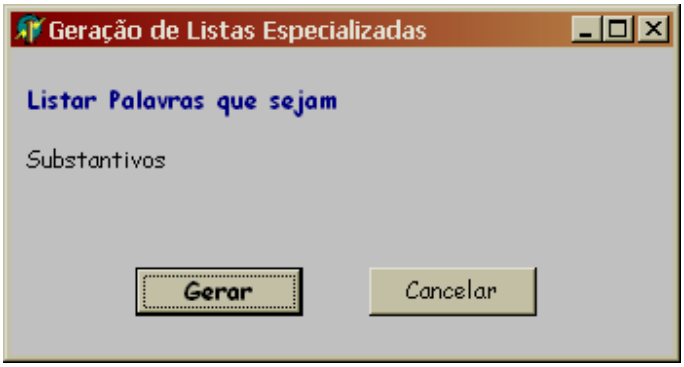

Figura 34 – Tela Final

Essas listas são arquivos tipo txt e as informações são expressas no mesmo formato usado para representar os dados do léxico.

Capítulo 6

Avaliação do Sistema

6.1 Avaliação da Interface de Acesso

A avaliação de um sistema não deve ser considerada uma etapa do processo de desenvolvimento e, se o tempo permitir, não deve ser realizada somente no final de todo o processo. Idealmente, o processo de avaliação deve ocorrer durante todo o ciclo de vida e os resultados devem ser usados como fonte de informação para realizar as modificações necessárias (DIX ET AL., 1998).

Os três principais objetivos de uma avaliação são:

- assegurar a funcionalidade do sistema
- analisar os efeitos que a interface tem sobre o usuário
- identificar qualquer problema específico com o sistema.

A avaliação das funcionalidades do sistema é muito importante, pois estas devem estar de acordo com as especificações do usuário. Isso inclui não somente disponibilizar a funcionalidade, mas fazê-lo de forma clara o suficiente para que o usuário saiba exatamente quais tarefas deve realizar para alcançar seus objetivos.

Outro fator importante é a análise do impacto que uma interface causa (ou pode causar) em um usuário. É importante considerar quão fácil é aprender a manipular o sistema, qual a relação do usuário em relação ao grau de dificuldade e, ainda, identificar áreas em que o sistema sobrecarrega o usuário de alguma forma, por exemplo, exigindo que grande quantidade de informação seja memorizada.

Para este trabalho foi proposta a avaliação dos módulos de consulta aos dados segundo dois métodos: a avaliação Heurística e a avaliação baseada em observação *Think Aloud*, descritos a seguir.

Avaliação Heurística: uma heurística é uma diretriz que pode ser usada para apoiar uma decisão ou criticar uma decisão tomada. A Avaliação Heurística[24], desenvolvida por Jakob Nielsen e Rolf Molich, é um método para estruturar a crítica a um sistema, usando um

[24] Informações detalhadas sobre o método apresentado podem ser obtidas http://www.useit.com/papers/heuristic/ (visitado em

conjunto de heurísticas genéricas. A idéia principal é que vários avaliadores critiquem, independentemente, a usabilidade do sistema. Nielsen e Molich definem um conjunto de dez heurísticas. Entretanto, para este trabalho, foi usado um subconjunto de sete heurísticas, apresentado a seguir.

1. **visibilidade do status do sistema**: o sistema deve sempre manter o usuário informado sobre o que está fazendo, em um tempo de resposta razoável.
2. **controle e liberdade do usuário**: as escolhas equivocadas do usuário levam o sistema a um estado indesejado. O sistema deve sempre oferecer, de forma clara e aparente, funções que permitam ao usuário desfazer/refazer uma determinada operação.
3. **consistência e padronização**: o sistema não deve usar palavras, ícones ou ações diferentes que tenham o mesmo significado. Ele deve seguir os padrões já estabelecidos.
4. **prevenção de erros**: o sistema deve, antes de ter mensagens de erros de boa qualidade, tentar evitar que os erros ocorram. Isso pode ser conseguido com uma modelagem cuidadosa, que previna e informe o usuário sobre as ações realizadas.
5. **equivalência entre o sistema e o mundo real**: o sistema deve comunicar-se com vocabulário familiar ao usuário, seguir as convenções do mundo real, apresentando as informações em uma ordem lógica e natural.
6. **projeto minimalista e estético**: os diálogos apresentados pelo sistema não devem conter informações irrelevantes ou raramente necessárias. Cada informação extra compete com as informações relevantes e diminui a visibilidade relativa.
7. **ajuda e documentação**: embora seja melhor que o sistema possa ser utilizado sem ajuda ou documentação, é necessário que tais informações estejam disponíveis. Essas informações devem ser de fácil entendimento e recuperação, listar os passos que podem ou devem ser seguidos pelo usuário para realizar determinada tarefa, e não devem ser muito extensas.

.02/08/2001)

Think Aloud: é um método simples, que se baseia na observação do modo como o usuário real interage com o sistema: o avaliador observa e "grava" as ações do usuário durante a interação. Entretanto, a observação pura não é suficiente para identificar quais as reais intenções do usuário ao realizar uma ação. Dessa forma, pede-se que o usuário descreva, em voz alta, todas as suas ações, o que ele pretende fazer e por que ele realizou a ação daquela forma.

As vantagens desse método são (DIX ET AL., *op cit.*):

- o processo é fácil de ser entendido e aprendido pelo avaliador;
- o usuário é encorajado a criticar o sistema;
- o avaliador pode identificar claramente os pontos confusos no momento em que ocorrem e, dessa forma, identificar as áreas problemáticas.

As interações dos usuários devem ser retidas em algum meio (fita cassete, por exemplo) para análise posterior.

Resultados obtidos com a Avaliação Heurística

Como explicado anteriormente, o método de avaliação proposto por Nielsen e Molich (1990),(MOLICH & NIELSEN, 1990) visa avaliar a usabilidade do sistema através das críticas feitas pelo usuário.

Para que a etapa de avaliação pudesse ser cumprida, foi necessário realizar um planejamento de todo o processo. Durante o planejamento da avaliação foi definido, com base nos trabalhos de Nielsen[25], o número de cinco usuários. Ele afirma, depois de vários experimentos realizados, que é razoável recomendar o número de cinco usuários e que o número exato de avaliadores só pode ser definido com uma análise de custo-benefício para o projeto. Dessa forma, neste trabalho, optamos por, inicialmente, usar cinco usuários para avaliar os módulos de consulta aos dados da DIADORIM. Caso a obtenção de novas informações com o último usuário se mostrasse alta, esse número seria aumentado.

A Figura 35 mostra o ganho de informações obtidas a partir dos experimentos de cada usuário.

[25] Explicações sobre o método de avaliação e sobre a determinação do número de usuários podem ser encontradas em http://www.useit.com/papers/heuristic/heuristica_evaluation.html

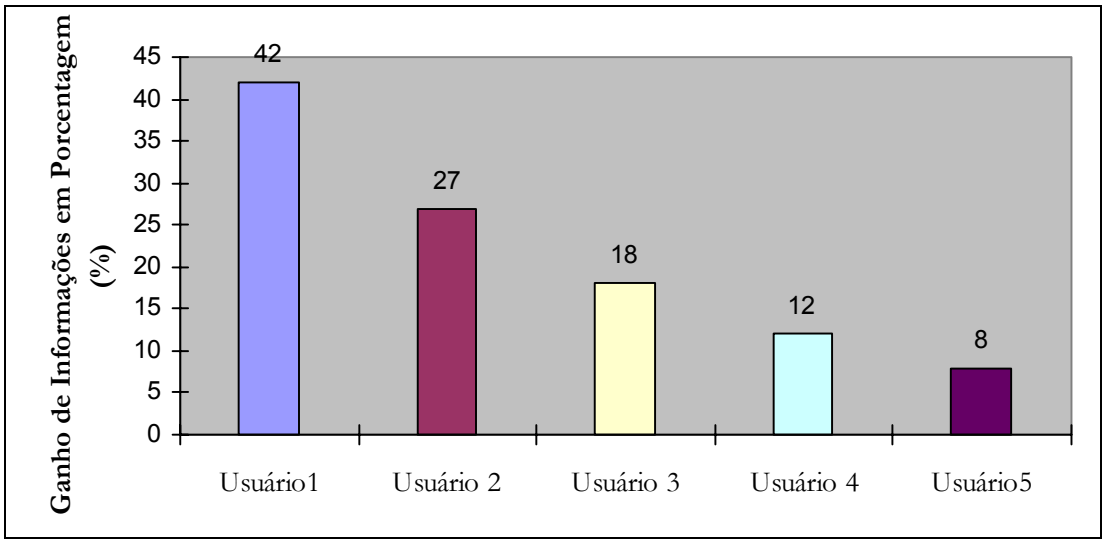

Figura 35 – Gráfico Ganho de Informações X Números de Usuários

Como a porcentagem de informações novas, obtidas com o experimento do usuário 5, não foi muito alta, decidiu-se pelo término dos experimentos.

Resultados obtidos

Após a análise dos dados, pôde-se perceber que vários pontos, representados pelo conjunto de heurísticas definido no Capítulo 5, apresentaram problemas, que serão relatados de maneira sucinta posteriormente. Uma das possíveis causas para alguns dos problemas apontados pelos usuários foi a definição do conjunto de requisitos de cada um dos módulos de forma bastante genérica. Isso fez com que, por exemplo, algumas possibilidades de consulta aos dados morfossintáticos não fossem previstas na fase de projeto do módulo e, quando definidas pelo usuário, o sistema não conseguiu tratá-las de forma adequada. Outro problema apontado foi o uso de palavras pouco familiares ao usuário, embora os termos fossem comuns para usuários ligados à área de PLN.

A seguir serão apresentados os problemas que todos os usuários apontaram durante a interação com a interface, divididos em três grupos: 1) problemas encontrados no módulo de consulta aos dados morfossintáticos, 2) problemas encontrados no módulo de consulta aos dados do *Thesaurus* e 3) problemas encontrados nos dois módulos. Os problemas apontados individualmente também foram considerados, mas não serão relatados aqui.

Os números apresentados em parênteses na descrição dos problemas encontrados em cada um dos módulos correspondem às heurísticas atingidas por cada problema.

Problemas encontrados no módulo de consulta aos dados morfossintáticos

- Demora em responder à primeira consulta realizada (1)
- Consultas não previstas levam o sistema a um estado indesejado, sem respostas adequadas ao usuário (1 e 4)
- Em caso de consultas restritas por categoria, o usuário não é avisado sobre a não existência de determinada classificação da palavra pesquisada (1)
- Não é possível realizar consultas com máscara com restrição de categoria. (4)
- O usuário não consegue cancelar uma consulta realizada (2)
- Caso o usuário realize várias consultas seguidas, antes que o resultado da consulta anterior seja apresentado, o sistema demora a responder à última consulta realizada (2)

Problemas encontrados no módulo de consulta aos dados do *Thesaurus*

- O usuário não é avisado sobre a inexistência do conjunto de sinônimos ou antônimos da palavra pesquisada (1)
- Não é permitido que o usuário realize consultas com máscara (2)
- Palavras usadas como rótulo das acepções encontradas causa confusão ao usuário (5)
- A forma de apresentação dos dados (em círculo) não é adequada (6)

Problemas encontrados em ambos os módulos

- Tamanho e cor de fonte não padronizados (6)
- Cores usadas nos botões de "OK" diferentes (2)
- Uso de palavras pouco familiares ao usuário (5)
- Mensagens de erro e advertência devem ser diferenciadas (6)
- Evitar rolagem vertical de tela (6)
- Habilitar a execução da busca pressionando a tecla "Enter" (2)
- Explicar termos desconhecidos na tela de ajuda (5)

Todos os problemas apontados foram analisados e estão sendo corrigidos em uma nova versão da interface de acesso aos dados. Alguns problemas apontados, como o uso de termos pouco familiares ao usuário será solucionado com a explicação de tais termos na tela de ajuda de cada módulo, pois o uso dos mesmos é fundamental em aplicações de PLN.

Resultados obtidos com *Think Aloud*

A avaliação *Think Aloud* foi realizada com usuários especialistas em PLN, que forneceram informações muito importantes no que diz respeito ao modo de apresentação das informações e na forma de controle das ações que deveriam ser realizadas pelo usuário durante a interação do mesmo com a interface.

Entre todas as informações que puderam ser obtidas, as mais relevantes foram:

- Alterar a ordem de apresentação das informações morfossintáticas de cada palavra de acordo com a classe gramatical em questão;
- Permitir que o usuário selecione a forma canônica de uma palavra consultada e esta palavra passe a ser nova entrada de busca;
- Permitir que o usuário visualize histórico de palavras consultadas no módulo de consulta aos dados morfossintáticos;
- Representar a rede de relações entre os itens consultados no *Thesaurus*, para ajudar o usuário a "abstrair" o sentido de cada palavra, já que não existem descrições sobre o significado de cada uma;
- Permitir o uso de máscaras na consulta aos dados do *Thesaurus;*
- Indicar ao usuário qual a prioridade entre as várias acepções de uma mesma palavra.

Os problemas e dificuldades encontrados pelos usuários foram, em geral, os mesmos apontados pelos usuários que realizaram a avaliação heurística. As sugestões particulares serão analisadas e aquelas que forem adequadas serão implementadas em uma nova versão dos módulos de consulta.

6.2 Avaliação de desempenho da DIADORIM em situações extremas

Os testes para avaliação do desempenho computacional da base de dados foram realizados de maneira bastante simples: foi implementado um pequeno procedimento que abre várias conexões com a base de dados e realiza várias operações simultaneamente. Esse teste têm como objetivo verificar o comportamento da base de dados - o tempo de resposta, se o sistema consegue responder a todas as solicitações, se fica instável ou se pára de responder - em situações em que vários usuários estariam conectados, realizando diversas operações.

Para este trabalho os testes foram divididos em três grupos, com 10, 20 e 30 conexões cada. Cada conexão executa um conjunto de operações de seleção simples, que variavam entre 80 e 130 operações. Os resultados obtidos são apresentados na Tabela 2. As colunas indicam: 1) número de usuários conectados simultaneamente, 2) quantidade de operações executadas por cada usuário, 3) tempo de resposta obtido para a execução do conjunto total de operações. Esses teste foram realizados em um máquina com processador Pentium II - MMX, 266 MHz, 128 Mb RAM.

Tabela 2 – Avaliação de desempenho da DIADORIM

Número de conexões	Quant. operações	Tempo de resposta (em média)
10	80	00:01:33
10	110	00:01:38
10	130	00:02:15
20	80	00:03:14
20	110	00:04:28
20	130	00:05:14
30	80	00:04:53
30	110	00:06:41
30	130	00:07:52

Apesar de os tempos de resposta obtidos serem elevados não nos pareceram excessivos dado o cenário utilizado: vários usuários conectados, realizando um grande conjunto de operações simultaneamente, além do tamanho da base. Num cenário real, mesmo que vários usuários acessem a interface de consulta, o número de operações que cada um estará

realizando, simultaneamente, deverá ser bem inferior ao conjunto proposto para o teste. Além disso, o tempo de resposta obtido pelo usuário da interface de consulta dependerá de outros fatores, por exemplo, o tráfego de dados pela rede, a velocidade da conexão, entre outros.

Dessa forma, o desempenho da Diadorim superou as expectativas, uma vez que o sistema não ficou instável, não deixou de produzir respostas e não provocou travamento em qualquer dos casos.

Conclusões e Trabalhos Futuros

O desenvolvimento de um recurso computacional como o apresentado neste trabalho é um processo complexo e demorado, que exige uma análise cuidadosa dos dados a serem integrados, as relações existentes entre eles e, acima de tudo, a melhor maneira de integrá-los, para evitar que haja qualquer tipo de perda de informação.

A colaboração entre lingüistas e cientistas de computação foi essencial para o desenvolvimento do trabalho, uma vez que seria muito difícil, sem o apoio de um especialista, desenvolver uma estrutura para a organização dos dados de forma tão adequada aos propósitos do trabalho. Além disso, o entendimento de muitas questões referentes às diferenças entre os modelos que estavam sendo integrados só foi possível com a intervenção de especialistas.

Do ponto de vista de banco de dados, foi um trabalho bastante desafiador: o uso do modelo relacional para desenvolver uma aplicação em que os dados armazenados podem ser relacionados hierarquicamente entre si, e a manipulação de um volume de dados tão significativo fizeram com que o modelo de dados e o sistema de gerenciamento fossem escolhidos com cautela, já que o desempenho da base de dados é um fator de grande importância neste trabalho. Como há módulos de acesso aos dados via Web, o tempo de resposta do sistema às consultas realizadas é de grande importância e as escolhas realizadas satisfizeram os requisitos estabelecidos no início do projeto.

Todas as dificuldades encontradas durante o projeto tornaram o trabalho ainda mais interessante e desafiador. Em particular, destacam-se a natureza e o tamanho da base. A natureza, lingüística, requereu uma modelagem cuidadosa que contou com o apoio de lingüistas, e que previu inclusões de informações ainda não disponíveis, mas que certamente enriquecerão este recurso. O tamanho da base, por outro lado, se mostrou muito maior do que os similares da literatura, fazendo com que as decisões computacionais mais tradicionais (modelo relacional) fossem preferidas às apregoadas na literatura mais recente (features, OO). Certamente as circunstâncias que motivaram este projeto - dados já existentes e compromissados com outros aplicativos - também foram determinantes por ocasião das

tomadas de decisão. Privilegiou-se, além dos compromissos acima citados, eficiência, segurança, funcionalidade e extensibilidade.

Trabalhos para um futuro próximo incluem a implementação das modificações sugeridas pelos usuários julgadores e, principalmente quanto ao módulo Thesaurus, sua extensão quanto a exemplos de uso relativos a cada acepção, e a adoção e representação de uma ontologia que acrescente mais informação semântica à base. Tais tarefas são, no entanto, complexas e volumosas, razão pela qual não foram implementadas por ocasião deste projeto.

Finalmente, acreditamos que a disponibilização da Diadorim na Web contribui para um maior acesso do usuário comum a recursos lingüísticos do português do Brasil até então não reunidos num único aplicativo.

Referências Bibliográficas

BECHARA, E. (1976). Moderna gramática portuguesa. São Paulo: Companhia Editora Nacional.

BENVENISTE, E. (1966). Les niveaux de l'analyse linguistique. In Problèmes de linguistique générale. Paris: Gallimard.

CHOMSKY, N. (1970). Remarks on nominalization. In Jacobs, R. A. & Rosenbaum, P. (Eds.) Readings in English Transformational Grammar. Waltham, Mass: Ginn and Company.

CHOMSKY, N. & LASNIK, H. (1977). Filters and control. Linguistic Inquiry, 8:3, 425-504.

CUNHA, C. & CINTRA, L. (1985). Nova gramática do português contemporâneo. Rio de Janeiro: Nova Fronteira.

DIAS-DA-SILVA, B.C. ET AL. (2000) Construção de um Thesaurus para o Português do Brasil In: *Encontro para o Processamento da Língua Portuguesa Escrita e Falada*, *5*. Pp 1-11, Outubro, Atibaia.

DIX, A.; FINLAY, J.;ABOWD, G.; BEALE, R. (1999) *Human-Computer Interaction*, 2ed. Prentice Hall Europe.

EAGLES (1993). EAGLES Lexicon Architeture – EAGLES Document EAG-CLWG-LEXARCH/B.
Disponível em 09/10/2000 (http://www.ilc.pi.cnr.it/EAGLES96/lexarch/lexarch.html)

ELMASRI, R.; NAVATHE, S.B. (2000) *Fundamentals of Database Systems* Addison Wesley, 3ed.

EVANS, R.; KILGARRIFF, A. (1995) MRDs, Standards and How To Do Lexical Engineering In: *Language Engineering Convention, II Proceeding*, pp 125-132. Londres, Outubro.

FELLBAUM, C. (1999) *WordNet – An Eletronic Lexical Database*, MIT Press, Massachusetts.

GREGHI, J.G.; MARTINS, R.T.; NUNES, M.G.V. (2001) O processo e desenvolvimento da BDL-NILC, *Série de Relatórios do Núcleo Interinstitucional de Lingüística Computacional,* nº 152, NILC-TR-01-7, Outubro.

IDE, N.; VÉRONIS, J. (1992) Modeling Lexical Databases. In: *International Conference ALLC-ACH' 94*. Oxford.

IDE, N.; LE MAITRE, J.; VÉRONIS, J. (1993) Outline of a Model for Lexical Databases *Information Processing & Management,* 29(2), pp159-186. Disponível em 09/10/2000 (http://www.up.univ-mrs.fr/~veronis/publis.html)

JACKENDOFF, R. (1983). *Semantics and cognition*. Cambridge, MASS: The MIT Press.

JANSZ,K. (1998) *Intelligent processing, storage and visualisation of dictionary information.* Sydney, Austrália, Tese – Universidade de Sydney.

KAPLAN, R.; BRESNAN, J. (1982) Lexical-functional grammar: A formal system for grammatical representation. *Mental Representation of Grammatical Relations.* Cambridge, Massachussets: MIT Press.

KATZ, J.; FODOR, J. (1963). *The structure of a semantic theory*. Language, 39, 170-210.

MARSLEN-WILSON, W. D.; TYLER, L. K. (1980). *The temporal structure of spoken language understanding. Cognition*, 8, 1-71.

MARSLEN-WILSON, W. D.; WELSH, A. (1978). Processing interactions and lexical access during word recognition in continuous speech. *Cognitive Psychology*, 10, 29-63.

MARTINS, R.T.; HASEGAWA, R.; NUNES, M.G.V.; MONTILHA, G.; OLIVEIRA Jr., O.N. (1998a) Linguistic issues in the development of ReGra: a Grammar Checker for Brazilian Portuguese. *Natural Language Engineering.* Volume 4 (Part 4 December 1998): p287-307;Cambridge University Press.

MARTINS, R.T.; RINO,L.H.M; NUNES, M.G.V. (1998b) As Regras Gramaticais para a Decodificação UNL-Português no Projeto UNL, *Série de Relatórios do Núcleo Interinstitucional de Lingüística Computacional*, NILC-TR-98-1, nº 67, Fevereiro.

MEDIN, D. L.; ORTONY, A. (1989). Psychological essentialism. In. S. Vosniadou and A. Ortony (Eds.) Similarity and analogical reasoning. New York: Cambridge University Press.

MEDIN, D. L.; SCHAFFER, M. M. (1978). Context theory of classification learning. Psychological Review, 85, 207-238.

MICROSOFT (1995) Microsoft SQL Server - Manuais impressos do aplicativo.

MURPHY, G. L.; MEDIN, D. L. (1985). The role of theories in conceptual coherence. Psychological Review, 92, 289-316.

MILLER, G. A.; BECKWITH, R.; FELLBAUM, C.; GROSS, D.; MILLER, K. (1993). *Introduction to WordNet: An On-line Lexical Database.* ftp://ftp.cogsi.princeton.edu/pub/wornet/5papers.ps (visitado em 07/1999).

MOLICH, R., NIELSEN, J. (1990). Improving a human-computer dialogue, *Communications of the ACM* **33**, 3 (March), 338-348.

NIELSEN, J., MOLICH, R. (1990). Heuristic evaluation of user interfaces. *Proc. ACM CHI'90* (Seattle, WA, 1-5 April), 249-256.

NUNES, M.G.V. ET AL. (1996). *A Construção de um Léxico da Língua Portuguesa do Brasil para Suporte à Correção Automática de Textos*. Rel. Técnico do ICMC, No. 42. ICMC/USP – São Carlos, Agosto.

NUNES, M.G.V. et. al (2000) Relatório Técnico Parcial do Programa PADCT-III-CDT/MCT: *Revisor Gramatical e Ferramentas de Auxílio à Escrita*. Processo RC: 3.1.3-0012/98, Convênio 8.898.0591.00. Fevereiro.

NUNES, M.G.V. et. al (2001) Relatório Técnico Final do Programa PADCT-III-CDT/MCT: *Revisor Gramatical e Ferramentas de Auxílio à Escrita*. Processo RC: 3.1.3-0012/98, Convênio 8.898.0591.00. Dezembro.

OLIVEIRA JR., O. N.; MARTINS, R. T.; RINO, L. H. M.; NUNES, M. G. V. (2001) O uso de interlíngua para comunicação via Internet: O Projeto UNL/Brasil. *Série de Relatórios do Núcleo Interinstitucional de Lingüística Computacional,* NILC-TR-01-3, Julho, 14p

SHIEBER, S.M. (1986) *An Introduction to Unification-based Approaches to Grammar* CSLI Lecture Notes Series, Chicago: University of Chicago Press.

REICHENBACH, H. (1947). *Elements of symbolic logic*. Berkeley, CA: University of California Press.

RINO, L.H.M.; MARTINS, R.T.; MARCHI, A.R.; KUHN, D.C.S.; PINHEIRO, G.M.; PARDO, T.A.S.; DI FELIPPO, A.; NUNES, M.G.V. (2001) Projeto TraSem: A Investigação teórica sobre o Problema da Ambigüidade Categorial. *Série de Relatórios do Núcleo Interinstitucional de Lingüística Computacional,* nº 142, NILC-TR-01-1, Abril.

ROCHA LIMA, C. H. (1972). *Gramática normativa da língua portuguesa*. Rio de Janeiro: Livraria José Olympio.

ROSCH, E. H. (1973). On the internal sctructure of perceptual and semantic categories. In T. E. Moore (Ed.) Cognitive development and the acquisition of language. New York: Academic Press. pp. 111-144.

ROSCH, E. H. (1975). Cognitive representation of semantic categories. *Journal of Experimental Psychology*: General, 104, 192-133.

UCHIDA, H.; ZHU, MEIYING; DELLA SENTA, T. (1999). Universal Networking Language: a gift for a millenium. Tokyo: United Nations University.

ZAJAC, R; (1998). The Habanera Lexical Knowledge Base Management System – In: *International Conference on Language Resources And Evaluation, 1*. Granada, Spain, 28-30 May.

WITTMAN, L.H.; RIBEIRO, R.D. (1998) Recursos lingüísticos e processamento morfológico do Português: o PALAVROSO e o projeto LE-PAROLE. In: *Encontro para Processamento Computacional da Língua Escrita e Falada, 3.,* PUCRS, Porto Alegre, Brasil, p. 109-117.

URLs Utilizadas[26]

Informações sobre etiquetadores
http://www.ling.lancs.ac.uk/monkey/ihe/linguistics/corpus2/2types.htm
(visitado em 30/01/2001)

WordNet
http://www.cogsci.princeton.edu/~wn (visitado em 23/01/2002)
ftp://ftp.cogsi.princeton.edu/pub/wornet/5papers.ps (visitado em 07/1999).

EuroWordNet
http://www.hum.uva.nl/~ewn (visitado em 26/01/2002)

Projeto Corelli
http://crl.nmsu.edu/Research/Projects/corelli/index.html (visitado em 30/11/2001)

Human Text Initiative (HTI)
http://www.hti.umich.edu (visitado em 30/11/2001)

Text Encoding Initiative (TEI)
http://www.hti.umich.edu/t/tei/ (visitado em 30/11/2001)

Expert Advisory Group on Language Engineering Standards (EAGLES)
http://www.ilc.pi.cnr.it/EAGLES96/home.html (visitado em 30/11/2001)

Projeto Parole
http://www.hltcentral.org/parole (visitado em 10/01/2002)

Thesaurus Eletrônico para o Português do Brasil
http://143.107.183.175/site2001/tools/tep.htm

Universal Networking Language (UNL)
http://143.107.183.175/site2001/projetos/unl.htm

Avaliação Heurística
http://www.useit.com/papers/heuristic/ (visitado em 02/08/2001)

Como Conduzir uma Avaliação Heurística
http://www.useit.com/papers/heuristic/heuristica_evaluation.html (visitado em 02/08/2001)

Informações sobre o SQL Server
http://www.microsoft.com/sql

[26] As datas indicam a última consulta realizada à URL indicada

Apêndice A
Detalhes sobre a implementação da DIADORIM

Mapeamento

O primeiro passo dado em direção à implementação da base de dados foi o mapeamento realizado a partir do Diagrama Entidade-Relacionamento para o Modelo Relacional. O próximo passo foi realizar a normalização das tabelas para evitar a redundância de informações. Vale dizer que, para algumas tabelas, por motivos de implementação, a normalização realizada não foi obedecida. A Figura 1 apresenta o mapeamento realizado para a DIADORIM.

CLASSIFICAÇÃO
(Codigo , Canonica)

CONJUNCAO
(Codigo , Papel)

E_SINONIMO
(Lexema1, Lexema2, Acepcao)

E_ANTONIMO
(Lexema1, Lexema2, Acepcao)

GRUPO1
(Codigo , Gênero, Número)

GRUPO1A
(Codigo , Classe , Tipo)

GRUPO2
(Codigo , Tipo)

GRUPO2A
(Codigo, Tref, Tev, Modo, Pessoa)

GRUPO3
(Codigo, Classe, Tipo)

MORFOLOGIA1
(Codigo , Componentes,Tipo, Atributos)

MORFOLOGIA3
(Codigo , Componentes, Tipo, Atributos)

PALAVRA
(Lexema)

PRONOME_PESSOAL
(Codigo, Pessoa , Tonicidade)

SINTAXE1
(Codigo , Regencia, ListPrep)

SINTAXE1A
(Codigo , EstrArg)

SINTAXE2
(Codigo , ListPrep)

SINTAXE2A
(Codigo , Regencia)

SINTAXE2B
(Codigo , EstrArg)

VERBO
(Codigo , Pessoa)

Figura 1 – Mapeamento realizado para a definição das tabelas da DIADORIM

Estimativa de tamanho

Para criar uma base de dados é necessário definir um arquivo que armazenará os dados. Além do arquivo de dados, referenciado como *Data Device* no SQL Server, é necessário criar

um arquivo que irá armazenar os logs de todas as transações realizadas na base (*Log Device*). Esses arquivos devem ter seus tamanhos definidos no momento de sua criação, sendo necessário, dessa forma, realizar uma estimativa sobre o tamanho que a base de dados deverá assumir. Esse cálculo deve ser realizado da seguinte forma (MICROSOFT, 1995)[27]:

- Definiu-se o tamanho máximo de um registro da tabela.

  Subtotal = soma dos bytes ocupados por cada campo de tamanho fixo + soma dos bytes ocupados por cada campo de tamanho variável + 2 bytes

  Tamanho do registro = Subtotal + ((Subtotal/256) + 1) + (Número de colunas de tamanho variável + 1) + 2

- Calculou-se o número de páginas de dados usadas[28].

$$\text{Número de registros por página} = 2016 / \text{Tamanho do registro}$$

$$\text{Número de páginas de dados necessárias} = \frac{\text{Número máximo de registros da tabela}}{\text{Número de registros por página}}$$

- Calculou-se o tamanho dos registros de índice.

Para cada registro, há um *overhead* de 5 bytes. Deve-se efetuar o seguinte cálculo:

Subtotal = Soma dos bytes dos campos de índice com tamanho variável + Soma dos bytes dos campos de índice com tamanho variável + 5

Tamanho do registro de índice = Subtotal + ((Subtotal/256) + 1) + (Número de colunas de tamanho variável + 1) + 2

- Calculou-se o número de páginas indexadas.

Número de registros indexados por página = 2016/ Tamanho do registro indexado – 2

$$\text{Número de páginas de índice de nível 0} = \frac{\text{Número de páginas de dados}}{\text{Número de registros indexados por página}}$$

* Como o valor obtido foi maior que 1, a divisão prosseguiu usando o resultado como próximo dividendo até que o resultado fosse 1.

[27] Os dois bytes adicionados aos cálculos realizados são o *overhead* usado pelo SQL Server para armazenamento interno.

[28] O SQL Server usa páginas de recuperação de dados de tamanho 2K. Cada página usa 32 bytes de *overhead*. Dessa forma, o cálculo do número de páginas de dados usadas deve ser realizado usando o valor 2016 (2048 – 32=2016)

$$\text{Número de páginas de nível 1} = \frac{\text{Número de páginas de índice de nível 0}}{\text{Número de registros indexados por página}}$$

$$\text{Número de páginas de nível 2} = \frac{\text{Número de páginas de nível 1}}{\text{Número de registros indexados por página}}$$

Finalmente, foi realizada a soma do total de páginas de dados necessárias com o total de páginas de todos os níveis e o resultado será o número de páginas, de tamanho 2K, necessárias para a tabela analisada. Mais detalhes e exemplos podem ser obtidos em (MICROSOFT, *op cit.*) Esse processo foi repetido para todas as tabelas e, dessa forma, pôde-se estimar um valor máximo da base de dados: 396Mb.