Universidade Estadual de Campinas
Faculdade de Engenharia Elétrica e de Computação
Departamento de Comunicações

Modelos da Língua para o Português do Brasil Aplicados ao Reconhecimento de Fala Contínua: Modelos Lineares e Modelos Hierárquicos (*Parsing*)

por

Luis Augusto de Sá Pessoa
Eng. Eletricista (UFPE, 1996)

Orientador: Prof. Dr. Fábio Violaro
DECOM – FEEC - UNICAMP
Co-orientador: Prof. Dr. Plínio A. Barbosa
IEL – UNICAMP

Dissertação apresentada à Faculdade de Engenharia Elétrica e de Computação da UNICAMP como requisito parcial para a obtenção do título de Mestre em Engenharia Elétrica.

Campinas, 24 de fevereiro de 1999

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

P439m

Pessoa, Luis Augusto de Sá
Modelos da língua para o português do Brasil aplicados ao reconhecimento de fala contínua: modelos lineares e modelos hierárquicos (*Parsing*). / Luis Augusto de Sá Pessoa.--Campinas, SP: [s.n.], 1999.

Orientador: Fábio Violaro, Plínio A. Barbosa
Dissertação (mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e Computação.

1. Reconhecimento automático da voz. 2. Modelos linguísticos. 3. Gramática gerativa. I. Violaro, Fábio. II. Barbosa, Plínio A. III. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. IV. Título.

Resumo

O reconhecimento de fala contínua baseado somente em informações acústicas não proporciona bons resultados [Lippmann97]. Modelos da Língua podem ser usados para caracterizar, capturar e explorar as regularidades da língua natural, melhorando o desempenho do sistema de reconhecimento.

Esta Tese apresenta o estudo e a implementação de Modelos da Língua para o português do Brasil. São propostos dois modelos *bigram* de classes de palavras (modelos lineares) e um modelo baseado em gramática independente de contexto (modelo hierárquico). Um dos modelos bigram emprega classificação manual de palavras (classes gramaticais) e o outro emprega classificação automática através do algoritmo Simulated Annealing. O modelo baseado em gramática foi desenvolvido com base em uma teoria de Gramática Gerativa [Chomsky65] e utiliza a Sintaxe X-barra [Jackendoff77].

Os Modelos da Língua foram avaliados através de um sistema de reconhecimento de fala contínua dependente do locutor desenvolvido por Morais [Morais97].

Este trabalho também apresenta um algoritmo de decodificação, baseado no algoritmo *Level Building* [Rabiner*85], que leva em conta as restrições do Modelo da Língua durante o procedimento de busca.

Abstract

Continuous speech recognition based only on acoustic information does not provide good results [Lippmann97]. Language Models can be used to characterize, capture and exploit the language regularities, improving the recognition system performance.

This Thesis presents the study and implementation of Language Models for Brazilian Portuguese. Two word class bigram language models (linear models) and one model based on context-free grammar (hierarchical model) are proposed. One bigram model uses manual classification of words (grammatical classes) and the other one uses automatic classification performed by the Simulated Annealing algorithm. The grammar-based model is implemented employing a theory of Generative Grammar [Chomsky65] and uses the X-bar Syntax [Jackendoff77].

The Language Models are evaluated using a speaker dependent continuous speech recognition system developed by Morais [Morais97].

This work also presents a decoding algorithm, based on Level Building algorithm [Rabiner*85], which considers the Language Model constraints during the search procedure.

CIÊNCIA E TEMPERANÇA

"*E à ciência, a temperança; à temperança, a paciência; à paciência, a piedade*." - (II PEDRO, 1:6.)

Quem sabe precisa ser sóbrio.

Não vale saber para destruir.

Muita gente, aos primeiros contactos com a fonte do conhecimento, assume atitudes contraditórias. Impondo idéias, golpeando aqui e acolá, semelhantes expositores do saber nada mais realizam que a perturbação.

É por isso que a ciência, em suas expressões diversas, dá mão forte a conflitos ruinosos ou inúteis em política, filosofia e religião.

Quase todos os desequilíbrios do mundo se originam da intemperança naqueles que aprenderam alguma coisa.

Não esqueçamos. Toda ciência, desde o recanto mais humilde ao mais elevado da Terra, exige ponderação. O homem do serviço de higiene precisa temperança, a fim de que a sua vassoura não constitua objeto de tropeço, tanto quanto o homem de governo necessita sobriedade no lançamento das leis, para não conturbar o espírito da multidão. E não olvidemos que a temperança, para surtir o êxito desejado, não pode eximir-se à paciência, como a paciência, para bem demonstrar-se, não pode fugir à piedade, que é sempre compreensão e concurso fraternal.

Se algo sabes na vida, não te precipites a ensinar como quem tiraniza, menosprezando conquistas alheias. Examina as situações características de cada um e procura, primeiramente, entender o irmão de luta.

Saber não é tudo. É necessário fazer. E para bem fazer, homem algum dispensará a calma e a serenidade, imprescindíveis ao êxito, nem desdenhará a cooperação, que é a companheira dileta do amor.

Emmanuel

(Livro “Vinha de Luz”, pelo médium Francisco Xavier)

Aos meus avôs e avós,

Luís, Amélia, José e Maria.

Agradecimentos

Agradeço ao Bom Deus que me amparou em todas as horas e me inspirou nos momentos difíceis, dando-me a oportunidade de receber o apoio dos amigos e de pessoas que nem conhecia. Agradeço a Ele por permitir que eu viesse tão longe de casa realizar um sonho e crescer como pessoa.

Agradeço aos meus pais, Edmilson e Célia, e aos meus irmãos, que sempre acreditaram em mim e, com muito amor, pouparam-me de tantas preocupações, sempre buscando incentivar-me a continuar. Meus agradecimentos à minha tia Socorro e aos meus primos pelo amor que me deram, não somente durante esses dois anos, mas ao longo da minha vida.

Agradeço aos meus orientadores Fábio Violaro e Plínio Barbosa, pela paciência, pelas conversas, por acreditarem em minhas propostas, e por muito me ajudarem com conhecimentos e orientações que foram essenciais ao desenvolvimento do trabalho. Quero agradecer também à prof. Charlotte pela colaboração indispensável, pela simpatia, e pela compreensão das nossas limitações.

Quero agradecer a amiga Noêmia pela paciência em esclarecer minhas dúvidas e pelas agradáveis conversas, e também a todas as outras pessoas da administração que colaboraram para a realização deste trabalho.

Agradeço aos queridos amigos Fabrício, Aldvan, Raquel, Rony, Leonardo, Flávio, Masato, Irene, Paula Sampaio, Fábio César, Antônio Marcos, Lôla, Luzia e família, Kátia e Carlos que a todo momento me auxiliaram, seja no trabalho de pesquisa com sugestões e debates, seja alegrando-me e fortalecendo-me nos momentos difíceis, tornando minha caminhada sempre mais suave.

Meus agradecimentos também aos amigos e a todos aqueles que não foram citados aqui, mas que colaboraram ativamente durante esses dois anos.

Quero também agradecer o incentivo da CAPES que financiou a minha tese de mestrado.

Sumário

Lista de Figuras

Lista de Tabelas

1 Introdução

1.1 Reconhecimento de Fala Contínua e Modelos da Língua

Um dos objetivos do reconhecimento de fala é construir dispositivos que transcrevam a fala em texto automaticamente. No caso de alguns sistemas, a transcrição pode não ser o objetivo final, e sim, uma etapa intermediária num complexo processo de compreensão da fala, possivelmente terminando com ações em resposta ao que foi dito. Esta tecnologia proporciona aplicações como comunicação *hands-free* e *eyes-free* com computadores e outras máquinas, acesso remoto a base de dados e sistemas de tradução automática.

A pesquisa em reconhecimento de fala distribui-se principalmente entre três áreas: reconhecimento de palavras isoladas, reconhecimento de fala contínua e compreensão da fala.

No reconhecimento de fala contínua, o reconhecimento é realizado sobre uma seqüência de palavras (frase) pronunciadas naturalmente e sem pausas.

Apesar de já terem sido desenvolvidos diversos sistemas de reconhecimento de fala contínua (vide [Lippmann97]), eles estão longe da capacidade do ser humano de lidar com as diferenças de pronúncia, presença de ruído, frases gramaticalmente incorretas, ou mesmo, palavras desconhecidas. Além disso, quase sempre o objetivo do homem é compreender aquilo que foi dito e, neste ponto, concorrem diversas fontes de conhecimento: conhecimento da sintaxe e semântica da língua, conhecimento do assunto falado, referência a frases anteriores, etc.

Outros problemas também dificultam o reconhecimento de fala contínua: o número de palavras do enunciado é normalmente desconhecido, a fronteira entre palavras é desconhecida (para não dizer incerta) e pode haver a ocorrência de efeitos de coarticulação entre palavras.

O reconhecimento de fala pode ser visto como um problema de busca, durante o qual muitas hipóteses (seqüências de palavras) são criadas e expandidas até que a frase completa seja reconhecida. O reconhecimento baseado somente em informações acústicas não permite bom desempenho do sistema, principalmente quando se utiliza um grande vocabulário de palavras. Neste sentido, informações lingüísticas podem "aproximar" os sistemas artificiais do sistema de reconhecimento de fala humano, proporcionando ganhos de desempenho (menores taxas de erro, maior robustez, etc.) [Lippmann97].

Os *Modelos da Língua*[1] têm o propósito de caracterizar, capturar e explorar as regularidades da língua natural, permitindo "disciplinar" a combinação de palavras durante a criação das hipóteses. A modelagem da língua pode solucionar alguns problemas do reconhecimento: redução do espaço de busca e resolução de ambigüidades acústicas.

Sistemas de reconhecimento que não utilizam conhecimento lingüístico permitem que uma palavra seja seguida por qualquer outra palavra, fazendo com que o espaço de busca tenha, por exemplo, V^N possibilidades para uma sentença de N palavras, onde V é o número de palavras do vocabulário. Usando conhecimento lingüístico podemos reduzir o número de possibilidades, descartando frases agramaticais. Isto tanto pode reduzir o tempo de reconhecimento quanto leva a menores taxas de erro, pois diminui o número de possibilidades de erro.

Durante o reconhecimento, muitas hipóteses possuem características acústicas similares e freqüentemente haverá confusão entre elas. Utilizando informações sintáticas, por exemplo, poderemos classificar apropriadamente as alternativas existentes, resolvendo a ambigüidade acústica e encontrando a hipótese correta.

Existem Modelos da Língua baseados em diversas técnicas:

- Modelo da língua uniforme: a idéia aqui é considerar todas as palavras equiprováveis [LeeKF89];

[1] Estamos adotando esta nomenclatura para a expressão "*Language Model*", por ser a mais adequada para descrever aquilo que tais sistemas realizam. A linguagem (segunda possível tradução de "*language*") é a capacidade do ser humano em se comunicar através de uma (ou mais) língua específica. Esta mesma nomenclatura já é adotada em Portugal.

- Redes de estados finitas: o conjunto das frases legais é representado como uma rede de estados finita na qual as transições correspondem às ocorrências das palavras [Rabiner*85] [LeeCH89];
- Modelos estatísticos *m-gram*: modelos em que a ocorrência de uma palavra está associada a uma probabilidade que depende das *m*-1 palavras anteriores [LeeKF89];
- Modelos baseados em gramática: são tipicamente modelos baseados em gramática estocástica independente de contexto [Jurafsky*95], mas também existem outros tipos como aqueles baseados em gramática de ligação (*link grammar* [Lafferty*92]).

Outros Modelos da Língua são baseados em conceitos como CART (*Classification and Regression Trees*) [Bahl*89] e máxima entropia [Stolcke*96] [Jelinek96].

1.2 Objetivos do Trabalho

Um dos principais objetivos deste trabalho é despertar o interesse pela pesquisa sobre modelagem da língua aplicada ao reconhecimento de fala no Brasil, pois, até onde sabemos, não existem outros trabalhos do gênero no país.

Outro objetivo do trabalho é a implementação de três tipos de Modelos da Língua: dois modelos bigram de classes de palavras e um modelo baseado em gramática independente de contexto. Com isto, pretendemos não só introduzir conceitos necessários ao desenvolvimento de modelos da língua em sistemas de reconhecimento de fala, mas também avaliar as dificuldades existentes e os benefícios decorrentes da utilização dos Modelos da Língua.

1.3 Estrutura da Tese

No capítulo 2, apresentamos uma breve introdução ao sistema de reconhecimento de fala contínua utilizado neste trabalho (desenvolvido por Morais [Morais97], baseado em um modelo híbrido HMM/MLP). Também apresentamos a base matemática e os conceitos teóricos necessários à implementação dos Modelos da Língua *bigram* de classes de palavras e do modelo baseado em gramática independente de contexto.

No capítulo 3, descrevemos a implementação do modelo bigram de classes baseado em classes gramaticais de palavras (segundo a gramática tradicional) e do modelo bigram de classes que utiliza classificação automática através do algoritmo *Simulated Annealing*. Apresentamos e avaliamos os resultados obtidos no treinamento dos modelos (estimação dos parâmetros).

No capítulo 4, descrevemos a implementação do Modelo da Língua baseado em gramática independente de contexto. O Modelo da Língua foi desenvolvido com base numa teoria de Gramática Gerativa [Chomsky65], seguindo o modelo de gramática sintagmática proposto em [Raposo78] mas utilizando também a Sintaxe $\overline{X}$ (X-barra) [Jackendoff77].

No capítulo 5, apresentamos a necessidade de um algoritmo que utilize a informação linguística durante a decodificação (algoritmo de busca integrada). Descrevemos as modificações necessárias no algoritmo de decodificação *Level Building* [Rabiner*85] para que este leve em conta o Modelo da Língua.

No capítulo 6, apresentamos os resultados obtidos no reconhecimento quando empregamos os três modelos da língua desenvolvidos.

No capítulo 7, realizamos as considerações finais sobre o trabalho, apresentando sugestões para trabalhos futuros.

2 Sistema de Reconhecimento de Fala Contínua

2.1 Introdução

O reconhecimento automático de fala contínua pode ser colocado como a busca pela seqüência de palavras correspondente à elocução de entrada. Os sistemas existentes baseiam-se normalmente em princípios de reconhecimento estatístico de padrões e *assumem* que a elocução de entrada corresponderá à seqüência de palavras mais provável avaliada pelo sistema, segundo os modelos adotados.

Costuma-se representar a elocução por uma seqüência de vetores de parâmetros extraídos do sinal de fala (cepstrais, mel-cepstrais, PLP, etc.) [Rabiner*93], conforme ilustra a Figura 2-1. Neste trabalho, o sinal de fala é segmentado em quadros de 10 ms e obtemos vetores formados por 12 parâmetros mel-cepstrais referentes a uma janela de análise de 20 ms (centrada em cada quadro). Representaremos a seqüência de vetores de parâmetros acústicos por $O = \{o_1,...,o_T\}$.

Para encontrar a seqüência de palavras $\hat{W} = \hat{w}_1...\hat{w}_N$, aplica-se o critério da máxima probabilidade a posteriori:

$$\hat{W} = \arg\max_{W} P(W \mid O) \qquad (2\text{-}1)$$

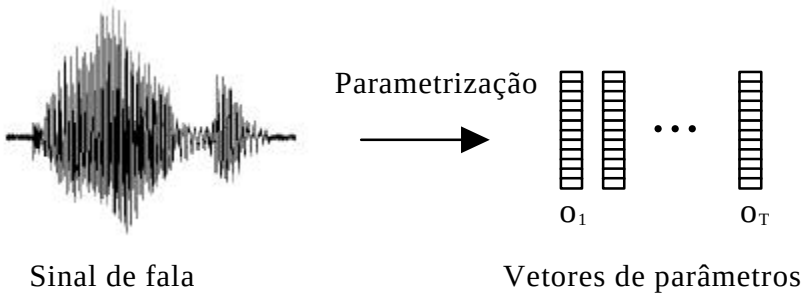

Figura 2-1: Processo de parametrização.

Aplicando a regra de Bayes, podemos decompor e escrevê-la em termos mais apropriados:

$$\hat{W} = \arg\max_{W}\left\{\frac{P(O \mid W).P(W)}{P(O)}\right\} \qquad (2\text{-}2)$$

O termo $P(O)$ é constante para qualquer seqüência de palavras testada e por isso pode ser retirado do processo de decisão. A seqüência $\hat{W}$ corresponderá à seqüência W que maximiza o produto $P(O \mid W).P(W)$, conforme definido por:

$$\hat{W} = \arg\max_{W}\{P(O \mid W).P(W)\} \qquad (2\text{-}3)$$

O termo $P(O \mid W)$ é avaliado pelo **Modelo Acústico** e representa a probabilidade do modelo da sentença W gerar a seqüência observada de vetores O. Neste trabalho, utilizaremos o modelo acústico desenvolvido por Morais [Morais97] baseado no modelo híbrido HMM/MLP [Morgan*95a] (tratado na seção 2.2).

O termo $P(W)$ é avaliado pelo **Modelo da Língua** (tratado nas seções 2.3 e 2.4) e consiste na probabilidade a priori de observar a seqüência de palavras W, independente do sinal observado.

Na figura seguinte, temos o diagrama geral do sistema de reconhecimento de fala contínua.

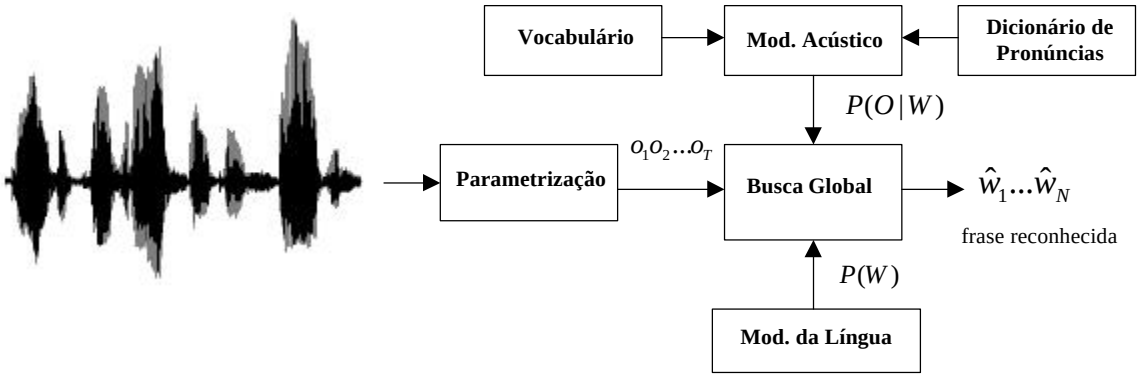

Figura 2-2: Diagrama geral de um sistema de reconhecimento de fala.

O objetivo deste trabalho é desenvolver modelos que permitam estimar o termo $P(W)$.

Inicialmente, propomos dois modelos estatísticos da língua nos quais assumimos que a ocorrência de uma palavra depende da ocorrência da palavra anterior (modelo bigram).

Se utilizássemos um modelo bigram de palavras, a ocorrência de uma palavra estaria diretamente ligada à palavra imediatamente anterior, conforme ilustrado na Figura 2-3

Os modelos desenvolvidos consideram que a relação entre os pares de palavras não será estabelecida diretamente, mas através de *classes de palavras* [Sepsy*97], conforme veremos na seção 2.3.

Usando classes gramaticais de palavra, podemos representar a idéia de um modelo bigram baseado em classes através da Figura 2-4

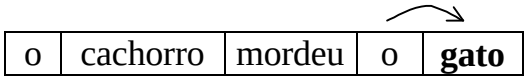

Figura 2-3: Ocorrência de palavras segundo modelo bigram de palavras.

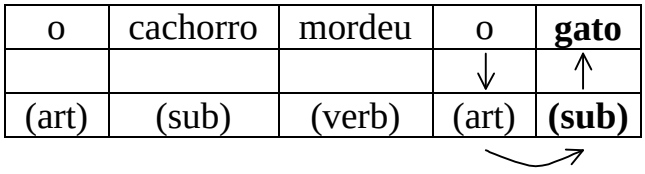

Figura 2-4: Ocorrência de palavras segundo o modelo bigram de classes.

Para desenvolver um Modelo da Língua, não precisamos rigorosamente estimar a probabilidade a priori $P(w_1...w_N)$. Em vez disso, podemos utilizar um modelo que simplesmente aceite ou não determinadas seqüências de palavras. Assim, propomos um modelo sintático da língua implementado através de uma gramática independente de contexto, conforme veremos na seção 2.4.

2.2 Modelo Acústico

Em reconhecimento de fala, assumimos que uma elocução é corretamente representada por uma seqüência $O = o_1 o_2 ... o_T$ de vetores de parâmetros extraídos do sinal de fala. O objetivo dos modelos acústicos é permitir o cálculo da verossimilhança da seqüência observada dado o modelo de cada elocução possível. No caso do reconhecimento de palavras isoladas, isso corresponde a calcular o valor de $P(O \mid v_k)$ para cada palavra v_k do vocabulário $\mathbf{V} = \{v_1,...,v_V\}$. A palavra reconhecida $\hat{w}$ pode ser encontrada fazendo $\hat{w} = \max_{v_k} P(O \mid v_k)$, supondo que todas as palavras são equiprováveis.

O reconhecimento de fala utilizando modelos ocultos de Markov (*Hidden Markov Models* ou HMM's) foi introduzido a partir da década de 70 por pesquisadores da Carnegie Mellon University e da IBM, aplicando a teoria de modelagem estatística por cadeias de Markov (um excelente tutorial sobre HMM pode ser encontrado em [Rabiner*93]).

O HMM pode ser representado por um conjunto de estados conectados por transições e pode ser visto como uma máquina de estados finita que a cada unidade de tempo *t* muda do estado *i* para o estado *j*, gerando um vetor de parâmetros acústicos ao entrar em cada estado. As transições são probabilísticas e estão associadas a uma *probabilidade de transição* e a emissão de vetores acústicos é governada por uma certa *distribuição de probabilidade de emissão*.

O HMM aplicado à modelagem de sinais de fala é então uma composição de dois processos estocásticos: a seqüência *oculta* de estados, que modela a variabilidade temporal da voz, e a seqüência observada de vetores acústicos, que modela sua variabilidade espectral.

A utilização do HMM em reconhecimento de fala admite duas hipóteses: cadeias de Markov de primeira-ordem (probabilidade de transição para o próximo estado depende apenas do estado atual) e independência entre os vetores acústicos da seqüência observada.

Podemos classificar um HMM de acordo com o tipo de distribuição de probabilidade de observação: HMM contínuo, semi-contínuo ou discreto.

Os HMM's contínuo e semi-contínuo utilizam densidades de probabilidade de observação definidas sobre espaços contínuos (mistura de gaussianas, por exemplo). Mais informações podem ser encontradas em [SHLT96] e [Rabiner*93].

No caso do HMM discreto, a distribuição de probabilidade de observação está definida sobre um espaço discreto e finito. Os elementos que definem um HMM discreto são:

a) O número *Ne* de estados q_i no modelo, onde $1 \le i \le Ne$;

b) O alfabeto $X = \{x_1,...,x_M\}$ dos possíveis símbolos observados;

c) Matriz de probabilidades de transição entre estados $A = \{a_{ij}\}$, na qual $a_{ij} = P(q_{t+1} = j \mid q_t = i)$, $1 \le i \le Ne$ e $1 \le j \le Ne$.

d) Matriz de probabilidade de emissão de símbolos $B = \{b_j(x_k)\}$, na qual $b_j(x_k) = P(o_t = x_k \mid q_t = j)$.

e) Distribuição de estado inicial $\Pi = \{p_i\}$, onde $p_i = P(q_1 = i)$.

Para compreender melhor o que é um HMM, vamos apresentar o modelo "Urna-Bola" proposto em [Rabiner*93].

Imagine um conjunto de N urnas e dentro de cada urna um grande número de bolas coloridas. Assumimos que existem M cores distintas para as bolas. Considere agora que uma pessoa escolhe inicialmente uma urna, de acordo com um processo aleatório qualquer. Desta urna, tira-se aleatoriamente uma bola colorida e a cor desta bola é definida como uma *observação*. A bola então é recolocada na urna de origem. Uma nova urna é escolhida por um procedimento aleatório que leva em conta a ocorrência da urna anterior, e a seleção da bola é repetida novamente. Este processo gera uma seqüência finita de cores, o qual podemos modelar como a saída observada de um HMM.

Deve-se perceber que o HMM mais simples que corresponde ao modelo urna-bola é aquele no qual cada urna corresponde a um estado e, para cada estado, teremos as probabilidades de ocorrência de cada cor (vide Figura 2-5).

A partir da urna inicial, a escolha das urnas é governada pelas probabilidades de transição de estado. Deve também ser notado que as urnas possuem bolas de todas as cores, entretanto cada urna possui diferentes composições de bolas coloridas. Assim, a ocorrência de uma bola não identifica a urna de origem: a seqüência de cores é o processo observado, enquanto a seqüência de urnas é um processo oculto.

$b_1(1) = P(vermelho)$ $b_2(1) = P(vermelho)$ $b_N(1) = P(vermelho)$
$b_1(2) = P(amarelo)$ $b_2(2) = P(amarelo)$ $b_N(2) = P(amarelo)$
... ...
$b_1(M) = P(verde)$ $b_2(M) = P(verde)$ $b_N(M) = P(verde)$

Figura 2-5: Modelo Urna-Bola representado através de um HMM discreto.

Os HMM's podem ser classificados de acordo com a estrutura de transição entre os estados. Em reconhecimento de fala, costuma-se adotar o modelo *left-right* (Figura 2-6), no qual os estados são percorridos sucessivamente da esquerda para a direita, de forma que uma vez abandonados não podem ser novamente visitados. Um HMM discreto do tipo *left-right* possui a matriz *A* caracterizada por $a_{ij} = 0 \quad \forall i > j$.

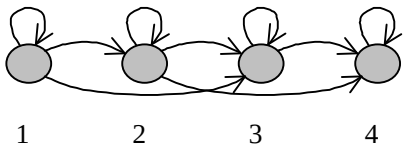

Figura 2-6: Exemplo de HMM tipo *left-right*.

A estimativa de máxima verossimilhança dos parâmetros do modelo pode ser feita aplicando o algoritmo ***Baum-Welch***, apresentado em [Rabiner*93], a partir de um conjunto de elocuções de treinamento.

O cálculo da probabilidade $P(O|W)$ relativa a uma seqüência observada $O = o_1,...,o_T$ e a uma determinada seqüência de palavras *W*, pode ser obtida através do algoritmo recursivo ***Forward-Backward*** [Rabiner*93].

Na prática, a probabilidade $P(O|W)$ é estimada encontrando a seqüência de estados $Q = q_1 q_2 ... q_T$ que maximiza $P(O,Q|W)$, ou seja, assumimos que $P(O|W) \cong \max_Q P(O,Q|W)$. A aproximação é calculada eficientemente através do algoritmo de ***Viterbi*** [Rabiner*93].

Idealmente, deveríamos construir um modelo para cada elocução, entretanto, tal estratégia somente é possível para aplicações muito restritas nas quais o número de elocuções possíveis é pequeno. Normalmente, adota-se um esquema hierárquico no qual cada sentença é modelada como uma seqüência de palavras e cada palavra é modelada como uma seqüência de unidades sub-lexicais (tais como fones, difones e demissílabas), conforme ilustrado na Figura 2-7.

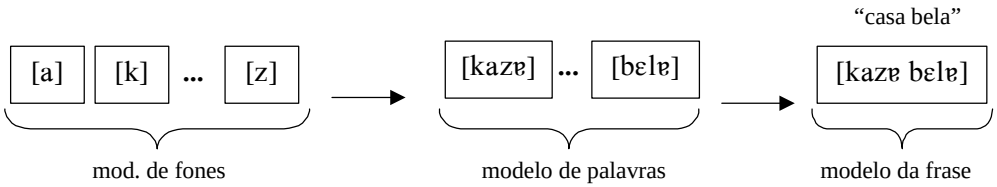

Figura 2-7: Esquema hierárquico na construção dos modelos acústicos.

Neste trabalho, utilizaremos *modelos de fones* construídos por um HMM de um único estado (vide [Morais97]). O modelo de uma palavra será obtido através da concatenação dos modelos de fones, conforme apresentado na figura abaixo.

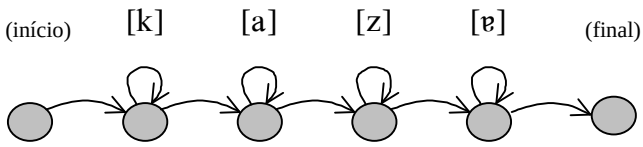

Figura 2-8: HMM proposto para a palavra "casa".

Foram definidos os modelos de 35 fones independentes do contexto mais um modelo para o silêncio, sendo cada modelo um HMM de um único estado. As probabilidades de emissão de símbolos serão estimadas através de uma rede neural MLP a partir da seqüência de vetores acústicos, conforme veremos na seção seguinte.

Também foi incorporado ao sistema de reconhecimento, um modelo de duração de palavra proposto em [Rabiner*85] que associa a cada palavra uma função densidade de probabilidade gaussiana:

$$f_w(d) = \frac{1}{s_w\sqrt{2p}}.\exp\left(-\frac{(d-\bar{d}_w)^2}{2.s_w^2}\right) \qquad (2\text{-}4)$$

onde $\bar{d}_w$ e s_w representam, respectivamente, a média e o desvio padrão da duração da palavra w avaliadas a partir de uma base de treinamento (para maiores detalhes, vide [Morais97]). Este modelo de duração de palavras será usado no capítulo 5, sendo equivalente ao termo denominado P_{dur}^{w}. O modelo de duração também tratado na apresentação dos resultados no reconhecimento (capítulo 6), sendo referido como MDUR.

2.2.1 Modelo Híbrido HMM/MLP

Redes neurais artificiais (RNA) também podem ser aplicadas ao reconhecimento de fala [Lippmann89]: redes ***Multilayer Perceptron*** (**MLP**), redes recorrentes, *Time-Delay Neural Networks* (TDNN), além de outras.

Uma RNA pode ser usada para classificar fones [Waibel*88], ou mesmo para reconhecer palavras isoladas [Peeling*88] [Lang*90], mas as tentativas de realizar o reconhecimento de fala contínua utilizando somente redes neurais não obtiveram grande sucesso.

Posteriormente, descobriu-se que a saída de uma RNA usada como classificador pode ser interpretada como uma estimativa das probabilidades a posteriori das classes de saída condicionadas à entrada da rede [Richard*91].

Uma RNA pode ser usada, então, para estimar a probabilidade a posteriori de um estado do HMM dado a evidência acústica $P(q_k \mid o_t)$. A partir da probabilidade a posteriori, podemos obter as probabilidades de emissão, aplicando a regra de Bayes, conforme verificamos em [Morgan*95b]:

$$p(o_t \mid q_k) = \frac{P(q_k \mid o_t).p(o_t)}{P(q_k)} \qquad (2\text{-}5)$$

Na prática, utiliza-se a probabilidade de emissão escalonada para o HMM, pois durante o reconhecimento, o termo $p(o_t)$ é constante para todas as classes e não interfere no processo de decisão:

$$\frac{p(o_t \mid q_k)}{p(o_t)} = \frac{P(q_k \mid o_t)}{P(q_k)} \qquad (2\text{-}6)$$

Neste trabalho, empregaremos o modelo híbrido desenvolvido por Morais [Morais97] que combina HMM e rede neurais MLP como estimadores das probabilidades de emissão do HMM.

Uma rede MLP possui normalmente uma estrutura do tipo *feed forward* com uma camada de entrada (composta pelas variáveis de entrada), uma camada de saída e zero ou mais camadas intermediárias (chamadas camadas escondidas), conforme ilustra a Figura 2-9.

Cada camada calcula um conjunto de funções discriminantes lineares (definidas por uma matriz de pesos sinápticos) seguidas por uma função não-linear.

A princípio, redes MLP com um número suficiente de neurônios escondidos permitem definir um mapeamento qualquer entre a entrada e saída. Os parâmetros da rede MLP (pesos sinápticos) podem ser "treinados" para associar a saída desejada com a entrada através do algoritmo ***Back-Propagation***. Para uma discussão mais aprofundada sobre redes MLP, sugerimos [Haykin94].

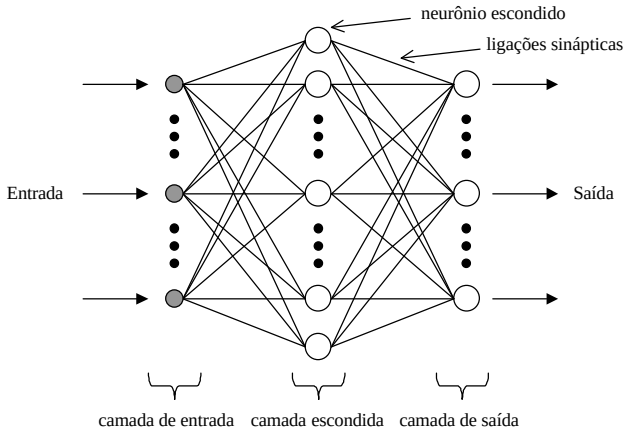

Figura 2-9: Rede MLP com três camadas.

Neste trabalho, o modelo híbrido HMM/MLP emprega uma rede MLP de três camadas, com 84 entradas correspondentes a 7 quadros de 10 ms representados por um vetor acústico de 12 parâmetros mel-cepstrais (para cada quadro). Utilizou-se uma camada escondida com 70 neurônios e uma camada de saída com 36 neurônios correspondentes a cada fone possível nos modelos das palavras (vide [Morais97]).

Na Figura 2-10, temos uma representação do modelo híbrido descrito acima. Observe que a entrada da rede MLP está centrada no quadro referente ao vetor acústico o_t e os demais vetores correspondem à informação contextual (três quadros à direita e à esquerda do quadro central).

A função não-linear usada em todas as camadas da rede MLP foi a função logística, definida por $f(x) = \frac{1}{1+\exp(-x)}$.

Como as saídas da rede são tratadas como uma estimação das probabilidades a posteriori $P(q_k \mid o_t)$, é necessário realizar uma normalização das saídas de maneira a garantir que o somatório seja igual à unidade.

As probabilidades de auto-transição dos HMM's dos fones foram estimadas inicialmente em função da duração média dos fones.

O treinamento da rede MLP foi realizado de forma supervisionada usando o algoritmo *Back-Propagation* com um *corpus* segmentado e etiquetado manualmente.

O modelo híbrido foi então submetido a um processo de reestimação, no qual as probabilidade de transição dos modelos de Markov e os parâmetros da rede são reestimados pelo algoritmo REMAP [Morais97]. Uma explicação detalhada da construção e do treinamento do modelo híbrido pode ser encontrada em [Morais97].

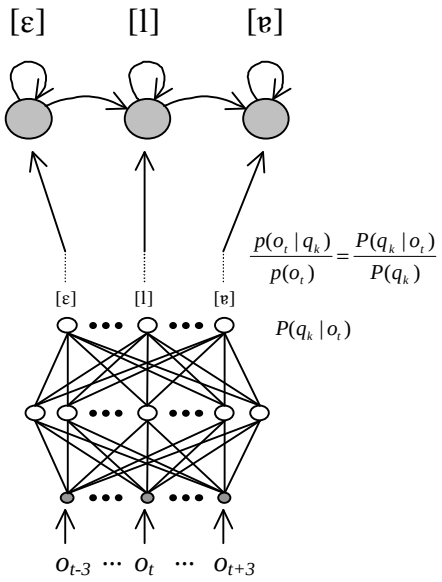

Figura 2-10: Representação simplificada do modelo híbrido HMM/MLP.

2.3 Modelo Estatístico da Língua

De forma geral, implementar um Modelo da Língua num sistema de reconhecimento de fala implica em introduzir mecanismos que estabeleçam restrições à combinação das palavras na

formação das frases reconhecidas. Utilizando um sistema de reconhecimento baseado em modelos e métodos estatísticos, parece natural que também recorramos a modelos estatísticos da língua, já que normalmente precisamos calcular a probabilidade *a priori* das seqüências de palavras. De fato, a modelagem estatística da língua é amplamente utilizada em reconhecimento de fala contínua conforme podemos verificar em [Jardino96], [Jelinek96] e [Suhm*94].

Em nosso caso, o problema foi estruturado de tal forma que o procedimento de busca pela seqüência de palavras inclui o cálculo do termo $P(W) = P(w_1 ... w_N)$.

Assumindo a ocorrência das palavras como um processo estocástico, podemos representar a probabilidade da seqüência através de (2-7).

$$P(W) = P(w_1).\prod_{n=2}^{N} P(w_n \mid w_1 ... w_{n-1}) \qquad (2\text{-}7)$$

O modelo de língua adotado deve ser capaz de estimar as probabilidades $P(w_n \mid w_1 ... w_{n-1})$ para qualquer seqüência de palavras. Entretanto, pode-se perceber facilmente que o número de seqüências de palavras possíveis é proibitivamente alto. Para $n = 4$, por exemplo, com um vocabulário de 1000 palavras, teríamos de estimar da ordem de 10^{12} valores de probabilidade condicional!

Para simplificar o problema, podemos assumir que a escolha da palavra w_n não depende de toda a seqüência passada (história) $h_n = w_1 ... w_{n-1}$ e, desta forma, é razoável propor que agrupemos as várias seqüências passadas em *classes de equivalência* [Jelinek96]. O problema agora reduz-se a calcular (2-8) onde $h_n \rightarrow \Phi(h_n)$ é um mapeamento da história h_n em *m* classes.

$$P(w_n \mid w_1 ... w_{n-1}) = P(w_n \mid \Phi(h_n)) \qquad (2\text{-}8)$$

$$P(W) = P(w_1).\prod_{n=2}^{N} P(w_n \mid \Phi(h_n)) \qquad (2\text{-}9)$$

Existem diversas maneiras de efetuar este mapeamento. Uma maneira simples de defini-lo é considerar que ele depende somente das *m*-1 últimas palavras anteriores. Neste caso, agruparemos todas as seqüências com as mesmas *m*-1 palavras finais na mesma classe.

Exemplificando para *m*=2 e *m*=3 temos, respectivamente, as expressões (2-10) e (2-11). Usualmente, o modelo definido por (2-10) é chamado de modelo **bigram**, enquanto (2-11) corresponde ao modelo **trigram**.

$$P(W) = P(w_1).\prod_{n=2}^{N} P(w_n \mid w_{n-1}) \tag{2-10}$$

$$P(W) = P(w_1).P(w_2 \mid w_1)\prod_{n=3}^{N} P(w_n \mid w_{n-2}w_{n-1}) \tag{2-11}$$

As probabilidades condicionais podem ser estimadas a partir das freqüências relativas sobre uma base de dados de treinamento constituída de um conjunto de frases, pelo método da máxima verossimilhança, através das expressões (2-12) e (2-13), onde $N\{\cdot\}$ representa o número de ocorrências observadas.

$$P(w_n \mid w_{n-1}) \cong \frac{N\{w_{n-1}w_n\}}{N\{w_{n-1}\}} \tag{2-12}$$

$$P(w_n \mid w_{n-2}w_{n-1}) \cong \frac{N\{w_{n-2}w_{n-1}w_n\}}{N\{w_{n-2}w_{n-1}\}} \tag{2-13}$$

Uma forma de construir modelos da língua estatisticamente mais confiáveis usando pequenas bases de treinamento consiste em reduzir a quantidade de parâmetros a serem estimados através do agrupamento das palavras em classes, conforme verificamos em [Suhm*94]. Podemos observar em [Ney*94] que existem várias maneiras de definir e aplicar o mapeamento em classes de palavras.

Definindo um mapeamento determinístico $w \rightarrow G(w)$ que classifica cada uma das V palavras do vocabulário como pertencendo a uma classe dentre um conjunto de K classes, podemos calcular a probabilidade $P(w_n \mid w_1 ... w_{n-1})$ através de (2-14).

$$P(w_n \mid w_1 ... w_{n-1}) = P(w_n \mid G(w_n)).P(G(w_n) \mid w_1 ... w_{n-1}) \tag{2-14}$$

Os termos $P(w_n \mid G(w_n))$ e $P(G(w_n) \mid w_1...w_{n-1})$ serão estimados a partir das freqüências relativas das palavras nas frases de treinamento de maneira similar ao que foi feito em (2-12) e (2-13).

Combinando os mapeamentos $w \to G(w)$ e $h_n \to \Phi(h_n)$, podemos utilizar também o modelo definido por (2-15).

$$P(w_n \mid w_1...w_{n-1}) = P(w_n \mid G(w_n)).P(G(w_n) \mid \Phi(w_1...w_{n-1})) \qquad (2\text{-}15)$$

Outro modelo pode ser definido usando o mapeamento em classes para a palavra atual e também para as palavras anteriores (chamado de modelo simétrico em [Ney*94]). Assumindo um modelo bigram, ou seja, tomando o mapeamento da história como $\Phi(w_1...w_{n-1}) = G(w_{n-1})$, podemos chegar à equação (2-16).

$$P(w_n \mid w_{n-1}) = P(w_n \mid G(w_n)).P(G(w_n) \mid G(w_{n-1})) \qquad (2\text{-}16)$$

Estimar $P(w_n \mid w_{n-1})$, para todas as combinações de duas palavras, significa avaliar um total de $V.(V-1)$ parâmetros independentes[1]. Usando (2-16), teremos de avaliar somente $K.(K-1)$ parâmetros independentes referentes a $P(G(w_n) \mid G(w_{n-1}))$ e $K.(V-1)$ parâmetros independentes referentes a $P(w_n \mid G(w_n))$.

Uma vez que o número de classes, K, é normalmente muito menor que o número de palavras do vocabulário, V, teremos grande redução no número de parâmetros a serem estimados e, consequentemente, a necessidade de uma base de treinamento menor.

De forma geral, podemos construir um modelo *m*-gram baseado em classes, simétrico, adotando um mapeamento do tipo $\Phi(w_1...w_{n-1}) = G(w_{n-m+1})...G(w_{n-1})$ e utilizando a equação (2-17).

$$P(w_n \mid w_{n-m+1}...w_{n-1}) = P(w_n \mid G(w_n)).P(G(w_n) \mid G(w_{n-m+1})...G(w_{n-1})) \qquad (2\text{-}17)$$

[1] O termo $V-1$ deve-se ao fato de que $P(v_1 \mid v_i) + P(v_2 \mid v_i) + ... + P(v_V \mid v_i) = 1$ e, portanto, apenas $V-1$ parâmetros precisam ser calculados.

2.3.1 Modelo Bigram de Classes de Palavras

Neste trabalho, adotaremos modelos da língua bigram de classes de palavras. O mapeamento das palavras em classes não será necessariamente determinístico, podendo uma palavra estar associada a várias classes com diferentes probabilidades.

Considere um vocabulário de palavras $\mathbf{V} = \{v_1,...,v_V\}$ e um conjunto de classes $\mathbf{C} = \{c_1,...,c_K\}$. Usaremos o símbolo w_n para representar uma palavra qualquer na *n*-ésima posição da frase e o símbolo g_n para representar a classe correspondente. Podemos, então, estimar as probabilidades condicionais de palavra através da expressão:

$$P(w_n \mid w_{n-1}) = \sum_{\forall g_{n-1}} \sum_{\forall g_n} P(w_n \mid w_{n-1}, g_{n-1}, g_n).P(g_n \mid w_{n-1}, g_{n-1}).P(g_{n-1} \mid w_{n-1}) \qquad (2\text{-}18)$$

Considerando algumas aproximações, podemos escrever (vide [Jardino*93]):

$$P(w_n \mid w_{n-1}) \cong \sum_{\forall g_{n-1}} \sum_{\forall g_n} P(w_n \mid g_n).P(g_n \mid g_{n-1}).P(g_{n-1} \mid w_{n-1}) \qquad (2\text{-}19)$$

Para interpretar a equação (2-19), podemos considerar a ocorrência das palavras como resultado do processo ilustrado na Figura 2-11.

Um mapeamento determinístico $G(.)$ pode ser obtido considerando que $G(.)$ leva uma palavra w para a classe $G(w)$ com probabilidade um e para as demais classes com probabilidade zero. Dessa forma, teremos as expressões (2-20) e (2-21).

$$P(w_n \mid g_n) = 0, \quad \forall g_n \neq G(w_n) \qquad (2\text{-}20)$$

$$P(g_{n-1} \mid w_{n-1}) = 0, \quad \forall g_{n-1} \neq G(w_{n-1}) \qquad (2\text{-}21)$$

Aplicando (2-20) e (2-21) à expressão (2-19), os somatórios desaparecem e obtemos a equação (2-16) referente ao modelo simétrico com mapeamento determinístico.

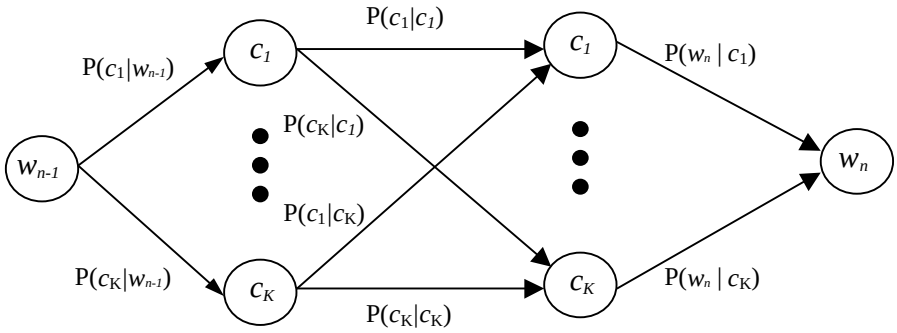

Figura 2-11: Representação do processo de ocorrência de palavras.

A classificação das palavras pode ser feita manualmente, segundo algum critério preestabelecido, possivelmente tirando vantagem da existência natural de classes de palavras na língua (verbos, substantivos, adjetivos, pronomes, etc.), ou pode ser usado algum procedimento automático normalmente baseado em conceitos de teoria de informação como em [Moisa*95] e [Jardino*93].

As fronteiras das frases também podem ser consideradas no cálculo de $P(W)$ através da utilização do marcador "$" como se este fosse mais uma palavra do vocabulário. Observe que o marcador de fronteira de frase define sua própria classe, sendo ele mesmo o único integrante.

Tomando, então, a estrutura $(\$, w_1, w_2, ..., w_N, \$)$, podemos definir (2-22).

$$P(\$\ w_1 w_2 ... w_N\ \$) = P(\$ \mid w_N) \cdot P(w_N \mid w_{N-1}) \cdot ... \cdot P(w_2 \mid w_1) \cdot P(w_1 \mid \$) \cdot P(\$) \qquad (2\text{-}22)$$

Os valores de $P(w_n \mid w_{n-1})$ já foram discutidos. Por simplificação, tomamos $P(\$) = 1$. Considerando os termos $P(w_1 \mid \$)$ e $P(\$ \mid w_N)$, podemos escrever (2-23) e (2-24).

$$P(w_1 \mid \$) = \sum_{\forall g_1} P(w_1 \mid g_1).P(g_1 \mid \$) \qquad (2\text{-}23)$$

$$P(\$ \mid w_N) = \sum_{\forall g_N} P(\$ \mid g_N).P(g_N \mid w_N) \qquad (2\text{-}24)$$

Mesmo utilizando modelos bigram baseados em classes de palavras, não é possível evitar o problema de pares de classes não observados (o que levará a uma *proibição* de certos pares de palavras).

Para garantir que as probabilidades condicionais de classe sejam não-nulas, aplicaremos o *método de interpolação linear* apresentado em [Ney*94], no qual a probabilidade condicional é o resultado de uma média entre a freqüência relativa e uma distribuição geral $h(g_n \mid g_{n-1})$:

$$P(g_n \mid g_{n-1}) = (1-l).\frac{N(g_{n-1}, g_n)}{N(g_{n-1})} + l.h(g_n \mid g_{n-1}) \qquad (2\text{-}25)$$

onde temos que $0 < l < 1$ e $N(g_{n-1}) > 0$.

Neste caso, adotamos a distribuição uniforme $h(g_n \mid g_{n-1}) = \frac{1}{K}$, onde K é o número de classes de palavras.

Assim teremos as probabilidades condicionais estimadas por:

$$P(g_n \mid g_{n-1}) = (1-l).\frac{N(g_{n-1}, g_n)}{N(g_{n-1})} + \frac{l}{K} \qquad (2\text{-}26)$$

Definindo a probabilidade condicional dos pares não observados como p_0, obtemos um fator $l = K.p_0$. Para garantir que $0 < l < 1$, teremos que fazer $p_0 < \frac{1}{K}$.

Finalmente, podemos escrever:

$$P(g_n \mid g_{n-1}) = a.\frac{N(g_{n-1}, g_{n-1})}{N(g_{n-1})} + p_0 \qquad (2\text{-}27)$$

onde $a = 1 - K.p_0$ é o fator de desconto das probabilidades condicionais e o valor total descontado é redistribuído uniformemente entre os pares de classes (vide [Jardino*93]).

Como desejamos somente evitar que as probabilidades condicionais sejam nulas, adotamos $p_0 = 10^{-30}$.

Melhores resultados podem ser obtidos aplicando-se métodos de suavização mais elaborados, como o método proposto por Katz [Katz95], mas deixaremos este ponto para ser explorado por trabalhos futuros.

2.4 Modelo da Língua Baseado em Gramática Formal

Na década de 50, Chomsky [Chomsky57] iniciou a formulação da teoria de Gramática Gerativa, na tentativa de explicar a capacidade do falante de uma língua em distinguir as frases gramaticais das agramaticais.

A idéia básica por traz da gramática gerativa é de que o falante possui um "conhecimento interiorizado da língua" e que utiliza-o na criação e entendimento das frases.

Como resultado dos trabalhos em gramática gerativa, constatou-se que cada língua é um sistema altamente organizado e estruturado, devendo o falante possuir este sistema interiorizado de algum modo.

A primeira hipótese seria de que o falante memoriza todas as frases e estruturas possíveis da sua língua. Contra esta hipótese, temos a verificação de que o falante é capaz de entender e produzir frases que não conhecia previamente. Além disso, o número de frases de uma língua pode ser considerado infinito enquanto a memória de um indivíduo é necessariamente limitada.

A hipótese aceita é de que o conhecimento interiorizado da língua está estruturado na forma de *um corpo de generalizações, princípios e regras mais abstrato mas também mais simples e em número finito* [Raposo78], o qual chamaremos de gramática.

O Modelo da Língua proposto nesta seção foi desenvolvido com base em uma teoria de Gramática Gerativa [Chomsky65] [Raposo78], a partir do uso de uma **gramática independente de contexto (GIC)** [Raposo78].

Os modelos m-gram descritos na seção anterior servem-se das vantagens da modelagem estatística, mas permitem apenas descrever a estrutura linear das frases da língua. As gramáticas gerativas, por outro lado, descrevem naturalmente a estrutura hierárquica das frases.

Do ponto de vista prático, a escolha da teoria lingüística que fundamenta o Modelo da Língua certamente dependerá das características de cada aplicação, pois deverá haver um compromisso entre vários fatores, entre eles a complexidade do modelo (maior ou menor custo computacional) e a melhoria proporcionada no desempenho do sistema de reconhecimento (maior ou menor taxa de erro).

2.4.1 Língua e Gramática Formal

O conceito de gramática formal foi desenvolvido por Chomsky [Chomsky57], como resultado dos esforços de aplicar à descrição das línguas naturais alguns resultados fundamentais alcançados no domínio da lógica matemática e, em especial, no domínio das funções recursivas. Essa tentativa tinha como objetivo representar os resultados da análise lingüística empírica num corpo teórico rigoroso, sistemático e integrado.

Para abordarmos o conceito de gramática formal, definiremos inicialmente o conceito de língua. Segundo Chomsky [Chomsky57], **Língua** é o conjunto finito ou infinito de seqüências (frases), sendo cada uma delas finita em comprimento e construídas por concatenação sobre um conjunto finito de símbolos (vocabulário ou alfabeto). Essa definição é adequada tanto para as línguas naturais quanto para as línguas artificiais.

Assim, dada uma língua L, a questão é como representá-la. Se a língua for finita (número finito de frases), basta enumerar sucessivamente cada uma de suas frases. Se for infinita, o problema é encontrar um processo finito que torne possível enumerar o conjunto infinito de frases. A todo processo finito capaz de enumerar o conjunto das frases de uma língua, e somente estas, damos o nome de gramática de L ou $G(L)$.

Uma das formas de constituir o processo de geração das seqüências de uma língua L é através da definição de um conjunto de funções $f_1, f_2, ..., f_n$ que estabelecem uma correspondência entre pares de seqüências de L.

Considere, por exemplo, as seguintes frases da língua L^*: { ab, cdecde, f, ff, cde, abab }. Verifica-se que podemos agrupar estas seqüências em pares: { ab, abab; cde, cdecde; f, ff } e que podemos associar a este conjunto de pares uma função f_i de maneira que dada uma seqüência “k” , a função f_i produz uma nova seqüência “kk”.

A relação explicitada pela função f_i pode ser formalmente definida através de regras de equivalência.

Considere, por exemplo, um vocabulário $V = \{a, b, c\}$ e as regras de equivalência entre as frases da língua L^*:

$abc \sim a$ Regra 1

$ab \sim ba$ Regra 2

Através deste sistema de regras, podemos concluir que uma seqüência como “a” é equivalente a uma seqüência como “babcc”:

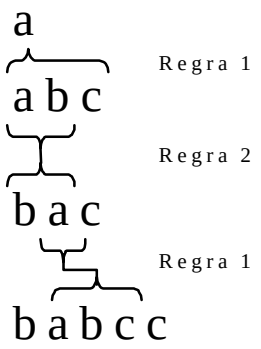

Figura 2-12: Aplicação das regras de equivalência.

Para definirmos uma gramática formal, não adotaremos regras simétricas do tipo $X \sim Y$, que permitem substituir X por Y e vice-versa, mas sim, regras orientadas da forma $X \rightarrow Y$, chamadas de **regras de reescrita** ou **regras de produção**, que permitem reescrever (substituir) o símbolo X como Y.

Segundo [Chomsky65], para ser adequada, toda gramática de uma língua deverá satisfazer duas condições :

a) Ser capaz de gerar a totalidade das frases dessa língua. Nesse sentido, a gramática permite descrever a capacidade do falante da língua (sua *competência*, em termos *Chomskyanos*) para distinguir as seqüências bem formadas das seqüências mal formadas (isto é, fazer julgamentos de gramaticalidade);
b) Ser capaz de associar ou atribuir a cada uma das frases uma descrição estrutural que represente as relações entre seus elementos ou sua estrutura hierárquica;

Podemos representar uma **gramática formal** G como:

$$G = (V_N, V_T, R, S) \tag{2-28}$$

onde V_N é o vocabulário de **símbolos não-terminais**, V_T é o vocabulário de **símbolos terminais**, R é o conjunto de regras de reescrita e S é um símbolo especial de partida (axioma).

Em nosso caso, V_T contém as palavras do vocabulário, enquanto V_N contém classes de palavras ou outras categorias relacionadas à estrutura das frases.

Se definirmos o conjunto V tal que $V = V_N \cup V_T$, podemos adotar V^* como o conjunto de todas as seqüências possíveis de símbolos de V. Analogamente, os conjuntos V_N^* e V_T^* designam todas as seqüências possíveis de símbolos de V_N e V_T, respectivamente.

A partir de agora, adotaremos a seguinte notação para os símbolos:

a) Símbolos terminais: a, b, c, ...
b) Símbolos não-terminais: A, B, C, ...
c) Seqüência contida em V^*: α, β, γ, ...

Considere $\gamma, \delta \in V^*$, a expressão $\gamma \Rightarrow \delta$ denota a existência de uma regra de reescrita em R através da qual a seqüência γ pode ser substituída por δ. Ou seja, a seqüência $\gamma = w_1 \alpha w_2$ pode ser substituída pela seqüência $\delta = w_1 \beta w_2$, pela aplicação da regra $\alpha \rightarrow \beta$ contida em R.

De maneira geral, as regras de reescrita contidas em R são da forma $\alpha \rightarrow \beta$, onde α e β são seqüências de símbolos contidos em V e α possui pelo menos um símbolo não-terminal.

Se uma seqüência γ for substituída por δ através da aplicação de um número finito de regras, isso será indicado pela notação $\gamma \dot{\Rightarrow} \delta$ (a relação $\dot{\Rightarrow}$ é o fechamento transitivo da relação $\Rightarrow$). O número de regras aplicadas também pode ser zero, permitindo escrever $\gamma \dot{\Rightarrow} \gamma$, ou seja, a relação $\dot{\Rightarrow}$ é reflexiva.

Se tivermos $S \dot{\Rightarrow} \delta$, ou seja, se pudermos obter δ a partir do símbolo inicial S, então chamamos δ de *forma frasal*[1].

Utilizando os conceitos acima, podemos representar a língua derivada da gramática G através de (2-29).

$$L(G) = \{x \mid x \in V_T^*, S \dot{\Rightarrow} x\} \qquad (2\text{-}29)$$

[1] O termo usado em inglês é *sentencial form*.

2.4.2 Modelo Baseado em Gramática Independente de Contexto

Definiremos a seguir quatro tipos básicos de gramática identificados por Chomsky [Chomsky65]:

a) Gramática irrestrita

Não há restrições quanto à combinação de símbolos e pode-se descrever toda e qualquer língua capaz de ser representada através de um sistema formal.

Definir uma gramática irrestrita é apenas dizer que uma língua é um sistema de determinado tipo, especificando um número fixo de palavras em cada frase, por exemplo.

b) Gramática sensível a contexto

Neste caso as regras de produção são do tipo $w_1 A w_2 \rightarrow w_1 b w_2$, onde b é não-vazio e $b \in V^*$, estabelecendo que A pode ser substituído por b no contexto w_1, w_2.

c) Gramática independente de contexto

As regras de produção para este tipo de gramática são da forma $A \rightarrow b$, onde já sabemos que $A \in V_N$ e $b \in V^*$.

d) Gramática regular

Numa gramática regular, as regras de reescrita possuem um único símbolo não-terminal na parte esquerda da regra e no máximo um símbolo não-terminal na parte direita: $A \rightarrow aB$, $A \rightarrow Ba$ ou $A \rightarrow a$.

Dentre os tipos apresentados, as gramáticas regulares e independentes do contexto são as mais usadas em sistemas de reconhecimento de fala (sistemas como DRAGON e HARPY [Deller*93] usam redes de estados finitas, que são equivalentes a gramáticas regulares, enquanto o *Berkeley Restaurant Project* [Jurafsky*94] emprega as denominadas **gramáticas estocásticas independentes de contexto**, que correspondem a GIC's cujas regras de produção estão associadas a probabilidades).

A gramática regular ou *gramática de estados finita* descreve apenas a estrutura linear de um padrão e sua limitação consiste no fato de que ela não pode expressar a estrutura hierárquica da

língua natural. Por outro lado, pode representar qualquer conjunto recursivamente enumerável, sendo usada devido à simplicidade de implementação e à eficiência no reconhecimento.

A gramática independente de contexto é adotada como base para modelos da língua por algumas razões, dentre as quais temos:

a) Possui pequena complexidade e permite descrever a estrutura hierárquica das frases de uma língua;
b) Existem algoritmos de análise eficientes para uma GIC, tais como o algoritmo CYK (Cocke-Younger-Kasami) [Kasami65] [Younger67] e Earley[Earley70];

A gramática independente de contexto descreve a estrutura hierárquica de uma frase e essa estrutura é determinada ao longo do processo de análise.

Esta gramática também pode ser modificada para incorporar probabilidades às regras de produção e tornar-se uma **gramática estocástica independente de contexto** [Jurafsky*95].

A gramática independente de contexto é caracterizada por regras de reescrita $A \rightarrow b$ com um único símbolo não-terminal do lado esquerdo e permitir um número qualquer de símbolos terminais e não-terminais do lado direito. Tal regra é chamada de independente de contexto por permitir que os símbolos sejam expandidos independente dos símbolos adjacentes na frase (o contexto).

Diz-se que a gramática está na **forma normal de Chomsky** (FNC), se as regras de produção são da forma $A \rightarrow B\ C$, rescrevendo um símbolo não-terminal como dois símbolos não-terminais, ou da forma $A \rightarrow a$, na qual a é um símbolo terminal (em nosso caso, uma palavra). Permitindo que as regras de produção sejam também da forma $A \rightarrow B$, isto é, a produção de um símbolo não-terminal a partir de outro não-terminal, teremos uma gramática na **forma normal de Chomsky estendida** (FNCE).

Dizemos que uma gramática *reconhece* uma frase quando ela pode gerar esta frase através da aplicação de suas regras de reescrita. Embora ainda não tenhamos discutido a estrutura das frases no português, será simples perceber que a gramática da Tabela 2-1 é capaz de gerar a frase “gatos comem ratos”, conforme podemos verificar na Tabela 2-2. Os símbolos F, SN, SV, N e V possuem seu significado dentro da estrutura da língua, mas por enquanto vamos considerá-los apenas como símbolos (maiores informações serão dadas no capítulo 4).

$F \rightarrow SN\ SV$	(1)
$SN \rightarrow N$	(2)
$SV \rightarrow V\ SN$	(3)
$N \rightarrow$ gatos	(4a)
$N \rightarrow$ ratos	(4b)
$V \rightarrow$ comem	(5)

Tabela 2-1: Exemplo de regras para gramática independente de contexto.

Seqüência	**Regra aplicada**
F	(Símbolo de partida)
SN SV	Regra (1)
N SV	Regra (2)
Gatos SV	Regra (4a)
Gatos V SN	Regra (3)
Gatos comem SN	Regra (5)
Gatos comem N	Regra (2)
Gatos comem ratos	Regra (4b)

Tabela 2-2: Aplicação das regras no reconhecimento da frase “gatos comem ratos”.

A **derivação** de uma seqüência de símbolos terminais w é uma seqüência de reescritas (2-30) sobre V, na qual a primeira forma derivada j_0 consiste no próprio símbolo inicial S e a última forma derivada j_M é a seqüência de símbolos terminais w.

$$S = j_0 \Rightarrow ... \Rightarrow j_M = w \qquad (2\text{-}30)$$

Chamamos de **derivação à esquerda**[1], a derivação na qual, a cada passo, somente o símbolo não-terminal mais à esquerda da forma derivada é reescrito. Cada passo da derivação à esquerda será representado como $\Rightarrow_l$ e por extensão teremos também $\dot{\Rightarrow}_l$.

Tomando, por exemplo, a forma derivada $j_m = aXg$, na qual $a \in V_T^*$, $X \in V_N$ e $g \in V^*$, obtemos a forma derivada $j_{m+1} = abg$ por derivação à esquerda, reescrevendo o símbolo não-terminal mais à esquerda, X, através da regra $(X \rightarrow b) \in R$.

Dessa forma, visto que a regra aplicada estará sempre associada ao símbolo não-terminal expandido a cada passo, a derivação à esquerda poderá ser representada unicamente pela seqüência de regras $r_1,...,r_M$ aplicadas a cada passo da análise.

Um maneira bastante clara de representar a derivação de uma seqüência de símbolos é utilizando árvores.

Uma **árvore** t pode ser definida formalmente como um grafo acíclico diretamente conectado. Seja N o conjunto de nós deste grafo e $l(.)$ uma função de mapeamento destes nós num certo conjunto de símbolos.

Definimos a relação de **dominância imediata** (DI) como a relação existente entre dois nós da árvore que estão ligados diretamente por um arco. Na Figura 2-13, o nó n_1 domina imediatamente n_2, n_3 e n_4, ou seja, os nós n_2, n_3 e n_4 são imediatamente dominados por n_1. A partir da DI podemos definir a relação de **dominância** (D) através da expressão (2-31). A relação D é transitiva, anti-simétrica e reflexiva.

$$DI(n_1, n_3) \cap DI(n_3, n_5) \rightarrow D(n_1, n_5) \qquad (2\text{-}31)$$

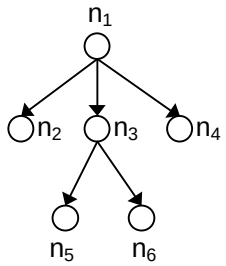

Figura 2-13: Árvore representando relação de dominância imediata.

[1] Em inglês: *leftmost derivation.*

Definimos também a relação de **precedência** *P* relacionada com a ordem (na horizontal, arbitrada aqui da esquerda para a direita) dos nós da árvore.

A partir da Figura 2-14 podemos escrever as seguintes relações de precedência: $P(n_2, n_3)$, $P(n_2, n_5)$ e $P(n_2, n_6)$.

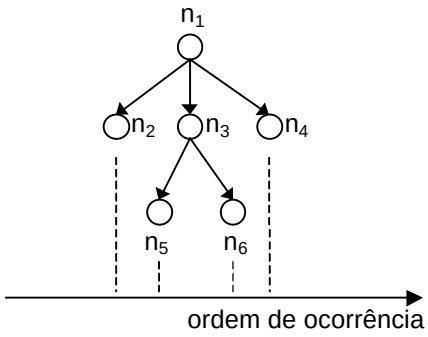

Figura 2-14: Relação de precedência.

Assumindo que não há cruzamentos entre os arcos da árvore, torna-se fácil deduzir as expressões gerais:

$$\forall n, n', n'' \quad P(n, n') \cap D(n, n'') \rightarrow P(n'', n') \tag{2-32}$$

$$\forall n, n', n'' \quad P(n, n') \cap D(n', n'') \rightarrow P(n, n'') \tag{2-33}$$

Utilizando, por exemplo, as relações $P(n_2, n_3)$ e $D(n_3, n_5)$, referentes à Figura 2-14, e aplicando a expressão (2-33), chegamos facilmente à relação de precedência $P(n_2, n_5)$.

Podemos definir alguns termos relativos à árvore t. O nó **raiz da árvore** é representado por $R(t)$. Os elementos extremos da árvore (segundo a relação de dominância *D*) são chamados de **folhas da árvore**. A seqüência ordenada (segundo a relação de precedência *P*) dos rótulos das folhas é chamada de **produto da árvore**, representado por $Y(t)$. Os elementos não-extremos segundo *D* são chamados de *nós internos*.

Uma **árvore de análise** ou **árvore de derivação** da seqüência de símbolos terminais *w* gerados por uma gramática independente de contexto *G* é uma árvore que obedece às seguintes condições:

a) A raiz da árvore é rotulada com o símbolo inicial *S*: $l(R(t)) = S$

b) As folhas da árvore devem ser rotuladas com elementos de V_T (terminais), ou seja, o produto da árvore deve ser a própria seqüência $w : Y(t) = w$.

c) Os nós internos devem ser rotulados com elementos de V_N (não-terminais).

d) Se existe um nó n tal que $DI(n, n_1),...,DI(n, n_K)$ e sabendo que $P(n_i, n_j)$ $\forall i < j$ e $l(n) = X$, $l(n_1) = X_1$, ..., $l(n_K) = X_K$, então existe uma regra de produção em *G* da forma $X \rightarrow X_1...X_K$.

e) Não existe cruzamento de ramos da árvore, sendo válidas as relações de precedência e dominância estabelecidas em (2-32) e (2-33).

Definimos *T(G)* como o conjunto de todas as árvores de análise geradas pela gramática *G*, conforme (2-34).

$$T(G) = \{t \mid Y(t) \in L(G)\} \tag{2-34}$$

Uma gramática é dita **finitamente ambígua** se e somente se existe um número finito de árvores de análise para qualquer seqüência finita pertencente a *L(G)*. Isso eqüivale a exigir que nenhum símbolo não-terminal possa ser reescrito como ele mesmo em um ou mais passos. Obviamente assumiremos que as gramáticas adotadas em nosso trabalho são finitamente ambíguas.

Aplicar a gramática para *analisar* uma frase não é simplesmente reconhecê-la, mas identificar que regras foram utilizadas e definir a descrição estrutural da frase ou o conjunto de relações entre seus elementos.

Voltando à frase “gatos comem ratos”, podemos definir a árvore de análise da Figura 2-15 como a representação estrutural da análise realizada na Tabela 2-2.

A capacidade de gerar frases gramaticais de uma língua é chamada de **capacidade gerativa fraca**. Gramáticas que geram o mesmo conjunto de frases são ditas **fracamente equivalentes**. Já as gramáticas que produzem ainda a mesma representação estrutural são chamadas **fortemente equivalentes**.

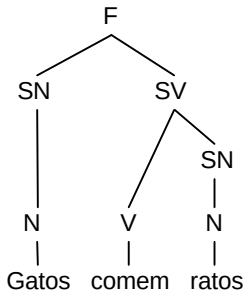

Figura 2-15: Estrutura de análise da frase “gatos comem ratos”.

O Modelo da Língua baseado em GIC será usado pelo sistema de reconhecimento como um modelo sintático que determina que frases são aceitas ou não, segundo a gramática. O modelo atuará como um preditor de palavras durante o procedimento de busca pela seqüência de palavras reconhecida, conforme representamos na Figura 2-16.

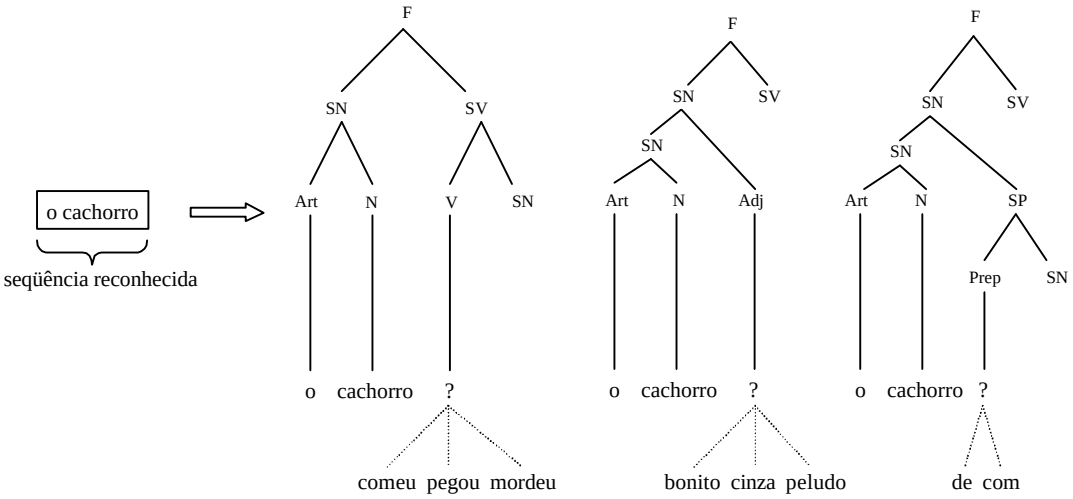

Figura 2-16: Procedimento de predição da próxima palavra.

Dessa forma, em vez de realizarmos uma busca exaustiva pelo espaço das seqüências de palavras possíveis, descartamos as frases agramaticais, diminuindo a quantidade de palavras consideradas a cada passo de decisão (menor perplexidade) e, possivelmente, obtendo menores taxas de erro de palavra.

3 Construção do Modelo Bigram de Classes de Palavras

3.1 Introdução

Neste capítulo, descreveremos a construção de dois tipos de modelos estatísticos da língua. Na seção 3.2, apresentamos um modelo bigram baseado em classes que utiliza classificação manual de palavras segundo a classificação gramatical adotada na gramática tradicional da língua portuguesa. Na seção 3.3, apresentamos um modelo bigram baseado em classes que utiliza classificação automática de palavras tendo como critério a minimização da perplexidade sobre um texto de treinamento.

Dispomos de um conjunto de 247 frases (que chamaremos de conjunto Base) sendo 132 frases fornecidas pelo Instituto de Estudos da Linguagem (IEL) e 115 frases retiradas do jornal Folha de São Paulo.

As frases provenientes do jornal Folha de São Paulo foram colhidas em dias diferentes ao longo de três semanas, procurando-se sempre variar o contexto de onde eram retiradas.

As frases fornecidas pelo IEL foram montadas pela prof. Sandra Madureira (PUC-SP) a partir de pesquisas nos meios de comunicação que constituem usuários potenciais dos sistemas de síntese de fala como mensagens de bancos, companhias aéreas e sistemas de informação por telefone.

Para a classificação manual, utilizamos 204 frases obtidas a partir do conjunto Base. Estas 204 frases possuem um total de 1474 palavras sendo 682 palavras distintas.

Estabelecemos que seriam utilizadas principalmente frases constituídas por uma oração ("o saldo é suficiente", “o preço do café aumentou”, “a sela foi guardada numa cela nos subterrâneos do castelo”, “a TELEBRÁS está investindo em pesquisa”). Frases com orações reduzidas de infinitivo (vide [Cunha85], p. 594) ocorrem em menor número (“Brasil tenta colocar satélite em órbita hoje”). Evitamos portanto estruturas mais complexas como as envolvendo orações subordinadas, coordenadas, apostos, entre outras.

Para a classificação automática, na etapa de avaliação dos algoritmos de classificação, utilizamos 211 frases obtidas a partir do conjunto Base. Estas 211 frases possuem um total de 1687 palavras sendo 686 palavras distintas.

Posteriormente, adicionamos 259 frases ao conjunto das 211 frases de treinamento, resultando em 470 frases com 3665 palavras sendo 1472 palavras distintas. O conjunto das 259 frases é formado por 200 frases provenientes de [Alcaim*92] e 59 frases retiradas do jornal Folha da Tarde.

3.2 Usando Classificação Manual das Palavras

Nesta seção, apresentaremos a implementação de um modelo bigram baseado em classes gramaticais segundo a classificação de palavras adotada na gramática tradicional da língua portuguesa.

O vocabulário é formado por 682 “palavras distintas” classificadas em 20 classes. Consideramos palavras distintas aquelas que diferem na representação grafemática. Neste caso, a palavra “a” pode ser artigo, preposição ou pronome, dependendo do contexto em que se encontra. Também chamamos de “palavra” as locuções e certos conjuntos de palavras vistos como uma unidade, como por exemplo, substantivos próprios compostos por mais de uma palavra (“Adelaide Barroso”). Às locuções será atribuída a classe referente à sua função na frase (adverbial, prepositiva e assim por diante). Nomes próprios compostos são considerados substantivos.

As palavras foram classificadas de acordo com a frase em que se encontravam e, por isso, uma mesma palavra pode possuir mais de uma classe associada.

As estatísticas foram obtidas a partir desta classificação através de programas desenvolvidos em C++, capazes de manipular as frases de treinamento.

3.2.1 Considerações sobre a Língua

Podemos dizer que a frase constitui o menor texto possível dentro do discurso e, na sua realização mais dependente do contexto, a frase toma a forma de uma interjeição, a qual representa globalmente a situação a que se refere (“Pare!”, “Atenção, pista escorregadia”, etc.). No outro extremo, teríamos a oração, uma estrutura normalmente centrada em um verbo (“Gatos comem ratos”) com o qual se faz uma declaração (predicado) sobre um dado tema (sujeito).

As interjeições, frases indicativas, imperativas, exclamativas e interrogativas estão vinculadas ao contexto em que se enunciam. Somente as frases declarativas podem ser totalmente independentes da situação e, portanto, completamente interpretáveis sem seu respaldo.

Qualquer que seja o tipo de frase, ela representará determinado conteúdo através de um sistema hierárquico de unidades (palavras, sintagmas[1], orações) relacionadas entre si por um conjunto de mecanismos formais. A *sintaxe* é a parte da gramática que descreve as regras segundo as quais estas unidades se combinam para formar frases.

Podemos dizer que a oração é composta basicamente por sintagmas nominais (**SN**) e um sintagma verbal (**SV**) [Cunha85].

O SN é toda unidade que tem por núcleo um substantivo, pronome substantivo, numeral ou palavra substantivada *(“o meu cachorro peludo* mordeu *a criança*”, “*ele* comprou *três livros*”). Este núcleo admite a presença de **determinantes** (artigos, numerais, pronomes adjetivos) e de **modificadores** (adjetivos ou expressões adjetivas). Na oração podemos encontrar vários sintagmas nominais, mas somente um deles será o sujeito.

O sintagma verbal se forma em torno da forma verbal e pode ser completado por sintagmas nominais e modificado por advérbios ou expressões adverbiais (“o meu cachorro peludo *mordeu a criança*”, “ele *comprou três livros*”). A estrutura hierárquica de uma frase pode ser representada através de um diagrama em árvore, como está representado na figura seguinte.

[1] O sintagma corresponde a um grupo de elementos, relacionados entre si, no qual um elemento desempenha a função de núcleo. Temos, por exemplo, sintagmas nominais cujo núcleo é um nome (substantivo, pronome substantivo, etc.), sintagmas adjetivais cujo núcleo é um adjetivo e assim por diante.

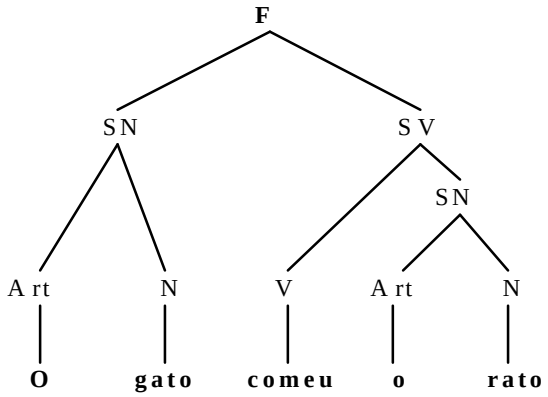

Figura 3-1: Exemplo de estrutura hierárquica de uma frase.

Por enquanto, estaremos preocupados somente com a *estrutura linear da frase*, ou seja, com a seqüência de categorias lexicais, pois utilizaremos um modelo no qual a ocorrência de uma palavra depende apenas da palavra imediatamente anterior (modelo bigram).

No caso da frase "o gato comeu o rato", a estrutura linear seria **(Art)(N)(V)(Art)(N)**, onde Art = artigo, N = nome e V = verbo (vide Figura 3-1).

A estruturação hierárquica das frases no português e a forma de utilizar esta informação no modelamento da língua será tratada somente no capítulo seguinte.

Continuando uma análise superficial das frases no português, seguindo os conceitos da gramática tradicional [Cunha85], podemos definir alguns termos que compõem a oração:

a) Termos essenciais

Temos o **sujeito,** que é formado por um sintagma nominal: "*o meu cachorro peludo* mordeu a criança".

Temos também o **predicado,** que pode ser nominal (verbo de ligação + predicativo do sujeito, onde o núcleo do predicado é um substantivo, adjetivo ou pronome), verbal (cujo núcleo é um verbo significativo) ou verbo-nominal (verbo significativo + predicativo do sujeito): "(ele) está atrasado" (nominal), "(ele) chegou" (verbal), "(ele) chegou atrasado" (verbo-nominal).

Na falta de um dos termos essenciais, dá-se o fenômeno que chamamos de **elipse** ("Boa cidade, Santa Rita": verbo de ligação "é" subentendido).

b) Termos integrantes

O **complemento nominal** vem ligado a substantivos, adjetivos e advérbios por meio de preposição, completando-lhes o sentido: "ele explicou a demora *do ônibus*".

O **complemento verbal** integra o sentido do verbo (objeto direto, objeto indireto e predicativo do objeto): "o cachorro mordeu *a criança*", "ele comprou *três livros*".

De forma geral, encontramos como núcleos destes termos: um substantivo, pronome, numeral ou palavra substantivada.

c) Termos acessórios

São termos que se ligam a um nome ou a um verbo para precisar-lhes o significado. Embora tragam um dado novo à oração, não são indispensáveis ao entendimento. Temos **adjunto adnominal** que delimita o significado de um substantivo (adjetivo, locução adjetiva, artigo, pronome adjetivo e numeral), **adjunto adverbial** e **aposto**.

Em português, como nas demais línguas românicas, predomina a *ordem direta*, isto é, os termos da oração dispõem-se preferentemente na seqüência :

SUJEITO + VERBO + OBJETO DIRETO + OBJETO INDIRETO

ou

SUJEITO + VERBO + PREDICATIVO

Essa preferência pela ordem direta é mais sensível nas orações enunciativas ou declarativas:

O professor	entregou	o livro	ao aluno
(sujeito)	(verbo)	(obj. dir.)	(obj. ind.)

Sua casa	é	linda
(sujeito)	(verbo)	(predicativo)

Queremos chamar atenção para a forma como palavras de diferentes classes gramaticais se ligam para formar a oração. São as regularidades desta estrutura que desejamos capturar no modelo estatístico que propomos neste trabalho.

Embora a oração apresente uma estrutura hierárquica e não uma estrutura linear, um modelo simplificado poderá levar a alguns resultados úteis e oferecer referências para modelos mais elaborados.

3.2.2 Definindo as Classes de Palavra

A classificação das palavras foi baseada na classificação adotada na gramática tradicional da língua portuguesa conforme apresentado em [Cunha85]. Adotamos algumas subdivisões de classes e também classificações que visam diferenciar palavras de uma maneira funcional (como no caso das classes *v-lig*, *v-aux*, *v*, *v-part*, *v-ger* e *v-inf*). Observe o conjunto das classes de palavras na tabela abaixo.

Símbolos	Significado	Símbolos	Significado
Sub	Substantivo	pron-pess	Pronome pessoal
Art	Artigo	pron-dem	Pronome demonstrativo
Adj	Adjetivo	pron-poss	Pronome possessivo
num	Numeral	pron-ind	Pronome indefinido
adv	Advérbio (simples e locução)	v	Verbo
prep	Preposição (simples e locução)	v-part	Verbo no particípio
conj	Conjunção	v-ger	Verbo no gerúndio
prep+art	Preposição+Artigo	v-inf	Verbo no infinitivo
prep+pron-pess	Preposição+Pronome pessoal	v-aux	Verbo auxiliar
prep+pron-dem	Preposição+Pronome demonstrativo	v-lig	Verbo de ligação

Tabela 3-1: Classes de palavras usadas no sistema

Visto que somente frases declarativas compostas por uma oração (ou duas sendo uma delas oração reduzida) foram analisadas, não encontraremos pronomes relativos nem pronomes interrogativos nestas frases. As conjunções podem ser encontradas dentro de um período simples quando ligam unidades de mesma natureza (no caso de enumerações de indivíduos, por exemplo).

Na figura seguinte, temos dois exemplos de classificação de palavras.

O	laço	e	a	fita	ficaram	muito	pequenos
(art)	(sub)	(conj)	(art)	(sub)	(v-lig)	(adv)	(adj)

Foi	detectado	um	problema	em	seu	cartão
(v-aux)	(v-part)	(art)	(sub)	(prep)	(pron-poss)	(sub)

Figura 3-2: Exemplos de classificação das palavras.

A seguir discutiremos as classes estabelecidas na Tabela 3-1, mostrando de maneira geral os critérios adotados na definição das classes e na classificação das palavras.

a) Substantivos

O substantivo é a palavra com que designamos os seres em geral (pessoas, lugares, instituições, espécie e também noções, ações, estados e qualidades quando tomados como seres).

Os substantivos são tradicionalmente classificados como concretos ou abstratos, próprios ou comuns e coletivos. Eles sofrem flexões de número, gênero e grau. Em todas as situações serão designados simplesmente pela classe **sub**.

Um problema surge quando tratamos de substantivos próprios (nomes de pessoas, em particular), pois estes são freqüentemente compostos por mais de uma palavra. Neste trabalho, decidimos considerar nomes próprios como se fossem um só elemento (classificaremos "Adelaide Barroso" como *sub*, por exemplo). Outra opção seria adotar uma classe de *substantivos próprios*, o que pode ser interessante para verificar diferenças estatísticas em relação aos substantivos comuns.

b) Artigos

Os artigos ("o", "a", "os", "as", "um", "uma") se antepõem aos substantivos para indicar que se trata de um ser já conhecido (definido) ou que se trata de um simples representante de dada espécie da qual não se faz menção anterior (indefinido).

As formas simples do artigo são designadas por **art**, enquanto as formas combinadas (contração com preposição) serão classificadas como **prep+art**.

c) Adjetivos

O adjetivo é essencialmente um modificador do substantivo, servindo para caracterizar seres, objetos ou noções expressas pelo substantivo. Os adjetivos também sofrem flexão de número, gênero e grau, mas todas as variações são definidas como **adj**.

d) Numerais

Usamos os numerais para indicarmos uma quantidade exata de elementos ou para assinalarmos o lugar que eles ocupam numa série. Os numerais podem ser cardinais, ordinais, multiplicativos ou fracionários, todos representados simplesmente por **num**.

Observe que os numerais combinam-se entre si, usando também a conjunção “e” (caso do número “vinte e cinco”). Neste caso, não poderíamos colocar explicitamente todas as combinações possíveis de numerais, pois elas são em número infinito.

e) Advérbios

O advérbio é fundamentalmente um modificador do verbo, mas pode modificar também um adjetivo, outro advérbio ou até toda a oração.

Os advérbios podem ser divididos em grupos segundo as circunstâncias ou outras idéias acessórias que expressem. Aqui não faremos esta diferenciação, colocando-os no mesmo grupo denominado **adv**.

Também classificamos como **adv** as locuções adverbiais formadas pela associação de uma preposição com substantivo, adjetivo ou advérbio (“logo após”).

f) Preposições

As preposições são palavras invariáveis que relacionam dois termos de uma oração de forma que o sentido do primeiro é explicado ou completado pelo segundo.

Classificamos como **prep** as preposições simples (expressas por um só vocábulo) e as compostas. As preposições compostas ou locuções prepositivas são formadas de dois ou mais vocábulos sendo o último uma preposição simples (vide [Cunha85]).

g) Conjunções

As conjunções são vocábulos que servem para relacionar duas orações ou dois termos semelhantes na mesma oração. As conjunções que relacionam dois termos ou orações com idêntica função gramatical recebem o nome de coordenativas: “o menino *e* a menina caíram”, “o vento estava forte *e* o céu estava azul ”.

As conjunções subordinativas são aquelas que ligam duas orações entre as quais existe uma relação de dependência: “o menino caiu *porque* a menina empurrou a cadeira”.

Visto que trabalharemos a princípio com frases de uma única oração, as conjunções que aparecem nos exemplos de treinamento são conjunções coordenativas inseridas nas enumerações e formação de números, todas classificadas como **conj**.

h) Pronomes

Os pronomes desempenham na oração as funções equivalentes às exercidas pelos elementos nominais. Existem seis espécies de pronomes: pessoais, possessivos, demonstrativos, relativos, interrogativos e indefinidos.

Pessoais

Caracterizam-se por denotarem as três pessoas gramaticais, por poderem representar (3ª pessoa) uma forma nominal anteriormente expressa e por variarem de forma segundo a função e a acentuação recebida (“eu”, “nós”, “me”, “o”, “lhe”, “mim”, etc.). Todos serão classificados como **pron-pess.**

Possessivos

Pronomes que dão a idéia de posse e estão estreitamente ligados aos pronomes pessoais (“meu”, “minha”, “nosso”, “teu”, etc.). Estes serão classificados como **pron-poss.**

Demonstrativos

Os demonstrativos apresentam formas variáveis (este, estes, esta, estas, etc.) e invariáveis (isto, isso, aquilo), mas todos serão classificados como **pron-dem.**

A combinação dos demonstrativos com preposições (em+este = neste, em+aquilo = naquilo, etc.) serão classificadas como **prep+pron-dem**.

Indefinidos

Os pronomes indefinidos aplicam-se à 3ª pessoa gramatical num sentido vago e indeterminado. Apresentam também formas variáveis (algum, alguns, alguma, etc.) e invariáveis (alguém, ninguém, etc.). As locuções pronominais indefinidas são grupos de palavras que eqüivalem a pronomes indefinidos: “cada um”, “cada qual”, “todo aquele que”, etc. Tanto os pronomes indefinidos quanto as locuções serão designadas como **pron-ind.**

i) Verbos

O verbo possui variações de número, de pessoa, de modo, de tempo, de aspecto e de voz. Por este motivo, um único verbo possuirá um grande número de formas e certamente não é viável armazená-las explicitamente, devido às limitações de espaço de memória. Uma possível solução seria definir regras que permitissem gerar estas variações sem que precisássemos defini-las explicitamente. Entretanto, neste trabalho, adotaremos o primeiro procedimento por simplificação.

Dividiremos os verbos nas seguintes classes, seguindo as orientações da gramática tradicional:

Infinitivo

Classificados como **v-inf** (“comer”, “ensinar”, etc.)**.**

Gerúndio

Classificados como **v-ger** (“comendo”, “ensinando”, etc.)**.**

Particípio

Classificados como **v-part** (“comido”, “ensinado”, etc.)**.**

Verbos auxiliares

Os conjuntos formados de um verbo auxiliar com um verbo principal são chamados de locuções verbais. Nas locuções, somente o auxiliar é conjugado enquanto o verbo principal vem sempre numa das formas nominais.

Como não existe uniformidade de critério lingüístico para definir a auxiliaridade, decidimos considerar como auxiliares somente os verbos “ser”, “haver”, “estar” e “ter” (são auxiliares de uso mais freqüente, pois participam da formação dos tempos compostos e voz passiva). Estes verbos serão classificados como **v-aux,** os demais verbos serão classificados segundo as classes restantes.

Verbos de ligação

Os verbos de ligação ou copulativos servem para unir duas palavras ou expressões de caráter nominal. Assim, não acrescentam propriamente uma idéia nova ao sujeito, funcionando apenas como elo entre este e seu predicativo (“o cachorro *é* peludo”). Os verbos de ligação serão classificados como **v-lig.**

Demais formas verbais

Os verbos significativos que são núcleos do predicado verbal e não estão incluídos nas classes definidas acima serão designados simplesmente como **v**.

Por razões práticas, o conjunto de frases de treinamento é bastante reduzido. Assim, realizamos uma contagem das classes nas frases de treinamento de modo a verificar até que ponto os resultados obtidos são representativos (vide Figura 3-3).

Notamos que algumas dessas classes são pouco observadas. Os casos mais críticos são das contrações **prep+pron_pess, prep+pron_dem** e da forma verbal **v_ger.**

Os pronomes em geral apresentam valores baixos, talvez devido ao tipo de frases colhidas. As maiores ocorrências correspondem aos substantivos, artigos, adjetivos e preposições, pois são justamente os elementos formadores do sintagma nominal. Também os verbos classificados como **v** ocorrem freqüentemente, pois são núcleos do sintagma verbal.

O elevado número de ocorrências de substantivos se explica facilmente se lembrarmos que o substantivo ocupa o papel de núcleo do SN e este pode ocorrer diversas vezes ao longo de uma oração.

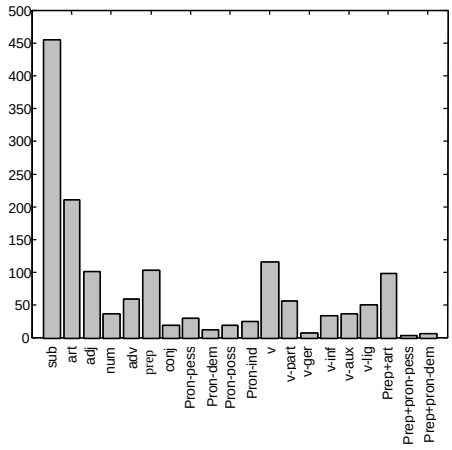

Figura 3-3: Freqüência das classes nas frases de treinamento

3.2.3 Obtendo as Probabilidades Condicionais

Podemos estimar a probabilidade condicional de classes de palavras, $P(g_n \mid g_{n-1})$, através do estimador de máxima verossimilhança dado por:

$$P(g_n \mid g_{n-1}) = \frac{N(g_{n-1}, g_n)}{N(g_{n-1})} \qquad (3\text{-}1)$$

Observe que $N(g_{n-1}, g_n)$ representa o número de ocorrências de pares, enquanto $N(g_{n-1})$ representa o número de ocorrências da classe g_{n-1} nas frases de treinamento.

O procedimento de contagem e avaliação das estatísticas foi realizado automaticamente através de um programa desenvolvido em C++.

Tomamos inicialmente o arquivo texto das frases de treinamento e executamos uma classificação manual das palavras, gerando um outro arquivo texto conforme indicado na Figura 3-4.

ele guarda a sela do cavalo numa prateleira de uma antiga cela
ele guarda a sela numa prateleira de uma cela do palácio
a sela do cavalo é guardada numa prateleira de uma antiga cela
a sela foi guardada numa cela nos subterrâneos do castelo
(...)

Arquivo texto com frases

pron-pess v art sub prep+art sub prep+art sub prep art adj sub
pron-pess v art sub prep+art sub prep art sub prep+art sub
art sub prep+art sub v-aux v-part prep+art sub prep art adj sub
art sub v-aux v-part prep+art sub prep+art sub prep+art sub
(...)

Arquivo texto com estrutura linear das frases

Figura 3-4: Procedimento manual de classificação das palavras

Depois da classificação, executamos os programas que processam o arquivo texto com as estruturas e calculam as estatísticas desejadas.

Lembrando da teoria de probabilidade, fica claro que a estimação através de (3-1) será tanto melhor quanto maior for a quantidade de dados. Desta forma, podemos perceber a importância da consideração feita anteriormente a respeito de classes com baixa ocorrência (principalmente para "prep+pron-pess", "prep+pron-dem" e "v-ger").

Os valores estimados de probabilidade condicional são mostrados na forma de imagem (Figura 3-5), na qual tons mais escuros representam valores mais altos de probabilidade.

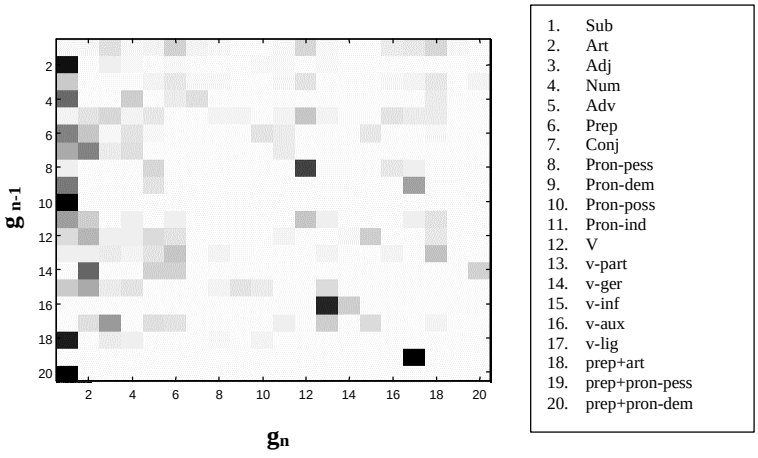

Figura 3-5: Probabilidade condicional $P(g_n \mid g_{n-1})$

Temos ao todo 400 valores de probabilidade condicional e não podemos comentá-los todos neste trabalho, entretanto, podemos analisar alguns pontos mais significativos segundo conhecimentos da língua portuguesa.

Observe inicialmente a diagonal da matriz representada na Figura 3-5. Cada elemento da diagonal representa o termo $P(g_n \mid g_{n-1})$, ou seja, a probabilidade de uma classe ocorrer dado que ela ocorreu na posição anterior. O gráfico da Figura 3-6 deixa claro que somente algumas classes "permitem" repetição. Temos os numerais e os advérbios que permitem efetivamente este tipo de construção, os substantivos (nas enumerações) e os verbos no particípio (lembre que o particípio pode funcionar como adjetivo).

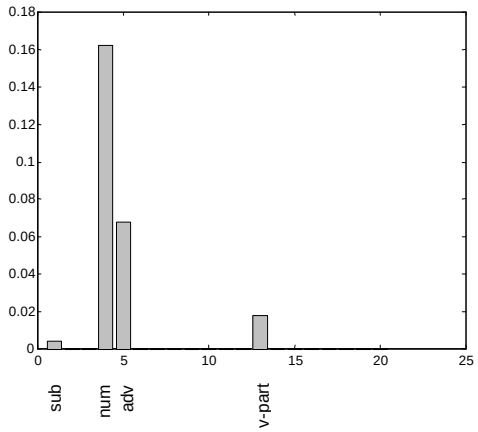

Figura 3-6: Diagonal da matriz de probabilidade condicional

Nas figuras seguintes, mostramos valores de probabilidade condicional relacionada a elementos pertencentes ao sintagma nominal (SN). Já sabemos que principalmente os substantivos desempenham o papel de núcleo do SN. Na Figura 3-7 percebemos que as classes mais prováveis de seguir um substantivo são "adj", "prep", "prep+art" e verbos. As três primeiras classes aparecem ligadas ao substantivo na formação de adjuntos adnominais e complementos nominais. O "verbo" aparece quando passamos do SN ao SV, e o verbo então sucede o substantivo([…sub]$_{SN}$[verb …]$_{SV}$).

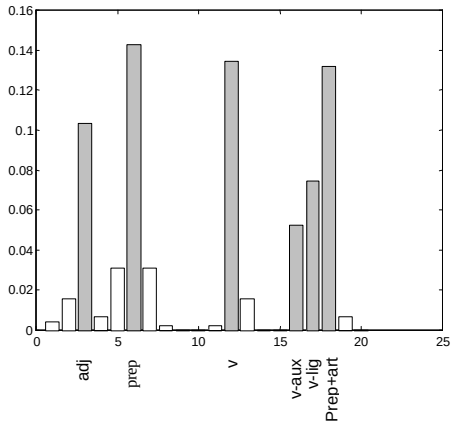

Figura 3-7: Valores de $P(c \mid sub)$

Sabemos que a colocação "mais comum" do artigo é imediatamente antes do substantivo, conforme podemos verificar na Figura 3-8 e Figura 3-9.

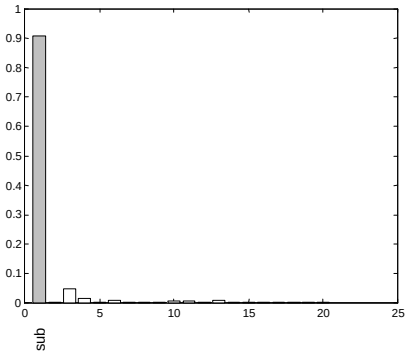

Figura 3-8: Valores de $P(c \mid art)$

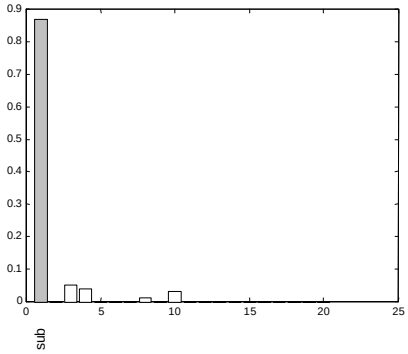

Figura 3-9: Valores de $P(c \mid prep + art)$

Observe que os dois resultados são bastante similares, certamente devido à presença do artigo como último elemento na construção "prep+art", o que faz com que este elemento comporte-se como artigo com relação ao elementos seguintes. O comportamento das contrações ainda será discutido na seção 3.2.4.

Os pronomes pessoais também aparecem como núcleos do SN (normalmente sozinhos), fazendo fronteira com o SV. A Figura 3-10 mostra "v" (verbo) como elemento mais provável seguindo o pronome pessoal. Logicamente, devido às simplificações assumidas aqui e aos poucos exemplos com pronome pessoal (veja Figura 3-3), os resultados não são tão representativos quanto deveriam.

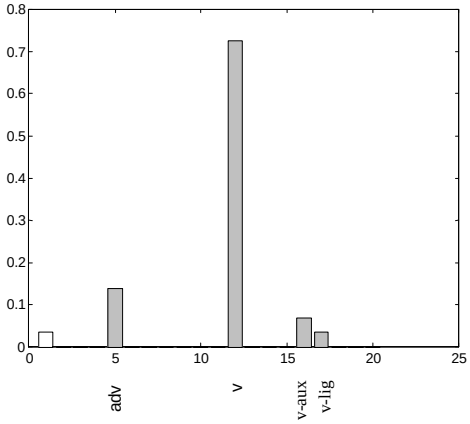

Figura 3-10: Valores de $P(c \mid pron - pess)$

Analisando agora a classe "v" (verbos em geral exceto formas nominais, verbos de ligação e auxiliares), podemos lembrar que os termos que podem seguir "v" nas frases declarativas são objetos diretos, objetos indiretos e advérbios. De maneira geral, o objeto direto é composto por um SN e o objeto indireto, por um SN precedido de preposição. Isso explica os resultados obtidos para valores de $P(c \mid v)$ mostrados na Figura 3-11. Observe a probabilidade mais alta de artigos, substantivos (objeto direto), advérbios, preposições e contrações de preposições com artigos (objeto indireto e outros casos), e verbos no infinitivo (locuções verbais e orações reduzidas).

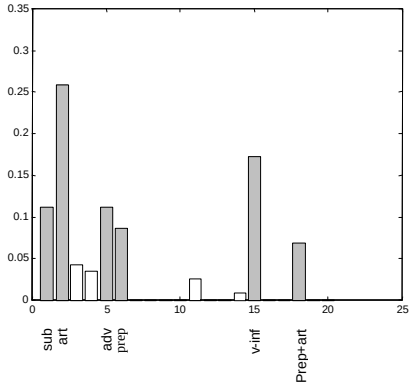

Figura 3-11: Valores de $P(c \mid v)$

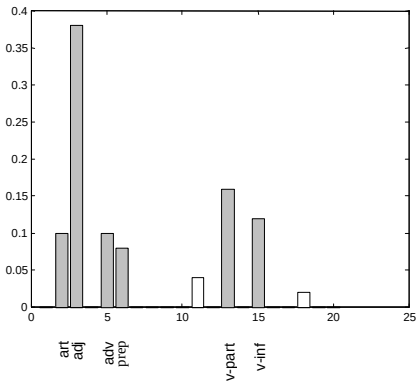

Figura 3-12: Valores de $P(c \mid v - lig)$

Na Figura 3-12, temos a probabilidade condicional de classe dado um verbo de ligação ("v-lig"). Observe a maior probabilidade do elemento seguinte ser adjetivo ("adj") ou verbo no particípio ("v-part"), que muitas vezes tem valor de adjetivo. Este comportamento concorda com o fato do verbo de ligação participar da formação do predicado nominal cujo núcleo é o predicativo

do sujeito (freqüentemente um adjetivo ou mesmo verbo no particípio). O verbo no infinitivo aparece em orações reduzidas.

Voltando à Figura 3-5, poderemos ver que a probabilidade condicional de classe dado um advérbio é bem distribuída entre as classes, não havendo um valor que se destaque pela sua amplitude (o valor mais elevado corresponde à classe c = "v"). O mesmo comportamento possui $P(c \mid adj)$, com valor máximo para classe c = "sub". No caso dos verbos auxiliares ("v-aux") podemos notar um grande máximo para $P(v-part \mid v-aux)$, principalmente devido à presença da voz passiva nas frases analisadas.

3.2.4 Tratando as Contrações com Preposição

Conforme dissemos anteriormente, definimos as contrações com preposição como um tipo de classes. Assim temos "prep+art", "prep+pron-pess" e "prep+pron-dem", para contrações de preposição com artigo, pronome pessoal e pronome demonstrativo, respectivamente. O comportamento das contrações na frase será extremamente influenciado pela posição dos elementos constituintes da contração.

Nas figuras seguintes (Figura 3-13 e Figura 3-14) podemos ver os gráficos referentes a $P(prep \mid c)$ e $P(prep + art \mid c)$, onde c corresponde à classe indicada no eixo horizontal dos gráficos. Observe na Figura 3-8 e Figura 3-9, o comportamento das probabilidades condicionais $P(c \mid art)$ e $P(c \mid prep + art)$.

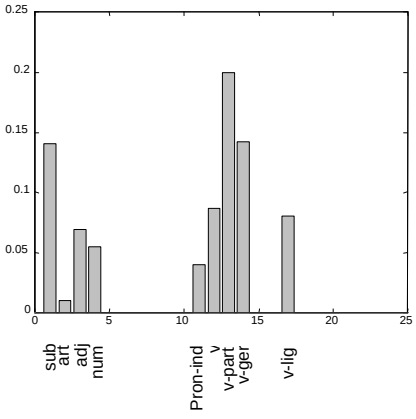

Figura 3-13: Valores de $P(prep \mid c)$

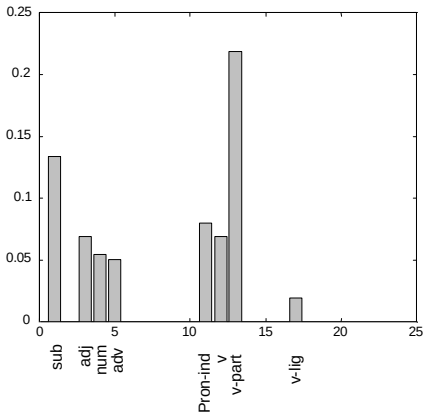

Figura 3-14: Valores de $P(prep + art \mid c)$

Este conjunto de gráficos indica a similaridade de comportamento entre “prep” e “prep+art” com relação a elementos (classes) posicionados à esquerda, e a similaridade entre “art” e “prep+art” com relação a elementos posicionados à direita. Os elementos à esquerda de “prep+art” vêem-no como “prep”, enquanto os elementos à direita vêem-no como “art”.

As maiores dissimilaridades entre $P(prep \mid c)$ e $P(prep + art \mid c)$ ocorrem quando observamos “v-ger” (cuja quantidade de exemplos na base de dados foi insuficiente), “adv” e “v-lig”.

Esta característica apresentada pela contração “prep+art” pode ser estendida a outras contrações. Assim, através das probabilidades condicionais relativas a “prep”, “pron-pess” e “pron-dem”, é possível estimar as probabilidades condicionais onde aparecem “prep+pron-pess” e “prep+pron-dem” cujo número de exemplos na base de dados foi muito pequeno para conduzir a resultados corretos.

Podemos então fazer as aproximações (3-2) e (3-3), sendo necessário fazer depois a normalização da matriz de probabilidade condicional obtida anteriormente.

$$P(\text{prep} + \text{pron - pess} \mid c) \cong P(\text{prep} \mid c) \tag{3-2}$$

$$P(c \mid \text{prep} + \text{pron - pess}) \cong \text{P}(c \mid \text{pron - pess}) \tag{3-3}$$

A matriz de probabilidade condicional de classe obtida com o procedimento acima foi a matriz efetivamente usada durante os testes com o sistema de reconhecimento de fala contínua.

3.2.5 Definindo as Probabilidades de Iniciar e Finalizar Frase

Além da probabilidade condicional calculada entre classes de palavras, determinamos também a probabilidade $P(c \mid \$)$ de uma classe iniciar a frase e a probabilidade $P(\$ \mid c)$ de uma classe terminar a frase. As probabilidades desejadas podem ser estimadas através de (3-4) e (3-5).

$$P(c \mid \$) = \frac{N(\text{"classe } c \text{ iniciando frase"})}{Número\_de\_frases} \tag{3-4}$$

$$P(\$ \mid c) = \frac{N(\text{"classe } c \text{ terminando frase"})}{N(c)} \tag{3-5}$$

Assim, podemos representar uma frase como uma estrutura linear usando o marcador de fronteira $ de frase, conforme ilustra a Figura 3-15.

$	O	laço	e	a	fita	ficaram	muito	pequenos	$
($)	(art)	(sub)	(conj)	(art)	(sub)	(v-lig)	(adv)	(adj)	($)

Figura 3-15: Estrutura linear da frase usando o marcador de fronteira $

Os valores estimados de probabilidade condicional para início e fim de frase podem ser vistos nas duas figuras seguintes.

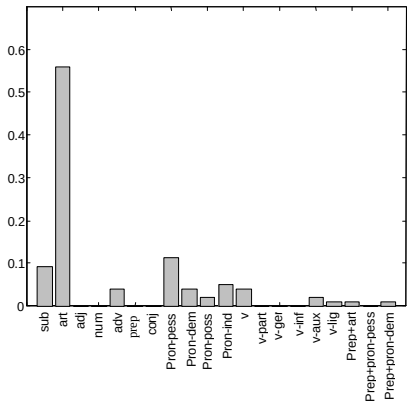

Figura 3-16: Probabilidade de uma classe iniciar a frase

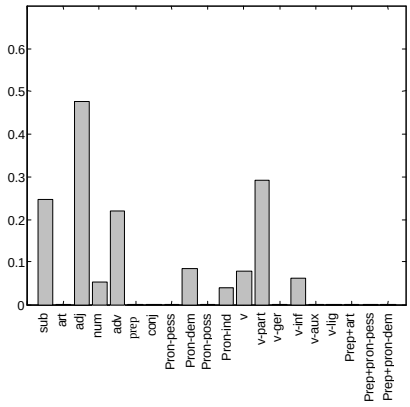

Figura 3-17: Probabilidade de uma classe terminar a frase

3.3 Usando Classificação Automática das Palavras

Nesta seção trataremos de algoritmos baseados em métodos estatísticos que permitem a classificação automática de palavras a partir de um conjunto de frases de treinamento. As classes não são previamente definidas segundo critérios lingüísticos e nem as palavras etiquetadas, somente o número de classes é fixado.

Assumiremos a partir daqui que cada palavra pode pertencer somente a uma classe e, ao contrário da classificação manual, a divisão em classes constituirá uma partição do espaço de palavras.

Poderíamos permitir também que cada palavra pertencesse a mais de uma classe, obtendo melhores resultados na classificação, como pode ser visto em [Jardino*93], entretanto, isso também aumentaria a complexidade do nosso sistema.

O modelo bigram de classes foi construído a partir da classificação de palavras obtida, calculando as probabilidades condicionais de classe, conforme definido nas equações (3-1), (3-4) e (3-5).

3.3.1 Algoritmos de Classificação Automática

Na comparação entre dois sistemas de reconhecimento de fala, torna-se necessário avaliar os modelos da língua e "quantificar" até que ponto eles facilitam a tarefa de reconhecimento. Uma maneira usual de medir a dificuldade imposta por uma língua, durante a tarefa de busca pela seqüência de palavras correta, é calcular o número médio de palavras que podem suceder uma seqüência de palavras encontrada anteriormente. Chamamos este valor de **perplexidade**.

Num caso limite, supondo um vocabulário de V palavras, onde cada palavra pudesse suceder qualquer outra com mesma probabilidade, teríamos claramente uma perplexidade V.

Formalmente, podemos definir a perplexidade usando conceitos provenientes da teoria de informação. Considere as seqüências de palavras como realizações de uma fonte de informação (língua) segundo alguma lei estocástica. Definindo uma seqüência de palavras de comprimento N como $W = w_1...w_N$, podemos escrever a entropia associada a esta fonte como [Deller*93]:

$$H(W) = -\lim_{N\to\infty} \frac{1}{N} \left\{ \sum_{\forall w_1...w_N} P(w_1...w_N).\log_2 P(w_1...w_N) \right\} \qquad (3\text{-}6)$$

O termo $P(w_1...w_N)$ corresponde à probabilidade da fonte gerar a seqüência dada. Se as palavras fossem geradas pela fonte de uma maneira independente, teríamos :

$$H(W) = -\sum_{\forall w \in \mathbf{V}} P(w).\log_2 P(w) \qquad (3\text{-}7)$$

Na prática, calcula-se a estimativa da entropia a partir de uma seqüência de comprimento N finito, mas suficientemente longo e de estimativas das probabilidades $P(w_1...w_N)$:

$$\hat{H}(W) = -\frac{1}{N}.\log_2 \hat{P}(w_1 w_2 ... w_N) \qquad (3\text{-}8)$$

Observe que a entropia estimada $\hat{H}(W)$ dá uma idéia do grau de dificuldade que os sistemas de reconhecimento terão de enfrentar pois avalia a incerteza média na determinação de uma palavra gerada pela fonte de informação. Para línguas naturais temos apenas estimativas da entropia real $H(W)$.

Em modelos de língua baseados em redes de estados finitas, a perplexidade representa o fator de ramificação médio a partir de um nó qualquer e pode ser definida em função da entropia estimada $\hat{H}(W)$ resultando em:

$$PP = 2^{\hat{H}(W)} \qquad (3\text{-}9)$$

Reorganizando os termos das expressões (3-8) e (3-9), podemos ainda escrever a perplexidade como (3-10).

$$PP = \hat{P}(w_1 w_2 ... w_N)^{-\frac{1}{N}} \qquad (3\text{-}10)$$

Os algoritmos de classificação desenvolvidos baseiam-se justamente na minimização da perplexidade sobre o conjunto das frases de treinamento.

Embora não a utilizemos diretamente, pois isso elevaria desnecessariamente a complexidade computacional dos algoritmos, trataremos como se a perplexidade fosse a própria função de custo a ser minimizada, por uma questão de simplificação.

As técnicas usadas nos algoritmos de classificação foram a minimização de Monte Carlo, um algoritmo K-Means [Kneser*93] [Urbela*95] e *Simulated Annealing* [Jardino*93] [Moisa*95]. A seguir, discutiremos os três algoritmos implementados.

Minimização de Monte Carlo (MC)

Propomos um procedimento simples de minimização da perplexidade sobre o texto de treinamento que consiste em estabelecer uma configuração (divisão das palavras em classes) inicial e a partir desta efetuar mudanças no sistema, levando uma palavra de sua classe inicial para uma outra classe. A escolha da palavra e da classe-destino são aleatórias (seleção de Monte Carlo). A nova configuração será aceita somente se não houver aumento da perplexidade.

A seguir temos o funcionamento geral do algoritmo:

Algoritmo Monte Carlo

Ajusta a configuração inicial;
Calcula a perplexidade;
Repete até que um critério de parada seja alcançado
  Seleciona uma palavra v dentre as palavras do vocabulário;
  Seleciona uma classe c dentre o conjunto de classes;
  Calcula a variação ΔPP de perplexidade referente à mudança de v para c;
  Se $\Delta PP \leq 0$, aceita nova configuração;
Fim-repete;

Um dos problemas referentes a este algoritmo é que ele só garante convergência para um mínimo local que muitas vezes pode ser inadequado, levando-nos a ter de repetir o algoritmo algumas vezes para escolher o melhor resultado. Suas vantagens residem na simplicidade e rapidez.

Algoritmo K-Means (KM)

Outro algoritmo usado na classificação automática de palavras pode ser visto em [Kneser*93] [Urbela*95]. O procedimento consiste basicamente em mover cada palavra para a classe que proporciona menor perplexidade e repetir o processo até que nenhuma melhora seja conseguida.

Da mesma forma que anteriormente, poderemos chegar a um mínimo global, mas o algoritmo só garante convergência a mínimos locais de perplexidade.

A seguir, apresentamos o algoritmo K-Means.

Algoritmo K-Means

```
Ajusta a configuração inicial;
Calcula a perplexidade;
Repete até que um critério de parada seja alcançado
  Repete para toda palavra v do vocabulário
    Repete para toda classe c
      Calcula a variação ΔPP de perplexidade referente à mudança de v para c;
    Fim-repete;
    Move palavra v para classe que proporciona menor perplexidade;
  Fim-repete;
Fim-repete;
```

Pudemos observar que este algoritmo possui convergência mais lenta que o algoritmo anterior, e apresenta, em média, valores de perplexidade muito próximos daqueles conseguidos com os outros métodos.

A configuração inicial é definida colocando as $K-1$ palavras mais freqüentes (K é o número de classes) numa classe separada. Na classe restante, colocamos as demais palavras do vocabulário.

Algoritmo Simulated Annealing (SA)

Em problemas como do "caixeiro-viajante" [Kirkpatrick*83] ou em problemas de otimização combinatória de maneira geral, pode-se usar uma técnica denominada *Simulated Annealing* [Aarts*89] para se atingir o ótimo global do sistema.

Observando que o problema de classificação automática de palavras visando minimizar a perplexidade é um problema de otimização combinatória, podemos aplicar a técnica de Simulated Annealing como método de otimização.

Em física da matéria condensada, "annealing" corresponde ao processo térmico usado para obter estados de baixa energia de um sólido em um "banho quente":

1. Aumente a temperatura até que ela chegue ao valor no qual o sólido derrete;
2. Diminua lenta e cuidadosamente a temperatura até que as partículas se organizem totalmente;

A técnica de *Simulated Annealing* vem da simulação de *Monte Carlo* do processo termodinâmico descrito acima e pode ser realizada através da repetição do chamado algoritmo *Metropolis*:

Algoritmo Metropolis:

Dado um estado i de energia E_i, podemos gerar um estado de energia E_j através de um mecanismo de perturbação. Se $E_j \leq E_i$, a transição é aceita. Se $E_j > E_i$, a transição é aceita com probabilidade $\exp\left(\frac{E_i - E_j}{e}\right)$, onde e é um parâmetro de controle.

Considere então um problema de otimização combinatória onde $f(.)$ é a função de custo adotada e S o espaço de soluções (possíveis configurações ou estados) do sistema. O objetivo é partir de um estado inicial i_{start} e chegar ao ótimo global i_{opt} que define o mínimo (ou máximo) global da função de custo $f(.)$, conforme definem as equações (3-11) e (3-12).

$$f(i_{opt}) \leq f(i) \quad \forall i \in S \quad \text{(minimização)} \tag{3-11}$$

$$f(i_{opt}) \geq f(i) \quad \forall i \in S \quad \text{(maximização)} \tag{3-12}$$

Para resolver o problema de classificação automática usando SA, apresentamos o algoritmo proposto em [Aarts*89]:

Algoritmo Simulated Annealing:

```
begin
 INITIALIZE( i_start , e_0 , L_0 )
 k = 0
 i = i_start
 repeat
  for l = 1 to L_k
  begin
   GENERATE ( j from S_i )
   if f(j) ≤ f(i) then i = j
   else if exp( (f(i) - f(j)) / e_k ) > random[0,1] then i = j
  end
  k = k + 1
  CALCULATE_LENGTH( L_k )
  CALCULATE_CONTROL( e_k )
 until stop_criterion
end
```

Dado um estado $i \in S$, podemos gerar (seleção de Monte Carlo) um estado j dentro da vizinhança (S_i) do estado i, onde a probabilidade de aceitação do estado j é dada por (3-13).

$$P\{\text{aceita } j \mid \text{estado\_ atual} = i\} = \begin{cases} 1 & \text{, se } f(j) \leq f(i) \\ \exp\left(\dfrac{f(i) - f(j)}{e_k}\right) & \text{, se } f(j) > f(i) \end{cases} \tag{3-13}$$

O número de transições efetuadas em cada iteração é definida por L_k. O parâmetro de controle tem valor inicial $e_0 > 0$ e vai diminuindo a cada iteração.

Fazendo o parâmetro de controle cair exponencialmente a zero enquanto o número de transições tende a infinito, chegaremos ao ótimo global $i_{ópt}$.

Na prática, o número de transições precisa ser finito, por isso, adotaremos os seguintes procedimentos:

1 – Valor inicial do parâmetro de controle

O valor de e_0 deve ser alto, de forma a permitir que qualquer transição seja aceita. Neste caso, o procedimento adotado foi calcular o custo médio das transições ($f(j) - f(i)$) e definir um valor de e_0 tal que levasse a uma alta taxa de aceitação (tipicamente da ordem de 95%).

2 – Decremento do parâmetro de controle

A estratégia de "resfriamento" consiste em adotar $e_{k+1} = a.e_k$ onde $k = 0,1,2,\ldots$. O valor típico de a usado fica entre 0,8 e 0,99 pois a diminuição do parâmetro de controle deve ser lento, correspondendo à lenta diminuição da "temperatura do sistema".

3 – Número de transições a cada iteração

As transições são aceitas com probabilidade decrescente e o número de transições deve ser tal que um quase-equilíbrio seja atingido a cada iteração, dessa forma, o correto seria fazer $L_k \to \infty$ para $e_k \to 0$. Entretanto, limitamos o número de transições a um valor máximo $\overline{L}$.

4 – Condição de parada

A execução do algoritmo pode terminar quando a função de custo permanece inalterada durante algumas iterações ou definindo um número máximo de iterações.

Para o problema de classificação automática de palavras, desejamos minimizar a perplexidade avaliada sobre o texto de treinamento. A configuração inicial deve ser tal que proporcione uma alta perplexidade (alta energia inicial, correspondendo ao "aquecimento" do

sistema). As novas configurações (estados) vão sendo geradas movendo uma palavra de sua classe para uma nova classe. Novamente, tanto a palavra quanto a nova classe são escolhidas aleatoriamente (seleção de Monte Carlo). Os demais aspectos de funcionamento do algoritmo seguem o que foi discutido acima.

3.3.2 Acelerando a Classificação: Minimização Eficiente da Perplexidade

Um dos pontos decisivos para tornar rápidos e viáveis os programas de classificação automática discutidos neste trabalho é a minimização eficiente da perplexidade (vide [Martin*95]).

Na verdade, não utilizaremos diretamente a perplexidade como fator a ser minimizado, mas uma função denominada FL, relacionada à log-probabilidade da seqüência completa de palavras nas frases de treinamento (3-14).

Pode ser facilmente percebido que a perplexidade está relacionada a esta função através da equação (3-15).

$$FL = -\log_2 P(w_1 ... w_N) \qquad (3\text{-}14)$$

$$PP = 2^{\frac{1}{N}.FL} \qquad (3\text{-}15)$$

Visto que se trata de uma função monótona crescente e o fator N permanece constante, podemos minimizar a perplexidade, minimizando o termo FL, o que corresponde à maximização da probabilidade da seqüência de palavras.

Uma vez calculada a função FL, somente alguns termos precisam ser recalculados a cada mudança de palavra de uma classe-origem para uma classe-destino. A seguir mostraremos como pode ser feito o cálculo da função FL a partir das frases de treinamento.

Adotaremos um modelo bigram da língua e usaremos um marcador de fronteira de frase ($), de maneira que o texto será representado como $\{w_1 w_2 ... w_N\} = \{\$\, w_1^{(1)} ... w_{N_1}^{(1)}\, \$ ... \$\, w_1^{(L)} ... w_{N_L}^{(L)}\, \$\}$, onde w_n^k simboliza a *n*-ésima palavra da *k*-ésima frase, N_k é o número total de palavras na *k*-ésima frase e L é o número de frases. Consequentemente, temos que o número total de palavras será dado por $N_1 + N_2 + ... + N_L + L + 1 = N$, considerando o marcador de fronteira como mais uma palavra do vocabulário.

Considerando todo o texto de treinamento, podemos escrever a probabilidade da seqüência de palavras como (3-16) e definir a log-probabilidade da seqüência como (3-17), onde o termo $P(w_1) = P(\$) = 1$, como já foi definido anteriormente.

$$P(w_1 ... w_N) = \prod_{n=2}^{N} P(w_n \mid w_{n-1}) \tag{3-16}$$

$$FL = -\log P(w_1 ... w_N) = -\sum_{n=2}^{N} \log P(w_n \mid w_{n-1}) \tag{3-17}$$

Avaliando o somatório acima sobre todo o texto, chegaremos à expressão dada por (3-18), onde v_i e v_j correspondem a duas palavras do vocabulário (incluindo \$).

$$FL = -\sum_i \sum_j N\{v_i, v_j\} . \log P(v_j \mid v_i) \tag{3-18}$$

Considerando que a classificação de palavras é realizada de maneira que cada palavra v pertence somente a uma classe definida por $G(v)$ e que o marcador de fronteira de frase define sua própria classe, podemos escrever a relação (3-19) (vide seção 2.3).

$$P(v_j \mid v_i) = P(v_j \mid G(v_j)) \cdot P(G(v_j) \mid G(v_i)) \tag{3-19}$$

Uma estimativa de máxima verossimilhança das probabilidades no segundo membro da equação (3-19) é dada por (3-20), onde $N\{.\}$ corresponde ao número de ocorrências do argumento dentro de todo o texto de treinamento.

$$P(v_j \mid v_i) \cong \frac{N\{v_j\}}{N\{G(v_j)\}} \cdot \frac{N\{G(v_i), G(v_j)\}}{N\{G(v_i)\}} \tag{3-20}$$

Combinando as equações (3-18) e (3-20), podemos manipular os termos e encontrar a expressão (3-21).

$$FL = -\sum_{i,j} N\{v_i, v_j\}.\log N\{v_j\} + \sum_{i,j} N\{v_i, v_j\}.\log N\{G(v_j)\} - \\ - \sum_{i,j} N\{v_i, v_j\}.\log N\{G(v_i), G(v_j)\} + \sum_{i,j} N\{v_i, v_j\}.\log N\{G(v_i)\} \quad (3\text{-}21)$$

Definiremos as contagens de palavras a partir das contagens de pares de palavras através da expressão:

$$N\{v_i\} = \sum_j N\{v_i, v_j\} \quad (3\text{-}22)$$

As contagens relativas às classes de palavras podem ser obtidas através das contagens bigram de palavras:

$$N\{c_m, c_n\} = \sum_{i:G(v_i)=c_m} \sum_{j:G(v_j)=c_n} N\{v_i, v_j\} \quad (3\text{-}23)$$

$$N\{c\} = \sum_{v:G(v)=c} N\{v\} \quad (3\text{-}24)$$

Usando as expressões acima, podemos reescrever a equação (3-21) obtendo a equação a seguir.

$$FL = -\sum_i N\{v_i\}.\log N\{v_i\} + 2.\sum_i N\{c_i\}.\log N\{c_i\} - \sum_{i,j} N\{c_i, c_j\}.\log N\{c_i, c_j\} \quad (3\text{-}25)$$

Nos algoritmos de classificação automática, não é preciso calcular a perplexidade relativa ao texto a cada movimento de uma palavra entre sua classe-origem c_o e a classe-destino escolhida c_d. Em vez disso, calcula-se diretamente a variação ΔFL correspondente à mudança. Esta variação é utilizada nos algoritmos no lugar da variação da perplexidade ΔPP. A utilização de ΔFL reduz a complexidade computacional do problema e acelera substancialmente os algoritmos de classificação.

O termo ΔFL pode ser definido a partir de (3-25) considerando somente os termos referentes às classes afetadas pelo deslocamento da palavra, eliminando aqueles que permanecem constantes (3-26).

$$\begin{aligned} FL_{\Delta} = {} & 2.N\{c_o\}.\log N\{c_o\} + 2.N\{c_d\}.\log N\{c_d\} - \\ & - \sum_{\forall c} N\{c,c_o\}.\log N\{c,c_o\} - \sum_{\substack{\forall c \\ \neq c_o}} N\{c_o,c\}.\log N\{c_o,c\} - \\ & - \sum_{\substack{\forall c \\ \neq c_o}} N\{c,c_d\}.\log N\{c,c_d\} - \sum_{\substack{\forall c \\ \neq c_o \\ \neq c_d}} N\{c_d,c\}.\log N\{c_d,c\} \end{aligned} \tag{3-26}$$

O termo FL_{Δ} é avaliado antes e depois da mudança de classe da palavra e ΔFL é definido como a variação encontrada (3-27).

$$\Delta FL = FL_{\Delta}^{(t+1)} - FL_{\Delta}^{(t)} \tag{3-27}$$

As contagens de classe e pares de classe também podem ser ajustadas eficientemente, através das expressões (3-28) - (3-33), considerando c_o e c_d as classes origem e destino, respectivamente, da palavra v.

$$N\{c_o\} = N\{c_o\} - N\{v\} \tag{3-28}$$

$$N\{c_d\} = N\{c_d\} + N\{v\} \tag{3-29}$$

$$N\{c_o,c\} = N\{c_o,c\} - N\{v,c\} \tag{3-30}$$

$$N\{c,c_o\} = N\{c,c_o\} - N\{c,v\} \tag{3-31}$$

$$N\{c_d,c\} = N\{c_d,c\} + N\{v,c\} \tag{3-32}$$

$$N\{c,c_d\} = N\{c,c_d\} + N\{c,v\} \tag{3-33}$$

3.3.3 Realizando alguns Testes de Classificação

Como exemplo de funcionamento, puramente ilustrativo, elaboramos um pequeno conjunto de 10 frases com um total de 41 palavras sendo 33 palavras distintas (vide Tabela 3-2). As frases não foram escolhidas por acaso, pode-se perceber que se trata de frases declarativas compostas por uma oração e predicado nominal, nas quais indicamos estados ou qualidades dos seres ou objetos.

Podemos perceber a existência de algumas classes de palavras que desempenham funções específicas dentro dessas orações: artigos, preposições, substantivos, adjetivos e verbos de ligação.

Destas, apenas os substantivos estão inseridos em mais de uma estrutura, como em “a **bola**” e “o **céu**” (precedidas por artigo) em comparação com “de **maria**” (precedida por preposição). Não queremos, entretanto, analisar a função desses substantivos nas frases, mas apenas ressaltar que as unidades adjacentes são de natureza diferente, fato importante pois devemos lembrar que o modelo adotado é bigram.

a bola é redonda
o céu é azul
os sapatos são feios
as meninas estão tristes
sapos são pequenos
a menina está suja
coelhos são bonitos
a casa de maria é grande
as casas das pessoas são caras
dinheiro era importante

Tabela 3-2: Conjunto de frases de treinamento do exemplo

Estabelecendo, então, seis classes de palavras e executando o algoritmo de classificação MC, obtivemos a divisão em classes mostrada na Tabela 3-3 e uma perplexidade final de 6,0.

classe 0	**classe 1**	**classe 2**	**classe 3**	**classe 4**	**classe 5**
redonda(1)	são(4)	a(3)	é(3)	sapatos(1)	bola(1)
azul(1)		as(2)	céu(1)	sapos(1)	o(1)
feios(1)		casa(1)	estão(1)	coelhos(1)	os(1)
tristes(1)		casas(1)	está(1)	pessoas(1)	meninas(1)
pequenos(1)			maria(1)		menina(1)
suja(1)			era(1)		de(1)
bonitos(1)					das(1)
grande(1)					dinheiro(1)
caras(1)					
importante(1)					

Tabela 3-3: Divisão em classes usando minimização de Monte Carlo (exemplo).

Os valores entre parêntesis, ao lado de cada palavra, correspondem ao número de ocorrências no texto-exemplo.

Na classe 0 podemos observar que foram agrupados todos os adjetivos existentes. Na classe 1, somente o verbo de ligação “são” e na classe 4, temos somente substantivos. Nas demais classes, temos combinações de diversos tipos de palavras e dificilmente podemos justificá-las de maneira convincente

Executando diversas vezes o algoritmo MC, poderemos eventualmente atingir o mínimo global, dependendo da complexidade do espaço de busca. Entretanto, a divisão em classes não será, em geral, tão simples de analisar como no caso acima.

Executando o algoritmo KM sobre o texto-exemplo, obtivemos a divisão em classes mostrada na Tabela 3-4. A perplexidade final obtida foi 14,3. Comparando este valor com aquele obtido para o algoritmo MC (6,0), constatamos que o algoritmo KM conduziu o sistema a um mínimo local de perplexidade.

classe 0	**classe 1**	**classe 2**	**classe 3**	**classe 4**	**classe 5**
o(1), céu(1), azul(1), os(1) sapatos(1), feios(1), meninas(1) estão(1), tristes(1), sapos(1) pequenos(1), menina(1), está(1) suja(1), coelhos(1), bonitos(1) casa(1), de(1), maria(1) grande(1), casas(1), das(1) pessoas(1), caras(1), dinheiro(1) era(1), importante(1)	são(4)	a(3)	é(3)	as(2)	bola(1) redonda(1)

Tabela 3-4: Divisão em classes usando algoritmo K-Means (exemplo).

Aplicando o algoritmo SA ao texto-exemplo, obtivemos a divisão em classes mostrada na Tabela 3-5 e uma perplexidade final de 4,8 (a menor perplexidade obtida até agora).

Podemos perceber as seguintes classes de palavras: Adjetivos (classe 0), substantivos (classe 1 e classe 4), preposições (classe 2), artigos (classe 3) e verbos (classe 5).

Os substantivos que fazem parte da estrutura *(artigo)(substantivo)* estão na classe 0, enquanto os que fazem parte da estrutura *(substantivo)* (no início da oração: “sapos”, “coelhos” e “dinheiro”) ou *(preposição)(substantivo)* foram colocados na classe 4.

Deve ser notado que tal divisão em classes proporciona uma “regularidade” na seqüência das classes e, consequentemente, uma menor perplexidade.

classe 0	classe 1	classe 2	classe 3	classe 4	classe 5
redonda(1), azul(1), feios(1), tristes(1), pequenos(1), suja(1), bonitos(1), grande(1), caras(1), importante(1),	bola(1) céu(1) sapatos(1) meninas(1) menina(1) casa(1) casas(1)	de(1) das(1)	a(3) o(1) os(1) as(2)	sapos(1) coelhos(1) maria(1) pessoas(1) dinheiro(1)	é(3) são(4) estão(1) está(1) era(1)

Tabela 3-5: Divisão em classes usando *Simulated Annealing* (exemplo).

3.3.4 Treinamento do Modelo

Utilizando o conjunto de treinamento com 211 frases, aplicamos os algoritmos MC, KM e SA para a classificação automática das palavras.

Inicialmente, todas as palavras foram colocadas inicialmente numa mesma classe e a perplexidade inicial foi de 225. Em seguida, os algoritmos foram executados até que a condição $\frac{\Delta FL}{FL} < 0{,}001$ fosse atingida.

Este procedimento foi repetido para diferentes números de classes, obtendo-se a perplexidade final, as probabilidades condicionais de classe, assim como a classificação das palavras.

Na tabela seguinte apresentamos os valores de perplexidade final obtidos pelo algoritmo MC.

Número de classes	Perplexidade
20	60
40	32
60	23
100	14
200	8

Tabela 3-6: Perplexidade final no treinamento usando MC.

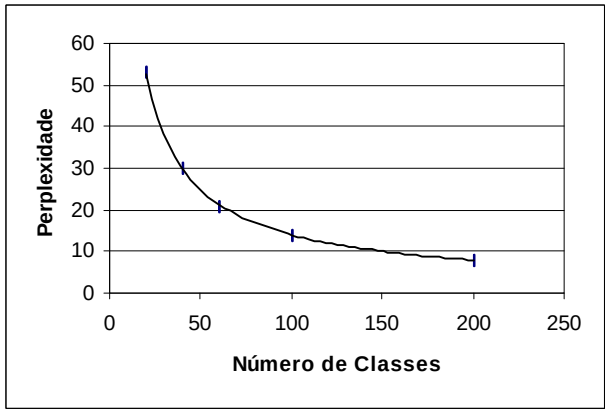

Figura 3-18: Comportamento da perplexidade com o número de classes (MC).

Através da Figura 3-18, podemos perceber que aumentando o número de classes, não obtemos uma diminuição proporcional da perplexidade, comportamento também observado em outros trabalhos [Jardino*93] [Moisa*95].

Utilizando o algoritmo KM, observamos que os valores de perplexidade obtidos (Tabela 3-7) estão muito próximos (e por vezes abaixo) daqueles conseguidos com o algoritmo MC.

O comportamento da perplexidade conforme aumentamos o número de classes é similar ao apresentado pelo algoritmo anterior, conforme verificamos na Figura 3-19.

Número de classes	**Perplexidade**
20	56
40	33
60	22
100	14
200	8

Tabela 3-7: Perplexidade final no treinamento usando KM

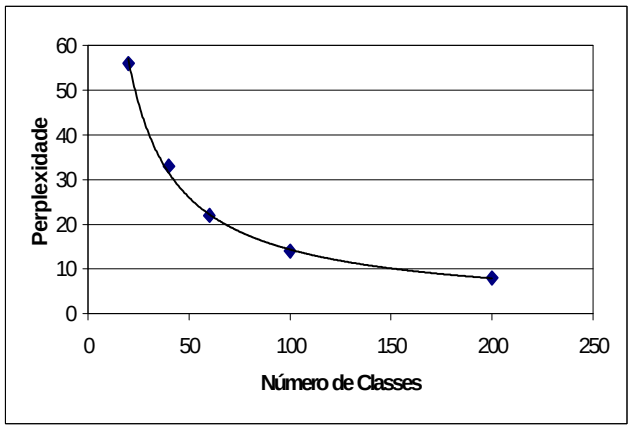

Figura 3-19: Comportamento da perplexidade com o número de classes (KM)

Aplicando o algoritmo SA ao texto de treinamento com 211 frases, obtivemos os valores de perplexidade final apresentados na Tabela 3-8.

Número de classes	Perplexidade
20	53
40	30
60	21
100	14
200	8

Tabela 3-8: Perplexidade final no treinamento usando SA.

O comportamento da perplexidade conforme aumentamos o número de classes é mostrado na Figura 3-20. Os valores obtidos são menores que usando os outros métodos.

Todos os algoritmos mostrados preocupam-se em minimizar a perplexidade sobre o texto de treinamento, diferindo na técnica que empregam para tal. Dentre eles, o mais rápido foi o MC, provavelmente devido a sua simplicidade. O mais lento e que necessita de um ajuste cuidadoso de seus parâmetros é o SA. O KM é relativamente simples mas converge mais lentamente que o MC.

Comparando os valores de perplexidade final obtidos sobre o conjunto de 211 frases de treinamento (Figura 3-21), observa-se que os melhores resultados são do SA e os piores são do MC, embora todos estejam bastante próximos.

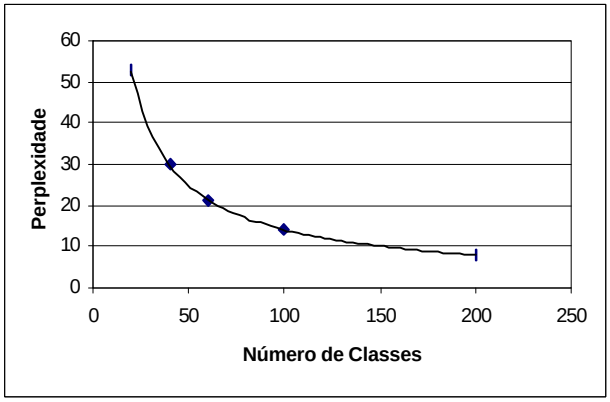

Figura 3-20: Comportamento da perplexidade com o número de classes (SA)

O comportamento da perplexidade conforme aumentamos o número de classes também sugere que podemos adotar um número "ótimo" de classes, pois além de determinado valor, a perplexidade final não sofre grandes diminuições. Analisando a Figura 3.21, poderíamos assumir que o número ótimo está entre 50 e 100 classes.

A desvantagem de usar um número grande de classes está no fato de perdermos o poder de generalização do modelamento estatístico baseado em classes, já que assim estamos nos aproximando das probabilidades bigram de palavra (caso em que temos $K = V$). Por outro lado, um pequeno número de classes pode significar uma alta perplexidade e por conseguinte uma maior dificuldade no reconhecimento.

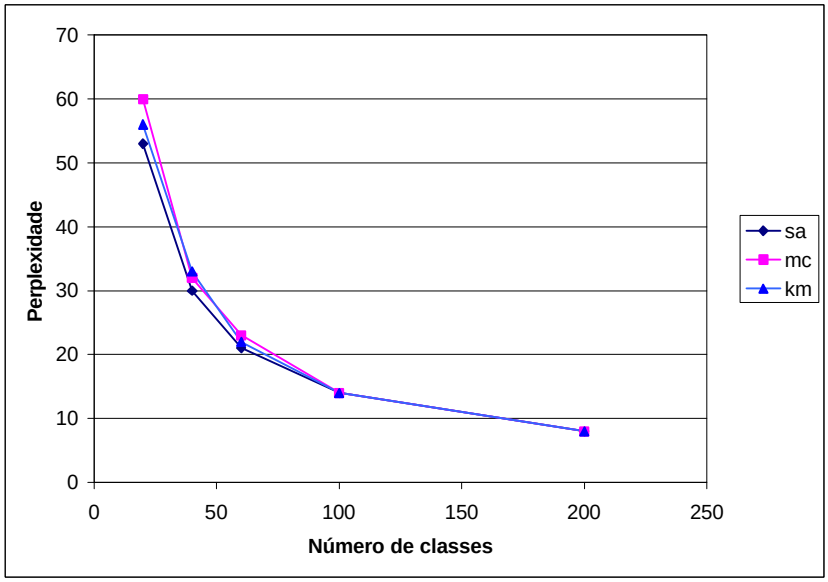

Figura 3-21: Perplexidade final sobre o texto de treinamento para os três algoritmos

Tipicamente, os resultados usando SA costumam ser melhores que os demais, como pode ser visto em [Moisa*95]. Entretanto, o menor valor de perplexidade pode não corresponder ao menor valor de erro de palavra durante o reconhecimento, já que os dois parâmetros só estão indiretamente relacionados. Neste sentido, pode ser compensador adotar técnicas mais rápidas e simples que o SA, como o KM ou MC.

Analisar a divisão em classes das 686 palavras do vocabulário torna-se uma tarefa difícil pela complexidade das relações entre essas palavras nas frases de treinamento. Por esse motivo, optamos por mostrar alguns resultados a partir do texto-exemplo anterior, pois tratava-se de um "experimento controlado". Além disso, a pouca quantidade de frases de treinamento faz com que muitas palavras não possuam exemplos em número suficiente para serem colocadas na classe mais adequada. Uma lista com a classificação de palavras obtida usando SA pode ser vista no Apêndice C.

Baseando-se nos melhores resultados preliminares utilizando o algoritmo *Simulated Annealing*, resolvemos adotá-lo na implementação dos modelos da língua que utilizam classificação automática de palavras.

Utilizando agora o conjunto de treinamento com 470 frases obtivemos os resultados apresentados na Tabela 3-9 e Figura 3-22

Considerando o número total de palavras no conjunto de treinamento (3665 palavras), mas levando em conta a esparsidade natural da matriz de probabilidades condicionais, resolvemos limitar o número máximo de classes em 80.

Número de classes	**Perplexidade**
20	97
40	66
60	47
80	38

Tabela 3-9: Perplexidade final no treinamento usando SA e conjunto de 470 frases.

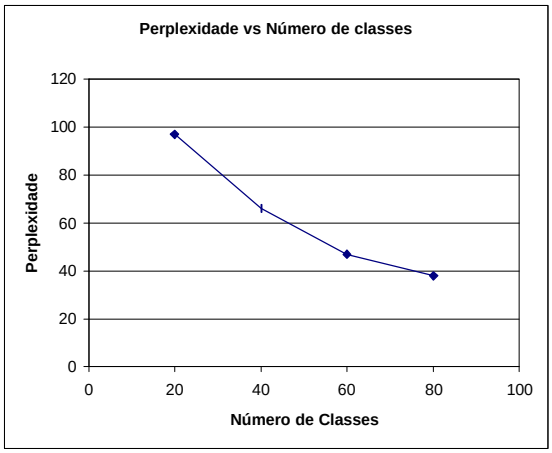

Figura 3-22: Comportamento da perplexidade com o número de classes (SA).

Para o treinamento utilizando o conjunto com 470 frases (3665 palavras), utilizamos o mesmo critério de $\frac{\Delta FL}{FL} < 0{,}001$.

Foram necessárias normalmente cerca de 500 épocas para concluir cada treinamento (cada época com 5000 mudanças de classe e uma taxa de decremento do parâmetro de controle de 0,95). O tempo de treinamento num computador Pentium II 300 Mhz ficou em torno de 3h.

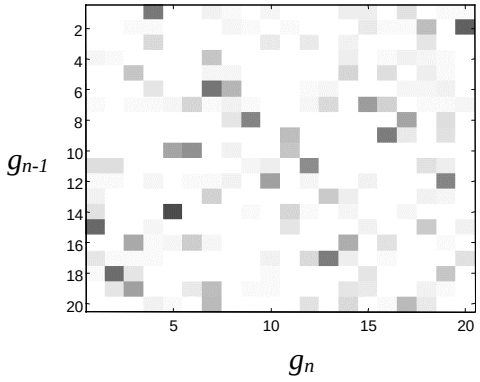

Figura 3-23: Probabilidades condicionais $P(g_n \mid g_{n-1})$ usando SA para 20 classes.

Na Figura 3-23, temos o gráfico correspondente às probabilidades condicionais de classe do Modelo da Língua obtido usando SA com 20 classes. Podemos verificar que mesmo utilizando somente 20 classes de palavras, a matriz de probabilidades condicionais ainda permanece esparsa.

4 Construção do Modelo da Língua Baseado em Gramática Independente de Contexto

4.1 Incursão pela Estrutura do Português

O Modelo da Língua proposto nesta seção foi desenvolvido com base em uma teoria de Gramática Gerativa [Chomsky65], a partir do uso de uma **gramática independente de contexto** (GIC).

Utilizaremos como ferramenta a **análise em constituintes imediatos** (análise em CI), seguindo o modelo apresentado em [Raposo78], mas utilizando também a Sintaxe $\overline{X}$ (X-barra) [Jackendoff77].

Apresentaremos aqui a estrutura de alguns tipos de frase da língua portuguesa, com o objetivo de fornecer as estruturas básicas que permitam a construção da gramática independente de contexto para o Modelo da Língua do sistema de reconhecimento de fala.

Outras linhas teóricas poderiam ter sido adotadas, mas não pretendemos discutir ou investigar aqui as vantagens de aplicação de uma ou de outra teoria, pois trata-se de um estudo complexo a que se dedicam muitos lingüistas, cujas conclusões ainda divergem em vários pontos. Ver [McClosky88] para se ter uma idéia da complexidade da tarefa.

4.1.1 Análise em Constituintes Imediatos

Sabemos que o significado da frase resulta, em parte, do significado individual das palavras. Porém, este significado também depende da composição global ou da ordenação que estas palavras têm na frase. Pode-se perceber que uma alteração na ordem das palavras muitas vezes altera o significado da frase:

O cachorro mordeu a criança pequena (4-1)

A criança pequena mordeu o cachorro (4-2)

Apesar de possuírem as mesmas palavras, as duas frases acima possuem significados bem diferentes, justamente por diferirem na ordenação das palavras.

Observa-se também que a ordenação das palavras não é aleatória, mas obedece a determinadas regras, pois uma seqüência qualquer pode tornar-se agramatical[1], ou seja, não pertencente à língua portuguesa:

*A cachorro criança mordeu o pequena (4-3)

A idéia de que as frases podem ser construídas pela combinação linear de palavras dá margem a uma *concepção linear* da estrutura das frases.

O modelo linear é o mais simples que se pode construir a partir da língua, não propondo qualquer estruturação interna à frase. Entretanto, existem certas palavras que se associam com mais facilidade. Na frase (4-1), podemos perceber os seguintes *grupos naturais* de palavras:

criança	pequena

(4-4)

o	cachorro

(4-5)

[1] A partir deste ponto, as frases agramaticais serão precedidas por um asterisco.

Esses grupos possuem importância tanto pelo significado que expressam, como pela influência na prosódia da frase falada. Por outro lado, associações como (4-6) não formam grupos naturais com as mesmas características apresentadas por (4-4) e (4-5).

mordeu	a	(4-6)

Observamos que alguns grupos naturais associam-se com outras unidades (palavras ou grupos), com as quais constituem grupos de dimensão maior. A frase é uma estruturação desses elementos em sucessíveis níveis de complexidade, mantendo uma relação de dependência.

Podemos pensar, então, que além da estrutura linear, a frase possui outra estruturação interna que define certas relações entre determinadas palavras na frase, a qual chamaremos de *estrutura hierárquica*.

Analisar as frases e descobrir os grupos naturais ou constituintes hierarquizáveis que formam a sua estrutura constitui a base do método denominado **análise em constituintes imediatos** (análise em CI).

Uma maneira de proceder à análise em CI é iniciando com a frase completa e dividindo-a nos maiores grupos naturais possíveis. Repetindo o procedimento com os blocos encontrados, chegaremos à estrutura hierárquica de constituintes da frase.

Nas seções seguintes, trataremos dos critérios usados na divisão de uma frase em grupos naturais. Por enquanto, tomemos a seguinte análise da frase (4-1):

o cachorro	mordeu a criança pequena	(4-7)

Cada uma das partes acima poderá ser dividida sucessivamente em outros grupos:

o cachorro		mordeu a criança pequena	
o	cachorro	mordeu a criança pequena	
		mordeu	a criança pequena
			a criança pequena
			criança pequena

Figura 4-1: Divisão em grupos da frase “o cachorro mordeu a criança pequena”.

A análise em CI da Figura 4-1 pode ser apropriadamente representada através da estrutura em árvore da figura seguinte:

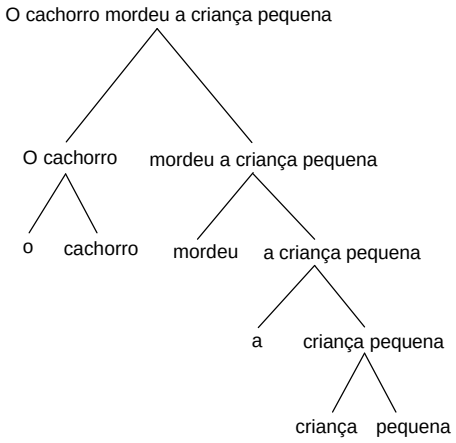

Figura 4-2: Análise em CI da frase “o cachorro mordeu a criança pequena”.

Um dos dispositivos que permitem a visualização da estrutura dos constituintes da frase é o *diagrama em árvore* ou simplesmente *árvore* (cf. seção 2.4.2). Podemos, então, representar a análise da Figura 4-2 através da árvore da Figura 4-3, na qual cada nó representa um dos blocos ou constituintes da frase.

A árvore de análise permite descrever tanto a relação hierárquica dos constituintes nas frases, quanto a sua classificação em determinadas categorias ou classes de constituintes, ambas definidas em termos da *relação de dominância* (cf. seção 2.4.2).

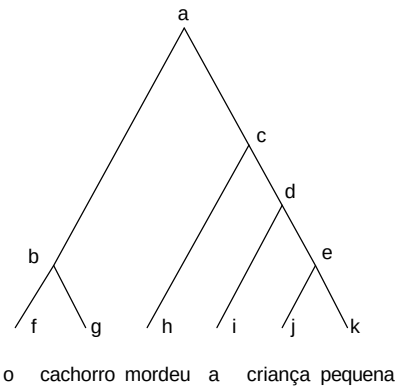

Figura 4-3: Árvore referente à análise em CI.

Podemos dizer que uma seqüência de elementos é um constituinte do tipo *a*, se *a* dominar exaustivamente todos os elementos da seqüência. Verificamos (vide Figura 4-3), por exemplo, que "o cachorro" é constituinte do tipo **b**, visto que **b** domina a seqüência. Da mesma forma, "mordeu a criança pequena**"** é constituinte do tipo **c**, pois este domina a seqüência. Por outro lado, "o cachorro mordeu**"** não pode ser considerado constituinte do tipo **b**, já que a relação de dominância não é satisfeita.

A relação formal que define o conceito de *constituinte imediato* é a relação de *dominância imediata*: dois ou mais nós serão constituintes imediatos de um determinado nó se forem imediatamente dominados por este. Dessa forma, os nós **b** e **c** são constituintes imediatos do nó **a**, pois são imediatamente dominados por este, o que é equivalente a dizer que "o cachorro" (nó b) e "mordeu a criança pequena" (nó c) são constituintes imediatos da frase "o cachorro mordeu a criança pequena" (nó a).

Nas seções seguintes, atribuiremos rótulos especiais aos nós da árvore de análise, em vez de usar simplesmente letras minúsculas, como na Figura 4-3.

4.1.2 Definindo os Constituintes Imediatos da Frase

As frases diferem quanto ao comprimento, à composição de palavras ou mesmo à construção interna, mas pontos comuns em sua estrutura interna podem ser captados por meio da análise em CI e da representação em árvore, procurando definir categorias que agrupem blocos similares e permitam visualizar as diferenças e semelhanças entre as frases.

Considerando as frases (4-8), podemos aplicar a análise em CI para obter os blocos iniciais que compõem cada frase, ou seja, os constituintes imediatos das frases.

Marta foi ao teatro (4-8)
O cachorro mordeu a criança
O meu primo que veio da China morreu

Podemos então dividir cada frase em dois blocos principais, conforme mostrado a seguir.

Marta	foi ao teatro	(4-9)
O cachorro	mordeu a criança	
O meu primo que veio da China	morreu	

Percebe-se facilmente que os blocos correspondentes de cada frase podem ser permutados resultando em frases gramaticais (4-10), sendo esta uma justificativa para incluí-los numa mesma categoria gramatical.

O meu primo que veio da China	mordeu a criança	(4-10)
O cachorro	foi ao teatro	
Marta	morreu	

Podemos propor outras frases em que tal teste não funciona, mas devido a uma incompatibilidade semântica entre os blocos e não porque eles não pertençam a categorias diferentes, conforme percebemos abaixo:

A casa da minha irmã	é linda	(4-11)
Marta	foi ao teatro	
*A casa da minha irmã	foi ao teatro	

Segundo Raposo [Raposo78], se efetuarmos a divisão da frase num ponto diferente, teremos um conjunto de blocos com uma capacidade de substituição muito menor. Se a divisão das frases for feita após o verbo, por exemplo, a substituição dos blocos gerará frases agramaticais:

*O meu primo que veio da China morreu	a criança	(4-12)
*O cachorro mordeu	ao teatro	
*Marta foi		

O primeiro bloco de cada uma das frases em (4-9) é classificado como *Sintagma Nominal* (SN), porque seu elemento central é um nome (Marta, cachorro, primo). O segundo bloco das frases

será classificado como *Sintagma Verbal* (SV), pois contém um verbo como elemento nuclear (foi, mordeu, morreu).

As estruturas em árvore simplificadas relativas às frases (4-8) seguem a mesma estrutura apresentada na Figura 4-4.

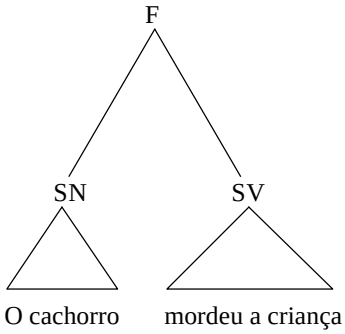

Figura 4-4: Árvore simplificada da frase “o cachorro mordeu a criança”.

Outros testes podem ser aplicados para verificar que os blocos correspondentes compartilham das mesmas características: **topicalização**, comparação com voz passiva e **colocação em posição de contraste** (vide [Raposo78]).

A topicalização consiste basicamente em deslocar o grupo de palavras para o início ou final da frase. Ex.: "*o cachorro* mordeu a criança" $\rightarrow$ "mordeu a criança, *o cachorro*".

A colocação em posição de contraste consiste em colocar o grupo de palavras entre as palavras "foi" e "que". Ex.: "*o cachorro* mordeu a criança" $\rightarrow$ "foi *o cachorro* que mordeu a criança".

4.1.3 Análise do Sintagma Nominal

Nesta seção, abordaremos a estrutura interna de constituintes do SN. Anteriormente, tomamos o SN que aparece na posição que a gramática tradicional classifica como sujeito, mas o SN pode ocorrer em diversas posições na estrutura da frase, inclusive como constituinte imediato do SV, ou mesmo como parte de um outro SN.

Tomemos a frase “o cachorro do vizinho mordeu a criança”. Ao analisá-la, seguindo a modelo inicialmente proposto por Chomsky [Chomsky65], podemos identificar o SN que desempenha o papel de sujeito da oração: “o cachorro do vizinho”. Aplicando a análise em CI,

dividimos o SN em dois constituintes “o cachorro” e “do vizinho”. A seqüência “o cachorro” constitui um outro SN, enquanto que a seqüência "do vizinho" é um Sintagma Preposicional (**SP**), formado por uma Preposição (**P**) mais um Sintagma Nominal. O elemento "do" corresponde justamente à contração da preposição "de" e do determinante "o" que faz parte do SN "o vizinho". O SN "o cachorro" pode ser analisado em dois constituintes: "o", classificado como Determinante (**Det**), e "cachorro", classificado como Nome (**N**), núcleo do SN "o cachorro". A estrutura completa do SN-sujeito pode ser vista na figura seguinte.

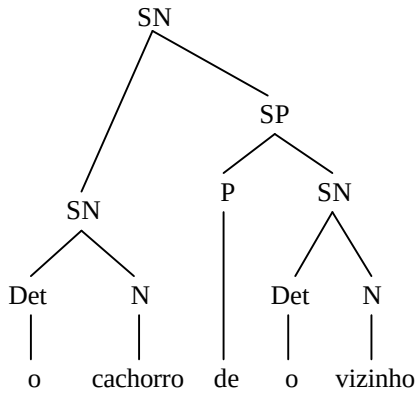

Figura 4-5: Estrutura do SN “o cachorro do vizinho”.

Observe que o produto da árvore da Figura 4-5 (“o cachorro de o vizinho”) não corresponde exatamente à frase inicial (“o cachorro do vizinho”). Para obtê-la, precisaríamos executar uma *transformação* sobre a árvore de modo que os elementos “de o” sejam substituídos pela forma contraída “do”.

Voltaremos a discutir o fenômeno das contrações com preposição na seção 4.2. Por enquanto, as frases serão analisadas como proposto na Figura 4-5.

À classe dos determinantes pertencerão os elementos que a gramática tradicional chama de artigos (o, a, um, uma, os, as) e pronomes demonstrativos (aquele, aquela, aquilo, este, esta, isto esse, essa, isso, etc.). Observe que estes elementos ocorrem antes do nome e não podem coexistir num mesmo SN, conforme podemos observar nos exemplos a seguir:

*O aquele cachorro (4-13)

*Um o cachorro

*Este aquele cachorro

Tomando agora o SN “todos os meus dois cachorros”, podemos observar a existência de outros tipos de elementos além do nome e do determinante. Usando o sistema adotado em [Raposo78], podemos definir a estrutura de constituintes da Figura 4-6 que acrescenta as categorias: pré-determinante ou **PréDet** (categorias que ocorrem à esquerda do determinante) e pós-determinante **PósDet** (categorias que ocorrem à direita do determinante).

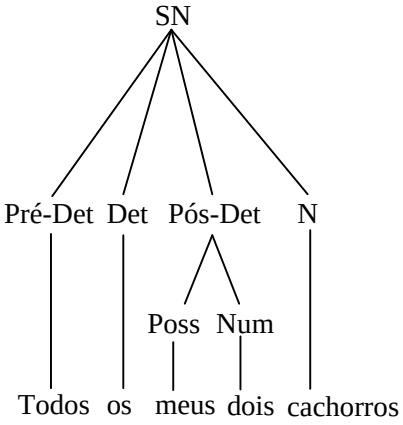

Figura 4-6: Estrutura do SN “todos os meus dois cachorros”.

Dentro do **PréDet** teremos categorias denominadas quantificadores (**Quant**): algum, algum de, todo, todos, qualquer, etc. Formando o **PósDet**, distinguimos duas categorias: Possessivos ou **Poss** (meu, teu, seu, etc.) e Numeral ou **Num** (dois, três, quatro, etc.).

Até agora, consideramos que o SN é uma estrutura formada em torno de um nome**,** seu núcleo, ao qual podemos ter associadas as categorias **Det**, **PréDet** e **PósDet**. Seguindo [Raposo78], podemos analisar seqüências que incluem adjetivos, conforme mostrado na figura a seguir.

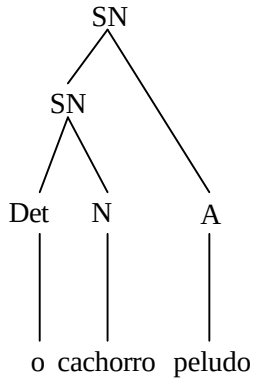

Figura 4-7: Estrutura do SN “o cachorro peludo”.

O adjetivo (**A**) "peludo" combinado ao SN "o cachorro" forma outro SN situado num nível superior da hierarquia.

Esta combinação entre **SN** e **A** funciona relativamente bem para os adjetivos pospostos ao nome, mas traz problemas quando tentamos analisar SNs com adjetivos antepostos, pois a estrutura em árvore adotada não permite cruzamento de ramos.

Conforme ressaltado em [Raposo78], a análise de SNs contendo adjetivos é uma tarefa complicada e divide a opinião de vários lingüistas.

Diante do problema relativo aos adjetivos, resolvemos aplicar a teoria denominada *Sintaxe* $\overline{X}$ [Jackendoff77], na qual utiliza-se uma categoria intermediária, $\overline{N}$, posicionada entre o Nome (*N*) e o Sintagma Nominal (chamado de $\overline{\overline{N}}$).

Por simplificação, usaremos os símbolos *N, N', N''* no lugar de *N,* $\overline{N}$ e $\overline{\overline{N}}$, seguindo a mesma notação adotada por Radford em [Radford88].

Segundo Radford, as categorias X, X' e X'' serão estruturadas como mostrado na Figura 4-8.

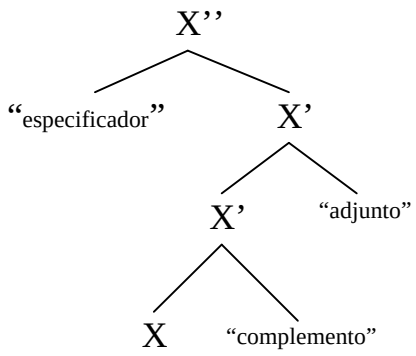

Figura 4-8: Estrutura de constituintes na Sintaxe $\overline{X}$.

Com relação ao Sintagma Nominal (SN ou N''), na posição de *especificador* estarão as categorias *PréDet*, *Det* e *PósDet*. Os adjetivos aparecem como *adjuntos* e os sintagmas preposicionais podem ser *adjuntos* ou *complementos*, dependendo de sua relação com o núcleo (N).

Na figura seguinte, temos os exemplos de um nome associado a um adjetivo posposto ("o cachorro **peludo"** - Figura 4-9a), a um adjetivo anteposto ("o **pequeno** barco" - Figura 4-9b), e a um numeral posposto ("a casa **doze**" - Figura 4-9c), todos funcionando como adjuntos.

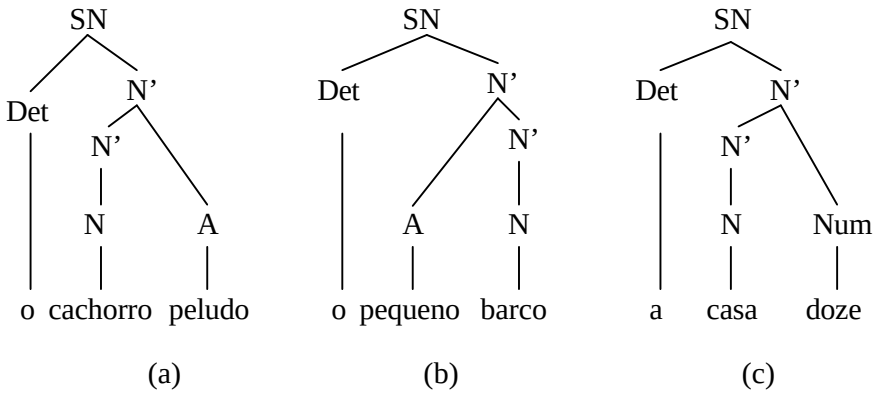

Figura 4-9: Estrutura SN formada com adjetivos e numeral.

Na Figura 4-10, temos a análise da frase "o estudante de Física com um rádio" utilizando a Sintaxe $\overline{X}$. Observe que o SP "com um rádio" funciona como adjunto de "estudante", enquanto que o SP "de Física" funciona como seu complemento (neste caso, um *complemento nominal*, segundo a gramática tradicional [Cunha85]).

Uma discussão sobre a estrutura envolvendo complementos e adjuntos pode ser encontrada em [Radford88].

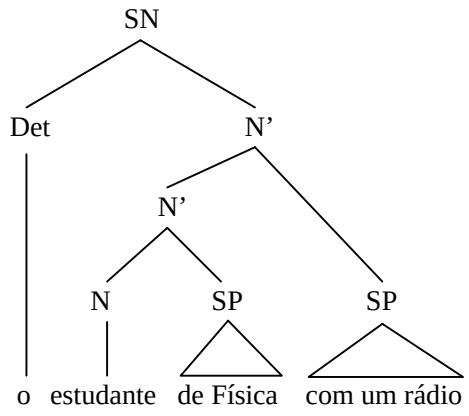

Figura 4-10: Estrutura do SN "o estudante de Física com um rádio" usando Sintaxe $\overline{X}$.

Dentro deste modelo, as estruturas da Figura 4-5 e da Figura 4-6 serão redefinidas como mostrado na figura seguinte.

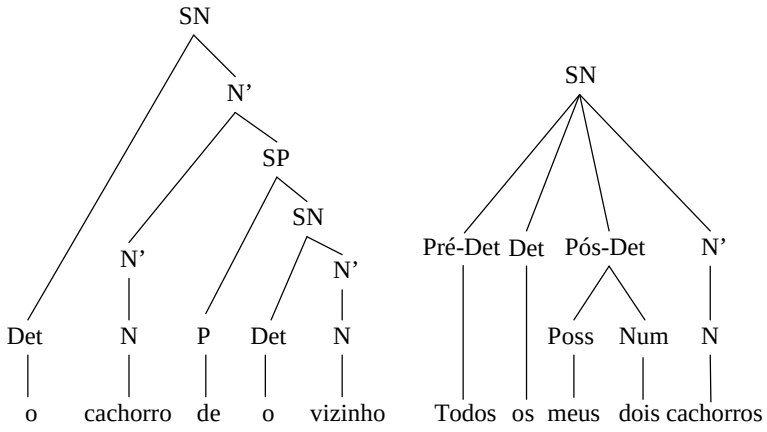

Figura 4-11: Estrutura do SN "o cachorro do vizinho".

A utilização da Sintaxe $\overline{X}$, neste trabalho, foi motivada inicialmente pelo problema de colocação dos adjetivos na estrutura do SN. No entanto, segundo Radford (vide [Radford81], p.112): "*existe alguma evidência empírica como suporte para a existência de categorias intermediárias entre as categorias lexicais e os sintagmas, de modo que poderíamos substituir as categorias X e SX da estrutura sintagmática pelo sistema mais rico da Sintaxe X-barra*"[1]. Para uma verificação dessas evidências, sugerimos consultar [Radford88] (capítulos 4 e 5).

Podemos estender a aplicação da Sintaxe $\overline{X}$ a sintagmas adjetivais (SA), conforme mostra a figura seguinte.

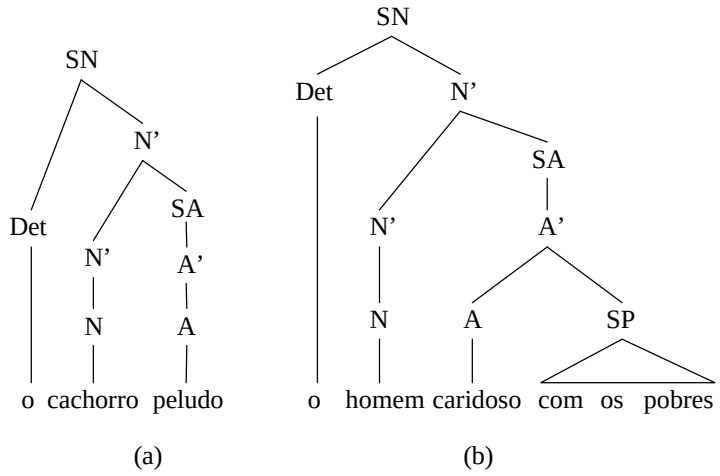

Figura 4-12: Sintagmas adjetivais usando sintaxe X-barra.

[1] Tradução minha

Na Figura 4-12b, “caridoso com os pobres” é um Sintagma Adjetival formado pelo adjetivo “caridoso” (núcleo) e seu complemento “com os pobres”.

A sintaxe $\overline{X}$ permite ainda representar estruturas de SA mais complexas como “o homem mais caridoso com os pobres” (vide [Radford88]), ilustrado na Figura 4-13.

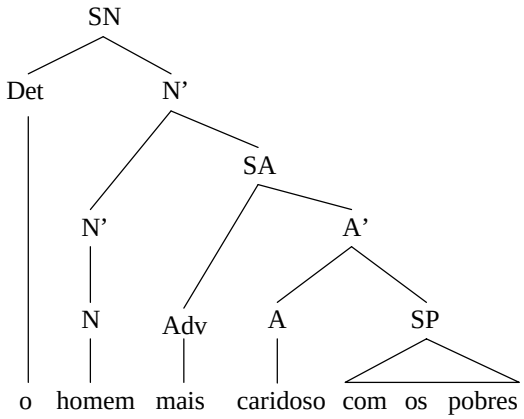

Figura 4-13: Estrutura SA usando Sintaxe $\overline{X}$.

O SN também pode ser formado por parte de uma estrutura frasal, conforme [Raposo78]. Tomemos, por exemplo, a frase "o cachorro que mordeu o menino morreu". Podemos analisá-la em dois constituintes:

o cachorro que mordeu o menino	morreu

(4-14)

Podemos justificar a divisão acima aplicando alguns testes e verificando que outras divisões não produzem grupos naturais. Supondo, por exemplo, a divisão em (4-15) e realizando os testes de topicalização e colocação em posição de contraste, obteremos as frases agramaticais (4-16) e (4-17), respectivamente.

o cachorro	que mordeu o menino	morreu

(4-15)

*que mordeu o menino morreu, o cachorro (4-16)

*foi *o cachorro* que *que mordeu o menino morreu* (4-17)

O mesmo não ocorre com os grupos obtidos na divisão (4-14), na qual os testes acima resultam em frases gramaticais:

morreu, o cachorro que mordeu o menino (4-18)

foi *o cachorro que mordeu o menino* que *morreu* (4-19)

Voltando à divisão (4-14), vemos que o primeiro constituinte corresponde ao SN que desempenha o papel de sujeito em relação ao verbo principal "morreu".

Este SN pode ser analisado em dois constituintes imediatos:

o cachorro	que mordeu o menino

Novamente, o primeiro constituinte corresponde a um SN. O segundo constituinte, que chamaremos de REL, é formado obrigatoriamente pelo pronome relativo "que" e por parte de uma estrutura frasal à qual falta um SN (neste caso o SN-sujeito). O SN ausente é na verdade referenciado pelo relativo "que" da oração relativa.

Por simplificação, não diferenciaremos a frase incompleta (faltando o SN-sujeito ou SN-objeto) que faz parte do nó REL, colocando-a dominada por um nó F.

A estrutura em árvore correspondente ao SN "o cachorro que mordeu o menino" pode ser vista na Figura 4-14.

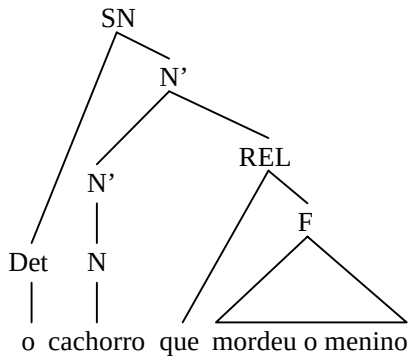

Figura 4-14: SN composto por estrutura frasal.

Permitiremos que um SN também seja formado pela coordenação de dois (ou mais) SNs, através do morfema “e”, conforme mostra a Figura 4-15.

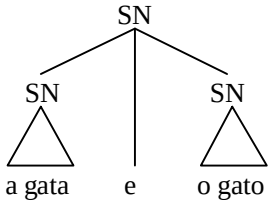

Figura 4-15: SN composto por estrutura interna de coordenação.

Seguindo o mesmo modelo, podemos construir um N’ formado por coordenação para explicar seqüências como “calças azuis e camisas brancas de Marta”, na qual podemos ter tanto as “calças azuis” quanto as “camisas brancas” pertencentes a “Marta”, ou somente as “camisas brancas” pertencentes a “Marta” (Verifique a Figura 4-16).

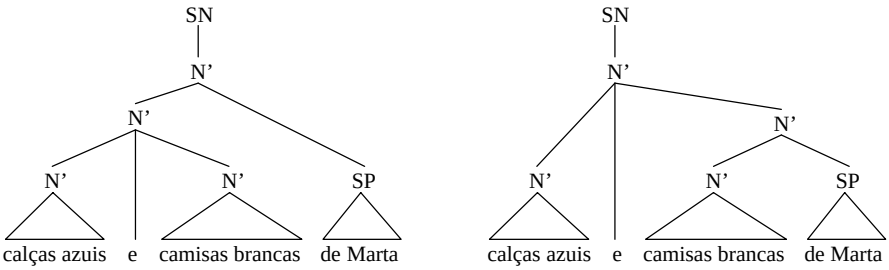

Figura 4-16: Duas interpretações possíveis para “calças azuis e camisas brancas de Marta”.

Finalmente, em frases como "**ele** pegou o livro", definiremos que o SN pode corresponder simplesmente a um pronome pessoal (**Pess**).

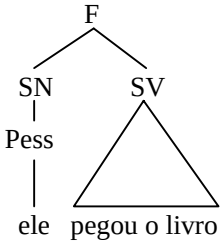

Figura 4-17: SN formado somente por pronome pessoal

4.1.4 Análise do Sintagma Verbal

Nesta seção, apresentaremos a estrutura interna do segundo constituinte imediato da frase, aquele que se forma em torno de um núcleo verbal: O Sintagma Verbal (SV).

Inicialmente, aplicaremos a mesma análise feita por em [Raposo78], mas o modelo final utilizará os conceitos da Sintaxe $\overline{X}$.

Tomemos a frase “o cachorro mordeu a criança”. Na seção anterior, dividimos esta frase em dois constituintes imediatos (vide Figura 4-4), sendo o segundo deles um SV (correspondente à seqüência “mordeu a criança”). A análise mais evidente deste SV consiste em dividi-lo em dois constituintes: O verbo “mordeu” e a seqüência “a criança”.

Pelo que já vimos, a seqüência “a criança” é um SN. Além disso, observe que podemos trocar as posições dos dois SNs da frase, resultando justamente na frase (gramatical) “a criança mordeu o cachorro”.

Definimos então a seguinte estrutura para o SV "mordeu a criança":

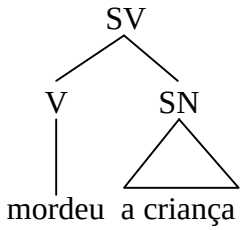

Figura 4-18: Estrutura básica do SV.

Passemos agora à frase “o professor entregou o livro ao aluno”. Procedemos à análise em CI da frase, obtendo:

o professor	entregou o livro ao aluno

(4-20)

O segundo bloco da frase corresponde a um SV. Para continuar a análise, temos as seguintes alternativas:

entregou	o livro	ao aluno	(4-21)

entregou o livro	ao aluno	(4-22)

entregou	o livro ao aluno	(4-23)

Realizando alguns testes, é possível perceber que a análise mostrada em (4-21) é mais promissora.

Observe, por exemplo, o teste da topicalização relativa à divisão (4-22). A seqüência "entregou o livro" não funciona como grupo natural:

*__Entregou o livro__, o professor ao aluno.

Realizando o teste da colocação em posição de contraste, relativa a divisão (4-23), temos que "o livro ao aluno" também não funciona como grupo natural:

*Foi **o livro ao aluno** que o professor entregou.

Por outro lado, usando os mesmos testes, conclui-se que a divisão (4-21) parece correta, ou seja, "o livro" e "ao aluno" constituem grupos naturais distintos e juntamente com "entregou" formam o sintagma verbal:

Foi o livro que o professor entregou ao aluno	(posição de contraste)
Foi ao aluno que o professor entregou o livro	(posição de contraste)
O livro, o professor entregou-o ao aluno	(topicalização)
Ao aluno, o professor entregou o livro	(topicalização)

A divisão adotada pode ser colocada em termos da estrutura em árvore da figura seguinte:

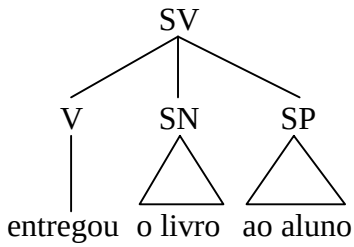

Figura 4-19: Estrutura do SV contendo SN e SP.

A gramática tradicional classifica a seqüência "o livro" como objeto direto e a seqüência "ao aluno" como objeto indireto. O conjunto "entregou o livro ao aluno" corresponde ao predicado da oração.

Tomemos agora a frase “o professor entregou o livro de matemática ao aluno”. O SV constituinte imediato da frase é “entregou o livro de matemática ao aluno” e contém dois SPs: “de matemática” e “ao aluno”. Podemos notar, entretanto, que os dois SPs estabelecem relações diferentemente na frase.

A seqüência “de matemática” especifica o tipo de livro dado ao aluno, ou seja, o SN “o livro” e o SP “de matemática” formam um novo grupo natural (um novo SN) “o livro de matemática”. O SP “ao aluno” indica o *receptor* do objeto dado (“livro”) , sendo constituinte imediato do SV.

A estrutura interna do SV pode ser representada pela árvore da Figura 4-20.

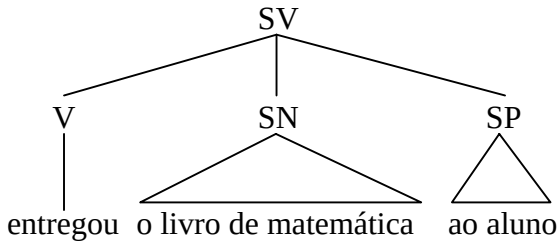

Figura 4-20: Estrutura do SV contendo SN composto.

Todas as estruturas discutidas acima, consideram o verbo e seus complementos (SN e SP na Figura 4-19, por exemplo) diretamente ligados ao nó SV. Utilizando a Sintaxe $\overline{X}$, estes três elementos serão constituintes imediatos de um nó V’, seguindo a mesma estrutura geral apresentada na Figura 4-8. Podemos redefinir a estrutura da Figura 4-19 como mostrado a seguir:

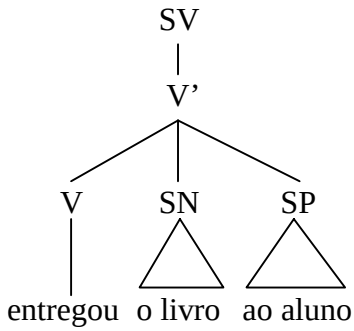

Figura 4-21: Sintagma Verbal usando Sintaxe $\overline{X}$.

O emprego da Sintaxe $\overline{X}$ na estruturação interna do sintagma verbal tem o objetivo principal de tratar melhor a ocorrência de locuções verbais e advérbios.

As locuções verbais são formadas por verbo auxiliar (**Vaux**) seguido pelo verbo principal numa forma nominal: infinitivo (**Vinf**), particípio (**Vpart**) ou gerúndio (**Vger**). Na figura seguinte, apresentamos as estrutura adotadas para as frases “o aluno está lendo o livro” e "o jarro foi quebrado pelo gato".

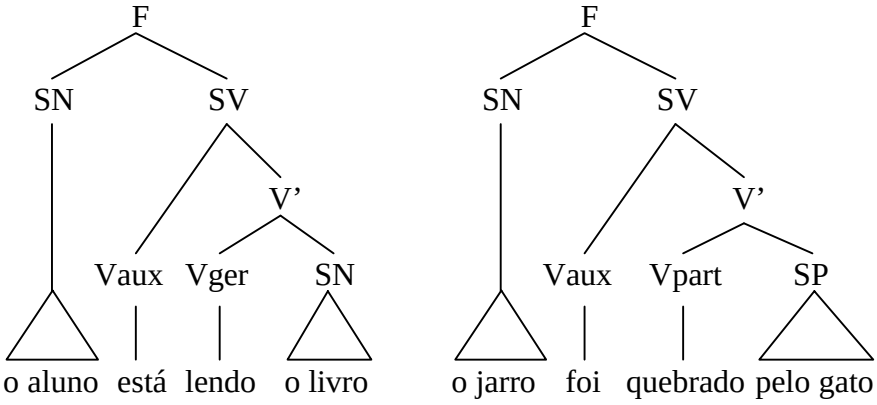

Figura 4-22: Estrutura do SV com locução verbal.

Um tipo diferente de frase é aquela que apresenta verbos como ser, estar e parecer, funcionando como **verbos copulativos** (verbos de ligação) nas seguintes frases:

O professor está cansado (4-24)

O rapaz é um engenheiro

Esses e alguns outros verbos são por vezes tratados diferentemente e alguns não os consideram verbos plenos, mas morfemas gramaticais com características verbais, não lhes atribuindo o rótulo *V* na representação em árvore, segundo [Raposo78].

Esse verbos apresentam algumas características que os distinguem dos demais: podem ocorrer com um SA como complemento e não podem possuir mais de um complemento.

Poderíamos tratar as frase com verbos copulativos como citado em [Raposo78], atribuindo-lhes o mesmo tipo de estrutura apresentada na figura seguinte. Nestes casos, o verbo copulativo é visto apenas como elemento de ligação entre o sujeito (“o rapaz” e “o professor”) e o predicativo (“um engenheiro” e “cansado”).

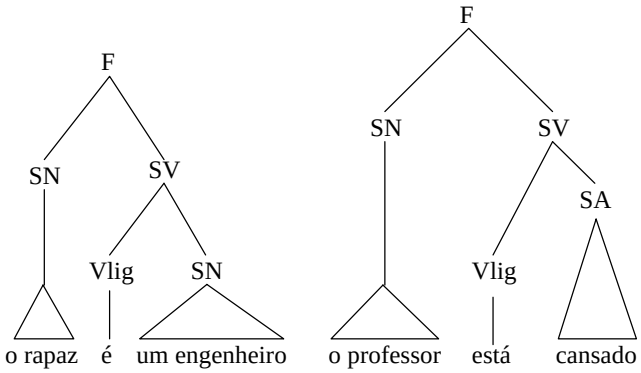

Figura 4-23: Frases com verbos copulativos.

Por questões práticas, no sentido de simplificar o Modelo da Língua e não nos aprofundarmos em uma discussão complexa, trataremos os verbos de ligação de maneira similar aos “verbos plenos”, adotando o modelo mostrado na figura abaixo. Este mesmo tratamento *aproximado* dos verbos de ligação foi adotado em [Raposo78], mas sugerimos que trabalhos futuros utilizem um modelo mais adequado, baseado em teoria lingüística bem fundamentada.

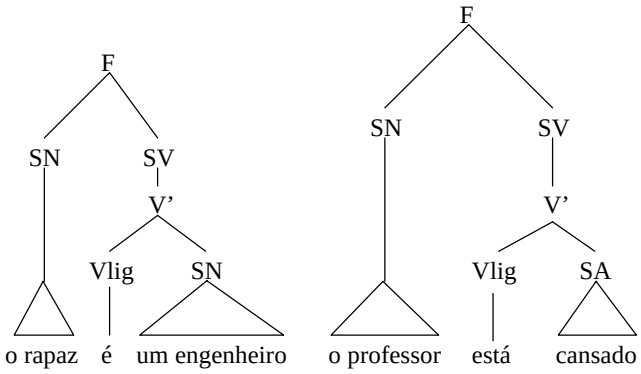

Figura 4-24: Tratamento aproximado dos verbos copulativos.

Da mesma forma que no caso do SN, o SV também pode ter parte de sua estrutura formada por uma estrutura frasal. Observe, por exemplo, a frase “a mulher disse que o padre estava doente”. A seqüência “o padre estava doente” constitui uma oração que complementa a oração parcial “a mulher disse”, à qual falta o complemento (aquilo que foi dito pela mulher). Adotamos, então, a estrutura em árvore da figura abaixo.

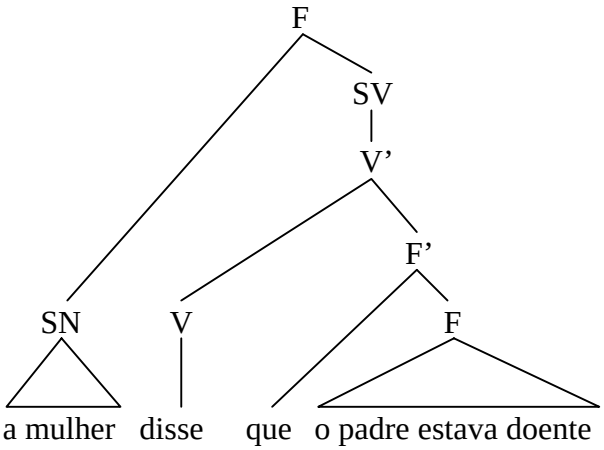

Figura 4-25: Estrutura frasal funcionando como complemento de verbo.

4.1.5 Definindo Outros Constituintes e Estruturas

Definiremos algumas estruturas que envolvem os constituintes vistos até o momento. Começaremos lembrando que ao analisar a estrutura das frases, admitimos que esta possuía somente dois constituintes imediatos: O SN e o SV. Na verdade, podemos identificar um outro constituinte imediato da frase.

Observe as duas frases seguintes:

O aluno entregou a prova ao professor
O aluno entregou a prova na semana passada

A primeira frase possui a mesma estrutura da frase cujo SV está mostrado na Figura 4-19, ou seja, o SV possui dois constituintes imediatos: um SN e um SP (complementos).

A segunda frase é semelhante à primeira, no que se refere à estrutura linear:

O aluno	entregou	a prova	ao professor
SN	V	SN	SP

O aluno	entregou	a prova	na semana passada
SN	V	SN	SP

Entretanto, a função sintática do SP na segunda frase difere com relação à primeira. Podemos perceber esta diferença através de um teste simples apresentado em [Raposo78], no qual fazemos uma pergunta relativa à frase analisada e verificamos sua gramaticalidade:

*O que fez o aluno **ao professor?**
O que fez o aluno **na semana passada**?

A primeira pergunta torna-se agramatical pois o SP “ao professor” é complemento do verbo e estará incluído na resposta à pergunta “o que **fez** o aluno?”. A pergunta correta seria “O que fez o aluno?” e a resposta “Entregou a prova ao professor”.

No caso da segunda pergunta, o SP “na semana passada” possui características adverbiais e não funcionará como complemento do verbo.

A diferenciação em termos de estrutura em árvore (Figura 4-26) é necessária para ressaltar as diferentes funções sintáticas desempenhadas pelos dois SPs. Seguindo o mesmo modelo adotado em [Raposo78] teremos as estruturas de análise da figura seguinte.

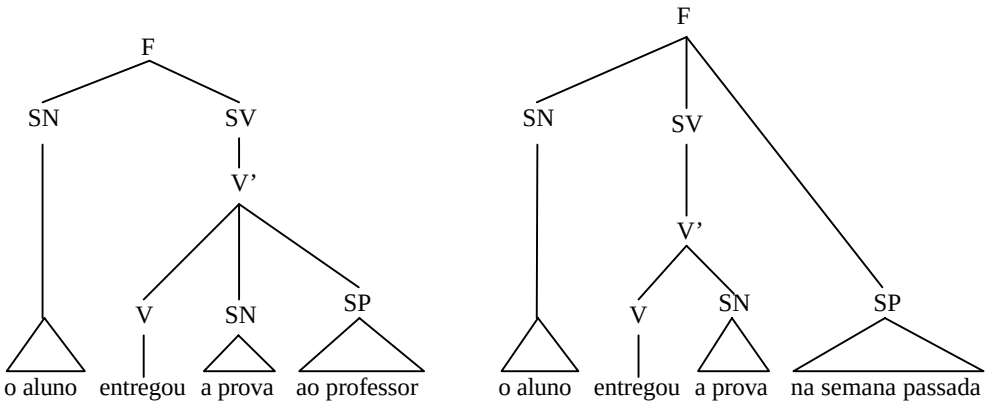

Figura 4-26: Diferenciação entre SP complemento verbal .

Também utilizaremos estruturas como a estrutura da frase "o aluno entregou a prova na semana passada" com frases como "o aluno entregou a prova ontem" (veja a figura seguinte).

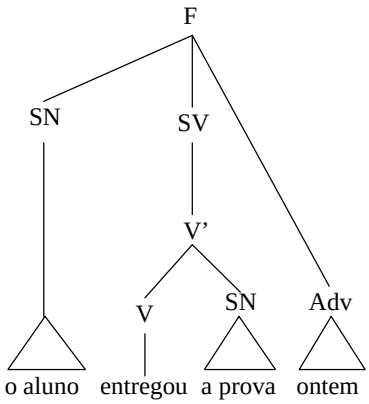

Figura 4-27: Frase com advérbio "ontem" .

Existem casos em que o advérbio está mais diretamente ligado ao SV (indicando a maneira como se dá a ação). Tomando as frases "o aluno rapidamente entregou a prova" e "o aluno entregou a prova rapidamente", podemos aplicar a Sintaxe $\overline{X}$ para obter as seguintes árvore de análise:

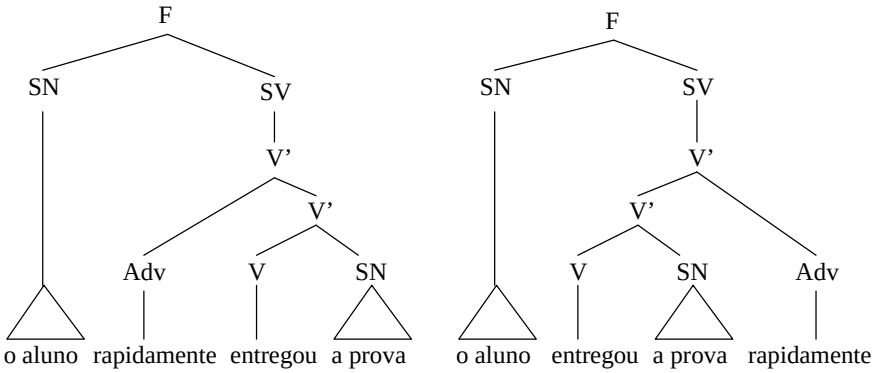

Figura 4-28: Advérbios na estrutura do SV.

A posição ocupada pelo advérbio na frase é algo irregular e não desejamos avaliar aqui todas as possibilidades, mas somente os casos discutidos acima. Os casos em que o advérbio ocupa posições diferentes na frase não serão tratados aqui, pois muitas vezes obtemos árvores de análise com cruzamento entre ramos, conforme mostrado na Figura 4-29. Nesta situação, a árvore de análise não poderá ser gerada por gramática independente de contexto, pois uma condição necessária é de que não exista cruzamento entre os ramos da árvore (vide seção 2.4.2). O mesmo tipo de problema é discutido em [Radford88].

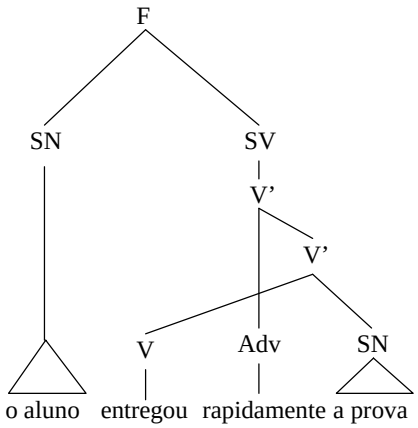

Figura 4-29: Problema de cruzamento de ramos da árvore.

Com relação a frases negativas do tipo “o aluno não entregou a prova”, não entraremos em detalhes. Elas serão simplesmente tratadas como mostra a Figura 4-30.

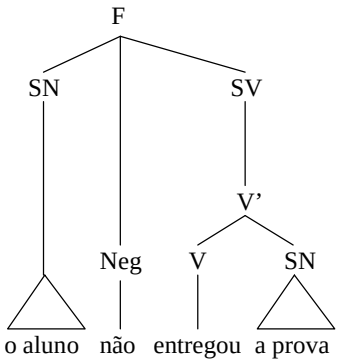

Figura 4-30: Estrutura de frase negativa.

As frases compostas por duas oração ligadas por coordenação, como na frase “O empresário foi ao banco e os caixas estavam cheios”, serão tratadas como mostra a figura a seguir:

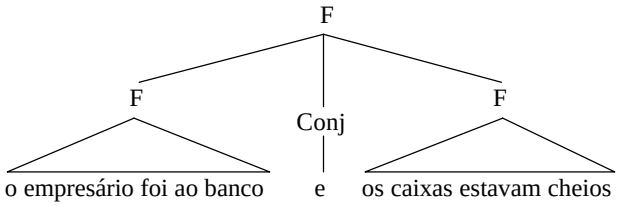

Figura 4-31: Orações coordenadas.

4.1.6 Observando as Ambigüidades Estruturais

Por vezes, uma frase pode possuir duas interpretações semânticas diferentes. Esta ambigüidade semântica decorre do fato de podermos atribuir duas estruturas distintas para a mesma frase, ou seja, atribuir diferentes relações entre os seus elementos de acordo com a interpretação usada.

Tomemos, por exemplo, a frase “O rapaz viu a moça com um binóculo”, cuja estrutura linear está representada a seguir.

SN	V	SN	SP
o rapaz	viu	a moça	com um binóculo

Pelo fato de termos acesso somente à estrutura linear da frase, não sabemos qual das interpretações é a correta:

1. O rapaz viu a moça e ela estava usando um binóculo (Figura 4-32a).
2. O rapaz usou um binóculo para ver a moça (Figura 4-32b).

Na figura seguinte temos a representação estrutural das duas interpretações acima.

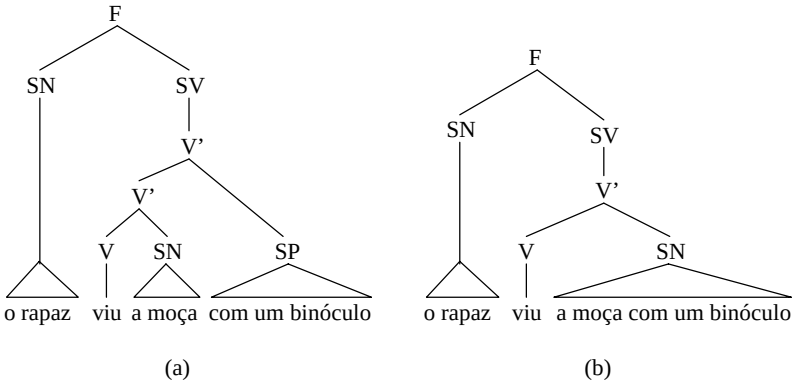

Figura 4-32: Ambigüidade estrutural das frases.

Observe que numa frase como “a sentença era muito longa” existe uma dúvida com relação ao sentido da palavra “sentença” (“sentença” = ”frase” ou “sentença” = “decisão judicial”), mas não existe uma ambigüidade estrutural.

A impossibilidade de lidar corretamente com as ambigüidades semânticas é um forte argumento contra a análise linear e a favor da hipótese de que as frases possuem uma estrutura hierárquica implícita e não somente a estrutura linear [Raposo78].

Quando ouvimos uma frase, podemos definir o seu sentido através do contexto em que está inserida. Imaginamos que a ambigüidade estrutural leva a confusões de interpretação, mas também pode ser um mecanismo útil na comunicação humana, segundo Eduardo Raposo: “A ambigüidade estrutural é um mecanismo de economia lingüistica, e, como tal, extremamente útil à comunicação humana (...). Se não existisse a ambigüidade estrutural, para cada tipo de situação contextual, necessitaríamos possivelmente de frases com uma organização linear diferente, o que constituiria uma sobrecarga para a nossa capacidade de processamento lingüístico”.

4.2 Construção da Gramática Independente de Contexto

A construção da gramática independente de contexto consiste na definição das regras de produção ou regras de reescrita da gramática. As regras serão formadas com base nas estruturas em árvore definidas nas seções anteriores. A gramática obtida poderá gerar um conjunto infinito de frases, mas é importante que na fase de teste do sistema, todas as frases usadas sejam corretamente analisadas através desta gramática.

Tomando a frase “o cachorro mordeu a criança”, podemos definir a seguinte árvore de análise:

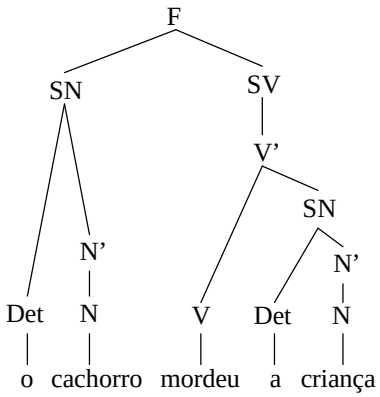

Figura 4-33: Árvore da frase “o cachorro mordeu a criança”.

Desta árvore podemos observar que o nó F divide-se em dois constituintes imediatos (SN e SV), correspondendo a subestrutura seguinte.

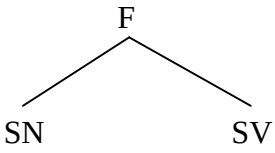

Figura 4-34: Subestrutura da frase.

Podemos gerar a subestrutura anterior através da seguinte regra de produção:

$$F \rightarrow SN \;\; SV \tag{4-25}$$

Observe que a regra acima define F como nó dominante e SN e SV como seus constituintes imediatos, respeitando também a ordenação linear na subestrutura.

Continuando a analisar a estrutura da Figura 4-33, verificamos que o SN se divide em dois constituintes (Det e N'). De maneira análoga à anterior, podemos definir a regra (4-26) que gera a subestrutura da Figura 4-35.

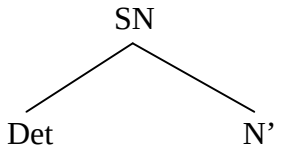

Figura 4-35: Subestrutura do SN.

$$SN \rightarrow Det \;\; N' \qquad (4\text{-}26)$$

Considerando uma subestrutura abstrata como a da Figura 4-36 , podemos definir a regra correspondente como (4-27).

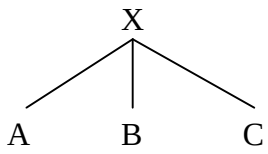

Figura 4-36: Subestrutura genérica.

$$X \rightarrow A\,B\,C \qquad (4\text{-}27)$$

Usando o algoritmo acima, podemos escrever as seguintes regras de produção referentes à árvore da Figura 4-33:

$$F \rightarrow SN \;\; SV \tag{4-28}$$
$$SV \rightarrow V'$$
$$SN \rightarrow Det \;\; N'$$
$$N' \rightarrow N$$
$$V' \rightarrow V \; SN$$

Deve-se observar que aplicando as regras acima, obteremos a subestrutura:

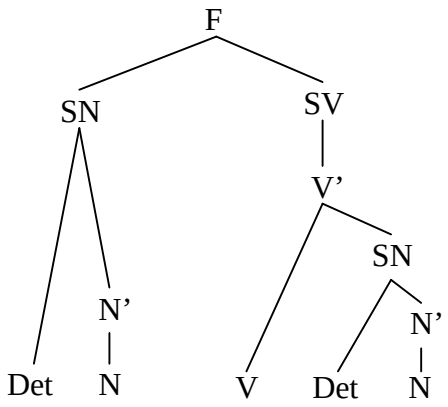

Figura 4-37: Subestrutura obtida pela aplicação das regras de produção.

Para chegar à árvore da Figura 4-33, precisaremos das regras que introduzem os elementos lexicais (as palavras):

$$Det \rightarrow o \tag{4-29}$$
$$Det \rightarrow a$$
$$N \rightarrow criança$$
$$N \rightarrow cachorro$$
$$V \rightarrow mordeu$$

Para simplificar a representação destas regras, podemos utilizar a notação $N \rightarrow criança, cachorro$ indicando que uma das palavras deve ser escolhida na expansão do símbolo N. Assim, as regras (4-29) podem ser escritas como:

$$Det \rightarrow o, a \tag{4-30}$$
$$N \rightarrow criança, cachorro$$
$$V \rightarrow mordeu$$

Aplicando todo o conjunto de regras obtido, é possível encontrar a árvore de análise da Figura 4-33.

A árvore parcial da Figura 4-37 pode ajustar-se a um número infinito de frases que possuam a mesma estrutura básica da frase "o cachorro mordeu a criança" (por exemplo, "o gato comeu o rato" e "as meninas compraram o livro").

Poderíamos analisar as novas frases, mantendo inalterado o conjunto de regras de reescrita (4-28) e introduzindo as novas palavras na regras que expandem as categorias lexicais (4-30). Para as frases "o gato comeu o rato" e "as meninas compraram o livro", bastaria utilizar as regras de reescrita (4-31), juntamente com as regras (4-28), para obter a árvore de análise da Figura 4-38.

$$Det \rightarrow o, a, as \tag{4-31}$$
$$N \rightarrow criança, cachorro, livro, meninas, gato, rato$$
$$V \rightarrow mordeu, comeu, compraram$$

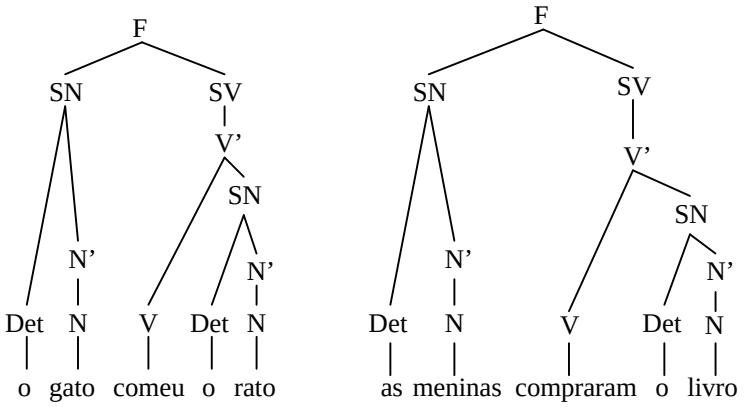

Figura 4-38: Árvores de análise das novas frases.

Tomando agora a frase "o cachorro de Marta mordeu a criança pequena", temos a árvore de análise da figura seguinte.

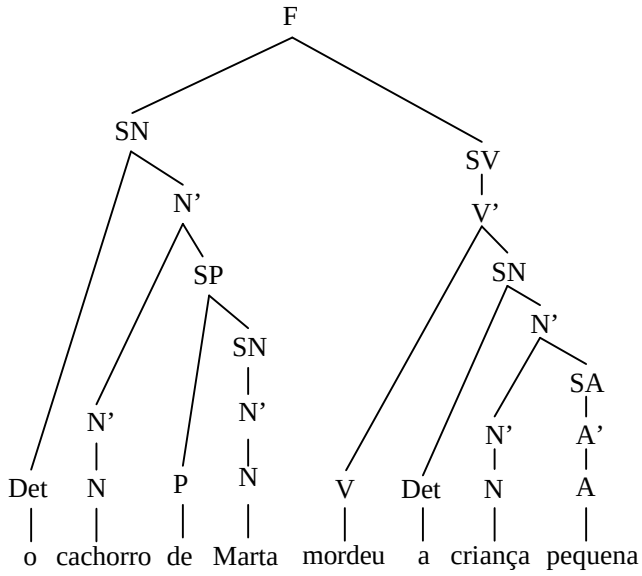

Figura 4-39: Árvore da frase “o cachorro de Marta mordeu a criança pequena”.

Seguindo o mesmo procedimento adotado no exemplo anterior, podemos chegar ao seguinte conjunto de regras:

$$
\begin{aligned}
F &\rightarrow SN\ SV & N' &\rightarrow N \\
SN &\rightarrow Det\ N' & SP &\rightarrow P\ SN \\
SN &\rightarrow N' & SA &\rightarrow A' \\
SV &\rightarrow V' & A' &\rightarrow A \\
N' &\rightarrow N'\ SP & V' &\rightarrow V\ SN \\
N' &\rightarrow N'\ SA
\end{aligned}
\qquad (4\text{-}32)
$$

No conjunto de regras temos duas expansões possíveis para o símbolo N’: $N' \rightarrow N'\ SP$ e $N' \rightarrow N'\ SA$. As duas regras podem ser expressas através de uma única regra utilizando as chaves para indicar que uma das categorias deve ser escolhida: $N' \rightarrow N' \begin{Bmatrix} SP \\ SA \end{Bmatrix}$.

No caso do símbolo SN, temos as regras de expansão $SN \rightarrow Det\ N'$ e $SN \rightarrow N'$, que podem ser representadas por $SN \rightarrow (Det)\ N'$, onde os parêntesis indicam que a categoria Det é opcional na expansão.

Observe na estrutura acima que alguns nós são expandidos em apenas uma nova categoria, conforme ilustra a Figura 4-40.

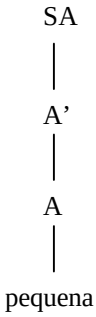

Figura 4-40: Expansões do tipo $A \rightarrow B$.

Essas expansões podem parecer inúteis a princípio (poderíamos tentar simplificá-las, eliminando categorias, por exemplo), mas elas tornam as regras de produção mais homogêneas, principalmente, à medida que estruturas mais complexas vão sendo consideradas.

Devido à existência de frases compostas por orações coordenadas no conjunto de teste, resolvemos propor uma regra de produção que considere estas estruturas: $F \rightarrow F$ "e" F.

No capítulo anterior e também na seção 4.1.3, chamamos a atenção para o fenômeno das contrações com preposição. Elas constituem um problema na definição das regras de produção, já que por limitações no sistema de reconhecimento usado, torna-se inviável realizar transformações levando elementos como “de” e “o” na forma contraída “do”. Visto que o Modelo da Língua atuará predizendo a palavra seguinte a uma frase parcial, o tratamento das contrações com preposição, utilizando as regras de produção definidas até aqui, implicaria num algoritmo de busca que considerasse essas hipóteses.

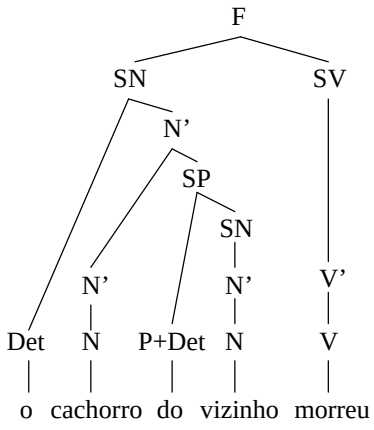

Figura 4-41: Estrutura da frase “o cachorro do vizinho morreu”.

Para resolver o problema, sem utilizar métodos complexos, propomos tratar as contrações com preposição da mesma forma que foi feito no capítulo anterior, colocando-as numa categoria distinta, denominada **P+Det** (Preposição+Determinante). Na Figura 4-41, mostramos a frase "o cachorro do vizinho morreu" tratada desta maneira.

No apêndice A, apresentamos o conjunto de regras de produção adotado para o modelo de língua baseado em gramática independente de contexto. As regras foram escritas manualmente a partir das estruturas discutidas em toda a seção 4.1, aplicando os procedimentos que acabamos de apresentar.

4.3 Implementando o Analisador

Analisar uma frase implica em definir uma descrição estrutural da frase segundo o modelo proposto pela gramática, assumindo que a sentença é gramatical. Até o momento, discutimos a construção da gramática cujo conjunto de regras descrevem conhecimentos sintáticos da língua. Para utilizar este modelo no reconhecimento de fala, falta-nos adotar um *procedimento de análise (parsing)* que aplique as regras de produção da gramática na construção da análise associada às frases de entrada.

Para as gramáticas independentes de contexto, a forma clássica de representar a análise é através da estrutura em árvore, conforme apresentado nos capítulos anteriores.

Existem duas estratégias que podem ser adotadas na análise de uma frase: Análise *top-down* ou *bottom-up* (vide [Deller*93]). A análise top-down consiste em começar pelo símbolo inicial e aplicar as regras no sentido dos símbolos terminais (para baixo), até que o produto da árvore corresponda à seqüência de palavras analisada A análise bottom-up parte da seqüência de palavras e aplica as regras até que seja produzida a árvore de análise com nó raiz rotulado por S.

Para as GICs, os algoritmos de análise mais conhecidos são o algoritmo CYK (Cocke-Younger-Kasami) [Younger67] [Kasami65] e o algoritmo Earley [Earley70]. No que diz respeito à complexidade computacional, estes algoritmos apresentam, no pior caso, complexidade $O(n^3)$, onde n corresponde ao comprimento da frase. Existe um fator multiplicativo que depende do tamanho da gramática e vale $O(G^2)$, onde G é o tamanho da gramática (expresso apropriadamente em termos do número de regras e do número de símbolos não-terminais). Na prática, a maioria dos

algoritmos trabalha muito melhor que o pior caso e o fator limitante é o tamanho da gramática (para maiores detalhes, consulte [SHLT96]).

Neste capítulo, descreveremos a implementação do analisador sintático em linguagem C++, usando o algoritmo Earley, pois este apresenta grande eficiência em tempo e espaço de busca, além de não exigir que as regras de produção da GIC estejam numa forma específica (na forma normal de Chomsky, como às vezes é assumido). Além disso, o algoritmo Earley pode ser modificado para utilizar gramáticas estocásticas independentes de contexto [Stolcke95], possibilitando o cálculo das *probabilidades de prefixo* usadas na predição de palavras ou na extração das probabilidades *m*-gram [Stolcke*94][1].

4.3.1 Algoritmo Earley

O algoritmo Earley [Earley70] é um algoritmo do tipo *top-down* que constrói *derivações à esquerda* de seqüências de símbolos terminais usando o conjunto de regras de produção da gramática independente de contexto. O analisador busca todas as possíveis derivações consistentes com a seqüência de entrada. À medida que o algoritmo avança, as derivações existentes são expandidas ou interrompidas, conforme novas opções são criadas ou ambigüidades são resolvidas.

O analisador Earley mantém um **conjunto de estados** para cada palavra de entrada, o qual armazena todas as derivações em andamento. Os conjuntos de estados juntos formam o **quadro Earley** .

Um **estado** é representado como (4-33), onde *X* é um símbolo não-terminal da gramática , l e m são seqüências de não-terminais e/ou terminais, enquanto *i* e *k* são índices de posição na seqüência de palavras de entrada explicados mais adiante.

$$i :\ _{k}X \rightarrow l.m \qquad (4\text{-}33)$$

[1] Conhecendo a probabilidade $P(w_1...w_n)$ de prefixos arbitrários $w_1...w_n$, podemos calcular as probabilidades relativas às palavras seguintes usando $P(w_{n+1} \mid w_1...w_n) = \frac{P(w_1...w_n w_{n+1})}{P(w_1...w_n)}$.

Os estados são criados a partir das regras de produção da gramática. O estado (4-33) corresponde à regra de reescrita (4-34).

$$X \rightarrow lm \qquad (4\text{-}34)$$

Suponha que a seqüência de palavras é representada por $w_1,...,w_N$, onde *N* é o número de palavras da frase. O índice *i* em (4-33) define que a seqüência de palavras $w_1,...,w_i$ encontra-se a esquerda do ponto (palavras já processadas pelo analisador). Os estados que possuem o mesmo valor de *i* estão reunidos no conjunto de estados *i*.

Existirão, na verdade, *N*+1 conjuntos de estados no quadro Earley, pois o *conjunto de estados 0* descreve o estado do analisador antes que qualquer palavra seja processada.

Em (4-33), o não-terminal *X* está sendo expandido a partir da posição *k* da entrada, ou seja, a derivação a partir de *X* produzirá uma seqüência de palavras iniciando na posição *k*. A expansão de *X* utilizando a regra (4-34) foi concluída até a posição (na regra) indicada pelo ponto e definida (na seqüência de palavras) pelo valor de *i*.

Um **estado completo** é aquele cujo símbolo não terminal do lado esquerdo já foi totalmente expandido, conforme podemos verificar pela posição do ponto em (4-35).

$$i :\ _kX \rightarrow lm. \qquad (4\text{-}35)$$

A operação do analisador é definida em termos de três operações básicas que agem sobre o conjunto de estados atual e sobre o símbolo de entrada atual, possivelmente adicionando novos estados ao quadro Earley:

a) Predição

Para cada estado $i :\ _kX \rightarrow l.Ym$, onde *Y* é o primeiro símbolo não-terminal após o ponto, aplica-se todas as regras do tipo $Y \rightarrow v$, expandindo *Y* e adicionando estados do tipo $i :\ _iY \rightarrow .v$ ao quadro Earley.

Um estado produzido por predição é chamado de *estado predito* e cada predição corresponde a uma expansão em potencial de um símbolo não-terminal numa derivação à esquerda.

b) Exame

Para cada estado do tipo $i: {}_kX \rightarrow l.am$, no qual a representa um símbolo terminal igual ao símbolo de entrada atual ($w_{i+1} = a$), adicione o estado $i+1: {}_kX \rightarrow la.m$ ao quadro Earley.

O procedimento de exame garante que os símbolos terminais produzidos numa derivação sejam iguais aos símbolos terminais de entrada.

c) Conclusão

Para cada *estado completo* $i: {}_jY \rightarrow v.$ e cada estado no conjunto j ($j \leq i$) que possua o símbolo Y imediatamente à direita do ponto, isto é, estados do tipo $j: {}_kX \rightarrow l.Ym$, adicione o estado $i: {}_kX \rightarrow lY.m$ ao quadro.

Um estado produzido pelo procedimento de conclusão será chamado de *estado completado*. Cada conclusão corresponde à finalização de uma expansão de um não-terminal iniciada por um passo de predição.

Utilizando o conjunto de regras de reescrita (4-36), já definidas na seção 4.2, apresentamos na Tabela 4-1 a aplicação do algoritmo Earley na análise da frase “o cachorro mordeu a criança”.

$$F \rightarrow SN\ \ SV \qquad (4\text{-}36)$$
$$SV \rightarrow V'$$
$$SN \rightarrow Det\ \ N'$$
$$N' \rightarrow N$$
$$V' \rightarrow V\ SN$$
$$Det \rightarrow o$$
$$Det \rightarrow a$$
$$N \rightarrow criança$$
$$N \rightarrow cachorro$$
$$V \rightarrow mordeu$$

conjunto 0	conjunto 1	conjunto 2	conjunto 3	conjunto 4	conjunto 5
	o	*cachorro*	*mordeu*	*a*	*criança*
${}_0e \rightarrow .\,\mathrm{F}$ **(predição)** ${}_0\mathrm{F} \rightarrow .\,\mathrm{SN\ SV}$ ${}_0\mathrm{SN} \rightarrow .\,\mathrm{Det\ N'}$ ${}_0\mathrm{Det} \rightarrow .\,\mathrm{o}$ ${}_0\mathrm{Det} \rightarrow .\,\mathrm{a}$	**(exame)** ${}_0\mathrm{Det} \rightarrow \mathrm{o}.$ **(conclusão)** ${}_0\mathrm{SN} \rightarrow \mathrm{Det}.\,\mathrm{N'}$ **(predição)** ${}_1\mathrm{N'} \rightarrow .\,\mathrm{N}$ ${}_1\mathrm{N} \rightarrow .\,\mathrm{cachorro}$ ${}_1\mathrm{N} \rightarrow .\,\mathrm{menino}$	**(exame)** ${}_1\mathrm{N} \rightarrow \mathrm{cachorro}.$ **(conclusão)** ${}_1\mathrm{N'} \rightarrow \mathrm{N}.$ ${}_0\mathrm{SN} \rightarrow \mathrm{Det\ N'}.$ ${}_0\mathrm{F} \rightarrow \mathrm{SN}.\,\mathrm{SV}$ **(predição)** ${}_2\mathrm{SV} \rightarrow .\,\mathrm{V'}$ ${}_2\mathrm{V'} \rightarrow .\,\mathrm{V\ SN}$ ${}_2\mathrm{V} \rightarrow .\,\mathrm{mordeu}$	**(exame)** ${}_2\mathrm{V} \rightarrow \mathrm{mordeu}.$ **(conclusão)** ${}_2\mathrm{V'} \rightarrow \mathrm{V}.\,\mathrm{SN}$ **(predição)** ${}_3\mathrm{SN} \rightarrow .\,\mathrm{Det\ N'}$ ${}_3\mathrm{Det} \rightarrow .\,\mathrm{o}$ ${}_3\mathrm{Det} \rightarrow .\,\mathrm{a}$	**(exame)** ${}_3\mathrm{Det} \rightarrow \mathrm{a}.$ **(conclusão)** ${}_3\mathrm{SN} \rightarrow \mathrm{Det}.\,\mathrm{N'}$ **(predição)** ${}_4\mathrm{N'} \rightarrow .\,\mathrm{N}$ ${}_4\mathrm{N} \rightarrow .\,\mathrm{cachorro}$ ${}_4\mathrm{N} \rightarrow .\,\mathrm{criança}$	**(exame)** ${}_4\mathrm{N} \rightarrow \mathrm{criança}.$ **(conclusão)** ${}_4\mathrm{N'} \rightarrow \mathrm{N}.$ ${}_3\mathrm{SN} \rightarrow \mathrm{Det\ N'}.$ ${}_2\mathrm{V'} \rightarrow \mathrm{V\ SN}.$ ${}_2\mathrm{SV} \rightarrow \mathrm{V'}.$ ${}_0\mathrm{F} \rightarrow \mathrm{SN\ SV}.$ ${}_0e \rightarrow \mathrm{F}.$

Tabela 4-1: Exemplo de aplicação do algoritmo Earley (quadro Earley).

Os estados deveriam ser representados como $i : {}_kX \rightarrow l.m$, mas na Tabela 4-1, representamos cada estado por ${}_kX \rightarrow l.m$. O índice i está indicado na parte superior da tabela (número do conjunto). Temos, por exemplo, o estado ${}_0\mathrm{SN} \rightarrow \mathrm{Det}.\,\mathrm{N'}$ no conjunto 1, que corresponde ao estado Earley $1 : {}_0\mathrm{SN} \rightarrow \mathrm{Det}.\,\mathrm{N'}$, seguindo a notação definida em (4-33).

Utilizando uma gramática independente de contexto $G = (V_N, V_T, R, S)$, a análise de uma seqüência de entrada será realizada partindo do estado inicial (4-37) e executando exaustivamente as três operações sobre cada conjunto de estados até que nenhum estado seja criado.

$$0 : {}_0e \rightarrow .S \qquad (4\text{-}37)$$

Devemos ressaltar que o símbolo de partida S corresponde ao símbolo F na gramática definida pelas regras (4-36).

Após processado o último símbolo de entrada (palavra), o analisador conclui que a seqüência de entrada foi reconhecida pela gramática *G*, se for verificada a existência do estado $N : {}_0e \rightarrow S.$, onde *N* é o comprimento da seqüência de entrada.

No exemplo da Tabela 4-1, observa-se que o estado $5 : {}_0e \rightarrow \mathrm{F}.$ está presente no conjunto 5, portanto a gramática (4-36) reconhece a frase “o cachorro mordeu a criança”.

Uma vez que o estado final é atingido, podemos usar um procedimento recursivo que recupere a árvore de análise correspondente à seqüência de entrada (vide [Stolcke95]). Entretanto, antes de definir este procedimento, vamos observar a relação existente entre os estados Earley e a árvore de análise da frase.

Em [Stolcke95] temos o seguinte lema: "Um analisador Earley produz o estado $i:{}_kX \to l.m$ se somente se existir uma derivação parcial (derivação à esquerda) $S \stackrel{*}{\Rightarrow} w_1...w_{k-1}Xn \Rightarrow w_1...w_{k-1}lmn \stackrel{*}{\Rightarrow} w_1...w_{k-1}w_k...w_imn$, correspondente ao prefixo de entrada $w_1...w_i$".

Lembrando que existe uma relação biunívoca entre a derivação à esquerda e a seqüência de regras de reescrita aplicadas (cf. seção 2.4.2), podemos encontrar a árvore de análise, usando a relação existente entre regra de reescrita e estrutura em árvore (vide regra (4-27) e Figura 4-36 na seção 4.2).

Assim, o procedimento recursivo que recupera a árvore de análise, tomará um estado $i:{}_kX \to l.m$ como entrada e retornará a árvore correspondente à derivação parcial associada a este estado.

A árvore de análise da seqüência de entrada (frase) será obtida executando este procedimento recursivo sobre o estado final $N:{}_0e \to S.$ gerado pelo algoritmo Earley[1]. Na prática, o algoritmo "segue" a ligação entre os estados Earley (vide Tabela 4-2), gerando árvores parciais que compõem a árvore de análise da frase.

A seguir, apresentamos o procedimento recursivo que constrói a árvore de análise [Stolcke95]:

Procedimento *Recupera_Árvore* ($i:{}_kX \to l.m$ **)**

1. Se $l = e$, retorne uma árvore sem nós-filhas, com a raiz rotulada como X.

[1] Devido a ambigüidades estruturais e outras situações, poderemos obter não apenas uma árvore de análise, mas várias possíveis árvores associadas a uma mesma frase.

2. Se l terminar em um símbolo terminal a ($l = l'a$), então faça T = *Recupera_Árvore(* $i-1$: ${}_kX \rightarrow l'.am$ *)*, acrescente uma folha com rótulo *a* como o nó-filha mais à direita da raiz de *T* e retorne *T*.
3. Se l terminar em um símbolo não-terminal Y ($l = l'Y$), então encontre o estado predecessor ${}_jY \rightarrow n.$ do estado atual, chame este procedimento recursivamente para obter as árvores T = *Recupera_Árvore(* j: ${}_kX \rightarrow l'.Ym$ *)* e *T'* = *Recupera_Árvore(* i: ${}_jY \rightarrow n.$ *)*, acrescente a árvore *T'* como nó-filha mais à direita da raiz de *T* e retorne *T*.

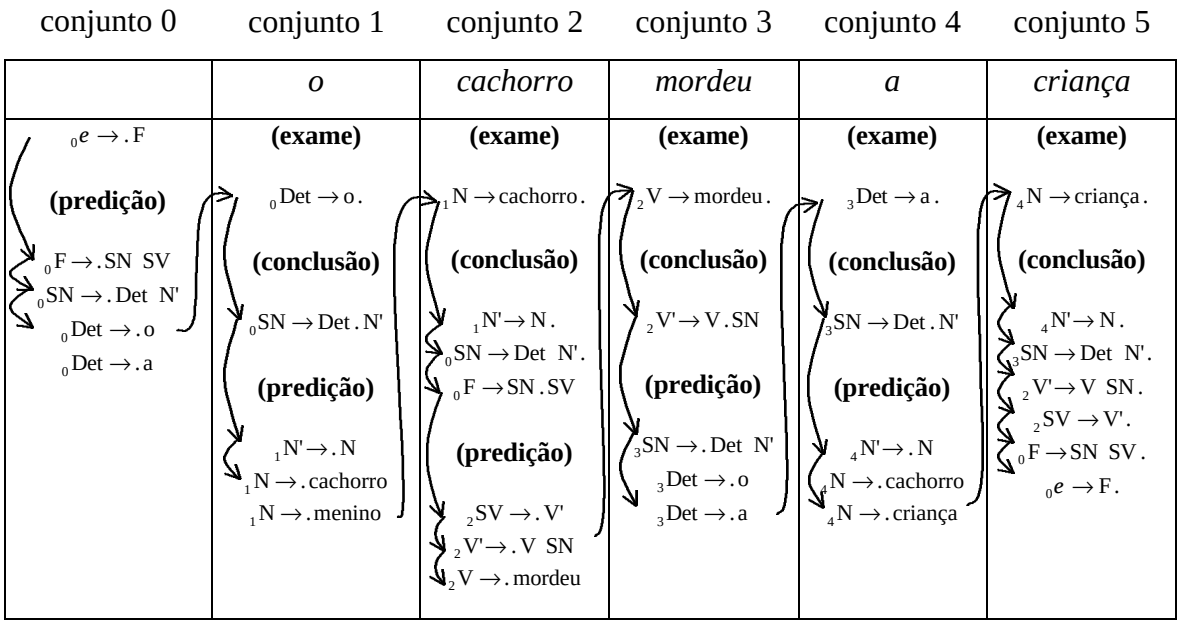

Tabela 4-2: Ligação entre os estados no quadro Earley.

4.3.2 Construção do Analisador Usando Programação Orientada a Objetos

A estrutura do analisador Earley tem muita semelhança com redes de estados finita: cada derivação construída pelo analisador Earley corresponde a uma seqüência de estados *ligados* por predição, exame e conclusão. Dentre as implementações eficientes para redes de estados finitas em linguagem C++, temos aquelas que utilizam programação orientada a objetos. Seguindo a mesma linha, criamos as classes **Estado**, **Conjunto** e **Quadro** que servem de base para a implementação do analisador.

A classe Estado permite a criação dinâmica de estados Earley que serão gerenciados pelos objetos da classe Conjunto. A seqüência de conjuntos é controlada por um objeto da classe Quadro.

A utilização do analisador exige apenas o instanciamento de um objeto Quadro e a execução dos métodos da classe para: definição da seqüência de palavras (frase completa ou parcial), análise da frase, construção da árvore de derivação, etc.

A programação orientada a objetos simplifica bastante a implementação do analisador Earley e a alocação dinâmica dos objetos permite grande flexibilidade durante a análise, pois não existe qualquer limitação *definida* para o comprimento da frase ou número de regras da gramática, exceto a própria limitação de memória da máquina usada.

O analisador implementado permite ainda que iniciemos a análise de parte de uma frase e, em seguida, adicionemos as palavras restantes para a conclusão da análise. Isso permite que utilizemos o Modelo da Língua como preditor de palavras durante o algoritmo de busca pela seqüência de palavras reconhecida.

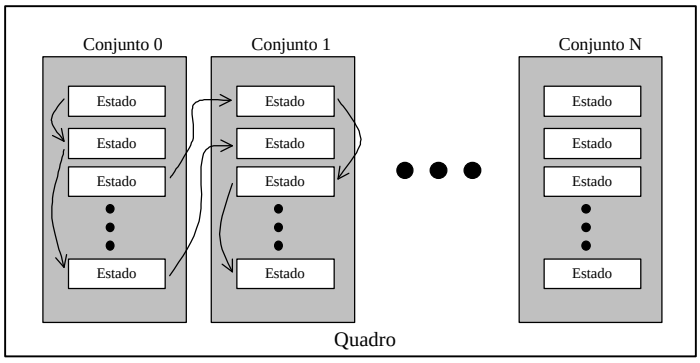

Figura 4-42: Estrutura de estados, conjuntos e quadro usada no analisador.

5 O Algoritmo de Busca

5.1 Introdução

Uma vez desenvolvidos o modelo acústico e o Modelo da Língua, resta definir um procedimento que aplique os dois modelos e encontre a seqüência de palavras mais provável, $\hat{W}$, que corresponderá à elocução de entrada.

Idealmente, deveríamos postular cada seqüência de palavras $W = w_1 w_2 ... w_N$ possível e tomar a frase reconhecida $\hat{W}$ como a seqüência W que maximiza o produto $P(O|W).P(W)$. Tal estratégia só pode ser adotada se o número de seqüências possíveis é suficientemente pequena, pois, de outra forma, o espaço de busca torna-se demasiadamente grande para uma busca exaustiva.

Como em todo problema de busca, podemos adotar duas estratégias principais: *depth-first* ou *breadth-first* [Young96] [Rich*94].

Na estratégia *depth-first*, as hipóteses mais promissoras são seguidas até o final da elocução ser atingido. Como exemplo, temos o algoritmo *Stack* [Bahl*83] e o algoritmo A* [Rich*94].

Na estratégia *breadth-first*, as hipóteses são tratadas em paralelo. Algoritmos de decodificação usando *breadth-first* exploram o princípio de otimalidade de Bellman e são normalmente chamados de decodificadores de Viterbi.

A estratégia de busca pode ser ou não integrada [Ortmanns*97] (explorar todas as fontes de conhecimento ao mesmo tempo durante a busca). No caso da busca não-integrada, podemos optar por considerar o Modelo da Língua somente numa etapa de pós-processamento. Podemos estimar o

valor de $P(W)$ e utilizá-lo para reclassificar o conjunto das n-melhores seqüências de palavras que maximizam $P(O|W)$. Trata-se, entretanto, de uma simplificação da busca que pode levar facilmente a uma seqüência de palavras incorreta.

Optamos por desenvolver uma busca integrada, realizada através de um algoritmo baseado no algoritmo *Level Building* (LB) [Rabiner*85] que realiza a decodificação usando os princípios de Programação Dinâmica (PD) também baseados no princípio de otimalidade de Bellman [LeeCH*89]: "um conjunto ótimo de soluções tem a propriedade de que qualquer que seja a primeira decisão, as decisões subseqüentes devem ser ótimas com relação ao resultado da primeira".

Outro tipo de algoritmo que propõe busca integrada pode ser encontrado em [Ortmanns*97].

Para entendermos o funcionamento do algoritmo Level Building, entendamos primeiro o que faz o algoritmo de Viterbi.

Sabemos que o modelo da língua permite que avaliemos o termo $P(O|W)$. Na prática, entretanto, realiza-se uma aproximação, assumindo que $P(O|W) \cong P(O,\hat{Q}|W)$, onde $\hat{Q} = \hat{q}_1\hat{q}_2...\hat{q}_T$ é a seqüência ótima de estados dos modelos correspondentes a seqüência de palavras $W = w_1 w_2 ... w_N$. De fato, teremos que $P(O,\hat{Q}|W) = \max_Q P(O,Q|W)$. Foi visto que tal aproximação não prejudica o desempenho do sistema e ainda proporciona grande redução do tempo de reconhecimento [LeeKF89]. O algoritmo de Viterbi é um método que calcula eficientemente o valor de $P(O,\hat{Q}|W)$, além de recuperar a seqüência de estados ótima $\hat{Q} = \hat{q}_1\hat{q}_2...\hat{q}_T$.

Resta-nos agora saber como combinar as palavras de forma eficiente visando maximizar o produto $P(O|W).P(W)$. A consideração de todas as seqüências possíveis de palavras é inviável quando mesmo possuímos um pequeno vocabulário. O algoritmo LB representa uma forma de maximização que contorna a busca exaustiva pela seqüência ótima de palavras.

Inicialmente, executa-se o algoritmo de Viterbi e determina-se os caminhos ótimos que chegam ao último estado de cada palavra e para cada instante de tempo, conforme ilustra a Figura 5-1. Este passo corresponde à busca pela palavra que ocupa a primeira posição da frase reconhecida (primeiro nível). Observe que os caminhos são encontrados para toda a elocução.

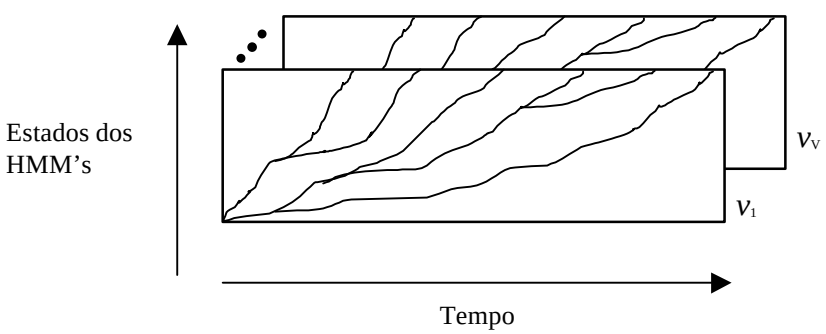

Figura 5-1: Execução do LB para palavras na primeira posição da frase (primeiro nível)

Passando ao segundo nível (busca pela palavra da segunda posição na frase), verifica-se que não precisamos de todas as informações obtidas no passo anterior, mas somente dos maiores valores de $P(o_1 o_2 ... o_t, q_1 q_2 ... q_t \mid w_1)$ dentre todas as palavras testadas, para cada instante de tempo. Isto porque no segundo nível desejamos encontrar também os caminhos ótimos e, para que o caminho seja ótimo de forma "global" termos que partir dos "máximos" encontrados no nível anterior, já que estes também correspondem a caminhos ótimos. Neste caso, efetua-se uma "redução de nível" que consiste em manter somente os maiores valores de $P(o_1 o_2 ... o_t, q_1 q_2 ... q_t \mid w_1)$, comparando-se todas as palavras testadas, conforme ilustra a figura a seguir.

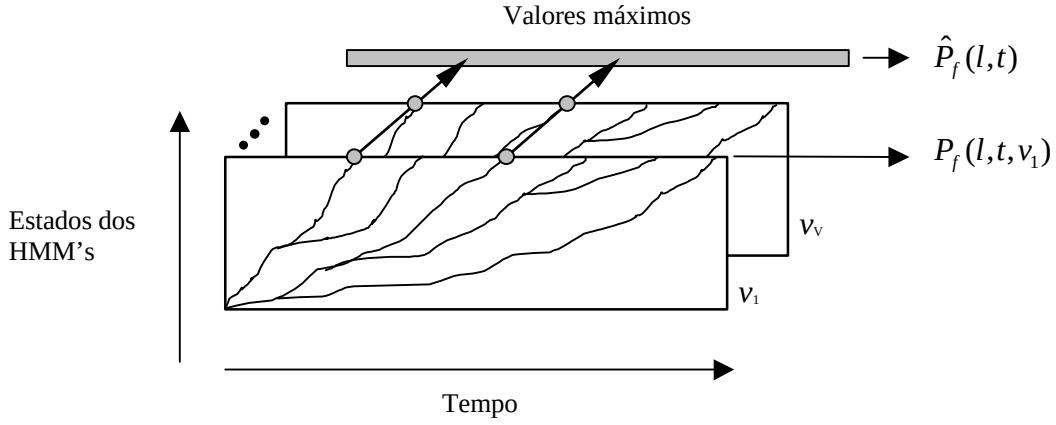

Figura 5-2: Processo de redução de nível no LB.

Quando abordarmos o algoritmo e suas expressões, verificaremos que os valores $P(o_1 o_2 ... o_t, q_1 q_2 ... q_t \mid w_1)$ de cada palavra são armazenados na matriz $P_f(l,t,w)$, onde l indica o nível, t indica o tempo e w, o índice da palavra. Os valores máximos obtidos na redução de nível

seriam armazenados numa matriz do tipo $\hat{P}_f(l,t)$, mas como não executaremos realmente uma redução de nível, no lugar desta matriz, utilizaremos a matriz $\hat{P}_f(l,t,m)$ onde m indica o *m*-ésimo melhor candidato dentre as V palavras testadas. A palavra vencedora em cada tempo t também fica armazenada numa matriz $\hat{W}(l,t)$, de maneira a permitir a posterior recuperação de toda a seqüência de palavras.

No segundo nível, os caminhos de Viterbi partem dos pontos finais do primeiro nível, utilizando como valores iniciais, aqueles armazenados em $\hat{P}_f(l,t)$, conforme ilustra a figura seguinte.

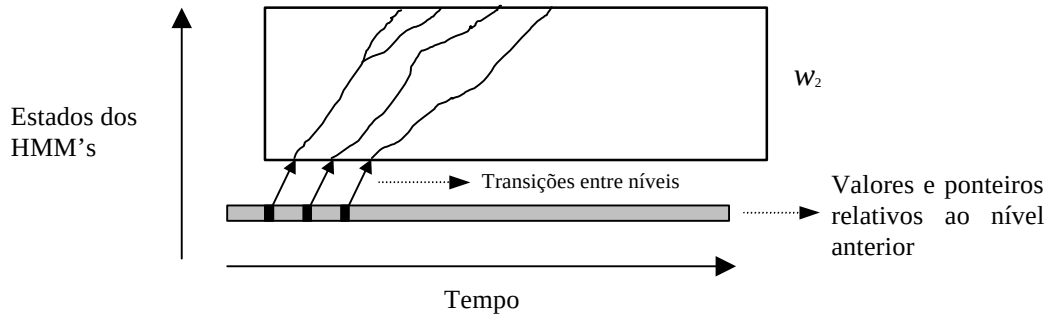

Figura 5-3: Continuação no segundo nível.

Os modelos da língua desenvolvidos atuarão exatamente nas transições entre um nível e o nível seguinte (vide Figura 5-3), pois este é o ponto que permite restringir a combinação de palavras durante o processo de busca.

O algoritmo avança até que se atinja um número máximo de níveis que corresponde ao máximo número de palavras permitido numa frase. Analisando os valores armazenados em $\hat{P}_f(l,t)$, obtem-se a seqüência de palavras reconhecida a partir das demais matrizes e ponteiros de controle usados ao longo do algoritmo.

Para conhecer o funcionamento do algoritmo Level Building em detalhes, sugerimos consultar [Rabiner*85] e [Rabiner*93]).

5.2 Algoritmo de Busca Integrada

Supondo um vocabulário de três palavras { a, b, c } e adotando um modelo bigram de palavras, podemos explicar o funcionamento do algoritmo proposto através da estrutura mostrada na Figura 5-4, na qual as linhas cheias correspondem aos modelos de Markov das palavras, enquanto as linhas tracejadas, que não foram totalmente mostradas por questão de clareza, correspondem às transições entre palavras. As probabilidades $P(v_j \mid v_i)$ correspondem às probabilidades fornecidas pelo Modelo da Língua, onde v_i e v_j são duas palavras do vocabulário.

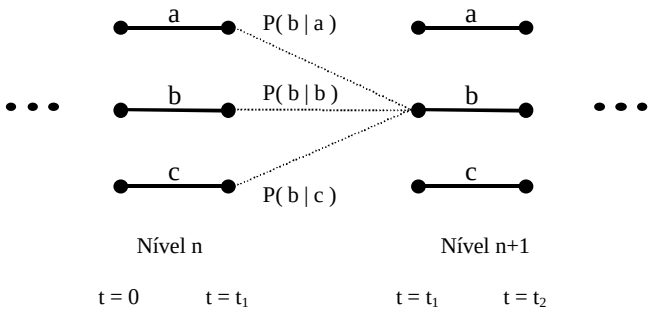

Figura 5-4: Estrutura de busca da seqüência de palavras reconhecida

Observe que a palavra b (nível $n+1$) pode ser iniciada a partir do último estado do HMM de uma das três palavras anteriores (nível n).

No algoritmo Level Building, antes de passar ao nível $n+1$, realizamos uma *redução de nível*, que corresponde a manter a palavra com maior verossimilhança final no tempo t_1 (valor estimado de $P(o_1...o_{t_1} \mid w_1...w_n)$), enquanto as outras palavras são descartadas. Neste caso, ainda não é possível considerar as probabilidades de transição de palavra, pois não conhecemos qual será a palavra seguinte.

Mantendo esta estratégia, estaremos prejudicando demasiadamente a atuação do Modelo da Língua. Para demonstrar este fato, considere o seguinte caso hipotético no qual atribuímos alguns valores numéricos às verossimilhanças finais de cada palavra do diagrama da Figura 5-4.

Suponha que a frase correspondente à elocução seja “ca” e que o reconhecimento será realizado com Level Building de dois níveis. Suponha, ainda, que as palavras “b” e “c” sejam muito parecidas acusticamente e, neste caso, adotamos os seguintes valores de verossimilhança final para

as hipóteses do nível 1 no tempo t_1 : 0,4 para a palavra “a”, 0,9 para “b” e 0,85 para “c”. A Figura 5-5 representa a situação que acabamos de descrever.

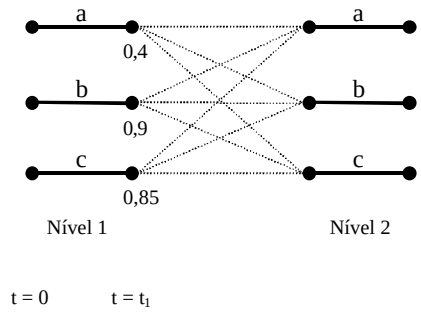

Figura 5-5: Exemplo de busca usando Level Building com dois níveis.

Aplicando a redução de nível, a palavra vencedora será a palavra “b”, visto que a hipótese correspondente possui a maior verossimilhança final (igual a 0,9). Neste ponto, um erro foi cometido, pois a primeira palavra deveria ser “c” (por hipótese, assumimos a frase “ca”).

A partir da primeira palavra (“b”), consideramos as demais hipóteses, conforme mostra a Figura 5-6. Observe que somente agora poderemos considerar as probabilidades referentes ao Modelo da Língua, pois sabemos a palavra anterior ("b") e as possíveis palavras seguintes ("a", "b" e "c").

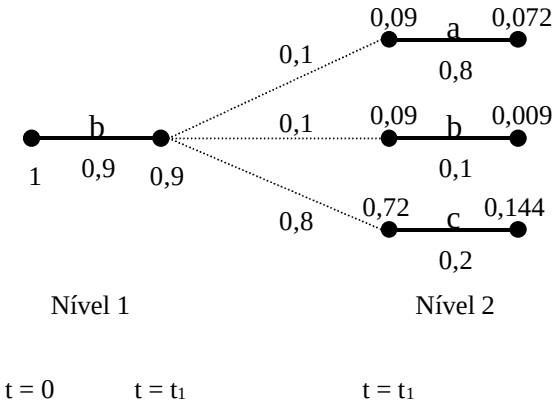

Figura 5-6: Passagem para o próximo nível da busca.

Devido ao elevado valor de $P(c \mid b)$ (os valores são arbitrários), em comparação com os valores de $P(a \mid b)$ e $P(b \mid b)$, é possível que a próxima palavra reconhecida seja a palavra “c”

(lembre-se de que a palavra correta deveria ser "a"), e então reconheceríamos a frase “bc”, ao invés da frase correta “ca”.

Esta seqüência de erros poderia ter sido evitada, se não tivéssemos realizado a redução de nível no nível 1, mantendo todas as palavras e “ligando” a segunda palavra à palavra do nível 1 que proporciona maior probabilidade (já considerando as probabilidades condicionais do Modelo da Língua).

Neste caso, a palavra “a” do segundo nível seria ligada à palavra “c” do primeiro nível, considerando todas as possibilidades: seqüência “aa” com probabilidade acumulada de 0,4 x 0,3 ou “ba” com 0,9 x 0,1 ou “ca” com 0,85 x 0,8 = 0,68 (observe a figura seguinte).

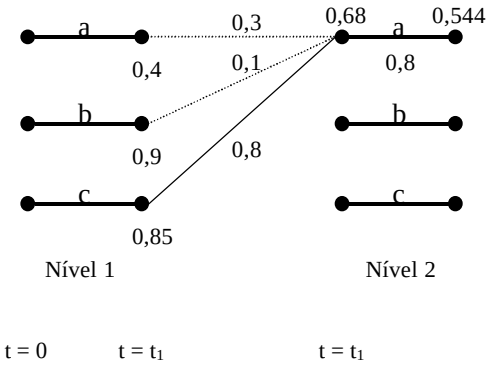

Figura 5-7: Estrutura de busca pela seqüência de palavras reconhecida

Apesar do exemplo anterior ser um caso particular, pode ser facilmente percebido que a aplicação do algoritmo LB padrão, quando se utilizam modelos de língua m-gram, resulta num procedimento de decisão (entre palavras) que está fora dos princípios da Programação Dinâmica, já que a probabilidade de transição entre duas palavras, $P(v_j \mid v_i)$, não é considerada durante a redução de nível. A redução de nível pode levar facilmente à eliminação de hipóteses promissoras e, até mesmo, da seqüência correta de palavras (uma discussão sobre este problema pode ser encontrada em [Ortmanns*97]).

A redução de nível traz, entretanto, a vantagem de reduzir a quantidade de informação a ser armazenada e também acelera o processo de decodificação, visto que diversas hipóteses incorretas são abandonadas. De fato, mesmo para um vocabulário de algumas centenas de palavras, o processo de busca se tornaria inviável se mantivéssemos todas as hipóteses (palavras) de níveis anteriores, como sugere a Figura 5-7.

O algoritmo proposto neste trabalho é uma modificação do algoritmo Level Building. Em vez da redução de nível tradicional, usaremos uma técnica denominada poda de histograma [Ortmanns*97] (*histogram pruning*) na qual somente as M hipóteses com maior verossimilhança final são mantidas e as demais são eliminadas. Assim, temos uma redução de nível que gera múltiplos candidatos, na qual reduzimos a quantidade de informação armazenada, mas também nos aproximamos do procedimento de busca ideal.

Somente para ilustrar, tomemos o exemplo anterior e consideremos uma poda de histograma mantendo dois candidatos.

Através da Figura 5-8, observa-se que o "caminho" que leva à seqüência correta "ca" permanece possível. Neste caso, o resultado final dependerá também do resultado da busca dentro de cada modelo de Markov de palavra no segundo nível (resultado do algoritmo de Viterbi).

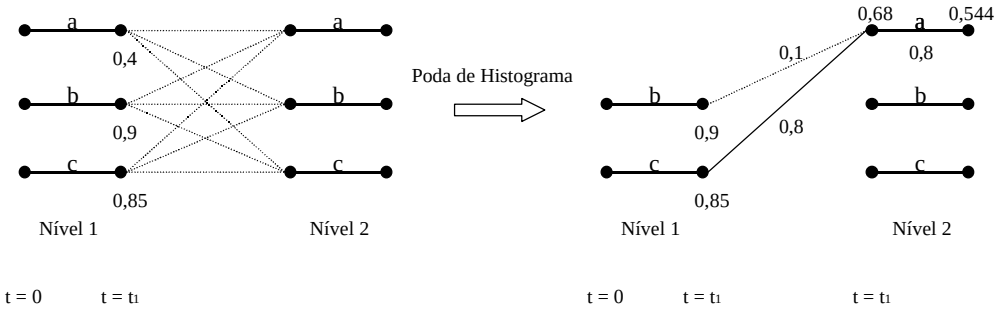

Figura 5-8: Estrutura de busca pela seqüência de palavras reconhecida

5.3 Implementação do Algoritmo

A probabilidade de ocorrência da seqüência de palavras $w_1...w_N$, considerando um Modelo da Língua bigram de palavras, pode ser escrita como (5-1), considerando também o marcador de início e final de frase \$ (por simplificação, adotamos sempre $P(\$) = 1$).

$$P(\$, w_1, w_2, ..., w_N, \$) = P(\$ \mid w_N).P(w_N \mid w_{N-1})...P(w_2 \mid w_1).P(w_1 \mid \$) \qquad (5\text{-}1)$$

Como foi mencionado anteriormente, os termos $P(w_n \mid w_{n-1})$ devem ser introduzidos ao longo do processo de decodificação.

Seguindo o mesmo esquema de apresentação do algoritmo Level Building adotado em [Rabiner*85], apresentamos a seguir o algoritmo de busca integrada proposto neste trabalho.

Definições:

Seja o_t, o vetor de parâmetros acústicos no instante t, os termos a_{ij}^w e $b_j^w(o_t)$ são, respectivamente, a probabilidade de transição do estado i para o estado j e a probabilidade de emissão do vetor o_t no estado j, correspondentes ao modelo de Markov da palavra w. Assim, definimos os seguintes termos:

$$\begin{aligned}
&\overline{a}_{ij}^{\,w} = \log_{10}\left(a_{ij}^{\,w}\right) \\
&\overline{b}_j^{\,w}(o_t) = \log_{10}\left[b_j^w(o_t)\right] \\
&\overline{P}(w_n \mid w_{n-1}) = \log_{10}\left[P(w_n \mid w_{n-1})\right] \\
&\overline{P}(w \mid \$) = \log_{10}\left[P(w \mid \$)\right] \\
&\overline{P}(\$ \mid w) = \log_{10}\left[P(\$ \mid w)\right]
\end{aligned} \qquad (5\text{-}2)$$

Os estados dos modelos das palavras estão numerados desde $j = 1$ até $j = Ne$, onde Ne^w corresponde ao número de estados no modelo da palavra w.

A seqüência de vetores de parâmetros acústicos $O = o_1 ... o_T$ corresponde ao intervalo $1 \le t \le T$ (elocução completa), onde T é o número de vetores de parâmetros acústicos (em nosso caso, vetores de parâmetros mel-cepstrais).

A variável $d_t(j)$ corresponderá à log-probabilidade conjunta da seqüência parcial de estados e da seqüência observada de vetores acústicos, $d_t(j) = \log_{10} P(o_1 ... o_t, q_1 q_2 ... q_{t-1} j)$, onde j é o estado do HMM da palavra w no tempo t.

Algoritmo:

Nível $l = 1$

Executar os procedimentos a seguir para cada palavra w do vocabulário

1 - Inicialização

Inicializamos a variável $d_t(j)$ com a probabilidade da palavra começar a frase usando a equação:

$$d_0(0) = \overline{P}(w \mid \$) \tag{5-3}$$

O tempo $t = 0$ é usado como inicialização durante a busca. Os demais valores de $d_t(j)$ são definidos como:

$$\begin{aligned} d_0(j) &= -\infty, \quad j = 1,2,3,4...Ne^w \\ d_0(0) &= -\infty, \quad t = 1,2,3,...T \end{aligned} \tag{5-4}$$

Na prática, o símbolo ∞ indica que um valor *alto* deve ser definido.

2 – Recursão (Viterbi)

Fazendo $1 \le t \le T$ e $1 \le j \le Ne^w$, calculamos:

$$d_t(j) = \max_{0 \le i \le j} \left\{ d_{t-1}(i) + \overline{a}_{ij}^w \right\} + \overline{b}_j^w(o_t) \tag{5-5}$$

3 - Finalização

Calculamos a log-probabilidade final de cada palavra para $1 \le t \le T-1$:

$$\begin{aligned} P_f(l,t,w) &= d_t(Ne^w) + \overline{a}_{ij}^w + \overline{P}_{dur}^w \\ B(l,t,w) &= 0 \end{aligned} \tag{5-6}$$

A matriz $P_f(l,t,w)$ armazena a log-probabilidade do último estado relativa a cada palavra w, para cada nível l e tempo t. O termo $\overline{P}_{dur}^w = \log_{10} P_{dur}^w$ corresponde a log-probabilidade relativa

ao modelo de duração da palavra w (vide seção 2.2). Observe que P_{dur}^{w} é equivalente ao termo $f_w(d)$ calculado na equação (2-4), sendo que o valor de d, duração da palavra, pode ser obtido a partir dos ponteiros de retorno, realizando um *backtracking* (vide [Morais97]).

A matriz auxiliar $B(l,t,w)$ armazena os ponteiros de retorno que permitem encontrar a seqüência ótima de palavras.

Calculamos também os termos referentes ao último instante de tempo ($t = T$), considerando a log-probabilidade da palavra *w* finalizar a frase $\overline{P}(\$ \mid w)$:

$$\begin{aligned} P_f(l,T,w) &= d_T(Ne^w) + \overline{a}_{ij}^{w} + \overline{P}_{dur}^{w} + \overline{P}(\$ \mid w) \\ B(l,T,w) &= 0 \end{aligned} \tag{5-7}$$

4 – Poda de histograma

No final do nível $l = 1$, mantemos somente as M palavras com maior probabilidade acumulada final, descartando todas as demais palavras, bem como suas informações armazenadas.

Assim, para $1 \le t \le T$ e $1 \le m \le M$, fazemos:

$$\begin{aligned} \hat{P}_f(l,t,m) &= \underset{w}{\text{Mmax}}\left(P_f(l,t,w), m\right) \\ \hat{B}(l,t,m) &= 0 \\ \hat{W}(l,t,m) &= \underset{w}{\text{argMmax}}\left(P_f(l,t,w), m\right) \\ \hat{N}(l,t,m) &= 0 \end{aligned} \tag{5-8}$$

A função $\underset{w}{\text{Mmax}}\left(P_f(l,t,w), m\right)$ retorna o *m*-ésimo maior valor assumido por $P_f(l,t,w)$ variando-se o índice de palavra w, para determinados valores de l e t. A função $\underset{w}{\text{argMmax}}\left(P_f(l,t,w), m\right)$ retorna o índice w da palavra selecionada por $\underset{w}{\text{Mmax}}\left(P_f(l,t,w), m\right)$.

Na prática, portanto, a matriz $\hat{P}_f(l,t,m)$ armazena os M maiores valores de $P_f(l,t,w)$ relativos às M palavras vencedoras da poda de histograma. A matriz $\hat{B}(l,t,m)$ armazena os ponteiros de retorno relacionados e a matriz $\hat{W}(l,t,m)$ armazena o índice dessas M palavras

vencedoras. O índice m é usado para identificar as palavras vencedoras na poda de histograma e varia desde 1 (melhor candidato) até M .

Níveis $l > 1$

Devemos executar os procedimentos seguintes para todas as palavras.

1 – Inicialização

Devemos realizar os seguintes procedimentos:

$$\begin{aligned} &d_0(j) = -\infty, \qquad 0 \le j \le Ne^w \\ &d_t(0) = \max_m \left(P_f(l-1,t,m) + \overline{P}(w \mid \hat{W}(l-1,t,m)) \right), \quad 1 \le t \le T \\ &N(0,t) = \arg\max_m \left(P_f(l-1,t,m) + \overline{P}(w \mid \hat{W}(l-1,t,m)) \right), \quad 1 \le t \le T \\ &a_t(0) = t, \quad 0 \le t \le T \end{aligned} \tag{5-9}$$

2 – Recursão

O procedimento de recursão anterior (5-5) é repetido aqui, acrescentando-se a atualização dos ponteiros de retorno $a_t(j)$ e $N(j,t)$:

$$\begin{aligned} d_t(j) &= \max_{0 \le i \le j} \left\{ d_{t-1}(i) + \overline{a}_{ij}^w \right\} + \overline{b}_j^w(o_t) \\ a_t(j) &= a_{t-1}\left(\arg\max_{0 \le i \le j} \left\{ d_{t-1}(i) + \overline{a}_{ij}^w \right\} \right) \\ N(j,t) &= N\left(\arg\max_{0 \le i \le j} \left\{ d_{t-1}(i) + \overline{a}_{ij}^w \right\}, t-1 \right) \end{aligned} \tag{5-10}$$

3 – Finalização

No final do nível, os ponteiros devem ser atualizados da mesma forma que para o primeiro nível, com exceção de $B(l,t,w)$.

$$
\begin{aligned}
&P_f(l,t,w) = d_t(Ne^w) + \overline{a}_{ij}^{\,w} + \overline{P}_{dur}^{\,w}, \quad 1 \le t \le T-1 \\
&B(l,t,w) = a_t(Ne^w), \quad 1 \le t \le T \\
&N_f(t,w) = N(Ne^w,t), \quad 1 \le t \le T
\end{aligned}
\tag{5-11}
$$

Para $t = T$, fazemos:

$$
P_f(l,T,w) = d_T(Ne^w) + \overline{a}_{ij}^{\,w} + \overline{P}_{dur}^{\,w} + \overline{P}(\$ \mid w) \tag{5-12}
$$

4 – Poda de histograma

Devemos repetir a poda de histograma realizada para o nível zero. Assim, para $1 \le t \le T$ e $1 \le m \le M$, fazemos:

$$
\begin{aligned}
&\hat{P}_f(l,t,m) = \underset{w}{\mathrm{Mmax}}\left(P_f(l,t,w), m\right) \\
&\hat{B}(l,t,m) = B\left(l,t,\underset{w}{\mathrm{argMmax}}\left(P_f(l,t,w), m\right)\right) \\
&\hat{W}(l,t,m) = \underset{w}{\mathrm{argMmax}}\left(P_f(l,t,w), m\right) \\
&\hat{N}(l,t,m) = N_f\left(t, \underset{w}{\mathrm{argMmax}}\left(P_f(l,t,w), m\right)\right)
\end{aligned}
\tag{5-13}
$$

No fim da execução do algoritmo, a frase reconhecida pode ser obtida através dos ponteiros de retorno $\hat{W}(l,t,m)$, $\hat{B}(l,t,m)$ e $\hat{N}(l,t,m)$ (vide [Morais97] e [Rabiner*85]).

5.3.1 Utilização do Algoritmo com Modelo Bigram de Classes de Palavras

Para utilizar o modelo bigram de classes de palavras, deveremos estimar as probabilidades condicionais $P(w_n \mid w_{n-1})$ a partir das probabilidades condicionais de classe de palavra, segundo as equações apresentadas a seguir.

$$
P(w_n \mid w_{n-1}) = \sum_{\forall g_n} \sum_{\forall g_{n-1}} P(w_n \mid g_n).P(g_n \mid g_{n-1}).P(g_{n-1} \mid w_{n-1}) \tag{5-14}
$$

$$P(w_1 \mid \$) = \sum_{\forall g_1} P(w_1 \mid g_1).P(g_1 \mid \$) \tag{5-15}$$

$$P(\$ \mid w_N) = \sum_{\forall g_N} P(\$ \mid g_N).P(g_N \mid w_N) \tag{5-16}$$

O termo $P(w_n \mid g_n)$ foi estimado considerando todas as palavras pertencentes à classe c_m como equiprováveis, conforme estabelece (5-17).

$$P(w_n \mid g_n) \cong \frac{1}{\text{número de palavras na classe } g_n} \tag{5-17}$$

Poderíamos também ter estimado esta probabilidade a partir da ocorrência das palavras da classe g_n nas frases de treinamento, conforme a equação (5-18), mas não o fizemos em função do tamanho reduzido da base de treinamento.

$$P(w_n \mid g_n) \cong \frac{N(w_n, g_n)}{N(g_n)} = \frac{N(w_n \in g_n)}{N(g_n)} \tag{5-18}$$

Da mesma forma, o termo $P(g_n \mid w_n)$ foi estimado considerando equiprováveis todas as classes atribuídas à palavra w_n, conforme (5-19), mas também poderíamos ter usado estimativas sobre as frases de treinamento segundo (5-20).

$$P(g_n \mid w_n) \cong \frac{1}{\text{número de classes da palavra } w_n} \tag{5-19}$$

$$P(g_n \mid w_n) \cong \frac{N(w_n \in g_n)}{N(w_n)} \tag{5-20}$$

As estimativas (5-18) e (5-20) possuem a desvantagem de serem função direta das palavras do vocabulário, o que implica em possuirmos um número maior de frases nas quais apareçam estas palavras. Assim, além de armazenar a matriz correspondente a $P(g_n \mid g_{n-1})$, teríamos de armazenar os valores estimados de $P(w_n \mid g_n)$ e $P(g_n \mid w_n)$. Adicionalmente, para a inclusão de uma nova palavra, teríamos de reestimar os dois últimos termos sobre uma nova base de dados que incluísse frases contendo a nova palavra.

Utilizando (5-17) e (5-19), estamos fazendo uma grande aproximação, mas facilitamos o processo de estimação, o qual pode ser realizado sobre um conjunto de frases construídas a partir de um vocabulário de palavras qualquer e não obrigatoriamente igual ao empregado pelo sistema de reconhecimento. Podemos verificar a utilização deste tipo de aproximação em [LeeKF89].

5.3.2 Utilização do Algoritmo com Modelo Baseado em Gramática

Para utilizar o Modelo da Língua baseado em gramática independente de contexto, definiremos uma função $S(h_n, w_n)$, onde $h_n = w_1,...,w_{n-1}$ é a seqüência anterior já decodificada e w_n é a palavra seguinte. A função $S(h_n, w_n)$ assume o valor um, se a seqüência parcial $w_1...w_n$ é reconhecida pela gramática (utilizando o analisador), e assume o valor zero, se a seqüência parcial $w_1...w_n$ não é reconhecida pela gramática.

O Modelo da Língua baseado em GIC atuará como preditor da palavra seguinte à seqüência $h_n = w_1,...,w_{n-1}$, como já foi explicado na seção 2.4.2. Assim, temos apenas que limitar a aplicação do algoritmo de busca às palavras w_n que fazem $S(h_n, w_n) = 1$ (lembre-se de que $h_n = w_1,...,w_{n-1}$ é obtido através dos ponteiros de retorno $\hat{W}(m,l,t)$, $\hat{B}(m,l,t)$ e $\hat{N}(m,l,t)$ (vide [Morais97] e [Rabiner*85]).

A Figura 5-9 ilustra uma busca já realizada nos níveis 1 e 2 e sendo iniciada no nível 3. No final do nível 2 no tempo t, a seqüência anterior de palavras é "o cachorro" e o Modelo da Língua baseado em GIC fornece uma lista de palavras possíveis ("mordeu", "pegou", "bonito", "de", "com", etc.). Neste caso, as palavras que não se encontram na lista não podem expandir o caminho indicado na figura a partir do nível 3.

Apesar do procedimento de predição de palavras reduzir o espaço de busca no reconhecimento, pode-se observar que o custo computacional do processo permanece muito alto, pois a análise de seqüências de palavras é realizada para cada instante de tempo ($1 \le t \le T$) e para todos os níveis ($1 \le l \le L$, onde L é o número máximo de níveis do algoritmo de busca).

De fato, chegamos à conclusão que o algoritmo de busca Level Building não é muito apropriado para sistemas de reconhecimento de fala contínua que utilizem modelos da língua baseados em GIC, se o objetivo é uma aplicação em tempo real. No sentido de obter algoritmos mais adequados, sugerimos consultar [Ortmanns*97].

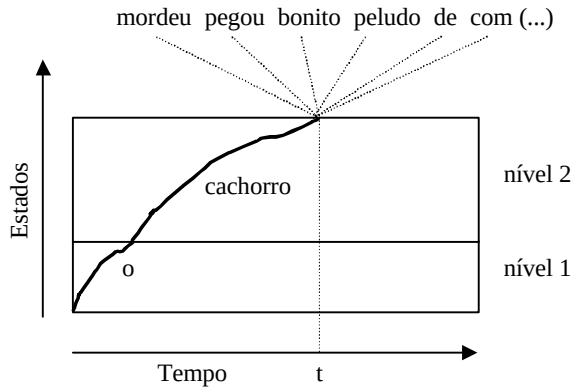

Figura 5-9: Procedimento de busca usando predição de palavra.

Para se ter uma idéia do problema, o reconhecimento de uma frase com duração de 3 s leva cerca de 2 min usando modelo bigram (o tempo de reconhecimento quando não usamos modelos da língua é praticamente o mesmo, pois o custo computacional introduzido por modelo bigram é mínimo) e leva cerca de 10 min usando o modelo baseado em GIC, numa máquina Pentium II 300Mhz. Deve-se lembrar que estes valores foram conseguidos sem a preocupação de otimizar os algoritmos usados, ou seja, o objetivo do nosso trabalho não é desenvolver uma aplicação em tempo real.

6 Resultados no Reconhecimento

6.1 Considerações Iniciais

Utilizando os modelos da língua desenvolvidos, avaliamos o desempenho do sistema de reconhecimento de fala contínua em função da taxa de erro de palavra, calculada através da fórmula $ErroPal = \frac{S + D + I}{N} \times 100$, onde N é o número total de palavras nas frases de teste e $S + D + I$ corresponde ao número total de erros de substituição (S), exclusão (D) e inserção (I) de palavras (vide [SHLT96]).

Foi utilizado um sistema de reconhecimento de fala contínua dependente do locutor, desenvolvido por Morais [Morais97], que emprega o modelo híbrido HMM/MLP. O sistema possui um vocabulário de 312 palavras.

Para os modelos bigram de classes de palavras, o conjunto de frases de teste é formado por 74 frases com um total de 499 palavras. Procurou-se selecionar um conjunto de frases de teste que estivesse de acordo com os critérios de seleção das frases usadas para o treinamento do Modelo da Língua, mas foram permitidas algumas frases do tipo “A cotação do dólar aumentou mas as bolsas fecharam em baixa”, na qual vemos a presença de duas orações ligadas por conjunção coordenativa. Também permitimos que frases como “Vinte e cinco reais” fossem utilizadas embora a quantidade de exemplos deste tipo no conjunto de treinamento não tenha sido significativa.

Para o Modelo da Língua baseado em gramática independente de contexto, utilizamos um conjunto de teste de 44 frases com um total de 306 palavras. Por limitações práticas, não pudemos

utilizar um número maior de frases. Além disso, foram excluídas as frases que não constituíam orações ("vinte e cinco reais", "curva perigosa", "saldo: vinte e cinco reais", etc.) ou possuíam uma estrutura diferente daquelas que foram discutidas no capítulo 4 (presença de aposto: "o convênio, um documento de trinta páginas, tem permitido o intercâmbio"). Deve-se salientar, entretanto, que a gramática desenvolvida “reconhece” um número infinito de frases e o fato de não utilizarmos um conjunto maior de frases de teste não demostra uma limitação da gramática, mas resulta de uma limitação prática (base de dados) que deverá ser superada no futuro.

6.2 Apresentação dos Resultados

Comparamos a utilização do modelo bigram de classes gramaticais com o modelo de duração de palavra (vide capítulo 2). Avaliamos o desempenho do reconhecimento nos seguintes casos: sem utilizar Modelo da Língua ou modelo de duração de palavra, utilizando modelo de duração de palavra (MDUR), utilizando somente Modelo da Língua (ML) e, finamente, utilizando o Modelo da Língua e o modelo de duração de palavra (ML+MDUR).

Modelo Usado	Erro de Palavra (%)
<nenhum>	56,5
MDUR	24,8
ML	22,2
ML+MDUR	19,2

Tabela 6-1: Reconhecimento usando ML baseado em classes gramaticais.

Alguns exemplos de frases reconhecidas também ajudam a identificar que tipos de problema a modelagem da língua pode solucionar. Na Tabela 6-2, apresentamos três frases reconhecidas pelo sistema: “o saldo é suficiente”, “o saldo sempre está disponível” e “as contas chegaram atrasadas”. Apresentamos os resultados obtidos no final do algoritmo de busca para os oito níveis do *Level Building* com maior verossimilhança acumulada final, organizados em ordem decrescente de verossimilhança, ou seja, a frase vencedora vem sempre em primeiro lugar. Na apresentação das frases , a “vírgula” e o “ponto” são usados para representar a presença de silêncio nas frases.

<**nenhum**>	**MDUR**
, o saldo o é suficiente , , o saldo o é a suficiente , , os a o o é a suficiente , , os a o do o é a suficiente , , os a o do o é a a suficiente , , os a o do o é a a suficiente e , , os a o do o é a a suficiente e e , , os a o do o é a a suficiente e e e , , o saldo cento e está das , nome e ao , , o saldo o cento e está das , nome e ao , , o saldo o cento e está das , nome e a o , , o saldo cento e está das , nome ao , , o saldo o cento e está e das , nome e a o , , o saldo cento e está da disponível ao , , o saldo o cento e está e das , nome e a a o , , o saldo cento e está da disponível , , **as contas chegaram atrasadas** , , as com do as chegaram atrasadas , , as com do das se e é a não atrasadas , , as com do os as chegaram atrasadas , , as com do as e é a não atrasadas , , as conta as chegaram atrasadas , , as com do o das se e é a não atrasadas , , as com do os a as chegaram atrasadas ,	, o saldo o é suficiente , , o saldo o o é suficiente , , o saldo o o é a suficiente , o saldo é suficiente , , o saldo o o é a suficiente e , , o os a o o o o é suficiente , , o os a o o o o o é suficiente , , o os a o o o o o é a suficiente , , **o saldo sempre está disponível** , , o saldo cento três está disponível , , o saldo cento três está da disponível , , o saldo o cento e e está disponível , , o saldo o cento e e está da disponível , , o saldo o cento e e está da disponível ao , , o saldo o cento e e está da disponível a o , , o saldo o cento e e está da disponível e a o , , **as contas chegaram atrasadas** , , a as contas chegaram atrasadas , , as com do as chegaram atrasadas , , as com da o as chegaram atrasadas , , a as com da o as chegaram atrasadas , , a as com da o as chegaram a atrasadas , , as com do as e de é a não atrasadas , , as com da o as e de é a não atrasadas ,
ML	**ML+MDUR**
, o saldo o é suficiente , , , o saldo o é suficiente , , **o saldo é suficiente** , , , o saldo o é suficiente , , , , o saldo o é suficiente , , , , , , , , o saldo o é suficiente , , , , , , o saldo o é suficiente , , , , , , , o saldo o é suficiente , , , , **o saldo sempre está disponível** , , **o saldo sempre está disponível** , , , **o saldo sempre está disponível** , , , , o saldo sempre está as , nome ao , , o saldo sempre está as , nome ao , , , o saldo sempre está as , nome ao , , , , o saldo sempre está as , nome ao , , , , , saldo sempre está disponível , , **as contas chegaram atrasadas** , , , **as contas chegaram atrasadas** , , , **as contas chegaram atrasadas** , , , , **as contas chegaram atrasadas** , , , , , **as contas chegaram atrasadas** , , , , , , **as contas chegaram atrasadas** , , , , , , , **as contas chegaram atrasadas** , , , , , , , , **as contas chegaram atrasadas** , , , , , , ,	, **o saldo é suficiente** , , o saldo o é suficiente , , o saldo o é suficiente , , , os plano real suficiente , , o saldo o é suficiente , , , , o saldo o é suficiente , , , , , o saldo o é suficiente , , , , , , o saldo o é suficiente , , , , , , , **o saldo sempre está disponível** , , **o saldo sempre está disponível** , , , saldo sempre está disponível , , o saldo sempre está disponível ao , , , o saldo sempre está as , nome ao , , o saldo sempre está as , nome ao , , , o saldo sempre está as , nome e a com , , o saldo sempre está as , nome e a com , um . **as contas chegaram atrasadas** . , , **as contas chegaram atrasadas** . , , **as contas chegaram atrasadas** , . , , **as contas chegaram atrasadas** , , , , , **as contas chegaram atrasadas** , , , , , , as contas chegaram a atrasadas , , , , . nacional aceitarão atrasadas . , , as contas chegaram a atrasadas , , , , ,

Tabela 6-2: Frases reconhecidas usando ML e MDUR.

De forma geral, o reconhecimento utilizando o Modelo da Língua produz frases que estão mais "próximas" das frases corretas, pois existem sobretudo menos inserções de palavras curtas ("do", "a", "os", etc.). Observe que sem usar Modelo da Língua ou de duração de palavra obtemos seqüências como "**os a o do o é a suficiente**", a qual não chega a ser tão distante, do ponto de vista

acústico, da frase correta: "**o saldo é suficiente**". Usando somente o modelo de duração de palavras, conseguimos melhores resultados no reconhecimento, mas ainda ocorrem o mesmo tipo de seqüências ("**o os a o o o o é suficiente**" ou "**o saldo o é suficiente**").

Usando o Modelo da Língua, conseguimos melhorar sensivelmente as frases obtidas, conforme se observa na Tabela 6-2.

Exceto pela análise anterior, todos os resultados no reconhecimento apresentados a seguir foram obtidos empregando o modelo de duração de palavra.

Durante a etapa de avaliação dos modelos bigram de classes com classificação automática, quando treinamos os modelos utilizando o conjunto de 211 frases, comparamos os resultados do reconhecimento empregando os modelos que utilizam os algoritmos de classificação SA e MC. Estes resultados preliminares, apresentados na figura seguinte, indicaram que as menores taxas de erro de palavra são obtidas quando usamos o algoritmo *Simulated Annealing*.

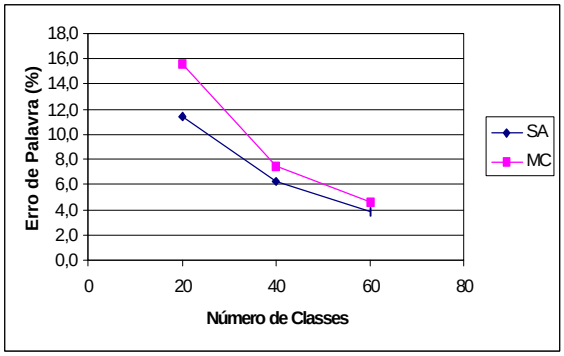

Figura 6-1: Erro de palavra vs. número de classes.

Resolvemos, então, adotar o algoritmo SA como algoritmo de classificação automática com base nos melhores resultados (perplexidade e erro de palavra) apresentados nos primeiros testes realizados. De fato, o algoritmo SA busca (pelo menos idealmente) o mínimo global de perplexidade, enquanto os algoritmos KM e MC garantem encontrar somente mínimos locais (vide seção 3.3).

Utilizando o modelo bigram de classes (classificação automática usando SA), treinado a partir do conjunto de 470 frases, obtivemos os seguintes resultados finais no reconhecimento:

Número de classes	Erro de palavra (%)	Perplexidade no treinamento	Comentário
	24,8		sem modelo da língua
20	16,8	97	
40	8,4	66	
60	7,0	47	
80	6,0	38	

Tabela 6-3: Reconhecimento usando modelo bigram de classes (classificação automática com Simulated Annealing).

A taxa de erro de palavra não decresce linearmente a medida que aumentamos o número de classes, conforme contatamos na Figura 6-2. De fato, a perplexidade final no treinamento exibe um comportamento semelhante ao comportamento da taxa de erro de palavra no reconhecimento.

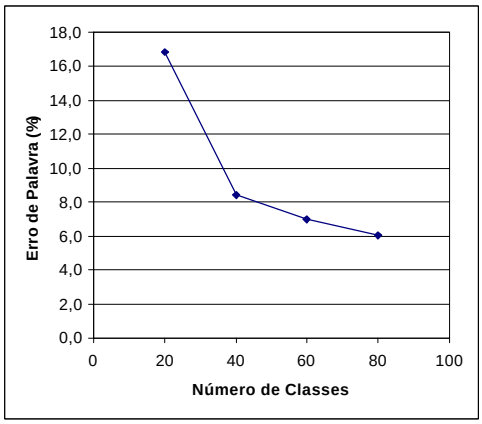

Figura 6-2: Erro de palavra vs. Número de classes.

Segundo [Jardino*93], quando utilizamos poucas classes, aumentamos o poder de generalização do Modelo da Língua, mas as probabilidades não são suficientemente *restritivas* para proporcionar bons resultados no reconhecimento. Quando o número de classes é grande demais, o modelo refletirá mais as características do texto de treinamento, permitindo pouca generalização. Isso explicaria porque a taxa de erro de palavra cai rapidamente quando passamos de 20 classes para 40 classes, mas varia pouco a medida que aumentamos mais o número de classes.

Aplicando somente o Modelo da Língua baseado em gramática independente de contexto (GIC), conseguimos uma taxa de erro de palavra de 12,7% durante o reconhecimento, usando o conjunto de teste de 44 frases. A taxa de erro de palavra conseguida sem utilizar qualquer Modelo da Língua foi de 21,9%.

Para possibilitar uma comparação com os outros modelos da língua implementados, resolvemos reavaliá-los usando o mesmo conjunto de teste (vide Tabela 6-4).

Modelo Usado	**Erro de Palavra (%)**
<nenhum>	21,9
20 classes (manual)	14,1
GIC	12,7
60 classes (SA)	6,5
60 classes (SA) + GIC	5,6

Tabela 6-4: Reconhecimento usando modelos bigram de classes e modelo baseado em GIC.

Utilizando o Modelo da Língua bigram de 20 classes (classificação manual) obtivemos uma taxa de erro de palavra de 14,1%. Uma taxa um pouco maior do que aquela usando GIC.

Aplicando o modelo bigram de 60 classes (classificação automática usando SA) obtivemos uma taxa de erro de 6,5%, quase metade da taxa de erro obtida quando usamos GIC.

Utilizando o modelo bigram de 60 classes (SA) em conjunto com o modelo baseado em GIC, obtivemos uma taxa de erro de palavra de 5,6%.

Apesar de utilizarmos um conjunto de teste reduzido (44 frases com 306 palavras), deve-se observar que a taxa de erro de palavra no reconhecimento diminuiu cerca de 42%, quando passamos do sistema que não utiliza modelos da língua para o sistema que utiliza o modelo baseado em GIC. Isso demonstra o potencial do modelo baseado em GIC, considerando que ele ainda constitui um Modelo da Língua bastante simplificado.

Na Tabela 6-5, temos resultados no reconhecimento similares àqueles mostrados na Tabela 6-2, mas usando o Modelo da Língua baseado em GIC e o modelo bigram de 60 classes (classificação automática usando SA) combinado ao modelo com GIC.

GIC	BIGRAM+GIC
, **o saldo é suficiente** ,	, **o saldo é suficiente** ,
, o saldo a o vôo é suficiente ,	, **o saldo é suficiente** , ,
, cotação é suficiente ,	, saldo é suficiente ,
, o saldo é suficiente , vôo	, **o saldo é suficiente** , , ,
, o saldo a o vôo é suficiente , vôo	, **o saldo é suficiente** , , , ,
, o saldo a o vôo é a suficiente , vôo	, **o saldo é suficiente** , , , , ,
, o saldo a o vôo é a a suficiente , vôo	, **o saldo é suficiente** , , , , , ,
, o saldo a o vôo é a a suficiente , , vôo	, **o saldo é suficiente** , , , , , , ,
, **o saldo sempre está disponível** ,	, **o saldo sempre está disponível** ,
, o saldo cento três está disponível ,	, **o saldo sempre está disponível** , ,
, saldo sempre está disponível ,	, **o saldo sempre está disponível** , , ,
, o saldo cento três está disponível , vôo	, saldo sempre está disponível ,
, o saldo cento três está da disponível , vôo	, **o saldo sempre está disponível** , , , ,
, o saldo cento três está disponível a o , vôo	, **o saldo sempre está disponível** , , , , ,
, o saldo cento três está as , nome ao , vôo	, **o saldo sempre está disponível** , , , , , ,
, o saldo cento três está as , nome a o , vôo	, **o saldo sempre está disponível** , , , , , , ,
, , **as contas chegaram atrasadas** ,	, , **as contas chegaram atrasadas** ,
, , as contas chegaram atrasadas , vôo	, , **as contas chegaram atrasadas** , ,
, , as contas chegaram a atrasadas , vôo	, , **as contas chegaram atrasadas** , , ,
, , as contas chegaram a a atrasadas , vôo	, , **as contas chegaram atrasadas** , , , ,
, , as contas chegaram a a atrasadas , , vôo	, , **as contas chegaram atrasadas** , , , , ,
, baixa contas chegaram atrasadas ,	, , **as contas chegaram atrasadas** , , , , , ,
, , as contas chegaram a a atrasadas , , , vôo	, baixa contas chegaram atrasadas ,
, , as contas chegaram a a atrasadas , , , , vôo	**, , as contas chegaram atrasadas , , , , , ,**

Tabela 6-5: Frases reconhecidas usando GIC e Bigram de 60 classes (SA).

No Apêndice B, apresentamos uma amostra das frases reconhecidas quando utilizamos os Modelos da Língua desenvolvidos neste trabalho.

7 Conclusão

7.1 Discussão Geral

Neste trabalho, foram desenvolvidos três modelos da língua para um sistema de fala contínua baseado em modelo híbrido HMM/MLP: um modelo bigram de classes gramaticais, um modelo bigram de classes com classificação automática utilizando o algoritmo *Simulated Annealing* (SA) e um modelo sintático baseado em gramática independente de contexto (GIC).

O modelo bigram de classes gramaticais possui suas classes definidas segundo a classificação gramatical de palavras adotada pela gramática tradicional. A classificação das palavras é feita manualmente, por isso, a preparação do texto de treinamento é custosa e exige bom conhecimento da língua. Este modelo possui a vantagem de que a inclusão de uma nova palavra no vocabulário implicaria somente na indicação das classes gramaticais a que ela pertence.

O modelo bigram de classes construído com classificação automática de palavras proporcionou melhores resultados no reconhecimento do que o modelo bigram de classes gramaticais. Utilizando 20 classes de palavras, obtivemos taxa de erro de palavra de 19,2% com classes gramaticais e 16,8% com classificação automática (conjunto de treinamento com 470 frases e 3665 palavras). Usando classificação automática com 40 classes, obtemos uma taxa de erro de palavra de 8,2% no reconhecimento.

Observe que enquanto a classificação manual utiliza somente informações morfológicas das palavras, a classificação automática tende a capturar outras relações entre as palavras (objetivando sempre minimizar a perplexidade avaliada sobre o texto de treinamento).

No modelo que usa classificação manual, o número de classes poderia ser aumentado, diferenciando as palavras quanto ao gênero, número ou mesmo características semânticas, embora isto não seja uma tarefa muito simples.

No caso da classificação automática, podemos variar facilmente o número de classes, buscando uma melhor relação entre o número de classes e a taxa de erro de palavra. Nesse sentido, poderíamos tentar definir um procedimento que encontrasse o número *ótimo* de classes, entretanto deixaremos essa idéia para ser abordada em trabalhos futuros.

Na classificação automática, as palavras são divididas em classes de forma não-supervisionada, facilitando o processo de construção do Modelo da Língua (nenhum conhecimento da língua é necessário e evita-se a preparação manual do conjunto de treinamento), entretanto as classes obtidas não têm necessariamente um significado, como no caso da classificação manual.

O Modelo da Língua baseado em GIC proporcionou melhores resultados no reconhecimento (menores taxas de erro de palavra) do que o modelo bigram de classes gramaticais. Deve-se lembrar que embora não utilize modelagem estatística, o modelo baseado em GIC é um modelo sintático que utiliza informações sobre a estrutura hierárquica das frases, enquanto os modelos bigram *percebem* apenas a estrutura linear.

O modelo bigram de classes com classificação automática consegue proporcionar melhores resultados no reconhecimento (menor taxa de erro de palavra e menor tempo de reconhecimento) do que o Modelo da Língua baseado em GIC, entretanto temos alguns motivos para investir em modelos da língua baseados em gramática formal:

- Podem utilizar teorias lingüísticas que modelam muito melhor as frases da língua do que os modelos *m*-gram;
- Podem ser continuamente aperfeiçoados à medida que surjam teorias lingüísticas mais sofisticadas e métodos de análise mais eficientes;
- Constituem uma parte ativa em *sistemas de compreensão de fala* (vide [Jurafsky*94]);
- Possibilitam grande redução no espaço de busca e, consequentemente, uma redução no tempo de reconhecimento, mas isso só será conseguido quando empregarmos algoritmos de decodificação que façam uso dessa característica (vide capítulo 5);
- Podem ser combinados a métodos estatísticos com o objetivo de aumentar o desempenho dos sistemas de reconhecimento de fala (vide [Jurafsky*95]).

Tantas vantagens têm evidentemente um custo que é a maior complexidade do sistema e a necessidade de interação entre grupos de diferentes áreas (engenharia, lingüística, ciência da computação, etc.). A interação entre os vários especialistas, embora não seja uma tarefa simples, torna-se necessária devido à multidisciplinaridade da pesquisa (reconhecimento de fala e modelagem da língua).

Enquanto não são aperfeiçoados o Modelo da Língua baseado em GIC e o algoritmo de decodificação, os modelos bigram de classes com classificação automática parecem ser a melhor opção dentre aqueles apresentadas aqui.

Em geral, modelos bigram e trigram são largamente utilizados pelos sistemas de reconhecimento de fala atuais devido à simplicidade e aos bons resultados conseguidos. Entretanto, vários esforços têm sido feitos para desenvolver modelos mais sofisticados (vide [SHLT96], seções 1.6 e 3.6).

7.2 Contribuições

Com este trabalho, esperamos despertar o interesse de pesquisadores do país pelo tema tratado aqui: modelagem da língua aplicada ao reconhecimento de fala contínua. Procuramos também evidenciar a necessidade de interação entre as áreas de engenharia, lingüística e computação no desenvolvimento de novos trabalhos. Além disso, podemos citar as seguintes contribuições:

- Até onde sabemos, este é o primeiro trabalho realizado no Brasil que trata de modelagem da língua aplicada a sistemas de reconhecimento de fala;
- Levantamento bibliográfico de técnicas aplicadas à modelagem da língua em sistemas de reconhecimento de fala contínua;
- Proposta de técnicas de modelagem da língua que podem servir de base para outros trabalhos no Laboratório de Processamento Digital de Fala (LPDF) do Departamento de Comunicações da Faculdade de Engenharia Elétrica e de Computação da UNICAMP;
- Implementação de modelos que já podem ser usados em novos sistemas de reconhecimento de fala desenvolvidos no LPDF;

7.3 Sugestões para Trabalhos Futuros

Temos algumas sugestões para novos trabalhos:

- Desenvolvimento de algoritmos de busca mais eficientes e mais apropriados a sistemas de reconhecimento de fala contínua que empreguem modelos da língua;
- Utilização de teorias lingüisticas mais sofisticadas como base para modelos da língua baseados em gramática formal;
- Modelagem da língua que explore outros conhecimentos além do sintático (semântico, por exemplo);
- Utilização de gramáticas mais sofisticadas como gramáticas estocásticas independentes de contexto (*stochastic context-free grammars*) [Jurafsky*95] e gramáticas de ligação (*link grammars*) [Lafferty*92] [Stolcke*96];
- Aplicação de técnicas que permitam o aprendizado automático das estruturas da língua como programação evolutiva e técnicas conexionistas;

Apêndice A: Regras da Gramática Independente de Contexto

As regras da gramática independente de contexto (GIC) foram determinadas a partir das estruturas discutidas no capítulo 4, usando o procedimento descrito na seção 4.2.

Para serem usadas pelo analisador implementado (vide capítulo 4.3), as regras devem estar disponíveis num arquivo texto e devem usar a notação definida a seguir.

a) Regras que expandem categorias não-lexicais

São as regras que apresentam símbolos não-terminais no lado direito da expansão (como em $F \rightarrow SN \ \ SV$). Elas devem ser escritas no formato (texto):

F > SN SV;

O ponto-e-vírgula (;) no final da regra é necessário para que o analisador identifique o seu final.

b) Regras que expandem categorias lexicais

São regras que apresentam símbolos lexicais (palavras) no lado direito da expansão ($N \rightarrow$ menino , cachorro). Elas devem ser escritas no formato:

N = menino cachorro;

O ponto-e-vírgula deve ser colocado somente depois da última palavra, mesmo que exista a quebra da linha:

N = menino cachorro
bola jarro casa;

c) Comentários

Os comentários devem ser iniciados por (%) e devem terminar com (;).

No conjunto de regras, foram utilizados alguns símbolos com o objetivo de diminuir o número de regras e permitir a análise de estruturas que envolvem números..

Foi utilizado o símbolo **Vnom**, reescrito como **Vpart, Vinf** ou **Vger**, para diminuir o número de regras que definem locuções verbais. Sem utilizar o símbolo Vnom**,** teríamos que repetir as mesmas regras para cada símbolo Vpart, Vinf e Vger.

O símbolo **NumC** permite a formação de números como “vinte e cinco”, utilizando a recursividade das regras de reescrita da gramática.

A seguir, apresentamos o texto do arquivo que contém o conjunto de regras utilizado.

```
% Regras para Gramática independente de Contexto;
% Baseadas em Sintaxe X-barra;

% Expansão de categorias não-lexicais;

F       > SN SV;
F       > SV;
F       > NEG SV SN;
F       > F Conj F;
F       > SN NEG SV;
F       > NEG SV;
F       > SN SV SP;
F       > SN SV Adv;
F       > Adv SN SV;
F       > SP SN SV;
F       > SV SN;

SN      > Pess;
SN      > Dem;
SN      > SN 'e' SN;
SN      > SN 'nem' SN;
SN      > PreDet N';
SN      > PreDet Det N';
SN      > Det N';
SN      > Det PosDet N';
SN      > PreDet Det PosDet N';
SN      > PosDet N';
```

```
SN      > N';

N'      > N' SP;
N'      > N' SA;
N'      > A N';
N'      > N' REL;
N'      > N' NumC;
N'      > N' 'e' N';
N'      > N' 'nem' N';
N'      > N SP;
N'      > N;

SA      > Adv A';
SA      > A';

A'      > A SP;
A'      > A;

SP      > P SN;
SP      > P+Det SN;

REL     > 'que' F;

F'      > 'que' F;

SV      > V';
SV      > Vaux V';

V'      > V SN SP;
V'      > V SP SN;
V'      > V SN;
V'      > V SP;
V'      > V SA;
V'      > V F';
V'      > V;
V'      > Vlig SA;
V'      > Vlig F';
V'      > Vlig SN;
V'      > Vlig SP;
V'      > Adv V';
V'      > V' Adv;
V'      > V' SP;
V'      > Vnom SN SP;
V'      > Vnom SP SN;
V'      > Vnom SN;
V'      > Vnom SP;
V'      > Vnom SA;
V'      > Vnom F';
V'      > Vnom;

Vnom    > Vpart;
Vnom    > Vinf;
Vnom    > Vger;


PosDet > Poss;
PosDet > Poss NumC;
PosDet > NumC;
```

```
NumC  > Num NumC;
NumC  > Num;
NumC  > Num 'e' NumC;


% Expansão de categorias lexicais;

Det      = a as o os um uma;

P        = a até com de em entre para por sobre;

Det      = esta isto aquele este;

P+Det    = deste naquele;

P+Det    = à ao aos da das do na nas no nos pelo pelos;

N        = acordo ajustes alunos analistas aplicações atratividade atualização aumento
           banco bancos bolsa bolsas brasil caderneta café caixas cartão
           centro chamada cheque cidade cliente clientes código comparecimento
           condomínio conforto conselho consumo conta contas contribuintes
           convênio cotação crédito cruzeiros culturas cumprimento curva dados
           depósitos descontos destino determinação dezembro dia disposição documento
           dólar eficácia eficiência embarque empresa empresário estação financiamento
           formulários fortaleza funcionário futuro governo horas ibge importação
           imposto indexadores início instituições integração intercâmbio
           junho juros maioria medida mercado mês milhões minutos modificações
           momento nome notícia número operações opinião oportunidade páginas
           país passageiros perda pesquisa plano portão poupança preço preços
           prestação problema projeto quadra queda reais recife regras resolução
           reunião rio-sul saldo saques semana semanas taxas telebrás telecomunicações
           telesp tempo trabalho universidades valor vencimento vigor vôo;

V        = aceitarão aguardaremos aumentou chegaram colocará continuam convencemos
           deve devem devemos entregou fecharam foi haverá
           impulsionou informa passa passará permita permite posso preciso
           sofrerá tem terá tivemos trata una visa;

Vaux     = é está estarão estão foi haverá parece são será tem terá continuam;

Vlig     = é está estarão estão foi parece são será continuam ficará;

Vpart    = assinado considerado corrigidas detectado impulsionado instalados
           localizado medida permitido preciso registrado substituído vinculadas;

Vger     = beneficiando investindo provocando subindo;

Vinf     = afirmar antecipar aproveitar desenvolver fazer investir partir seguir ser;

A        = baixa anterior atrasadas baixa baixo brasileira cadastral central
           diferentes disponíveis disponível eletrônicos estável estatal explícita
           expresso imediato inadequado incompleto insuficiente interessante interno
           melhores monetário nacional necessário novo passada passado pequena
           perigosa prezado próximo pública radicais seguinte suficiente
           telefônica telefônicas última últimas;

Conj     = e mas ou porque quando;

Adv      = ainda amanhã aqui assim atrás bastante consideravelmente diariamente
```

mais muito normalmente ontem recentemente sempre;

NEG = não;

Dem = este esta isto aquele;

Poss = seu seus sua;

Pess = ele você;

PreDet = todas todos;

Num = cento cinco cinqüenta dezenove dezesseis dezessete dois duzentos mil nove oitenta quarenta quatro quatrocentos quinhentos quinze sessenta sete setenta três trezentos trinta um vinte;

'e' = e;

'que' = que;

'nem' = nem;

% Fim do arquivo;

Apêndice B: Algumas Frases Reconhecidas

Todos os resultados apresentados a seguir foram obtidos utilizando o sistema híbrido HMM/MLP + modelo de duração de palavra.

As palavras substituídas ou incluídas foram colocadas em negrito, enquanto as palavras excluídas foram indicadas por um asterisco (*) na posição que ocupavam na frase.

a) Frases corretas

a cotação do dólar aumentou e as bolsas fecharam em baixa
a cotação do dólar aumentou mas as bolsas fecharam em baixa
a bolsa ficará estável ou sofrerá uma pequena queda
não haverá ajustes nem modificações radicais no plano
foi detectado um problema em seu cartão ele deve ser substituído
é necessário que o convênio permita o intercâmbio
posso afirmar -lhes que o convênio permite o intercâmbio
o convênio que foi assinado recentemente permite o intercâmbio
o convênio permite o intercâmbio porque visa a integração entre alunos de culturas diferentes
o convênio que foi assinado recentemente permite o intercâmbio
é suficiente
isto é suficiente
o saldo é suficiente
o saldo de sua conta é suficiente
o saldo disponível é insuficiente
o saldo disponível em sua conta é insuficiente
isto parece insuficiente

o saldo parece insuficiente
o saldo sempre está disponível
o saldo sempre está disponível no início do mês
no início do mês o saldo está disponível
esta é a última chamada para o vôo sete três sete da rio sul
esta é uma pesquisa de opinião pública
o valor de sua conta telefônica é baixo
é de trinta mil cruzeiros o valor de sua conta telefônica
o vencimento de sua prestação será no dia quatro de junho
o preço aumentou
o preço do café aumentou
o preço do café expresso aumentou
o preço do café aumentou consideravelmente
o preço do café aumentou consideravelmente na semana passada
aumentou o preço do café
as taxas de juros no mercado interno estão subindo bastante
as contas chegaram atrasadas
as contas chegaram muito atrasadas ontem
as contas telefônicas deste mês chegaram muito atrasadas ao banco
ontem as contas chegaram aqui muito atrasadas
chegaram atrasadas
chegaram atrasadas todas as contas telefônicas deste mês
o governo aumentou o imposto no mês passado
o governo aumentou o imposto sobre importação
o governo entregou os formulários aos contribuintes
o governo entregou aos contribuintes os formulários
o banco colocará a sua disposição o novo cheque
a conta telefônica em nome de adelaide barroso terá vencimento amanhã
o mercado foi considerado inadequado
o mercado foi considerado inadequado pelos analistas
o mercado foi considerado inadequado naquele momento
diariamente
curva perigosa
dia vinte do sete
sim
não
saldo vinte e cinco reais
estação santa cruz
os bancos atrás de mais eficiência
descontos de até cinqüenta porcento

número incompleto
vinte e cinco
vinte e cinco reais
cento e vinte e cinco
cento e vinte e cinco reais
quatrocentos e quarenta e nove
dois mil cento e vinte e cinco
dezesseis mil e quinhentos
oitenta milhões trezentos e sessenta mil e duzentos e setenta e um
a telebrás, a empresa de telecomunicações brasileira, está investindo em pesquisa
a telebrás, uma empresa estatal, está investindo em pesquisa
tivemos recentemente a seguinte notícia: a telebrás passará a investir mais em pesquisa
telesp informa dezenove horas e trinta minutos
empresário, é preciso antecipar o futuro
prezado cliente, aguardaremos o seu comparecimento
o código foi registrado pelo funcionário
o convênio um documento de trinta páginas tem permitido o intercâmbio
os caixas eletrônicos não aceitarão depósitos
os caixas eletrônicos não aceitarão mais depósitos a partir da quinze horas
as operações continuam
as operações de crédito continuam

b) Sem modelo da língua

, a cotação * dólar aumentou e as bolsas fecharam em baixa ,
, a cotação * dólar aumentou **mais** as bolsas fecharam **o** em baixa ,
, a bolsa ficará **está rio** sofrerá uma pequena queda ,
, não haverá ajustes nem modificações **a de queda** no plano ,
, foi detectado * problema **vinte** seu cartão , ele **ele** ser substituído ,
, é necessário * * convênio permita **ou** intercâmbio ,
, posso afirmar **ele se** o convênio permite * intercâmbio ,
, o convênio que foi **é** assinado recentemente permite o intercâmbio ,
, é suficiente ,
, isto **o** é suficiente ,
, o saldo **o** é suficiente ,
, o saldo de sua conta **até** suficiente **em** ,
, o saldo disponível **com** é insuficiente **em** ,
, o saldo disponível em sua **o com parece** insuficiente ,

, isto parece insuficiente ,
, o saldo parece **ele** insuficiente ,
, o saldo sempre está disponível ,
, o saldo está sempre **se número mil vencimento as** ,
, **de** início * **melhores** o saldo está disponível ,
, esta é * última chamada **por os ou os esta** três sete **das** rio-sul ,
, **estão** * uma pesquisa de opinião pública ,
, o valor de sua conta telefônica é baixo ,
, é de trinta mil cruzeiros o valor de sua conta telefônica ,
, o vencimento de **sul o** prestação será no dia quatro **visa uma um** ,
, **ou** preço **o** aumentou ,
, o preço do café aumentou ,
, o preço * café expresso **o** aumentou ,
, o preço do café aumentou consideravelmente ,
, o preço do café aumentou consideravelmente na semana passada ,
, aumentou * preço do café ,
, as taxas de juros no mercado interno estão subindo bastante **em vinte** ,
, , as contas chegaram atrasadas ,
, as contas chegaram muito atrasadas **com o telesp** ,
, as contas telefônicas * **recentemente** chegaram muito atrasadas ao banco ,
, **com o ele** as contas chegaram **partir** muito **a** atrasadas ,
, chegaram atrasadas ,
, chegaram **com** atrasadas todas as contas telefônicas deste mês ,
, **ao** governo aumentou * imposto no mês passado ,
, o governo aumentou * imposto sobre importação ,
, o governo entregou * formulários aos contribuintes ,
, o governo entregou aos contribuintes **nos** formulários ,
, o banco colocará * sua disposição * novo cheque ,
, a conta telefônica **e** nome **dia** adelaide barroso **da** , terá vencimento **ao momento** ,
, o mercado foi considerado inadequado ,
, o mercado foi considerado inadequado pelos analistas ,
, **uma é a do** foi considerado inadequado naquele momento ,
, diariamente ,
curva perigosa
, dia vinte do sete ,
cinco
, não ,
, saldo , **de** vinte cinco **real -lhes** ,
, estação santa cruz ,
, **ou** bancos atrás * mais eficiência ,

, descontos **a** até cinqüenta porcento ,
, número incompleto ,
vinte * cinco
, **de** vinte * cinco **regras** ,
, cento **ele** vinte * cinco ,
, cento e vinte * cinco **regras** ,
, quatrocentos e quarenta e nove ,
, dois mil cento e vinte * cinco ,
, dezesseis mil e quinhentos ,
, **com setenta** milhões trezentos * sessenta mil **mil** duzentos * setenta e **uma** ,
, **até** telebrás , **assim** empresa **das** telecomunicações brasileira , está investindo **de** pesquisa ,
, **até** telebrás , uma empresa estatal , está investindo em pesquisa ,
, tivemos recentemente * seguinte notícia , a telebrás passará * investir mais **vinte** pesquisa ,
, telesp informa , dezenove horas e trinta minutos ,
, **é** empresário , é preciso antecipar o futuro ,
, prezado cliente , aguardaremos o seu comparecimento ,
, **no** código foi registrado **da** pelo funcionário ,
, o convênio * documento de trinta páginas , **trinta** permitido * intercâmbio ,
, os caixas eletrônicos não aceitarão depósitos ,
, os caixas eletrônicos não aceitarão mais depósitos a partir **da sete vinte** horas

c) Usando modelo bigram de 20 classes gramaticais

, a cotação * dólar aumentou **de** as bolsas fecharam em baixa ,
, a cotação * dólar aumentou **mais** as bolsas fecharam **uma** baixa .
, a bolsa ficará estável * sofrerá uma pequena queda ,
, não haverá ajustes nem modificações radicais no plano ,
, foi detectado * problema em seu cartão , ele **permite** ser substituído .
, é necessário * * convênio permita o intercâmbio .
, posso afirmar **início** * * convênio permite * intercâmbio ,
, o convênio * foi assinado recentemente permite o intercâmbio .
, é suficiente ,
, isto é suficiente ,
, o saldo é suficiente ,
, o saldo de sua conta é suficiente ,
, o saldo disponível é insuficiente ,
, o saldo disponível em sua conta **parece** insuficiente ,
, -**lhes** parece insuficiente ,
, o saldo parece ser insuficiente ,

, o saldo sempre está disponível ,
, o saldo está sempre **se número** no **vencimento as , um**
, **do** início * **melhores** * saldo está disponível ,
, esta é * última **a** chamada **por os bolsa é ser** sete **das** rio sul ,
, **estão** * uma pesquisa de opinião pública ,
, o valor de sua conta telefônica é baixo ,
, é de trinta mil cruzeiros **no** valor de sua conta telefônica .
, o vencimento de sua prestação será no dia quatro de **uma** .
, **os** preço aumentou ,
, o preço do café aumentou ,
, o preço * café expresso aumentou ,
, o preço do café aumentou consideravelmente ,
, o preço do café aumentou consideravelmente na semana passada ,
, aumentou * preço do café ,
, as taxas de juros no mercado interno estão subindo bastante ,
. as contas chegaram atrasadas .
, as contas chegaram muito atrasadas **com o telesp** ,
, as contas telefônicas * **recentemente** chegaram muito atrasadas ao banco ,
, **convênio** as contas chegaram **bastante** muito atrasadas ,
, chegaram atrasadas .
, chegaram atrasadas todas as contas telefônicas deste mês ,
, o governo aumentou * imposto no mês passado ,
, o governo aumentou * imposto sobre importação ,
, o governo entregou * formulários aos contribuintes ,
, o governo entregou os contribuintes **nos** formulários ,
, o banco colocará * **sobre** disposição * novo cheque ,
, a conta telefônica **e** nome **a** adelaide barroso , terá vencimento **ao momento** ,
, o mercado foi considerado inadequado ,
, o mercado foi considerado inadequado pelos analistas ,
, **um** mercado foi considerado inadequado naquele momento ,
, diariamente ,
curva perigosa
, dia vinte do sete ,
se vinte
, não ,
, saldo , **de** vinte * cinco reais ,
, estação santa cruz ,
, os bancos atrás de mais eficiência ,
, descontos * até cinqüenta porcento ,
, número incompleto ,

, vinte * cinco
, **um** vinte * cinco reais ,
, **as entre** vinte * cinco ,
, **será entre** vinte * cinco **regras** ,
, **a** quatrocentos e quarenta e nove ,
, **a** dois mil cento e vinte * cinco ,
, **a** dezesseis mil e quinhentos ,
, **o setenta** milhões trezentos * sessenta mil e duzentos * setenta e **uma** ,
, a telebrás , **assim** empresa **das** telecomunicações brasileira , está investindo **de** pesquisa ,
, a telebrás , uma empresa estatal , está investindo em pesquisa ,
, tivemos recentemente * seguinte notícia . a telebrás passará **vinte deste** mais **vinte** pesquisa ,
, telesp informa . dezenove horas * trinta minutos ,
, empresário , é preciso antecipar o futuro ,
, **a** prezado cliente , aguardaremos o seu comparecimento ,
, o código foi registrado pelo funcionário ,
, o convênio * documento de trinta páginas , tem permitido * intercâmbio ,
, os caixas eletrônicos não aceitarão depósitos ,
, os caixas eletrônicos não aceitarão mais depósitos a partir das **seguinte** horas

d) Usando modelo bigram de 40 classes com classificação automática através do algoritmo Simulated Annealing

, a cotação * dólar aumentou e as bolsas fecharam em baixa ,
, a cotação * dólar aumentou mas as bolsas fecharam em baixa ,
, a bolsa ficará **está rio** ou sofrerá uma pequena queda ,
, não haverá ajustes nem modificações radicais no plano ,
, foi detectado um problema em seu cartão , ele **pelo** ser substituído ,
, é necessário * **do** convênio permita o intercâmbio ,
, posso afirmar -lhes que o convênio permite o intercâmbio ,
, o convênio que foi assinado recentemente permite o intercâmbio ,
, é suficiente ,
, isto é suficiente ,
, o saldo **com** é suficiente ,
, o saldo de sua conta é suficiente ,
, o saldo disponível é insuficiente ,
, o saldo disponível em sua conta **parece** insuficiente ,
, isto parece insuficiente ,
, o saldo parece ser insuficiente ,

, o saldo sempre está disponível ,
, o saldo está sempre disponível no início do mês ,
, **de** início do **melhores** * saldo está disponível ,
, esta é * **últimas** chamada para o vôo sete três sete **das** rio sul ,
, **estão** * uma pesquisa de opinião pública ,
, o valor de sua conta telefônica é baixo ,
, é de trinta mil cruzeiros o valor de sua conta telefônica ,
, o vencimento de sua prestação será no dia quatro **afirmar , -lhes**
, o preço **o** aumentou ,
, o preço do café aumentou ,
, o preço do café expresso aumentou ,
, o preço do café aumentou consideravelmente ,
, o preço do café aumentou consideravelmente na semana passada ,
, aumentou o preço do café ,
, as taxas de juros no mercado interno estão subindo bastante ,
, , as contas chegaram atrasadas ,
, as contas chegaram muito atrasadas ontem **é um** ,
, as contas telefônicas deste mês chegaram muito atrasadas ao banco ,
, ontem as contas chegaram aqui muito atrasadas ,
, chegaram atrasadas ,
, chegaram atrasadas todas as contas telefônicas deste mês ,
, o governo aumentou o imposto mês passado ,
, o governo aumentou o imposto sobre importação ,
, o governo entregou os formulários aos contribuintes ,
, o governo entregou aos contribuintes os formulários ,
, o banco colocará a sua disposição * novo cheque ,
, a conta telefônica **de** nome **na** adelaide barroso , terá vencimento **aumentou** ,
, o mercado foi considerado inadequado ,
, o mercado foi considerado inadequado pelos analistas ,
, o mercado foi considerado inadequado naquele momento ,
, diariamente ,
curva perigosa
, dia vinte do sete ,
cinco
, não ,
, saldo , **de** vinte e cinco reais ,
, estação santa cruz ,
, **ou** bancos atrás de mais eficiência ,
, descontos **na** até cinqüenta porcento ,
, número incompleto ,

vinte e cinco
, vinte e **sim com regras** ,
, cento e vinte e cinco ,
, cento e vinte e cinco reais ,
, quatrocentos e quarenta e nove ,
, dois mil cento e vinte e cinco ,
, dezesseis mil e quinhentos ,
, oitenta milhões trezentos e sessenta mil e duzentos * setenta * **seu** ,
, a telebrás , a empresa **das** telecomunicações brasileira , está investindo **de** pesquisa ,
, a telebrás , uma empresa estatal , está investindo em pesquisa ,
, tivemos recentemente a seguinte notícia , a telebrás passará * investir mais em pesquisa ,
, telesp informa , dezenove horas e trinta minutos ,
, **é** empresário , é preciso antecipar o futuro ,
, prezado cliente , aguardaremos * seu comparecimento ,
, o código foi registrado pelo funcionário ,
, o convênio **do** documento de trinta páginas , tem permitido o intercâmbio ,
, os caixas eletrônicos não aceitarão depósitos ,
, os caixas eletrônicos não aceitarão mais depósitos a partir das quinze horas

e) Usando modelo da língua baseado em gramática independente de contexto

, a cotação **no** dólar aumentou e as bolsas fecharam em baixa ,
, a cotação **no** dólar aumentou **mais** as bolsas **e** fecharam **uma** baixa ,
, a bolsa ficará **e** estável **vôo** sofrerá uma pequena queda ,
, não haverá ajustes nem modificações radicais no plano ,
, o convênio que foi assinado recentemente permite o intercâmbio ,
, o convênio permite * intercâmbio porque visa a integração entre **a** alunos de culturas diferentes ,
, convênio que foi assinado recentemente , permite * intercâmbio ,
, é suficiente ,
, isto é suficiente ,
, o saldo é suficiente ,
, o saldo disponível é insuficiente ,
, o saldo disponível em sua conta **parece** insuficiente ,
, isto parece insuficiente ,
, o saldo parece * insuficiente ,
, o saldo sempre está disponível ,
, esta é última chamada **por** o vôo sete três sete **das** rio-sul ,
, **estão** * uma pesquisa de opinião pública ,
, o valor de sua conta telefônica é baixo ,

, é de trinta mil cruzeiros o valor de sua conta telefônica ,
, o vencimento de sua prestação será no dia quatro **dia uma** ,
, **os** preço aumentou ,
, o preço do café aumentou ,
, o preço do café expresso aumentou ,
, o preço do café aumentou consideravelmente ,
, o preço do café aumentou consideravelmente na semana passada ,
, aumentou * preço do café ,
, as taxas de juros no mercado interno estão subindo bastante ,
, , as contas chegaram atrasadas ,
, chegaram atrasadas ,
, o governo aumentou * imposto no mês passado ,
, o governo aumentou * imposto sobre importação ,
, o governo entregou * formulários aos contribuintes ,
, o governo entregou aos contribuintes **nos** formulários ,
, o banco colocará * sua disposição * novo **de** cheque ,
, o mercado foi considerado inadequado ,
, o código foi registrado pelo funcionário ,
, os caixas eletrônicos não aceitarão depósitos ,
, as operações continuam ,
, as operações de crédito continuam ,
, * saldo **disponível * incompleto** * suficiente , **una**
, **de** início * melhores o saldo está disponível ,
, as contas chegaram muito atrasadas **com o telesp** ,
, chegaram atrasadas todas as contas telefônicas deste mês ,
, **um** mercado foi considerado inadequado naquele momento ,

f) Usando modelo da língua baseado em gramática + modelo bigram de 60 classes automáticas

, a cotação do dólar aumentou e as bolsas fecharam em baixa ,
, a cotação do dólar aumentou mas as bolsas fecharam em baixa ,
, a bolsa ficará estável ou sofrerá uma pequena queda ,
, não haverá ajustes nem modificações radicais no plano ,
, o convênio que foi assinado recentemente permite o intercâmbio ,
, o convênio permite o intercâmbio porque visa a integração entre alunos de culturas diferentes ,
, o convênio que foi assinado recentemente , permite o intercâmbio ,
, é suficiente ,

, isto é suficiente ,
, o saldo é suficiente ,
, o saldo disponível é insuficiente ,
, o saldo disponível em sua conta **parece** insuficiente ,
, isto parece insuficiente ,
, o saldo parece * insuficiente ,
, o saldo sempre está disponível ,
, esta é a última chamada para o vôo sete três sete **das** rio-sul ,
, isto é uma pesquisa de opinião pública ,
, o valor de sua conta telefônica é baixo ,
, é de trinta mil cruzeiros o valor de sua conta telefônica ,
, o vencimento de sua prestação será no dia * * **comparecimento** ,
, o preço aumentou ,
, o preço do café aumentou ,
, o preço do café expresso aumentou ,
, o preço do café aumentou consideravelmente ,
, o preço do café aumentou consideravelmente na semana passada ,
, aumentou * preço do café ,
, as taxas de juros no mercado interno estão subindo bastante ,
, , as contas chegaram atrasadas ,
, chegaram atrasadas ,
, o governo aumentou * imposto no mês passado ,
, o governo aumentou * imposto sobre importação ,
, o governo entregou os formulários aos contribuintes ,
, o governo entregou aos contribuintes os formulários ,
, o banco colocará a sua disposição * novo cheque ,
, o mercado foi considerado inadequado ,
, o código foi registrado pelo funcionário ,
, os caixas eletrônicos não aceitarão depósitos ,
, as operações continuam ,
, as operações de crédito continuam ,
, o saldo de sua conta é suficiente ,
, **reunião mil e a resolução a junho** está disponível , ,
, as contas chegaram muito atrasadas ontem ,
, chegaram atrasadas todas as contas telefônicas deste mês ,
, o mercado foi considerado inadequado naquele momento ,

Apêndice C: Palavras Classificadas Usando Simulated Annealing

A lista seguinte apresenta a classificação de palavras obtida usando o algoritmo Simulated Annealing com 100 classes. O número entre parêntesis ao lado de cada palavra indica o número de ocorrências da palavra.

classe 0
entregou(2)
provocando(1)
adotou(1)
inteiro(1)
dado(1)
danificada(1)
toda(1)

classe 1
centro(2)
sete(6)
banco(7)
sentia(1)

classe 2
ficará(1)
você(3)
sempre(2)
fortaleza(1)
estatal(1)
rápida(1)
policiais(1)
eu(3)

classe 3
intercâmbio(8)
anterior(1)
futuro(1)
exclusão(1)
vírus(2)
comprimento(1)
mercosul(1)
escapar(1)

classe 4
em(21)
boicotam(1)

classe 5
interessante(1)
início(2)
preço(7)
explícita(1)
sonho(1)
dentro(1)
julgamento(1)
subterrâneos(1)
interior(1)
nordeste(1)

classe 6
ele(14)
isto(2)
esta(3)
passageiros(1)
empresário(1)
segundo(1)
rebeldes(2)
raul(1)
ela(1)

classe 7
dólar(2)
café(5)
conselho(1)
congo(1)
sudão(1)
cavalo(2)
palácio(2)
castelo(1)
empréstimo(3)
vestido(1)
conde(2)

classe 8
e(36)

classe 9
cotação(2)
medida(1)
determinação(2)
idéia(1)
senadora(1)
sela(6)
cessão(4)
seção(1)
gola(3)

classe 10
bolsa(2)
empresa(2)
morte(1)
simulação(1)
lei(1)
afirmação(1)
discurso(1)
participação(1)
onda(1)
basílica(1)
manchas(1)
-la(1)

classe 11
projeto(3)
condomínio(1)
valor(2)
imposto(3)
aumento(4)
documento(1)
período(1)
grupo(1)
laço(5)

classe 12
na(10)
última(2)
semana(2)
outra(1)
vez(1)
lesão(1)
primeira(2)
segunda(3)
respingou(1)

classe 13
contas(5)
regras(3)
urbanas(1)
saias(1)

classe 14
ou(2)
econômico(2)

classe 15
conta(5)
pública(1)
prestação(1)
disposição(1)
barroso(1)
brasileira(1)
por(2)
entram(1)
marcha(1)

classe 16
governo(7)
país(7)
próximo(2)
pai(2)
coração(1)

classe 17
sobre(3)
tem(2)
todos(7)
impulsionou(2)
virtuais(1)
ecoavam(1)

classe 18
ajustes(1)
atrasadas(6)
aqui(1)
analistas(2)
disponíveis(1)
usa(1)
pequenos(1)

classe 19
seu(6)
una(1)
órbita(1)
à(1)

classe 20
bancos(6)
caixas(3)
crimes(1)
termos(1)
passos(4)

classe 21
queda(1)
disponível(5)
cheque(1)
imediato(1)
natural(1)
calendário(1)
contaminado(2)
quadrada(1)
cela(4)
manca(1)
madura(1)
furada(1)
limpa(1)

classe 22
substituído(1)
passado(1)
amanhã(2)
perigosa(1)
reais(3)
paulo(1)
incompleto(1)
pobres(1)
aliados(1)
formidáveis(1)
vermelhos(1)

classe 23
para(8)
justiça(1)

classe 24
entre(1)
nos(4)
aos(2)
impulsionado(1)
poderá(1)

classe 25
vinculadas(1)
mês(6)
devido(3)
paço(4)

classe 26
vôo(1)
mercado(5)
código(1)
principal(2)
voto(1)
lançamento(1)
carro(1)
levaram(1)
arsenal(1)

classe 27
são(6)
criar(1)
atingiu(2)
carga(1)

classe 28
universidades(1)
deste(2)
consumo(4)
último(1)
majestoso(2)

classe 29
conforto(1)
cinco(7)
nove(2)
quinhentos(1)
continuam(5)
cachorros(1)
tribais(1)

classe 30
trinta(6)
milhões(1)
sessenta(1)
vários(2)
porcos(1)

classe 31
integração(4)
junho(1)
importação(2)
poupança(1)
dezembro(2)
nascimento(1)
corrupção(1)
manga(15)

classe 32
radicais(1)
será(1)
juros(1)
outros(1)
novas(1)
cercadas(1)
viral(1)
fidel(1)
proporções(1)
mulheres(1)
assis(1)
abalos(1)
paris(1)
ouvidos(2)
fica(1)

classe 33
vinte(9)
quarenta(1)
duzentos(1)
setenta(1)
financiamento(3)
ingredientes(1)

classe 34
aguardaremos(1)
últimas(1)
-se(1)
níveis(1)
exatamente(1)
golpe(1)
pela(1)
sério(1)
extremamente(3)

classe 35
alunos(1)
vencimento(2)
nome(1)
caderneta(1)
curva(1)
atrás(1)
descontos(1)
número(2)
grupos(1)
testes(1)

classe 36
diferentes(1)
eficiência(1)
porcento(1)
semanas(1)
rapidamente(1)
militares(1)
pacientes(1)
indetectáveis(1)
vacinas(1)
anos(1)
isso(1)
militar(1)
segurança(1)
acreditou(1)
lasso(3)

classe 37
bolsas(2)
taxas(1)
operações(5)
famílias(1)
modelos(1)
estratégias(1)

classe 38
reunião(1)
chamada(1)
passada(2)
segunda-feira(1)
cerebral(1)
lista(1)
sessão(5)

classe 39
modificações(1)
ser(3)
fazer(6)
doze(1)

classe 40
que(13)
antecipar(1)
ibge(1)
derrubaria(1)
deles(1)

classe 41
a(79)
nessa(1)

classe 42
oportunidade(1)
telebrás(3)
seguinte(1)
resolução(1)
seguir(3)
explosão(1)
fome(1)
partilha(1)
galeria(1)

classe 43
assinado(3)
considerado(4)
tivemos(1)
argentino(1)
preso(1)
descoberto(1)
comunistas(1)

classe 44
vamos(2)
brasil(1)
agora(1)
tropas(1)
eles(1)
naquela(1)
menem(1)
comprei(1)
serzi(1)

classe 45
manter(2)
tenta(1)
eletronicamente(1)
numa(5)
noite(1)
tentou(1)
ocorrerá(1)
logo(1)

classe 46
formulários(2)
momento(2)
cruz(1)
indexadores(1)
preços(2)
drogas(2)
epidêmicas(1)
sul-coreano(1)
gritava(1)
mudar(1)

classe 47
permita(1)
permite(7)
cruzeiros(1)
permitido(1)
condenou(1)
descrédito(1)
reduzir(1)

classe 48
eficácia(1)
cidade(1)
atratividade(1)
folha(1)
certidão(1)
região(1)
luta(1)
reconciliação(1)
oposição(1)
camisa(3)
minha(1)

classe 49
mas(1)
convencemos(1)
consideravelmente(3)
contribuintes(2)
nacional(1)
retomar(2)
interessa(1)

classe 50
três(3)
horas(6)
central(4)
livros(1)
pessoas(1)
discriminada(1)

classe 51
colocará(1)
destino(1)
notícia(1)
passará(1)
cumprimento(1)
normalmente(1)
continuou(1)
ataque(1)
hotéis(1)
caiu(1)
carrega(1)
envolve(1)
transformar(1)
após(2)

classe 52
porque(1)
quando(1)
devemos(1)
das(5)
só(1)
explodiu(1)
católica(1)

classe 53
o(93)

classe 54
um(13)
cerca(1)
ficou(1)
manchada(1)

classe 55
telefônica(3)
terá(1)
passo(1)
dele(1)
complexos(1)
rumo(1)

classe 56
do(39)

classe 57
fecharam(2)
problema(1)
aplicações(1)
investindo(2)
violência(1)
satélite(1)
previsto(1)
feita(1)
escondidas(1)
coloca(1)
desfilavam(1)

classe 58
não(12)
afirmar(1)
estão(1)
informa(1)
participante(1)
ficaram(1)
parecem(1)

classe 59
dezenove(1)
aceitarão(3)
reduz(1)
ganhou(1)
podemos(1)
elimina(1)
conseguiu(1)
ter(1)
sujá(1)

classe 60
chegaram(6)
telefônicas(2)
mafioso(1)
costumam(1)
varia(1)

classe 61
visa(1)
aproveitar(1)
com(3)
depósitos(3)
prolongar(1)
guarda(2)
fará(2)
sujou(1)

classe 62
os(19)
naquele(2)
testam(1)

classe 63
posso(1)
estação(1)
telesp(1)
prezado(1)
ainda(2)
colômbia(1)
étnica(1)
coquetel(1)
polícia(1)
tudo(1)
nós(1)
nenhum(2)
nada(1)

classe 64
recentemente(3)
localizado(1)
ontem(3)
todas(1)
inadequado(4)
publicou(1)
voltam(1)
discutiremos(1)
arregacei(1)

classe 65
páginas(1)
real(3)
órgãos(2)
quase(1)
internacional(1)
detalhes(1)

classe 66
dois(2)
dezesseis(1)
atualização(5)
seus(2)
já(2)
granadas(2)
duas(3)
sob(1)

classe 67
sua(7)
opinião(1)
adelaide(1)
telecomunicações(1)
corrigidas(1)
mudanças(1)
atingida(1)

classe 68
convênio(10)
presidente(3)
encontro(1)

classe 69
é(20)
passa(1)
parece(2)

classe 70
perda(1)
investir(1)
partir(3)
biblioteca(1)
mensagem(1)
guerra(2)
bomba(2)
apoiar(1)
bandeira(1)
minoria(1)

classe 71
culturas(1)
mais(7)
cinqüenta(3)
acordo(2)
esquerda(1)
exercícios(1)
estudar(1)

classe 72
cartão(1)
trabalho(2)
diariamente(1)
sim(1)
comparecimento(1)
funcionário(1)
modelo(4)
hoje(1)
fechada(1)
europa(1)

classe 73
uma(17)
dela(1)
fita(1)

classe 74
até(3)
criados(1)
ratos(1)
vítimas(1)
num(1)
supersônicas(1)
divulgados(1)

classe 75
sofrerá(1)
plano(5)
sul(2)
defendendo(1)
investigação(1)
comer(1)

classe 76
aumentou(10)
expresso(1)
monetário(1)
todo(1)
pode(1)
ocorreu(1)

classe 77
se(3)
dia(5)
melhores(1)
quinze(3)
dezessete(1)
precisavam(1)
às(1)

classe 78
dados(1)
conquistaram(1)
servirá(1)
participam(1)
analisou(1)
continua(1)
condenação(1)
fez(1)
embarcou(1)
deu(2)
pôs(1)

classe 79
santa(1)
cliente(1)
nas(1)
desafia(1)
espalhar(1)
provocou(1)
espanhola(1)
aconteceu(1)
queríamos(1)
analista(1)
vai(1)
modo(1)

classe 80
haverá(1)
-lhes(1)
muito(4)
pelos(2)
estarão(1)
necessariamente(1)

classe 81
pelo(2)
este(2)
colocar(1)
corações(1)
casos(1)
seria(2)
intervalo(1)
sala(1)
prateleira(3)

classe 82
mil(6)
trezentos(1)
minutos(1)
rivais(1)
edifícios(1)
países(1)

classe 83
nem(1)
deve(2)
interno(1)
eletrônicos(3)
devem(5)
acha(1)
podiam(2)

classe 84
da(17)

classe 85
pequena(1)
quadra(1)
cópia(1)
ação(1)
rara(1)
estava(2)
passarela(1)
antiga(2)
pessoa(1)
suculenta(1)

classe 86
foi(19)
jornal(2)

classe 87
está(13)
embarque(1)

classe 88
baixa(2)
pesquisa(7)
bastante(2)
vigor(1)
eleição(3)
museu(1)
vasos(1)
hora(1)
azul(1)
resolvidas(1)

classe 89
de(65)

classe 90
estável(1)
desenvolver(1)
saques(1)
contra(3)
realiza(1)
armados(1)
apoio(1)
comi(2)

classe 91
trata(1)
quatro(2)
instituições(1)
enviar(1)
sangue(2)
caminho(1)
março(1)
suco(1)
caldo(3)

classe 92
tempo(1)
saldo(9)
novo(2)
ressarcimento(1)
substituto(1)
arreio(1)

classe 93
necessário(1)
suficiente(4)
insuficiente(4)
baixo(1)
preciso(1)
humanos(1)
israelense(1)

classe 94
as(21)
lendas(1)

classe 95
detectado(1)
registrado(1)
espalhada(1)
relata(1)
adiado(1)
reservado(1)
ignorado(1)
guardada(2)
interrompida(1)
produzida(1)

classe 96
recife(1)
cento(3)
quatrocentos(1)
oitenta(2)
crédito(4)
valores(1)
forte(1)

classe 97
ao(6)
rio(1)
pedem(1)

classe 98
no(18)

classe 99
subindo(1)
cadastral(5)
clientes(1)
foram(3)
estavam(1)
passagens(1)
corridas(1)
céu(1)

Referências

[Aarts*89] Aarts, E., Korst, J., *Simulated Annealing and Boltzman Machines*. John Wiley & Sons, 1989.

[Alcaim*92] Alcaim, A., Solewics, J.A., Moraes, J.A., *Freqüência de Ocorrência dos Fones e Listas de Frases Foneticamente Balanceadas no Português Falado no Rio de Janeiro.* Revista da Sociedade Brasileira de Telecomunicações, v. 7, n. 1, Dezembro 1992.

[Bahl*83] Bahl, L.R., Jelinek, F., Mercer, R.L., *A Maximum Likelihood Approach to Continuous Speech Recognition.* IEEE Transactions Pattern Analysis and Machine Intelligence, v. 5, n. 2, March 1983.

[Bahl*89] Bahl, L.R., Brown, P.F., Souza, P.V., Mercer, R.L., *A Tree-Based Statistical Language Model for Natural Language Speech Recognition.* IEEE Transactions on Acoustics, Speech and Signal Processing, v. 37, n. 7, July 1989.

[Chomsky57] Chomsky, N., *Syntatic Structures.* The Hague: Mounton, 1957.

[Chomsky65] Chomsky, N. *Aspects of the Theory of the Syntax*. Cambridge: MIT Press, 1965

[Cunha85] Cunha, C., *Nova Gramática do Português Contemporâneo*. Editora Nova Fronteira, 1985.

[Deller*93] Deller, J., Proakis, J.G., Hansen, J.H.L., *Discrete-Time Processing of Speech Signals*. Macmillan, 1993.

[Earley70] Earley, J., *An Efficient Context-Free Parsing Algorithm.* Communications of the Association for Computing Machinery, 13(2), p.94-102, 1970.

[Haykin94] Haykin, S., *Neural Networks: A Comprehensive Foundation*. Prentice Hall, New Jersey, 1994.

[Jackendoff77] Jackendoff, R.S., $\overline{X}$ *Syntax: A Study of Phrase Structure.* MIT Press, Cambridge, Mass, 1977.

[Jardino*93] Jardino, M., Adda, G., *Automatic Determination of a Stochastic Bi-Gram Class Language*. Proceedings of ICASSP, II – 41, 1993.

[Jardino96] Jardino, M., *Multilingual Stochastic N-Gram class Language Models.* IEEE ICASSP, v. 1, p. 161-163, 1996.

[Jelinek96] Jelinek, F., *Language Modeling for Speech Recognition*. Proceedings of the

ECAI 96: Workshop on Extended Finete State Models of Language, ed. A. Kornai, 1996.

[Jurafsky*94] Jurafsky, D., Wooters, C., Segal, J., Stolcke, A., Fosler, E., Tajchman, G., Morgan, N., *The Berkeley Restaurant Project*. ICSLP, v. 4, p. 2139 – 2142, 1994.

[Jurafsky*95] Jurafsky, D., Wooters, C., Segal, J., Stolcke, A., Fosler, E., Tajchman, G., Morgan, N., *Using a Stochastic Context-Free Grammar as a Language Model for Speech Recognition*. IEEE ICASSP, v. 1, p. 189 – 192, 1995.

[Kasami65] Kasami, T., *An Efficient Recognition and Syntax Algorithm for Context-Free Languages.* Technical Report AF-CRL-65-758, Air Force Cambridge Research Laboratory, Bedford, MA, 1965.

[Katz95] Katz, S., *Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer*. IEEE Transactions on Acoustics, Speech and Signal Processing, 35(3), p. 400-401, 1995.

[Kirkpatrick*83] Kirkpatrick, S., Gelatt Jr, C.D., Vecchi, M.P., *Optimization by simulated annealing*. Science, n. 220, p. 671 – 680, 1983.

[Kneser*93] Kneser, R., Ney, H., *Improved Clustering Techniques for Class-based Statistical Language Modeling*. EUROSPEECH'93, Berlin, Set., 1993.

[Lafferty*92] Lafferty, J., Sleator, D., Temperley, D., *Grammatical Trigrams: A Probabilistic Model of Link Grammar.* Proceedings of the 1992 AAAI Fall Symposium on Probabilistic Approaches to Natural Language, Cambridge, MA, 1992.

[Lang*90] Lang, K.J., Waibel, A.H., *A Time-Delay Neural Network Architecture for Isolated Word Recognition.* Neural Netwoks, v. 3, n. 1, p. 23-43, 1990.

[LeeCH*89] Lee, C.H., Rabiner, L.R., *A Frame Synchronous Network Search Algorithm for Connected Word Recognition*. IEEE Transactions on Acoustics, Speech and Signal Processing, v. 37, n. 11, p. 1649-1658, Nov., 1989.

[LeeKF89] Lee, K.F., *Automatic Speech Recognition: The Development of the SPHINX System*. The Kluwer International Series in Engineering and Computer Science, 1989.

[Lippmann89] Lippmann, R.P., *Review of Neural Networks for Speech Recognition.* Neural Computation, v. 1, n. 1, p. 1-38, 1989.

[Lippmann97] Lippmann, R.P., *Speech Recognition by Machines and Humans.* Speech Communication, 22, p. 1-15, 1997.

[Martin*95] Martin, S., Liermann, J., Ney, J., *Algorithms for Bigram and Trigram Word Clustering*. EUROSPEECH'95, p. 1253-1256, 1995.

[McClosky88] McClosky, J., *Syntatic Theory*. Linguistics: The Cambridge Survey, ed. Frederick J. Newmeyer, Cambridge University Press, v. 1, p. 19-59, 1988

[Moisa*95] Moisa, L., Giachin, E., *Automatic Clustering of Words for Probabilistic Language Models*. EUROSPEECH'95, p. 1249-1252, 1995.

[Morais97] Morais, E. S., *Reconhecimento Automático de Fala Contínua Empregando Modelos Híbridos ANN + HMM*. Campinas, UNICAMP, 1997.

[Morgan*95a] Morgan, N., Bourlard, H.A., *Continuous Speech Recognition: An Introduction to the Hibrid HMM/Connectionist Approach*. IEEE Signal Processing Magazine, p. 25-42, Maio, 1995.

[Morgan*95b] Morgan, N., Bourlard, H.A., *Neural Networks for Statistical Recognition of Continuous Speech*. Proceedings of the IEEE, v. 83, n. 5, p. 742-770, Maio,1995.

[Ney*94] Ney, H., Essen, U., Kneser, R., *On Structuring Probabilistic Dependences in Stochastic Language Modelling*. Computer Speech and Language, v. 8, p. 1-38, 1994.

[Ortmanns*97] Ortmanns, S., Ney, H., Aubert, X., *A Word Graph Algorithm for Large Vocabulary Continuous Speech Recognition*. Computer Speech and Language, v. 11, p. 43-72, 1997.

[Peeling*88] Peeling, S.M., Moore, R.K., *Isolated Digit Recognition Experiments Using the Multilayer Perceptron*. Speech Communication, v. 7, p. 403-409, 1988.

[Rabiner*85] Rabiner, L., Levinson, S.E., *A Speaker-Independent, Syntax-Directed, Connected Word Recognition System Based on Hidden Markov Models and Level Building*. IEEE Transactions on Acoustic, Speech and Signal Processing, v. 33, n. 3, p. 561-573, Jun., 1985.

[Rabiner*93] Rabiner, L., Juang, B.H., *Fundamentals of Speech Recognition*. PTR Prentice-Hall, 1993.

[Radford81] Radford, A., *Transformational Syntax*. Cambridge Textbooks in Linguistics, Cambridge University Press, 1981

[Radford88] Radford, A., *Transformational Grammar*. Cambridge University Press, 1988

[Raposo78] Raposo, E. P., *Introdução à Gramática Generativa: Sintaxe do Português*. Moraes Editores, 1978.

[Rich*94] Rich, E., Knight, K., *Inteligência Artificial.* Makron Books, 1994.

[Richard*91] Richard, M.D., Lippmann, R.P., *Neural Network Classifiers Estimate Bayesian a Posteriori Probabilities*. Neural Computation, 3, p. 461-483, 1991

[Sepsy*97] Sepsy, M., Horvat, B., *Statistical Language Modeling Based on Classes*. COST249, p. 17-18, Roma, Fev., 1997.

[SHLT96] *Survey of the State of the Art in Human Language Technology*. Ed. R. A. Cole, J. Mariani, H. Uszkoriet, A. Zaenen, V. Zue. Cambridge, MIT Press, 1996.

[Stolcke*94] Stolcke, A., Segal, J., *Precise n-gram Probabilities from Stochastic Context-Free Grammars.* Proceedings of the 31th Annual Meeting of the Association for Computational Linguistics, p. 74-79, New Mexico State University, Las Cruces, NM, 1994.

[Stolcke*96] Stolcke, A., Chelba, C., Engle, D., Jimenez, V., Mangu, L., Printz, H., Ristad, E., Rosenfeld, R., Wu, D., Jelinek, F., Khudanpur, S., *Structure and Performance of a Dependency Language Model.* Eurospeech97, v. 5, p. 2775 – 2778, 1997.

[Stolcke95] Stolcke, A., *An Efficient Probabilistic Context-Free Parsing Algorithm that Computes Prefix Probabilities.* Association for Computational Linguistics, 1995.

[Suhm*94] Suhm, B., Waibel, A., *Towards Better Language Models for Spontaneous Speech*. Proceedings of ICSLP, 1994.

[Urbela*95] Ueberla, J. P., *More Efficient Clustering of N-Grams For Statistical Language Modeling*. EUROSPEECH'95, p. 1257-1260, 1995.

[Waibel*88] Waibel, A., Hanazawa, T., Hinton, G., Shikano, K., Lang, K., *Phoneme Recognition: Neural Networks vs. Hidden Markov Models.* Proceedings of ICASSP (NY, NY), p. 107-110, 1988.

[Young*96] Young, S., *A Review of Large-Vocabulary Continuous-Speech Recognition*. IEEE Signal Processing Magazine, p. 45-57, Set., 1996.

[Younger67] Younger, D. H., *Recognition and Parsing of Context-Free Languages in Time n^3*. Information and Control, 10(2):189-208, 1967.

Glossário

Algoritmo de análise – Algoritmo responsável por aplicar as regras de reescrita de uma gramática durante a análise de uma frase (vide seção 4.3).

Algoritmo de busca – Num sistema de reconhecimento de fala, trata-se do algoritmo responsável por encontrar a palavra ou seqüência de palavras correspondente à elocução de entrada (vide capítulo 5).

Algoritmo de busca integrada – Algoritmo de busca que leva em conta as restrições do Modelo da Língua durante o processo de busca e não como uma etapa de pós-processamento (vide seção 5.2).

Algoritmo de decodificação – Vide algoritmo de busca.

Análise em constituintes imediatos – Divisão de uma frase em grupos naturais de palavras com o objetivo de estudar e obter a estrutura hierárquica desta frase (vide seção 4.1.1).

Árvore de análise – Representação da análise de uma frase através de uma estrutura em árvore (vide seção 2.4.2).

Árvore de derivação – Representação em forma de árvore de uma derivação, podendo corresponder à árvore de análise de uma frase.

Capacidade gerativa fraca – Capacidade de uma gramática de gerar as frases gramaticais de uma língua.

Colocação em posição de contraste – Consiste em colocar um grupo de palavras de uma frase entre as palavras "foi" e "que". Ex.: "o cachorro mordeu o menino" $\rightarrow$ "foi *o cachorro* que *mordeu o menino*".

Conjunto Earley – Grupo de Estados Earley relacionados à mesma posição na frase, ou seja, a mesma palavra (vide seção 4.3).

Constituintes imediatos – Dois ou mais nós de uma árvore serão constituintes imediatos de um determinado nó se forem imediatamente dominados por este último (vide seção 4.1.1).

Derivação – Seqüência de aplicações das regras de reescrita que leva uma seqüência de símbolos em outra seqüência de símbolos terminais e/ou não-terminais (vide seção 2.4.2).

Derivação à esquerda – Derivação na qual são expandidos os símbolos não-terminais que

estão posicionados mais à esquerda da seqüência de símbolos (vide seção 2.4.2).

Dominância – Dizemos que o nó n_1 de uma árvore *domina* o nó n_2 da mesma árvore quando n_1 está posicionado acima de n_2 e existe uma seqüência de arcos que leva o nó n_1 ao nó n_2 (vide seção 2.4.2).

Dominância imediata – Dizemos que n_1 *domina imediatamente* n_2, Quando n_1 domina n_2 e existe um arco que liga diretamente os dois nós (vide seção 2.4.2).

Estado Earley – Um estado Earley é representado por $i :_k X \rightarrow l.m$ e está associado a uma derivação parcial que faz parte da análise de uma frase (vide seção 4.3).

Forma Normal de Chomsky (FNC) – Dizemos que uma gramática está na Forma Normal de Chomsky quando suas regras de reescrita são da forma $A \rightarrow B\ C$ ou $A \rightarrow a$, onde A, B, C são símbolos não-terminais e a é um símbolo terminal (vide seção 2.4.2).

Forma Normal de Chomsky Estendida (FNCE) – Dizemos que uma gramática está na Forma Normal de Chomsky Estendida quando suas regras de reescrita são da forma $A \rightarrow B\ C$, $A \rightarrow B$ ou $A \rightarrow a$, onde A, B, C são símbolos não-terminais e a é um símbolo terminal (vide seção 2.4.2)

Gramática estocástica independente de contexto (GEIC) – Gramática independente de contexto cujas regras de reescrita estão associadas a probabilidades.

Gramática finitamente ambígua – Gramática que atribui um número finito de árvores de análise às frases reconhecidas pela gramática.

Gramática formal – Uma gramática formal pode ser definida como $G = \{V_N, V_T, R, S\}$, onde V_N é o conjunto de símbolos não-terminais, V_T é o conjunto de símbolos terminais, R é o conjunto de regras de reescrita ou regras de produção e S é símbolo inicial (vide seção 2.4.1).

Gramática independente de contexto (GIC) – Gramática cujas regras de reescrita são da forma $A \rightarrow b$, onde A é um símbolo não terminal e b é uma seqüência não-vazia de símbolos terminais e/ou não-terminais (vide seção 2.4.2).

Gramáticas fortemente equivalentes – Gramáticas que atribuem a mesma estrutura de análise às frases reconhecidas.

Gramáticas fracamente equivalentes – Gramáticas que geram o mesmo conjunto de frases.

Língua derivada – A Língua derivada de uma gramática $G = \{V_N, V_T, R, S\}$ é o conjunto

de frases (seqüências de símbolos terminais contidos em V_T) obtidas através da aplicação das regras de produção de R, partindo do símbolo inicial S.

Língua – Conjunto finito ou infinito de seqüências (frases), cada uma contendo um número finito de elementos e construída por concatenação sobre um conjunto finito de símbolos (vocabulário ou alfabeto).

Modelo acústico – Parte do sistema de reconhecimento de fala que possibilita o cálculo do termo $P(O|W)$, onde $O = o_1...o_T$ é a seqüência de vetores de parâmetros acústicos que representa a elocução e $W = w_1...w_N$ é uma possível seqüência de palavras correspondente à elocução (vide seção 2.2).

Modelo *m*-gram - Modelo da Língua no qual a ocorrência de uma palavra depende somente das *m*-1 palavras anteriores (vide seção 2.3).

Modelo bigram – Modelo da Língua no qual a ocorrência de uma palavra depende somente da palavra anterior (vide seção 2.3.1).

Modelo da Língua – Parte do sistema de reconhecimento de fala que permite avaliar o termo $P(W)$, ou seja, a probabilidade a priori de uma seqüência $W = w_1...w_N$ de palavras.

Perplexidade – A perplexidade corresponde ao número médio de palavras que pode seguir uma determinada seqüência de palavras anterior. Formalmente, a perplexidade é definida como $PP = 2^{H(W)}$, onde $H(W) = \frac{1}{N} \cdot \log_2 P(w_1...w_N)$ é a entropia da seqüência $W = w_1...w_N$ (vide seção 3.3).

Poda de histograma – Tratando-se de algoritmos de busca, a poda de histograma corresponde a manter as *N* melhores hipóteses (de acordo com alguma função de custo) e eliminar as hipóteses restantes (vide seção 5.2).

Precedência – A relação de precedência está relacionada à ordem na horizontal dos nós de uma árvore (vide seção 2.4.2).

Quadro Earley – Reúne todos os conjuntos Earley e todas as derivações realizadas na análise de uma frase (vide seção 4.3).

Regra de produção – As regras de produção são representadas por $a \rightarrow b$ e indicam que o símbolo a pode ser substituído por b (a é reescrito como b).

Regra de reescrita – Vide regra de produção.

Símbolos não-terminais – Representam classes de palavras ou categorias relacionadas à estrutura das frases.

Símbolos terminais – Tratando-se de línguas naturais, são as palavras do vocabulário.

Sintagma nominal – Grupo de palavras de uma frase que está estruturado em torno de um núcleo nominal (vide seções 4.1.2 e 3.2.1).

Sintagma verbal – Grupo de palavras de uma frase que está estruturado em torno de um núcleo verbal (vide seções 4.1.2 e 3.2.1).

Sintaxe X-barra – Teoria que sustenta a existência de categorias intermediárias entre as categorias lexicais e os sintagmas correspondentes (vide seção 4.1.3).

Topicalização – Consiste em deslocar um grupo de palavras de uma frase para o início ou final da frase. Ex.: "o cachorro mordeu o menino" $\rightarrow$ "mordeu o menino, o cachorro".

Verbos copulativos – Chamados também de verbos de ligação, são considerados elementos de ligação entre o sujeito e o predicativo.