

IMPLEMENTAÇÃO DE UM SISTEMA DE RECONHECIMENTO DE FALA CONTÍNUA  
COM AMPLO VOCABULÁRIO PARA O PORTUGUÊS BRASILEIRO

Rafael Teruszkin Tevah

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS  
PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE  
FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS  
PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA  
ELÉTRICA.

Aprovada por:

---

Prof. Fernando Gil Vianna Resende Junior, Ph.D.

---

Prof. Abraham Alcaim, Ph.D.

---

Prof<sup>a</sup>. Mariane Rembold Petraglia, Ph.D.

RIO DE JANEIRO, RJ - BRASIL

JUNHO DE 2006

TEVAH, RAFAEL TERUSZKIN

Implementação de um Sistema de  
Reconhecimento de Fala Contínua com  
Amplio Vocabulário para o Português  
Brasileiro [Rio de Janeiro] 2006

XI, 115 p. 29,7 cm (COPPE/UFRJ,  
M.Sc., Engenharia Elétrica, 2006)

Dissertação - Universidade Federal do  
Rio de Janeiro, COPPE

1. Reconhecimento de Voz

I. COPPE/UFRJ    II. Título (série)

*Aos meus pais, Dirce e Acacio  
e meus irmãos, Patrícia e Eduardo.*

## *Agradecimentos*

*Ao professor Fernando Gil pelo apoio, orientação e tempo dedicado ao projeto.*

*Ao Ranniery Maia, Ricardo Cirigliano e Filipe Barbosa, pela introdução ao HTK e disponibilização do transcritor automático de palavras.*

*Ao professor Steve Young e sua equipe pela disponibilização do HTK e ATK, além da pronta resposta aos meus e-mails com dúvidas.*

*Aos meus pais, irmãos, tios, primos e amigos por todo o apoio, incentivo e paciência.*

*Aos meus avós Abrão e Cecília, em memória, pelo exemplo pessoal que eles foram.*

*À Lailinha, por todo apoio, carinho e companheirismo, sobretudo durante a reta final deste trabalho.*

*À toda comunidade científica, pelos vários e-mails trocados, esclarecimento de dúvidas e dicas extremamente úteis.*

*Ao povo brasileiro que, através de impostos pagos, financia o sistema brasileiro de ensino público superior.*

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

## IMPLEMENTAÇÃO DE UM SISTEMA DE RECONHECIMENTO DE FALA CONTÍNUA COM AMPLO VOCABULÁRIO PARA O PORTUGUÊS BRASILEIRO

Rafael Teruszkin Tevah

Junho/2006

Orientador: Fernando Gil Vianna Resende Junior

Programa: Engenharia Elétrica

Este trabalho descreve a implementação de um sistema de reconhecimento de fala contínua com amplo vocabulário para o português brasileiro. O sistema desenvolvido utiliza ferramentas disponíveis nos pacotes HTK e ATK. Um algoritmo de conversão de seqüência fonética em texto baseado em DTW e *One-Stage* foi proposto. Foram realizados testes para analisar a correlação entre as seguintes variáveis no contexto de reconhecimento de fala contínua: taxa de acerto de palavras, perplexidade, distintos modelos de linguagem, complexidade computacional e tamanho do vocabulário. Foi utilizada uma base de voz para o treinamento estocástico dos modelos acústicos baseados em HMMs contínuos, e foi desenvolvida uma base de texto para treinamento dos modelos de linguagem baseados em *n*-gramas. Foram testados vocabulários entre 3.528 e 60.000 palavras e a melhor taxa de acerto obtida no primeiro caso foi de 90% no reconhecimento de sentenças com 9 a 12 palavras, e de 81% no segundo caso, ambos dependentes de locutor, com perplexidades variando entre 250 e 350 e tempos de processamento inferiores a um minuto por sentença. O conversor fonema-grafema baseado em DTW teve sucesso em 100% dos testes no reconhecimento de frases fechadas, tendo aplicado à sua entrada uma seqüência fonética com 22% de erros, em média. Com o conversor baseado em *One-Stage*, obteve-se uma taxa de 91% no reconhecimento de palavras conectadas utilizando um dicionário de 183 palavras.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

IMPLEMENTATION OF A LARGE VOCABULARY CONTINUOUS SPEECH  
RECOGNITION SYSTEM FOR BRAZILIAN PORTUGUESE

Rafael Teruszkin Tevah

June/2006

Advisor: Fernando Gil Vianna Resende Junior

Department: Electrical Engineering

This work presents the implementation of a large vocabulary speech recognition system for Brazilian Portuguese. The implemented system uses tools available on HTK and ATK toolkits. An algorithm for converting phonetic sequences to graphemes based on DTW and One-Stage was proposed. Tests were conducted in order to check the correlation on the context of continuous speech recognition among the following variables: word recognition rate, perplexity, distinct language models, computational complexity and vocabulary size. A speech database was used to train the stochastic acoustic models based on continuous HMMs, and a textual database was developed to train language models based on  $n$ -grams. Vocabularies ranging between 3.528 and 60.000 words were tested and the best accuracy rate obtained for the first case was 90% when recognizing sentences with 9 to 12 words, and 81% for the second case, both of them being speaker dependent, with perplexities ranging between 250 to 350 and processing times less than one minute per sentence. The phonetic sequence to grapheme converter based on DTW succeeded in 100% of tests related to closed sentences recognition when receiving as input a phonetic sequence with 22% of errors, in average. The converter based on One-Stage presented an accuracy rate of 91% for connected word recognition using a dictionary with 183 words.

## Sumário:

<b>1. Introdução</b>	<b>1</b>
1.1. Motivação	1
1.2. Aplicações	3
1.2.1. Sistemas de Interface	3
1.2.2. Sistemas Transcritores	3
1.3. Objetivos	4
1.4. Estrutura da Dissertação	4
<b>2. Fundamentos e Estado da Arte</b>	<b>5</b>
2.1. Modelagem de Sistemas CSR	5
2.1.1. Extração de Parâmetros	7
2.1.2. Modelagem Acústica	8
2.1.3. Modelo de Linguagem	13
2.1.4. Decodificador	17
2.2. Estado da Arte	18
2.3. Softwares de Apoio à Implementação de Sistemas CSR	20
<b>3. Estudo de Decodificadores e a Estratégia <i>One-Pass</i></b>	<b>22</b>
3.1. Algoritmos de Busca	22
3.1.1. Decodificação de Viterbi com Feixe de Busca	25
3.1.2. Decodificação de Pilha (stack decoding)	26
3.1.3. Busca em Múltiplas Passadas	26
3.1.4. Busca em Avanço e Retorno (forward-backward)	26
3.2. Fundamentos para uma Busca Síncrona	27
3.3. Algoritmo One Pass usando um Léxico Linear	29
3.3.1. Definição do Espaço de Busca	29
3.3.2. Recursividade com Programação Dinâmica	31
3.3.3. Busca em Feixe	33
3.4. Algoritmo One Pass usando um Léxico em Árvore	34
3.4.1. Definição do Espaço de Busca	34
3.4.2. Recursividade com Programação Dinâmica	36
3.4.3. Extensão para LM do tipo Trigrama	37
3.5. Estratégia de Poda no Algoritmo One Pass	38
3.5.1. Poda Acústica	39
3.5.2. Poda com Modelo de Linguagem	39
3.5.3. Poda por Histograma	39
<b>4. Implementação de um Sistema CSR a partir do HTK</b>	<b>40</b>
4.1. Preparação de Dados	41
4.2. Treinamento do Modelo Acústico	41
4.3. Treinamento do Modelo de Linguagem	43
4.4. Reconhecimento de Fonemas	44
4.5. Reconhecimento de Palavras	45
<b>5. Proposta de Algoritmo: Conversor Fonema-Grafema</b>	<b>46</b>
5.1. Conversão Baseada em DTW	46
5.2. Conversão Baseada em One-Stage	48
<b>6. Preparação de Bases para o Português Brasileiro</b>	<b>51</b>
6.1. Arquivos de Fala	51
6.2. Textos Escritos	52
6.3. Dicionário Fonético	53
<b>7. Testes e Resultados</b>	<b>54</b>
<b>8. Conclusões e Trabalhos Futuros</b>	<b>66</b>
<b>Referências</b>	<b>69</b>
<b>Apêndice 1</b>	<b>76</b>
<b>Apêndice 2</b>	<b>77</b>
<b>Apêndice 3</b>	<b>78</b>
<b>Apêndice 4</b>	<b>90</b>
<b>Apêndice 5</b>	<b>91</b>

## Índice de Figuras:

Figura 1: Tecnologias utilizadas na construção de interfaces de voz [2].....	1
Figura 2: Evolução no Reconhecimento Automático da Fala (ASR) .....	2
Figura 3: Proposta de Sistema de Conversação Telefônica para Deficientes Auditivos .....	4
Figura 4: Diagrama em blocos de um sistema de reconhecimento de fala contínua (CSR) baseado em modelos estatísticos de subunidades de palavras.....	6
Figura 5: Front-end com processador baseado em MFCCs.....	7
Figura 6: Modelo de fonema baseado em HMM.....	8
Figura 7: União de estados acusticamente indistinguíveis .....	12
Figura 8: Agrupamento baseado em árvores de decisão .....	12
Figura 9: Histórico de taxas de erro nas avaliações do DARPA realizadas entre 1988 a 1999 [4].....	20
Figura 10: Exemplo de Espaço de Busca .....	22
Figura 11: Algoritmos de Busca .....	24
Figura 12: Dinâmica da busca em feixe ( <i>beam search</i> ) [39] .....	25
Figura 13: Estrutura de um modelo de trifone e o espaço de busca associado [63].....	28
Figura 14: Espaço de busca para um vocabulário de 3 palavras (A,B,C) .....	30
Figura 15: Exemplo de percurso de alinhamento no tempo .....	30
Figura 16: Recombinações de caminho dentro de uma mesma palavra e entre palavras .....	30
Figura 17: Ilustração dos ponteiros de retorno .....	32
Figura 18: Ilustração de vetores de rastreamento .....	33
Figura 19: Dicionário léxico de pronúncia organizado em formato de árvore .....	34
Figura 20: Recombinação do LM bigrama e manipulação de silêncio (Sil) intra-frase em um léxico em árvore (vocabulário de 3 palavras: A, B e C) [63].....	35
Figura 21: LM trigrama em um léxico em árvore (vocabulário de 3 palavras: A, B e C) .....	38
Figura 22: Estrutura de ferramentas disponíveis no pacote HTK para testes de sistemas CSR [13].....	40
Figura 23: Processo de treinamento dos modelos acústicos .....	42
Figura 24: Processo de treinamento e teste de um modelo de linguagem. ....	43
Figura 25: Implementação de um sistema de reconhecimento de fonemas .....	44
Figura 26: Implementação de um sistema de reconhecimento de palavras .....	45
Figura 27: Diagrama de blocos do conversor PS2G.....	46
Figura 28: Exemplo de “alinhamento fonético” implementado no algoritmo DTW, durante o reconhecimento da palavra “congresso” com as restrições locais e globais implementadas. ...	47
Figura 29: Exemplo de “alinhamento fonético” implementado através do algoritmo <i>One-Stage</i> . Os grafemas reconhecidos correspondem às referências com índices 1, 3 e 2. Outros planos de referências que não fizeram parte da solução, não aparecem na figura.....	49
Figura 30: Relacionamento entre as Bases de Dados utilizadas. ....	51
Figura 31: Organização das bases de treino e teste para obtenção de taxas de acerto ( $\overline{WRR}$ ) média.....	55
Figura 32: Evolução da perplexidade em função do número de frases utilizadas no treino do LM.....	56
Figura 33: Evolução da $\overline{WRR}$ em função do número de frases usadas no treino do modelo de linguagem. ....	57
Figura 34: Evolução do tempo de processamento médio por sentença em função do número de frases usadas no treino do modelo de linguagem. ....	57
Figura 35: Evolução da $\overline{WRR}$ em função do número gaussianas usadas no treino do modelo acústico. ....	58
Figura 36: Evolução do $\overline{Tp}$ em função do número gaussianas usadas no treino do modelo acústico. ....	58
Figura 37: Evolução da $\overline{WRR}$ em função do número gaussianas usadas no treino do modelo acústico. ....	59
Figura 38: Evolução do $\overline{Tp}$ em função do número gaussianas usadas no treino do modelo acústico. ....	59
Figura 39: Crescimento do dicionário em função do número de frases utilizadas. ....	60
Figura 40: Evolução da perplexidade em função do aumento do dicionário .....	61

Figura 41: Evolução da $\overline{WRR}$ em função do tamanho do vocabulário e com os parâmetros do feixe de busca reduzidos em 20%. .....	62
Figura 42: Evolução do $\overline{Tp}$ em função do tamanho do vocabulário e com os parâmetros do feixe de busca reduzidos em 20%. .....	62
Figura 43: Evolução da $\overline{WRR}$ em função do tamanho do vocabulário para o reconhecimento baseado em bigramas e desconsiderando trifones entre-palavras. ....	62
Figura 44: Evolução do $\overline{Tp}$ em função do tamanho do vocabulário para o reconhecimento baseado em bigramas e desconsiderando trifones entre-palavras. ....	63
Figura 45: Evolução da perplexidade de modelos bigrama em função do tamanho dos dicionários. ....	64
Figura 46: Evolução da $\overline{WRR}$ em função do tamanho do vocabulário, para os sistemas baseados no conversor PS2G e no HTK. ....	65
Figura 47: Evolução do $\overline{Tp}$ em função do tamanho do vocabulário, para os sistemas baseados no conversor PS2G e no HTK. ....	65
Figura 48: $WRR$ observada em testes preliminares com independência de locutor. ....	90

## Índice de Tabelas:

Tabela 1: Perplexidades típicas para vários domínios [2].	16
Tabela 2: Perplexidades e Taxas de Erro ( <i>WER</i> ) associadas a diferentes tarefas [40].	16
Tabela 3: Algoritmo One-Pass com DP usando um léxico linear [63].	33
Tabela 4: Algoritmo One-Pass com DP usando um léxico em árvore [63].	37
Tabela 5: Distância entre os fonemas de seqüências de teste e de referência (similaridade).	47
Tabela 6: Lista com os fonemas (40) utilizados no sistema de CSR [98].	53
Tabela 7: Medição da perplexidade de modelos treinados com diferentes quantidades de frases.	55
Tabela 8: Medição da $\overline{WRR}$ em função da variação do modelo de linguagem.	56
Tabela 9: Medição do tempo de processamento médio em função da variação do modelo de linguagem.	57
Tabela 10: Medição da $\overline{WRR}$ em função da variação do número de gaussianas.	58
Tabela 11: Medição da $\overline{WRR}$ em função da variação do número de gaussianas.	59
Tabela 12: Medição da perplexidade em função do aumento do dicionário.	60
Tabela 13: Medição do $\overline{Tp}$ em função da aumento do tamanho do dicionário e comparação com os resultados obtidos no reconhecimento baseado em trigramas e trifones entre-palavras.	63
Tabela 14: Exemplos de resultados processados pelo HTK após um reconhecimento de palavras utilizando bigramas e o dicionário de 60.000 palavras. Entre colchetes, aparecem os erros encontrados.	63
Tabela 15: Divisão entre locutores e grupos frásicos da base.	90

## **Lista de Acrônimos:**

<b>AI</b>	Artificial Intelligence
<b>API</b>	Application Programming Interface
<b>ASR</b>	Automatic Speech Recognition
<b>BSD</b>	Berkeley Software Distribution
<b>CFG</b>	Context-free Grammar
<b>CHMM</b>	Continuous HMM
<b>CMN</b>	Cepstral Mean Normalization
<b>CSR</b>	Continuous Speech Recognition
<b>CWR</b>	Connected Word Recognition
<b>DCT</b>	Discrete Cosine Transform
<b>DHMM</b>	Discrete HMM
<b>DUR</b>	Discrete Utterance Recognition
<b>DP</b>	Dynamic Programming
<b>DTW</b>	Dynamic Time Warping
<b>FFT</b>	Fast Fourier Transform
<b>HMM</b>	Hidden Markov Model
<b>IWR</b>	Isolated Word Recognition
<b>LM</b>	Language Model
<b>LPC</b>	Linear Predictive Coding
<b>LVCSR</b>	Large Vocabulary Continuous Speech Recognition
<b>LVR</b>	Large Vocabulary Recognition
<b>MDE</b>	Metadata Extraction
<b>MFCC</b>	Mel-Frequency Cepstral Coefficient
<b>OCR</b>	Optical Character Recognition
<b>OOV</b>	Out of Vocabulary
<b>PLP</b>	Perceptual Linear Predictive coding
<b>PS2G</b>	Phonetic Sequence to Graphemes
<b>PS2GC</b>	Phonetic Sequence to Graphemes Converter
<b>PSTN</b>	Public Switched Telephone Network
<b>RT</b>	Rich Transcription
<b>RTN</b>	Recursive Transition Network
<b>SCHMM</b>	Semi-Continuous HMM
<b>SLM</b>	Stochastic Language Models
<b>STT</b>	Speech-to-Text Transcription
<b>TDT</b>	Topic Detection and Tracking
<b>URA</b>	Unidade de Resposta Audível
<b>VQ</b>	Vector Quantization
<b>WER</b>	Word Error Rate
<b>WRR</b>	Word Recognition Rate

# 1. Introdução

## 1.1. Motivação

Tal como os sistemas robóticos se baseiam na capacidade humana de movimentação e articulação, orientação espacial e expressões faciais, o histórico de pesquisas na área de processamento da fala também vai ao encontro da habilidade que o homem apresenta ao falar, ouvir, ler, escrever, reconhecer pessoas pela voz, verter palavras de um idioma para outro e, tantas outras características ligadas ao tema, aparentemente simples, porém intrinsecamente complexas.

Atento a nossa realidade, todo o esforço despendido nessa área de pesquisa ganha ainda mais força quando vinculado ao suporte a deficientes físicos. Os sistemas construídos com esta finalidade representam uma ferramenta imprescindível à inclusão digital destes indivíduos. Comparações de desempenho (*benchmarking*) entre os sistemas desenvolvidos na área e destes com seres humanos são constantemente realizadas [1], de forma a criar um patamar que viabilize a comunicação de igual para igual entre homem-máquina e máquina-máquina.

Tecnologias de processamento da fala em conjunto com técnicas avançadas de lingüística, inteligência artificial e recuperação de dados, viabilizam a modelagem de um sistema de interface falada entre homens e máquinas conforme apresentado por Ronald A. Cole em “*Survey of the State of the Art in Human Language Technology*” [2] e representado de forma resumida na Figura 1. Nessa figura, observa-se o papel atribuído a um bloco gerenciador do sistema que, vinculado a um banco de dados e embutido de inteligência artificial e conhecimento lingüístico, permitiria um diálogo em linguagem natural entre homem e máquina através das suas interfaces de reconhecimento e síntese de voz.

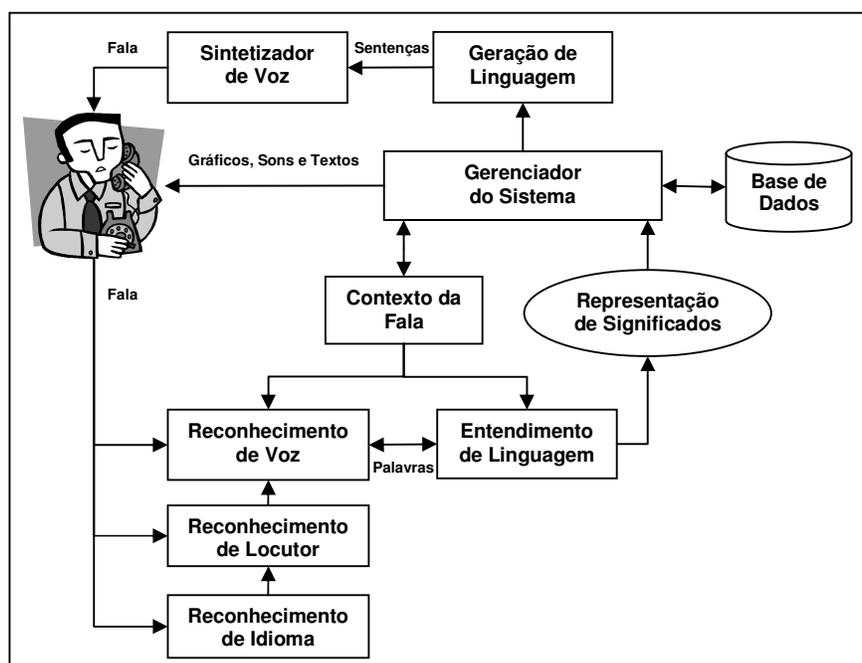


Figura 1: Tecnologias utilizadas na construção de interfaces de voz [2]

Pesquisadores no campo de reconhecimento automático da fala, tema desta dissertação, foram responsáveis por um número significativo de avanços nas duas últimas décadas, influenciados por áreas também em destaque como processamento de sinais, algoritmos, arquitetura de software e hardware. A evolução desta área está resumidamente descrita na Figura 2 e, seu avanço foi viabilizado principalmente pela adoção das seguintes técnicas e metodologias listadas abaixo:

- Adoção do paradigma de reconhecimento de fala baseado na detecção de padrões (*Pattern Matching*) em contrapartida a modelos acústico-fonéticos ou baseados em Inteligência Artificial [3];
- Utilização de uma grande base de amostras da fala discursadas por uma diversificada população de locutores com intuito de treinar os modelos das unidades fundamentais da linguagem;
- Organização de todas as fontes de conhecimento da fala e linguagem em uma estrutura de rede de estados finita;
- Uso de métodos de busca heurística que utilizam programação dinâmica para localizar a melhor seqüência de palavras para os textos falados em um léxico estruturado em rede.

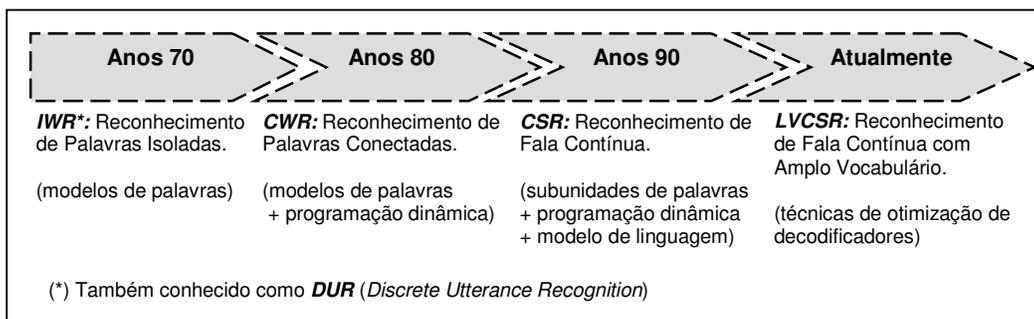


Figura 2: Evolução no Reconhecimento Automático da Fala (ASR)

Universidades e indústria têm tentado resolver problemas práticos da área, de forma a possibilitar o reconhecimento da fala natural. O objetivo é a construção de sistemas que possam ser utilizados sem a necessidade de treinamento intensivo por parte do usuário e com taxas de erros mínimas. A última década testemunhou um progresso significativo na tecnologia de reconhecimento de fala. Em 1997 já se reportava taxas de erro da ordem de 10% para sistemas de reconhecimento de fala contínua com vocabulários amplos (20.000 palavras ou mais) nos testes realizados regularmente nos EUA através do Departamento de Defesa [4].

Nesse contexto, o objetivo desta dissertação foi estudar as diversas técnicas utilizadas pelos sistemas de fala contínua que implementam o estado da arte. Mais especificamente, algoritmos de busca, considerados como o coração destes sistemas, foram estudados com detalhe. A metodologia que utiliza modelos acústicos e lingüísticos no reconhecimento de fala foi adaptada para o português brasileiro. Uma metodologia alternativa, baseada no reconhecimento parcial de trifones, foi implementada. Testes variando os principais parâmetros utilizados em fala contínua foram realizados de forma a viabilizar a construção futura de um sistema de reconhecimento de fala contínua, com amplo vocabulário e completamente adaptado ao nosso idioma.

## **1.2. Aplicações**

São diversas as aplicações que utilizam ou poderiam utilizar sistemas de reconhecimento de voz. Exemplos comuns da aplicabilidade de técnicas de reconhecimento de voz são: transcrição de texto, comando de dispositivos por voz, recuperação de dados, atendimento eletrônico por telefone, biometria, entre outros. Os casos de uso podem ser razoavelmente distribuídos em dois grupos principais descritos a seguir: sistemas de interface e sistemas transcritores.

### **1.2.1. Sistemas de Interface**

O grupo, aqui denominado por sistemas de interface, reúne os sistemas que utilizam técnicas de reconhecimento de voz para acelerar ações de comando ou navegação por menus de sistema facilitando o acesso de usuários convencionais e viabilizando por completo o uso de computadores por usuários portadores de deficiência física.

Sistemas que utilizam reconhecimento de voz em sua interface já foram alvo de estudo no passado [5] e hoje fazem parte do dia-a-dia, como celulares que discam através de comandos de voz, centrais telefônicas ou URAs (Unidade de Resposta Audível) que solicitam que o usuário fale o assunto o qual deseja informação e sistemas como o DOSVOX [6], responsável pela inclusão digital de milhares de portadores de deficiência física no Brasil e no exterior.

As possibilidades e flexibilidades do reconhecimento de voz têm levado uma série de empresas, de diversos setores, a optar por esta tecnologia em seus sistemas de atendimento. A facilidade do uso da voz como interface pode resultar em inúmeros desdobramentos positivos, tais como redução de custos operacionais, aumento da segurança, criação/manutenção de imagem de modernidade e aumento de receitas.

A aplicação de técnicas de reconhecimento de voz para sistemas de interface, não exige alto grau de aprofundamento na área e é extremamente motivadora. Esse motivo, aliado ao rápido desenvolvimento de computadores pessoais e dispositivos portáteis, fará com que sistemas com interfaces de voz estejam razoavelmente popularizados em poucos anos.

### **1.2.2. Sistemas Transcritores**

Diferente dos sistemas de interface, os sistemas transcritores têm por objetivo captar e transcrever em linguagem corrente o que um usuário estiver falando independente de um contexto específico. Não há uma preocupação em tomar uma determinada ação, porém simplesmente transcrever. Estes sistemas funcionam como uma secretária virtual que vai apenas tomando nota do que está sendo dito. Os requisitos técnicos nestes sistemas são bem maiores do que nos sistemas de interface. Grandes vocabulários e restrições lingüísticas fazem com que a complexidade computacional cresça exponencialmente. Atualmente, inúmeros sistemas comerciais já cumprem razoavelmente bem esse papel, porém ainda há muito que evoluir nessa área.

A mesma técnica usada em sistemas de ditado vem sendo pesquisada com a finalidade de desenvolver um sistema de conversação telefônica para deficientes

auditivos [7]. O objetivo é capacitar os portadores de deficiência a utilizarem a rede de telefonia pública (PSTN), facilitando assim sua integração à sociedade. O funcionamento deste sistema, composto por módulos de síntese e reconhecimento de voz, está representado na Figura 3. A proposta é que os blocos e dispositivos de entrada e saída estejam integrados em um único dispositivo portátil, de baixo custo e fácil utilização.

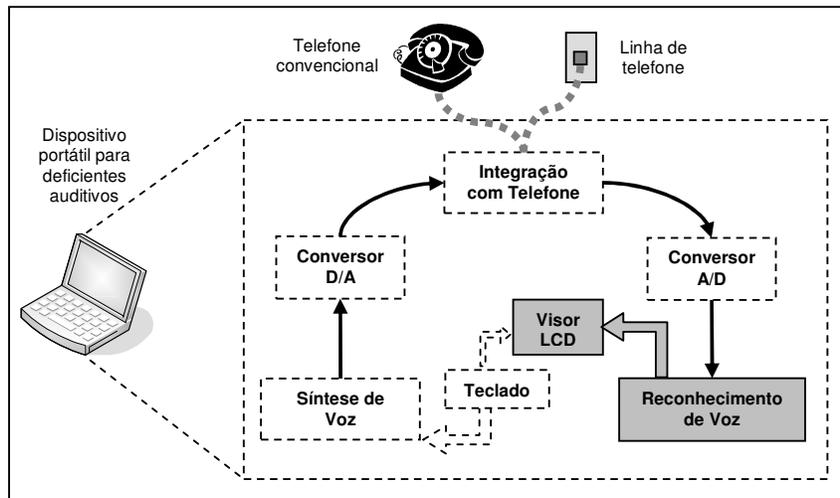


Figura 3: Proposta de Sistema de Conversação Telefônica para Deficientes Auditivos.

### 1.3. Objetivos

Resumidamente, os objetivos deste trabalho foram:

- Implementação de um sistema CSR com amplo vocabulário para o português brasileiro baseado em pacotes disponíveis na Internet
- Proposição de algoritmo para conversão de seqüência fonética em texto
- Obtenção de curvas que relacionem variáveis conhecidas de sistemas CSR como: taxa de acerto de palavras, perplexidade, diferentes modelos de linguagem, complexidade computacional, tamanho do vocabulário, etc.
- Obtenção ou desenvolvimento de bases de dados que viabilizem os testes

### 1.4. Estrutura da Dissertação

Os próximos capítulos desta dissertação estão organizados da seguinte maneira: o Capítulo 2 apresenta fundamentos teóricos e o estado da arte em sistemas de reconhecimento de fala contínua com amplo vocabulário. O Capítulo 3 descreve os algoritmos de busca utilizados em sistemas de reconhecimento de fala contínua (decodificadores) e sua inserção dentro do contexto de Inteligência Artificial. São também descritos, neste capítulo, os algoritmos mais utilizados da família *One-Pass*. No Capítulo 4 descreve-se a implementação realizada do sistema de reconhecimento de fala contínua baseado no HTK e no ATK. O Capítulo 5 traz uma proposta de algoritmo de busca utilizando resultados parciais gerados por um sistema de reconhecimento de fonemas. O Capítulo 6 mostra como as bases de treinamento e teste, utilizadas nesta dissertação, foram desenvolvidas. O Capítulo 7 traz os resultados de testes realizados e no Capítulo 8 são apresentadas conclusões sobre este trabalho e sugestões de caminhos a serem seguidos.

## 2. Fundamentos e Estado da Arte

Um sistema de reconhecimento de fala converte o sinal acústico observado em sua representação ortográfica correspondente. Houve um grande avanço na resolução deste problema através da utilização de um modelo estatístico de distribuição conjunta,  $P(W, X)$ , entre a seqüência de palavras pronunciadas  $W$  e a seqüência de informações acústicas observadas  $X$ , numa abordagem conhecida como modelo *fonte-canal* [2]. Nesta abordagem, o sistema de reconhecimento procura uma estimativa  $\hat{W}$ , da seqüência de palavras pronunciadas, a partir da evidência acústica observada  $X$ , usando a distribuição de probabilidades *a posteriori*,  $P(W | X)$ . Para minimizar a taxa de erro, o sistema escolhe a seqüência de palavras que maximiza essa distribuição:

$$\hat{W} = \arg \max_w [P(W | X)] = \arg \max_w \left[ \frac{P(W)P(X | W)}{P(X)} \right] \quad (1)$$

Na equação (1), após a aplicação do Teorema de Bayes, a distribuição *a posteriori* é decomposta em  $P(W)$ , a probabilidade *a priori* da seqüência de palavras  $W$ , e  $P(X | W)$ , que é a probabilidade de observar a evidência acústica  $X$  quando a seqüência  $W$  é pronunciada. A distribuição  $P(W)$  refere-se às palavras que poderiam ter sido pronunciadas (a fonte) e está associada a um modelo de linguagem. O modelo de probabilidade de uma observação  $P(X | W)$  (o canal) é chamado de modelo acústico.

Além de reconhecimento de voz, o modelo *fonte-canal* vem sendo muito utilizado em reconhecimento ótico de caracteres (OCR – *Optical Character Recognition*), onde a seqüência observada é a imagem de caracteres impressos, ou ainda no reconhecimento de escrita a mão, onde a observação consiste numa seqüência de traços numa tela. Da mesma forma, utiliza-se o modelo *fonte-canal* na tradução por máquina (MT – *Machine Translation*), onde a observação é uma seqüência de palavras num idioma e  $W$  representa a tradução desejada em outro idioma [2].

### 2.1. Modelagem de Sistemas CSR

Sistemas atuais de reconhecimento de fala contínua (CSR – *Continuous Speech Recognition*) com amplo vocabulário (LVR – *Large Vocabulary Recognition*) são estritamente baseados nos princípios de reconhecimento estatístico de padrões. Os métodos básicos em que se aplicam estes princípios têm ainda forte influência de sistemas pioneiros da década de 70 [8], [9]. A arquitetura representada na Figura 4 é praticamente um consenso na área e é composta pelos seguintes componentes: interface de captação e extração de parâmetros do sinal de fala (conhecido como *front-end*), modelos acústicos, léxico de palavras (opcional), modelo lingüístico e, principalmente, o decodificador. Estes blocos serão explorados no restante desta e das próximas seqüências.

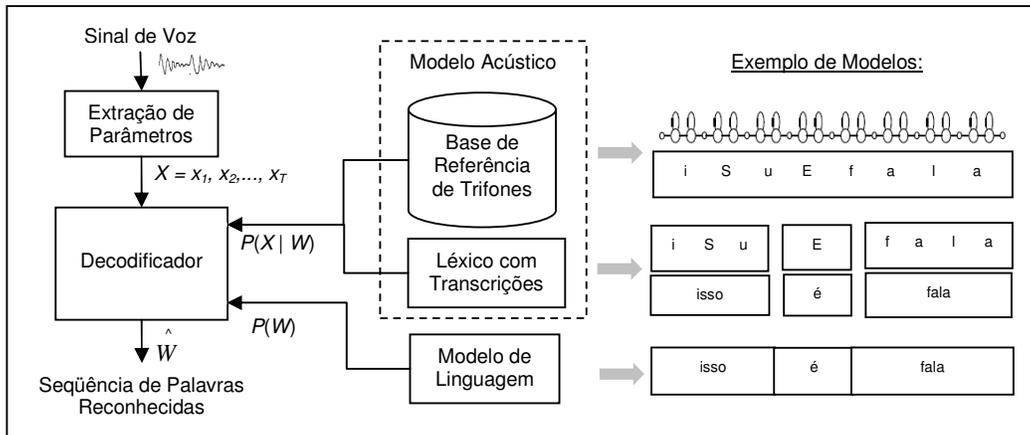


Figura 4: Diagrama em blocos de um sistema de reconhecimento de fala contínua (CSR) baseado em modelos estatísticos de subunidades de palavras.

Um trecho de fala desconhecido, representado na Figura 4, é recebido pelo *front-end* e é convertido em uma seqüência de vetores acústicos  $X = \{x_1, x_2, \dots, x_T\}$ . Este trecho se refere a uma seqüência de palavras  $W = \{w_1, w_2, \dots, w_n\}$ , com  $n$  desconhecido, e é responsabilidade do sistema CSR determinar a seqüência mais provável,  $\hat{W}$ , para os vetores acústicos observados,  $X$ , nos termos definidos pela equação (1). Ainda na Figura 4, pode-se observar um exemplo de como estes termos se relacionam. A seqüência de palavras  $W = \text{"isso é fala"}$  é postulada pelo decodificador e o modelo de linguagem computa sua probabilidade  $P(W)$ . Cada palavra é então convertida numa seqüência de sons básicos ou fonemas usando o dicionário de pronúncia (léxico). Para cada fonema há um modelo estatístico correspondente denominado modelo escondido de Markov (HMM – *Hidden Markov Model*). A seqüência de HMMs necessária para representar a fala postulada é concatenada formando um único modelo composto e a probabilidade deste modelo gerar a seqüência observada  $X$  é calculada. Esta é a probabilidade  $P(X | W)$  que o sistema busca. A princípio, este processo pode ser repetido para todas as seqüências de palavras possíveis e a seqüência mais provável é selecionada como saída do reconhecedor.

Para converter a arquitetura desenhada acima em um sistema prático, é requerida a solução de alguns problemas. Primeiramente, a parametrização do *front-end* deve extrair do sinal da fala toda a informação acústica necessária de forma compacta e compatível com os modelos acústicos baseados em HMM. Em segundo lugar, os HMMs, propriamente ditos, devem representar com precisão as distribuições de cada som em cada um dos muitos contextos em que eles possam ocorrer. Os parâmetros necessários são estimados a partir de bases de falas que dificilmente cobrem todos possíveis contextos. Em terceiro lugar, o modelo de linguagem deve ser projetado para dar precisão às predições de palavras levando-se em conta seu histórico recente. No entanto, assim como para os HMMs, a variabilidade da fala é um problema sempre presente e o modelo de linguagem deve estar apto a lidar com seqüências de palavra para as quais não houve ocorrência nos dados de treino. Finalmente, o processo delineado acima para encontrar  $W$  enumerando todas as possíveis seqüências de palavras é computacionalmente impraticável para um amplo léxico de palavras. Entretanto, seqüências de palavras com potencial podem ser exploradas em paralelo, descartando hipóteses tão logo as mesmas se tornem improváveis. Este processo é denominado decodificação (*decoding*) e o projeto de decodificadores eficientes é crucial na realização de sistemas CSR práticos e capazes de realizar operações rápidas e com precisão nas plataformas de computação existentes. As seções abaixo abordam cada um destes problemas com detalhe.

### 2.1.1. Extração de Parâmetros

Uma suposição importante feita no desenho dos reconhecedores atuais é de que o sinal de fala pode ser considerado estacionário durante um intervalo de alguns milissegundos. Assim, a função prioritária do estágio de parametrização denominada *front-end* é dividir o sinal de fala em blocos e de cada bloco derivar uma estimativa suavizada do espectro. O espaçamento entre blocos é tipicamente de 10ms e os blocos são normalmente superpostos para se obter uma janela maior de análise, tipicamente de 25ms. Para ilustrar um processamento típico, a Figura 5 mostra o *front-end* de um reconhecedor baseado em coeficientes *mel-cepstrais* (MFCCs - *Mel-Frequency Cepstral Coefficients*) [10].

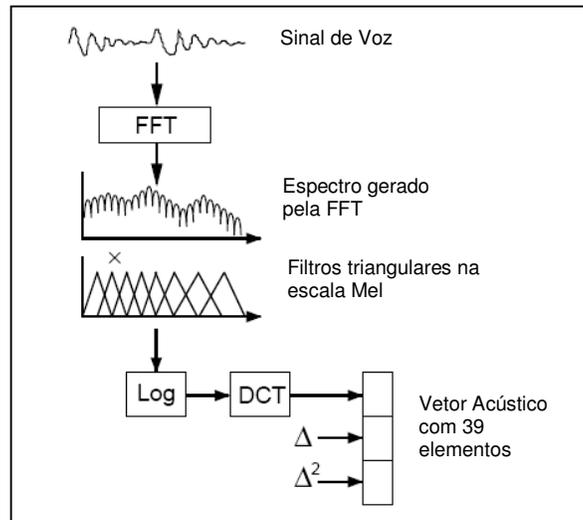


Figura 5: Front-end com processador baseado em MFCCs

Na maioria dos sistemas CSR atuais, a energia do sinal juntamente aos 12 primeiros coeficientes *mel-cepstrais* são computados para formar um vetor acústico básico de 13 elementos. Como será visto adiante, a modelagem acústica assume que cada vetor acústico é decorrelacionado com os seus vizinhos. Essa suposição é um pouco pobre visto que os requisitos físicos do aparato vocal humano garantem que há continuidade entre sucessivas estimativas espectrais. No entanto, adicionando-se as diferenciais de primeira e segunda ordem aos coeficientes estáticos básicos, irá reduzir enormemente o problema [11], [12]. Usualmente isto é aproximado com ajuste feito por regressão linear sobre uma janela cobrindo dois vetores antes e dois após o vetor calculado [13]:

$$d_t = \frac{\sum_{\theta=1}^{\Theta} \theta (c_{t+\theta} - c_{t-\theta})}{2 \sum_{\theta=1}^{\Theta} \theta^2} \quad (2)$$

onde  $d_t$  é coeficiente diferencial computado no tempo  $t$  em termos dos coeficientes estáticos correspondentes  $c_{t+\theta}$  e  $c_{t-\theta}$ . O parâmetro  $\Theta$  indica o tamanho da janela de regressão, usualmente igual a 2. Quando a equação (2) é utilizada para o cálculo dos coeficientes de 1ª e 2ª ordem, o vetor acústico final fica com 39 coeficientes.

Para compensar os efeitos de longa duração no espectro causados principalmente pelo uso de diferentes microfones ou canais de áudio (incluindo a distância entre o

locutor e o microfone) é usual realizar uma subtração espectral de todos os vetores de entrada, operação esta, conhecida como *Cepstral Mean Normalization* (CMN). Na prática, o cálculo da média espectral é atualizado a cada segmento de forma a proteger o sistema de reconhecimento contra variações do canal (média móvel) [14]:

$$\mu' = \alpha(\mu - x) + x \quad (3)$$

onde  $\mu'$  é a média atualizada e  $\alpha$  é uma constante de tempo (normalmente igual a 0.995). Um valor inicial  $\mu_0$  é carregado inicialmente no cálculo da média móvel para facilitar o reconhecimento dos primeiros segmentos.

Um ponto importante a ser enfatizado é o grau com que os *front-ends* existentes em reconhecedores modernos têm evoluído para otimizar a subsequente busca por padrões estatísticos. Na descrição feita acima, por exemplo, a compressão logarítmica, aplicação da DCT (*Discrete Cosine Transform*) e o cálculo dos coeficientes delta (1ª e 2ª ordem) foram todos introduzidos primeiramente para satisfazer as suposições requeridas pelo componente de modelagem acústica.

### 2.1.2. Modelagem Acústica

O propósito da modelagem acústica é prover um método que calcule a verossimilhança de qualquer seqüência de vetores  $X$ , dada uma seqüência de palavras  $W$ . A princípio, a distribuição de probabilidade requerida,  $P(X | W)$ , pode ser modelada através de inúmeras palavras e o cálculo estatístico de seqüências de vetores correspondentes. No entanto, esse método é impraticável para sistemas com amplo vocabulário e, ao invés disso, as palavras modeladas pelo sistema são decompostas em seus respectivos fonemas.

Cada fonema é representado por um HMM de primeira ordem que contém, tipicamente, três estados emissores e uma topologia simples do tipo esquerda-direita (*left-right*), conforme ilustrado na Figura 6. Estados de entrada e saída, não emissores, são acrescentados à modelagem para facilitar a união entre modelos. O estado de saída do modelo de um fonema pode ser unido com o estado de entrada de outro para criar um HMM composto. Isto permite que modelos de fonema sejam unidos para formar palavras e estas unidas para formar frases completas. No entanto, o modelo de pausa, por ser estacionário, é representado normalmente por uma topologia mais simples, constituída apenas de um estado emissor de saída.

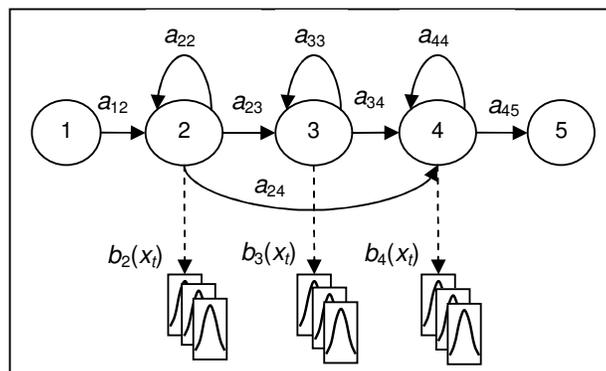


Figura 6: Modelo de fonema baseado em HMM

Um HMM pode ser facilmente entendido como um gerador de seqüências de vetores. É uma máquina de estados finita que modifica seu estado a cada unidade de tempo e, a cada tempo  $t$  e estado  $j$ , um vetor acústico de fala  $x_t$  é gerado com densidade de probabilidade  $b_j(x_t)$ . Além disso, a transição de um estado  $i$  para um estado  $j$  também é probabilística e governada por uma probabilidade discreta  $a_{ij}$ .

A função de probabilidade conjunta de uma seqüência de vetores  $X$  e uma seqüência de estados  $S$ , dado um modelo  $M$ , é calculada pelo produto das probabilidades de transição de estados com as probabilidades de emissão de saída. Assim, a probabilidade conjunta de uma seqüência de vetores acústicos  $X$  e uma seqüência de estados  $S=s(1),s(2),s(3),\dots,s(T)$  é

$$P(X, S | M) = a_{s(0)s(1)} \prod_{t=1}^T b_{s(t)}(x_t) a_{s(t)s(t+1)} \quad (4)$$

onde o estado inicial  $s(0)$  é restringido ao estado de entrada e  $s(T+1)$  é restringido ao estado de saída. Na prática, apenas a seqüência observada  $X$  é conhecida, enquanto a suposta seqüência  $S$  fica escondida pelo modelo. Por isto a técnica é denominada Modelos Escondidos de Markov. A probabilidade  $P(X | M)$ , necessária para resolver a Equação (1), é facilmente encontrada somando a Equação (4) sobre todas as possibilidades de seqüências de estado:

$$P(X | M) = \sum_S P(X, S | M) \quad (5)$$

Um método recursivo e eficiente para realizar este cálculo é o algoritmo avanço-retorno (*Forward-Backward*). Uma característica crucial desse algoritmo é que ele também permite o cálculo da probabilidade de se estar num estado específico do modelo em um instante específico de tempo. Isso leva a um procedimento simples e eficiente conhecido como algoritmo *Baum-Welch* [15], o qual procura estimativas de máxima verossimilhança para o conjunto de parâmetros de cada HMM. É importante ressaltar que a existência do algoritmo *Baum-Welch* foi um dos fatores chave que tornaram o HMM a tecnologia dominante para modelagem acústica.

Alternativamente,  $P(X | M)$  pode ser aproximada procurando-se uma seqüência de estados que maximize a Equação (4). Novamente, existe um simples algoritmo que eficientemente realiza esse cálculo. Chama-se algoritmo de *Viterbi* e, como será discutido adiante, esse algoritmo é importante na decodificação onde a determinação da seqüência de estados mais provável é a chave para o reconhecimento de uma seqüência de palavras desconhecidas.

Os fundamentos descritos acima foram estudados profundamente por anos. No entanto, foi apenas na década de 90 que foram desenvolvidos métodos que permitiriam que modelos de fonemas baseados em HMM apresentassem a discriminação acústica necessária para reconhecimento de fala contínua com amplo vocabulário e independência de locutor.

### 2.1.2.1. Modelagem Contínua de Trifones

Por questões de desempenho e proteção contra um eventual *underflow* no cálculo de probabilidades, a Equação (4) é usualmente reescrita num formato logarítmico:

$$\log(P(X, S | M)) = \sum_{t=0}^T \log(a_{s(t)s(t+1)}) + \sum_{t=1}^T \log(b_{s(t)}(x_t)) \quad (6)$$

As probabilidades de transição  $a_{s(t)s(t+1)}$  modelam a estrutura temporal da fala e a escolha da função de probabilidade de saída,  $b_{s(t)}(x_t)$ , é crucial já que ela deve modelar toda variabilidade espectral intrínseca na fala real, tanto de um mesmo locutor como de locutores distintos.

Sistemas menos complexos, baseados em HMM, utilizam funções de probabilidade de saídas discretas em conjunto com um quantizador vetorial. Nestes sistemas, cada vetor acústico recebido é trocado pelo índice do vetor mais próximo de um *codebook* pré-computado e a função de probabilidade de saída consiste apenas da busca feita em uma tabela de memória (*look-up table*) que contem as probabilidades de saída de cada possível índice dentre os vetores quantizados (VQ – *Vector Quantized*). Esse método é computacionalmente muito eficiente, no entanto, a quantização introduz um ruído que limita a precisão que pode ser obtida. Portanto, sistemas mais eficientes utilizam distribuições de saída parametrizadas por densidades contínuas que modelam diretamente os vetores acústicos [16], [17], [18]. A distribuição normalmente escolhida é a mistura de  $M$  Gaussianas multivariadas:

$$b_j(x_t) = \sum_{m=1}^M c_{jm} \mathcal{N}(x_t, \mu_{jm}, \Sigma_{jm}) \quad (7)$$

onde  $c_{jm}$  é o peso do componente  $m$  da mistura, no estado  $j$ , e  $\mathcal{N}(x, \mu, \Sigma)$  denota uma Gaussiana multivariada de média  $\mu$  e covariância  $\Sigma$ ,

$$\mathcal{N}(x, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} e^{-\frac{1}{2}(x-\mu)'\Sigma^{-1}(x-\mu)} \quad (8)$$

Até agora, assumiu-se de forma implícita ser necessário apenas um HMM por fonema e, como para o português são requeridos aproximadamente 40 fonemas, poderia se pensar que apenas 40 HMMs devem ser treinados. Na prática, efeitos contextuais causam uma larga variação na maneira como sons diferentes são produzidos (chamam-se de alofones os fonemas existentes em diferentes contextos). Para se conseguir uma boa discriminação fonética, distintos HMMs devem ser treinados para cada um dos diferentes contextos de um fonema. A maneira usual de se resolver esse problema é a utilização de trifones, onde cada um dos fonemas possui modelos distintos de HMM para os pares formados por fonemas situados à direita e a esquerda. Por exemplo, suponha que a notação  $x-y+z$  representa o fonema  $y$  ocorrendo após o fonema  $x$  e antes do fonema  $z$ . A frase “Muito Prazer!” poderia ser representada pela seqüência fonética “*sil m u j~ t u p r a z e X sil*” e, para o caso de trifones, a seqüência poderia ser transcrita como:

*sil sil-m+u m-u+j~ u-j~+t j~-t+u t-u+p u-p+r p-r+a r-a+z a-z+e z-e+X e-X+sil sil*

Note que os contextos de trifones abrangem as fronteiras entre palavras e as duas instancias do fonema “u” devem ser representadas por diferentes HMMs, pois seus contextos são diferentes. Este uso dos chamados trifones entre-palavras (*cross-word triphones*) resulta em uma modelagem mais precisa, no entanto leva a complicações no decodificador como será visto adiante. Sistemas mais simples resultam do uso de trifones intrapalavra (*word-internal triphones*), onde o exemplo acima se tornaria:

*sil m+u m-u+j~ u-j~+t j~+t+u t-u p+r p-r+a r-a+z a-z+e z-e+X e-X sil*

Aqui, menos modelos distintos são necessários, simplificando tanto o problema de estimação de parâmetros como o projeto do decodificador. No entanto, o custo é a perda de habilidade em se modelar efeitos contextuais nas fronteiras de palavras e, consideravelmente, na modelagem de fala fluente.

O uso de misturas de Gaussianas nas distribuições de saída permite que as distribuições de cada estado sejam modeladas com muita precisão. No entanto, quando trifones são usados, o sistema resultante requer o treino de muitos parâmetros. Por exemplo, um sistema com trifones entre-palavras e grande vocabulário irá tipicamente precisar de cerca de 60.000 trifones<sup>1</sup>. Na prática, reporta-se que com cerca de 10 componentes de mistura se obtém bom desempenho em sistemas LVR [19]. Assumindo que as covariâncias são todas diagonais, um sistema que, por exemplo, utilize 39 elementos em vetores acústicos, requer um conjunto de cerca de 790 parâmetros por estado. Assim, 60.000 modelos de trifones, cada um com 3 estados, resultariam num total de 142 milhões de parâmetros.

O problema de muitos parâmetros e poucos dados de treino é absolutamente crucial no projeto de um reconhecedor de fala estatístico. Uma tentativa de se resolver este problema foi através da união de todos os componentes das Gaussianas em um mesmo conjunto compartilhado por todos os estados HMM. Nesses sistemas, conhecidos como semi-contínuos ou como união de misturas (*tied-mixture*), apenas os pesos dos componentes da mistura são específicos para cada estado e estes podem ser suavizados por processos de interpolação [20], [21], [22]:

$$b_j(x_t) = \sum_{m=1}^M c_{jm} \mathcal{N}(x_t, \mu_m, \Sigma_m) \quad (9)$$

onde, diferente da Equação (7), a distribuição de probabilidade  $\mathcal{N}(x, \mu, \Sigma)$  independe do estado  $j$ . Comparações entre HMMs de densidade discreta, semi-contínua e contínua apontaram, num primeiro momento, os semi-contínuos como superiores [23]. Todavia, técnicas de suavização [24] contribuíram para que os sistemas de densidade contínua fossem popularizados. Em particular, união de estados (*state-tying*) [25], [26] e união de componentes (*component-tying*) baseados em fonemas [27] foram bastante estudados e utilizados. O uso destas técnicas de união, combinadas aos HMMs de densidade contínua, levou a ganhos substanciais na precisão da modelagem.

A idéia da técnica *state-tying* é unir os estados que são acusticamente indistinguíveis (chamados comumente de *senones*). Isso permite que os dados associados a cada estado individualmente, sejam combinados num recurso em comum e, assim, estimados de forma mais robusta. Isso está ilustrado na Figura 7. No topo da figura, cada trifone tem sua própria distribuição de saída. Após as uniões, vários estados passam a compartilhar as mesmas distribuições.

<sup>1</sup> Considera-se aqui o idioma Inglês, modelado com 45 fonemas e  $45^3=91125$  trifones possíveis, porém nem todos podem ocorrer devido às restrições fonéticas da linguagem.

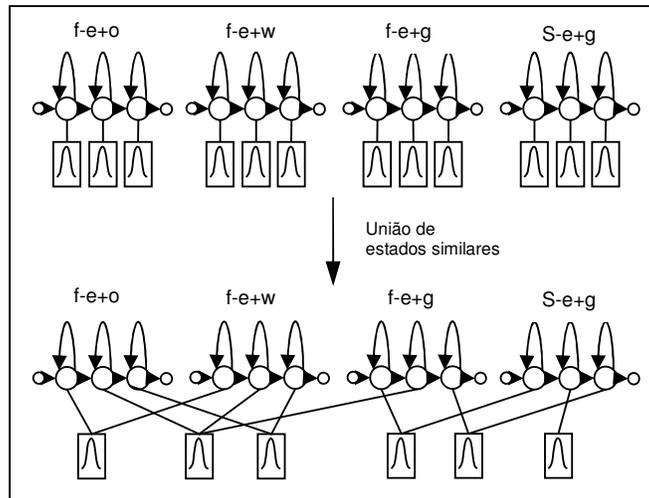


Figura 7: União de estados acusticamente indistinguíveis

A escolha sobre quais estados devem ser unidos pode ser realizada através de árvores de decisão fonética [28], [29], [30]. Este método envolve a construção de uma árvore binária para cada fonema e posição de estado. Cada árvore tem, em cada um dos nós, uma pergunta fonética do tipo sim/não, como “O contexto á esquerda é nasal?”. Inicialmente, todos os estados de um dado fonema são posicionados no nó raiz da árvore. De acordo com as respostas, o conjunto de estados é sucessivamente dividido até que os estados tenham alcançado os nós terminais da árvore. Todos os estados depositados em um mesmo nó terminal são então unidos. Por exemplo, a Figura 8 ilustra o caso em que se unem todos os estados centrais de todos os trífones do fonema /E/ (como em “céu”). Todos os estados são conduzidos árvore abaixo e, dependendo da resposta às questões, eles terminam em um dos nós terminais escurecidos. Por exemplo, no caso ilustrado, o estado central de “s-E+u” irá se juntar ao primeiro nó terminal já que o seu contexto a direita não é uma consoante-central e o seu contexto a esquerda não é nasal.

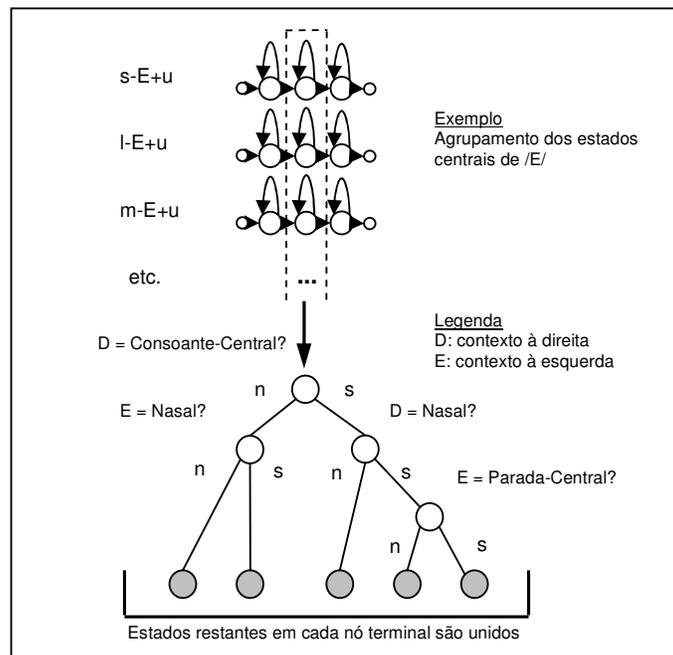


Figura 8: Agrupamento baseado em árvores de decisão

As questões em cada nó são escolhidas para maximizar a verossimilhança entre os dados de treino e o conjunto resultante da união de estados. Na prática, as árvores de decisão fonética resultam em agrupamentos de estados compactos e de boa qualidade, os quais possuem dados suficientes para estimar de forma robusta as misturas de Gaussianas das funções de probabilidade de saída. Além disso, elas podem ser usadas para sintetizar um HMM para qualquer possível contexto, apareça ele nos dados treino, ou não, simplesmente descendo na árvore e usando as distribuições de estado associadas aos nós terminais. Finalmente, árvores de decisão fonéticas podem ser usadas para incluir mais que simples contextos de trifone. Pode-se, por exemplo, usar perguntas que envolvam  $\pm 2$  fonemas (fonemas vizinhos dos vizinhos), além de levar em conta a presença de fronteiras entre palavras.

### 2.1.3. Modelo de Linguagem

O propósito do modelo de linguagem (LM – *Language Model*) é prover um mecanismo que estime a probabilidade de uma palavra  $w_k$ , em uma sentença, em função das palavras que a antecedam, respectivamente,  $w_1, w_2, \dots, w_{k-1}$ . Esta probabilidade, representada por  $P(W)$  e definida na Equação (1), é essencial para se obter bons resultados em sistemas CSR:

$$P(W) = P(W_1^k) = P(w_1, \dots, w_k) \quad (10)$$

onde reescreve-se  $P(W)$  como  $P(W_1^k)$ , acrescentando os índices 1 e  $k$  de forma a representar uma seqüência com  $k$  palavras. A probabilidade conjunta de ocorrência de palavras da Equação (10) pode ser substituída por um produto de suas probabilidades condicionais da seguinte forma:

$$\begin{aligned} P(W_1^k) &= P(w_1)P(w_2 | w_1)P(w_3 | w_1, w_2) \dots P(w_k | w_1, \dots, w_{k-1}) \\ &= P(w_1) \prod_{i=2}^k P(w_i | w_1, \dots, w_{i-1}). \end{aligned} \quad (11)$$

Uma maneira simples, porém efetiva, de se obter estas probabilidades é com a utilização de  $n$ -gramas, onde assume-se que  $w_k$  depende apenas das  $n-1$  palavras anteriores a ela, na seqüência:

$$P(W_1^k) \approx P(w_1) \prod_{i=2}^k P(w_i | W_{i-n+1}^{i-1}). \quad (12)$$

Na tentativa de capturar a correlação existente entre palavras vizinhas, os  $n$ -gramas acabam simultaneamente absorvendo sintaxe, semântica e pragmática existente nas frases observadas. Isso os faz extremamente efetivos em idiomas como inglês ou português onde a ordem das palavras é importante visto que os efeitos contextuais mais fortes normalmente vêm dos vizinhos mais próximos. Além disso, as distribuições de probabilidade dos  $n$ -gramas podem ser computadas diretamente de textos prontos e, portanto, não há um requerimento de se definir regras explícitas de lingüística como uma gramática formal do idioma.

A princípio, os  $n$ -gramas podem ser estimados através de simples contadores de freqüência e armazenados em uma tabela de memória (*look-up table*). Por exemplo, para o caso de trigramas ( $n=3$ ),

$$P(w_k | w_{k-2}, w_{k-1}) = \frac{t(w_{k-2}, w_{k-1}, w_k)}{b(w_{k-2}, w_{k-1})}. \quad (13)$$

onde  $t(a,b,c)$  é o número de vezes que o trigrama  $a,b,c$  aparece nos dados de treino e  $b(a,b)$  é o número de vezes que o bigrama  $a,b$  aparece. Nas Equações (11) e (12),  $P(w_1)$  é calculado de forma conveniente para lidar com a condição inicial, de forma que normalmente, associa-se uma nova palavra,  $w_0$ , a um símbolo de início de frase, e prossegue-se da forma indicada pela Equação (13). Pode-se, claramente, perceber o problema de que, para um vocabulário com  $V$  palavras, existem  $V^3$  potenciais trigramas. Mesmo para um vocabulário restrito a 10.000 palavras ( $10^{12}$  possíveis trigramas), este número é extremamente grande. O fato de que muitos dos possíveis trigramas não aparecerão nos dados de treino e muitos outros aparecerão apenas uma ou duas vezes, faz com que a estimativa da Equação (13) seja muito pobre. Em resumo, há um problema agudo de escassez de dados.

Uma das soluções de treinamento projetada como solução à escassez de dados é o uso de uma combinação de técnicas de desconto (*discounting*) e retrocesso (*backing-off*) [31], [32]. Desconto significa que as contagens de trigramas que ocorrem com maior frequência são reduzidas e a massa probabilística resultante em excesso é redistribuída entre os trigramas que ocorrem com menos frequência, ou mesmo não ocorram. O retrocesso é aplicado quando há pouquíssimos trigramas para formar qualquer tipo de estimativa (ex.: apenas uma ou duas ocorrências nos dados de treino). Ele envolve a substituição da probabilidade de um trigrama por uma probabilidade de bigrama escalada, que é

$$P(w_k | w_{k-2}, w_{k-1}) = B(w_{k-2}, w_{k-1})P(w_k | w_{k-1}). \quad (14)$$

onde  $B(\cdot)$  é um fator de escalamento, calculado de forma a garantir que a distribuição de probabilidades condicionais some um:

$$\sum_{w_{k-2}, w_{k-1}} P(w_k | w_{k-2}, w_{k-1}) = 1. \quad (15)$$

Ainda que uma estimativa robusta das probabilidades de trigramas requeira um cuidado considerável, os problemas decorrentes são solucionáveis e um bom desempenho tem sido obtido. A técnica de  $n$ -gramas possui deficiências óbvias que resultam da sua inabilidade de explorar restrições de maior amplitude como a concordância entre sujeito e verbo de uma frase [33]. Como conseqüência, várias alternativas foram estudadas como modelos baseados em árvore [34], modelos em treliça [35], modelos com gatilhos [36], modelos com histórico [37] e  $n$ -gramas variáveis [38]. Entretanto, em geral, todas estas tentativas resultaram em apenas pequenos ganhos de desempenho sobre um considerável custo computacional. Sendo assim, após quase duas décadas de pesquisa, os modelos de linguagem conhecidos como bigramas e trigramas ainda dominam os sistemas LVR. Nestes sistemas, têm se usado, para modelagem da linguagem, bases de dados de 1 milhão a 500 milhões de palavras, correspondendo a vocabulários de 1 mil a 267 mil palavras distintas, para construção dos modelos trigrama [2].

Sistemas CSR menos complexos, que utilizam vocabulários menores e mais restritivos, utilizam modelos de linguagem determinísticos que se baseiam em estruturas bem definidas de dados (ou conjunto de regras) como o *Context-Free Grammar* (CFG) e o *Recursive Transition Network* (RTN) [39]. Nestes modelos,

apenas determinadas seqüências de palavras são aceitas, de forma que  $P(W)$ , retorna apenas 1 ou 0, indicando a aceitação ou não, da seqüência, pelo modelo.

Devido ao seu formalismo, os CFGs têm uma aplicação natural para interpretação de linguagem natural, enquanto os modelos baseados em  $n$ -gramas funcionam muito bem em aplicações onde não se tenha um conhecimento prévio do domínio<sup>2</sup>. Os CFGs, são restritivos o bastante para se conseguir eficientes manipuladores (*parsers*) da linguagem e analisadores sintáticos. O problema é que a gramática nunca é completa o suficiente, de forma que um sistema CSR baseado em CFG é bom apenas quando se sabe a estrutura das sentenças que se deve falar, diminuindo assim a usabilidade do mesmo. Por outro lado, o modelo de  $n$ -gramas é treinado estatisticamente com uma grande quantidade de dados, de forma que a dependência entre palavras próximas, capturada pelo modelo, pode acomodar simultaneamente estruturas sintáticas e semânticas da linguagem, muita das quais, dificilmente poderiam ser descritas por regras.

### 2.1.3.1. Perplexidade de um Modelo de Linguagem

Os modelos de linguagem tendem a minorar as incertezas (diminuir a entropia) do conteúdo das sentenças e facilitar o reconhecimento. Por exemplo, se existem em média poucas palavras que podem seguir uma dada palavra em um LM, o sistema de reconhecimento terá menos opções para verificar e o desempenho será melhor do que se existissem muitas palavras possíveis. Este exemplo sugere que uma forma conveniente de medir a dificuldade imposta pelo LM no processo de busca deve envolver o número médio das palavras que possam seguir outras. Se o LM for visto como um grafo, com terminais associados a transições entre palavras, por exemplo, então esta medida estaria relacionada com o fator de ramificação médio em todos os pontos de decisão do grafo. De forma simplificada, esta é a quantidade medida pela perplexidade, formalizada a seguir.

Consideremos uma seqüência de palavras  $W = \{w_1 w_2 w_3 \dots w_N\} = W_1^N$  como um processo randômico. A entropia estimada deste processo é definida como [40]:

$$\hat{H}(W) = - \lim_{N \rightarrow \infty} \frac{1}{N} \log_2 P(W_1^N) \quad (16)$$

onde  $P(W_1^N)$  é a probabilidade da seqüência  $W_1^N$  estimada pelo modelo de linguagem. O cálculo anterior considera  $W_1^N$  como um processo ergódico. Sendo  $W_1^N$  uma seqüência de palavras suficientemente longa, podemos aproximar (16) como:

$$\hat{H}(W) \approx - \frac{1}{N} \log_2 P(W_1^N) \quad (17)$$

A perplexidade  $PP(W)$  de um LM  $P(W)$  é definida como o recíproco da probabilidade média (geométrica) associada pelo modelo a cada palavra no conjunto de teste  $W_1^N$ . Esta é uma medida relacionada à entropia e é conhecida como perplexidade do conjunto de teste:

<sup>2</sup> Entende-se como domínio, a especificidade de uma área em questão, ou seja, a diversidade de palavras referentes ao tema e como estas podem ser combinadas.

$$PP(W) = \frac{1}{\sqrt[N]{P(W)}} \approx 2^{\hat{H}(W)} \quad (18)$$

A perplexidade pode ser interpretada como a média geométrica do fator de ramificação do texto quando apresentado ao LM. A perplexidade definida pela Equação (18) tem dois parâmetros-chave: o LM e a seqüência de palavras  $W$ . A perplexidade do conjunto de teste<sup>3</sup> avalia a capacidade de generalização do LM. Pode-se dizer que uma baixa perplexidade está relacionada a um possível melhor desempenho do reconhecedor. Isto acontece devido a perplexidade ser essencialmente uma medida estatisticamente pesada do fator de ramificação do conjunto de treino. Quanto maior a perplexidade, mais ramos o reconhecedor de fala precisará considerar estatisticamente.

A perplexidade de um LM é função do domínio e tamanho do vocabulário utilizado. Perplexidades típicas alcançadas com modelos  $n$ -gram em textos na língua Inglesa variam de 50 a 1000, dependendo do tipo de texto. Na Tabela 1, tem-se um quadro comparativo para diversas aplicações [2]:

Tabela 1: Perplexidades típicas para vários domínios [2].

Domínio	Perplexidade
Radiologia	20
Medicina de emergência	60
Jornalismo	105
Fala geral	247

Como será visto na Seção 2.2, a medida mais comum de avaliação de sistemas CSR é a taxa de palavras erradas ( $WER$  - *Word Error Rate*). Modelos de linguagem podem ser avaliados, mantendo-se o modelo acústico fixo, e observando-se como as diferentes técnicas impactam a  $WER$ . Contudo, essa estratégia possui um custo computacional alto, sendo comum a utilização da perplexidade nos estágios iniciais do desenvolvimento destes modelos. Isso é justificado pela forte correlação entre  $WER$  e  $PP(W)$  como indica a expressão [40]:

$$WER \approx -12.37 + 6.48 \log_2 (PP(W)) \quad (19)$$

A Tabela 2 a seguir, mostra o relacionamento entre tamanho do vocabulário, perplexidade e  $WER$  para diferentes tarefas de reconhecimento de fala contínua.

Tabela 2: Perplexidades e Taxas de Erro ( $WER$ ) associadas a diferentes tarefas [40].

Corpus	Vocabulário	Perplexidade	$WER$
<i>TI Digits</i>	11	11	~0.0%
<i>OGI Alphadigits</i>	36	36	8%
<i>Resource Management (RM)</i>	1.000	60	4%
<i>Air Travel Information Service (ATIS)</i>	1.800	12	4%
<i>Wall Street Journal</i>	20000	200-250	15%
<i>Broadcast News</i>	> 80.000	200-250	20%
<i>Conversational Speech</i>	> 50.000	100-150	30%

<sup>3</sup> Entende-se por conjunto de teste, as seqüências de palavras que não tenham sido utilizadas para derivar o modelo de linguagem.

#### 2.1.4. Decodificador

Nas seções anteriores foram descritos os principais componentes de um sistema LVR: extração de parâmetros, modelo acústico e modelo de linguagem. Para que se realize o reconhecimento de voz usando estes componentes, é realizada uma busca em todas as possíveis seqüências de palavras  $W$ , maximizando  $P(W | X)$ , conforme descrito em (1). Este é um problema de busca e sua solução é a tarefa do decodificador. O termo decodificador, no contexto de reconhecimento de voz, foi criado numa analogia a terminologia usada nos métodos que utilizam estados finitos para decodificação [41] no campo da teoria da informação.

A decodificação é um processo de busca no qual uma seqüência de vetores correspondentes a características acústicas do sinal de voz é comparada com modelos de palavras. De uma maneira geral, o sinal de voz e suas transformações não fornecem uma indicação clara das fronteiras entre palavras nem do número total de palavras em uma dada locução, de modo que a determinação destas é parte do processo de decodificação. Neste processo, todos os modelos das palavras (formadas por seus respectivos modelos de fonemas) são comparados com uma seqüência de vetores acústicos. O número de modelos cresce com o vocabulário, e pode gerar espaços de busca grandes, o que torna o processo de busca oneroso em termos computacionais, e portanto lento. Em geral, esta etapa do reconhecimento, nos sistemas modernos, é responsável por praticamente todo o esforço computacional no reconhecimento de fala contínua e, portanto, é a que determina a velocidade final desses sistemas.

Durante o processo de maximização da Equação (1), repetida por conveniência a seguir, o termo  $P(X | W)$  é expandido em função do modelo acústico, vinculando os estados dos HMMs à emissão de saídas nos mesmos. Dessa forma,  $P(X | W)$  passa a ser calculado como a soma de todas as possibilidades de transições entre as possíveis seqüências de estados do modelo sob hipótese:

$$\begin{aligned} \hat{W}_1^N &= \arg \max_{W_1^N} \left\{ P(W_1^N) P(X_1^T | W_1^N) \right\} \\ &= \arg \max_{W_1^N} \left\{ P(W_1^N) \sum_{S_1^T} P(X_1^T, S_1^T | W_1^N) \right\} \\ &\approx \arg \max_{W_1^N} \left\{ P(W_1^N) \max_{S_1^T} (P(X_1^T, S_1^T | W_1^N)) \right\} \end{aligned} \quad (20)$$

onde  $W_1^N = w_1 \dots w_N$  representa a hipótese de seqüência de palavras (compostas por seus respectivos HMMs de subpalavras),  $S_1^T = \{s_1 \dots s_T\}$ , a hipótese de seqüência de estados dentro do modelo e  $X_1^T = \{x_1 \dots x_T\}$ , os vetores acústicos observados. O somatório nesta equação é então substituído por uma maximização, num processo referido como Aproximação de Viterbi [9]. Ao invés de somar sobre todos os caminhos, consideramos apenas o caminho mais provável.

Neste processo de maximização, o espaço de busca pode ser descrito como uma rede onde se busca o melhor alinhamento temporal entre a seqüência de entrada e os estados dos modelos. A busca pode ser realizada em dois níveis: no nível de estados ( $S_1^T$ ) e no nível de palavras ( $W_1^N$ ). É possível recombina eficientemente as hipóteses

nos dois níveis usando programação dinâmica (DP - *Dynamic Programming* [42]), limitando a explosão combinatória do número de hipóteses de busca.

Estratégias de busca baseadas em DP são atualmente usadas com sucesso num grande número de tarefas de reconhecimento de voz, as quais variam, desde o reconhecimento de seqüências de dígitos até sistemas LVR, que aceitam entradas de fala sem nenhum tipo de restrição. Muitas variantes da busca baseada em DP já eram conhecidas na década de 70 [43], [44], [45], [46], [47], [48]. Nas três décadas passadas, estas e outras estratégias relacionadas à DP se tornaram surpreendentemente eficazes em lidar com vocabulários de 20 k palavras ou mais. Ainda assim, até uma década atrás, ainda era muito controverso, entre os mais experientes no assunto, se a enorme complexidade inerente de sistemas LVR poderia ser resolvida por DP.

Os decodificadores, citados nesta seção, serão abordados com mais detalhes na Seção 3. Entretanto, algumas estratégias principais adotadas ao longo dos anos em sistemas LVR foram definidas no Apêndice 1, de forma a delinear a complexidade do problema e as diversas formas como estes sistemas vêm sendo projetados:

## **2.2. Estado da Arte**

As opiniões e estratégias seguidas pelos sistemas de reconhecimento de voz modernos se distinguem bastante do que se imaginava há três décadas, principalmente, em relação à modelagem acústico-fonética e aos processos de busca. Naquela época, um bom número de estudiosos predizia que um considerável progresso poderia ser alcançado, caso “técnicas primitivas” como reconhecimento por padrões estatísticos e métodos como a busca em feixe (abordada em detalhe na Seção 3) fossem descartados. No entanto, a experiência obtida nas últimas décadas, mostra que o julgamento feito por Klatt sobre os princípios dos sistemas DRAGON e HARPY, desenvolvidos em 1976, é mais verdadeiro hoje do que antes [49]:

*“It might seem to someone versed in the intricacies of phonology and the acoustic-phonetic characteristics of speech that a search of a graph of expected acoustic segments is a naïve and foolish technique to use to decode a sentence. In fact, such a graph and search strategy (and probably a number of other simple models) can be constructed and made to work very well indeed if the proper acoustic-phonetic details are embodied in the structure.”*

A última década tem testemunhado um progresso significativo na tecnologia de reconhecimento de fala. O progresso substancial obtido na tecnologia básica levou os sistemas modernos a vencerem as barreiras de independência de locutor, fala contínua e vocabulários extensos. Existem vários fatores que contribuíram para este rápido progresso:

- A chegada da era do HMM em contrapartida a métodos determinísticos (Ex. DTW - *Dynamic Time Warping*).
- Foi feito um grande esforço no sentido de desenvolver grandes bases de dados de fala para o treinamento e avaliação de sistemas.
- Estabelecimento de normas de avaliação de desempenho. Até uma década atrás, os pesquisadores treinavam e testavam seus sistemas usando dados coletados localmente, e não foram muito cuidadosos em delinear os conjuntos

de treinamento e testes. Conseqüentemente, era muito difícil comparar o desempenho dos vários sistemas, e ainda, o desempenho de um sistema era geralmente degradado quando este era apresentado a dados novos. A recente disponibilidade de grandes bases de dados no domínio público para alguns idiomas (inglês, espanhol, francês, etc.), associada à especificação de padrões de avaliação, resultou em uma documentação uniforme de resultados de testes, contribuindo para uma maior confiabilidade na monitoração dos progressos alcançados.

- Os avanços na tecnologia dos computadores influenciaram indiretamente o progresso nesta área. A disponibilidade de computadores mais rápidos e com grandes capacidades de memória permitiu aos pesquisadores realizar várias experiências em larga escala e em um curto espaço de tempo. Isto significa que o intervalo de tempo entre uma idéia e a sua implementação e avaliação foi bastante reduzido. De fato, sistemas de reconhecimento de fala com desempenho razoável podem rodar em microcomputadores comuns em tempo real, sem hardware adicional, um fato inimaginável há alguns anos.

Atualmente, a precisão de sistemas CSR é tipicamente descrita pela taxa de erros no reconhecimento de palavras (*WER* - *Word Error Rate*), definida como [50]:

$$WER = \frac{S + I + D}{N} \quad (21)$$

onde *N* é o número total de palavras na seqüência de teste e *S*, *I* e *D* são, respectivamente, o número total de erros por substituição (*substitution*), inserção (*insertion*) e supressão (*deletion*) na seqüência reconhecida. Correspondentemente, a taxa de reconhecimento de palavras (*WRR* - *Word Recognition Rate*) é calculada por:

$$WRR = 1 - WER \quad (22)$$

Talvez a tarefa mais popular, e potencialmente mais útil, com baixa perplexidade (*PP* = 11) é o reconhecimento de dígitos conectados. Para o inglês americano, o reconhecimento independente de locutor de seqüências de dígitos pronunciados de forma contínua e restringidos ao comprimento de números telefônicos pode alcançar uma taxa de erro de 0,3% quando o número de dígitos numa seqüência é conhecido.

Uma das tarefas de média perplexidade mais conhecidas é utilizando uma base de mil palavras chamada *Resource Management*, na qual se faz indagações sobre vários navios no oceano Pacífico. O melhor desempenho independente de locutor nesta tarefa é de menos de 4% (*WER*), usando um LM de pares de palavras que limita as palavras possíveis que seguem a uma dada palavra (*PP* = 60).

Quanto à questão do reconhecimento de fala espontânea, no domínio do Serviço de Informação de Viagens Aéreas (*Air Travel Information Service*, *ATIS*), por exemplo, foram relatadas [4] taxas de erros de menos de 3% para um vocabulário de aproximadamente duas mil palavras e um LM do tipo bigrama com uma perplexidade por volta de 15.

Tarefas com alta perplexidade, com vocabulários de milhares de palavras, são destinadas principalmente para aplicações de ditado. Depois de trabalhar em sistemas de palavras isoladas e dependentes de locutor por muitos anos, a comunidade tem voltado suas atenções desde 1992 para o reconhecimento de fala contínua para grandes vocabulários (20k palavras ou mais), alta perplexidade (*PP* » 200) e

independência de locutor (Figura 9). Em comparações realizados no ano de 1997, o melhor sistema apresentou uma taxa de erro de 9,9% nos testes realizados regularmente nos EUA através do Departamento de Defesa [4].

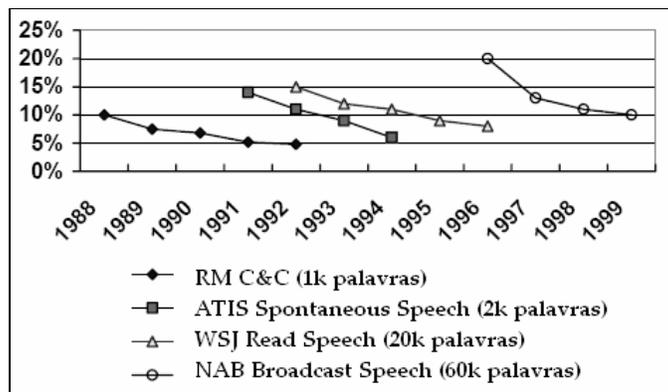


Figura 9: Histórico de taxas de erro nas avaliações do DARPA realizadas entre 1988 a 1999 [4].

O Grupo de Fala do NIST (*National Institute of Standards and Technology*) [51], nos EUA, realiza um trabalho de coordenação e centralização dos avanços obtidos na área de processamento de fala através das diversas comunidades envolvidas. Além de implantarem padrões para a comparação de sistemas, fomentam a discussão e troca de experiências e realizam uma série de testes de comparação de desempenho entre os sistemas que implementam o estado da arte. Estes testes são divididos entre diversas tarefas de avaliação, algumas das quais são descritas no Apêndice 2.

### 2.3. Softwares de Apoio à Implementação de Sistemas CSR

Nesta seção estão listados, com uma breve descrição, os principais sistemas abertos, direcionados a pesquisa em reconhecimento de voz e que implementam o estado da arte na área. Estes sistemas são freqüentemente utilizados nas avaliações do DARPA, além de serem o ponto de partida para a maioria dos pesquisadores da área e de áreas afins:

#### CUED HTK

O HTK (*Hidden Markov Model Toolkit*) [13] é definido como um conjunto portátil de ferramentas de software para construção e manipulação de HMMs. Teve sua primeira versão desenvolvida em 1989 pelo professor Steve Young do Departamento de Engenharia da Universidade de Cambridge (CUED - *Cambridge University Engineering Department*). É usado principalmente para pesquisa em reconhecimento de voz, porém é também bastante utilizado em outras aplicações, como pesquisas em síntese de voz, reconhecimento de caracteres e seqüenciamento de DNA. O HTK é uma das ferramentas mais utilizadas e divulgadas da área. Consiste em conjunto de bibliotecas e ferramentas com código-fonte disponíveis em C, porém compatíveis com C++. As ferramentas contidas nele provêm sofisticadas funcionalidades para análise da fala, treino de HMMs, testes e análise de resultados. O software suporta tanto misturas de Gaussianas com densidade contínua, bem como distribuições discretas e pode ser utilizado para construir sistemas complexos baseados em HMMs. A versão disponível do HTK é a 3.3, liberada em 25/07/2005. Contém extensiva documentação e exemplos em um documento conhecido como "HTK Book", bem como um vasto fórum de discussão disponível no sítio [13]. Como desvantagens do HTK, podem-se

apontar o fato dele não suportar modelos de linguagem baseados em trígama (suporta apenas o treino) e ter sua licença restrita apenas ao uso não comercial.

## **ATK**

O ATK (API for HTK) [14] é uma interface de programação para o HTK e foi criado para facilitar o desenvolvimento de aplicações experimentais baseadas no mesmo. Consiste em uma camada de programação orientada a objeto, em C++, que faz acesso às bibliotecas padrão do HTK. Isso permite que novos reconhedores construídos a partir de versões customizadas do HTK sejam compilados com o ATK e, então, testados em sistemas já em funcionamento. Assim como o HTK, o ATK é compatível com as plataformas de sistemas operacionais Unix® e Microsoft Windows® (através de linha de comando e, apenas no Windows, disponibiliza interface gráfica). O ATK roda com múltiplos processos em paralelo (*multi-threaded*) e permite que múltiplas entradas de comandos (por voz, clique do mouse, gestos, etc.) sejam combinadas em um mesmo pacote de dados. O ATK pode retornar os resultados de reconhecimento de voz palavra-a-palavra, assim que são reconhecidas, de forma a reduzir o atraso a um mínimo, permitindo que sistemas de resposta rápida sejam construídos. O decodificador do ATK é uma implementação independente do HTK, com melhor desempenho e suporte a trigramas. Atualmente o ATK está na versão 1.4.1, liberada em 02/06/2004.

## **CMU SPHINX**

O Sphinx (esfinge, em português) [52] foi desenvolvido na Universidade Carnegie Mellon (CMU - *Carnegie Mellon University*) de forma complacente aos projetos Sphinx fundados pelo DARPA e criados de forma a estimular a criação de ferramentas e aplicativos de processamento de fala e fazer avançar o estado da arte em reconhecimento de voz, bem como, nas áreas correlatas, como sistemas de diálogo e síntese de voz. Os termos de licença para as ferramentas e bibliotecas disponíveis no Sphinx são derivados do modelo de distribuição de software da Berkeley (BSD - *Berkeley Software Distribution*). Não há restrição alguma quanto ao uso comercial ou redistribuição. O Sphinx está na versão 4, lançada em 04/06/2004, e implementa o estado da arte para sistemas de reconhecimento de voz baseados em HMMs contínuos. A biblioteca possui um excelente tutorial on-line e sua arquitetura é descrita num documento intitulado "Sphinx-4 Whitepaper". Da versão 3 para a versão atual, o Sphinx foi completamente reescrito e portado para a plataforma de linguagem de programação JAVA®.

## **ISIP Toolkit**

O principal objetivo do pacote de ferramentas do ISIP (*Institute for Signal and Information Processing*) [53], coordenado pelo professor Joe Picone e localizado na Universidade do Estado do Mississippi (MSU - *Mississippi State University*), é disponibilizar um sistema de reconhecimento de voz que implementa o estado da arte, de forma modular e sem custos de licença, o qual pode ser facilmente modificado para satisfazer as necessidades de pesquisadores. O sistema é construído sobre uma vasta hierarquia de classes de propósito geral escritas em C++ e que implementam conceitos matemáticos, de processamento de sinais e estruturas de dados necessárias ao desenvolvimento de sistemas. O ISIP Toolkit está na versão 5.15, lançada em 02/08/04, e possui documentação on-line bastante didática.

### 3. Estudo de Decodificadores e a Estratégia *One-Pass*

Há dois componentes principais no cálculo do custo computacional de um sistema de reconhecimento de voz: computação de probabilidades dos modelos acústicos e busca. No caso de sistemas baseados em HMM, o primeiro componente se refere à probabilidade de um dado estado de um HMM emitir os coeficientes extraídos da fala observada num instante de tempo. O segundo se refere à busca pela melhor seqüência de palavras dado um trecho de fala. O custo computacional da busca não é afetado pela complexidade dos modelos acústicos, e sim determinado pelo tamanho da tarefa (vocabulário / variabilidade fonética). Dessa forma, a eficiência do decodificador passa a ser fator determinante no desempenho de sistemas de reconhecimento com vocabulários médios e grandes.

#### 3.1. Algoritmos de Busca

Algoritmos de busca são usualmente associados e referenciados à área de Inteligência Artificial (AI) e têm sido amplamente estudados e aplicados em sistemas especializados, jogos e no acesso a informações. Na literatura, encontram-se diversos métodos genéricos de busca em grafos, porém, ainda que o conceito básico destes algoritmos seja independente da tarefa onde serão aplicados, a eficiência dos mesmos sempre depende de como exploramos o conhecimento específico ao domínio do problema.

A idéia de busca implica na movimentação, exame dos dados e tomada de decisão sobre se o objeto de busca já foi encontrado ou não. Geralmente, problemas de busca podem ser representados usando o paradigma de busca num espaço de estados. Costuma-se definir o espaço por uma tripla  $(S, O, G)$ , onde  $S$  (*start*) é um conjunto de estados iniciais,  $O$  é um conjunto de operadores ou regras (*operators*) aplicados em um estado para gerar uma transição (e seu custo associado) para um outro estado, e  $G$  (*goal*) é um conjunto de estados alvo da busca (Figura 10). A solução no paradigma da busca no espaço de estados consiste em procurar um caminho do estado inicial para o estado alvo. A representação do espaço de estados é comumente identificada por um grafo direcionado onde cada nó corresponde a um estado e cada arco corresponde à aplicação de um operador ou regra onde há transições de um estado para outro. Por conta disso, a busca no espaço de estados é equivalente a busca em um grafo com algumas funções objetivo (alvo).

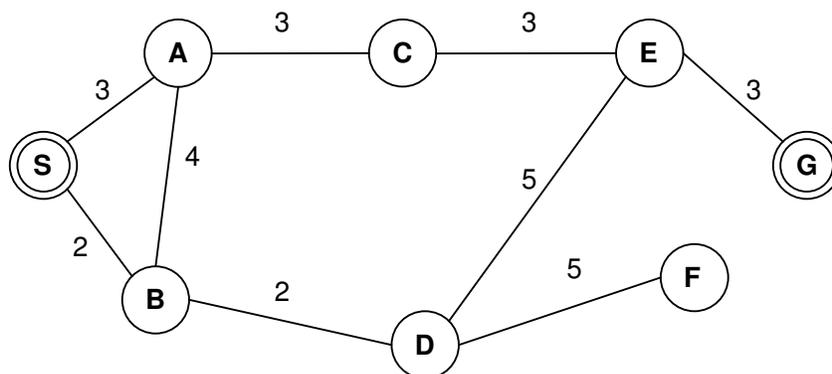


Figura 10: Exemplo de Espaço de Busca

Algoritmos de busca são divididos basicamente em dois grupos: Não Informados (ou Cegos) e Informados (ou Heurísticos) [54]. A busca não informada é utilizada quando o objetivo do problema é encontrar um caminho aceitável ao invés do melhor caminho. Este tipo de busca trata todos os nós do espaço de busca da mesma forma e segue cegamente com a busca sem utilizar conhecimento algum do domínio relativo ao problema, sendo portanto também conhecido como busca uniforme ou busca exaustiva (ou ainda, força-bruta) por tentar exaustivamente chegar à solução por todas as possibilidades de caminho. Em AI, normalmente não há muito interesse pela busca não informada, no entanto, os algoritmos desenvolvidos para este fim têm grande aplicabilidade em muitos algoritmos que realizam busca informada. A razão de interesse é que os algoritmos de busca não informada não exploram o espaço de busca de maneira randômica, porém sempre com algum método bastante sistemático.

Quando o espaço de busca se torna muito grande, métodos de busca não informados tornam-se impraticáveis fazendo com que algum tipo de orientação seja necessário para guiar o processo de busca. Este desejado guia vem do conhecimento específico do domínio do problema e é referido na literatura como informação heurística. Algoritmos de busca que utilizam heurística são conhecidos como Informados. Há sempre uma grande variedade nas diferentes heurísticas para o domínio do problema. Algumas heurísticas podem reduzir o esforço de busca sem sacrificar a otimalidade, enquanto outras podem reduzir bastante o esforço de busca porém resultam apenas em soluções sub-ótimas. Em muitos dos problemas práticos, a diferença entre heurísticas é usualmente uma troca entre a qualidade da solução e o custo para achá-la. A Figura 11 mostra uma taxionomia que engloba a maioria dos algoritmos de busca encontrados em livros e trabalhos de AI.

Métodos de busca informada basicamente utilizam uma função heurística para reordenar a lista de nós que serão expandidos na busca. O nó que apresenta o melhor valor de acordo com a heurística usada será explorado (expandido) primeiro. Algumas estratégias de busca informada também podem definitivamente alguns caminhos parciais não promissores para reduzir o espaço de busca. É por isso que a busca informada (heurística) também é referenciada como poda heurística. Um exemplo de busca com heurística seria o clássico problema em que um viajante precisa decidir a melhor rota (seqüência de cidades) para sua viagem de forma a percorrer uma menor distância total. Uma heurística para este problema seria, por exemplo, usar as distâncias lineares entre cada cidade e a cidade de destino.

A escolha da função heurística é crítica para os resultados da busca. Se usarmos uma função que superestime a distância entre alguns nós, o resultado da busca pode ser sub-ótimo. Portanto, funções heurísticas que não superestimem as distâncias (subestimem) são normalmente usadas em métodos de busca com a intenção de achar soluções ótimas. Heurísticas dessa forma são, por natureza, otimistas e os métodos de busca que as utilizam são denominados admissíveis. Métodos de busca admissíveis garantem que, se existe uma solução para o problema, então a solução encontrada pelo método será ótima [54]. No exemplo do problema do viajante, a heurística usada é otimista, vista que a distância linear sempre será menor que a distância real. Isso garante que o problema resolvido dessa forma, encontrará uma solução ótima.

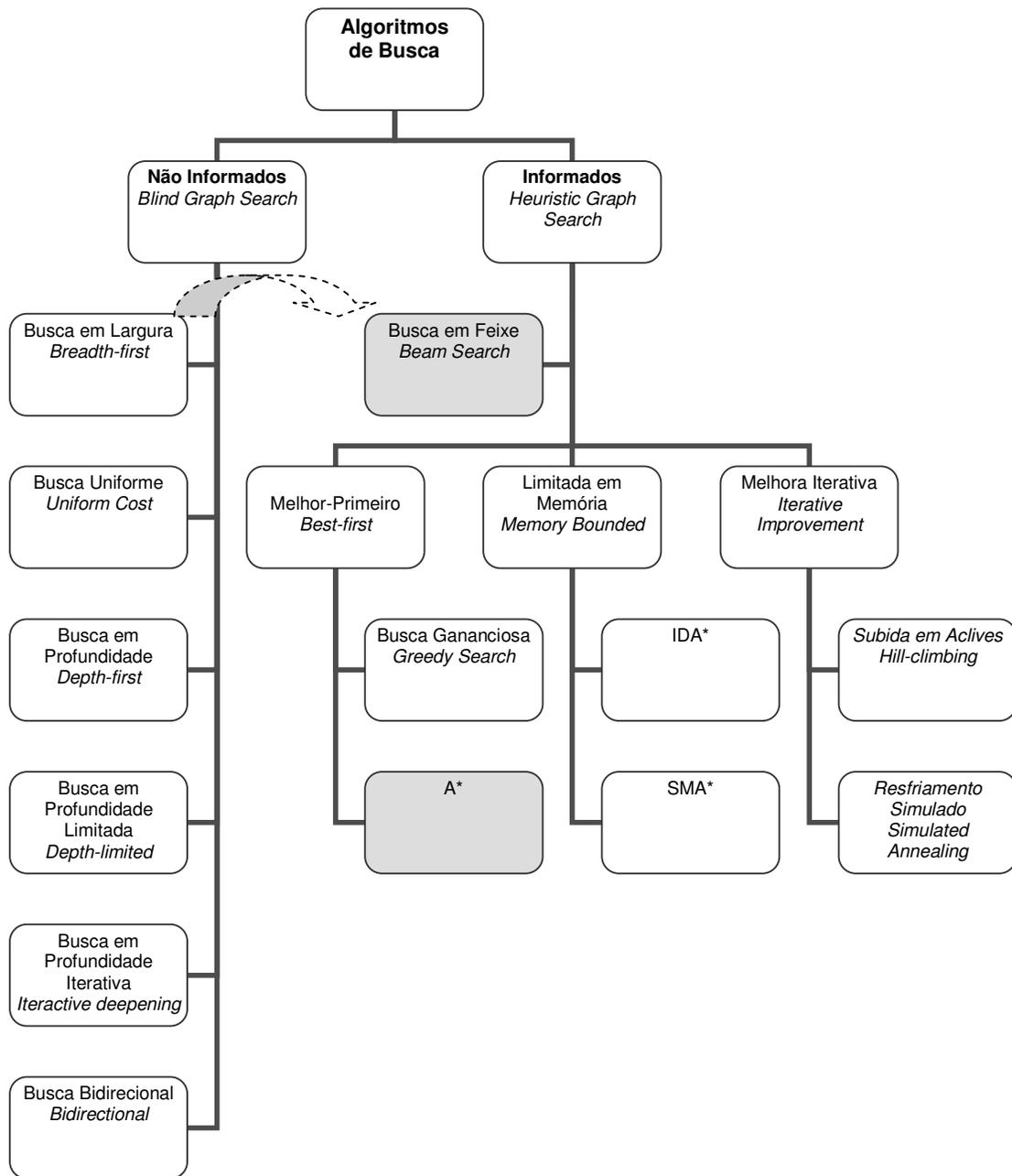


Figura 11: Algoritmos de Busca

No contexto dos algoritmos de busca, a tarefa de reconhecimento de voz se resume a busca pela seqüência mais provável de palavras, para um dado trecho de fala, e resultaria em um espaço de estados exponencial caso todas as seqüências possíveis de palavras fossem consideradas. Este problema foi principalmente resolvido por dois métodos diferentes: decodificação de Viterbi (*Viterbi decoding*) [55], [50] utilizando busca em feixe (*beam search*) [56] ou decodificação de pilha (*stack decoding*) [57], [58], [59] que é uma variação do algoritmo A\* [60]. Há ainda algumas soluções híbridas que combinam a decodificação de Viterbi com o algoritmo A\* [61] e outras que, visando melhor desempenho, utilizam múltiplas passadas (*multi-pass search*) ou passadas de avanço e retorno (*forward-backward search*). Busca em feixe e A\* são algoritmos de busca informada. Estes e outros algoritmos de busca informada, assim como de busca não informada, estão classificados na Figura 11.

Os algoritmos *Viterbi beam search* e *stack decoding*, bem como as demais soluções híbridadas citadas, são brevemente descritas a seguir.

### 3.1.1. Decodificação de Viterbi com Feixe de Busca

A decodificação de Viterbi é um algoritmo de programação dinâmica que procura no espaço de estados a mais provável seqüência de estados que modele o trecho de fala de entrada. O espaço de estados é construído pelos chamados Palavra-HMMs, que são formados pela concatenação dos HMMs dos trifones que as constituem. Todos os Palavra-HMMs constituídos dessa forma são percorridos pelo algoritmo de busca em paralelo. Como o espaço de estados é grande, mesmo para aplicações com vocabulário de tamanho médio, a heurística da busca em feixe é normalmente aplicada para limitar a busca, através da poda (*pruning*) de estados menos prováveis. A combinação do algoritmo de busca e do método de poda utilizados é referida como busca em feixe de Viterbi (*Viterbi beam search*). A decodificação de Viterbi é uma busca síncrona no tempo que processa a fala, segmento a segmento, atualizando todos os estados associados a um segmento antes de passar para o próximo.

Por ser síncrona, a busca em feixe é possivelmente a técnica mais utilizada em sistemas de reconhecimento de voz [62], [63], [64]. É um algoritmo de busca ao estilo da busca em largura (*breadth-first*), no qual os nós de uma determinada altura  $h$  são analisados antes de passar para nós em uma altura  $h+1$  em relação à raiz. Porém, diferente da busca em largura, a busca em feixe expande, a cada passo, apenas os nós que apresentam uma probabilidade alta de sucederem. Apenas estes nós expandidos permanecem no feixe e o restante é ignorado (podado) aumentando assim a eficiência da busca. A seta na Figura 11 é uma referência da relação entre a busca em largura e a busca em feixe. A Figura 12 [39], por outro lado, ilustra os passos de uma busca em feixe.

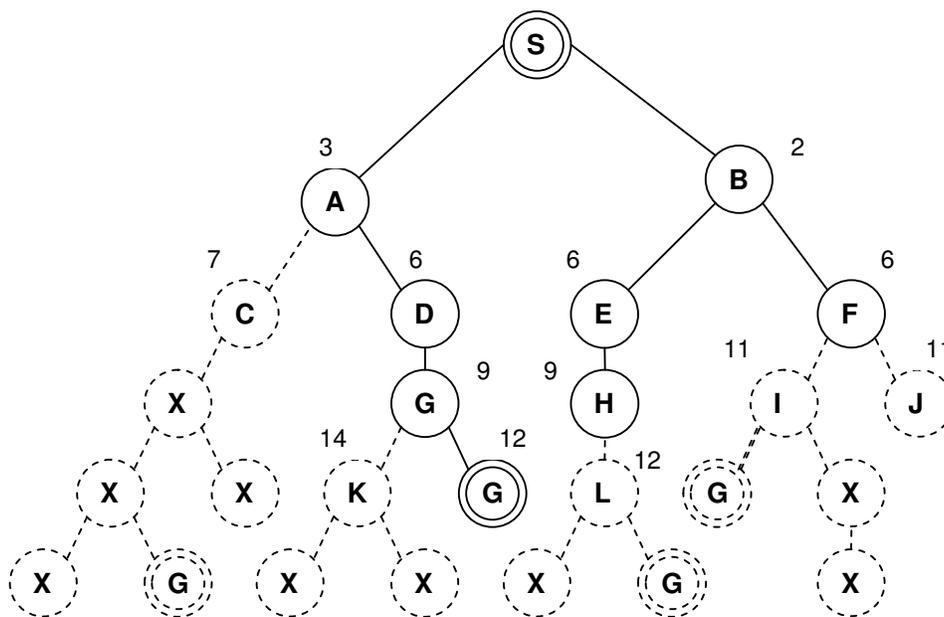


Figura 12: Dinâmica da busca em feixe (*beam search*) [39]

### **3.1.2. Decodificação de Pilha (*stack decoding*)**

O algoritmo *stack decoding* mantém uma pilha com hipóteses parciais<sup>4</sup> ordenadas em ordem decrescente em função da verossimilhança a posteriori. Em cada passo, o algoritmo retira a melhor hipótese da pilha. Se for uma hipótese completa, ela é retornada. Do contrário, a hipótese é expandida em mais uma palavra, de forma que, tentando todas as possíveis extensões, avalia-se a hipótese (parcial) resultante com respeito ao trecho de fala entrada e, então, se reinsere na pilha ordenada. Qualquer conjunto dentre as melhores hipóteses (*N-best*) [65] pode ser gerado dessa maneira. Para evitar um crescimento exponencial no conjunto de todas as possibilidades de seqüências de palavras nos sistemas com médio e grande vocabulário, as hipóteses parciais são expandidas apenas por um conjunto limitado de palavras a cada passo. Esses candidatos são identificados em uma passada com rápida associação (*fast match*) [66], [67]. Como nesta dissertação, todos os experimentos foram baseados na decodificação por Viterbi, não exploraremos em mais detalhes a decodificação por pilha, ficando assim apenas uma breve referência.

### **3.1.3. Busca em Múltiplas Passadas**

Algoritmos de busca em múltiplas passadas (*multi-pass*) [68], [65], [69], [70], [71], [72] empregam uma estratégia de refinamento de etapas para realizar a decodificação (*coarse-to-fine strategy*). Com essa abordagem, modelos computacionalmente mais simples são inicialmente usados para produzir uma lista com hipóteses de palavras mais prováveis. Estas hipóteses são depois refinadas usando modelos mais detalhadas e que demandam maior esforço computacional. A busca referente à primeira passada (normalmente conhecida como localização rápida – *fast match*) produz uma lista denominada *N-best*, com as possíveis seqüências de palavras, ou então um grafo de palavras (ou treliça) como seu resultado.

Como exemplo, a passada inicial pode ser realizada usando apenas trifones intrapalavra e um LM do tipo bigrama para gerar a lista hipotética de candidatos. Então, numa segunda passada do decodificador, um LM do tipo trigrama pode ser utilizado com trifones entre-palavras. A busca em duas passadas resultante tem desempenho comparável à busca de Viterbi com apenas uma passada, porém normalmente requer menos recursos computacionais. Uma importante e emergente variante da técnica de decodificação de pilha utilizando múltiplas passadas é a busca por envelope (*envelope search*) [61].

### **3.1.4. Busca em Avanço e Retorno (*forward-backward*)**

Algoritmos de busca em avanço e retorno (*forward-backward search*) implementam primeiramente uma busca síncrona, no sentido do tempo, que visa facilitar uma segunda busca em sentido contrário, mais complexa e que requer maior esforço computacional [68], [69]. Esse método tem geralmente o efeito de acelerar a busca da passada de retorno, dado que o número de hipóteses a serem exploradas é bastante reduzido pela busca no sentido de avanço.

Um modelo acústico ou lingüístico simplificado é usado para realizar uma rápida e eficiente busca no sentido de avanço, onde as pontuações acumuladas por hipóteses

---

<sup>4</sup> Uma hipótese parcial reflete um trecho inicial da fala de entrada. Uma hipótese completa, ou simplesmente hipótese, reflete o trecho completo da fala.

parciais que estejam acima de um patamar de corte são armazenadas em cada estado. Logo em seguida, uma busca em feixe de Viterbi normal, executada no interior das palavras, é realizada na direção contrária ao tempo para gerar uma lista com as  $N$  melhores hipóteses. A busca nessa direção computa uma alta pontuação para uma hipótese somente quando também há um bom caminho na direção de avanço que leve a um final de palavra naquele instante de tempo.

Os algoritmos de busca em avanço e retorno facilitaram bastante a implementação, em tempo real, de tarefas de reconhecimento de voz em larga escala. A busca no sentido de retorno é rápida o suficiente para ser realizada sem nenhum atraso perceptível após a busca em avanço. Esta pode ser feita de forma subótima e, portanto, demandando menor esforço computacional.

### **3.2. Fundamentos para uma Busca Síncrona**

Como dito anteriormente, o problema do reconhecimento de fala pode ser organizado em uma hierarquia de redes de estados com um número finito de nós e arcos correspondentes às fontes de conhecimento acústico, fonético e sintático e suas interações. Em um sistema de reconhecimento de fala, o conhecimento acústico está relacionado à forma de parametrização do sinal de voz (parâmetros LPC, cepstrais, etc.), o conhecimento fonético, à transcrição fonética das palavras do vocabulário, e o conhecimento sintático, ao LM. O reconhecimento de uma locução corresponde a encontrar o caminho ótimo através da rede de estados finitos.

Esta busca pode ser realizada através de decodificação seqüencial usando os conceitos de programação dinâmica e o princípio da otimalidade definido por Bellman [73]: *“um conjunto de decisões ótimas tem a propriedade de, qualquer que tenha sido a primeira decisão, as decisões restantes precisam ser ótimas em relação à saída da primeira decisão”*. Em termos do problema de encontrar o melhor caminho através de uma rede de estados finita, o princípio da otimalidade permite que a decodificação seja feita de modo síncrono, pois toda a informação requerida para os caminhos ótimos locais está disponível, e os caminhos ótimos globais podem ser encontrados a partir dos caminhos ótimos locais.

Para a tarefa de reconhecimento de palavras conectadas, é conveniente decompor a rede em dois níveis: nível de frases (gramático) e nível intrapalavra. Cada um dos níveis tem propriedades completamente diferentes. O nível intrapalavra é geralmente um modelo de palavra, que pode ser um HMM da palavra inteira, ou uma representação da palavra formada pela concatenação de HMMs de subunidades acústicas (ex., trifones - Figura 13), definido como Palavra-HMM. O nível gramático é representado por uma rede gramática (de acordo com o LM), na qual os nós representam fronteiras de palavras, e os arcos representam modelos de palavras.

Para realizar a busca em uma rede de estados finita é necessário estabelecer uma medida de custo (ex., distância ou verossimilhança) associada ao caminho. Esta medida inclui o custo de estar em um nó intrapalavra, o custo de fazer transições de um nó intrapalavra para outro e o custo de entrar em um arco gramático. Na tarefa de reconhecimento de fala, o custo acumulado de um caminho que passa por um determinado nó na rede de estados finita no instante  $t$  pode ser definido como o negativo da verossimilhança acumulada do caminho no instante  $t$ . Esta verossimilhança é definida como o logaritmo da probabilidade daquele caminho. Assim, a rede resultante é uma rede de estados finita estocástica onde o custo de um

caminho depende da seqüência de observação, do tempo que o sistema ficou em determinado nó, e da história de transições do caminho.

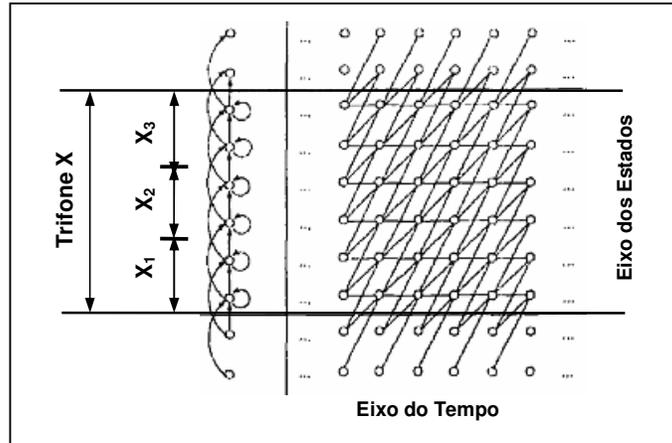


Figura 13: Estrutura de um modelo de trifone e o espaço de busca associado [63]

Com todos os custos atribuídos convenientemente, o procedimento de busca pelo melhor caminho na rede de estados finita é essencialmente o mesmo de encontrar o caminho de custo mínimo através da rede, ou equivalentemente, realizar uma decodificação de máxima verossimilhança.

Pela Equação (20), repetida abaixo, fundamenta-se o reconhecimento da fala por meio da decodificação de máxima verossimilhança,

$$\hat{W}_1^N = \arg \max_{W_1^N} \left\{ P(W_1^N) \max_{S_1^T} \left( P(X_1^T, S_1^T | W_1^N) \right) \right\} \quad (23)$$

onde  $W_1^N$  representa uma seqüência de palavras, chamada de Frase-HMM por ser uma concatenação de Palavra-HMMs. A probabilidade conjunta de se observar a seqüência de vetores acústicos  $X_1^T = \{x_1 \dots x_T\}$  e a seqüência de estados  $S_1^T = \{s_1 \dots s_T\}$ , percorrida através deste Frase-HMM é dada por:

$$\begin{aligned} P(X_1^T, S_1^T | W_1^N) &= \prod_{t=1}^T P(x_t, s_t | s_{t-1}, W_1^N) \\ &= \prod_{t=1}^T \left[ P(s_t | s_{t-1}, W_1^N) P(x_t | s_t) \right] \end{aligned} \quad (24)$$

onde  $P(x_t, s_t | s_{t-1}, W_1^N)$  denota o produto da probabilidade de transição do estado  $s_{t-1}$  para o estado  $s_t$ , pela probabilidade de emissão de  $x_t$  no estado  $s_t$  do Frase-HMM  $W_1^N$ . Estas duas probabilidades são descritas a seguir:

- A probabilidade de emissão de saída  $P(x_t | s)$  para o estado acústico  $s$  leva em conta a possibilidade de que alguma técnica de compartilhamento de estados tenha sido utilizada (conforme Seção 2.1.2). Note-se que, para a probabilidade de emissão  $P(x_t | s)$ , abusou-se um pouco da notação de forma

a não se distinguir necessariamente entre o estado  $s$  de um modelo de trífone ou palavra e a probabilidade de emissão genérica associado ao mesmo.

- A probabilidade de transição  $P(s_t | s_{t-1}, W_1^N)$  depende da seqüência de palavras  $W_1^N$  sob hipótese. Isso implica que uma rede de estados finitos (Frase-HMMs) deve ser levada em consideração para cada seqüência de palavras  $W_1^N$ .

Conforme mencionado, o espaço de busca é descrito como uma rede onde se procura o melhor alinhamento de estados. A busca deve ser realizada em dois níveis: no nível de estados ( $S_1^T$ ) e no nível de palavras ( $W_1^N$ ). Como será observada, a aproximação por máximo possibilita recombinar eficientemente as hipóteses em ambos os níveis, através de DP. Dessa forma, a explosão combinatória do número de hipóteses de busca pode ser limitada, o que é uma das principais características da DP. Ao mesmo tempo, as hipóteses de busca são construídas e avaliadas sempre de forma síncrona, no sentido da esquerda para a direita. Essa propriedade permite uma eficiente estratégia de poda para eliminar hipóteses pouco prováveis.

### **3.3. Algoritmo One Pass usando um Léxico Linear**

#### **3.3.1. Definição do Espaço de Busca**

Nesta seção, considerando um léxico de estrutura linear, descreve-se o algoritmo *One Pass* que representa a base para a grande maioria das estratégias de busca síncrona em apenas uma passada. Originalmente (década de 70), o algoritmo *One Pass* foi projetado para tarefas de reconhecimento com pequenos vocabulários, como o reconhecimento de seqüências de dígitos [74], [75], [76], [47]. Durante os últimos 30 anos, no entanto, esses algoritmos e suas extensões se mostraram eficientes na manipulação de vocabulários com 20 mil palavras ou mais.

O termo “léxico linear” denota o fato de que as palavras são manipuladas de forma completamente dissociada do processo de busca. Ao contrário de um léxico organizado em forma de árvore (visto na Seção 3.4), não há compartilhamento entre as palavras tão logo as hipóteses de busca sejam concebidas. Para um vocabulário de três palavras (A, B e C), o espaço de busca está representado na Figura 14. Existem dois tipos de transição: as transições acústicas representando as probabilidades dos modelos acústicos que formam as palavras (Figura 13), e as transições de linguagem, representando as probabilidades do LM. Na Figura 14, assume-se o bigrama como LM. Nela, para cada par possível de palavras ( $v, w$ ), há uma transição do LM que é associado a uma probabilidade condicional do tipo bigrama  $P(w | v)$ , que liga o término de uma palavra predecessora  $v$  com o começo de uma palavra  $w$ . Para o reconhecimento de palavras, desdobra-se a máquina de estados finitos sobre o eixo do tempo de uma ocorrência da fala. Por simplicidade, a Figura 14 não cobre os detalhes dos modelos acústicos e exibe apenas as transições do LM nos tempos  $t_2$  e  $t_3$ . Tanto as transições acústicas, como as transições de linguagem devem ser consideradas em todos os instantes de tempo. Como resultado, há um grande número de seqüências de estados possíveis, e todas as combinações de estado e tempo devem ser consideradas sistematicamente durante o reconhecimento.

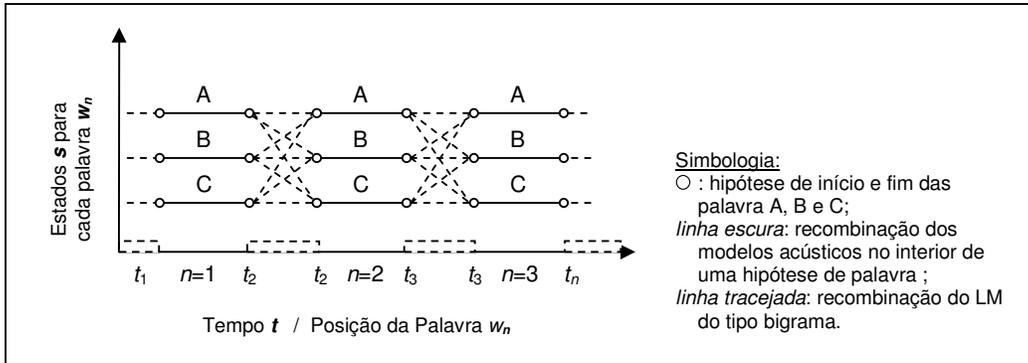


Figura 14: Espaço de busca para um vocabulário de 3 palavras (A,B,C)

Levando-se em conta a aproximação por máximos, o problema de busca pode ser especificado da seguinte forma: Deseja-se associar cada vetor acústico observado no tempo  $t$ , a uma chave dupla contendo estado e a palavra associada. Este mapeamento é visto como um percurso de alinhamento no tempo e é dado por uma seqüência de chaves duplas (estado, palavra), da forma  $(s_1, w_1), \dots, (s_t, w_t), \dots, (s_T, w_T)$ , onde  $T$  é o número de segmentos da elocução. Um exemplo de percurso de alinhamento no tempo para o reconhecimento de palavras conectadas é representado na Figura 15. Para estes caminhos, há óbvias restrições de continuidade ou regras de transição, conforme indicado na Figura 16.

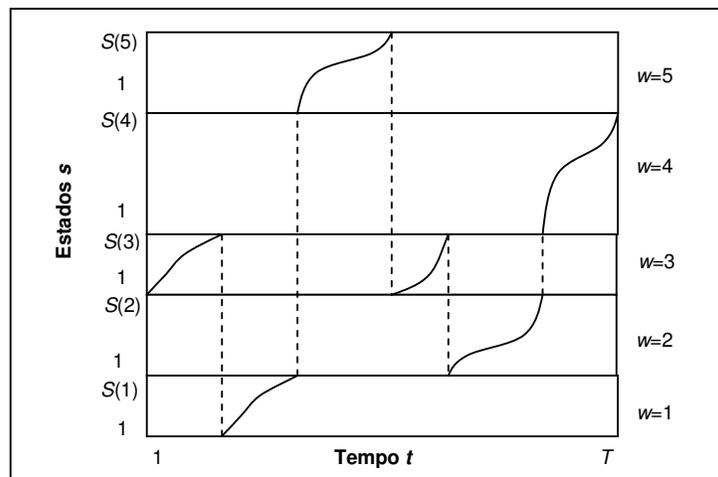


Figura 15: Exemplo de percurso de alinhamento no tempo

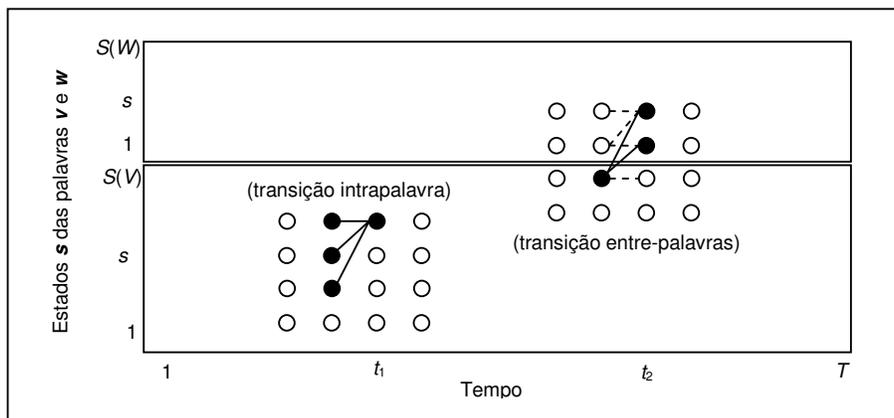


Figura 16: Recombinações de caminho dentro de uma mesma palavra e entre palavras

Dado que os modelos de palavras são obtidos através da concatenação de modelos de trifones, as regras de transição no interior de uma palavra (esquerda da Figura 16) são aquelas inerentes aos HMMs, conforme visto na Figura 13. Nas fronteiras entre palavras (direita da Figura 16), devem-se permitir transições que liguem o estado de saída  $S_v$  de uma palavra predecessora  $v$  com os estados de entrada  $s=1$  e  $s=2$  de uma palavra  $w$ . A busca baseada em DP, que será apresentada a seguir, permite o cálculo das probabilidades:

$$P(w_1 \dots w_t) P(x_1 \dots x_t; s_1 \dots s_t | w_1 \dots w_t) \quad (25)$$

caminhando no sentido da esquerda para a direita, sobre o tempo  $t$ , e computando a otimização da seqüência de palavras desconhecidas ao mesmo tempo. Note que as seqüências desconhecidas de palavras e de estados são determinadas simultaneamente. Dentro da estratégia de aproximação por máximos ou critério de Viterbi, o algoritmo apresentado, baseado em DP, apresenta uma forma fechada de solução para a interdependência entre alinhamento temporal não linear, detecção de fronteiras entre palavras e identificação de palavras no reconhecimento contínuo da fala [74], [77], [44], [48], [76], [78], [47].

### 3.3.2. Recursividade com Programação Dinâmica

O conceito chave do algoritmo *One-Pass*, cuja estratégia é baseada em DP, está apoiado em duas quantidades principais, definidas a seguir:

$Q(t, s, w)$  := pontuação do melhor caminho até o tempo  $t$  que termina no estado  $s$  da palavra  $w$ .  
(a pontuação ou *Score* é dada pelo negativo da verossimilhança)

$B(t, s, w)$  := tempo em que inicia o melhor caminho até o tempo  $t$  que termina no estado  $s$  da palavra  $w$ .  
(conhecido como ponteiro de retorno ou *Back pointer*)

Levando em conta os tamanhos de memória atualmente disponíveis, pode-se concluir que o ponteiro de retorno (*back pointer*) não é absolutamente necessário em tarefas com pequenos vocabulários, como reconhecimento de seqüências de dígitos. No entanto, para vocabulários de 20 mil palavras ou mais, é essencial reduzir os requerimentos de espaço o tanto quanto possível.

Conforme indicado na Figura 15, existem dois tipos de regras de transição de percurso, a saber: as regras no interior de palavras e as regras nas fronteiras entre palavras. O conceito de DP aponta o uso dessas regras para a decomposição do percurso em duas partes e define relações de recorrência que podem ser resolvidas com ajuda de tabelas, que, para o caso descrito, seria a tabela  $Q(t, s, w)$ . Numa configuração mais genérica para problemas de otimização, esse conceito é usualmente referido como princípio de otimalidade de *Bellman* [73]. No interior da palavra, define-se a seguinte equação de recursividade

$$Q(t, s, w) = \max_{s'} \{ P(x_t, s | s', w) Q(t-1, s', w) \} \quad (26)$$

$$B(t, s, w) = B(t-1, s'_{\max}(t, s, w), w)$$

onde, de forma semelhante a Equação (4),

$$P(x_t, s | s', w) = a_{s'} b_s(x_t) \quad (27)$$

e  $s'_{\max}(t, s, w)$  é o estado predecessor ótimo para a hipótese  $(t, s, w)$ . Os ponteiros de retorno  $B(t, s, w)$  são propagados simplesmente de acordo com as decisões da DP, como mostrado na Figura 17 e reportam o tempo de início para cada final de palavra sob hipótese. Quando se encontra uma fronteira de palavras em potencial, realiza-se a recombinação entre as palavras predecessoras, definida por:

$$H(w, t) = \max_v \{P(w|v)Q(t, S_v, v)\} \quad (28)$$

onde  $P(w|v)$  é a probabilidade condicional do LM referente ao bigrama  $(v, w)$ . O símbolo  $S_v$  denota o estado de saída da palavra  $v$ . Para permitir que palavras sucessoras sejam iniciadas, introduz-se um estado especial  $s=0$  para qual se passa tanto a pontuação, quanto o índice de tempo:

$$\begin{aligned} Q(t-1, s=0, w) &= H(w, t-1) \\ B(t-1, s=0, w) &= t-1 \end{aligned} \quad (29)$$

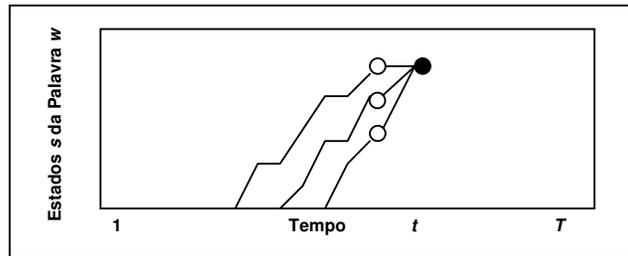


Figura 17: Ilustração dos ponteiros de retorno

As equações, definidas em (29), assumem que antes dos estados  $s=1, \dots, S_v$  serem avaliados para cada palavra  $w$ , os estados de inicialização  $s=0$  devem ser computados. O mesmo índice de tempo  $t$  é usado intencionalmente, porque o LM não “absorve” um vetor acústico. Note-se que as pontuações  $Q(t, s, w)$  capturam tanto as probabilidades dependentes de observações acústicas resultantes dos HMMs, quanto as probabilidades do LM.

As operações a serem realizadas estão resumidas na Tabela 3. A seqüência de vetores acústicos extraídos do sinal de fala de entrada é processada estritamente da esquerda para a direita. De acordo com as equações definidas pela DP, dois níveis podem ser distinguidos na Tabela 3: o nível acústico onde as recombinações são realizadas no interior das palavras e o nível de par de palavras onde as recombinações são realizadas levando em conta os bigramas do LM. O procedimento de busca trabalha de acordo com uma estratégia de busca em largura, síncrona no tempo, ou seja, todas as hipóteses para seqüências de palavras são estendidas em paralelo para cada vetor acústico de entrada. Para reduzir os requerimentos de espaço em memória, recomenda-se introduzir um vetor de rastreamento  $v_0$  (*traceback array*), adicionalmente aos ponteiros de retorno. A cada instante, o vetor de rastreamento é usado para gravar as decisões tomadas quanto as melhores hipóteses de término de palavra e seus instantes de início. Usando o vetor de rastreamento, a seqüência de palavras reconhecida pode ser obtida eficientemente a partir de algumas consultas em seus índices, partindo do final da ocorrência, conforme a Figura 18.

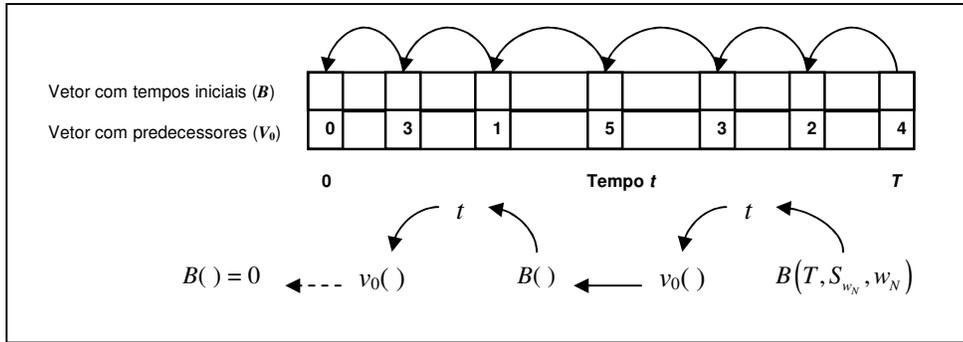


Figura 18: Ilustração de vetores de rastreamento

Tabela 3: Algoritmo One-Pass com DP usando um léxico linear [63]

Proceder sobre tempo $t$ da esquerda para a direita	
Nível Acústico: Processar as hipóteses (estado,palavra)	
- Inicialização:	$Q(t-1, s=0, w) = H(w, t-1)$ $B(t-1, s=0, w) = t-1$
- Alinhamento no tempo:	$Q(t, s, w)$ usando DP
- Propagar os ponteiros de retorno:	$B(t, s, w)$
- Podar as hipóteses menos prováveis	
- Limpar as listas de referência utilizadas	
Nível de par de palavras: Processar hipóteses de final de palavra	
Para cada palavra $w$ faça:	
	$H(w, t) = \max_v \{P(w v)Q(t, S_v, v)\}$
	$v_0(w, t) = \arg \max_v \{P(w v)Q(t, S_v, v)\}$
- Guardar o melhor predecessor	$v_0 := v_0(w, t)$
- Guardar a melhor fronteira	$\tau_0 := B(t, S_{v_0}, v_0)$

### 3.3.3. Busca em Feixe

Dado que, para uma janela fixa de tempo, todas as hipóteses (estado, palavra) cobrem a mesma porção do sinal de entrada, suas pontuações podem ser diretamente comparadas. Isso permite ao sistema evitar uma busca exaustiva e, ao contrário, realizar uma busca orientada a dados (*data-driven search*), ou seja, focar a busca nas hipóteses que exibem maior probabilidade de resultarem na melhor seqüência de estados [44]. A cada janela de 10ms, tipicamente, a pontuação da melhor hipótese é determinada, então todas as hipóteses cujas pontuações caíam abaixo da pontuação ótima por mais de um fator fixo são podadas, ou seja, removidas de considerações posteriores no processo de busca. Testes experimentais reportam que para este tipo de busca em feixe, dependendo da acústica do sinal de entrada e das restrições do LM, apenas uma pequena fração do total de hipóteses (estado, palavra) deve ser processada a cada 10ms do sinal de fala e, apesar desta poda, o número de erros no reconhecimento permanece virtualmente constante. Esta estratégia de busca em feixe será considerada com maior detalhe, a seguir, no contexto da organização do léxico de pronúncia em forma de árvore. Adicionalmente, para explorar completamente as vantagens computacionais desta estratégia de busca em feixe, a construção dinâmica

do espaço ativo de busca é possível como será discutido na seção seguinte. O algoritmo *One-Pass*, baseado em DP, combinado com a busca em feixe forma o fundamento dos componentes de busca, tanto para sistemas de reconhecimento com pequenos vocabulários, como para sistemas com amplos vocabulários [79], [80], [81], [82], [83], [84], [85], [86], [77], [87], [88], [89], [90], [19].

### 3.4. Algoritmo One Pass usando um Léxico em Árvore

#### 3.4.1. Definição do Espaço de Busca

O algoritmo apresentado anteriormente, quando aplicado em tarefas de reconhecimento de fala com amplos dicionários (20.000 palavras ou mais), por razões de eficiência, deve ter o seu léxico de pronúncias reorganizado sob a forma de uma árvore de prefixos onde cada arco representa um modelo de fonema, seja ele dependente ou independente de contexto [91],[88],[89]. Parte de um léxico de pronúncias estruturado em árvore é exibido na Figura 19. A idéia de se usar uma representação em árvore foi sugerida na década de 70, no sistema denominado CASPER [92] e no acesso ao léxico do sistema Spectra (LAFS) [93]. No entanto, quando se utiliza um léxico em árvore no contexto de um LM, como o bigrama, e da DP, existem alguns detalhes técnicos, específicos de DP, que devem ser levados em conta e que requerem uma estruturação apropriada do espaço de busca [91],[88]. A seguir, o algoritmo aplicado a este contexto será apresentado em detalhe.

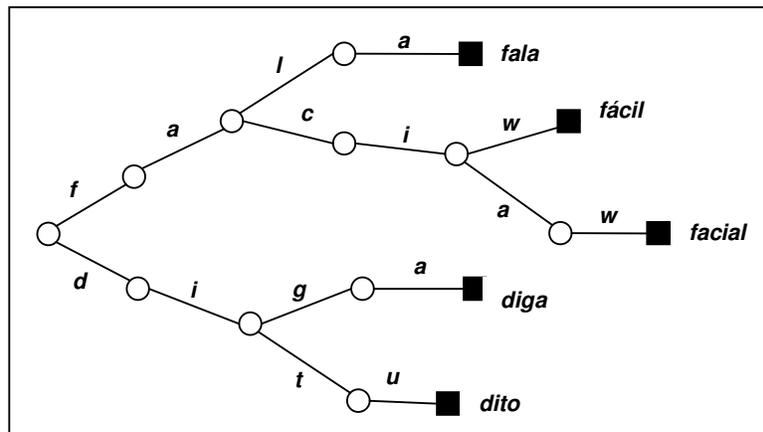


Figura 19: Dicionário léxico de pronúncia organizado em formato de árvore

Quando usamos um LM do tipo bigrama juntamente a este tipo de representação do léxico em árvore, percebemos o problema de que a identidade da palavra sob hipótese  $w$  é conhecida apenas quando uma folha da árvore é alcançada. Portanto, as probabilidades do LM podem apenas ser incorporadas depois de alcançado o estado terminal da segunda palavra do bigrama. Como resultado, podemos aplicar a probabilidade do LM apenas no final da árvore. Para tornar os princípios da DP possíveis, o espaço de busca é estruturado como se segue. Para cada palavra predecessora  $v$ , introduz-se uma cópia separada da árvore de pronúncias de forma que, durante o processo de busca, sempre será conhecida a palavra predecessora  $v$  quando o final de uma palavra  $w$  estiver sob hipótese. A Figura 20 ilustra o conceito para um vocabulário de 3 palavras (A, B e C), onde o léxico em árvore é representado em um esquema simplificado. É importante enfatizar que a Figura 20 mostra um espaço de busca conceitual, o qual é grande demais para ser totalmente representado

como um todo. Ao invés disso, como será visto, as partes ativas deste espaço de busca são construídas dinamicamente em combinação com a busca em feixe. Na montagem da Figura 20, aplica-se a probabilidade  $P(w | v)$  do bigrama, quando o estado final de uma palavra  $w$ , com predecessora  $v$ , é alcançado, e usa-se a pontuação total obtida para inicializar o léxico em árvore correspondente, ou seja, a árvore que possui a palavra  $w$  como predecessora.

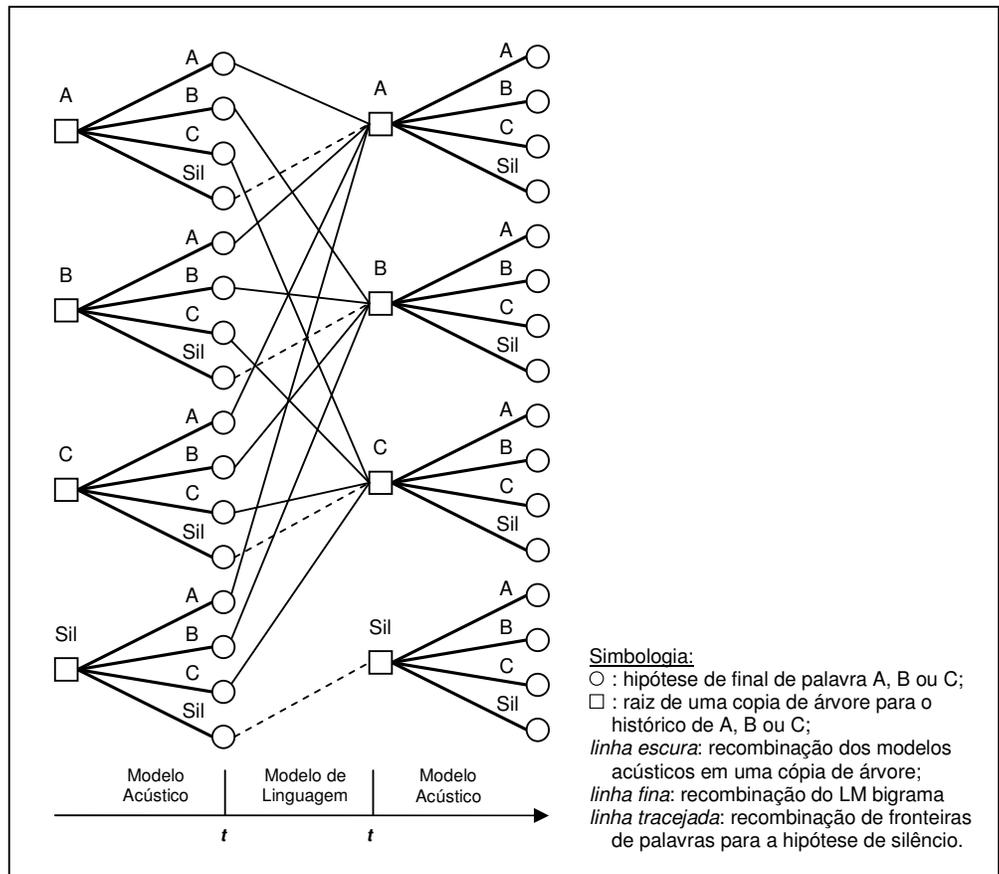


Figura 20: Recombinação do LM bigrama e manipulação de silêncio (Sil) intra-frase em um léxico em árvore (vocabulário de 3 palavras: A, B e C) [63]

No processo de reconhecimento, além das palavras faladas, deve-se contar com possíveis pausas entre as mesmas. Para lidar com as chamadas pausas intra-frase, utiliza-se um modelo especial de HMM para o silêncio (Sil), o qual tem uma cópia em separado adicionada a cada árvore. Além disso, para uma possível pausa no início da frase, utiliza-se uma cópia em separado do léxico para a primeira palavra da frase; a essa cópia da árvore é associado o modelo de silêncio como a palavra predecessora. Como resultado dessa abordagem, as cópias do modelo de silêncio não requerem um tratamento diferenciado e podem ser processadas como palavras regulares do vocabulário. No entanto, há uma exceção que deve ser tratada: nas fronteiras entre palavras, não há probabilidade do LM para os modelos de silêncio. Como mostrado na Figura 20, existem dois tipos de extensões de caminho e recombinações que são aquelas no interior de palavras e aquelas nas fronteiras entre palavras. No interior de palavras, existem as linhas mais escuras representando as transições dos HMMs. Nas fronteiras entre palavras, existem as linhas finas e tracejadas que representam as recombinações do LM bigrama. Assim como as recombinações acústicas, estas também são realizadas a cada janela de tempo (ex. 10ms), mas apenas para palavras que tenham atingido seu estado terminal. As linhas tracejadas estão relacionadas às

recombinações de cópias de silêncio intra-frase. Para inicializar uma nova hipótese de palavra, deve-se incorporar a probabilidade bigrama à pontuação e, então, determinar a melhor palavra predecessora. Assim, a melhor pontuação é propagada para a raiz do léxico associado, o qual é representado pelo símbolo ( $\square$ ). O símbolo ( $\circ$ ) denota um final de palavra.

### 3.4.2. *Recursividade com Programação Dinâmica*

Para uma especificação quantitativa do procedimento de busca, assume-se que cada arco do léxico em árvore é representado por um HMM. O índice de estados  $s$  é usado diretamente e assume-se que a estrutura do léxico é capturada pelas probabilidades de transição dos HMMs. Para formular a abordagem utilizando DP, introduzem-se as duas quantidades a seguir:

$Q_v(t,s) :=$  pontuação do melhor caminho até o tempo  $t$  que termina no estado  $s$  do léxico em árvore que possui como predecessor a palavra  $v$ .

$B_v(t,s) :=$  tempo inicial do melhor caminho até o tempo  $t$  que termina no estado  $s$  do léxico em árvore que possui como predecessor a palavra  $v$ .

Em outras palavras,  $B_v(t,s)$  é o ponteiro de retorno que aponta para o tempo de início da cópia do léxico em árvore que possui como predecessor a palavra  $v$ . Esse ponteiro de retorno é necessário para recuperação das palavras predecessoras a  $v$  e seus respectivos tempos iniciais, como complemento a definição de  $Q_v(t,s)$ . As duas quantidades são avaliadas usando a recursão da DP para  $Q_v(t,s)$ :

$$Q_v(t,s) = \max_{s'} \{P(x_t, s | s') Q_v(t-1, s')\} \quad (30)$$

$$B_v(t,s) = B_v(t-1, s'_{\max_v}(t,s))$$

onde  $s'_{\max_v}(t,s)$  é o estado predecessor ótimo para a hipótese  $(t,s)$  e a palavra predecessora  $v$ . Como antes, os ponteiros de retorno  $B_v(t,s)$  são propagados de acordo com as decisões da DP. Diferente da palavra predecessora  $v$ , o índice  $w$  da palavra sobre consideração só é necessário e conhecido quando uma hipótese de caminho alcança um nó terminal do léxico em árvore: cada nó terminal da árvore é etiquetado com a palavra correspondente do vocabulário.

Usando uma inicialização apropriada para  $s=0$ , a Equação (30) pode incluir a otimização sobre as fronteiras desconhecidas entre palavras, conforme mostrado a seguir. Nas fronteiras entre palavras, deve-se procurar a melhor palavra predecessora  $v$  para cada palavra  $w$ . Como no caso de um léxico linear, define-se:

$$H(w,t) = \max_v \{P(w | v) Q_v(t, S_w)\} \quad (31)$$

onde o estado  $S_w$  denota o estado terminal da palavra  $w$  no léxico em árvore.  $H(w,t)$  representa a probabilidade de, no tempo  $t$ , ocorrer o último segmento de uma palavra  $w$ . Para propagar as hipóteses de caminho pelos léxicos em árvore ou então iniciá-los, no caso de ainda não existirem, passa-se a pontuação e o índice de tempo antes de processar as hipóteses para o tempo  $t$ :

$$Q_v(t-1, s=0) = H(v, t-1) \quad (32)$$

$$B_v(t-1, s=0) = t-1$$

Os detalhes desse algoritmo estão resumidos na Tabela 4.

Tabela 4: Algoritmo One-Pass com DP usando um léxico em árvore [63]

Proceder sobre tempo $t$ da esquerda para a direita	
Nível Acústico: Processar as hipóteses (estado,palavra)	
- Inicialização:	$Q_v(t-1, s=0) = H(v, t-1)$ $B_v(t-1, s=0) = t-1$
- Alinhamento no tempo:	$Q_v(t, s)$ usando DP
- Propagar os ponteiros de retorno:	$B_v(t, s)$
- Podar as hipóteses menos prováveis - Limpar as listas de referência utilizadas	
Nível de par de palavras: Processar hipóteses de final de palavra	
Para cada palavra $w$ faça:	
	$H(w, t) = \max_v \{P(w v)Q_v(t, S_w)\}$ $v_0(w, t) = \arg \max_v \{P(w v)Q_v(t, S_w)\}$
- Guardar o melhor predecessor	$v_0 := v_0(w, t)$
- Guardar a melhor fronteira	$\tau_0 := B_{v_0}(t, S_w)$

### 3.4.3. Extensão para LM do tipo Trigrama

Até agora, considerou-se a abordagem de busca *One-Pass* apenas no contexto de LM do tipo bigrama. Para estender o método de busca em árvore de LM bigrama para trigrama, deve-se levar em conta que nos trigramas as probabilidades do LM são condicionadas às duas palavras anteriores ao invés de apenas uma [94], [95], [96]. Dessa forma, a incorporação do LM trigrama no método de busca em árvore requer uma reestruturação da organização do espaço de busca. A Figura 21 ilustra o espaço de busca usando o modelo trigrama. Para cada histórico com duas palavras ( $u, v$ ), introduz-se uma cópia em separado do léxico em árvore; na Figura 21, a raiz de cada cópia de árvore é etiquetada com o seu histórico de duas palavras. Como no caso do LM bigrama, a estrutura do espaço de busca é definida de forma que, na rede de busca, as probabilidades ou custos de cada ramo dependem apenas do ramo em si (junto com os seus nós de início e fim) e nada mais. Essa propriedade da rede de busca permite a aplicação direta dos princípios da DP. Note-se que, em comparação com o espaço de busca organizado para o LM bigrama, o tamanho do espaço de busca potencial cresce drasticamente por um fator adicional, que é o tamanho do vocabulário. Portanto, de forma a manter gerenciável o esforço de busca, uma estratégia eficiente de poda como descrito anteriormente é ainda mais crucial para o caso do LM trigrama.

Por simplicidade, omitiram-se as cópias de silêncio da Figura 21. Para permitir o silêncio intra-frase, utilizam-se os mesmos conceitos aplicados ao LM bigrama [91], [88]. Na recombinação do LM trigrama, deve-se conhecer a identidade das duas palavras predecessoras diferentes de silêncio. Dessa forma, uma cópia separada do modelo de silêncio é requerida para cada par de palavras predecessoras diferentes do silêncio em si.

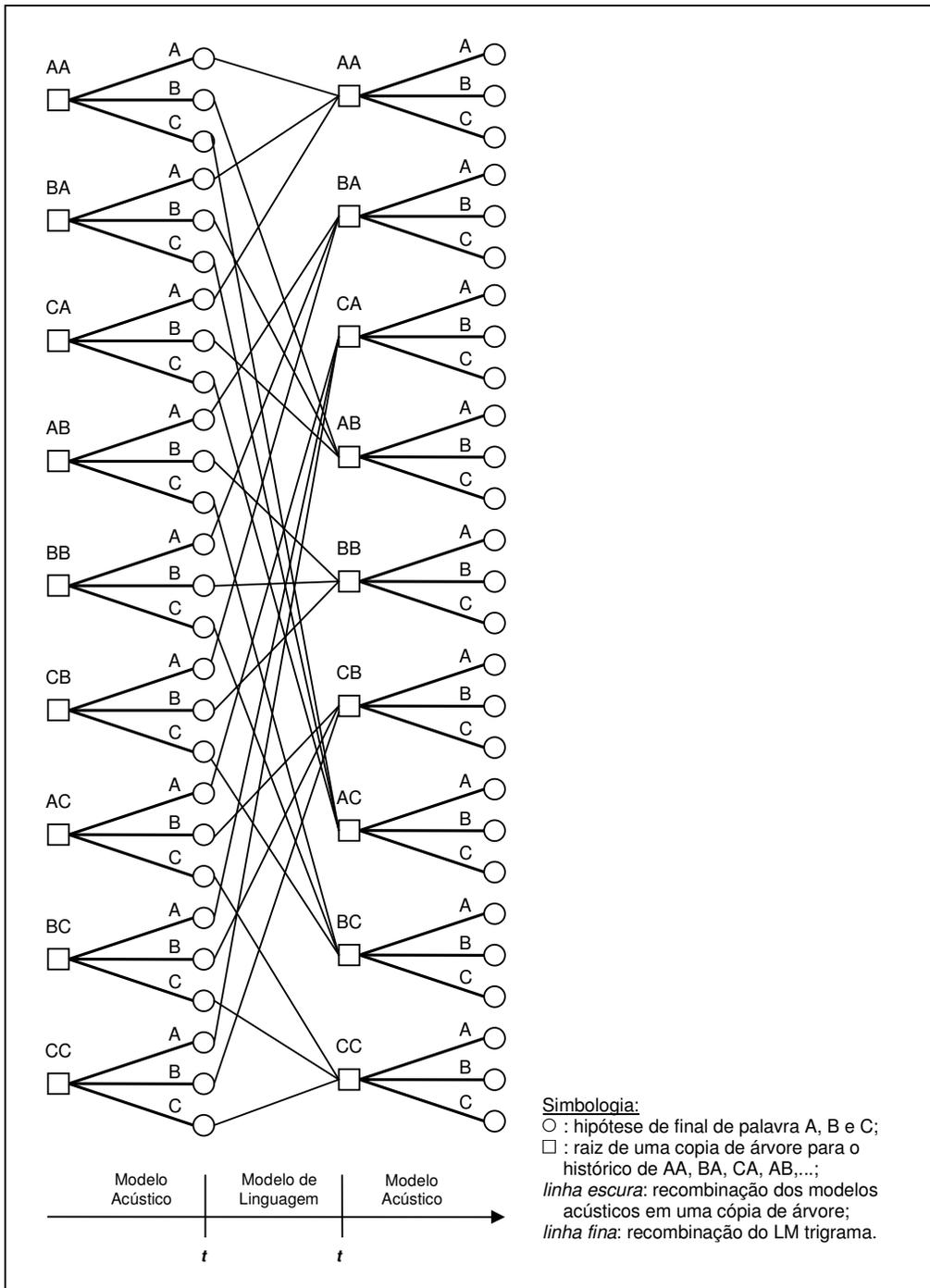


Figura 21: LM trígama em um léxico em árvore (vocabulário de 3 palavras: A, B e C)

### 3.5. Estratégia de Poda no Algoritmo One Pass

Conforme mencionado, a busca completa no espaço de estados é proibitiva. Para resolver esta questão usa-se a estratégia de busca em feixe síncrona no tempo, onde, para cada segmento de tempo, apenas as hipóteses mais promissoras são retidas. A abordagem de podar caminhos consiste em três passos que são executados em cada segmento de tempo, como descrito a seguir [97].

### 3.5.1. Poda Acústica

A poda acústica é usada para reter em processamentos subseqüentes apenas as hipóteses de estado cuja pontuação esteja próxima à melhor pontuação entre todas as hipóteses. A melhor pontuação entre hipóteses de estado,  $Q_{AC}(t)$ , é obtida através da seguinte equação:

$$Q_{AC}(t) = \max_{(v,s)} \{Q_v(t,s)\} \quad (33)$$

Um estado sob hipótese  $(t,s;v)$  é podado se:

$$Q_v(t,s) < f_{AC} Q_{AC}(t) \quad (34)$$

onde  $f_{AC}$  é chamado de limiar da poda acústica e é responsável pelo controle da largura do feixe de busca, ou seja, do número de hipóteses de estados remanescentes.

### 3.5.2. Poda com Modelo de Linguagem

A poda com modelo de linguagem (ou poda de início de palavra) é aplicada apenas na inicialização de hipóteses de árvores, da seguinte forma: nas hipóteses de início de palavras, a probabilidade do modelo de linguagem é incorporada à pontuação acumulada, segundo a Equação (31), e a melhor pontuação de cada palavra predecessora selecionada é usada para inicializar uma correspondente hipótese de árvore ou é propagada por essa árvore, caso a mesma já exista. As pontuações dessas hipóteses de árvore são, então, testadas por um passo adicional de poda:

$$Q_{LM}(t) = \max_v \{Q_v(t,s=0)\} \quad (35)$$

onde  $s=0$  é o estado fictício da raiz da árvore usado para inicialização. Dessa forma, a inicialização da hipótese de árvore  $(t,s=0;v)$  é removida da busca se:

$$Q_v(t,s=0) < f_{LM} Q_{LM}(t) \quad (36)$$

onde  $f_{LM}$  é conhecido como limiar da poda com modelo de linguagem.

### 3.5.3. Poda por Histograma

A poda por histograma limita o número de hipóteses de estados remanescentes em um número máximo,  $M_{Sta}$ . Se o número de estados ativos é maior que  $M_{Sta}$ , então apenas as melhores  $M_{Sta}$  hipóteses são retidas, enquanto as outras são removidas da busca. Esse método de poda é chamado de poda por histograma porque se utiliza um histograma das pontuações dos estados ativos na busca [97].

## 4. Implementação de um Sistema CSR a partir do HTK

As técnicas de reconhecimento de fala contínua discutidas neste trabalho foram estudadas e testadas com o auxílio das ferramentas e bibliotecas disponibilizadas nos pacotes do HTK e do ATK citados na Seção 2.3. O HTK foi selecionado por ser uma ferramenta simples, com farta documentação e amplamente utilizada na área. Sua utilização foi direcionada ao treinamento de modelos acústicos e lingüísticos. O ATK foi utilizado para o teste, pois além de apresentar desempenho superior ao HTK em alguns testes realizados, possui uma arquitetura orientada a objeto desenvolvida para operar em tempo real e oferece suporte a trigramas. Com essa estratégia, possibilita-se a implementação posterior de um sistema LVCSR em tempo real, dado o suporte oferecido pelas bibliotecas do ATK.

A Figura 22, extraída do “*HTK Book*”, apresenta uma proposta de treinamento e teste de um sistema CSR. As subseções seguintes irão descrever os passos indicados na figura, além de outros não representados, como a geração do léxico e o treinamento do modelo de linguagem. A razão destes passos não estarem de forma clara nesta figura se dá porque, normalmente, pesquisadores que trabalham, principalmente, com o idioma inglês, encontram estes elementos já prontos em grandes bases públicas. Há ainda o problema da acentuação da língua portuguesa com algumas ferramentas, como por exemplo, a que gera as redes de busca a partir de um conjunto de regras gramaticais (*HParse*). Esta ferramenta apresenta problemas com palavras do léxico que possuam acentuação. Este e outros problemas tiveram que ser contornados conforme a descrição a seguir, a qual foi baseada no “*HTKBook*” e em listas de discussão na Internet para usuários do HTK.

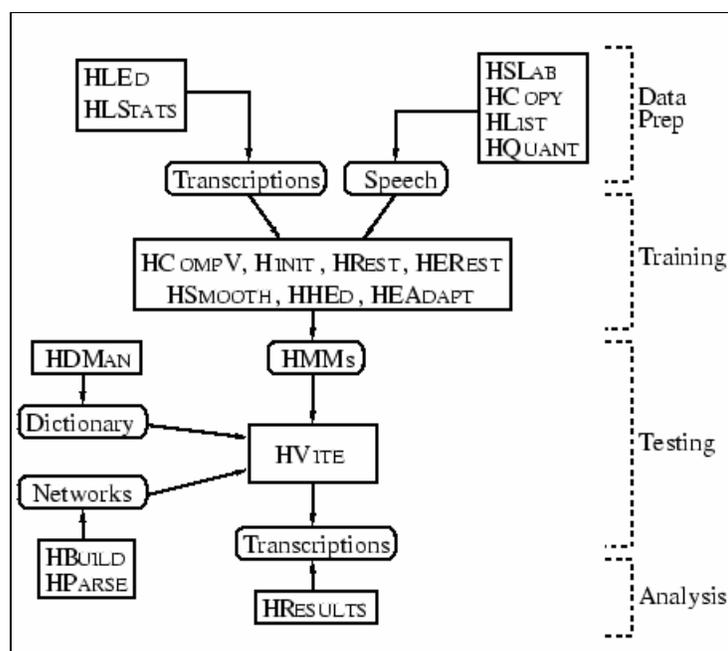


Figura 22: Estrutura de ferramentas disponíveis no pacote HTK para testes de sistemas CSR [13].

Nas descrições que seguem serão sempre listadas as ferramentas do HTK utilizadas para se alcançar certo objetivo. Este trabalho não tem o intuito de descrever estas ferramentas, e sim de fazer apenas uma breve referência às mesmas, de forma a se endereçar a forma como foram solucionados os problemas.

## 4.1. Preparação de Dados

Antes de iniciar o treinamento e teste do sistema, alguns arquivos precisam ser preparados, pois serão posteriormente utilizados. Os principais arquivos são:

- Lista dos arquivos de som que serão usados no treinamento e lista com as frases correspondentes.
- Lista dos arquivos de som que serão usados no teste e lista com as frases correspondentes.
- Léxico (dicionário com transcrições das palavras que constituem o vocabulário do sistema) e lista de fonemas.
- Lista com as frases para treinamento do modelo de linguagem.

Ao léxico devem ser acrescentadas duas novas entradas referentes ao início e fim de frases, normalmente representadas pelos símbolos <s> e </s>, respectivamente. Fora isso, todas as transcrições fonéticas do léxico devem ter ao seu final acrescentado o símbolo “sp”, de forma a possibilitar a ocorrência de silêncio entre palavras, como no exemplo abaixo:

```
...  
TESE t e z i sp  
TETO t e t u sp  
TETRA t e t r a sp  
...  
<s> sil  
</s> sil
```

## 4.2. Treinamento do Modelo Acústico

O treinamento acústico possui uma variante no seu processo que é a decisão entre considerar ou não trifones entre-palavras. Caso a decisão seja de não considerar, devem apenas ser treinados os trifones intrapalavra que aparecem na base de treino e de teste. Os trifones podem ser considerados a partir das transcrições fonéticas existentes no léxico. Trifones que não existam nas bases de treino e de teste são sintetizados a partir de uma árvore de decisão. No caso de considerarem-se os trifones entre-palavras, todas as possibilidades de combinação de fonemas são consideradas durante a sintetização dos modelos (para 38 fonemas, por exemplo, existem  $38^3=54872$  possibilidades de trifones). Os passos usados para treinamento do modelo acústico são descritos a seguir (Figura 23):

- Extração de parâmetros dos arquivos de fala, separados para o treinamento do sistema. Para cada arquivo é gerado um outro correspondente contendo os vetores de parâmetros calculados.

Ferramentas HTK: HCopy

- Inicialização dos modelos de fonemas (sem contexto). Primeiramente, as frases de teste são transcritas com os fonemas que compõem cada palavra. Nesse ponto, utiliza-se a técnica denominada *Flat Start*, para segmentação automática da base. Essa técnica consiste, basicamente, em três passos: o primeiro é desconsiderar as pausas existentes entre palavras para gerar uma primeira estimativa dos modelos de fonemas. Depois se cria um modelo de pausa a partir do modelo do silêncio de início e final de frase e reestimam-se

os modelos. Por último, executa-se um realinhamento da base de treino em função dos modelos e estes são novamente estimados.

Ferramentas HTK: *HLEd, HCompV, HERest, HHEd, HVite*  
Scripts usados: *makedefs, InsertSpHmdefs.*

- Conversão dos modelos de fonemas para trifones. Inicialmente as frases de treino são novamente transcritas, porém agora não mais com fonemas e sim com trifones, levando-se ou não em conta, os trifones entre-palavras. Os fonemes centrais dos trifones passam a referenciar os fonemas treinados anteriormente.

Ferramentas HTK: *HLEd, HHEd, HERest*  
Scripts usados: *maketrihed.*

- Compartilhamento de estados a partir de árvores de decisão. Criação de lista com todos os trifones necessários ao modelo acústico. No caso de considerarmos apenas trifones intrapalavra, o léxico utilizado deve conter tanto as palavras de treino quanto de teste. No caso de trifones entre-palavras serem considerados, a lista irá conter todas as possibilidades de trifones a partir da combinação de fonemas. Para criação da árvore de decisão é necessária a criação de uma lista de perguntas específicas para os fonemas que estão sendo utilizados e para o idioma em questão [98]. No final é gerado um arquivo mapeando modelos que compartilham distribuições.

Ferramentas HTK: *HHEd, HERest*  
Scripts usados: *CreateFullListWI, CreateFullList, mkclscript.*

- Aumento do número de Gaussianas dos modelos. Os passos anteriores geram modelos de trifone que contém apenas uma gaussiana, usada para gerar as probabilidades de emissão de saída em cada estado compartilhado. A técnica usada para aumentar o número de gaussianas das misturas consiste em retreinar os modelos para cada nova gaussiana adicionada.

Ferramentas HTK: *HHEd, HERest*

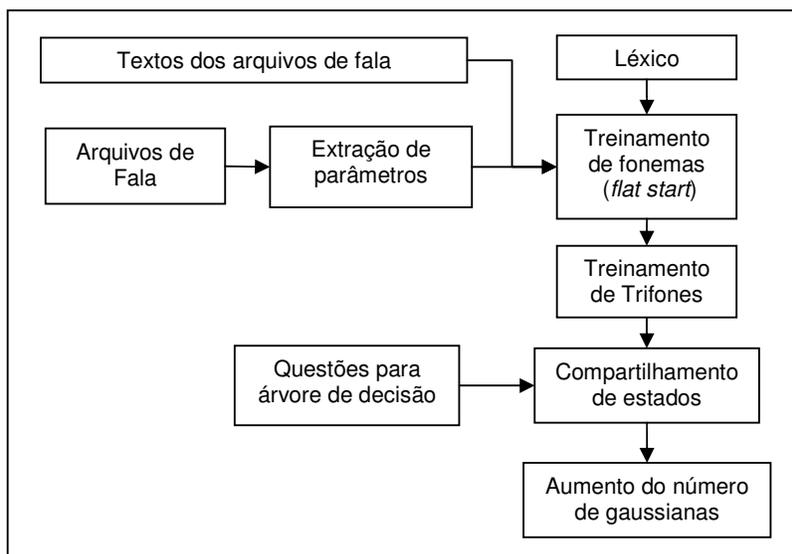


Figura 23: Processo de treinamento dos modelos acústicos

### 4.3. Treinamento do Modelo de Linguagem

O HTK contém um pacote de ferramentas destinadas ao treinamento, teste e análise de modelos de linguagem do tipo  $n$ -grama, denominado HLM (HTK *Language Model*). Para o treinamento do modelo de linguagem são requeridos o dicionário de teste do sistema CSR e um arquivo com as frases de onde serão extraídas as freqüências de palavras e o correspondente relacionamento entre elas. O dicionário de teste (léxico) é requerido visto que palavras encontradas nas frases de treino que não estejam no dicionário serão consideradas fora do vocabulário (OOV – *Out of Vocabulary*) pelo modelo de linguagem e não terão suas freqüências e relacionamentos contabilizados. É de fundamental importância que o arquivo de frases tenha os marcadores de início e final de sentença (<s>,</s>) em cada sentença, pois os mesmos serão contabilizados no treinamento do modelo, tal como palavras. Também é importante que sejam suprimidas as pontuações do texto e que números e siglas estejam adequadamente tratados. A seguir serão descritos os passos necessários para geração de modelos de linguagem do tipo unigrama, bigrama e trigrama e as correspondentes ferramentas do HTK utilizadas no processo de treino e teste do modelo (Figura 24):

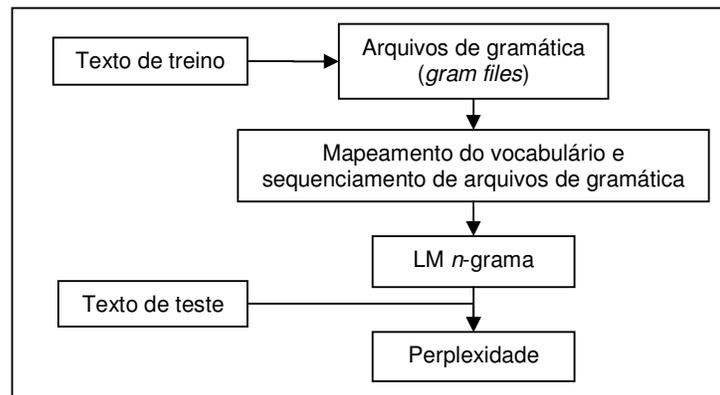


Figura 24: Processo de treinamento e teste de um modelo de linguagem.

- Contagem das freqüência de todas as palavras existentes nas frases de treino e dos relacionamentos entre cada palavra com até duas palavras anteriores (trigrama), incluindo o início e final de frase. Este dados são armazenados em arquivos denominados como gramaticais (*gram files*).

Ferramentas HTK: *LNewMap*, *LGPprep*, *LGCcopy*

- Identificação de palavras OOV. De todas as palavras identificadas no passo anterior, aqueles que não estão presentes no léxico do sistema são consideradas OOV e portanto excluídas do treinamento. Novos arquivos gramaticais são gerados depois desse processo.

Ferramentas HTK: *LSubset*

- Treinamento dos  $n$ -gramas. Com base nos arquivos gramaticais gerados no passo anterior, os modelos de unigrama, bigrama e trigrama podem ser computados, incluindo tanto as probabilidades de seqüências de palavras, bem como os fatores de escalamento usados para *back-off*, conforme a Seção 2.1.3.

Ferramentas HTK: *LBuild*, *LNorm*

- Teste dos  $n$ -gramas. Sempre que um modelo  $n$ -grama é gerado, é importante medir a perplexidade dele em função de uma base de teste, conforme visto na Seção 2.1.3.1.

Ferramentas HTK: *LPlax*

#### 4.4. Reconhecimento de Fonemas

Para o conversor de seqüência fonética em texto que será proposto no Capítulo 5, é necessário ter como entrada uma seqüência de fonemas reconhecidos por um sistema de reconhecimento de voz. Para implementação do reconhecimento de fonemas utiliza-se o treinamento acústico descrito anteriormente, configurado para considerar trifones entre-palavras. A razão disto deve-se ao fato de se desejar obter na saída do sistema qualquer seqüência de fonema, independente das seqüências permitidas pelo léxico de palavras. A seguir serão descritos os passos necessários para se realizar um reconhecimento de fonemas, a partir de uma entrada de voz, utilizando as ferramentas disponíveis no HTK (Figura 25):

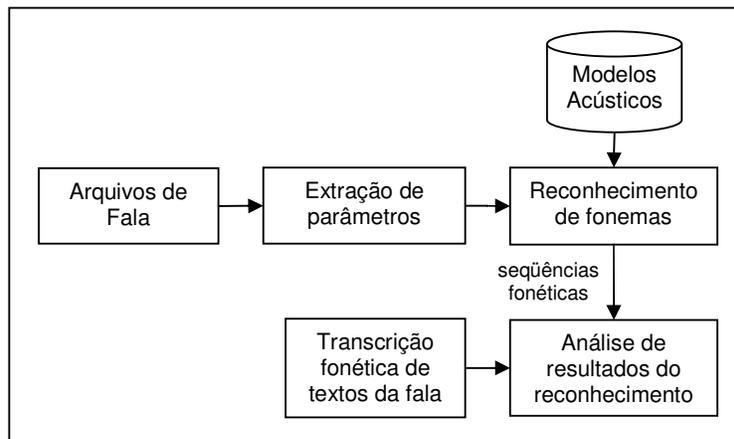


Figura 25: Implementação de um sistema de reconhecimento de fonemas

- Extração de parâmetros dos arquivos de fala, separados para o teste do sistema. Para cada arquivo é gerado um outro correspondente contendo os vetores de parâmetros calculados.

Ferramentas HTK: *HCopy*

- Reconhecimento dos fonemas. Por simplicidade, usou-se o HTK no lugar do ATK, dado que os testes não envolviam modelo de linguagem  $n$ -grama. Um dicionário e uma rede de fonemas, requeridos pelo decodificador, precisaram ser construídos. O dicionário de fonemas é um simples mapeamento entre um fonema e ele próprio. Isso indica ao decodificador quais devem ser as saídas do sistema. A rede de fonemas indica as transições possíveis que o decodificador pode fazer entre um fonema e outro.

Ferramentas HTK: *HParse, HVite*

Scripts usados: *makedict*

- Transcrição fonética de textos da fala. A transcrição fonética dos textos é baseada nas transcrições de suas palavras que são recuperadas a partir do léxico do sistema.

Ferramentas HTK: *HLEd*  
Scripts usados: *prompts2mlf*.

- Análise de resultados. Para a análise de resultados, utilizou-se a ferramenta do HTK que realiza um alinhamento entre as seqüências retornadas pelo decodificador e os textos transcritos das frases. Esse alinhamento gera uma taxa de erro de fonemas, conforme a *WER* definida em (21).

Ferramentas HTK: *HResults*

#### 4.5. Reconhecimento de Palavras

Para implementação do reconhecimento de palavras utiliza-se o treinamento acústico e modelo de linguagem descritos anteriormente. A seguir serão descritas as etapas de reconhecimento de palavras e análise de resultados, a partir de uma entrada de voz, utilizando as ferramentas disponíveis no HTK e ATK (Figura 26):

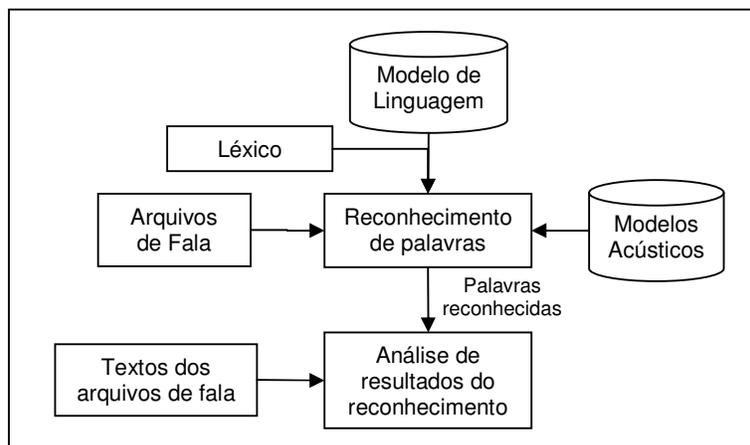


Figura 26: Implementação de um sistema de reconhecimento de palavras

- Para o reconhecimento de palavras é necessária a construção de uma rede de palavras, a qual é gerada a partir do modelo de linguagem e do léxico do sistema. Não é necessária a extração prévia de parâmetros da fala, pois este processamento é feito internamente pelo decodificador, o qual foi projetado para operar em tempo real. No entanto, deve-se fornecer um vetor contendo uma média inicial para os parâmetros, conforme a Equação (3).

Ferramentas HTK: *HBuild* (*HCopv* e *HCompV* usados para gerar vetor inicial)  
Ferramentas ATK: *AVite*

- Para a análise de resultados, utilizou-se a ferramenta do HTK que realiza um alinhamento entre as palavras reconhecidas pelo decodificador e os textos das frases. Esse alinhamento gera uma *WER*, conforme definição (21).

Ferramentas HTK: *HResults*  
Scripts usados: *prompts2mlf*.

## 5. Proposta de Algoritmo: Conversor Fonema-Grafema

A grande maioria dos sistemas de reconhecimento de voz como o HTK [13], SPHINX [52] e ISIP [53], já citados anteriormente, bem como outros tantos como IBM [99] e WHISPER [100], resolvem a tarefa de reconhecimento em apenas um estágio sem obter, como resultado intermediário, uma seqüência fonética (o conceito de estágio utilizado aqui difere do conceito de múltiplas passadas descrito na Seção 3.1.3). Uma maneira de se obter este resultado intermediário é através da utilização de uma configuração híbrida, onde se utilizam HMMs e decodificação de Viterbi para se realizar o reconhecimento de uma seqüência fonética (Seção 4.4) e então, utiliza-se DTW [101] ou *One-Stage* [76] para gerar uma seqüência com os grafemas correspondentes [102], [103]. A idéia de identificar inicialmente os segmentos da fala com seus respectivos fonemas já foi utilizada antes, porém em sistemas fundamentalmente baseados na análise fonética de segmentos e não no reconhecimento baseado em HMMs [104]. O Conversor Fonema-Grafema (PS2G - *Phonetic Sequence to Graphemes*) apresentado nesta seção, elimina o acoplamento entre modelos acústico e lingüístico e, por essa característica, apresenta a flexibilidade de ter de um lado um reconhecedor genérico de fonemas e, do outro, um conversor especializado baseado apenas em texto (Figura 27).

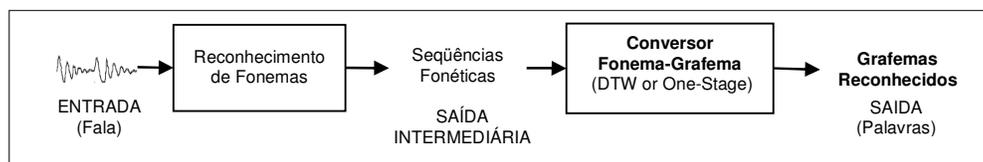


Figura 27: Diagrama de blocos do conversor PS2G.

### 5.1. Conversão Baseada em DTW

As seqüências fonéticas que são recebidas na entrada do conversor PS2G são exatamente as mesmas retornadas pelo reconhecedor de fonemas, como um resultado do processamento de amostras de palavras ou frases fechadas pertencentes ao vocabulário do sistema proposto. Obviamente, numa situação real, a seqüência com os fonemas retornados pelo reconhecedor contém erros em relação à sua transcrição ideal. As seqüências podem apresentar fonemas substituídos, inseridos ou mesmo apagados e é o papel do conversor PS2G reconhecer os grafemas corretos mesmo tendo como entrada seqüências com imperfeições.

Para resolver este problema, implementou-se uma variação do algoritmo DTW, usado usualmente em tarefas de reconhecimento de palavras isoladas. A idéia de se usar o DTW nestes sistemas, vem da capacidade que este algoritmo apresenta de alinhar diferentes sons no eixo do tempo e calcular uma medida de distância (ou custo) entre os mesmos, que é então utilizada para classificar as amostras em função de sua similaridade com os modelos treinados *a priori* (referências) [50].

No algoritmo DTW tradicional, a distância de Itakura ou Euclidiana é calculada entre um vetor de coeficientes do segmento  $i$  do som de teste, e um vetor  $j$  do som de referência [101]. A modificação implementada no conversor PS2G, para o cálculo de distâncias, associa uma estimativa da similaridade entre o fonema  $i$  da palavra de teste e o fonema  $j$  da palavra de referência, conforme a Tabela 5. Essa tabela mostra que fonemas idênticos não acrescentam nenhum custo ao cálculo de distância. Essa relação, no entanto, pode ser modificada de forma a acrescentar valores

intermediários como resultado da comparação entre fonemas distintos, porém, que apresentem uma sonoridade próxima.

Tabela 5: Distância entre os fonemas de seqüências de teste e de referência (similaridade).

Comparação de Fonemas	Distância Retornada
Fonemas são iguais	0
Fonemas são diferentes	1

Tendo o cálculo de similaridade entre fonemas, resolvido conforme a Tabela 5, o resto do algoritmo DTW segue como se estivesse comparando duas seqüências de sons (na verdade, vetores com coeficientes). Após todas as distâncias terem sido calculadas entre os fonemas das seqüências de teste e de referência, pode-se classificar qual delas mais se assemelha com o som de teste (aquele que gerou a seqüência fonética no reconhecedor de fonemas). Um exemplo deste “alinhamento fonético” pode ser visto na Figura 28 com a palavra “congresso”. Este exemplo mostra o alinhamento entre a seqüência fonética obtida como resultado de uma amostra da palavra “congresso” e a sua transcrição ideal. Os eventuais erros contidos nesta seqüência são tratados adequadamente pelo conversor PS2G.

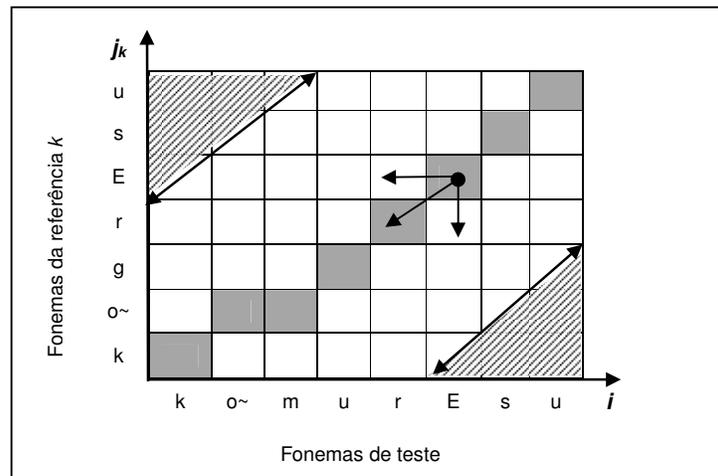


Figura 28: Exemplo de “alinhamento fonético” implementado no algoritmo DTW, durante o reconhecimento da palavra “congresso” com as restrições locais e globais implementadas.

Com respeito ao algoritmo DTW, além de se implementar as restrições de caminho locais definidas por

$$\left( \begin{array}{l} \text{caminho parcial} \\ \text{para o nó } (i, j_k) \end{array} \right) = D(i, j_k) = \min \left\{ \begin{array}{l} D(i-1, j_k) - \text{esquerda} \\ D(i-1, j_k-1) - \text{diagonal} \\ D(i, j_k-1) - \text{inferior} \end{array} \right\} + d(i, j_k), \quad (37)$$

restrições globais [101] também foram implementadas. As restrições locais controlam as alternativas de caminho e custos associados, para se chegar ao nó  $(i, j_k)$ , onde  $k$  representa o índice das referências. Ambas as restrições estão representadas na Figura 28. Na Equação (37),  $D(i-1, j_k)$  é a distância (ou custo) de se chegar do nó  $(1, 1)$  ao nó  $(i-1, j_k)$  do plano  $k$ . Este nó está a esquerda do nó  $(i, j_k)$  e  $d(i, j_k)$  é a distância calculada (similaridade) entre os fonemas  $i$  e  $j_k$ , conforme descrito na Tabela 5.

As restrições globais bloqueiam o algoritmo de alinhamento de tentar seguir caminhos que passem por regiões muito próximas ao canto superior esquerdo e ao

canto inferior direito do plano  $k$ . As áreas das regiões bloqueadas, quando somadas, se equivalem à área de um quadrado, cujos lados são iguais à metade do tamanho da menor das seqüências, entre teste e referência, com  $I$  e  $J_k$  fonemas cada, respectivamente. Na prática, se  $I > J_k$ , a área de busca é limitada por duas linhas. A primeira conecta o nó  $(1, J_k - J_k/2)$  ao nó  $(J_k/2, J_k)$  e a segunda, conecta o nó  $(I - J_k/2, 1)$  ao nó  $(I, J_k/2)$ , conforme apresentado na Figura 28. Se  $I < J_k$ , substitui-se  $J_k/2$  por  $I/2$  nos nós descritos anteriormente.

Depois que todos os nós  $(i, j_k)$  tenham sido calculados, para todas as  $k$  referências presentes no dicionário (satisfeitas as restrições globais), um procedimento de percurso reverso (*backtrack*) é iniciado em cada plano, partindo do nó  $(I, J_k)$  até chegar ao nó  $(1,1)$  do mesmo plano, mantendo-se a contagem do número de saltos ou transições realizadas no percurso, definida como  $salto_{s_k}$ . Cada salto realizado no percurso é regido pelas mesmas regras apresentadas em (37). A referência selecionada é a que possui seqüência fonética mais similar à seqüência de entrada (ou teste). Isto é revelado com a minimização de

$$K_{selecionado} = \arg_k \min \left( \frac{D(I, J_k)}{salto_{s_k}} \right), \quad (38)$$

onde a seqüência selecionada é a que possui índice igual a  $K_{selecionado}$ .

## 5.2. Conversão Baseada em One-Stage

Em tarefas de reconhecimento de fala contínua, o objetivo do conversor PS2G é detectar, em uma seqüência fonética de entrada, as partes que são mais similares às referências disponíveis no dicionário do sistema. As referências confrontadas são então retornadas como os grafemas reconhecidos pelo conversor. Se não acontecerem erros nesse processo, estes grafemas correspondem às palavras contidas na sentença, cuja amostra serviu de entrada para o reconhecedor de fonemas. A solução para este problema de conversão pode ser obtida com uma implementação do algoritmo *One-Stage* [76] proposto por Ney, na década de 80, para resolver eficientemente o problema de reconhecimento de palavras conectadas e que tem suas bases no algoritmo DTW. O algoritmo *One-Stage* é o precursor do algoritmo *One-Pass* descrito na Seção 3.3. Alguns anos antes de surgir o algoritmo *One-Stage*, um outro método proposto por Myers e Rabiner, denominado *Level Building* [48] [50], também foi utilizado para resolver o mesmo problema, no entanto, por alguns motivos práticos, como não ser síncrono no tempo e ter de fixar a quantidade de palavras da frase reconhecida, este algoritmo acabou caindo em desuso [105].

A vantagem de se utilizar o algoritmo *One-Stage* é sua habilidade em realizar três importantes operações simultaneamente: detecção de fronteiras, alinhamento não-linear no tempo e reconhecimento de palavras. Para que as seqüências fonéticas de teste e referências possam ser alinhadas,  $K$  planos com dimensões  $I \times J_k$  devem ser criados, onde  $K$  é o número de referências presentes no dicionário do sistema,  $I$  é o número de fonemas na seqüência de teste e  $J_k$  é o número de fonemas na da referência de índice  $k$ . Durante esse alinhamento, dois tipos de transição são permitidos entre nós  $(i, j, k)$ . O primeiro tipo de transição é bastante similar à restrição local descrita na seção anterior, para a implementação baseada em DTW. É permitida apenas para nós com índice  $j > 1$  (após o início da referência) e representa possíveis transições entre nós de um mesmo plano localizados a esquerda, abaixo e numa diagonal inferior à esquerda, conforme segue:

$$\left( \begin{array}{l} \text{caminho parcial} \\ \text{para o nó } (i, j, k) \\ \text{onde } j > 1 \end{array} \right) = D(i, j, k) = \min \left\{ \begin{array}{l} D(i-1, j, k) - \text{esquerda} \\ D(i-1, j-1, k) - \text{diagonal} \\ D(i, j-1, k) - \text{abaixo} \end{array} \right\} + d(i, j, k) \quad (39)$$

O segundo tipo de transição pode acontecer em nós com  $j = 1$  (fronteiras entre referências). Estes nós podem aceitar transições a partir de nós situados a esquerda do mesmo plano ou de nós situados também a esquerda, porém no extremo superior de outros planos, significando uma transição entre referências, conforme declarado a seguir:

$$\left( \begin{array}{l} \text{caminho parcial} \\ \text{para o nó } (i, j, k) \\ \text{onde } j = 1 \end{array} \right) = D(i, j, k) = \min \left\{ \begin{array}{l} D(i-1, j, k) - \text{esquerda} \\ \min_{k_2} (D(i-1, j_{k_2}, k_2)) \end{array} \right\} + d(i, j, k) \quad (40)$$

Da mesma forma como descrito na seção anterior, a distância local  $d(i, j, k)$  é calculada conforme a Tabela 5. Um exemplo destas transições (dentro e entre palavras) está ilustrado na Figura 29.

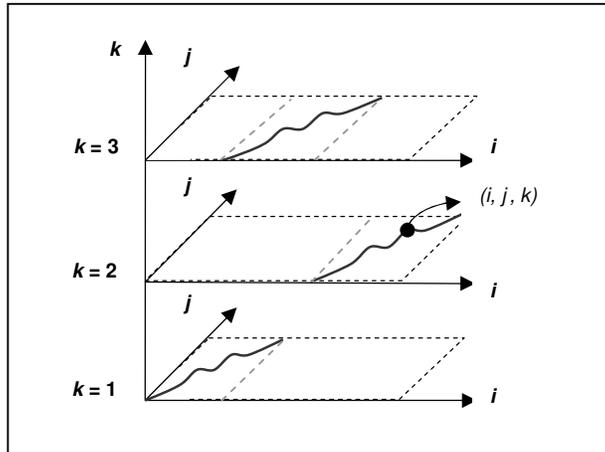


Figura 29: Exemplo de “alinhamento fonético” implementado através do algoritmo *One-Stage*. Os grafemas reconhecidos correspondem às referências com índices 1, 3 e 2. Outros planos de referências que não fizeram parte da solução, não aparecem na figura.

A minimização vista em (40) pode também levar em consideração a probabilidade de transição da referência  $k_2$  para referência  $k$ ,  $P(k | k_2)$ , baseando-se em sua ocorrência previamente computada a partir de um texto de treino. Esta probabilidade é conhecida como bigrama e pode ser implementada, de forma simples, conforme equação a seguir [39]:

$$P(k | k_2) = \frac{1 + C(k_2, k)}{V + \sum_k C(k_2, k)} \quad (41)$$

onde  $C(k_2, k)$  é o número de vezes que a transição entre  $k_2$  e  $k$  é vista no texto de treino e  $V$  é o tamanho do vocabulário do sistema. Tendo estes dados calculados de antemão, a minimização presente em (40), relacionada a  $k_2$ , é modificada da seguinte maneira:

$$D(i, j, k) = \min \left\{ \begin{array}{l} D(i-1, j, k) - \text{esquerda} \\ \min_{k_2} \left( D(i-1, J_{k_2}, k_2) + lm(k, k_2) \right) \end{array} \right\} + d(i, j, k) \quad (42)$$

onde,

$$lm(k, k_2) = -\log(P(k | k_2)) - \min_{k, k_2} (-\log(P(k | k_2))) . \quad (43)$$

O operador logarítmico é usado em (43), visto que as probabilidades calculadas a partir de (41) são muito baixas. A minimização é subtraída de forma a ajustar o custo dos bigramas mais prováveis como próximos à zero, o que significa uma não penalização no cálculo de distância, nestes casos.

Depois de terminados os cálculos de todos os nós de cada plano  $k$ , o algoritmo procura pelo nó  $(I, J_k, k)$  com a menor distância acumulada  $D(I, J_k, k)$ . Deste ponto em diante, um procedimento de caminho reverso (*backtrack*) é executado até que um nó da forma  $(1, 1, k)$  seja encontrado [76]. Todas as referências  $k$ , onde o procedimento de caminho reverso passa, são gravadas e depois invertidas gerando a seqüência dos grafemas reconhecidos pelo algoritmo.

## 6. Preparação de Bases para o Português Brasileiro

A disponibilidade de bases de dados é fator condicionante para o desenvolvimento de sistemas CSR. Bases de dados compartilhadas entre pesquisadores americanos e europeus foram um dos grandes motivos para o progresso obtido ao longo das últimas décadas nestas regiões. Para o português brasileiro, no entanto, ainda há uma grande carência de bases comuns, de forma que apenas iniciativas individuais de centros de pesquisa são eventualmente encontradas [105], [106]. Todavia, nenhuma dessas iniciativas atende completamente os requisitos de treinamento necessários para sistemas LVCSR em nosso idioma. Bases de dados para treinamento e teste destes sistemas são normalmente subdivididas em 3 partes, cada qual com sua finalidade:

- Arquivos de fala (áudio) com textos
- Massa de textos escritos
- Dicionário fonético

Pelas razões expostas e pelo fato de já existir uma base com arquivos de áudio disponibilizados pelo Prof. Fernando Gil Resende, optou-se, nesta dissertação, por aproveitar esse material de forma a constituir uma nova base de dados para o desenvolvimento de sistemas LVCSR. A seguir, descreve-se com detalhe como se projetou cada um dos 3 grupos mencionados acima (Figura 30).

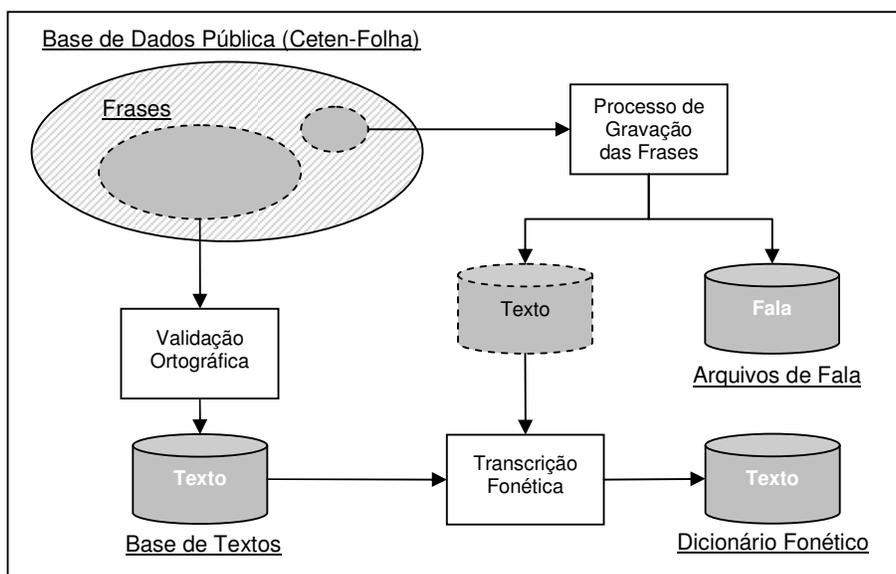


Figura 30: Relacionamento entre as Bases de Dados utilizadas.

### 6.1. Arquivos de Fala

Os arquivos de fala são utilizados para treinamento dos modelos acústicos e testes do sistema CSR. Bases de fala são normalmente agrupadas em bases geradas a partir de texto lido (*Read Speech*) ou através de gravações de fala espontânea (*Spontaneous Speech*). Neste trabalho, utilizaram-se 1000 frases gravadas em estúdio através da leitura de textos, por um mesmo locutor. Os textos utilizados foram extraídos de uma base de dados pública [107] de forma a se obter um balanceamento

fonético<sup>5</sup> para o português brasileiro [108]. A segmentação da base em frases, foi feita manualmente, dado que na gravação original, criou-se um arquivo de áudio único para cada grupo de 20 frases. Os arquivos foram todos gravados num formato “wav”, amostrados com 16 bits a 48kHz e em apenas um canal. Quando necessário, os textos foram modificados para refletir o que de fato foi falado pelo locutor. Nenhuma outra segmentação manual foi realizada, do contrário, vimos, na Seção 4.2, que durante o treinamento do modelo acústico, a base foi segmentada através da técnica denominada *flat start*.

Apesar de a base ter apenas um locutor, a identificação adotada para cada um dos 1000 arquivos já prevê a possibilidade de expansão para vários locutores e é realizada da seguinte maneira:

CD<sub>1</sub>D<sub>2</sub>D<sub>3</sub>D<sub>4</sub>D<sub>5</sub>D<sub>6</sub> (ex.: M010216), onde “C” simboliza um caractere e “D”, um dígito.

*C*: Identificação se o locutor é masculino (M) ou feminino (F)

*D<sub>1</sub>D<sub>2</sub>*: Identificação do locutor na base (01-99)

*D<sub>3</sub>D<sub>4</sub>*: Identificação do grupo a que a frase pertence (01-50)

*D<sub>5</sub>D<sub>6</sub>*: Identificação da frase no grupo (01-20)

No exemplo dado (M010216), a identificação corresponde a 16ª frase, da 2ª lista do primeiro locutor masculino.

## 6.2. Textos Escritos

Os textos escritos são utilizados para treinar o modelo de linguagem. Dado que o modelo utilizado é estatístico (*n*-grama), uma grande quantidade de textos foi coletada de forma a permitir uma estimativa do relacionamento existente entre palavras durante a fala. Na base utilizada para obtenção dos textos [107], realizou-se um trabalho de filtragem para que erros ortográficos, gramaticais, marcações incorretas, e estrangeirismos da base, entre outros erros, não prejudicassem os modelos. Para tal, utilizou-se o corretor ortográfico “*br.ispell*” [109] para validar a ortografia das palavras de cada frase analisada. Sempre que alguma palavra na frase não foi encontrada no dicionário do “*br.ispell*” (com tamanho de 273.760 palavras), então a frase como um todo foi descartada. Antes desta análise, no entanto, aplicou-se uma função para transcrever números e ordinais em textos, de forma que a frase:

*<s> gramado realiza de 20 a 23 de outubro o 6º festival do turismo </s>*

foi convertida para:

*<s> gramado realiza de vinte a vinte e três de outubro o sexto festival do turismo </s>*

onde os símbolos “<s>” e “</s>” indicam início e final de frase, respectivamente. Os textos associados aos arquivos de fala também foram monitorados para que não fizessem parte deste conjunto de textos (para posteriormente não mascarar a perplexidade do modelo de linguagem). Com estas restrições, coletou-se um total de 362.117 frases, dentro do conjunto original composto por aproximadamente 1,5 milhão.

---

<sup>5</sup> O termo balanceamento fonético, neste caso, significa que a lista de frases gerada tem uma distribuição fonética similar àquela encontrada na fala espontânea.

### 6.3. Dicionário Fonético

O dicionário fonético utilizado neste trabalho foi criado de forma automatizada por um software de transcrição fonética para o português brasileiro [110]. Essa automatização dinamizou a criação de diferentes dicionários e sua utilização nos testes de diversas configurações do sistema CSR implementado. Na Tabela 6, encontra-se a lista dos fonemas utilizados pelo transcritor e respectivos exemplos de transcrições realizadas. No Apêndice , encontra-se o dicionário referente às 1000 frases da base de áudio com suas respectivas transcrições.

Tabela 6: Lista com os fonemas (40) utilizados no sistema de CSR [98].

Símbolo	Exemplos
Vogais orais (7)	
a E e i O o u	l <u>á</u> pis, j <u>a</u> tob <u>a</u> , <u>á</u> b <u>a</u> co, cap <u>a</u> cete, cab <u>e</u> ça, ca <u>ç</u> a, lu <u>a</u> , ped <u>i</u> a é, mé <u>d</u> ico, paj <u>e</u> , ép <u>i</u> co, Pel <u>e</u> , pe <u>l</u> e, fer <u>r</u> o, vel <u>h</u> o capac <u>e</u> te, res <u>o</u> lver, res <u>o</u> lver, res <u>o</u> lver justi <u>ç</u> a, pa <u>i</u> s, sa <u>i</u> a, l <u>á</u> pis, id <u>i</u> ota, aque <u>l</u> es, ele, pe <u>l</u> e ó <u>o</u> pio, có <u>o</u> pia, jo <u>g</u> os, do <u>ç</u> as, so <u>z</u> inho, fo <u>r</u> te res <u>o</u> lver, jo <u>g</u> o, go <u>l</u> f <u>i</u> n <u>h</u> o, bo <u>l</u> o, co <u>r</u> baia <u>ç</u> u, Raul, cu <u>l</u> pa, ba <u>u</u> , cu <u>r</u> ru <u>u</u> , lo <u>g</u> o, consolo, tijolo
Vogais nasais (5)	
a~ e~ i~ o~ u~	avi <u>ã</u> o, campe <u>ã</u> o, and <u>a</u> r, tam <u>p</u> ar, can <u>ç</u> ão, cam <u>a</u> ent <u>ã</u> o, consci <u>ê</u> ncia, temp <u>o</u> , bem, men <u>o</u> s, dent <u>e</u> n <u>i</u> n <u>h</u> o, t <u>i</u> nt <u>a</u> , lat <u>i</u> na, imp <u>o</u> rta on <u>da</u> , campe <u>õ</u> es, so <u>m</u> os, ho <u>m</u> em, fr <u>o</u> n <u>h</u> a um, muit <u>o</u> , umb <u>i</u> go
Semi-vogais (4)	
w j w~ j~	nata <u>l</u> , fáci <u>l</u> , volt <u>a</u> r, eu, chap <u>e</u> u, qu <u>a</u> se, ja <u>u</u> la fui, pa <u>i</u> , sei, foi, carac <u>ó</u> is, hot <u>e</u> is, micr <u>ó</u> b <u>i</u> o, pátria n <u>ã</u> o, c <u>ã</u> o muit <u>o</u> , bem, parab <u>e</u> ns, comp <u>õ</u> e
Fricativas não vozeadas (3)	
f s S	f <u>e</u> sta, fan <u>f</u> arr <u>ã</u> o, af <u>t</u> a, af <u>l</u> uente sap <u>o</u> , ca <u>ç</u> ar, cresc <u>e</u> r, sess <u>ã</u> o, l <u>á</u> pis, t <u>ó</u> rax, cap <u>a</u> z, disc <u>o</u> , ca <u>ç</u> ca, des <u>ç</u> o, excess <u>o</u> ch <u>ã</u> , x <u>a</u> veco, cachorro
Fricativas vozeadas (3)	
z v Z	ca <u>ç</u> a, co <u>i</u> sa, qu <u>a</u> se, ex <u>a</u> to v <u>o</u> y <u>ó</u> , v <u>a</u> mos, avi <u>ã</u> o geladeira, trovej <u>a</u> r
Africativas (2)	
tS dZ	t <u>i</u> a, pac <u>o</u> te, constitu <u>i</u> nte, T <u>i</u> juca d <u>i</u> a, cid <u>a</u> de, disc <u>o</u>
Plosivas (6)	
b d t k g p	bar <u>b</u> a, abs <u>i</u> nto da <u>g</u> os, cid <u>a</u> de, do <u>m</u> inar, ad <u>m</u> inistrar to <u>d</u> os, pa <u>ç</u> o, constitu <u>i</u> nte ca <u>s</u> a, ca <u>ç</u> ca, qu <u>e</u> ro, quanto qu <u>e</u> rra, ga <u>t</u> o, ag <u>u</u> entar, ag <u>n</u> óst <u>i</u> co pap <u>a</u> i, psic <u>o</u> l <u>ó</u> gico, ap <u>t</u> o, rap <u>t</u> o
Líquidas (5)	
l L R X r	laran <u>ja</u> , pal <u>a</u> f <u>i</u> ta, leit <u>ã</u> o cal <u>h</u> ar, col <u>h</u> eita, mel <u>h</u> or carro, rua, ra <u>t</u> o, carga, ger <u>m</u> e cas <u>a</u> r, cert <u>o</u> , har <u>p</u> a, ar <u>c</u> o car <u>o</u> na, gar <u>o</u> to, fr <u>a</u> ngo, gr <u>a</u> xa, po <u>r</u> exemplo
Consoantes nasais (3)	
m n J	mam <u>a</u> e, em <u>a</u> , em <u>a</u> ncipar, m <u>a</u> rm <u>o</u> ta nome, aten <u>a</u> uar, enca <u>n</u> ação cas <u>i</u> n <u>h</u> a, gal <u>i</u> n <u>h</u> a
Silêncios (2)	
sil sp	silêncio do início / fim de uma locução silêncio no interior de uma locução (pausa)

## 7. Testes e Resultados

O foco dos testes que serão apresentados nesta seção foi o de entender, na prática, como se comportam os principais componentes de um sistema LVCSR: o dicionário fonético, modelo acústico, modelo de linguagem e o decodificador. A correlação existente entre estes componentes e o desempenho apresentado pelo sistema como um todo são analisados, tanto em função do tempo de resposta, quanto da taxa de acerto verificada.

Todos os testes foram realizados num computador de arquitetura Intel® dual processado (Xeon™ 3.0 MHz) com 2 GB de memória RAM. É importante ressaltar que o computador não estava 100% dedicado para esta finalidade, todavia, durante o tempo dos experimentos, não se verificou grande concorrência de outros processos. O sistema operacional utilizado foi o Linux Red Hat versão 3.2.

O software utilizado para realizar o treinamento dos modelos acústico e lingüístico foi o HTK versão 3.3 [13]. Para realização dos testes, utilizou-se o programa AVite, disponibilizado no pacote ATK, versão 1.4.1 [14]. A utilização de ambos está descrita na Seção 4.

A configuração básica utilizada no treinamento e teste do sistema foi idêntica à maioria dos trabalhos vistos na área e é normalmente referida como configuração padrão (*standard*). Abaixo está uma lista que detalha a configuração utilizada:

- Tamanho do segmento de fala considerado: 25ms
- Atualização de segmentos: a cada 10ms (conhecido como *shift*)
- Janelamento dos segmentos: janela de *Hamming*
- Pré-ênfase: 0.97
- Coeficientes computados em cada segmento: *Mel cepstrais*
- Canais no banco de filtros: 26
- Coeficiente de liftragem cepstral: 22
- Total de coeficientes: energia + 12 coeficientes *mel cepstrais* +  $\Delta$  (1ª derivada dos 13 coeficientes) +  $\Delta\Delta$  (2ª derivada). No total, o vetor calculado possui 39 coeficientes.
- Subtração de média espectral (CMN): *running average* (Seção 2.1.1).
- Modelagem acústica: HMMs contínuos
- Unidades acústicas (HMMs): trifones.
- Modelagem dos HMMs: 3 estados com emissão de saída (Seção 2.1.2).
- Compartilhamento de Estados: Árvores de decisão (Seção 2.1.2.1).
- Decodificador: Viterbi *Beam Search* (Seção 3.1.1)

As bases utilizadas foram construídas da forma descrita na Seção 6. A base de fala, composta por 1000 frases, de 9 a 12 palavras, foi testada separando-se 750 frases para treino e 250 para testes. Esta mesma separação foi realizada quatro vezes, dividindo-se a base em grupos de 250 frases e realizando-se as recombinações possíveis entre estes grupos. Os resultados apresentados a seguir representam a média e desvio padrão dos resultados obtidos com cada um dos grupos, assegurando estatisticamente a qualidade dos experimentos (Figura 31).

A seguir, apresentam-se os experimentos realizados, evoluindo-se gradualmente entre um experimento e o seguinte. Em todos os experimentos, o vocabulário mínimo utilizado contém as 3528 palavras que compõem as 1000 frases da base de fala.

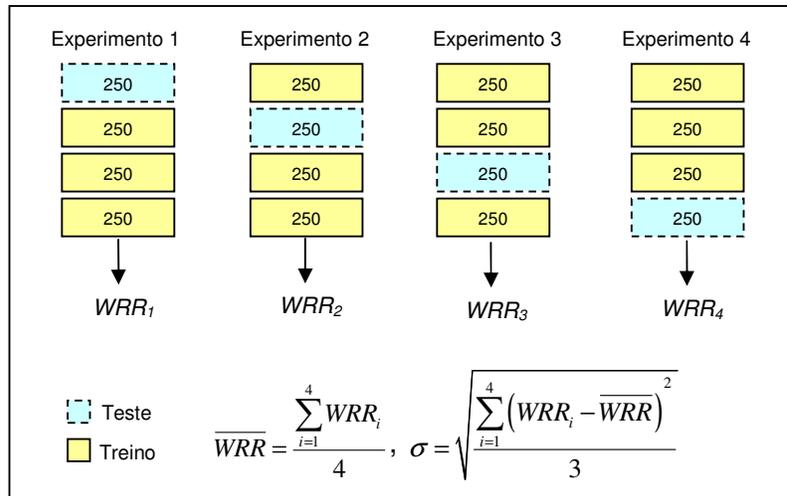


Figura 31: Organização das bases de treino e teste para obtenção de taxas de acerto ( $WRR$ ) média.

### 1) Treino do modelo de linguagem

Nesse experimento avaliou-se a evolução da perplexidade do modelo de linguagem em função do número de frases utilizadas para treiná-lo. O vocabulário foi mantido constante durante todo o experimento. Variou-se o número de frases utilizadas entre 1000 e 350.000 e os modelos de linguagem testados foram o bigrama e o trigrama. As frases utilizadas para medir as perplexidades de cada configuração foram as 1000 frases da base de fala (lembrando que não há intercessão entre esta base e a base de textos). Na tabela abaixo e nas próximas o termo *variação* se refere à variação percentual entre um resultado e o correspondente resultado anterior.

Tabela 7: Medição da perplexidade de modelos treinados com diferentes quantidades de frases.

# de Frases	Modelos		Trigrama		Perplexidade do trigrama em relação ao bigrama
	Bigrama				
1.000	569	<i>variação</i>	564	<i>variação</i>	-0,88%
50.000	388	-32%	347	-38%	-10,57%
100.000	352	-9%	305	-12%	-13,35%
150.000	332	-6%	283	-7%	-14,76%
200.000	321	-3%	269	-5%	-16,20%
250.000	308	-4%	255	-5%	-17,21%
300.000	301	-2%	243	-5%	-19,27%
350.000	298	-1%	236	-3%	-20,81%

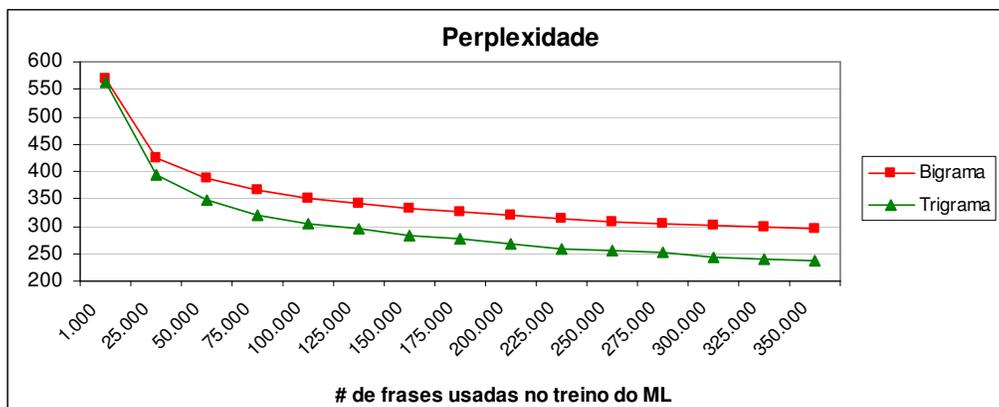


Figura 32: Evolução da perplexidade em função do número de frases utilizadas no treino do LM.

Como esperado, a perplexidade tende a diminuir à medida que o número de frases utilizadas no treino aumenta. Isso se deve ao fato de que a estatística dos modelos treinados melhora conforme mais ocorrências de pares e triplas de palavras são registradas na base de textos. As perplexidades encontradas nestes experimentos, nas fases mais estáveis das curvas apresentadas, estão de acordo com aquelas exibidas nas tabelas da Seção 2.1.3.1. É importante observar na Tabela 7, que a diferença de perplexidade medida para modelos bigrama e trigrama vai crescendo à medida que mais frases são utilizadas no treino. Com 350.000 frases essa diferença ultrapassa 20%.

## 2) Teste do sistema CSR em função de diferentes modelos de linguagem

Neste teste, os modelos treinados no experimento anterior foram usados no sistema CSR, de forma a se avaliar a taxa de acerto de palavras ( $\overline{WRR}$ ) em função do modelo de linguagem. O experimento começou com 1.000 frases da base de texto e terminou com 200.000 frases, pois a  $\overline{WRR}$  parou de evoluir, enquanto o tempo de processamento aumentou. Nestes experimentos utilizou-se apenas trifones intrapalavra e uma única gaussiana nos estados dos HMMs. Testes com modelos de linguagem do tipo unigrama (onde a probabilidade de ocorrência de uma palavra não tem relação com as palavras anteriores) foram incluídos nos experimentos realizados, apenas como comparação com outros modelos e somente nos experimentos até 100.000 frases, dado que os resultados mantiveram-se praticamente constantes nesse intervalo.

Tabela 8: Medição da  $\overline{WRR}$  em função da variação do modelo de linguagem.

# de Frases	Unigrama			Bigrama			Trigrama		
	$\overline{WRR}$	$\sigma$	variação	$\overline{WRR}$	$\sigma$	variação	$\overline{WRR}$	$\sigma$	variação
1.000	64,46%	1,27%	--	67,55%	1,42%	--	67,77%	1,02%	--
25.000	64,42%	1,22%	-0,06%	73,60%	1,85%	8,96%	74,71%	1,68%	10,25%
50.000	64,23%	1,04%	-0,30%	74,82%	1,70%	1,66%	76,25%	1,51%	2,05%
75.000	64,27%	1,07%	0,06%	75,24%	1,49%	0,56%	77,00%	1,61%	0,98%
100.000	64,22%	1,10%	-0,08%	75,92%	1,81%	0,91%	77,42%	1,66%	0,54%
125.000				76,35%	2,02%	0,56%	78,08%	1,62%	0,85%
150.000				76,71%	1,60%	0,47%	78,24%	1,33%	0,21%
175.000				76,67%	1,28%	-0,05%	78,05%	1,25%	-0,24%
200.000				76,86%	1,29%	0,25%	78,40%	1,24%	0,45%

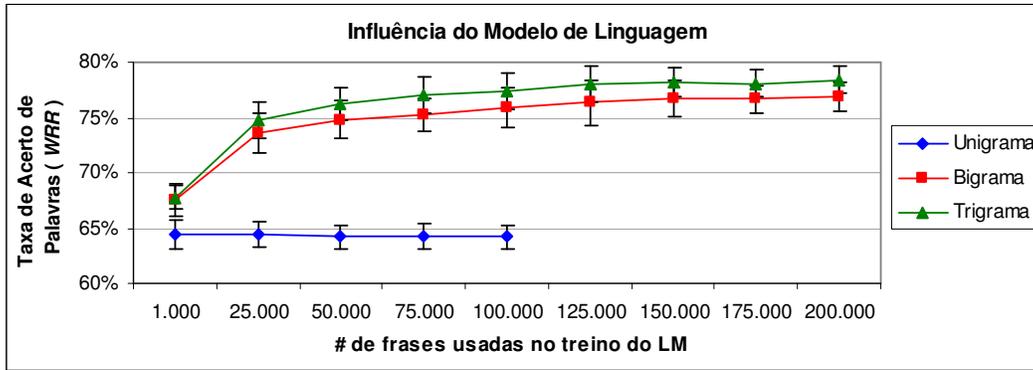


Figura 33: Evolução da  $\overline{WRR}$  em função do número de frases usadas no treino do modelo de linguagem.

Na tabela abaixo, exibem-se os tempos de processamento médio por sentença,  $\overline{T_p}$ , calculados da mesma forma que o  $\overline{WRR}$ .

Tabela 9: Medição do tempo de processamento médio em função da variação do modelo de linguagem.

# de Frases	Unigrama			Bigrama			Trigrama		
	$\overline{T_p}$	$\sigma$	Variação	$\overline{T_p}$	$\sigma$	variação	$\overline{T_p}$	$\sigma$	variação
1.000	32	2	--	33	2	--	34	1	--
25.000	30	1	-6,35%	35	3	6,82%	39	3	15,67%
50.000	32	1	6,78%	36	2	2,13%	40	2	3,23%
75.000	30	2	-5,56%	38	3	4,17%	42	2	4,38%
100.000	30	1	0,84%	37	4	-1,33%	43	3	2,40%
125.000				38	3	1,35%	43	2	1,17%
150.000				38	2	1,33%	44	3	1,73%
175.000				39	3	1,32%	46	3	3,98%
200.000				41	2	7,14%	47	3	2,19%

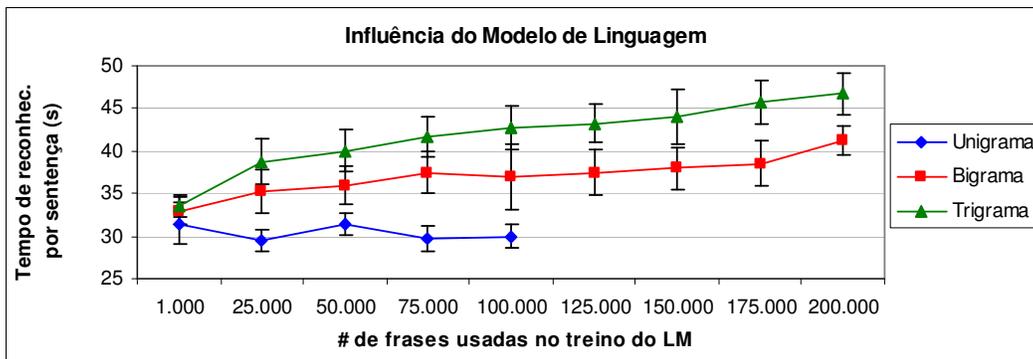


Figura 34: Evolução do tempo de processamento médio por sentença em função do número de frases usadas no treino do modelo de linguagem.

Nos experimentos acima, a  $\overline{WRR}$  parou de evoluir a partir de 150.000 frases e em contrapartida, o tempo de processamento aumentou sensivelmente. A razão desse fato pode estar relacionada a uma saturação do modelo lingüístico, que depois de um certo limite vai apenas aumentando o modelo, com probabilidades de seqüências de palavras pouco freqüentes na base de teste, e, portanto, insignificantes no resultado total. Por esse motivo, nos experimentos seguintes onde o modelo de linguagem é fixo, apenas o treinamento com 150.000 frases será considerado.

### 3) Teste do sistema CSR variando o número de gaussianas do modelo acústico

Neste teste, considera-se a mesma configuração do teste anterior, com o modelo de linguagem treinado com 150.000 frases. No entanto, o número de gaussianas utilizadas na distribuição de probabilidades de saída dos estados é variado, desde uma única gaussiana, como no exemplo anterior, até 20 gaussianas. Os modelos de linguagem considerados são bigrama e trigrama. Inicialmente o teste é realizado apenas com trifones intrapalavra (3A) e num segundo experimento (3B), também são considerados os trifones entre-palavras durante o treinamento e teste do sistema.

#### 3A) Apenas com trifones intrapalavra

Tabela 10: Medição da  $\overline{WRR}$  em função da variação do número de gaussianas.

Modelos Gaussianas	Bigrama			Trigrama			$\overline{WRR}$ do trigrama em relação ao bigrama
	$\overline{WRR}$	$\sigma$	variação	$\overline{WRR}$	$\sigma$	variação	
1	76,54%	1,47%	--	78,00%	1,37%	--	+1,90%
4	83,66%	0,68%	9,31%	84,81%	0,90%	8,73%	+1,37%
8	86,10%	0,55%	2,91%	87,36%	0,65%	3,01%	+1,47%
12	87,17%	0,45%	1,24%	88,42%	0,46%	1,21%	+1,44%
16	87,42%	0,46%	0,29%	88,60%	0,60%	0,21%	+1,35%
20	87,37%	0,69%	-0,05%	88,52%	0,98%	-0,10%	+1,31%

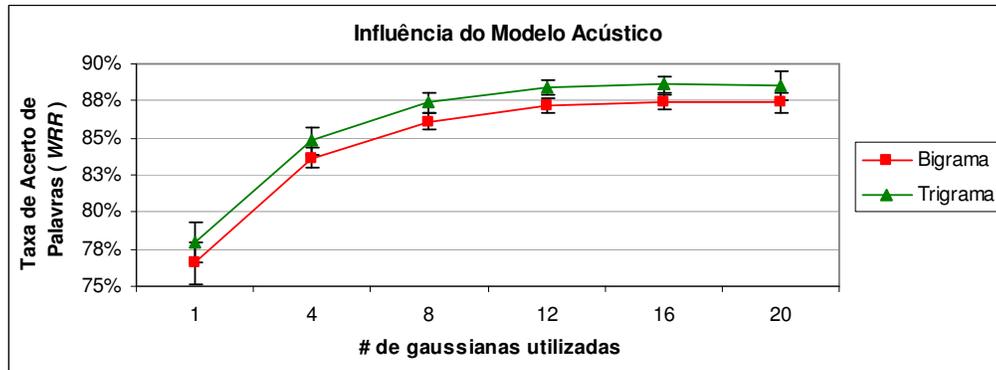


Figura 35: Evolução da  $\overline{WRR}$  em função do número gaussianas usadas no treino do modelo acústico.

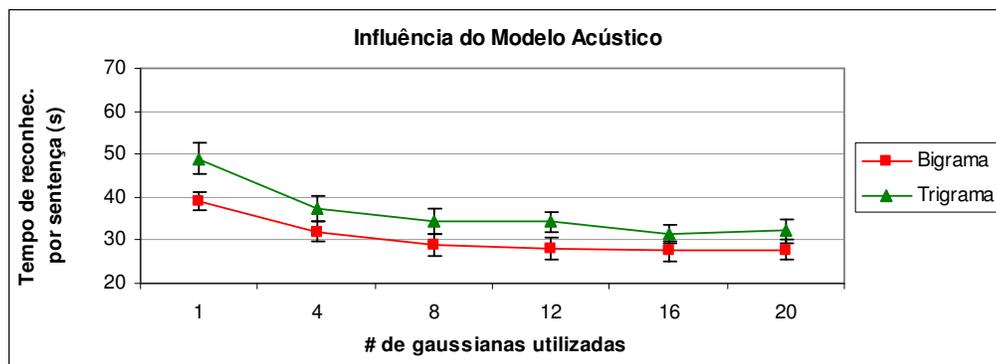


Figura 36: Evolução do  $\overline{Tp}$  em função do número gaussianas usadas no treino do modelo acústico.

Pela Tabela 10, percebe-se que os trigramas apresentam desempenho maior em relação aos bigramas, sempre acima de 1%. Essa relação diminui a medida que o número de gaussianas utilizadas no modelo aumenta, passando de 1,9% para 1,3%.

Um modelo acústico melhor treinado parece aliviar um pouco a influencia do modelo de linguagem durante a busca.

### 3B) Considerando também trifones entre-palavras

Tabela 11: Medição da  $\overline{WRR}$  em função da variação do número de gaussianas.

Modelos Gaussianas	Bigrama			Trigrama			$\overline{WRR}$ do trigrama em relação ao bigrama
	$\overline{WRR}$	$\sigma$	variação	$\overline{WRR}$	$\sigma$	variação	
1	78,75%	0,90%	--	79,78%	1,55%	--	+1,31%
4	85,19%	1,31%	8,17%	86,28%	1,18%	8,15%	+1,29%
8	87,37%	0,59%	2,56%	88,33%	0,68%	2,37%	+1,10%
12	88,11%	0,78%	0,85%	88,89%	0,78%	0,63%	+0,89%
16	88,25%	0,76%	0,16%	89,22%	0,84%	0,37%	+1,10%
20	88,42%	0,89%	0,19%	89,20%	0,81%	-0,02%	+0,89%

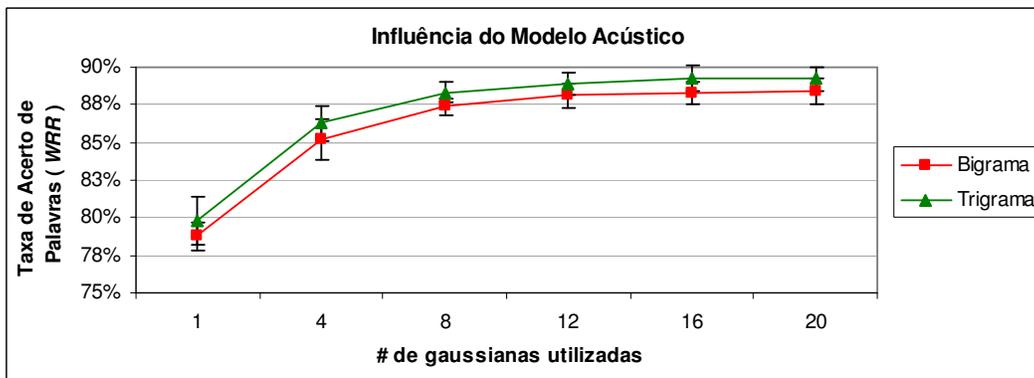


Figura 37: Evolução da  $\overline{WRR}$  em função do número gaussianas usadas no treino do modelo acústico.

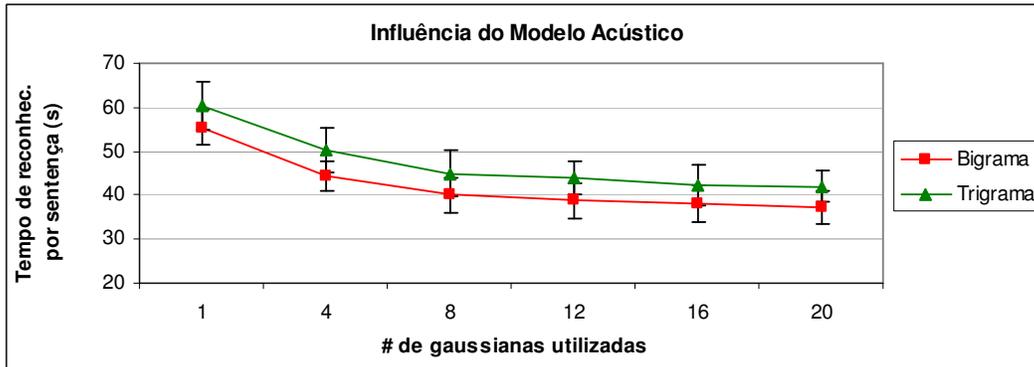


Figura 38: Evolução do  $\overline{Tp}$  em função do número gaussianas usadas no treino do modelo acústico.

Nos experimentos anteriores, o tempo de processamento caiu a medida que mais gaussianas foram adicionadas ao modelo. Com mais gaussianas o modelo acústico tem suas distribuições melhor caracterizadas. Conseqüentemente, a diminuição no tempo de processamento pode estar ligada a uma poda mais efetiva do decodificador no nível acústico e respectiva redução do espaço de busca. Como o custos computacionais envolvidos no processo de decodificação são muito superiores a qualquer outro custo computacional do reconhecedor de fala, a complexidade adicionada pelo aumento do número de gaussianas é anulada por uma possível melhora no desempenho do decodificador.

Como nestes experimentos a  $\overline{WRR}$  permaneceu constante acima de 16 gaussianas, os próximos experimentos irão considerar este número como padrão. Ao mudar a mistura de gaussianas de apenas 1 para 16, a  $\overline{WRR}$  aumentou em praticamente 10%. A utilização de trifones entre-palavras melhorou a  $\overline{WRR}$  em apenas 1%, porém o tempo de processamento por sentença aumentou em aproximadamente 10 segundos (+36%). O melhor resultado obtido com estas configurações foi de 90% de acerto no 3º grupo de teste da base de fala.

#### 4) Avaliação da perplexidade do LM em função do crescimento do dicionário

O método utilizado para aumentar o dicionário do sistema foi selecionar gradualmente as frases da base de texto e acrescentar as novas palavras que iam surgindo ao dicionário do sistema (inicialmente com 3528 palavras). Dessa forma, pode-se levantar a curva apresentada a seguir, onde o número de frases utilizadas varia de 0 a 350.000. O tamanho de dicionário alcançado neste último caso foi de 67.505 palavras.

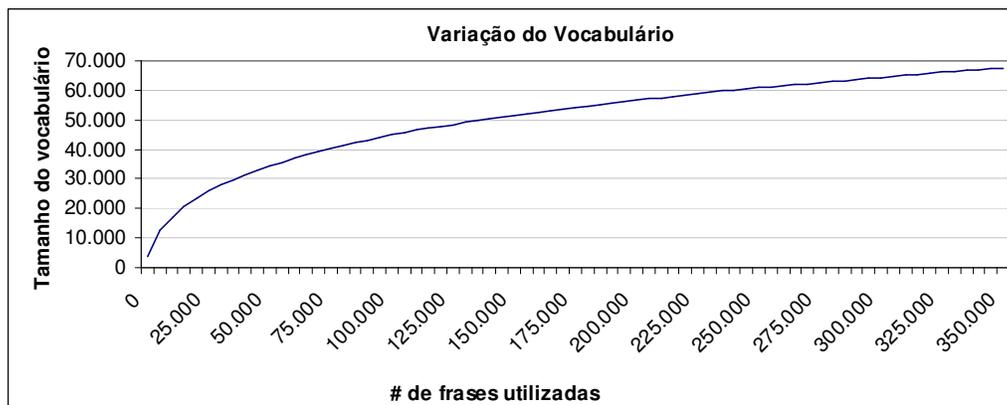


Figura 39: Crescimento do dicionário em função do número de frases utilizadas.

Se estabelecermos uma meta de 60k palavras, verifica-se pela curva acima que esse tamanho de dicionário é alcançado utilizando 240.000 frases. Fixando esse número de frases para o treino do modelo de linguagem, novamente a perplexidade do modelo de linguagem, para o caso de trigramas, é medida, porém para diferentes tamanhos de dicionário, conforme segue:

Tabela 12: Medição da perplexidade em função do aumento do dicionário

Modelo de Linguagem: 240.000 frases / trigramas			
Número de Frases	Tamanho do Dicionário	Perplexidade	Varição
0	3.5k	257	--
5.000	12.5k	294	14%
15.000	20.5k	301	2%
40.000	31k	304	1%
75.000	40k	306	1%
140.000	50k	307	0%
240.000	60k	307	0%

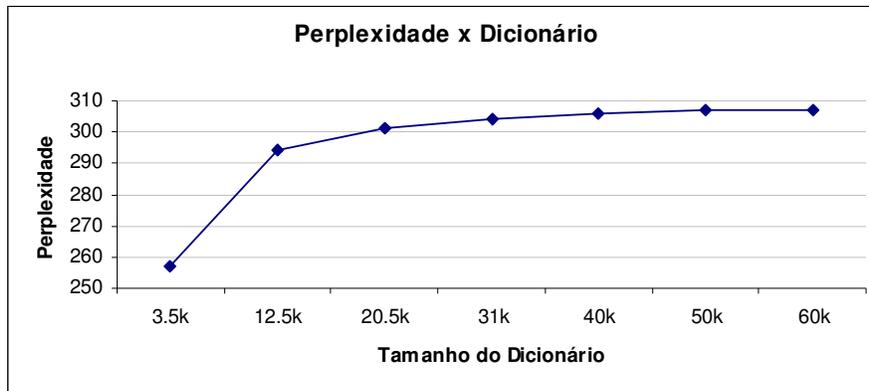


Figura 40: Evolução da perplexidade em função do aumento do dicionário

Uma possível razão para a perplexidade não variar muito da metade da curva em diante vai ao encontro do método como o dicionário foi expandido. No começo da expansão, palavras que possuem grande frequência na base são acrescentadas ao dicionário e competem dentro do modelo, com altas probabilidades. Posteriormente, o dicionário é expandido apenas com palavras pouco frequentes e cujas baixas probabilidades praticamente não pesam nas decisões do modelo.

### 5) Teste do sistema CSR variando o tamanho do vocabulário

A proposta deste experimento foi testar o comportamento do sistema CSR utilizado até então, conforme seu vocabulário é ampliado. Numa configuração inicial, utilizou-se o modelo de linguagem trigrama construído com 240.000 frases, 60k palavras, 16 gaussianas e trifones intrapalavra e entre-palavras. O objetivo foi medir o tempo de processamento médio por sentença visto que, com o dicionário original de 3528 palavras, o sistema já apresentava tempos na ordem de um minuto por sentença. O tempo medido nesta tarefa subiu para aproximadamente 8 minutos por sentença, de forma que a massa de testes que se desejava fazer nos moldes dos experimentos anteriores foi inviabilizada, além de ter sua aplicabilidade bastante reduzida.

Como opção, os parâmetros do sistema CSR que indicam o tamanho do feixe de busca foram reduzidos empiricamente em 20% de forma a se acelerar o tempo de processamento, mesmo sabendo que a taxa de acerto deveria cair. A queda da  $\overline{WRR}$  observada foi de aproximadamente 8%, passando para valores em torno de 81%, no entanto, o tempo de processamento medido para a tarefa de 60k palavras foi de pouco mais de 2 minutos por sentença e, para o dicionário original (3528 palavras), de aproximadamente 15 segundos. Dessa forma, pôde-se observar o efeito do feixe de busca no desempenho do decodificador. Percebeu-se também que o tempo de inicialização do decodificador cresce bastante quando o vocabulário é ampliado, indo de um tempo quase imperceptível com o vocabulário original, para praticamente 8 minutos com o vocabulário de 60k palavras.

Com a nova configuração do feixe de busca aplicada ao sistema, obteve-se a curva de dados a seguir:

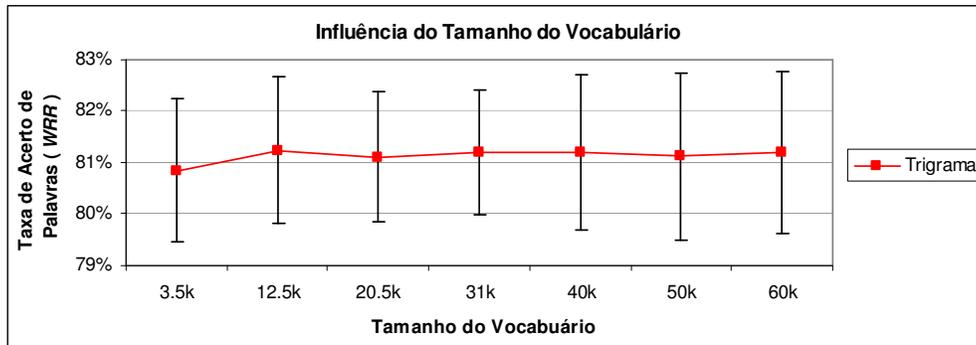


Figura 41: Evolução da  $\overline{WRR}$  em função do tamanho do vocabulário e com os parâmetros do feixe de busca reduzidos em 20%.

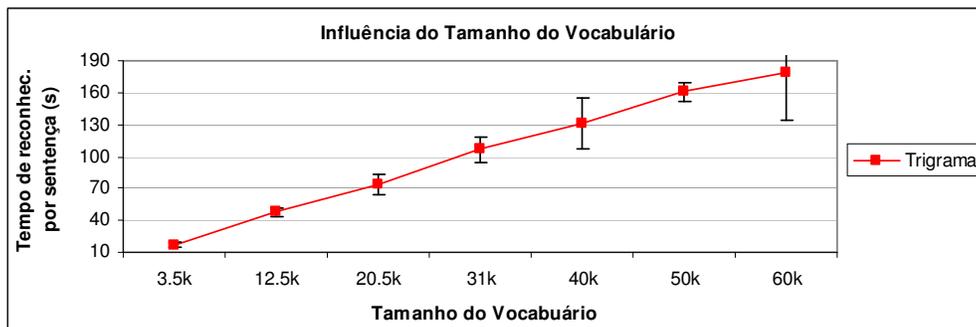


Figura 42: Evolução do  $\overline{Tp}$  em função do tamanho do vocabulário e com os parâmetros do feixe de busca reduzidos em 20%.

A  $\overline{WRR}$  medida permaneceu praticamente constante durante todo o experimento, enquanto o tempo de processamento continuou crescendo muito, indo de 17 segundos por sentença para cerca de 3 minutos. Uma possível razão para esse comportamento está ligada ao custo computacional adicionado à busca de forma a permitir a decodificação usando trigramas, conforme descrito na Seção 3.4.3. Para verificar este fato, o teste anterior foi repetido, porém utilizando bigramas e apenas trifones intrapalavra, dado que, conforme visto anteriormente, a consideração de trifones entre-palavras praticamente não teve efeito sobre a  $\overline{WRR}$  e representou um grande custo computacional ao reconhecimento (em média, +30%):

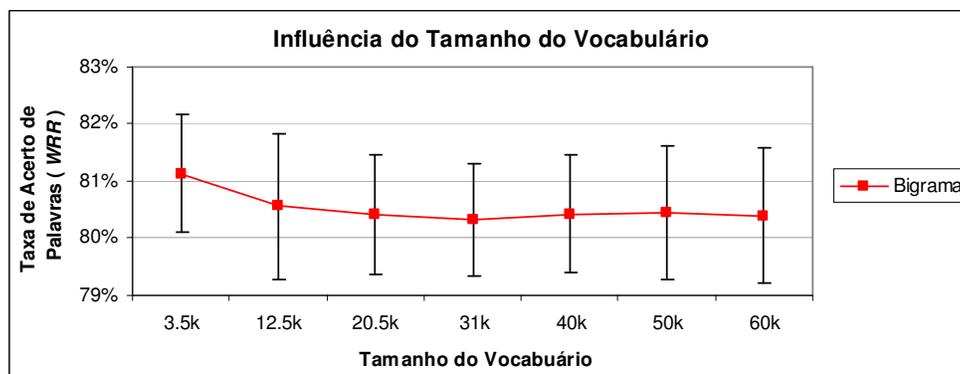


Figura 43: Evolução da  $\overline{WRR}$  em função do tamanho do vocabulário para o reconhecimento baseado em bigramas e desconsiderando trifones entre-palavras.

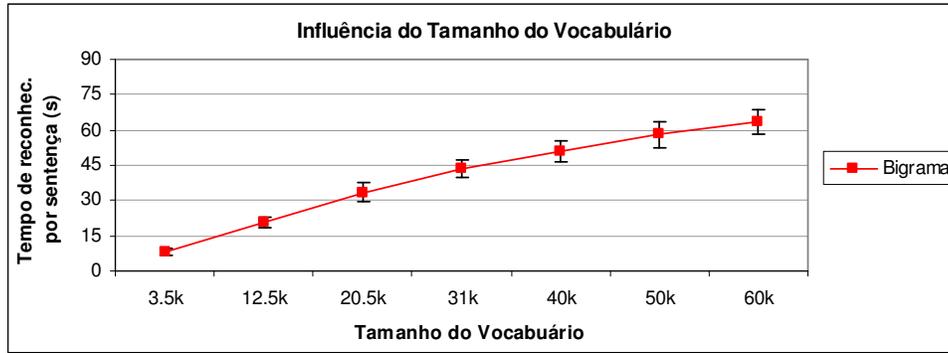


Figura 44: Evolução do  $\overline{T_p}$  em função do tamanho do vocabulário para o reconhecimento baseado em bigramas e desconsiderando trifones entre-palavras.

Tabela 13: Medição do  $\overline{T_p}$  em função da aumento do tamanho do dicionário e comparação com os resultados obtidos no reconhecimento baseado em trigramas e trifones entre-palavras.

Dicionário	Modelo	Bigrama			$\overline{T_p}$ do bigrama em relação ao trigrama
		$\overline{T_p}$	$\sigma$	variação	
3.5k		8	1	--	-52,24%
31k		44	4	443,75%	-59,25%
60k		64	5	46,55%	-64,48%

Conforme previsto, o  $\overline{T_p}$  por sentença caiu efetivamente (mais da metade), enquanto a  $\overline{WRR}$  manteve-se aproximadamente a mesma em relação aos testes que utilizaram modelo de linguagem trigrama e consideraram também trifones entre-palavras (foi percebida uma leve queda, inferior a 1%). Na tarefa de 60k palavras, a redução do  $\overline{T_p}$  chegou a praticamente 65%. Isto talvez explique o fato de muitos sistemas LVCSR utilizarem algoritmos de múltiplas passadas, sendo que na primeira passada, consideram-se normalmente apenas bigramas e trifones intrapalavra. Exemplos de sentenças reconhecidas pelo sistema e respectivas taxas de acerto calculadas podem ser observados na Tabela 14.

Tabela 14: Exemplos de resultados processados pelo HTK após um reconhecimento de palavras utilizando bigramas e o dicionário de 60.000 palavras. Entre colchetes, aparecem os erros encontrados.

M012602:	<b>60.00%</b> [H=6, D=2, S=2, I=0, N=10]
LAB:	<b>DÁ VONTADE DE EU METER O MICROFONE NA SUA CABEÇA</b>
REC:	DAVAM TARDE DE METER MICROFONE NA SUA CABEÇA
M012605:	<b>88.89%</b> [H=8, D=0, S=1, I=0, N=9]
LAB:	<b>UMA PÁGINA INTEIRA SERÁ DEDICADA A NOTAS E CURIOSIDADES</b>
REC:	UMA PAZ INTEIRA SERÁ DEDICADA A NOTAS E CURIOSIDADES
M012610:	<b>75.00%</b> [H=6, D=1, S=1, I=0, N=8]
LAB:	<b>O PRESIDENTE MANDOU CHAMÁ-LO SEGUNDO A IMPRENSA INTERNACIONAL</b>
REC:	O PRESIDENTE MANDOU CHAMÁ-LO SEGUNDA IMPRENSA INTERNACIONAL
M012619:	<b>77.78%</b> [H=7, D=1, S=1, I=0, N=9]
LAB:	<b>SEM A CIRURGIA HUMORISTA PODERIA MORRER EM SEIS MESES</b>
REC:	SEM A CIRURGIA HUMORISTA PODERIA MORREREM SEIS MESES
M012704:	<b>90.91%</b> [H=10, D=0, S=1, I=0, N=11]
LAB:	<b>A ÚNICA CHANCE DO BAHIA ACONTECEU AOS QUARENTA E TRÊS MINUTOS</b>
REC:	A ÚNICA CHANCE DO BAHIA ACONTECEU OS QUARENTA E TRÊS MINUTOS

Resultados publicados previamente em tarefas do mesmo porte para o português brasileiro (60k palavras), com base de dados apresentando independência de locutor, porém com menor variabilidade fonética, apresentam  $WRR$  na ordem de 63% [111]. Uma comparação entre este número e os números apresentados anteriormente (com

$WRR$  em torno de 81%) é difícil de ser realizada, dada a total incompatibilidade das bases [106] e ficando, assim, apenas como uma breve referência.

## 6) Testes do Conversor Fonema-Grafema (PS2G)

O conversor Fonema-Grafema, descrito no Capítulo 5, foi implementado em linguagem C++ e testado em ambiente Microsoft® Windows®. Os testes foram divididos em duas partes: testes com a implementação utilizando DTW (6A) e testes com a implementação baseada em *One-Stage* (6B). Ambos os testes recebem seqüências fonéticas retornadas pelo reconhecedor de fonemas descrito na Seção 4.4 e treinado com misturas de 16 gaussianas. A taxa de acerto média dessas seqüências fonéticas é de 78%, medida conforme a  $\overline{WRR}$ . Testes preliminares dos algoritmos propostos podem ser vistos em [103].

### 6A) Conversor PS2G Baseado em DTW

Nos testes do conversor baseado em DTW, as seqüências fonéticas retornadas pelo reconhecedor de fonemas foram testadas contra suas transcrições ideais, num reconhecimento, tipicamente, de frases fechadas. Em cada conjunto de teste, as 250 seqüências fonéticas foram comparadas com cada uma das 250 transcrições ideais correspondentes. O conversor acertou a frase associada à seqüência fonética em 100% dos casos, levando em média, 1,5 segundo por sentença.

Este bom resultado pode ser explicado pelo fato de existir uma grande variabilidade fonética entre uma frase e outra, facilitando, dessa forma, o trabalho de comparação do DTW, ainda que as seqüências fonéticas possuam em média 22% de erros. Os pequenos tempos medidos no reconhecimento estão ligados à baixa complexidade computacional da tarefa (nesse caso, similar a de um reconhecimento de palavras isoladas).

### 6A) Conversor PS2G Baseado em *One-Stage*

Nos testes do conversor baseado em *One-Stage* a tarefa se equivale a um reconhecimento de palavras conectadas (CWR – *Connected Word Recognition*). Pelo fato do conversor não apresentar um desempenho que viabilize testes com o dicionário de 3528 palavras, reduziu-se o dicionário de forma a conter apenas palavras contidas em subconjuntos das 250 frases de cada conjunto de teste. Para cada conjunto de teste, mediu-se a perplexidade de modelos bigrama com os respectivos dicionários, criados a partir de 25, 50, 75 e 100 frases. Os tamanhos médios dos dicionários obtidos de cada um destes subconjuntos foram de 183, 338, 477 e 601 palavras, respectivamente. As perplexidades médias associadas a esses dicionários, são exibidas na figura a seguir:

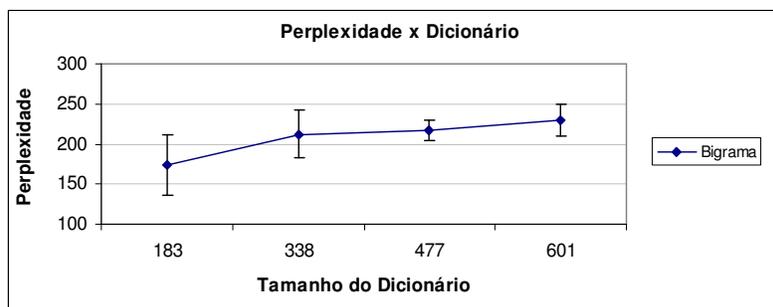


Figura 45: Evolução da perplexidade de modelos bigrama em função do tamanho dos dicionários.

Conforme esperado, a perplexidade aumenta em função do crescimento do dicionário, de forma semelhante ao que já foi observado em experimentos anteriores. Os bigramas treinados nesse experimento são utilizados pelo conversor conforme definido na Equação (42). A seguir, são apresentados os resultados de  $\overline{WRR}$  para cada uma destas tarefas, tendo como entrada as seqüências fonéticas retornadas pelo reconhecedor de fonemas em função de cada um dos subconjuntos de teste. De forma comparativa, os mesmos arquivos de fala processados pelo reconhecedor de fonemas também foram testados no reconhecedor de palavras baseado no HTK. Os resultados de ambos experimentos são exibidos na Figura 46 e Figura 47.

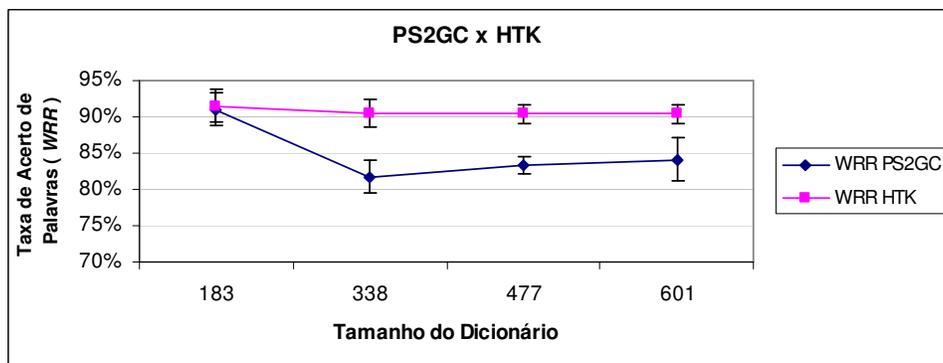


Figura 46: Evolução da  $\overline{WRR}$  em função do tamanho do vocabulário, para os sistemas baseados no conversor PS2G e no HTK.

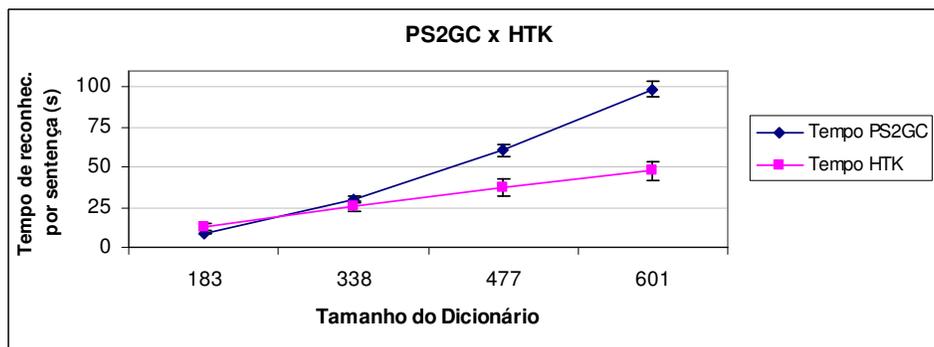


Figura 47: Evolução do  $\overline{Tp}$  em função do tamanho do vocabulário, para os sistemas baseados no conversor PS2G e no HTK.

Exceto na tarefa de 183 palavras, o desempenho do sistema CSR baseado no conversor PS2G mostrou-se inferior àquele obtido com o reconhecedor baseado no HTK. De fato, a técnica *One-Stage* foi utilizada no passado apenas para tarefas com pequenos dicionários como, por exemplo, reconhecimento de dígitos conectados [76]. Conforme descrito em [103], a utilização de bigramas no algoritmo *One-Stage*, definida pela Equação (42), representou uma melhora de 68% no reconhecimento de palavras.

## 8. Conclusões e Trabalhos Futuros

Neste trabalho foram estudados alguns aspectos da teoria referente ao reconhecimento de fala, com ênfase especial ao problema de reconhecimento de fala contínua com amplo vocabulário. Realizou-se uma revisão do estado da arte, sobretudo nas técnicas de decodificação por representarem o principal componente de um sistema LVCSR. Técnicas que viabilizam a implementação destes sistemas sem a necessidade de grandes bases de treino foram apresentadas, dentre elas, a técnica de compartilhamento de estados através de árvores de decisão.

Por ainda existir uma grande deficiência de bases de dados para sistemas de reconhecimento de voz no nosso idioma, optou-se nesse trabalho por utilizar uma nova base que preenchesse os requerimentos para reconhecimento de fala contínua. A base de fala composta por 1000 frases foi construída de forma a se obter um balanceamento fonético para o português brasileiro. A base de textos desenvolvida para treinar o modelo de linguagem, possui 350.000 frases processadas e com a ortografia verificada. A transcrição fonética dos textos foi realizada automaticamente por uma ferramenta desenvolvida para este fim [110]. Apesar de pequena, quando comparada a bases de fala compartilhadas para o idioma inglês, a base formada pelas 1000 frases apresentou bons resultados no sistema LVCSR dependente de locutor (chegando a 90%), e sua consistência foi aferida pelos baixos desvios medidos entre os experimentos com segmentos diferentes da base (sempre menor que 2%).

A implementação do sistema LVCSR apresentada nesta dissertação é baseada no decodificador Vitebi *Beam Search* com o léxico organizado em árvore, utilizando HMMs contínuos e modelos de linguagem do tipo  $n$ -grama (bigramas e trigramas). Os testes realizados a partir das bases de dados desenvolvidas foram planejados de forma a se entender melhor a relação de algumas variáveis importantes no projeto do modelo acústico e do modelo de linguagem e os efeitos que estas têm sobre o desempenho do decodificador tanto em função da sua taxa de acerto de palavras como do tempo de processamento.

Nos testes iniciais, os modelos baseados em unigramas, bigramas e trigramas foram comparados, de forma que uma enorme influência do modelo de linguagem sobre o desempenho do reconhecedor pôde ser verificada. A menor perplexidade medida para um modelo de linguagem ( $PP=236$ ) foi para o caso dos trigramas treinados com as 350.000 e com o dicionário contendo apenas as palavras encontradas nas 1000 frases da base de fala (3528 palavras). Esse número possivelmente poderia diminuir mais ainda com o crescimento da base de texto, porém a uma taxa muito lenta, como se pôde verificar pela tendência da curva apresentada. Para esse tamanho de vocabulário, não se verificou experimentalmente uma melhora na taxa de acerto do sistema para perplexidades menores que 283 (referente a modelos de linguagem construídos com apenas 150.000 frases). Ao contrário, verificou-se um aumento no tempo de processamento.

Quanto ao modelo acústico, pôde-se observar uma melhora na taxa de acerto ao se considerar os trifones entre-palavras, além dos intrapalavra, sempre com resultados superiores em aproximadamente 1%. No entanto, os tempos de decodificação se mostraram, em média, 30% maiores. Já o aumento do número de gaussianas na mistura de probabilidades de emissão de saída dos HMMs contínuos, além de melhorar a taxa de acerto do sistema, contribuiu para diminuir o tempo gasto pelo decodificador. Pelos resultados obtidos, 16 gaussianas se mostrou um número ótimo para compor a mistura.

Durante o aumento do tamanho do vocabulário de teste, pôde-se verificar o efeito do tamanho do feixe de busca no desempenho do sistema. Verificou-se também que o tamanho do dicionário pouco influi na *WRR*, porém está fortemente atrelado à complexidade computacional. O modelo de linguagem trígama, utilizado na decodificação de apenas uma passada, mostrou-se inviável para vocabulários com mais de 20k palavras, apresentando tempos de reconhecimento superiores a 3 minutos para um dicionário de 60k palavras (tempo gasto em frases de 9 a 12 palavras). A utilização de bigramas, nestes casos, reduziu a complexidade computacional em praticamente 60%, mantendo praticamente a mesma *WRR* (80%).

Quanto ao conversor fonema-grafema proposto nesta dissertação, foram verificados bons resultados na implementação baseada em DTW para tarefas de reconhecimento de frases fechadas. O tempo de reconhecimento por sentença foi de apenas 1,5 segundos e todos os testes foram bem sucedidos (100% de frases reconhecidas) quando as seqüências fonéticas recebidas como entrada apresentavam, em média, 22% de erros. A implementação baseada em *One-Stage* mostrou aplicabilidade apenas em tarefas de reconhecimento de fala conectada com pequenos dicionários (menores que 200 palavras). A taxa de acerto obtida para um dicionário de 183 palavras ( $PP=175$ ) foi de 91%, levando, em média, 8 segundos por sentença. A utilização de bigramas adaptada ao algoritmo *One-Stage* representou uma melhora de 68% no reconhecimento de palavras, em relação a testes que não utilizaram um modelo de linguagem no reconhecimento [103].

Em síntese, os resultados desta dissertação são:

- Foi implementado um sistema CSR com amplo vocabulário para o português brasileiro baseado no HTK e ATK.
- Foi realizado um tratamento nos arquivos de áudio e de textos da base de 1.000 frases utilizada.
- Foi desenvolvida uma base de textos para o treinamento de modelos de linguagem do tipo  $n$ -grama.
- Foi compilado um dicionário fonético utilizando um conversor grafema-fonema.
- Foi proposto um algoritmo para conversão de seqüência fonética em texto.
- Foram realizados testes comprovando:
  - viabilidade de treinamento de trifones a partir de bases não segmentadas e utilizando um conversor grafema-fonema.
  - viabilidade da síntese de trifones a partir de árvores de decisão.
  - diminuição da perplexidade em função do modelo  $n$ -grama utilizado (trigramas implicam em perplexidades menores que bigramas).
  - diminuição da perplexidade com o aumento do número de frases utilizadas para treinar o modelo de linguagem.
  - aumento da perplexidade com o aumento do tamanho do dicionário de teste do sistema.
  - que a *WRR* para modelos de trígama é pouco superior a modelos de bigramas que, por sua vez, é consideravelmente superior ao obtido com unigramas.
  - que a complexidade computacional para modelos de trígama é superior a modelos de bigramas que, por sua vez, é superior ao obtido com unigramas.
  - aumento da *WRR* em função do aumento do número de gaussianas e correspondente diminuição da complexidade computacional.
  - aumento da *WRR* ao considerar-se trifones entre-palavras e correspondente aumento da complexidade computacional.

- influência do feixe de busca na complexidade computacional.
- a *WRR* é pouco influenciada pela expansão do vocabulário de teste, por outro lado, a complexidade computacional é altamente influenciada.
- que a utilização de trígama é inviável em tarefas de amplo vocabulário implementado com decodificadores de apenas uma passada.
- é viável o reconhecimento de frases fechadas com um reconhecedor de fonemas e conversor PS2G baseado em DTW.
- é viável o reconhecimento de palavras conectadas, com pequenos dicionários, usando um reconhecedor de fonemas e conversor PS2G baseado em *One-Stage*.

Como tema de trabalhos futuros, são propostos os seguintes tópicos:

- Testes com independência de locutor (já em andamento, ver Apêndice 4)
- Aumento da base de treino
- Testes com bases ruidosas
- Inicialização do treino com bases segmentadas
- Utilização de  $n$ -gramas de classes de palavras
- Adição de variantes de pronúncia para palavras com altas taxas de erro
- Utilização de outros *toolkits* (e de outras técnicas de decodificação)
- Utilização de técnicas *multipass*
- Adaptação de locutor

## Referências

- [1] DESHMUKH, N., DUNCAN, R., GANAPATHIRAJU, A., and PICONE, J., "Benchmarking human performance for continuous speech recognition", *Proc. Fourth Int. Conf. Spoken Lang. Process.*, Philadelphia, PA, USA, Oct 1996, ISCA, Session SuP1P1.10.
- [2] COLE, R. A., MARIANI, J., USZKOREIT, H., ZAENEN, A. and ZUE, V., "Survey of the State of the Art in Human Language Technology", *Cambridge University Press*, Cambridge, UK, 1997 (<http://cslu.cse.ogi.edu/HLTsurvey>).
- [3] MADISSETTI, V.K., WILLIAMS, D.B., "Digital Signal Processing Handbook", *CRCnetBASE*, chapter 47, 1999.
- [4] PALLETT, D.S., et al., "1997 Broadcast News Benchmark Test Results: English and Non-English," *Proc. of the Broadcast News Transcription and Understanding Workshop*, 1998, Lansdowne, Virginia, Morgan Kaufmann Publishers.
- [5] TERUSZKIN, R., "Sistema Orientado a Objeto para Reconhecimento de Voz e Aplicação em Controle de Robô", Projeto Final de Curso do Departamento de Eletrônica e Computação, *Universidade Federal do Rio de Janeiro*, Novembro 2002.
- [6] DOSVOX – Computação para deficientes visuais, <http://intervox.nce.ufrj.br/dosvox>.
- [7] TERUSZKIN, R., RESENDE JR., F.G.V., "Sistema de Conversação Telefônica para Deficientes Auditivos: Módulo de Reconhecimento de Voz", *IV Congresso Ibero-Americano Sobre Tecnologias de Apoio a Portadores de Deficiência*, vol. II, pp. CO-237-241, Vitória - ES, Brasil, Fevereiro, 2006.
- [8] BAKER JK, "The Dragon System – an overview", *IEEE Trans. ASSP*, Vol. 23, No 1, pp24-29, 1975.
- [9] JELINEK, F., "Continuous Speech Recognition by Statistical Methods", *Proc. IEEE*, 64(4), pp532-556, 1976.
- [10] DAVIS, S.B, MERMELSTEIN, P., "Comparison of Parametric Representations of Monosyllabic Word Recognition in Continuous Spoken Sentences", *IEEE Trans ASSP*, Vol. 28, No 4, pp357-366, 1980.
- [11] FURUI, S., "Speaker-Independent Isolated Word Recognition using Dynamic Features of Speech Spectrum", *IEEE Trans. ASSP*, Vol. 34, No 1, pp52-59, 1986.
- [12] APPLEBAUM T., HANSON B., "Regression Features for Recognition of Speech in Noise", *Proc. ICASSP*, S14.26, Toronto, 1991.
- [13] YOUNG, S., "The HTK hidden Markov model toolkit: Design and philosophy", *Cambridge University Engineering Department*, UK, Tech. Rep. CUED/FINFENG/TR152, Sept., 1994. (<http://htk.eng.cam.ac.uk>).
- [14] ATK - API for HTK (<http://htk.eng.cam.ac.uk/develop/atk.shtml>)
- [15] BAUM, L.E., "An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes", *Inequalities*, vol. 1, pp 1-8, 1972.
- [16] LIPORACE, L.A., "Maximum-Likelihood Estimation for Multivariate Observations of Markov Sources", *IEEE Trans Information Th*, Vol. IT-28, No 5, pp729-734, 1982.
- [17] JUANG, B.H., "Maximum-Likelihood Estimation for Mixture Multivariate Stochastic Observations of Markov Chains", *AT&T Technical J*, Vol. 64, No 6, pp1235-1249, 1985.

- [18] BAHL, L.R., BROWN, P.F., de SOUZA, P.V., MERCER, R.L., "Speech recognition with continuous parameters hidden Markov models", *Computer Speech and Language*, Vol. 2, No 3/4, pp219-234, 1987.
- [19] WOODLAND, P.C, ODELL, J.J., VALTCHEV, V., YOUNG, S.J., "Large vocabulary continuous speech recognition using HTK", *Proc. ICASSP*, Vol. 19, pp. 125 - 128, April 1994.
- [20] HUANG, X.D., JACK, M.A., "Semi-continuous hidden Markov models for Speech Signals", *Computer Speech and Language*, Vol. 3, No 3, pp239-252, 1989.
- [21] BELLEGARDA, J.R., NAHAMORO, D., "Tied Mixture Continuous Parameter Modeling for Speech Recognition", *IEEE Trans ASSP*, Vol. 38, No 12, pp2033-2045, 1990.
- [22] PAUL, D.B., "The Lincoln Tied Mixture HMM Continuous Speech Recognizer", *Proc. DARPA Speech and Natural Language Workshop*, pp332-336, Hidden Valley, Pennsylvania, June, 1990.
- [23] HUANG, X.D., HON, H.W., HWANG, M.Y., LEE, K.F., "A Comparative Study of Discrete, Semi-Continuous and Continuous Hidden Markov Models", *Computer Speech and Language*, Vol. 7, No 4, pp359-368, 1993.
- [24] YOUNG, S.J., "The General Use of Tying in Phoneme-Based HMM Speech Recognizers", *Proc ICASSP*, Vol. 1, pp569-572, San Francisco, March, 1992.
- [25] HWANG, M.Y., HUANG, X., "Shared Distribution Hidden Markov Models for Speech Recognition", *IEEE Trans Speech and Audio Processing*, Vol. 1, No 4, pp414-420, 1993.
- [26] YOUNG, S.J., WOODLAND, P.C, "State Clustering in HMM-based Continuous Speech Recognition", *Computer Speech and Language*, Vol. 8, No 4, pp369-384, 1994.
- [27] DIGALAKIS, V., MONACO, P., MURVEIT, H., "Genones: Generalized Mixture Tying in Continuous Speech HMM-based Speech Recognizers", *IEEE Trans. Speech Audio Processing*, vol. 4, pp. 281 - 289, July 1996.
- [28] BAHL, L.R., de SOUZA, P.V., GOPALAKRISHNAN, P.S., NAHAMOO, D., PICHENY, M.A., "Context Dependent Modeling of Phones in Continuous Speech Using Decision Trees", *Proc DARPA Speech and Natural Language Processing Workshop*, pp264-270, Pacific Grove, Calif, Feb, 1991.
- [29] KANNAN, A., OSTENDORF, M., ROHLICEK, J.R., "Maximum Likelihood Clustering of Gaussians for Speech Recognition", *IEEE Trans on Speech and Audio Processing*, Vol. 2, No 3, pp453-455, 1994.
- [30] YOUNG, S.J., ODELL, J.J, WOODLAND, P.C, "Tree-Based State Tying for High Accuracy Acoustic Modeling", *Proc. Human Language Technology Workshop*, pp307-312, Plainsboro NJ, Morgan Kaufman Publishers, March, 1994.
- [31] KATZ, S.M., "Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer", *IEEE Trans ASSP*, Vol. 35, No 3, pp400-401, 1987.
- [32] NEY, H., ESSEN, U., KNESER, R., "On Structuring Probabilistic Dependences in Stochastic Language Modeling", *Computer Speech and Language*, Vol. 8, No 1, pp1-38, 1994.
- [33] JELINEK, F., "Up from Trigrams: the Struggle for Improved Language Models", *Proc Eurospeech*, pp1037-1040, Genoa, 1991.
- [34] BAHL, L.R., BROWN, P.F., de SOUZA, P.V., MERCER, R.L., "A Tree-Based Statistical Language Model for Natural Language Speech Recognition", *IEEE Trans ASSP*, Vol. 37, No 7, 1989.

- [35] WAEGNER, N.P., YOUNG, S.J., "A Trellis-based Language Model for Speech Recognition", *Proc ICSLP*, pp245-248, Banff, Canada, Oct, 1992.
- [36] LAU, R., ROSENFELD, R., ROUKOS, S., "Trigger-based Language Models: a Maximum Entropy Approach", *Proc ICASSP'93*, Vol. 2, pp45-48, Minneapolis, 1993.
- [37] BLACK, E., JELINEK, F., LAFFERTY, J., MAGERMAN, D.M., MERCER, R., ROUKOS, S., "Towards History-based Grammars: Using Richer Models for Probabilistic Parsing", *Proc DARPA, Spoken Language Workshop*, Feb, 1992.
- [38] DELIGNE, S., BIMBOT, F., "Language Modeling by Variable Length Sequences: Theoretical Formulation and Evaluation of Multigrams", *Proc ICASSP*, Vol. 1, pp169-172, Detroit, 1995.
- [39] HUANG, X., ACERO, A., HON, H.W., *Spoken Language Processing: A Guide to Theory, Algorithm and System Development*, New Jersey, Prentice Hall, chapter 11, 2001.
- [40] PICONE, J., "ECE 8463: Fundamentals of Speech Recognition", *Department of Electrical and Computer Engineering Mississippi State University*, Material Disponible On-line: [http://www.cavs.msstate.edu/hse/ies/publications/courses/ece\\_8463/lectures/current/](http://www.cavs.msstate.edu/hse/ies/publications/courses/ece_8463/lectures/current/)
- [41] FORNEY JR., G. D., "The Viterbi Algorithm," *Proc. IEEE*, vol. 61, pp. 268-278, March 1973.
- [42] VINTSYUK, T.K., "Speech Discrimination by Dynamic Programming", *Kibernetika (Cybernetics)*, vol. 4, No. 1, pp. 81-88, Jan.-Feb. 1968.
- [43] ITAKURA, F., "Maximum prediction residual principal applied to speech recognition", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. ASSP-23, pp67-72, Feb. 1975.
- [44] LOWERRE, B.T., REDDY, R., "The HARPY speech understanding system", *Trends in Speech Recognition*, pp340-360, W.A. Lea, Ed. Englewood Cliffs, NJ: Prentice-Hall, 1980.
- [45] SAKOE, H., CHIBA, S., "A dynamic programming approach to continuous speech recognition", *Proc. 7<sup>th</sup> Int. Congr on Acoustics*, Budapest, Hungary, Paper 20 C 13, pp65-68, August 1971.
- [46] VELICHKO, V.M., ZAGOREYKO, N.G., "Automatic recognition of 200 words", *Int. Journal Man-Machine Studies*, vol. 2, pp223-234, June 1970.
- [47] VINTSYUK, T.K., "Element wise recognition of continuous speech composed of words from a specified dictionary", *Kibernetika (Cybernetics)*, vol. 7, pp133-143, March-April 1971.
- [48] MYERS, C. S. and RABINER, L. R. A level building dynamic time warping algorithm for connected word recognition. . *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-29:284-297. April, 1981.
- [49] KLATT, D.B., "Overview of the ARPA speech recognition project", *Trends in Speech Recognition*, pp249-271, W.A. Lea, Ed. Englewood Cliffs, NJ: Prentice-Hall, 1980.
- [50] RABINER, L. R., JUANG, B., *Fundamentals on Speech Recognition*, New Jersey, Prentice Hall, 1996.
- [51] NIST Speech Group Website, <http://www.nist.gov/speech> (14/02/2006).
- [52] LEE, K., HON, H., and REDDY, R., "An Overview of the SPHINX Speech Recognition System", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, pp 35-45, Jan., 1990. (<http://www.speech.cs.cmu.edu/sphinx>)

- [53] DESHMUKH, N., GANAPATHIRAJU, A., HAMAKER, J., PICONE, J., and ORDOWSKI, M., "A public domain speech-to-text system", *Proceedings of the 6th European Conference on Speech Communication and Technology*, vol. 5, pp. 2127–2130, Budapest, Hungary, Sept., 1999. (<http://www.cavs.msstate.edu/hse/ies/projects/speech>)
- [54] RUSSELL, S. and NORVIG, P. "Artificial intelligence: a modern approach", *Prentice Hall*, 1995
- [55] VITERBI, A.J., "Error Bounds for Convolution Codes and an Asymptotically Optimum Decoding Algorithm", *IEEE Transactions on Information Theory*, vol. IT-13, Apr. 1967, pp. 260-269.
- [56] WINSTON, P.H., *Artificial Intelligence*, 3rd ed., 1992, Reading, MA, Addison-Wesley.
- [57] JELINEK, F. "A Fast Sequential Decoding Algorithm Using a Stack". *IBM J. Res. Develop.* vol. 13, pp. 675-685. Nov. 1969.
- [58] BAHL, L.R., JELINEK, F. and MERCER, R., "A Maximum Likelihood Approach to Continuous Speech Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-5, No. 2, Mar. 1983, pp. 179-190.
- [59] PAUL, D. B., "An Efficient A\* Stack Decoder Algorithm for Continuous Speech Recognition with a Stochastic Language Model", *Proceedings of DARPA Speech and Natural Language Workshop*, Feb. 1992, pp405-409.
- [60] NILSSON, N.J., "Problem Solving Methods in Artificial Intelligence", *McGraw-Hill*, New York, 1971.
- [61] GOPALAKRISHNAN, P.S., BAHL, L.R., and MERCER, R.L., "A Tree Search Strategy for Large-Vocabulary Continuous Speech Recognition", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 1995, pp 572-575.
- [62] LOWERRE, B.T., "The HARPY Speech Recognition System", *PhD Thesis in Computer Science Department*, 1976, Carnegie Mellon University.
- [63] NEY, H. J. and ORTMANN, S., "Dynamic Programming Search for Continuous Speech Recognition", *IEEE Signal Processing Magazine*, 1999. pp. 64-83.
- [64] SCHWARTZ, R., et al., "Context-Dependent Modeling for Acoustic-Phonetic Recognition of Speech Signals," *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, 1985, Tampa, FLA pp. 1205-1208.
- [65] SCHWARTZ, R., CHOW, Y.L., "The Optimal N-Best Algorithm: An Efficient Procedure for Finding Multiple Sentence Hypotheses", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Apr. 1990.
- [66] BAHL, L.R., BROWN, P.F., DE SOUZA, P.V., MERCER, R.L., "Obtaining Candidate Words by Polling in a Large Vocabulary Speech Recognition System", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1988.
- [67] BAHL, L.R., DE GENNARO, V., GOPALAKRISHNAN, P.S., MERCER, R.L., "A Fast Approximate Acoustic Match for Large Vocabulary Speech Recognition", *IEEE Transactions on Speech and Audio Processing*, Vol. 1, No. 1, Jan. 1993, pp 59-67.
- [68] CHEN, J.K., SOONG, F.K., "An N-best candidates-based discriminative training for speech recognition applications", *IEEE Trans. on Speech and Audio Processing*, vol. 2, No 1, Part II, pp 206-216, Jan. 1994.
- [69] NGUYEN, L., SCHWARTZ, R., ZHAO, Y., ZAVALIAGKOS, G., "Is N-best dead?", *Proc of DARPA Human Language Technology Workshop*, pp 386-388, Mar. 1994.

- [70] SCHWARTZ, R., AUSTIN, S. "A comparison of several approximate algorithms for finding multiple (N-best) sentence hypotheses", *Proc of the IEEE International Conference on Acoustics Speech and Signal Processing*, pp 701-704, 1991.
- [71] SCHWARTZ, R.M., AUSTIN, S., "Efficient high-performance algorithms for N-best search", *Proc of DARPA Speech and Natural Language Processing Workshop*, pp 6-11, Jun. 1990.
- [72] SOONG, F.K., HUANG, E.F., "A tree-trellis-based fast search for finding the N-best sentence hypotheses in continuous speech recognition", *Proc of the IEEE International Conference on Acoustics Speech and Signal Processing*, pp 705-708, 1991.
- [73] BELLMAN, R., "Dynamic Programming", *Princeton, NJ: Princeton University Press*, 1957.
- [74] BRIDLE, J.S., BROWN, M.D., CHAMBERLAIN, R.M., "An algorithm for connected word recognition", *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Paris, pp. 899-902, May, 1982.
- [75] NEY H., "Connected utterance recognition using dynamic programming", *Festschrifte der Akustik – FASE/DAGA'82, Federation of Acoustic Societies of Europe/Deutsche Arbeitsgemeinschaft für Akustik*, Gottingen, Germany, pp. 915-918, Sept., 1982.
- [76] NEY, H. "The Use of a One-Stage Dynamic Programming Algorithm for Connected Word Recognition", *IEEE Transactions on Acoustics, Speech and Signal Processing*, 32(2):263-271. April, 1984.
- [77] LEE, C.H., RABINER, L.R., "A frame-synchronous network search algorithm for continuous word recognition", *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-37, no 11, pp. 1649-1658, Nov., 1989.
- [78] SAKOE, H., "Two-level DP matching – a dynamic programming-based pattern matching algorithm for connected word recognition", *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-27, pp. 588-595, Dec. 1979.
- [79] ALLEVA, F., HUANG, X., Hwang, M.Y., "An improved search algorithm using incremental knowledge for continuous speech recognition", *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Minneapolis, MN, vol. II, pp. 307-310, April 1993.
- [80] AUBERT, X., DAGAST, C., NEY, H., STEINBISS, V., "Large vocabulary continuous speech recognition of *Wall Street Journal* corpus", *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Adelaide, Australia, vol. II, pp. 129-132, April, 1994.
- [81] CARDIN, R., NORMANDIN, Y., DEMORI, R., "High-performance connected digit recognition using codebook exponents", *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, San Francisco, CA, vol. 1, pp. 505-508, March, 1992.
- [82] DIGALAKIS, V., MURVEIT, H., "Genones: optimizing the degree of mixture tying in a large vocabulary hidden Markov model-based speech recognizer", *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Adelaide, Australia, vol. I, pp. 537-540, April, 1994.
- [83] GAUVAIN, J.I., LAMEL, L.F., ADDA, G., ADDA-DECKER, M., "The LIMSI speech dictation system evaluation on the ARPA *Wall Street Journal* task", *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Adelaide, Australia, vol. I, pp. 557-560, April, 1994.
- [84] HWANG, M.,Y., ROSENFELD, R., THAYER, F., MOSUR, R., CHASE, B., WENDE, R., HUANG, X., ALLEVA, F., "Improving speech recognition performance via phone-dependent VQ codebooks and adaptative language models in SPHINX II", *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Adelaide, Australia, vol. I, pp. 549-552, April, 1994.

- [85] KUBALA, H., ANASTASAKOS, A., MAKHOUL, J., NGUYEA, L., SCHWARTZ, R., "Comparative experiments on large vocabulary speech recognition", *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Adelaide, Australia, vol. I, pp. 561-564, April, 1994.
- [86] LACOATURE, R., NORMANDIN, Y., "Efficient lexical access strategies", *Proc. Europ. Conf. on Speech Communication and Technology*, Berlin, Germany, pp. 1537-1540, Sep., 1993.
- [87] MURVEIT, H., BUTZBERGER, J., DIGALAKIS, V., WEINTRAUB, M., "Large-vocabulary dictation using SRI's DECIPHER™ speech recognition system: Progressive search techniques", *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Minneapolis, MN, vol. II, pp. 319-322, April 1993.
- [88] NEY, H., HAEB-UMBACH, R., TRAN, B.H., OERDER, M., "Improvements in beam search for 10,000-word continuous speech recognition", *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, San Francisco, CA, vol. 1, pp. 13-16, March, 1992.
- [89] ORTMANNS, S., NEY, H., "Experimental analysis of the search space for 20,000-word speech recognition", *Proc. Fourth European Conference on Speech Communication and Technology*, Madrid, Spain, pp. 901-904, Sep., 1995.
- [90] STEINBISS, V., NEY, H., HAEB-UMBACH, R., TRAN, B.H., ESSEN, U., KNESER, R. et al, "The Philips research system for large-vocabulary continuous-speech recognition", *Third European Conference on Speech Communication and Technology*, Berlin, pp. 2125-2128, Sep., 1993.
- [91] HAEB-UMBACH, R., NEY, H., "Improvements in time-synchronous beam search for 10,000-word continuous speech recognition", *IEEE Trans. on Speech and Audio Processing*, vol. 2, pp. 353-356, April, 1994.
- [92] KLOVSTAD, J.M., MONDSHEIN, L.F., "The CASPERS linguistics analysis system", *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 23, pp. 118-123, Feb., 1975.
- [93] KLATT, D.H., "SCRIBER and LAFS: two new approaches to speech analysis", *Trends in Speech Recognition*, pp. 529-555, W.A. Lea Ed., Englewood Cliffs, NJ: Prentice-Hall, 1980.
- [94] NEY, H., "Search strategies for large-vocabulary continuous-speech recognition", *Speech Recognition and Coding – New Advances and Trends*, A.J., Rubio Aynse, J.M., Lopez Soler, Eds., NATO Advanced Studies Institute, Bubion, Spain, June-July, 1993, pp. 210-225, Berlin:Springer, 1995.
- [95] ODELL, J.J., VALTCHEV, V., WOODLAND, P.C., YOUNG, S.J., "A one-pass decoder design for large vocabulary recognition", *ARPA Spoken Language Technology Workshop*, Plainsboro, NJ, pp. 405-410, March, 1994.
- [96] ORTMANNS, S., NEY, H., EIDEN, A., "Language –model look ahead for large vocabulary speech recognition", *Proc. Int. Conf. on Spoken Language Processing*, Philadelphia, PA, pp. 2095-2098, Oct., 1996.
- [97] STEINBISS, V., TRAN, B.H., NEY, H., "Improvements in beam search", *Int. Conf. on Spoken Language Processing*, Yokohama, Japan, pp. 1355-1358, Sep. 1994.
- [98] MAIA, R.S., "Speech synthesis and phonetic vocoding for Brazilian Portuguese based on parameter generation from hidden Markov models", PhD Thesis, Nagoya Institute of Technology, Japan, 2006.
- [99] BAHL, L.R., BALAKRISHNAN-AIYER, S., FRANZ, M., GOPALAKRISHNAN, P.S., GOPINATH, R., NOVAK, M., PADMANABHAN, M. AND ROUKOS, S., "The IBM Large Vocabulary Continuous Speech Recognition System for the ARPA NAB News Task", *Proceedings of ARPA Spoken Language System Technology Workshop*, pp. 121-126, Jan. 1995.

- [100] HUANG, X., et al., "Microsoft Windows Highly Intelligent Speech Recognizer: Whisper", *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 93-96, 1995.
- [101] DELLER, J.R., PROAKIS, J.G., HANSEN, J.H., *Discrete-Time Processing of Speech Signals*, New York, MacMillan, 1993.
- [102] TERUSZKIN, R., RESENDE JR., F.G.V., "Sistema Híbrido HMM-DTW para Reconhecimento de Palavras Isoladas através de Trifones", *Semana da Eletrônica – Universidade Federal do Rio de Janeiro*, Rio de Janeiro - RJ, Brasil, Setembro, 2003.
- [103] TERUSZKIN, R., RESENDE JR., F.G.V., "Phonetic Sequence to Graphemes Conversion Based on DTW and One-Stage Algorithms", *VII Encontro para o Processamento Computacional da Língua Portuguesa Escrita e Falada*, Itatiaia - RJ, Brasil, Maio, 2006.
- [104] ZUE, V., et al., "The MIT SUMMIT System: A Progress Report", *Proc. of DARPA Speech and Natural Language Workshop*, 1989 pp. 179-189.
- [105] YNOGUTI, C. A., "Reconhecimento de Fala Contínua Utilizando Modelos Ocultos de Markov", *Faculdade de Engenharia Elétrica – UNICAMP*, Tese de Doutorado, Maio, 1999.
- [106] SPOLTECH Brazilian Portuguese Corpus, February, 2002. Disponível em: <http://www.cslu.ogi.edu/corpora/spoltech/>. Acesso em: 18/05/2006.
- [107] "CORPUS de Extractus de Textos Eletrônicos Nilc/ Folha de São Paulo (Ceten-Folha)". Disponível em: <http://acdc.linguateca.pt/cetenfolha/>. Acesso em: 14/11/2005.
- [108] CIRIGLIANO, R. J. R., et al., "Um Conjunto de 1000 Frases Foneticamente Balanceadas para o Português Brasileiro Obtido Utilizando a Abordagem de Algoritmos Genéticos", *XXII Simpósio Brasileiro de Telecomunicações – SBrT*, 2005.
- [109] "Dicionário br.ispell do português falado no Brasil, versão 3.0". Disponível em: <http://www.ime.usp.br/~ueda/br.ispell/>. Acesso em: 01/03/2006.
- [110] BARBOSA, F. L. F., et al., "Grapheme-phone transcription algorithm for a Brazilian Portuguese TTS", *Proceedings of PROPOR*, Faro, Portugal, 2003.
- [111] SILVA, Ê., BAPTISTA, L., FERNANDES, H., KLAUTAU, A., "Desenvolvimento de um Sistema de Reconhecimento Automático de Voz Contínua com Grande Vocabulário para o Português Brasileiro", *Congresso da Sociedade Brasileira de Computação (SBC) - Workshop TIL*, São Leopoldo, 2005.
- [112] ALCAIM, A., SOLEWICZ, J.A., MORAES, J.A., "Frequência de ocorrência dos fones e listas de frases foneticamente balanceadas no português falado no Rio de Janeiro", *Revista da Sociedade Brasileira de Telecomunicações (SBrT)*, Rio de Janeiro, v. 7, n. 1, p. 23-41, 1992.

## Apêndice 1

### *Estratégias utilizadas em decodificadores de fala contínua:*

**Busca integrada vs. não integrada:** Uma estratégia de busca é denominada como integrada se *todas* as fontes de conhecimento disponíveis, isto é, modelos acústico-fonéticos, restrições do léxico de pronúncias e LM, são exploradas no processo de busca ao mesmo tempo. Este conceito é tipicamente implementado em uma estratégia executada em apenas uma passada (a estratégia *one-pass* será vista em detalhe na Seção 3).

**Busca síncrona vs. não síncrona:** Uma estratégia de busca é dita síncrona no tempo (*time-synchronous*) se as hipóteses da busca são formadas de forma síncrona no tempo em relação à seqüência de vetores acústicos. Tipicamente, o conceito de sincronia no tempo está relacionado com a estratégia *one-pass*. A busca A\* ou *stack decoding* (também vista na Seção 3) é um exemplo de estratégia de busca que não é necessariamente síncrona no tempo.

**Única passada vs. múltiplas passadas:** Uma busca é dita como uma estratégia *one-pass*, se nela é executada apenas uma única passada sobre a seqüência de entrada, de forma oposta ao conceito de múltiplas passadas (*multi-pass*) ou múltiplos níveis (*multi-level*). A estratégia de busca *one-pass* é, virtualmente, sempre baseada em programação dinâmica.

**Busca condicionada por palavras vs. condicionada pelo tempo:** Estes termos se referem à forma como o espaço de busca é estruturado, especialmente no contexto de programação dinâmica. Numa busca condicionada por palavras (*word-conditioned*), cada hipótese de busca é condicionada na palavra predecessora. Isso implica que a otimização sobre o tempo final da palavra predecessora que é desconhecido, ou seja, a fronteira no tempo entre a palavra predecessora e a palavra sobre consideração, já foi levado em conta num estágio anterior da busca. Portanto, esse método é diferente de uma busca condicionada pelo tempo (*time-conditioned*), onde, para cada hipótese da busca, a dependência pelo tempo final da palavra predecessora é explicitamente retida e a otimização sobre as desconhecidas fronteiras entre palavras é executada como um passo final da busca.

**Melhor sentença vs. grafo de palavras:** O atributo “melhor sentença” (*single best*) é usado para denotar um conceito de busca que determina uma única seqüência de palavras mais passível de ser reconhecida. As alternativas são, entre outros, os conceitos intitulados “*n*-melhores” (*n-best*) e métodos baseados em grafos de palavras (*word graph*). A idéia de um grafo de palavras é organizar as melhores hipóteses de sentenças na forma de um grafo onde os ramos representam as palavras sob hipótese. Algumas vezes o termo “treliça de palavras” (*word lattice*) é usado como sinônimo.

**Léxico linear vs. léxico em árvore:** Para uma tarefa com dicionário pequeno é suficiente ter uma representação separada de cada palavra do vocabulário em termos de fonemas. Portanto, essa abordagem é referenciada com um léxico linear (*linear lexicon*). Para um grande vocabulário, no entanto, é, tipicamente, muito útil organizar o léxico de pronúncia como uma árvore, cujos arcos são os fonemas e as folhas representam palavras.

## Apêndice 2

**Principais avaliações realizadas pelo NIST com intuito de medir o desempenho dos sistemas desenvolvidos na área de reconhecimento da fala:**

**Transcrição Rica de Textos:** A avaliação de tarefas conhecidas como transcrição rica de textos (RT - *Rich Transcription*) tem como objetivo promover e medir os avanços em diversas tecnologias de ASR que implementam o estado da arte nessa área. Estas tecnologias têm o intuito de produzir transcrições da fala mais legíveis para humanos e mais usáveis por máquinas. Dessa forma, definiu-se um conjunto de tarefas de pesquisa o qual foi basicamente subdividido em tarefas de transcrições de fala em texto (STT – *Speech-to-Text Transcription*) e tarefas de extração de metadados (MDE – *Metadata Extraction*), como identidade do locutor, idioma falado, tempo de início e fim, entonação, etc. As bases de avaliação incluem transmissões de telejornal, conversações telefônicas e conversas em salas de reunião.

**Detecção e Rastreamento de Tópicos:** A pesquisa em detecção e rastreamento de tópicos (TDT - *Topic Detection and Tracking*) tem como objetivo contextualizar tópicos ou eventos de interesse numa coleção de histórias multimídia que se expande constantemente. As aplicações TDT organizam uma grande quantidade de informação ou agrupam grandes coleções de mídias não textuais. Existem 5 linhas de pesquisas definidas no programa TDT: Segmentação de Histórias, Rastreamento de Tópicos, Detecção de Tópicos, Detecção de Primeira História e Detecção de Ligações.

**Tradução por Máquina:** O objetivo do projeto de tradução por máquina (MT - *Machine Translation*) é desenvolver tecnologias que convertem áudio e texto de uma variedade de idiomas para textos de um idioma específico. Os textos traduzidos devem ser precisos e fluentes.

**Reconhecimento de Idioma:** A avaliação de tarefas de reconhecimento de idioma (LRE – *Language Recognition Evaluation*) tem como objetivo estabelecer uma linha base do desempenho e atual capacidade de se reconhecer idiomas falados em conversas telefônicas e se estabelecer uma base de conhecimento para futuros esforços de pesquisa na área.

**Reconhecimento de Locutor:** Se resume a avaliação de tarefas que têm como objetivo a identificação robusta da identidade de locutores (SR – *Speaker Recognition*) em bases de conversas telefônicas gravadas.

# Apêndice 3

## Léxico criado a partir das 1000 frases

Abaixo está a lista com as 3528 palavras distintas encontradas no conjunto de 1000 frases e suas respectivas transcrições fonéticas utilizadas para treino e teste do sistema de reconhecimento de fala contínua.

Aa  
ABAIXO abajSu  
ABALADO abaladu  
ABANDONAR aba~donaX  
ABANDONOU aba~donow  
ABASTECER abasteseX  
ABASTECIMENTO abastesime~tu  
ABATIMENTO abati~me~tu  
ABERNÉSSIA abeRnEsia  
ABERTO abeXtu  
ABERTURA abeXtura  
ABORDAR aboRdaX  
ABRAÇANDO abra~du  
ABRE abri  
ABRIL abriw  
ABRIR abriX  
ABRIR-SE abriXsi  
ABRIU abriw  
ABSOLUTO absoluto  
ACABA akaba  
ACABO akabu  
ACABOU akabow  
ACASO akazu  
ACEITA asejta  
ACEITARAM asejtara~w~  
ACERTARAM aseXtara~w~  
ACERTOU aseXtow  
ACESSO asesu  
ACHA aSa  
ACHAM aSa~w~  
ACHAVA aSava  
ACHO aSu  
ACIDENTE aside~tSi  
ACIDENTES aside~tSis  
ACIMA asi~ma  
ACOMPANHAMENTO ako~pa~Jame~tu  
ACOMPANHAR ako~pa~JaX  
ACONTECE ako~tesi  
ACONTECER ako~teseX  
ACONTECEU ako~tesew  
ACORDO akoRdu  
ACREDITA akredZita  
ACREDITAR akredZitaX  
ACREDITAVAM akredZitava~w~  
ACRESCIDO akresidu  
ACUSA akuza  
ACUSADO akuzadu  
ADIANTA adzia~ta  
ADMINISTRAÇÃO adZministrasa~w~  
ADMINISTROU adZministrow  
ADOLESCENTES adolese~tSis  
ADORÁVEIS adoravejs  
ADOTADAS adotadas  
ADOTAR adotaX  
ADOÇÃO adosa~w~  
ADQUIRIDAS adZkiridas  
ADULTOS aduwtus  
ADVERSÁRIA adZveXsaria  
ADVERSÁRIOS adZveXsarius  
ADVOGADAS adZvogadas  
ADVOGADO adZvogadu  
AEROPORTOS aeropoXtus  
AEROPORTUÁRIOS aeropoXtuarius  
AFA afa  
AFETIVAS afetSivas  
AFINAL afinaw  
AFIRMA afirMa  
AFIRMAM afirMa~w~  
AFIRMARAM afirMara~w~  
AFIRMOU afirMow  
AGENTES aZe~tSis  
AGIR aZiX  
AGONIA agonia  
AGORA agora  
AGOSTO agostu  
AGREMIÇÕES agre~miaso~j~s  
AGRESSIVAS agresivas  
AGRESSIVIDADE agresividadZi  
AGRESSIVO agresivu  
AGRICULTURA agrikuwtura  
AGRUPA agrupa  
AGRICOLA agrikola  
AGUARDAVAM agwaRdava~w~  
AGÊNCIA aZe~sia  
AGÊNCIAS aZe~sias  
AGÜENTAR agwe~taX  
AINDA ai~da  
AIRES ajris  
AJUDA aZuda  
AJUDAR aZudaX  
AJUSTADO aZustadu  
AJUSTE aZustSi  
ALBERTO awbeXtu  
ALCANÇOU awk~sow  
ALCOOLISMO awkoolizmu  
ALCORÃO awkora~w~  
ALEGRE alegri  
ALEGRES alegris  
ALEJARIAM alejZaria~w~  
ALEMANHA alema~Ja  
ALEMÃES alema~j~s  
ALESSANDRA ales~dra  
ALEXANDRE aleSa~dri  
ALFREDO awfredu  
ALFÂNDEGA awfa~dega  
ALGO awgu  
ALGUM awgu~  
ALGUMA awgu~ma  
ALGUMAS awgu~mas  
ALGUNS awgu~s  
ALGUÉM awge~j~  
ALHEIA aleJa  
ALIAI  
ALIADOS aliadus  
ALIMENTADAS alime~tadas  
ALIÁS aliajs  
ALMA awma  
ALMEIDA awmejda  
ALMOXARIFADO awmoSarifadu  
ALMOÇO awmosu  
ALQUIMISTA awkimista  
ALTA awta  
ALTERAÇÕES awteras~j~s  
ALTERNATIVA awteRnatSiva  
ALTO awtu  
ALTOS awtus  
ALTURA awtura  
ALUGA aluga  
ALUGUEL alugEw  
ALUNOS alu~nus  
ALVO awvu  
ALVOS awvus  
ALEM ale~j~  
AMA a~ma  
AMANDO ama~du  
AMANHÃ ama~Ja~  
AMAR amaX  
AMARELO amarelu  
AMEDRONTANDO amedro~ta~du  
AMERICANA amerika~na  
AMERICANAS amerika~nas  
AMERICANO amerika~nu  
AMES a~mis  
AMIGO amigu  
AMIGOS amigus  
AMISTOSOS amistozus  
AMIZADE amizadZi  
AMOR amoX  
AMOSTRAGEM amostraZe~j~  
AMOTINADOS amotSinadus  
AMPLIOU a~pliw  
AMPLO a~plu  
AMÉRICA a~Erika  
AMÉRICAS a~Erikas  
ANA a~na  
ANALISAM analiza~w~  
ANDAM a~da~w~  
ANDAR a~daX  
ANDRADE a~dradZi  
ANDRELINO a~drelinu  
ANDRÉ a~drE  
ANFAVEA a~favea  
ANGOLA a~gola  
ANIMAIS animajs  
ANIMAR animaX  
ANISTIA anis~tSi  
ANIVERSÁRIO aniveXsariu  
ANO a~nu  
ANOS a~nus  
ANTECIPOU a~tesipow  
ANTENAS a~te~nas  
ANTEONTEM a~te~te~j~  
ANTERIOR a~terioX  
ANTES a~tSis  
ANTITRUSTE a~tSitrustSi  
ANUAIS anuajs  
ANUAL anuaw  
ANÁLISE analizi  
ANÁLISES analizis  
ANÚNCIO anu~siu  
AO aw  
AOS aws  
APARECE aparesi  
APARECEM apares~j~  
APARECER apares~X  
APARECEU apares~ew  
APARECIA aparesia  
APARECIAM aparesia~w~  
APARENTAVA apare~tava  
APAREÇA aparesa  
APARTAMENTO apaXtame~tu  
APARTAMENTOS apaXtame~tus  
APENAS ape~nas  
APESAR apezaX  
APETITE ape~tSi~tSi  
APLAUDIRAM aplawdZira~w~  
APLICA aplika  
APLICADA aplikada  
APLICAR aplikaX  
APLICAÇÃO aplikasa~w~  
APLICAÇÕES aplikaso~j~s  
APOIAR apojaX  
APOIO apoju  
APONTAM apo~ta~w~  
APOSTADOS apostadus  
APREENDEU apre~de~w  
APREENDIDAS apre~dZidas  
APREENSÃO apre~sa~w~  
APRENDE apre~dZi  
APRENDER apre~deX  
APRESENTADO apreze~tadu  
APRESENTANDO apreze~ta~du  
APRESENTAR apreze~taX  
APRESENTARAM apreze~tara~w~  
APRESENTAÇÃO apreze~tasa~w~  
APROFUNDAR profu~daX  
APROVAR aprovaX  
APROVAÇÃO aprovasa~w~  
APROVE aprovi  
APROVEITAR provejtaX  
APTIDÕES aptido~j~s  
APURADA apurada  
APURADOS apuradus  
APURAÇÃO apurasa~w~  
APÓS apOjs  
AQUELES akelis  
AQUI aki  
AQUILO akilu  
AQUISITIVO akizitSivu  
AR aX  
ARAGÃO araga~w~  
ARGENTINA aRZe~tSi~na  
ARGENTINO aRZe~tSi~nu  
ARGENTINOS aRZe~tSi~nus  
ARGUMENTA aRgume~ta  
ARGUMENTAR aRgume~taX  
ARMADILHAS aRma~dZi~Las  
ARMADOS aRmadus  
ARMAS aRmas  
ARMAZENS aRmaze~j~s  
ARMÁRIO aRmariu  
AROMÁTICO aroma~tSi~ku  
ARQUEOLÓGICO aXkeo~lO~Ziku  
ARRASTOU aRastow  
ARRECADADO aRe~kadadu  
ARREDONDOU aRe~do~dow

ARRIBACÁS aRibasa-j-s  
ARRUDA aRuda  
ARSENAL aXsenaw  
ARTE aXtSi  
ARTICULANDO aXtSikula-du  
ARTILHEIRO aXtSiLejru  
ARTISTA aXtSista  
ARTISTAS aXtSistas  
ARTÍSTICA aXtSistSika  
ARÁBIA arabia  
ASas  
ASPECTOS aspektus  
ASSALTANTES asaWta-tSis  
ASSALTAVA asaWtava  
ASSALTO asaWtu  
ASSEGURA asegura  
ASSEGURADO asegurado  
ASSEGURAVA asegurava  
ASSEMBLEIA ase-bleja  
ASSENTAMENTOS ase-tame-tus  
ASSIM asi-  
ASSINATURAS asinaturas  
ASSIS asis  
ASSISTI asistSi  
ASSISTIAM asistSia-w-  
ASSOCIAÇÃO asosiasa-w-  
ASSOMBRAÇÃO aso-brasa-w-  
ASSUMEM asu-me-j-  
ASSUSTADOR asustadoX  
ASTRONAUTAS astronawtas  
ATACADO atakadu  
ATACANTE ataka-tSi  
ATACANTES ataka-tSis  
ATAQUES atakis  
ATENDER ate-deX  
ATENÇÃO ate-sa-w-  
ATINGEM atSi-Ze-j-  
ATITUDE atSitudzi  
ATIVIDADE atSividadzI  
ATIVIDADES atSividadzis  
ATLETAS atletas  
ATLÁNTICA atla-tSika  
ATOR atoX  
ATORES atoris  
ATRAPALHA atrapaLa  
ATRAPALHARAM atrapaLara-w-  
ATRASO atrazu  
ATRAVÉS atravEjs  
ATRAÇÕES atraso-j-s  
ATRAÍDOS atraidus  
ATRIBUIU atribuiw  
ATRIBUIU-SE atribuiwsi  
ATRIZES atrizis  
ATRÁS atrajs  
ATUA atua  
ATUAL atuaw  
ATUALIZADO atualizadu  
ATUALMENTE atuawme-tSi  
ATÉ ate  
AUDITOR awdZitoX  
AUDITÓRIO awdZitOriu  
AUGUSTO awgustu  
AULA awla  
AUMENTA awme-ta  
AUMENTOS awme-tus  
AUMENTOU awme-tow  
AUSTRALIANO awstralia-nu  
AUSÊNCIA awze-sia  
AUTOMATICAMENTE awtomatSikame-tSi  
AUTOMOTORES awtomotoris  
AUTOMÓVEIS awtomOvejs  
AUTOR awtoX  
AUTORA awtora  
AUTORIDADE awtoridadzi  
AUTORIZOU awtorizow  
AUXILIARES awSiliaris  
AVALIADOS awaliadus  
AVALIAÇÃO awaliasa-w-  
AVANÇADOS awa-sadus  
AVANÇARMOS awa-saRmus  
AVANÇO awa-su  
AVANÇOS awa-sus  
AVANÇOU awa-sow  
AVENIDA awenida  
AVES awis  
AVÓ awo  
AÇOUGUE asowge  
AÇÃO awa-w-  
AÇÕES awa-w-j-s  
AÉREA aErea  
Aí ai  
BAHIA baia  
BAIANA baja-na  
BAIANO baja-nu  
BAIRROS bajRus  
BAIXANDO bajSa-du  
BAIXAS bajSas  
BAIXO bajSu  
BAIXOS bajSus  
BALANÇA bala-sa

BALANÇAM bala-sa-w-  
BALANÇO bala-su  
BALAS balas  
BALELA baLEla  
BANCADA ba-kada  
BANCAS ba-kas  
BANCO ba-ku  
BANCOS ba-kus  
BANCÁRIA ba-karia  
BANCÁRIOS ba-karios  
BANDA ba-da  
BANDEIRA ba-dejra  
BANHO ba-Ju  
BARCO baXku  
BARGANHANDO baRga-Ja-du  
BARRIGA baRiga  
BARRIOS baRus  
BASE bazi  
BASQUETE basketSi  
BASTA basta  
BASTANTE basta-tSi  
BASTARÁ bastara  
BASTAVA bastava  
BATE ba-tSi  
BATENTE bate-tSi  
BATEU batew  
BATISMAL batSizmaw  
BATIZADO batSizadu  
BATIZADOS batSizadus  
BEBETO bebetu  
BEBÉ bebe  
BEIJANDO bejZa-du  
BELEZA beleza  
BELGAS bEwgas  
BEM be-j-  
BENEFICIADOS benefisiadus  
BENEVIDES benevidZis  
BENS be-j-s  
BESTEIRAS bestejras  
BETERRABA beteRaba  
BEZERRA bezeRa  
Bía bía  
BICHEIRO biSejru  
BICHEIROS biSejrus  
BICHO biSu  
BIENAL bienaw  
BILHÃO biLa-w-  
BILHÕES biLo-o-j-s  
BINACIONAL binasionaw  
BIODIVERSIDADE bioDiveXsidadzI  
BISCAIA biskaja  
BISCOITO biskojtu  
BISOL bizOw  
BOA boa  
BOAS boas  
BOCEJOU boseZow  
BOLA bola  
BOLADAS boladas  
BOLO bolu  
BOLÍVIA bolívia  
BOM bo-  
BONDE bo-dzi  
BORDO boRdu  
BOSQUE boske  
BOTAFOGO botafogu  
BOTELHO botELu  
BRANCO bra-ku  
BRANDIA bra-dzia  
BRASIL braziw  
BRASILEIRA brazilejra  
BRASILEIRO brazilejru  
BRASILEIROS brazilejrus  
BRASÁRIO brazariu  
BRASÍLIA brazília  
BRAZ brajs  
BRIGAMOS brigamus  
BRINCO bri-ku  
BRITÂNICA brita-nika  
BRITÂNICO brita-niku  
BRIZOLA brizola  
BRONZEADA bro-zeada  
BUENOS bue-nus  
BUROCRACIA burokrasia  
BURRICE buRisi  
BUSCA buska  
BUSCAR buskaX  
BUSCO busku  
BÁSICAS bazikas  
BÍBLIA bíblia  
BÓIA bOja  
BÓIAS-FRIAS bOjasfrias  
BÓSNIA bOznia  
CABEÇA kabesa  
CABINE kabi-ni  
CADA kada  
CADASTRADOS kadastradus  
CADEIA kadeja  
CADEIRAS kadejras  
CADERNETAS kadeRnetas  
CADERNO kadeRnu  
CADERNOS kadeRnus

CAFÉ kafE  
CAI kaj  
CAIR kaiX  
CAIU kaiw  
CAIXA kajSa  
CAIXINHA kajSi-Ja  
CAIXÃO kajSa-w-  
CALCINHA kaws-i-Ja  
CALOR kaloX  
CALÚNIA kalu-nia  
CAMA ka-ma  
CAMAROTE kamarotSi  
CAMARÕES kamaro-j-s  
CAMBIAL ka-biaw  
CAMINHO kami-Ju  
CAMINHÃO kami-Ja-w-  
CAMISA kamiza  
CAMISINHA kamizi-Ja  
CAMPANHA ka-pa-Ja  
CAMPANHAS ka-pa-Jas  
CAMPEONATO ka-peonatu  
CAMPEA ka-pea-  
CAMPEÃO ka-pea-w-  
CAMPO ka-pu  
CANADÁ kanada  
CANDIDATO ka-dzidatu  
CANDIDATOS ka-dzidatus  
CANDIDATURA ka-dzidatura  
CANETAS kanetas  
CANSOU ka-sow  
CANTO ka-tu  
CANÇÃO ka-sa-w-  
CAPACITADA kapasitada  
CAPITAIS kapitajs  
CAPITAL kapitaw  
CAPIVARI kapivari  
CAPRICHAR kapriSaX  
CAPTADAS kaptadas  
CAPTAM kapta-w-  
CAPTAÇÃO kaptasa-w-  
CAPÍTULO kapitulu  
CARA kara  
CARACTERÍSTICAS karakteristSikas  
CARACTERÍSTICOS karakteristSikus  
CARAS karas  
CARGO kaRgu  
CARLOS kaRlus  
CARNAVAL kaRnawaw  
CARNE kaRni  
CARNES kaRnis  
CARO karu  
CARREGAMENTO kaRegame-tu  
CARREIRA kaRejra  
CARRO kaRu  
CARROS kaRus  
CARTÃO kaXta-w-  
CARTÕES kaXto-j-s  
CARVÃO kaRva-w-  
CASA kaza  
CASAGRANDE kazagra-dzi  
CASAL kazaW  
CASAR kazaX  
CASCA kaska  
CASO kazu  
CATALISADORES katalizadwis  
CATEGÓRICO kategOriku  
CATÁLOGO katalogu  
CATÁLOGOS katalogus  
CAUSA kawza  
CAUSAS kawzas  
CAUSOU kawzow  
CAUTELOSO kawtelozu  
ÇAÇA-NIQUEIS kasanikejs  
CAÍ kai  
CAÍRAM kaira-w-  
CBF sebeefi  
CEDER sedeX  
CEDO sedu  
CELSIUS sewsiws  
CELSO sEwsu  
CEM se-j-  
CENAS se-nas  
CENOURA senowras  
CENOURAS senowras  
CENSO se-su  
CENTAVO se-tavu  
CENTO se-tu  
CENTRAL se-traw  
CENTRALIZAVA se-tralizawa  
CENTRO se-tru  
CENÁRIO senariu  
CERCA seXka  
CERCADINHO seXkadzi-Ju  
CEREBRAL se rebraw  
CERTO seXtu  
CERÂMICA sera-mika  
CESTAS sestas  
CHAMA Sa-ma  
CHAMADAS Samadas

CHAMADOS Samadus  
 CHAMAR SamaX  
 CHAMÁ-LO Samalu  
 CHANCE Sa-si  
 CHANCES Sa-sis  
 CHATEADA Sateada  
 CHAVE Savi  
 CHEFONA Sefo-na  
 CHEGA Sega  
 CHEGADO Segadu  
 CHEGAM Segã-w-  
 CHEGARAM Segara-w-  
 CHEGAVAM Segava-w-  
 CHEGOU Segow  
 CHEIA Seja  
 CHEIO Seju  
 CHEQUES Sekis  
 CHILE Sili  
 CHINES Sinejs  
 CHOCOLATE SokolatSi  
 CHORANDO Sora-du  
 CHOVER SoveX  
 CHUVA Suva  
 CHUVAS Suvas  
 CIBRASA sibraza  
 CIDADANIA sidania  
 CIDADE sidaZi  
 CIDADÃO sidada-w-  
 CIENTÍFICAS sie-tSifika  
 CIGANO sigã-nu  
 CIGARROS sigaRus  
 CIMA si-ma  
 CINCO si-ku  
 CINQUENTA si-kwe-ta  
 CINTILAÇÕES si-tSilaso-j-s  
 CINTO si-tu  
 CIRCULA siXkula  
 CIRCULAÇÃO siXkulasã-w-  
 CIRCUNSTANCIAS siXku-sta-siajs  
 CIRO siru  
 CIRURGIA siruRZia  
 CIRURGIÃO siruRZia-w-  
 CITADO sitadu  
 CIVIL siviw  
 CIVIS sivi  
 CLARA klara  
 CLARO klaru  
 CLASSE klasi  
 CLASSIFICADO klasifikadu  
 CLASSIFICAM-SE klasifika-w-si  
 CLASSIFICAR klasifikaX  
 CLASSIFICAÇÃO klasifikasa-w-  
 CLASSIFICOU klasifikow  
 CLICAR klikaX  
 CLIENTE klie-tSi  
 CLIENTES klie-tSis  
 CLIMA kli-ma  
 CLUBE klubi  
 CLUBES klubis  
 CLÁUDIO klawdziu  
 CLÍNICO kli-niku  
 CLÓVIS klOvis  
 COBRA kobra  
 COBRADOR kobraadoX  
 COBRANÇAS kobra-sas  
 COBRAR kobraX  
 COELHO koELu  
 COISA kojza  
 COISAS kojzas  
 COISSIMA kojzisima  
 COLABORADOR kolaboradoX  
 COLEGA kolega  
 COLEGIAL koleZiaw  
 COLETADOS koletadus  
 COLETE koletSi  
 COLEÇÃO kolesa-w-  
 COLOCADO kolokadu  
 COLOCANDO koloka-du  
 COLOCAR kolokaX  
 COLOCAÇÃO kolokasa-w-  
 COLOCAÇÕES kolokaso-j-s  
 COLOCOU kolokow  
 COLOMBIANO kolo-bia-nu  
 COLORIDOS koloridus  
 COLUMBIA kolu-bia  
 COLUNA kolu-na  
 COLÉGIO koleZiu  
 COLOMBIA kolo-bia  
 COM ko-  
 COMANDA komã-da  
 COMBATE ko-batSi  
 COMECE komesi  
 COMEMOROU komemorow  
 COMENTO kome-tu  
 COMER komeX  
 COMERCIAIS komeXsiajs  
 COMERCIAL komeXsiaw  
 COMERCIALIZADOS komeXsializadus  
 COMETE kometSi  
 COMETEU kometew  
 COMEÇA komesa  
 COMEÇAM komesa-w-  
 COMEÇAR komesaX  
 COMEÇARAM komesara-w-  
 COMEÇOU komesow  
 COMIDA komida  
 COMIDAS komidas  
 COMISSÃO komisa-w-  
 COMITÊ komite  
 COMO ko-mu  
 COMPANHEIROS ko-pa-Jejrus  
 COMPANHIA ko-pa-Jia  
 COMPARADA ko-parada  
 COMPARADO ko-paradu  
 COMPARECERAM ko-paresera-w-  
 COMPETIR ko-petSiX  
 COMPETITIVIDADE ko-petSitSividadZi  
 COMPETIÇÃO ko-petSisa-w-  
 COMPLETA ko-pleta  
 COMPLETO ko-pletu  
 COMPLETOU ko-pletow  
 COMPLEXO ko-pleksu  
 COMPORTAMENTO ko-poXtame-tu  
 COMPORTASSE ko-poXtasi  
 COMPORTO ko-poXtu  
 COMPOSIÇÕES ko-poziso-j-s  
 COMPOSTA ko-posta  
 COMPOSTAS ko-postas  
 COMPRANDO ko-pra-du  
 COMPRAR ko-praX  
 COMPRAS ko-pras  
 COMPREI ko-prej  
 COMPROMETEU ko-prometew  
 COMPROMETIDOS ko-prometSidus  
 COMPROMISSO ko-promisu  
 COMPUTADOR ko-putadoX  
 COMPUTADORES ko-putadoris  
 COMUM komu-  
 COMUNICADO komunikadu  
 COMUNICAÇÃO komunikasa-w-  
 COMUNS komu-s  
 COMÉRCIO komEXsiu  
 CONCEDIDOS ko-sezdizus  
 CONCEITO ko-sejtu  
 CONCENTRAM-SE ko-se-tra-w-si  
 CONCENTRARÃO ko-se-trara-w-  
 CONCENTRAÇÃO ko-se-trasa-w-  
 CONCILIAÇÕES ko-siliaso-j-s  
 CONCLUIR ko-kluiX  
 CONCLUSÃO ko-kluzã-w-  
 CONCLUSÕES ko-kluzo-j-s  
 CONCLUÍDAS ko-kluidas  
 CONCLUIRAM ko-kluirã-w-  
 CONCORDAR ko-koRdaX  
 CONCORRÊNCIA ko-koRe-sia  
 CONCURSO ko-kuXsu  
 CONDIÇÃO ko-dZisa-w-  
 CONDIÇÕES ko-dZiso-j-s  
 CONDUZIDA ko-duzida  
 CONEXÃO koneSa-w-  
 CONFEDERAÇÃO ko-federasa-w-  
 CONFERE ko-feri  
 CONFERÊNCIA ko-feresia  
 CONFESSA ko-fesa  
 CONFIRMADOS ko-fiRmadus  
 CONFORMAM ko-foRma-w-  
 CONFORTO ko-foXtu  
 CONFUSO ko-fuzu  
 CONFUSÃO ko-fuza-w-  
 CONGELAMENTO ko-Zelame-tu  
 CONHECE ko-Jesi  
 CONHECI ko-Jesi  
 CONHECIDO ko-Jesidu  
 CONHECIMENTO ko-Jesime-tu  
 CONQUISTA ko-kista  
 CONQUISTAR ko-kistaX  
 CONQUISTOU ko-kistow  
 CONSAGRADOS ko-sagradus  
 CONSAGROU ko-sagrow  
 CONSCIENTIZANDO ko-sie-tSiza-du  
 CONSCIÊNCIA ko-sie-sia  
 CONSEGUEM ko-sege-j-  
 CONSEGUIA ko-segja  
 CONSEGUIRAM ko-segjra-w-  
 CONSEGUIRMOS ko-segjRmus  
 CONSENSOS ko-se-sus  
 CONSERVAÇÃO ko-seRvasã-w-  
 CONSIDERADO ko-sideradu  
 CONSIDERAM ko-siderã-w-  
 CONSIDERAÇÕES ko-sideraso-j-s  
 CONSIDEROU ko-siderow  
 CONSISTE ko-sistsi  
 CONSOLAÇÃO ko-solasa-w-  
 CONSOLIDARIA ko-solidaria  
 CONSOLIDAÇÃO ko-solidasa-w-  
 CONSTANTE ko-stã-ti  
 CONSTITUCIONAL ko-stitusionaw  
 CONSTITUIÇÃO ko-stitujã-w-  
 CONSTRANGEDORA ko-stra-Zedora  
 CONSTRUÇÃO ko-strusa-w-  
 CONSULTADO ko-suwtdadu  
 CONSULTAS ko-suwts  
 CONSUMIDOR ko-sumidoX  
 CONSUMIDORES ko-sumidoris  
 CONSUMIDOS ko-sumidus  
 CONSUMO ko-su-mu  
 CONTA ko-ta  
 CONTABILIDADE ko-tabilidadZi  
 CONTAR ko-taX  
 CONTAS ko-tas  
 CONTATO ko-tatu  
 CONTATOS ko-tatus  
 CONTEMPLAR ko-te-plaX  
 CONTESTAÇÃO ko-testasa-w-  
 CONTEXTO ko-testu  
 CONTINGENTE ko-tSi-Ze-tSi  
 CONTINUA ko-tSinua  
 CONTINUAM ko-tSinua-w-  
 CONTINUAVAM ko-tSinuã-w-  
 CONTISTA ko-tSista  
 CONTRA ko-tra  
 CONTRACEPTIVO ko-traseptSivu  
 CONTRADIÇÕES ko-tradziso-j-s  
 CONTRAPARTIDAS ko-trapaXtSidas  
 CONTRATAR ko-trataX  
 CONTRATAÇÃO ko-tratasã-w-  
 CONTRIBUIR ko-tribuiX  
 CONTRIBUIÇÕES ko-tribujso-j-s  
 CONTROLA ko-trola  
 CONTROLAR ko-trolaX  
 CONTROLE ko-trolli  
 CONTUNDIU ko-tu-dZiw  
 CONTINUA ko-tSi-nua  
 CONVENCER ko-ve-seX  
 CONVERSOES ko-veXso-j-s  
 CONVERTE ko-veXSi  
 CONVIDADO ko-vidadu  
 CONVIVE ko-vivi  
 CONVIVER ko-viveX  
 CONVOCADA ko-vokada  
 CONVOCADOS ko-vokadus  
 CONVÊNIO ko-ve-niu  
 COORDENAR ko-oRdenaX  
 COPA kopa  
 COPAS kopas  
 COR koX  
 CORAÇÃO korasa-w-  
 CORES koris  
 CORINTIANO koritSiã-nu  
 CORONEL koronEw  
 CORPO koXpu  
 CORREGEDORIA koReZedoria  
 CORRER koReX  
 CORRESPONDÊNCIA koRespo-de-si  
 a  
 CORREÇÃO koResã-w-  
 CORREÇÕES koReso-j-s  
 CORRIDA koRida  
 CORTE koXtSi  
 CORÉIA korEja  
 COSTUMA-SE kostu-masi  
 COTADO kotadu  
 COTAÇÃO kotasa-w-  
 COVAS kovas  
 COXA koSa  
 CRAQUE kraake  
 CREDIBILIDADE kreDibilidãdZi  
 CREMARAM kre-mara-w-  
 CRESCER kresce-j-  
 CRESCIMENTO kresime-tu  
 CRIADO kriadu  
 CRIANÇAS kria-sas  
 CRIAR kriaX  
 CRIATIVIDADE kria-tSividãdZi  
 CRIAÇÃO kria-sã-w-  
 CRIME kri-mi  
 CRIMINAL kriminaw  
 CRIOLITA kriolita  
 CRIOU kriow  
 CRISE krizi  
 CRISTIANO kristSiã-nu  
 CRITICAR kritSikaX  
 CRITICOU kritSikow  
 CRUCIAL krusiaw  
 CRUZ krujs  
 CRUZEIRENSES kruzejre-sis  
 CRUZEIRO kruzejru  
 CRUZEIROS kruzejrus  
 CRÍTICA kritSika  
 CRÍTICAS kritSikas  
 CRÍTICOS kritSikus  
 CUIDADO kujdadu  
 CUIDADOSAMENTE kujdadOzãme-tSi  
 l  
 CUJA kuZa  
 CULPA kuwpa  
 CULPOSO kuwpozu  
 CULTURA kuwtura  
 CULTURAL kuwturaw  
 CUMPRIDO ku-primu  
 CUMPRIR ku-primu  
 CURAR-SE kuraXsi  
 CURIOSIDADES kuriozidãdZis

CURITIBA kuritSiba  
CURSO kuXsu  
CURSOS kuXsus  
CURTA kuXta  
CURTO kuXtu  
CUSPARADA kusparada  
CUSTA kusta  
CUSTAR kustaX  
CUSTO kustu  
CUSTOS kustus  
CUTUQUE-O kutukeu  
CÁLCULO ka kwulu  
CÁLCULOS kawkulus  
CÂMARA ka-mara  
CÂMBIO ka-biu  
CÂMERAS ka-meras  
CÉLIO s Eliu  
CÉREBRO s Erebru  
CÉU s Ew  
CÍNIA si-tSia  
CÓDIGO kOdZigu  
CÓPIA kO pia  
DADA da da  
DADOS da dus  
DALIdali  
DANCETERIAS da-seterias  
DANOS da-nus  
DANÇA da-sa  
DAQUI da ki  
DAR da X  
DARIA da ria  
DARTIU da XtSiw  
DARÁ da ra  
DAS da s  
DATA da ta  
DATAFOLHA data foLa  
DAVA-ME da vami  
DAIdai  
DE dZi  
DEBAIXO de ba jSu  
DEBATE de ba tSi  
DECANATO de ka na tu  
DECIDIRAM de si dZira-w-  
DECIDIRÁ de si dZira  
DECIDIU-SE de si dZi wsi  
DECISÃO de si za-w-  
DECISÕES de si zo-j-s  
DECLARAÇÕES de kla ra so-j-s  
DECLIVIDADE de kla vi da dZi  
DECORADO de ko ra du  
DECORATIVOS de ko ra tSi vus  
DECRETOU de kre to w  
DEDICADA de dZi ka da  
DEFASAGEM de fa za Z e-j-  
DEFENDE de fe~ dZi  
DEFENDERAM de fe~ de ra-w-  
DEFENSIVO de fe~ si vU  
DEFESA de fe za  
DEFINIDAS de fi ni das  
DEFINIDO de fi ni du  
DEFINIR de fi ni X  
DEFLAGRADO de fla gra du  
DEIXAR de jSa X  
DEIXARAM de jSa ra-w-  
DEIXE de jSi  
DEIXOU de jSo w  
DELA de la  
DELAS de la s  
DELE de li  
DELEGACIAS de le ga si as  
DELEGADA de le ga da  
DELEGADOS de le ga dus  
DELEGAÇÃO de le ga sa-w-  
DELEGAÇÕES de le ga so-j-s  
DELES de li s  
DEMAIS de ma j s  
DEMANDA de ma- da  
DEMISSÃO de mi sa-w-  
DEMÍTIU de mi tSi w  
DEMOCRACIA de mo kra si a  
DEMOGRÁFICO de mo gra fi ku  
DEMONSTRA de mo- stra  
DEMONSTRAR de mo- stra X  
DEMORA de mo ra  
DEMORAR de mo ra X  
DENSIDADE de- si da dZi  
DENTES de- tSi s  
DENTISTA de- tSi sta  
DENTRO de- tru  
DENUNCIAR de nu- si ow  
DEPARTAMENTO de pa Xtame-tu  
DEPENDER de pe~ de X  
DEPOIMENTOS de po jme-tus  
DEPOIS de po j s  
DEPUTADO de pu ta du  
DEPUTADOS de pu ta dus  
DEPÓSITOS de pO zitu s  
DERIVADOS de ri va dus  
DERROTA de Ro ta  
DESAFIO de za fi u

DESALOJADAS de za lo Za das  
DESAPARECE de za pa re si  
DESAPRENDE de za pre- dZi  
DESARMAR de za R ma X  
DESCARGA de ska R ga  
DESCARTA de ska Xt a  
DESCEM de se- j-  
DESCOBERTOS de sko be Xt us  
DESCOBIRE de sko bri  
DESCONFIO de sko- fi u  
DESCULPAS de sku wpas  
DESCULPÁVEL de sku wpave w  
DESDE de z dZi  
DESEJO de ze Zu  
DESEMPENHO de ze- pe- Ju  
DESEMPREGO de ze- pre gu  
DESENVOLVENDO de ze- v ow ve- du  
DESENVOLVIDOS de ze- v ow vi dus  
DESENVOLVIMENTO de ze- v ow vi me- tu  
DESEQUILÍBRIO de ze ki li bri u  
DESESTIMULOU de ze st i mu lo w  
DESFALCADA de fa l ka da  
DESFEITA de se fe i ta  
DESFILE de se fi li  
DESFILES de se fi lis  
DESGASTE de z ga st Si  
DESGRAÇADO de z gra sa du  
DESLIZAMENTO de z li za me- tu  
DESLOCAMENTOS de z lo ka me- tus  
DESMANDOS de z ma- dus  
DESNECESSÁRIA de z ne se sa ri a  
DESPENCA de se pe- ka  
DESPERDIÇADAS de se pe R dZi sa das  
DESPESA de se pe za  
DESPREZADOS de se pre za dus  
DESRESPEITO de z res pe j tu  
DESSA de sa s  
DESSAS de sa s  
DESSE de si  
DESSES de si s  
DESTA de sta  
DESTAQUES de sta ki s  
DESTE de st Si  
DESTINADO de st Si na du  
DESTRUIR de st ru i X  
DESTRUIRIA de st ru i ri a  
DESVALORIZAÇÃO de z va lo ri za sa-w-  
DESVALORIZE de z va lo ri zi  
DETALHE de ta Li  
DETENTOS de te- tus  
DETENÇÃO de te- sa-w-  
DETERIORADO de te ri o ra du  
DETETIVE de te tSi vi  
DETIDOS de tSi dus  
DETURPADA de tu X pa da  
DEU de w  
DEUS de ws  
DEVE de vi  
DEVEM de ve- j-  
DEVEMOS de ve- mus  
DEVERIA de ve ri a  
DEVERÁ de ve ra  
DEVERÃO de ve ra-w-  
DEVIA de vi a  
DEVIDO de vi du  
DEVOLUÇÃO de vo lu sa-w-  
DEZ de j s  
DEZEMBRO de ze- bru  
DEZENOVE de ze no vi  
DEZESSEIS de ze se j s  
DEZESSETE de ze se tSi  
DEZOITO de zo j tu  
DIA dZi a  
DIABO dZi a bu  
DIABOS dZi a bus  
DIAGNÓSTICO dZi ag no st Si ku  
DIANTE dZi a- tSi  
DIAS dZi a s  
DICIONÁRIO dZi si o na ri u  
DIESE dZi e zi  
DIETA dZi e ta  
DIFERENTE dZi fe re- tSi  
DIFERENÇA dZi fe re- sa  
DIFERENÇAS dZi fe re- sas  
DIFICULDADE dZi fi ku w da dZi  
DIFICULDADES dZi fi ku w da dZi s  
DIFICULTARIA dZi fi ku w ta ri a  
DIFÍCIL dZi fi si w  
DIGESTÃO dZi Z e sta-w-  
DIGO dZi gu  
DIMINUIR dZi mi nu i X  
DINAMIZAR dZi na mi za X  
DINHEIRO dZi ~ Je j ru  
DINÂMICO dZi na- mi ku  
DIPLOMACIA dZi plo ma si a  
DIREITA dZi re j ta  
DIREITO dZi re j tu  
DIRETAS dZi re ta s  
DIRETORIA dZi re to ri a  
DIREÇÃO dZi re sa-w-  
DIRIGEM dZi ri Z e- j-

DIRIGIDO dZi ri Z i du  
DISCIPLINA dZi si pli- na  
DISCORDA dZi sko R da  
DISCOS dZi sku s  
DISCRETO dZi sk re tu  
DISCURSO dZi sku X su  
DISCUTIR dZi sku tSi X  
DISFARÇA dZi si fa X sa  
DISFEZ dZi si fe j s  
DISPARADOS dZi pa ra dus  
DISPONÍVEL dZi spo ni ve w  
DISPOSIÇÃO dZi spo zi sa-w-  
DISPUTA dZi spu ta  
DISPUTOU dZi spu to w  
DISQUETE dZi sku te Si  
DISSE dZi si  
DISSERAM dZi se ra-w-  
DISSO dZi su  
DISTINTOS dZi st i tu- tus  
DISTORCIDA dZi sto X si da  
DISTRIBUIRÁ dZi st ri bu j ra  
DISTRIBUIU dZi st ri bu i w  
DISTRIBUÍDA dZi st ri bu i da  
DITAS dZi ta s  
DIVERSOS dZi ve X sus  
DIVERSÃO dZi ve X sa-w-  
DIVERTIR dZi ve X tSi X  
DIVERTIRAM dZi ve X tSi ra-w-  
DIVISÃO dZi vi za-w-  
DIVULGA dZi vu w ga  
DIVULGADA dZi vu w ga da  
DIVULGADO dZi vu w ga du  
DIVULGAR dZi vu w ga X  
DIZ dZi s  
DIZEM dZi ze- j-  
DIZENDO dZi ze- du  
DIZER dZi ze X  
DIZIA dZi zi a  
DIZIAM dZi zi a-w-  
DIÁRIO dZi a ri u  
DIÂMETRO dZi a- me tru  
DO du  
DOADOR do a do X  
DOADOS do a dus  
DOCE do si  
DOCUMENTO do ku me- tu  
DOCUMENTOS do ku me- tus  
DOCUMENTÁRIO do ku me- ta ri u  
DOENTE do e- tSi  
DOENÇA do e- sa  
DOG MÁTICA do g ma tSi ka  
DOIS do j s  
DOMINADOS do mi na dus  
DOMINARÁ do mi na ra  
DOMINGO do mi- gu  
DOMÍNIO do mi- ni u  
DOR do X  
DORMIR do R mi X  
DOROTÉIA do ro te E ja  
DOS dos  
DOSAR do Za X  
DOZE do zi  
DRAMATURGOS dra ma tu R gus  
DROGA dro ga  
DROGAS dro gas  
DUARTE du a XtSi  
DUAS du as  
DUPLICIDADE du pli si da dZi  
DURANTE du ra- tSi  
DURAÇÃO du ra sa-w-  
DUZENTAS du ze- ta s  
DUZENTOS du ze- tus  
DÁ da  
DÃO da-w-  
DÉBITO de bi tu  
DÉCADA de ka da  
DÉCIMO de si mu s  
DÉCIMOS de si mu s  
DÍZIMO dZi zi mu  
DÓLAR do la X  
DOLARES do la ri s  
DUVIDAS du vi da s  
Ei  
ECOLOGIA e ko lo Zi a  
ECOLÓGICO e ko lo Z i ku  
ECONOMIA e ko no mi a  
ECONOMIZOU e ko no mi zo w  
ECONÔMICA e ko no- mi ka  
ECONÔMICO e ko no- mi ku  
ECOTURISMO e ko tu ri z mu  
EDIFÍCIO de dZi fi si u  
EDIÇÃO de dZi sa-w-  
EDIÇÕES de dZi so-j-s  
EDUARDO e du a Ru du  
EDUCAÇÃO e du ka sa-w-  
EFEITOS e fe j tu s  
EFICIÊNCIA e fi si e- si a  
ELA Ela  
ELAS Ela s  
ELE eli  
ELEITO ele j tu

ELEITOR elejtoX  
 ELEITORAIS elejtorajs  
 ELEITORAL elejtoraw  
 ELES elis  
 ELEVADOS elevadus  
 ELIMINADA eliminada  
 ELIMINAMOS eliminamus  
 ELIMINAR eliminaX  
 ELIMINARMOS eliminaRmus  
 ELOGIOSA eloZiOza  
 ELÁSTICA elastika  
 ELÉTRICA elEtrika  
 ELÉTRICOS elEtrikus  
 EM e-j  
 EMANCIPAÇÃO ema-sipasa-w-  
 EMBAIXO e-bajSu  
 EMBALAGENS e-balaZe-j-s  
 EMBARAÇOSAS e-barasOzas  
 EMBARCOU e-baXkow  
 EMBASADA e-bazada  
 EMBRIAGAM e-briaga-w-  
 EMISSORAS emisoras  
 EMISSÃO emisa-w-  
 EMOCIONANTE emosiona-tSi  
 EMPATAM e-pata-w-  
 EMPATARAM e-patara-w-  
 EMPATE e-patSi  
 EMPREGADO e-pregadu  
 EMPREGADOS e-pregadus  
 EMPREGO e-pregu  
 EMPRESA e-preza  
 EMPRESAS e-prezas  
 EMPRESTAR e-prestaX  
 EMPRESÁRIOS e-prezarius  
 EMPRÉSTIMOS e-prEstSimus  
 ENCARREGADO e-kaRegadu  
 ENCERRADA e-seRada  
 ENCOMENDAS e-kome-das  
 ENCONTRADOS e-ko-tradus  
 ENCONTRAR e-ko-traX  
 ENCONTRAREM e-ko-trare-j-  
 ENCONTRO e-ko-tru  
 ENCRAVE e-kravi  
 ENDEREÇOS e-deresus  
 ENDOCRINOLOGIA e-dokrinoloZia  
 ENERGIA eneRZia  
 ENFEITAR e-fejtaX  
 ENFIM e-fi-  
 ENFRENTAR e-fre-taX  
 ENGANA e-ga-na  
 ENGENHARIA e-Ze-Jaria  
 ENGRAÇADO e-grasadu  
 ENORME enoRmi  
 ENQUADRAMENTO e-kwadrame-tu  
 ENQUANTO e-kwa-tu  
 ENSINO e-si-nu  
 ENTANTO e-ta-tu  
 ENTENDER e-te-deX  
 ENTRA e-tra  
 ENTRADA e-trada  
 ENTRADAS e-tradas  
 ENTRAR e-traX  
 ENTRARAM e-trara-w-  
 ENTRAVA e-trava  
 ENTRE e-tri  
 ENTREGUE e-trege  
 ENTRETANTO e-treta-tu  
 ENTREVISTA e-trevista  
 ENTREVISTADOS e-trevistadus  
 ENTREVISTOU e-trevistow  
 ENTROU e-trow  
 ENTUSIASMADO e-tuziazmadu  
 ENTUSIASMO e-tuziazmu  
 ENTÃO e-ta-w-  
 ENVELHECIMENTO e-veLesime-tu  
 ENVIADAS e-viadas  
 ENVOLVIMENTO e-vowvime-tu  
 ENXERGAR e-SeRgaX  
 ENÉAS enEas  
 EPISÓDIO epizOdZiu  
 EQUILIBRAR ekilibraX  
 EQUIPE ekipi  
 EQUIPES ekjpis  
 ERA era  
 ERAM era-w-  
 ERRADA eRada  
 ERRAMOS eRa-mus  
 ERRAR eRaX  
 ERRO eRu  
 ERROU eRow  
 ERUNDINA eru-dZi-na  
 ESCALAÇÃO eskalasa-w-  
 ESCALÕES eskalo-j-s  
 ESCLARECEU esklaresew  
 ESCLARECIMENTO esklaresime-tu  
 ESCOLA eskola  
 ESCOLAS eskolas  
 ESCOLHA eskola  
 ESCOLHIDO eskolidu  
 ESCOLHIDOS eskolidus  
 ESCONDIDAS eskodZidas  
 ESCREVERIA eskreveria  
 ESCRITOR eskritoX  
 ESFORÇO esfoXsu  
 ESFORÇOS esfoXsus  
 ESGOTOU eZgotow  
 ESNOBAR eznobaX  
 ESPACIAIS espasias  
 ESPECIAL espasias  
 ESPANHA espa-Ja  
 ESPANHOL espa-JOw  
 ESPANHOLA espa-Jola  
 ESPANHÓIS espa-JOjs  
 ESPAÇO espasu  
 ESPECIALISTA espesialista  
 ESPECIALMENTE espesiamme-tSi  
 ESPECIFICAMENTE espesifikame-tSi  
 ESPECTRO espektru  
 ESPECÍFICO espesifiku  
 ESPELHO espELu  
 ESPERADOS esperadus  
 ESPERAVA esperava  
 ESPETÁCULOS espetakulus  
 ESPIRITUAL espirituaw  
 ESPORTES espoxTis  
 ESPREMER espremeX  
 ESPÉCIE espEsi  
 ESQUECER eskeseX  
 ESQUEMA eske-ma  
 ESQUERDA eskeRda  
 ESQUERDO eskeRdu  
 ESSA Esa  
 ESSAS Esas  
 ESSE esi  
 ESSES esis  
 ESTA Esta  
 ESTABELECE estabelesi  
 ESTACIONAMENTO estasioname-tu  
 ESTADISTA estadZista  
 ESTADO estadu  
 ESTADOS estadus  
 ESTADUAL estaduaw  
 ESTAMOS esta-mus  
 ESTANTES esta-tSis  
 ESTAR estaX  
 ESTAREI estarej  
 ESTAREM estare-j-  
 ESTARIA estaria  
 ESTARIAM estaria-w-  
 ESTARÁ estara  
 ESTARÃO estara-w-  
 ESTAS Estas  
 ESTAVA estava  
 ESTAVAM estava-w-  
 ESTAÇÃO estasa-w-  
 ESTAÇÕES estaso-j-s  
 ESTE estSi  
 ESTEJA esteZa  
 ESTES estSis  
 ESTILO estSilu  
 ESTIMA-SE estSi-masi  
 ESTIMAM estSi-ma-w-  
 ESTIVER estSiveX  
 ESTIVERAM estSivera-w-  
 ESTIVEREM estSivere-j-  
 ESTOQUE estoke  
 ESTOU estow  
 ESTRANGEIRA estra-Zeira  
 ESTRANGEIROS estra-Zejrus  
 ESTRANHAS estra-Jas  
 ESTRANHO estra-Ju  
 ESTRATIFICADA estratSifikada  
 ESTRATÉGIA estratEzias  
 ESTRELAS estrelas  
 ESTRÉIA estrEja  
 ESTUDADA estudada  
 ESTUDANTE estuda-tSi  
 ESTUDANTES estuda-tSis  
 ESTUDO estudu  
 ESTUDOU estudow  
 ESTÁ esta  
 ESTÁDIO estadZiu  
 ESTÁGIO estaZiu  
 ESTÁVEIS estavejs  
 ESTÃO esta-w-  
 ESTÚDIOS estudZius  
 ETAPAS etapas  
 EU ew  
 EUFORIA ewforia  
 EUROPA ewropa  
 EUROPEU ewropew  
 EUROPÉIA ewropEja  
 EVENTO eve-tu  
 EVENTOS eve-tus  
 EVENTUAL eve-tuaw  
 EVENTUALMENTE eve-tuawme-tSi  
 EVIDENCIA evide-sia  
 EVIDENTE evide-tSi  
 EVITARIA evitaria  
 EVITE evitSi  
 EVITEI evitej  
 EVITÁ-LA evitala  
 EVOCAÇÃO evokasa-w-  
 EVOLUÇÃO evolusa-w-  
 EX-JOGADOR ezZogadoX  
 EXATAMENTE ezatame-tSi  
 EXATAS ezatas  
 EXCEPCIONAL essepsionaw  
 EXCESSIVO esseisivu  
 EXCESSO essesu  
 EXCEÇÃO essesa-w-  
 EXCLUÍ esklui  
 EXCLUSIVAMENTE eskluzivame-tSi  
 EXECUTIVO ezekuTisivu  
 EXEMPLO eze-plu  
 EXEMPLOS eze-plus  
 EXERCITAR ezeXsitaX  
 EXIBA eziba  
 EXIBIDA ezibida  
 EXIGE eziZi  
 EXIGEM eziZe-j-  
 EXIGIDAS eziZidas  
 EXIGIR eziZiX  
 EXIGÊNCIAS eziZe-sias  
 EXILADOS eziladus  
 EXISTA ezista  
 EXISTE ezistSi  
 EXISTEM eziste-j-  
 EXPECTATIVA espektatSiva  
 EXPEDITO espedZitu  
 EXPERIMENTARAM esperime-tara-w-  
 EXPERIÊNCIAS esperie-sias  
 EXPLICAÇÃO esplikasa-w-  
 EXPLICAÇÕES esplikaso-j-s  
 EXPLIQUE esplike  
 EXPLOÇÃO esploza-w-  
 EXPORTA espoXta  
 EXPOSIÇÃO espozisa-w-  
 EXPULSÕES espuzso-j-s  
 EXTERIOR esterioX  
 EXTERNA esteRna  
 EXTERNAS esteRnas  
 EXTERNO esteRnu  
 EXTERNOS esteRnus  
 EXTINÇÃO estSi-sa-w-  
 EXTRA estra  
 EXTREMAMENTE estremame-tSi  
 EXULTANTE ezuwta-tSi  
 EXÓTICAS ezOtSikas  
 FABRICANTES fabrika-tSis  
 FABRÍCIO fabrisiu  
 FACILMENTE fasiwme-tSi  
 FAÇA faka-w-  
 FAIXA fajSa  
 FAIXAS fajSas  
 FALA fala  
 FALADO faladu  
 FALAR falaX  
 FALHA faLa  
 FALHAMOS faLa-mus  
 FALHAR faLaX  
 FALO falu  
 FALOU falow  
 FALSO fawsu  
 FALTA fawta  
 FALTAM fawta-w-  
 FALTANDO fawta-du  
 FALTAR fawtaX  
 FALTAS fawtas  
 FALÊNCIA fale-sia  
 FAMA fama  
 FAMÍLIAS familias  
 FÁRIA faria  
 FAROESTES faroestSis  
 FARÃO fara-w-  
 FASE fazis  
 FASES fazis  
 FATO fatu  
 FATOR fatoX  
 FATOS fatus  
 FAVORECE favoresi  
 FAVORITA favorita  
 FAVORITISMO favoritSizmu  
 FAVORITOS favoritus  
 FAZ fajs  
 FAZEM faze-j-  
 FAZENDA faze-da  
 FAZENDAS faze-das  
 FAZENDO-Afaze-dua  
 FAZER fazeX  
 FAÇANHA fasa-Ja  
 FECHADO feSadu  
 FECHADOS feSadus  
 FECHO feSu  
 FECHOU feSow  
 FEDERAIS federajs  
 FEDERAL federaw  
 FEDERAÇÃO federasa-w-  
 FEDERAÇÕES federaso-j-s  
 FEIJADA fejZoda  
 FEIJÃO fejZa-w-  
 FEITA fejta

FEITAS fejtás  
 FEITO fejtú  
 FEITOS fejtús  
 FEMININA femini-na  
 FEMININO femini-nu  
 FEMINISTAS feministas  
 FERA fera  
 FERER ferí  
 FERIDA ferida  
 FERIDAS feridas  
 FERIDOS feridus  
 FERNANDO fernandu  
 FESTA festa  
 FESTAS festas  
 FESTIVAS festivas  
 FETO fetu  
 FEVEREIRO feverejru  
 FEZ fejs  
 FICA fika  
 FICANDO fika~du  
 FICAR fikaX  
 FICARAM fikara~w  
 FICARIA fikaria  
 FICARÃO fikara~w  
 FICO fiku  
 FICOU fikow  
 FIEL fiEw  
 FIFA fifa  
 FIGUEIREDO figejredu  
 FILHO filu  
 FILHOTE filotSi  
 FILME fiwmi  
 FILMES fiwmis  
 FILOSOFIA filozofia  
 FILÓSOFO filozofu  
 FIM fim  
 FINAL finaw  
 FINALIZAÇÕES finalizas~js  
 FINALMENTE finawme~tSi  
 FINANCEIRA fina~sejra  
 FINANCEIRAS fina~sejras  
 FINANCEIRO fina~sejru  
 FINANCIADA fina~siada  
 FINANCIAMENTO fina~siame~tu  
 FINANCIAMENTOS fina~siame~tus  
 FINLÂNDIA fila~dzia  
 FIQUEI fikej  
 FIRMAVA firmava  
 FISCAL fiskajs  
 FISCAL fiskaw  
 FISIONOMIA fizionomia  
 FIXANDO SE fisa~dusi  
 FIXO fisu  
 FIXOS fisus  
 FIXOU fisow  
 FIZ fiz  
 FIZEMOS fize~mus  
 FIZERAM fizera~w  
 FIÉIS fiEjs  
 FLEXA fleSa  
 FLORES floris  
 FLORIANÓPOLIS florianopolis  
 FLORIDA florida  
 FLU flu  
 FLUTUANTE flutua~tSi  
 FLUTUAÇÃO flutua~sa~w  
 FOGEM fogej  
 FOGO fogu  
 FOL folj  
 FOLHA folLa  
 FOR forX  
 FORA fora  
 FORAM fora~w  
 FORMA forma  
 FORMADO formadu  
 FORMADOR formadux  
 FORMAR formaX  
 FORMULAÇÃO formulasa~w  
 FORNECIMENTO fornese~tu  
 FORTALECER fortaleseX  
 FORTALECE-LO fortaleseLu  
 FORTE fortSi  
 FORÇAMOS forxsa~mus  
 FORÇAR forxsaX  
 FOS fos  
 FOSSE fosi  
 FOSSEM fose~j  
 FOTO fotu  
 FOTOGRAFAR fotografaX  
 FOTOS fotus  
 FOTÓGRAFO fotografu  
 FRANCESAS francesas  
 FRANCESES franceses  
 FRANCISCO francesisku  
 FRANCO franku  
 FRANQUIAS frankias  
 FRANÇA fransa  
 FRASE frazi  
 FRAUDE fraudzi  
 FREGUESES fregezis  
 FRENTE fre~tSi  
 FREQUENTEMENTE frekwe~teme~tSi  
 FREQUENTES frekwe~tSis  
 FRIAS frias  
 FRONTEIRA fro~tejra  
 FROTA frota  
 FRUTO frutu  
 FRUTOS frutus  
 FUGA fuga  
 FUGIDO fuzidu  
 FUGIR fuziX  
 FUL fui  
 FUNARO funaru  
 FUNCIONAM fu~sio~na~w  
 FUNCIONANDO fu~sio~na~du  
 FUNCIONÁRIO fu~sio~nariu  
 FUNCIONÁRIOS fu~sio~nariu  
 FUNDAMENTAL fu~dame~taw  
 FUNDAÇÃO fu~dasa~w  
 FUNDO fu~du  
 FUTEBOL futebow  
 FUTURAS futuras  
 FUTURO futuro  
 FUTUROS futuros  
 FÁBIO fabiu  
 FÁCIL fasiw  
 FÉRIAS ferias  
 FÍSICO fiziku  
 GABINETE gabinetSi  
 GABRIEL gabriEw  
 GALERIA galeria  
 GALPÃO gawpa~w  
 GANHA ga~Ja  
 GANHANDO ga~Ja~du  
 GANHARAM ga~Jara~w  
 GANHAVAM ga~Jawa~w  
 GANHO ga~Ju  
 GANHOS ga~Jus  
 GANHOU ga~Jow  
 GARANTA gara~ta  
 GARANTIA gara~tSi  
 GARANTIDA gara~tSi~da  
 GARANTIR gara~tSiX  
 GAROTOS garotus  
 GASTOS gastus  
 GASTRITITE gastritSi  
 GAZA gaza  
 GAÚCHO gauSu  
 GELADEIRA zeladejra  
 GENEROSO zenerozu  
 GENTE ze~tSi  
 GENÉRICO zenEriku  
 GENÉTICA zenEtsika  
 GENÉTICO zenEtsiku  
 GERA zera  
 GERADA zera~da  
 GERADOS zeradus  
 GERAL zera~w  
 GERAÇÃO zerasa~w  
 GERAÇÕES zeraso~js  
 GERENCIAMENTO zere~siame~tu  
 GERENTE zere~tSi  
 GERIR zeriX  
 GIL ziw  
 GILDA ziwda  
 GIRO ziru  
 GLOBO globu  
 GLÓRIA gloria  
 GOL gow  
 GOLS gows  
 GOUVEA gowvea  
 GOVERNO goveRnu  
 GOVERNOS goveRnus  
 GRADATIVAMENTE gradatSivame~tSi  
 GRADUAL graduaw  
 GRAMADO gramadu  
 GRANDE gra~dzi  
 GRANDES gra~dzis  
 GRATUITO gratui~tu  
 GRAUS gra~ws  
 GRAVATA gravata  
 GRAVAÇÕES gravaso~js  
 GRAVE gravi  
 GRAVES gravis  
 GRAÇA grasa  
 GREVE grevi  
 GRITO gritu  
 GROSSO grosu  
 GRUPO grupu  
 GRUPOS grupus  
 GRÁVIDA gravida  
 GRÊMIO gre~miu  
 GUARANI gwarani  
 GUERRA geRa  
 GUERREIRO gerejru  
 GUILHERME gjLermi  
 GUIMARÃES gjLemara~js  
 GUITARRISTA gjtaRista  
 GÁVEA gavea  
 GÊMEOS ze~meus  
 GÊNERO ze~neru  
 HABITACIONAIS abitacionajs  
 HAJA aZa  
 HAROLDO arOwdu  
 HAVERÁ vera  
 HAVIA avia  
 HEITOR ejtoX  
 HELENA ele~na  
 HENRIQUE e~rike  
 HIDRÁULICA idrawlika  
 HIPÓTESE ipOtezi  
 HISTÓRIA istOria  
 HISTÓRIAS istOrias  
 HOJE oZi  
 HOLANDA ola~da  
 HOMEM o~me~j  
 HOMENAGENS omenaZe~js  
 HOMENS ome~js  
 HOMICÍDIO omisidZiu  
 HOMOSSEXUAIS omoseSuajs  
 HOMOSSEXUAL omoseSuaw  
 HONDURAS o~duras  
 HORA ora  
 HORAS oras  
 HORTÊNCIA oXte~sia  
 HOSPITAL ospitaw  
 HOSPITALIDADE ospitalidadZi  
 HOTEL otEw  
 HOUVE owvi  
 HOUVER owveX  
 HUMANIDADE umanidadZi  
 HUMOR umoX  
 HUMORISTA umorista  
 HÁ a  
 HUTUS utus  
 IÁ ia  
 IDADE idadz  
 IDENTIDADE ide~tSidadZi  
 IDENTIFICADA ide~tSifikada  
 IDENTIFICADO ide~tSifikadu  
 IDIOTA idZiota  
 IDO idu  
 IDOSOS idozus  
 IDÉIA ide~ja  
 IDÉIAS ide~jas  
 IENES ie~nis  
 IGREJA igreZa  
 IGUAL igwaw  
 ILEGAL ilegaw  
 ILHÁ ilLa  
 ILHÉUS ilEws  
 ILUSTRADO ilustradu  
 ILUSTRAÇÕES ilustraso~js  
 IMAGEM imaZe~j  
 IMAGENS imaZe~js  
 IMAGINAR imazinax  
 IMAGINÁRIO imazinariu  
 IMEDIATAS ime~dZiatas  
 IMENSA ime~sa  
 IMITANDO imita~du  
 IMORTALIDADE imoXtalidadZi  
 IMPACTO i~paktu  
 IMPEDE i~pedZi  
 IMPIEDOSO i~pedozu  
 IMPOPULARIDADE i~popularidadZi  
 IMPORTA i~poXta  
 IMPORTANTE i~poXta~tSi  
 IMPORTANTES i~poXta~tSis  
 IMPORTAÇÕES i~poXtas~js  
 IMPOSSÍVEL i~posiwew  
 IMPOSTO i~postu  
 IMPOSTOS i~postus  
 IMPRENSA i~pre~sa  
 IMPUGNAR i~pugnaX  
 IMPUNEMENTE i~puneme~tSi  
 IMPUSERAM i~puzera~w  
 IMUNE imu~ni  
 IMÓVEIS imOvejs  
 INACABADO inakabadu  
 INAUGURADAS inawguradas  
 INCERTO i~seXtu  
 INCIDENTES i~side~tSis  
 INCLUI i~klui  
 INCOMODA i~komoda  
 INCOMODOU i~komodow  
 INCONSTITUCIONALIDADE i~ko~stSitu  
 sionalidadZi  
 INDEPENDENTE i~depe~de~tSi  
 INDEXAÇÃO i~desasa~w  
 INDICAÇÃO i~dzikas~w  
 INDISCUTÍVEL i~dziskusiwew  
 INDIVIDUAL i~dzividuaw  
 INDO i~du  
 INDÚSTRIA i~dustria  
 INEFICIENTE inefisie~tSi  
 INESQUECÍVEL ineskesiwew  
 INEVITÁVEL inewitawew  
 INFANTIL i~fa~tSiw  
 INFELIZ i~felis  
 INFELIZMENTE i~felizme~tSi  
 INFLAÇÃO i~flasa~w  
 INFORMADA i~formada  
 INFORMADOS i~formadus

INFORMAÇÃO i-foR masa-w-  
 INFORMAÇÕES i-foR maso-j-s  
 INFORMOU i-foR mow  
 INFRAERO i- frajru  
 INGLÊS i- glejs  
 INGRESSO i- gresu  
 INICIAIS inisiajs  
 INICIALMENTE inisia wme-tSi  
 INICIARAM inisiara-w-  
 INICIOU inisiow  
 INOCENTADO inose-tadu  
 INSCREVER i- skreveX  
 INSETOS i- setus  
 INSPIRADAS i- spiradas  
 INSPIRARAM i- spirara-w-  
 INSPIROU i- spirow  
 INSTALADO i- staladu  
 INSTALAÇÃO i- stalasa-w-  
 INSTANTE i- sta-tSi  
 INSTANTÂNEA i- sta-ta-nea  
 INSTIGANTE i- stSiga-tSi  
 INSTITUIÇÕES i- stSitujso-j-s  
 INSTÂNCIAS i- sta-sias  
 INSÔNIA i- so-nia  
 INTEGRAL i- tegraw  
 INTEGRANTES i- tegra-tSis  
 INTEGRAÇÃO i- tegrasa-w-  
 INTEIRA i- tejra  
 INTELCTUAIS i- telektuajs  
 INTELIGENTES i- teliZe-tSis  
 INTELIGÊNCIA i- teliZe-sia  
 INTENCIONAL i- te-sionaw  
 INTENSA i- te-sa  
 INTENSO i- te-su  
 INTENÇÃO i- te-sa-w-  
 INTERCOMUNICADOR i- teXkomunikad  
 oX  
 INTERESSA i- teresa  
 INTERESSADO i- teresadu  
 INTERESSANTE i- teresa-tSi  
 INTERESSE i- teresi  
 INTERIOR i- terioX  
 INTERMEDIÁRIA i- teRmedZiaria  
 INTERNA i- teRna  
 INTERNACIONAIS i- teRnasionajs  
 INTERNACIONAL i- teRnasionaw  
 INTERNALIZO i- teRnalizu  
 INTERPRETAR i- teXpretaX  
 INTERRUPÇÃO i- teRupsa-w-  
 INTERVALO i- teRvalu  
 INTERVALOS i- teRvalus  
 INTERVENÇÃO i- teRve-sa-w-  
 INTIMIDADE i- tSimidadZi  
 INTIMIDAR i- tSimidaX  
 INVASÕES i- vazoj-s  
 INVENTANDO i- ve-ta-đu  
 INVENTAR i- ve-taX  
 INVERNO i- veRnu  
 INVERTIDA i- veXtSida  
 INVESTIDORES i- vestSidoris  
 INVESTIGA i- vestSiga  
 INVESTIGAÇÕES i- vestSigaso-j-s  
 INVESTIMENTO i- vestSime-tu  
 INVESTIMENTOS i- vestSime-tus  
 INVESTINDO i- vestSi-du  
 INVESTIR i- vestSiX  
 INÍCIO inisiu  
 IR iX  
 IRIA i-ria  
 IRLANDA i-rla-da  
 IRMÃOS i- rma-w-s  
 IRRECONCILIÁVEL i- reko-siliavew  
 IRREVERSÍVEL i- revexsivew  
 IRÁ i- ra  
 IRÃO i- ra-w-  
 ISENTA i- ze-ta  
 ISOLAR i- zolaX  
 ISOLDA i- zowda  
 ISRAEL i- zraEw  
 ISSO i- su  
 ISTO i- stu  
 ITAIPU i- tajpu  
 ITALIANA i- talia-na  
 ITAMAR i- tamaX  
 ITAMARATY i- tamaratSi  
 ITÁLIA i- talia  
 IÊMEN i- e-me-  
 JAIR ZaiX  
 JANAINA Zanajna  
 JANEIRO Zanejru  
 JANELA ZaneIa  
 JAPONÊSES Zaponezis  
 JAPÃO Zapa-w-  
 JEITO Zejtu  
 JOBIM Zobi-  
 JOELHO ZoELu  
 JOGA Zoga  
 JOGADOR ZogadoX  
 JOGADORES Zogadoris  
 JOGAM Zoga-w-  
 JOGAR ZogaX  
 JOGAVAM Zogava-w-  
 JOGO Zogu  
 JOGOS Zogus  
 JORGINHO ZoRZi-Ju  
 JORNADAS ZoRnadas  
 JORNAIS ZoRnajs  
 JORNAL ZoRnaw  
 JORNALÍSTICOS ZoRnalistSikus  
 JOSÉ ZoZE  
 JOVENS Zove-j-s  
 JOÃO Zoaw-w-  
 JUDICIÁRIO Zudzisiariu  
 JUIZ Zuiz  
 JULGAMENTO Zuwgame-tu  
 JULHO ZuLu  
 JULIANA Zulia-na  
 JUNHO Zu-Ju  
 JUNIOR ZunioX  
 JUNTAS Zu-tas  
 JUNTO Zu-tu  
 JURARAM Zurara-w-  
 JURO Zuru  
 JUROS Zurus  
 JURÍDICAS ZuridZikas  
 JUS Zujs  
 JUSTIFICAR ZustSifikaX  
 JUSTIÇA ZustSisa  
 JUIZES Zuizis  
 JUÍZO Zuizuz  
 JÁ Za  
 JÔ Zo  
 LABORATÓRIOS laboratOrius  
 LADO ladu  
 LANÇADO la-sadu  
 LANÇOU la-sow  
 LARGURA laRgura  
 LATA lata  
 LATASAlataza  
 LATERAL lateraw  
 LAURA lawra  
 LAZER lazeX  
 LEGISLATIVA leZizlatSiva  
 LEGISLAÇÃO leZizlasa-w-  
 LEI lej  
 LEIA leja  
 LEIS lejs  
 LEITE lejSi  
 LEITOR lejtoX  
 LELÉ leIE  
 LEMA le-ma  
 LEMBRAL e-bra  
 LEMBRADOS le-bradus  
 LENTIDÃO le-tSida-w-  
 LENTO le-tu  
 LEONTINA leo-tSi-na  
 LER leX  
 LESIVO lezivuz  
 LESSA lesa  
 LESTE lesSi  
 LETALIDADE letalidadzI  
 LEU lew  
 LEVA leva  
 LEVANTA leva-ta  
 LEVANTAMENTO leva-tame-tu  
 LEVAR levaX  
 LEVAVAM levava-w-  
 LEVOU lewv  
 LHE Li  
 LIBERAL liberaw  
 LIBERTOU libeXtow  
 LIBRETOS libretus  
 LICENCIAMENTO lise-siame-tu  
 LIDERANÇAS lidera-sas  
 LIGA liga  
 LIGADO ligadu  
 LIGAR ligaX  
 LIGAÇÕES ligaso-j-s  
 LILA lila  
 LIMA li-ma  
 LIMITADO limitadu  
 LIMITADOS limitadus  
 LIMITE limitSi  
 LINCE li-si  
 LINCHADO li-Sadu  
 LINDAS li-das  
 LINHAS li-Jas  
 LISTA lista  
 LITERÁRIAS literarias  
 LITORAL litoraw  
 LIVRE livri  
 LIVRO livru  
 LIVROS livrus  
 LOCAIS lokajs  
 LOCAL lokaw  
 LOCALIZADA lokalizada  
 LOGO logu  
 LOJAL oZa  
 LOJAS loZas  
 LONDRES lo-dris  
 LONGE lo-Zi  
 LONGO lo-gu  
 LOTADAS lotadas  
 LOTAÇÃO lotasa-w-  
 LUA-DE-MEL luadZimEw  
 LUCRO lukru  
 LUGAR lugax  
 LUGARES lugaris  
 LULA lula  
 LUTA luta  
 LUTANDO luta-du  
 LUTO lutu  
 LUXEMBURGO luse-buRgu  
 LUXO luSu  
 LUZ lujs  
 LUIS lujs  
 LÁ la  
 LÉLIA lElia  
 LÍDER lideX  
 LÍNGUA li-gwa  
 LÚCIA lusia  
 MACACAS makakaks  
 MACACO makaku  
 MACHADO maSadu  
 MACIÇAMENTE masisame-tSi  
 MACONHA makokoJa  
 MADRUGADA madrugada  
 MAGALHÃES magala-j-s  
 MAIO maju  
 MAIOR majOX  
 MAIORES majOris  
 MAIORIA majoria  
 MAIS majs  
 MAJORITARIAMENTE mazoritariam  
 e-tSi  
 MAL maw  
 MALÁRIA malaría  
 MAMATA mamata  
 MANCHETE ma-SetSi  
 MANCUSO ma-kuzu  
 MANDADO ma-dadu  
 MANDARAM ma-dara-w-  
 MANDATO ma-datu  
 MANDOU ma-dow  
 MANEIRA manejra  
 MANGUEIRA ma-gejra  
 MANHÃ ma-Ja-  
 MANIFESTADA manifestada  
 MANIPULAÇÃO manipulasa-w-  
 MANTEVE ma-tevi  
 MANTIDO ma-tSidu  
 MANTÉM ma-te-j-  
 MANUAIS manuajs  
 MANUEL manuew  
 MANUTENÇÃO manute-sa-w-  
 MAPA mapa  
 MAPAS mapas  
 MAR maX  
 MARAVILHOSO maraviLozu  
 MARAVILHOSOS maraviLozus  
 MARCA maXka  
 MARCADA maXkada  
 MARCADO maXkadu  
 MARCAÇÃO maXkasa-w-  
 MARGEM maRZe-j-  
 MARGENS maRZe-j-s  
 MARIA maria  
 MARIDO maridu  
 MARIDOS maridus  
 MARIETA marieta  
 MARILENA marile-na  
 MARIZ maris  
 MAROLA marola  
 MARTINS maXtSi-s  
 MARÇO maXsu  
 MAS mas  
 MASSACRE masakri  
 MATA mata  
 MATAR mataX  
 MATERIAIS materiajs  
 MATERNIDADE mateRnidadzI  
 MATO matu  
 MATREIRO matrejru  
 MAU maw  
 MAÇÁ mas-a-  
 ME mi  
 MEADOS meadus  
 MEDALHA medaLa  
 MEDIADORES medZidoris  
 MEDICAMENTOS medzikame-tus  
 MEDIDA medZida  
 MEDIDAS medZidas  
 MEDO medu  
 MEDIÓCRE medZiokri  
 MEIA meja  
 MEIAS meias  
 MEIO meju  
 MEIOS meius  
 MELHOR meLoX  
 MELHORAR meLoraX  
 MELHORES meLoris  
 MELHORIA meLoria  
 MEMBRANA me-bra-na

MEMBRO me-bru  
MEMBROS me-brus  
MENOR menOX  
MENORES menOris  
MENOS me-nus  
MENSAL me-saw  
MENÇÃO me-sa-w-  
MERCADO meXkadu  
MERCADOS meXkadus  
MERCENÁRIA meXsenaria  
MERCOSUL meXkozuw  
MERENDA mere-da  
MEROS merus  
MESES mezis  
MESMA mezma  
MESMO mezmu  
METADE metaDzi  
METAMORFOSE metamoXfozi  
METAS metas  
METER metex  
METRO metru  
METROPOLITANA metropolitana-na  
METROS metrus  
MEU mew  
MEUS mews  
MICRO mikru  
MICROFONE mikrofo-ni  
MIGUEL migEw  
MIL miw  
MILHARES miLaris  
MILHÕES miLo-j-s  
MILITARES militaris  
MIM mi-  
MINATEL minatEw  
MINERAÇÃO minerasa-w-  
MINHA mi-Ja  
MINHAS mi-Jas  
MINISTRO ministru  
MINISTROS ministrus  
MINISTÉRIO ministEriu  
MINUTOS minutus  
MODA moda  
MODALIDADE modalidadZi  
MODELO modelu  
MODERADO moderadu  
MODERNAS modeRnas  
MODERNO modeRnu  
MODIFICAR modZifikaX  
MOEDA moeda  
MOEDAS moedas  
MOGI moZi  
MOLESTAMENTO molestame-tu  
MOLHADA moLada  
MOMENTO mome-tu  
MOMENTOS mome-tus  
MONSANTO mo-sa-tu  
MONTADAS mo-tadas  
MONTAGEM mo-taZej-  
MONTAR mo-taX  
MONTARAM mo-tara-w-  
MONTE mo-tSi  
MORADORES moradoris  
MORAES morajs  
MORALIDADE moralidadZi  
MORANDO mora-du  
MORDOMO moRdo-mu  
MORRE moRi  
MORRER moReX  
MORRERAM moRera-w-  
MORTALIDADE moXtalidadZi  
MORTE moXtSi  
MORTOS moXtus  
MOSTRA mostra  
MOSTRANDO mostra-du  
MOSTRAR mostraX  
MOSTROU mostrow  
MOTIVO motSiv  
MOTORES motoris  
MOTOS motus  
MOVELEIRO movelejru  
MOVIMENTAÇÃO movime-tasa-w-  
MOVIMENTO movime-tu  
MOÇAS mosas  
MUDA muda  
MUDANÇA muda-sa  
MUDANÇAS muda-sas  
MUDAR mudaX  
MUITA muj-ta  
MUITAS muj-tas  
MUITO muj-tu  
MUITOS muj-tus  
MULHER muLeX  
MULHERES muLeris  
MULTA muwta  
MUNDIAL mu-dZiaw  
MUNDO mu-du  
MUNICIPAIS munisipajs  
MUNICIPAL munisipaw  
MUNICIPALIZAÇÃO munisipalizacao-w-  
MUNICÍPIO munisipiu  
MUNIÇÃO munisa-w-

MUSEU muzew  
MUÇULMANA musuwma-na  
MUÇULMANOS musuwma-nus  
MÁ ma  
MÁQUINA makina  
MÁRCIA maXsia  
MÁRCIO maXsiu  
MÁRIO mariu  
MÁRMORE maRmori  
MÁXIMO maSimu  
MÃE ma-j-  
MÃO ma-w-  
MÉDIA meDzia  
MÉDICO meDziku  
MÉDICOS meDzikus  
MÉDIO meDziu  
MÉDIOS meDzius  
MÉRITO meRitu  
MÉXICO meSiku  
MÊS mejs  
MÍNIMO mi-nimu  
MÍSSEIS misejs  
MÓVEIS moVejs  
MÚMIA mu-mia  
MÚSCULOS muskulus  
MÚSICA muzika  
MÚSICAS muzikas  
NANA  
NACIONAL nasionaw  
NADA nada  
NAMORADA namorada  
NAMORANDO namora-du  
NAMORO namoru  
NAS nas  
NATAL nataw  
NATALINO natali-nu  
NATAÇÃO natasa-w-  
NAUFRAGADO nawfragadu  
NAZISTAS nazistas  
NAÇÃO nasa-w-  
NAÇÕES naso-j-s  
NECESSIDADE nesesisadZi  
NECESSÁRIA nesesaria  
NECESSÁRIO nesesaris  
NECESSÁRIOS nesesaris  
NEGAM nega-w-  
NEGOCIAÇÕES negosiaso-j-s  
NEGRA negra  
NEGRAS negras  
NEGROS negrus  
NEGÓCIO negOsiu  
NEGÓCIOS negOsius  
NELE neli  
NEM ne-j-  
NENHUM ne-Ju-  
NENHUMA ne-Ju-ma  
NEOLIBERALISMO neoliberalizmu  
NESSA nEsa  
NESSE nesi  
NESSES nesis  
NESTA nEsta  
NESTAS nEstas  
NESTE nestSi  
NESTOR nestoX  
NETO netu  
NEUTRAS newtras  
NINGUÉM ni-ge-j-  
NO nu  
NOBREZA nobreza  
NOITE nojtSi  
NOITES nojtSis  
NOME no-mi  
NOMEADO nomeadu  
NOMES no-mis  
NONO no-nu  
NORMAIS noRmajs  
NORMALMENTE noRmawme-tSi  
NORTE noXtSi  
NORTE-AMERICANOS noXtSiamerika-nus  
NORUEGA noruega  
NOS nus  
NOSSA nosa  
NOSSAS nosas  
NOTA nota  
NOTAS notas  
NOTIFICAÇÕES notSifikaso-j-s  
NOVA nova  
NOVE novi  
NOVECENTAS novese-tas  
NOVECENTOS novese-tus  
NOVELA novela  
NOVEMBRO nove-bru  
NOVENTA nove-ta  
NOVIDADE novidadeZi  
NOVIDADES novidadeZis  
NOVO novu  
NOVOS novus  
NUM nu-  
NUMA nu-ma  
NUNCA nu-ka  
NUNO nu-nu

NÃO na-w-  
NÉVOA nEvoa  
NÓS nojs  
NÚMERO nu-meru  
NÚMEROS nu-merus  
O u  
OBJETIVO obZetSivu  
OBJETO obZetu  
OBRA obra  
OBRAS obras  
OBRIGAR obrigaX  
OBRIGATORIEDADE obrigatorieda  
dzi  
OBSERVADORES observevadoris  
OBSERVADOS observevados  
OBSERVAR observevaX  
OBSERVÁVAMOS observevavamus  
OBSTÁCULO obstakulu  
OBTER obteX  
OCASIAO okazia-w-  
OCORRA okora  
OCORRE okori  
OCORRERAM okorera-w-  
OCORRERIA okoreria  
OCORREU okorew  
OCORRÊNCIA okore-sia  
OCUPA okupa  
OCUPARAM okupara-w-  
OFENSIVOS ofe-sivus  
OFERECE oferesi  
OFERECEM oferesej-  
OFERECIDOS oferesidus  
OFERTA ofeXta  
OFERTAS ofeXtas  
OFICIAL ofisiaw  
OFICINAS ofisi-nas  
OITAVA ojtava  
OITAVAS ojtavas  
OITENTA ojte-ta  
OITO ojtju  
OITOCENTOS ojtjose-tus  
OLGA olgwa  
OLHA ola  
OLHOS olus  
OLIMPIADAS olimpiadas  
OLIVEIRA olivejra  
OMISSO omisu  
OMISSÃO omisa-w-  
ONDE odZi  
ONTEM ote-j-  
ONZE ozi  
OPERAÇÃO operasa-w-  
OPERAÇÕES operaso-j-s  
OPINIÃO opinia-w-  
OPORTUNA opoXtuna  
OPORTUNIDADE opoXtunidadZi  
OPORTUNIDADES opoXtunidadZis  
OPTAR optaX  
OPÇÕES opso-j-s  
OPÔE opo-j-  
ORDEM orde-j-  
ORELHA oreLa  
ORGANIZACIONAIS orgorganizacionaj  
s  
ORGANIZADORES orgorganizadoris  
ORGANIZAR orgorganizaX  
ORGANIZAÇÃO orgorganizacao-w-  
ORIENTAÇÃO orientasa-w-  
ORIGEM oriZe-j-  
OS us  
OSCAR oskaX  
OSSOS osus  
OTAN ota-  
OTIMISMO otisimizmu  
OTIMISTA otisimista  
OU ow  
OUTONO-INVERNO owto-nui-veRnu  
OUTRA owtra  
OUTRAS owtras  
OUTRO owtru  
OUTROS owtrus  
OUTUBRO owtubru  
OUVIU owvi  
OVIU owvuw  
OVOS ovus  
PACIENTE pasie-tSi  
PACOTE pakotSi  
PACÍFICO pasifiku  
PADRE padri  
PADRÃO padra-w-  
PADRÕES padro-j-s  
PAGAM paga-w-  
PAGAMENTO pagame-tu  
PAGANDO paga-du  
PAGARÁ pagara  
PAGO pagu  
PAGOS pagus  
PAGOU pagow  
PALAVRA palavra  
PALAVRAS palavras  
PALCO pawku

PALESTINA palestSi~na  
 PALIATIVOS paliatiSivus  
 PALMEIRAS paWmejrás  
 PANELA panEla  
 PANO pa~nu  
 PAPEL papEw  
 PAPOU papow  
 PAR paX  
 PARA para  
 PARABÓLICAS parabÓlikas  
 PARADAS paradas  
 PARANÁ parana  
 PARAIBA paraiba  
 PARCEIROS paXsejrús  
 PARECE paresi  
 PARECERES pareseris  
 PARECIA paresia  
 PARIS paris  
 PARLAMENTO paRlame~tu  
 PARQUE paXke  
 PARREIRA paRejra  
 PARTE paXtSi  
 PARTES paXtSis  
 PARTICIPAM paXtSisipa~w  
 PARTICIPANTE paXtSisipa~tSi  
 PARTICIPANTES paXtSisipa~tSis  
 PARTICIPAR paXtSisipaX  
 PARTICIPOU paXtSisipow  
 PARTICULARES paXtSikularis  
 PARTICULARMENTE paXtSikulaR me~tSi  
 PARTIDA paXtSida  
 PARTIDO paXtSidu  
 PARTIDOS paXtSidus  
 PARTIDÁRIAS paXtSidarias  
 PARTIR paXtSiX  
 PARTIU paXtSiw  
 PARTOS paXtus  
 PASSA pasa  
 PASSADA pasada  
 PASSADAS pasadas  
 PASSADO pasadu  
 PASSAGEIROS pasaZejrús  
 PASSAM pasa~w  
 PASSAR pasaX  
 PASSARELA pasarEla  
 PASSARELAS pasarelas  
 PASSATEMPO pasate~pu  
 PASSAVA pasava  
 PASSEIO paseju  
 PASSES pasis  
 PASSOU pasow  
 PATOS patus  
 PATRIMÔNIO patrimo~niu  
 PATROCÍNIO patrosi~niu  
 PAULA pawla  
 PAULINO pawli~nu  
 PAULISTA pawlista  
 PAULISTANA pawlista~na  
 PAULISTANOS pawlista~nus  
 PAULISTAS pawlistas  
 PAULO pawlu  
 PAVIMENTO pavime~tu  
 PAVIMENTOS pavime~tus  
 PAZ pajs  
 PAÍS pais  
 PAÍSES paizis  
 PEDE pedzi  
 PEDIA pedzia  
 PEDIU pedziw  
 PEDRA pedra  
 PEGA pega  
 PEGAR pegaX  
 PEGOU pegow  
 PEIXE pejSi  
 PELA pEla  
 PELAS pelas  
 PELE peli  
 PELO pelu  
 PELOS pelus  
 PELÉ pelE  
 PENA pe~na  
 PENAS penajs  
 PENETRAÇÃO penetrasa~w  
 PENSA pe~sa  
 PENSADOR pe~sadoX  
 PENSAM pe~sa~w  
 PENSAMENTO pe~same~tu  
 PENSANDO pe~sa~du  
 PENSAR pe~saX  
 PENSASSE pe~sasi  
 PEPINOS pepi~nus  
 PEQUENAS peke~nas  
 PEQUENO peke~nu  
 PERCENTUAIS peXse~tuajs  
 PERCENTUAL peXse~tuaw  
 PERCO peXku  
 PERCORREM peXkoRe~j~  
 PERCUSSIONISTA peXkusionista  
 PERDEM peRde~j~  
 PERDEMOS peRde~mus  
 PERDENDO peRde~du  
 PERDER peRdeX  
 PERDERAM peRdera~w  
 PERDEU peRdew  
 PERDIDAS peRdzidas  
 PERDOE peRdoi  
 PERDUREM peRdure~j~  
 PEREIRA perejra  
 PERFEIÇÃO peXfejasa~w  
 PERFORMANCES peXfoRma~sis  
 PERFUMADAS peXfumadas  
 PERGUNTA peRgu~ta  
 PERGUNTAS peRgu~tas  
 PERIGO perigu  
 PERIGOSO perigozu  
 PERMANECER peRmaneseX  
 PERMANECERÃO peRmanesera~w  
 PERMITE peRmitSi  
 PERMITIA peRmitSiá  
 PERNA peRna  
 PERPLEXOS peXpleSus  
 PERSEGUIE peXsege  
 PERSEGUIDOS peXsegjdus  
 PERSONAGEM peXsonaZe~j~  
 PERSPECTIVA peXspektSiá  
 PERSUASIVO peXsuazivu  
 PERTINENTE peXtSine~tSi  
 PERTO peXtu  
 PERÍODO periodu  
 PESADAMENTE pezadame~tSi  
 PESO pezu  
 PESQUEIRO peskejru  
 PESQUISA peskiza  
 PESQUISADORES peskizadoris  
 PESQUISAS peskizas  
 PESSOA pesoa  
 PESSOAS pesoajs  
 PESSOAL pesoaw  
 PESSOAS pesoas  
 PESTANA pesta~na  
 PETISTA petiStá  
 PIA pia  
 PILOTO pilotu  
 PINHEIRO pi~Jejru  
 PINTORES pi~toris  
 PIOR piOX  
 PISO pizu  
 PISTA pista  
 PLANEJAMENTO planeZame~tu  
 PLANO pla~nu  
 PLANTADORES pla~tadoris  
 PLENA ple~na  
 PLENÁRIO plenariu  
 PLÁSTICO plástiSiku  
 POBRE pobri  
 PODE podzi  
 PODE-SE podZisi  
 PODEM pode~j~  
 PODER podeX  
 PODERIA poderia  
 PODERÁ podera  
 PODERÃO podera~w  
 PODIA podzia  
 POETAS poetas  
 POIS pois  
 POLICIAIS polisiajs  
 POLICIAL polisiaw  
 POLUIÇÃO poluijsa~w  
 POLÍCIA polisia  
 POLÍTICA poliSiá  
 POLÍTICAS poliSiás  
 POLÍTICO poliSiáku  
 POLÍTICOS poliSiáskus  
 PONTARIA po~taria  
 PONTO po~tu  
 PONTOS po~tus  
 POPULAR populaX  
 POPULAÇÃO populasa~w  
 POPULISMO populizmu  
 POR poX  
 PORQUE poXke  
 PORTA poXta  
 PORTADOR poXtadoX  
 PORTANTO poXta~tu  
 PORTAS poXtas  
 PORTE poXtSi  
 PORTO poXtu  
 PORTUGAL poXtugaw  
 PORTUGUESA poXtugeza  
 POSICIONAMENTO pozisioname~tu  
 POSITIVA pozitiá  
 POSIÇÃO pozisa~w  
 POSSA posa  
 POSSE posi  
 POSSIBILIDADE posibilidázdi  
 POSSIBILITAR possibilítaX  
 POSSO posu  
 POSSUEM posue~j~  
 POSSUI posui  
 POSSÍVEL posivew  
 POSTERIORMENTE posteriorme~tSi  
 POTENCIAL pote~siaw  
 POUCAS powkas  
 POUCO powku  
 POUCOS powkus  
 POUPADORAS powpadoras  
 POUAPANÇA powpa~sa  
 POVO powu  
 POXA poSa  
 PRAIA prajá  
 PRAIAS praias  
 PRANCHA pra~Sa  
 PRATELEIRAS pratelejras  
 PRATICADOS pratSiákus  
 PRATICAMENTE pratSiáame~tSi  
 PRATOS pratus  
 PRAZO prazu  
 PRAZOS prazus  
 PRECEDENTES pre~sedetSis  
 PRECIPITOU presipitow  
 PRECISAR presizaX  
 PRECISARIA presizaria  
 PRECISO presizu  
 PRECOCE prekosi  
 PREDILETO predziletu  
 PREFEITO prefjetu  
 PREFEITURA prefjetura  
 PREFERIA preferia  
 PREFERÊNCIA prefere~sia  
 PREFIRA prefira  
 PREJUDICADA prezudzikada  
 PREMIE premie  
 PREOCUPA preokupa  
 PREOCUPADO preokupadu  
 PREOCUPAM preokupa~w  
 PREOCUPAR preokupaX  
 PREOCUPAÇÃO preokupasa~w  
 PREPARAM prepara~w  
 PREPARANDO prepara~du  
 PRESAS prezás  
 PRESENTE preze~tSi  
 PRESENTES preze~tSis  
 PRESERVA prezeRva  
 PRESERVATIVOS prezeRvatSivus  
 PRESIDENCIAL prezide~siaw  
 PRESIDENTE prezide~tSi  
 PRESIDÊNCIA prezide~sia  
 PRESO prezu  
 PRESOS prezus  
 PRESSA presa  
 PRESSAS presas  
 PRESTÍGIO prestiZiu  
 PRETENDE prete~dzi  
 PRETENDO prete~du  
 PREVIDÊNCIA previde~sia  
 PREVISTA prevista  
 PREVISTO previstu  
 PREVISÃO previza~w  
 PREVÉ preve  
 PREVÊEM prevee~j~  
 PREÇO presu  
 PREÇOS presus  
 PRIMEIRA primejra  
 PRIMEIRAS primejras  
 PRIMEIRO primejru  
 PRIMEIROS primejrus  
 PRIMOS pri~mus  
 PRINCIPAIS pri~sipajs  
 PRINCIPAL pri~sipaw  
 PRINCIPALMENTE pri~sipawme~tSi  
 PRINCÍPIO pri~sipiú  
 PRIVILEGIOS privileZius  
 PRO pru  
 PROBLEMA proble~ma  
 PROBLEMAS proble~mas  
 PROCESSAM prosesa~w  
 PROCESSO prosesu  
 PROCESSOS prosesus  
 PROCON proko~  
 PROCURA prokura  
 PROCURAM prokura~w  
 PROCURANDO prokura~du  
 PROCURO prokuru  
 PROCUROU prokurow  
 PRODUTORES produtoris  
 PRODUTOS produtus  
 PRODUZIR produziX  
 PRODUÇÃO produ~sa~w  
 PROFESSORES profesoris  
 PROFISSIONAIS profisionajs  
 PROFISSIONAL profisionaw  
 PROFUNDAMENTE profu~dame~tSi  
 PROGRAMA progra~ma  
 PROGRAMAS progra~mas  
 PROGRAMAÇÃO programasa~w  
 PROGRESSISTA progresista  
 PROGRESSIVAMENTE progresivame~tSi  
 PROIBIDO projbidu  
 PROJETO proZetu  
 PROJETOS proZetus  
 PROMETER prometeX

PROMETO prometu  
 PROMISSORES promisoris  
 PROMOVE promovi  
 PRONTO pro-tu  
 PROPORCIONADOS propoXsionadus  
 PROPOSTA proposta  
 PROPOSTAS propostas  
 PROPUS propus  
 PROTAGONIZAR protagonizaX  
 PROTESTAR protestaX  
 PROTÓTIPO protOtSipu  
 PROVA prova  
 PROVAS provas  
 PROVOCAR provokaX  
 PROVÁVEL provavew  
 PROÍBA-Oproibau  
 PROÍBE proibi  
 PRÁTICA pratisika  
 PRÁTICAS pratisikas  
 PRÊMIOS pre-mius  
 PRÓ pro  
 PRÓPRIA prOpria  
 PRÓPRIAS prOprias  
 PRÓPRIO prOpriu  
 PRÓXIMA prOSima  
 PRÓXIMOS prOSimus  
 PSICOLÓGICAMENTE psikoloZikame-tSi  
 PSICOLÓGICA psikoloZika  
 PSICOLÓGICO psikoloZiku  
 PSIQUIATRA psikiatra  
 PUBLICADAS publikadas  
 PUBLICADO publikadu  
 PUBLICAM publika-w  
 PUBLICAÇÃO publikasa-w  
 PUDE pudzi  
 PUDER pudeX  
 PUDERAM puderaw-  
 PUSERAM puzeraw-  
 PÁDUA padua  
 PÁGINA pazina  
 PÁGINAS pazinas  
 PÂNICO paniku  
 PAES paj-s  
 PÉ pE  
 PÉLVICA pEwvika  
 PÉSSIMAS pEsimas  
 PÊNALTÍ pe-nawtSi  
 PÊNALTIS pe-nawtSis  
 PÓ pO  
 PÓLO pOlu  
 PÓS-MODERNA pOjmodeRna  
 PÓDE podzi  
 POE po-j-  
 PÚBLICA publika  
 PÚBLICAS publikas  
 PÚBLICO publiku  
 PÚRPURA pupura  
 QUADRADO kwadradu  
 QUADRANGULAR kwadragulaX  
 QUADRO kwadru  
 QUADRO kwadrus  
 QUADRÍCEPS kwadriseps  
 QUAI kwajs  
 QUAL kwaw  
 QUALIDADE kwalidadzi  
 QUALIDADES kwalidadzis  
 QUALIDADES kwalidades  
 QUALIFICADO kwalifikadu  
 QUALQUER kwawkeX  
 QUANDO kwa-du  
 QUANTO kwa-tu  
 QUARENTA kware-ta  
 QUARTEIROS kwaXtejro-j-s  
 QUARTO kwaXtu  
 QUARTOS kwaXtus  
 QUASE kwazi  
 QUATORZE kwatoRzi  
 QUATRO kwatru  
 QUATROCENTOS kwatrose-tus  
 QUE ke  
 QUEBRADA kebrada  
 QUEDA keda  
 QUEM ke-j-  
 QUENTE ke-tSi  
 QUER keX  
 QUEREM kere-j-  
 QUERER kereX  
 QUERIA keria  
 QUERIDO keridu  
 QUERO keru  
 QUESTIONAM keStSio-na-w-  
 QUESTÃO kesta-w-  
 QUILOMETROS kilo-metrus  
 QUINHENTAS ki-Je-tas  
 QUINTA-FEIRA ki-tafejra  
 QUINTO ki-tu  
 QUINZE ki-zi  
 QUISER kizeX  
 QUITAR kitaX  
 QUORUM kworu-

RAINHA Rai-Ja  
 RANGEL Ra-ZEW  
 RARAS Raras  
 RARO Raru  
 RAROS Rarus  
 RATOS Ratus  
 RAZÃO Raza-w-  
 RAZÕES Razo-j-s  
 RAIS Reajs  
 REAL Reaw  
 REALIDADE Realidadzi  
 REALIZADA Realizada  
 REALIZADOS Realizados  
 REALIZAR RealizaX  
 REALIZARAM Realizaraw-  
 REAPARECENDO Reaparese-du  
 REBAIXAMENTO RebaJSame-tu  
 REBELDES RebedwZis  
 RECEBER RebebeX  
 RECEBEU Rebebew  
 RECEIO Reseju  
 RECEITA Reseja  
 RECEPÇÃO Resepsa-w-  
 RECLAMA Reklama  
 RECLAMAM Reklama-w-  
 RECLAMANDO Reklama-du  
 RECLAMAR ReklamaX  
 RECLAMARAM Reklamara-w-  
 RECLAMAÇÕES Reklamaso-j-s  
 REGOLHIMENTO RekoLime-tu  
 RECOMENDÁVEIS Rekomedavejs  
 RECONSTRUIR Reko-struiX  
 RECORDE RekoRdZi  
 RECORDES RekoRdZis  
 RECORRER RekoReX  
 RECUPERADO Rekuperadu  
 RECUPERAÇÃO Rekuperasa-w-  
 RECUPEROU Rekuperow  
 RECURSOS RekuXsus  
 RECUSA Rekuza  
 RECUSOU Rekuzow  
 RECUSOU-SE Rekuzowski  
 RECÉM Rese-j-  
 REDES Redzis  
 REDUZIDO Reduzidu  
 REDUÇÃO Redusa-w-  
 REESTILIZADO Reestilizadu  
 REFERÊNCIA Refere-sia  
 REFERÊNCIAS Refere-sias  
 REFLETINDO RefletSi-du  
 REFLEXÃO RefleSa-w-  
 REFORMA Reforma  
 REFORMADA Reformada  
 REFORMAS Reformas  
 REFORMULAR ReformulaX  
 REFORÇAR ReforXsaX  
 REFUGIADOS Refuziadus  
 REFÚGIOS Refuzius  
 REGINA ReZina  
 REGIONAL ReZionaw  
 REGISTRA ReZistra  
 REGISTRADA ReZistrada  
 REGISTRADO ReZistradu  
 REGISTRO ReZistro  
 REGISTROU ReZistrow  
 REGIÃO ReZia-w-  
 REGRA Regra  
 REGRAS Regras  
 REIMPLANTADO Rei-pla-tadu  
 REIMPLANTE Rei-pla-tSi  
 REINO Rejnu  
 REIVINDICAÇÕES Rejvi-dzikaso-j-s  
 RELACIONADOS Relasionadus  
 RELACIONAL Relasionaw  
 RELACIONAMENTO Relasioname-tu  
 RELATIVIZA RelatSiviza  
 RELATOR RelatoX  
 RELAXA RelaSa  
 RELAXAR RelaSaX  
 RELAÇÃO Relasa-w-  
 RELAÇÕES Relaso-j-s  
 REMUNERA Remunera  
 REMUNERADOS Remuneradus  
 REMUNERAÇÃO Remunerasa-w-  
 REMÉDIOS RemedZius  
 RENDE Re-dzi  
 RENDEM Re-de-j-  
 RENDER Re-deX  
 RENDERIA Re-deria  
 RENOVAR Renovax  
 REPRESA Repreza  
 REPRESENTADO Repreze-tadu  
 REPRESENTANTE Repreze-ta-tSi  
 REPRESENTAR Repreze-taX  
 REPRESENTATIVIDADE Repreze-tatSividadz  
 REPRESENTAVA Repreze-tava  
 REPÓRTER RepOXteX  
 REPÚBLICA Republika  
 RESERVADAMENTE RezeRvadame-tSi  
 RESERVAS RezeRvas

RESGATE RezgataSi  
 RESIDÊNCIA ReZide-sia  
 RESISTIRÃO ReZistSira-w-  
 RESISTIU ReZistSiw  
 RESOLUÇÃO ReZolusa-w-  
 RESOLVIDAS ReZowvidas  
 RESOLVIDOS ReZowvidus  
 RESPEITO Reseju  
 RESPONDE Respo-dzi  
 RESPONDER Respo-deX  
 RESPONDEU Respo-dew  
 RESPONSABILIDADE Respo-sabilid  
 adzi  
 RESPONSÁVEIS Respo-savejs  
 RESPONSÁVEL Respo-savew  
 RESPOSTA Resposta  
 RESTA Resta  
 RESTANDO Resta-du  
 RESTAURADA Restawrada  
 RESTAURANTES Restawra-tSis  
 RESTO Restu  
 RESTRINGE Restri-Zi  
 RESULTADO Rezuwtadu  
 RESULTADOS Rezuwtadus  
 RESUMINDO Rezumidu  
 RETIRADA RetSirada  
 RETIRADO RetSiradu  
 RETIRADOS RetSiradus  
 RETIRARAM RetSiraraw-  
 RETORNADO RetoRnadu  
 RETRIBUIR RetribuiX  
 RETRÓGRADAS RetroGradas  
 RETÊM Rete-j-  
 REUNIÃO Rewnia-w-  
 REVELA ReveLa  
 REVELADO Reveladu  
 REVELOU Revelow  
 REVERTA ReveXta  
 REVISADO Revizadu  
 REVISTA Revista  
 REVISTOS Revistus  
 REVISÃO Reviza-w-  
 REVOLUÇÃO Revolusa-w-  
 REZE Rezi  
 REÚNE Reu-ni  
 RIBEIRO Ribejru  
 RICA Rika  
 RIGOROSAS RigorOzas  
 RIO Riu  
 RISCA Riska  
 RISCO Risku  
 RISCOS Riskus  
 RIVALDO Rivawdu  
 ROBERTO RobeXtu  
 RODADA Rodada  
 RODOPIOU Rodopiow  
 RODRIGUES Rodrigis  
 ROLOS Rolus  
 ROMA Ro-ma  
 ROMANCES Roma-sis  
 ROMÁRIO Romariu  
 ROMÊNIA Rome-nia  
 RONALDO Ronawdu  
 ROSA ROza  
 ROSAS ROzas  
 ROSE Rozi  
 ROSTO Rostu  
 ROTEIRO Rotejru  
 ROUBALHEIRA RowbaLejra  
 ROUBOU Rowbow  
 RUA Rua  
 RUANDESA Rua-deza  
 RUAS Ruas  
 RURAIS Rurajs  
 RUSSA Rusia  
 RUSSOS Rusus  
 RUIDO Ruidu  
 RÁPIDO Rapidu  
 RÚSSIA Rusia  
 SABE sabi  
 SABEM sabe-j-  
 SABER sabeX  
 SACO saku  
 SACOLAS sakolas  
 SAEM sajm  
 SAGITÁRIO saZitariu  
 SAGROU-SE sagrowsi  
 SALSaj  
 SAIAM saia-w-  
 SAIR saix  
 SAIRIA sajria  
 SAIRÁ sajra  
 SAIU saiw  
 SALA sala  
 SALADA salada  
 SALARIAL salariaw  
 SALTA sawta  
 SALVADOR sawwadoX  
 SALVANDO sawwa-du  
 SALÁRIO salariu  
 SALÁRIOS salarius

SAMBA sa-ba  
SAMBÓDROMO sa-bO dromu  
SANCIONAR sa-sionaX  
SANDRA sa-dra  
SANGUE sa-ge  
SANITARIA sanitaria  
SANTIDADE sa-tSida dZi  
SANTISTAS sa-tSistas  
SANTO sa-tu  
SANTOS sa-tus  
SAQUE sa-ke  
SATISFEITOS sa-tSisfejtus  
SAÍDO sa-idu  
SAÚDE sa-udZi  
SE si  
SECA se-ka  
SECRETARIA se-kretaria  
SECRETO se-kretu  
SECRETÁRIO se-kretariu  
SECREÇÃO se-kresa-w-  
SECUNDARISTAS se-ku-daristas  
SEDATIVO se-da-ti-vu  
SEGREDO se-gre-du  
SEGREDOS se-gre-dus  
SEGUIDA se-gj-da  
SEGUIDAS se-gj-das  
SEGUIDO se-gj-du  
SEGUNDA se-gu-da  
SEGUNDO se-gu-du  
SEGURADO se-gu-ra-du  
SEGURANÇA se-gu-ra-sa  
SEGURANÇAS se-gu-ra-sas  
SEGURO se-gu-ru  
SEGUROS se-gu-ru-s  
SEI se-j  
SEIS se-js  
SEISCENTOS se-je-se-tus  
SEJA se-za  
SEJAM se-za-w-  
SELEÇÃO se-le-sa-w-  
SELEÇÕES se-le-so-j-s  
SEM se-j-  
SEMANA se-ma-na  
SEMANAS se-ma-nas  
SEMELHANTE se-me-la-tSi  
SEMELHANTES se-me-la-tSis  
SEMIFINAIS se-mi-fina-js  
SEMIFINAL se-mi-fina-w  
SEMINÁRIO se-mi-nariu  
SEMPRE se-pri  
SENADOR se-na-doX  
SENADORES se-na-do-ri-s  
SENDO se-du  
SENSAÇÃO se-sa-sa-w-  
SENTE se-tSi  
SENTEM se-te-j-  
SENTIA se-tSi-a  
SENTIDO se-tSi-du  
SENTIU se-tSi-w  
SEPARADAMENTE se-pa-ra-da-me-tSi  
SEPARADO se-pa-ra-du  
SEPARATISTA se-pa-ra-tSi-sa  
SEQUEIROS se-ke-jrus  
SEQÜELA se-ke-wE-la  
SEQÜESTRADO se-ke-we-stradu  
SEQÜESTRO se-ke-we-stru  
SER se-X  
SEREM se-re-j-  
SERIA se-ria  
SERIADOS se-ria-dus  
SERIAM se-ria-w-  
SERRA se-ra  
SERVE se-rvi  
SERVIRIA se-rvi-ria  
SERVIU se-rvi-w  
SERVIÇO se-rvi-su  
SERÁ se-ra  
SERÃO se-ra-w-  
SESSENTA se-se-ta  
SESSÃO se-sa-w-  
SETE se-tSi  
SETECENTOS se-te-se-tus  
SETEMBRO se-te-bru  
SETENTA se-te-ta  
SETOR se-toX  
SETORES se-to-ri-s  
SEU se-w  
SEUS se-ws  
SEXO se-su  
SEXTA se-sa  
SEXTO se-sa  
SEXUAL se-su-a-w  
SEXUALIDADE se-su-a-li-da-dZi  
SEÇÃO se-sa-w-  
SIDO si-du  
SIGNIFICA signi-fi-ka  
SIGNIFICAVA signi-fi-ka-va  
SILVA si-w-va  
SILÊNCIO si-le-siu  
SIM si-  
SIMAS si-mas

SIMULTANEAMENTE si-mu-wt-a-ne-a-me-tSi  
SIMÃO si-ma-w-  
SINAIS si-na-js  
SINAL si-na-w  
SINCERA si-se-ra  
SINDICATO si-dZi-ka-tu  
SINTA si-ta  
SINTOMA si-to-ma  
SIQUEIRA si-ke-jra  
SISTEMA si-ste-ma  
SITUAÇÃO si-tua-sa-w-  
SITUAÇÕES si-tua-so-j-s  
SOA so-a  
SOARES so-a-ri-s  
SOB so-b  
SOBE so-bi  
SOBRE so-bri  
SOBREVIVENTES so-bre-vi-ve-tSi-s  
SOBREVIVER so-bre-vi-ve-X  
SOBREVIVÊNCIA so-bre-vi-ve-si-a  
SOCIAIS so-si-a-js  
SOCIAL so-si-a-w  
SOCIALISMO so-si-a-li-zmu  
SOCIALISTA so-si-a-li-sa  
SOCIEDADE so-si-e-da-dZi  
SOCORRO so-ko-ru  
SOFISTICADO so-fi-s-ti-ka-du  
SOFRERAM so-fr-e-ra-w-  
SOJA so-za  
SOL so-w  
SOLDADO so-w-da-du  
SOLDADOS so-w-da-dus  
SOLICITANDO so-li-si-ta-du  
SOLICITAR so-li-si-ta-X  
SOLIDARIEDADE so-li-da-ri-e-da-dZi  
SOLOS so-lus  
SOLTA so-w-ta  
SOLTEIRO so-w-te-jru  
SOLTO so-w-tu  
SOLUCIONAR so-lu-si-o-na-X  
SOLUÇÃO so-lu-sa-w-  
SOMA so-ma  
SOMAM so-ma-w-  
SOMARAM so-ma-ra-w-  
SOME so-mi  
SOMENTE so-me-tSi  
SOMOS so-mus  
SONHA so-ja  
SONO so-nu  
SORTE so-X-tSi  
SORTEIO so-X-te-ju  
SORVETE so-r-ve-tSi  
SOU so-w  
SOUBEMOS so-w-be-mus  
SOUZA so-w-za  
SOZINHA so-zi-ja  
SOZINHAS so-zi-ja-s  
SUA su-a  
SUAS su-as  
SUASSUNA su-a-su-na  
SUBIR su-bi-X  
SUBIU su-bi-w  
SUBPROCESSOS su-b-pro-se-sus  
SUBSTITUI su-b-si-tui  
SUBTIPO su-b-ti-pu  
SUCESSO su-se-su  
SUCURSAIS su-ku-X-sa-js  
SUCURSAL su-ku-X-sa-w  
SUFICIENTE su-fi-si-e-tSi  
SUFOCANDO su-fo-ka-du  
SUGERE su-ze-ri  
SUJA su-za  
SUJO su-za  
SUL su-w  
SUMIU su-mi-w  
SUMO su-mu  
SUPERA su-pe-ra  
SUPERGRILADA su-pe-r-gri-la-da  
SUPERMERCADOS su-pe-r-me-x-ka-dus  
SUPEROFERTA su-pe-ro-fe-X-ta  
SUPERFLUAS su-pe-r-flu-as  
SURGEM su-r-ze-j-  
SURGIR su-r-z-i-X  
SURPRESA su-X-pre-za  
SUSPEITA su-s-pe-ja  
SUSPENSA su-s-pe-sa  
SUSPENSE su-s-pe-si  
SUSPENSÃO su-s-pe-sa-w-  
SUSTOS su-s-tus  
SÚCIA su-si-a  
SÁBADO sa-ba-du  
SÃO sa-w-  
SÉCULO sE-ku-lu  
SÉRVIOS sE-r-vi-us  
SÍMBOLO si-bolu  
SÓ so-o  
TABELA ta-bE-la  
TABORDA ta-bo-r-da  
TAIS ta-js  
TALENTOS ta-le-tus  
TALVEZ ta-w-ve-je-s

TAMBÉM ta-be-j-  
TANQUE ta-ke  
TANTA ta-ta  
TANTAS ta-tas  
TANTO ta-tu  
TARDE ta-r-dZi  
TARJAS ta-r-za-s  
TAXA ta-sa  
TAXAS ta-sa-s  
TAXIMÉTRICA ta-sim-E-tri-ka  
TEATRAL te-a-tra-w  
TEATRO te-a-tru  
TECNOLOGIA te-ko-no-lo-zi-a  
TECNOLOGIAS te-ko-no-lo-zi-as  
TELA te-la  
TELAS te-las  
TELEFONE te-le-fo-ni  
TELEJORNALIS te-le-jo-r-na-js  
TELEVISÃO te-le-vi-za-w-  
TELHAS te-las  
TELE te-le  
TEM te-j-  
TEM-SE te-j-si  
TEMA te-ma  
TEMAS te-mas  
TEME te-mi  
TEMOR te-mo-X  
TEMOS te-mus  
TEMPERATURA te-pe-ra-tu-ra  
TEMPLO te-piu  
TEMPLOS te-plus  
TEMPO te-pu  
TEMPORADA te-po-ra-da  
TEMÁTICA te-ma-ti-ka  
TEMÁTICO te-ma-ti-ku  
TENDEM te-de-j-  
TENDER te-de-X  
TENDOTE te-du  
TENDÊNCIA te-de-si-a  
TENHA te-ja  
TENHO te-ju  
TENSÃO te-sa-w-  
TENTA te-ta  
TENTA-SE te-ta-si  
TENTANDO te-ta-du  
TENTARAM te-ta-ra-w-  
TENTOU te-to-w  
TEOLOGIA te-o-lo-zi-a  
TER te-X  
TERCEIRO te-X-se-ju  
TEREZHINA te-re-zi-ja  
TERIA te-ria  
TERIAM te-ria-w-  
TERMINAR te-r-mi-na-X  
TERMO te-r-mu  
TERMOMETROS te-r-mo-me-trus  
TERNO te-r-nu  
TERRA te-ra  
TERRENO te-re-nu  
TERRITÓRIO te-ri-to-ri-u  
TERÁ te-ra  
TERÇA-FEIRA te-X-sa-fe-ja  
TESE te-ze  
TETO te-tu  
TETRA te-ta  
TEVE te-ve  
TEXTO te-stu  
TIME ti-si-mi  
TIMES ti-si-mis  
TINHA ti-si-ja  
TINHAM ti-si-ja-w-  
TIPO ti-pu  
TIPOS ti-pus  
TIRADA ti-ri-da  
TIRAR ti-ri-a-X  
TIRO ti-si-ru  
TITULAR ti-si-tu-la-X  
TITÁ ti-si-ta-  
TIVER ti-si-ve-X  
TIVERAM ti-si-ve-ra-w-  
TOCADA to-ka-da  
TOCAR to-ka-X  
TOCOU to-ko-w  
TODA to-da  
TODAS to-das  
TODO to-du  
TODOS to-dus  
TOLEDO to-le-du  
TOLERÂNCIA to-le-ra-si-a  
TOLICE to-li-si  
TOMADA to-ma-da  
TOMAR to-ma-ra  
TORCEDORES to-X-se-do-ri-s  
TORCENDO to-X-se-du  
TORCIDA to-X-si-da  
TORNAM to-r-na-w-  
TORNO to-r-nu  
TORNOU-SE to-r-no-w-si  
TOTAL to-ta-w  
TOTALMENTE to-ta-w-me-tSi  
TRABALHA tra-ba-la

TRABALHADORES trabaLadoris  
 TRABALHAM trabaLa-w~  
 TRABALHAMOS trabaLa-mus  
 TRABALHANDO trabaLa-du  
 TRABALHAR trabaLaX  
 TRABALHARAM trabaLa-r~w~  
 TRABALHISTA trabaLista  
 TRABALHO trabaLu  
 TRABALHOS trabaLus  
 TRADICIONAL tradZisionaw  
 TRADICIONALMENTE tradZisionawme~tSi  
 TRAFICANTE trafika~tSi  
 TRAMA tra~ma  
 TRAMITAM tramita-w~  
 TRANQUILIDADE tra~kwilidaZi  
 TRANSARAM tra~zara-w~  
 TRANSAÇÃO tra~zasa-w~  
 TRANSFERIDOS tra~sferidus  
 TRANSFERÊNCIA tra~sferesia  
 TRANSIÇÃO tra~zisa-w~  
 TRANSPARENTE tra~spare~tSi  
 TRANSPIRA tra~spira  
 TRARIA traria  
 TRASEIRA trazejra  
 TRATA-SE tratasi  
 TRATANDO trata~du  
 TRATE tra~tSi  
 TRAVES travis  
 TRAVESSEIRO travesejru  
 TRAZ trajs  
 TRAZER trazeX  
 TREINADOR trejnadoX  
 TREINARAM trejnara-w~  
 TREINO trejnu  
 TREINOS trejnus  
 TREM tre~j~  
 TREPIDAR trepidaX  
 TREZE trezi  
 TREZENTAS treze~tas  
 TREZENTOS treze~tus  
 TRIBOS tribus  
 TRIBUNE tribu~ni  
 TRIBUTÁRIO tributariu  
 TRINTA tri~ta  
 TRIOS trius  
 TRISTÃO trista~w~  
 TROCAR trokaX  
 TROPA tropa  
 TRUQUE truke  
 TRÁFEGO trafegu  
 TRÁFICO trafiku  
 TRÁS trajs  
 TRÁNSITO tra~zitu  
 TRÊS trejs  
 TUCANO tuka~nu  
 TUCANOS tuka~nus  
 TUDO tudu  
 TUMA tu~ma  
 TUMULTUADA tumuwtuada  
 TURISMO turizmu  
 TURISTAS turistas  
 TURNO tu~Rnu  
 TURNOS tu~Rnus  
 TURÍSTICOS turistSikus  
 TÁ ta  
 TÁTICA ta~tSika  
 TÁO ta~w~  
 TÉCNICA tEknika  
 TÉCNICAS tEknikas  
 TÉCNICO tEkniku  
 TÉCNICOS tEknikus  
 TÊM te~j~  
 TÍMIDOS tSi~midus  
 TÍNHAMOS tSi~Jamus  
 TÍPICA tSipika  
 TÍPICAS tSipikas  
 TÍTULO tSitulu  
 TÍTULOS tSitulus  
 TÓQUIO tOkiu  
 UEFA uefa  
 UM u~  
 UMA u~ma  
 UMAS u~mas  
 UMBIGO u~bigu  
 UNIDADE unidaZi  
 UNIDADES unidaZis  
 UNIDAS unidas  
 UNIDO unidu  
 UNIDOS unidus

UNILATERALMENTE unilaterawme~tSi  
 UNIVERSIDADE univeXsidadZi  
 UNIVERSIDADES univeXsidadZis  
 UNIVERSO univeXsu  
 URNAS urnas  
 USAR uzaX  
 USAVAM uzava~w~  
 USE uzi  
 USO uzu  
 UTILIZADO utSilizadu  
 UISQUE uiske  
 VACINAÇÃO vasinasa~w~  
 VAGA vaga  
 VAGO vagu  
 VAGÃO vaga~w~  
 VAIVAJ vaivaj  
 VAIAS vaias  
 VALE vali  
 VALENDO vale~du  
 VALERIA valeria  
 VALOR valoX  
 VALORES valoris  
 VALORIZADOS valorizadus  
 VALORIZAÇÃO valorizasa~w~  
 VAMOS va~mus  
 VANTAGEM va~taZe~j~  
 VAREJO vareZu  
 VARIA varia  
 VARIADO variadu  
 VARIAM varia~w~  
 VARIAÇÃO variasa~w~  
 VASILHAS vazilas  
 VEIO veju  
 VEJO veZu  
 VELA vEla  
 VELHA vELa  
 VELHAS vELas  
 VELHO vELu  
 VELOZ veLOjs  
 VEM ve~j~  
 VENAL venaw  
 VENCEDOR ve~sedoX  
 VENCER ve~seX  
 VENCESLAU ve~sezlaw  
 VENDA ve~da  
 VENDAS ve~das  
 VENDENDO ve~de~du  
 VENDER ve~deX  
 VENDIDOS ve~dZidus  
 VENDO ve~du  
 VENEZUELA venezueLa  
 VENHO ve~Ju  
 VENTO ve~tu  
 VER veX  
 VERBA veRba  
 VERDADE veRdadZi  
 VERMELHO veRmELu  
 VERSÃO veXsa~w~  
 VERÃO vera~w~  
 VESTIBULARES vestSibularis  
 VESTIR vestSiX  
 VETAVA vetava  
 VEXAMES veSa~mis  
 VEZ vejs  
 VEZES vezis  
 VEÍCULO veikulu  
 VEÍCULOS veikulus  
 VIVI vi  
 VIA via  
 VIAGEM viaZe~j~  
 VIAGENS viaZe~j~s  
 VIANA via~na  
 VIAS vias  
 VIBRAÇÕES vibraso~j~s  
 VIDA vida  
 VIDAS vidas  
 VIDEOCLIPES videoklipis  
 VIDEOTEIPES videotejpis  
 VIGILÂNCIA vizila~sia  
 VIGOR vigoX  
 VIGORAR vigoRaX  
 VIGOROSO vigoRoZu  
 VINGANÇA vi~ga~sa  
 VINGARÁ vi~gara  
 VINHA vi~Ja  
 VINTE vi~tSi  
 VIOLA viola  
 VIOLADO violadu  
 VIOLÃO viola~w~  
 VIOLÊNCIA viole~sia

VIR viX  
 VIRAM vira~w~  
 VIRARAM vira~ra~w~  
 VIROU virow  
 VIRTUDE viXtudZi  
 VISITA vizita  
 VISITADOS vizitadus  
 VISITA~LO vizitalu  
 VISTAS vistas  
 VISUAL vizuaw  
 VISÍVEIS vizivejs  
 VITÓRIA vitOria  
 VIU viu  
 VIVA viva  
 VIVE vivi  
 VIVE~SE vivisi  
 VIVER viveX  
 VIVERIA viveria  
 VIVO vivu  
 VIES viEjs  
 VOCABULÁRIO vokabulariu  
 VOCÊ vose  
 VOCÊS vosejs  
 VOLTA vOwta  
 VOLTADO vowtadu  
 VOLTAM vOwta~w~  
 VOLTAM~SE vOwta~w~si  
 VOLTAMOS vowta~mus  
 VOLTAR vowtaX  
 VOLTARAM vowntara~w~  
 VOLTO vOwtu  
 VOLTOU vowntow  
 VOLUME volu~mi  
 VOLUNTARIOS volu~tariOza  
 VOLUNTÁRIOS volu~tarius  
 VONTADE vo~taZi  
 VOTADA votada  
 VOTARAM votara~w~  
 VOTAÇÃO votasa~w~  
 VOTO votu  
 VOTOS votus  
 VOU vov  
 VOZ vOjs  
 VALIDOS validus  
 VÁRIAS varias  
 VÁRIOS varius  
 VÃO va~w~  
 VÉSPERAS vesperas  
 VÊ ve  
 VÊM ve~j~  
 VÍDEO videu  
 VÍRGULA viRgula  
 VÍRUS virus  
 VÍTIMA vitSima  
 XAVIER zavieX  
 ZERO zeru  
 ZICO ziku  
 ZONA zo~na  
 ZÉLIA zElia  
 Áa  
 ASajs  
 ÁGUA agwa  
 ÁLCOOL awkooW  
 ÁREA area  
 ÁREAS areas  
 ÁRVORES aRvoris  
 ÁUSTRIA awstria  
 ÊE  
 ÉPOCA Epoka  
 ÉTICA EtSika  
 ÊXITO eSitu  
 ÍNDIA i~dZia  
 ÍNDICE i~dZisi  
 ÍNDICES i~dZisis  
 ÍNDIO i~dZiu  
 ÍNDIOS i~dZius  
 ÔBVIAS Obvias  
 ÔPERA Opera  
 ÔTICA OtSika  
 ÔTIMA OtSima  
 ÔNIBUS o~nibus  
 ÔNUS o~nus  
 ÚLTIMA uwtSima  
 ÚLTIMO uwtSimu  
 ÚLTIMOS uwtSimus  
 ÚNICA u~nika  
 ÚNICO u~niku  
 ÚTEIS utejs  
 ÚTERO uteru

## Apêndice 4

### **Reconhecimento de Fala Contínua Independente de Locutor:**

Testes preliminares com uma base independente de locutor puderam ser realizados juntando-se a base de 1000 frases, utilizada nos testes dessa dissertação, com uma outra base composta de 1600 locuções [105], gravadas por 40 locutores distintos (20 homens e 20 mulheres) e baseadas em 200 frases foneticamente balanceadas [112]. As 1600 frases foram divididas entre treino e teste respeitando simultaneamente os critérios definidos abaixo:

- 1280 frases para treino, 320 para teste
- 32 locutores para treino (16 H, 16 M), 8 para teste (4 H e 4 M)
- As 200 frases são divididas em 10 grupos: 16 grupos para treino, 4 para teste.

Identificando-se cada locutor com 3 letras no formato “SDD”, onde “S” representa o sexo do locutor, podendo ser “m” para homem ou “f” para mulher, e “DD” representa o número seqüencial do locutor por sexo na base: “m01”, portanto significa o primeiro locutor masculino da base. Com esta definição, os locutores e grupos frásicos da base, mencionados acima, foram divididos da forma descrita pela Tabela 15. A base de 1000 frases, citada anteriormente, foi usada inteiramente para treino do sistema.

Tabela 15: Divisão entre locutores e grupos frásicos da base.

<i>Grupos Frásicos e Locutores de Teste</i>	
1 a 4	m01,m07,m14,m20,f06,f09,f12,f13
<i>Grupos Frásicos e Locutores de Treino</i>	
5 a 8	m02,m16,m18,m23,f02,f07,f15,f20
9 a 12	m05,m09,m17,m21,f03,f08,f17,f18
13 a 16	m03,m06,m11,m24,f04,f10,f19,f22
17 a 20	m04,m12,m13,m15,f01,f05,f11,f21

Com as bases montadas, foi possível observar taxas de acerto do sistema independente de locutor para dicionários de tamanho 3.8k e 12.5k palavras, bigramas e apenas trifones intrapalavras. O dicionário de 3.8K palavras foi montado com base no dicionário já existente de 3.5k, gerado através das 1000 frases, adicionando-se as novas palavras existentes na base de 200 frases. As *WRRs* obtidas em testes com estes tamanhos de dicionário e diferentes configurações do feixe de busca (normal e reduzido em 20%) podem ser vistas na Figura 48.

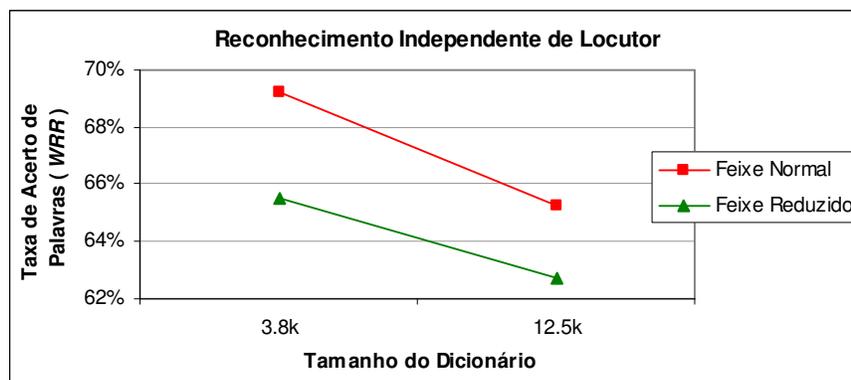


Figura 48: *WRR* observada em testes preliminares com independência de locutor.

## Apêndice 5

### *Lista de Publicações do Autor (reproduzidas em anexo):*

- TERUSZKIN, R., CONSORT, T. A., RESENDE JR., F. G. V., "Endpoint Detection Analysis for a Implementation of a Speech Recognition System Applied to Robot Control", *Proceedings of SAWCAS IEEE*, Rio de Janeiro, Brasil, Novembro, 2001.
- TERUSZKIN, R., RESENDE JR., F.G.V, VILLAS-BOAS, S.B., LIZARRALDE, F., "Biblioteca Orientada a Objetos para Reconhecimento de Voz e Aplicação em Controle de Robô", *Congresso Brasileiro de Automática (CBA)*, Natal, Brasil, Setembro, 2002.
- TERUSZKIN, R., RESENDE JR., F.G.V, "Sistema Híbrido HMM-DTW para Reconhecimento de Palavras Isoladas através de Trifones", *Semana da Eletrônica – Universidade Federal do Rio de Janeiro*, Rio de Janeiro - RJ, Brasil, Setembro, 2003.
- TERUSZKIN, R., RESENDE JR., F.G.V., "Sistema de Conversação Telefônica para Deficientes Auditivos: Módulo de Reconhecimento de Voz", *IV Congresso Ibero-Americano Sobre Tecnologias de Apoio a Portadores de Deficiência (IBERDISCAP)*, Vitória - ES, Brasil, Fevereiro, 2006.
- TERUSZKIN, R., RESENDE JR., F.G.V, "Phonetic Sequence to Graphemes Conversion Based on DTW and One-Stage Algorithms", *VII Encontro para o Processamento Computacional da Língua Portuguesa Escrita e Falada (PROPOR)*, Itatiaia - RJ, Brasil, Maio, 2006.