SÍNTESE DE VOZ BASEADA EM MODELOS OCULTOS DE MARKOV UTILIZANDO NOVA BASE DE DADOS PARA PORTUGUÊS BRASILEIRO

Dayana Sant'Anna Lole

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Fernando Gil Vianna Resende Junior

Rio de Janeiro
Janeiro de 2015

SÍNTESE DE VOZ BASEADA EM MODELOS OCULTOS DE MARKOV UTILIZANDO NOVA BASE DE DADOS PARA PORTUGUÊS BRASILEIRO

Dayana Sant'Anna Lole

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

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

_____
Profª. Mariane Rembold Petraglia, Ph. D.

_____
Prof. Abraham Alcaim, Ph. D.

RIO DE JANEIRO, RJ - BRASIL
JANEIRO DE 2015

Lole, Dayana Sant'Anna

Síntese de Voz baseada em Modelos Ocultos de Markov utilizando nova base de dados para Português Brasileiro/ Dayana Sant'Anna Lole. – Rio de Janeiro: UFRJ/COPPE, 2015.

XV, 100 p.: il.; 29,7 cm.

Orientador: Fernando Gil Vianna Resende Junior

Dissertação (mestrado) – UFRJ/ COPPE/ Programa de Engenharia Elétrica, 2015.

Referências Bibliográficas: p. 65-72.

1. Síntese de Voz. 2. Conversão Texto-Fala. 3. Processamento de Voz. I. Resende Junior, Fernando Gil Vianna. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

“Se alguém tem falta de sabedoria, peça a Deus, que a todos dá liberalmente.”

Tiago 1.5

A Deus e à minha família

AGRADECIMENTOS

A Deus, em primeiro lugar, pelo seu sustento nos momentos de maior dificuldade e pela sua benignidade em delinear cada um dos meus dias. Ele foi o meu melhor amigo nos momentos de dificuldade e minha maior orientação quando não sabia o que fazer.

Ao meu esposo, Marlon, que é um presente de Deus na minha vida e manifestou seu amor, amizade e companheirismo me ajudando a abrir mão do tempo necessário para o desenvolvimento deste trabalho e estando ao meu lado em cada passo desta jornada. Permanecer foi possível graças a sua mão segurando a minha.

Aos meus pais, pela compreensão da minha dificuldade, pelo respaldo em orações e pela companhia nas idas ao Fundão.

Aos meus avós, que me fizeram sentir amada e confortada durante todo o processo de estudo. Em especial, à minha avó-mãe, Maria de Lourdes, que esteve comigo até onde conseguiu e vibrava com cada nova conquista. Ela foi o combustível para a minha última volta.

Aos meus irmãos, que tornaram meus dias mais divertidos pelo seu carinho e cumplicidade. Particularmente, ao Daniel, que me atendia mesmo de madrugada para testes de interface.

Aos meus amigos, em especial à Nathália, Elaine e Patrícia, por toda a ajuda e carinho que, mesmo sem saber, cultivaram mais alegria no meu dia-a-dia.

Aos comandantes Rogério Lins, Pimenta e Alex Lopes, por todo seu apoio e compreensão.

Ao Luiz Vecchietti, que entrou no meu caminho nessa reta final e contribuiu não só para o meu crescimento profissional como pessoal.

Ao meu orientador Fernando Gil, pela atenção às minhas dificuldades e pela paciência em ouvir algumas ideias.

Aos professores Mariane e Abraham, que prontamente aceitaram participar da banca para a defesa desta dissertação.

A todos que realizaram os testes subjetivos e tanto contribuíram para a obtenção dos principais resultados deste trabalho.

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

SÍNTESE DE VOZ BASEADA EM MODELOS OCULTOS DE MARKOV UTILIZANDO NOVA BASE DE DADOS PARA PORTUGUÊS BRASILEIRO

Dayana Sant'Anna Lole

Janeiro/2015

Orientador: Fernando Gil Vianna Resende Junior

Programa: Engenharia Elétrica

Ao longo do tempo, os computadores têm angariado cada vez mais espaço no cotidiano contribuindo para uma crescente demanda por tecnologias de processamento de voz. Neste contexto, a área de síntese de voz tem o importante potencial de permitir a geração de fala a partir, por exemplo, da representação textual.

Metodologias estatísticas baseadas em Modelos Ocultos de Markov (HMM) têm sido aplicadas como forma de tornar o processo de síntese de fala mais simples e menos dependente de uma abundante massa de dados para a obtenção de boa prosódia. Além disso, tais técnicas possuem maior flexibilidade por permitirem a modificação de características da fala através da variação de parâmetros de modelos.

Todavia, a escassez de um *corpus* robusto e com grande volume de dados para o Português Brasileiro que possa ser empregado no treinamento desses sistemas tem se apresentado como uma dificuldade para a qualidade dos mesmos.

Desse modo, este trabalho contribui com o processamento de uma nova base que atenda a elevado padrão de qualidade, bem como a aplica na implementação de um sistema de síntese de voz baseado em HMM. A realização de testes subjetivos comprova o benefício da base bem como permite equiparar o sintetizador por HMM a sistemas comerciais utilizados na atualidade.

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

HIDDEN MARKOV MODELS BASED SPEECH SYNTHESIS USING A NEW DATABASE FOR BRAZILIAN PORTUGUESE

Dayana Sant'Anna Lole

January/2015

Advisor: Fernando Gil Vianna Resende Junior

Department: Electrical Engineering

Over time, computers have raised more and more in daily life and have contributed to growing demand for voice processing technologies. In this context, speech synthesis area has great potential to allow generating speech from the textual representation, for instance.

Statistical methods based on Hidden Markov Models (HMM) have been applied to make the process of speech synthesis simpler and less dependent on an abundant mass of data. Nevertheless, these techniques have greater flexibility by allowing modification of speech characteristics with model parameters variation.

However, the lack of a robust *corpus* and a large amount of data for the Brazilian Portuguese which can be used during these systems training has been presented as a problem for their quality.

Thereby, this work contributes to the processing of a new voice database that meets high quality standards, as well as apply it in the implementation of a HMM-based speech synthesizer. Subjective tests prove the benefit of the base and show how close the HMM synthesizer is to commercial systems.

Sumário

Lista de Figuras xi

Lista de Tabelas xiv

Lista de Acrônimos xv

Capítulo 1 Introdução 1

1.1 Motivação 1

1.2 Objetivos 3

1.3 Organização 4

Capítulo 2 Síntese de voz baseada em HMM 5

2.1 Introdução 5

2.2 Sistemas de Conversão Texto-Fala 6

2.2.1 Processamento da Linguagem Natural 6

2.2.2 Sintetizador de Fala 7

2.3 Modelos Ocultos de Markov 10

2.4 Geração de parâmetros 13

2.4.1 Critério: Máxima Verossimilhança 14

2.4.2 Critério: Expectativa-Maximização 16

2.4.3 Consideração da variância global 21

2.5 Conclusão 23

Capítulo 3 Treino com base de dados extensa para Português Brasileiro 25

3.1 Introdução 25

3.2 A base de dados 25

3.2.1 Projeto e Gravação 25
3.2.2 Edição 26
3.3 Preparação de dados 29
3.3.1 Desambiguação de homógrafos 30
3.3.2 Marcação de tonicidade 32
3.3.3 Separação silábica 34
3.3.4 Conversão grafema-fone 35
3.4 Treinamento 38
3.4.1 Geração de *labels* 38
3.4.2 Extração de parâmetros de fala 40
3.4.3 Treinamento dos HMM 40
3.5 Conclusão 44
Capítulo 4 Avaliações Subjetivas.....45
4.1 Introdução 45
4.2 Testes 45
4.2.1 Método de avaliação 45
4.2.2 Base de dados 46
4.2.3 Descrição 48
4.2.4 Plataforma 49
4.3 Resultados 52
4.3.1 Perfil dos avaliadores 52
4.3.2 Métricas 57
4.3.3 Remoção de *outliers* 57
4.3.4 Análise de resultados 58
4.4 Conclusão 62
Capítulo 5 Conclusão e Trabalhos Futuros 63
5.1 Conclusão 63

5.2 Trabalhos Futuros 64

Referências Bibliográficas 65

Apêndice A Prova da desigualdade de Jensen 73

A.1 Enunciado 73

A.2 Prova 73

Apêndice B Edição de áudio utilizando ProTools 10.0 77

B.1 Criação de Sessão 77

B.2 Segmentação das frases 81

B.3 Edição de áudio 83

B.4 Funcionalidades úteis 85

Apêndice C Treino do sintetizador utilizando HTS 88

C.1 Preparação do ambiente 88

C.2 Preparação de dados 88

C.2.1 Criação de dicionário fonético 88

C.2.2 Criação de arquivos de transcrição de palavras 90

C.2.3 Codificação de áudio 92

C.2.4 Segmentação automática 93

C.3 Treinamento 98

Lista de Figuras

Fig. 1.1 Visão geral da arquitetura de um sistema TTS típico 2

Fig. 2.1 Diagrama de Sistema de Conversão Texto-Fala 6

Fig. 2.2 Síntese de fala baseada em concatenação de formas de onda 8

Fig. 2.3 Síntese de fala baseada em HMM 9

Fig. 2.4 Modelo de Markov de 1ª ordem 10

Fig. 2.5 HMM de *S* estados sem saltos da esquerda para a direita 12

Fig. 2.6 Fluxograma do algoritmo de geração de parâmetros: "?" representa o critério de parada adotado 21

Fig. 3.1 Interface do ProTools 10.0 durante edição de áudio 27

Fig. 3.2 Etapas de treinamento dos HMM 29

Fig. 3.3 Trecho da análise de homógrafos da frase "Pesquisa é uma coisa que muda a toda hora." 32

Fig. 3.4 Marcação de sílaba tônica para a frase "Pesquisa é uma coisa que muda a toda hora." 33

Fig. 3.5 Separação silábica para a frase "Pesquisa é uma coisa que muda a toda hora." 35

Fig. 3.6 Rótulos gerados pela conversão grafema-fonema para a sentença "Pesquisa é uma coisa que muda a toda hora." 37

Fig. 3.7 Exemplo de arquivo *lab* para a frase "Pesquisa é uma coisa que muda a toda hora." 39

Fig. 3.8 Geração de observações 41

Fig. 3.9 Obtenção de árvore de decisão para o grupo *j* 43

Fig. 4.1 Interface de entrada para acesso aos testes 50

Fig. 4.2 Formulário para registro de novo avaliador 50

Fig. 4.3 Tela inicial dos testes 51

Fig. 4.4 Parte da página de avaliação com instruções ao usuário 51

Fig. 4.5 Tela inicial para usuário administrador com opção “Síntese de voz” 52

Fig. 4.6 Interface do sintetizador 52
Fig. 4.7 Distribuição de avaliadores da pesquisa 1 por faixa etária 53
Fig. 4.8 Distribuição de avaliadores da pesquisa 2 por faixa etária 54
Fig. 4.9 Distribuição de avaliadores da pesquisa 3 por faixa etária 54
Fig. 4.10 Distribuição de avaliadores da pesquisa 4 por faixa etária 54
Fig. 4.11 Distribuição de avaliadores da pesquisa 1 por grau de escolaridade 55
Fig. 4.12 Distribuição de avaliadores da pesquisa 2 por grau de escolaridade 55
Fig. 4.13 Distribuição de avaliadores da pesquisa 3 por grau de escolaridade 55
Fig. 4.14 Distribuição de avaliadores da pesquisa 4 por grau de escolaridade 56
Fig. 4.15 Distribuição de avaliadores das pesquisas 1 (a) e 2 (b) por sexo 56
Fig. 4.16 Distribuição de avaliadores das pesquisas 3 (a) e 4 (b) por sexo 56
Fig. 4.17 Frequência de respostas considerando cada frase para a pesquisa 1 59
Fig. 4.18 Frequência de respostas considerando cada frase para a pesquisa 2 60
Fig. 4.19 Frequência de respostas considerando cada frase para a pesquisa 3 61
Fig. 4.20 Frequência de respostas considerando cada frase para a pesquisa 4 61
Fig. B.1 Caixa de diálogo inicial 77
Fig. B.2 Caixa de diálogo para salvar o nome da nova sessão 78
Fig. B.3 Janela de edição 78
Fig. B.4 Inclusão de áudio na sessão 79
Fig. B.5 Caixa de diálogo para importação de áudio 80
Fig. B.6 Opções de importação de áudio 80
Fig. B.7 Janela de edição com nova trilha de áudio 80
Fig. B.8 Barra na trilha para acompanhamento do áudio tocado 81
Fig. B.9 Menu com ferramenta de seleção de trecho destacada 81
Fig. B.10 Trecho selecionado na trilha (fundo mais escuro) 81
Fig. B.11 Utilização da ferramenta para exportação de trecho 82
Fig. B.12 Caixa de diálogo de "Bounce" 82

Fig. B.13 Inserção de nome de arquivo a ser exportado 83

Fig. B.14 Menu onde é possível selecionar o modo de edição "SLIP" 83

Fig. B.15 Modo "SHUFFLE" selecionado 84

Fig. B.16 Edição utilizando modo "SLIP" 84

Fig. B.17 Resultado da edição em modo "SLIP" 84

Fig. B.18 Resultado da edição em modo "SHUFFLE" 85

Fig. B.19 Navegação na trilha de edição 85

Fig. B.20 Ferramenta de *zoom* horizontal 86

Fig. B.21 Alternativa da ferramenta de zoom horizontal 86

Fig. B.22 Ferramenta de *zoom* vertical 87

Fig. B.23 Menu de contexto 87

Fig. C.1 Exemplo de arquivo de *log*.....90

Fig. C.2 Exemplo de parte de arquivo *words.mlf* 91

Fig. C.3 Conteúdo do arquivo *mkphones0.led*, que deve conter uma linha em branco no final 91

Fig. C.4 Conteúdo do arquivo *mkphones1.led*, que deve conter uma linha em branco no final 92

Fig. C.5 Conteúdo do arquivo *config* 93

Fig. C.6 Conteúdo do arquivo *proto* 94

Fig. C.7 Conteúdo do arquivo *config2* 94

Fig. C.8 Exemplo de parte do arquivo *hmmdefs* 95

Fig. C.9 Exemplo de arquivo *macros*.....95

Fig. C.10 Conteúdo do arquivo *sil.hed* 96

Fig. C.11 Exemplo de parte do conteúdo do arquivo *aligned.mlf* 97

Fig. C.12 Estrutura de arquivos para o treinamento 99

Lista de Tabelas

Tab. 3-1 Equipamentos utilizados para a gravação da base 26

Tab. 3-2 Quantitativo de frases por conteúdo 28

Tab. 3-3 Regras aplicadas aos homógrafos 31

Tab. 3-4 Regras para determinação de tonicidade 34

Tab. 3-5 Alfabeto fonético SAMPA 36

Tab. 3-6 Informações nos arquivos de *label* 39

Tab. 4-1 Conjunto de frases foneticamente balanceadas aplicadas à pesquisa 1 46

Tab. 4-2 Conjunto de frases foneticamente balanceadas aplicadas à pesquisa 2 47

Tab. 4-3 Conjunto de frases foneticamente balanceadas aplicadas à pesquisa 3 47

Tab. 4-4 Conjunto de frases foneticamente balanceadas aplicadas à pesquisa 4 48

Tab. 4-5 Totais de participantes por avaliação 53

Tab. 4-6 Resultados da pesquisa 1 59

Tab. 4-7 Resultados da pesquisa 2 59

Tab. 4-8 Resultados da pesquisa 3 61

Tab. 4-9 Resultados da pesquisa 4 61

Tab. C-1 Ferramentas empregadas na preparação do ambiente 88

Lista de Acrônimos

EM – Expectativa-Maximização

G2P – *Grapheme to Phoneme* (Grafema para Fonema)

HD – *Hard Drive*

HMM – *Hidden Markov Models* (Modelos Ocultos de Markov)

HTK – *Hidden Markov Model Toolkit*

HTS – *HMM-based Speech Synthesis System*

JSAPI – *Java Speech* API

LPC – *Linear Prediction Coding*

MOS – *Mean Opinion Score*

NLP – *Natural Language Processing* (Processamento da Linguagem Natural)

PB – Português Brasileiro

PSOLA – *Pitch Synchronous Overlapp and Add*

PUC-RJ – Pontífica Universidade Católica do Rio de Janeiro

RAM – *Random Access Memory*

SAMPA – *Speech Assessment Methods Phonetic Alphabet*

SAPI – *Speech* API

SSD – *Solid-State Drive*

TTS – *Text-to-Speech* (Texto-para-Fala)

UFPA – Universidade Federal do Pará

UFSC – Universidade Federal de Santa Catarina

UNICAMP – Universidade de Campinas

Capítulo 1

Introdução

1.1 Motivação

Ao longo do tempo, as máquinas têm angariado cada vez mais espaço no cotidiano. Diversas funções complexas, antes desempenhadas exclusivamente por seres humanos, passam a ser executadas pelos computadores. Assim, há uma crescente demanda por tecnologias de processamento de voz que confiram aos dispositivos eletrônicos a capacidade de comunicação.

Adicionalmente, o cenário cada vez mais integrado de propagação de conhecimento, promovido por esses avanços tecnológicos, propõe o constante uso de mídias digitais para distribuir a informação. Nesse contexto, a pesquisa na área de processamento de sinais de fala, através dos estudos em síntese e reconhecimento de voz, tem alcançado destaque por promover uma dimensão mais natural à interação homem-máquina, cujo valor torna-se mais perceptível quando considerada a acessibilidade para deficientes físicos, como é o caso dos leitores de tela de computador ORCA [1] e DOSVOX [2].

O processamento de sinais de fala tem atraído não somente a comunidade acadêmica, como também despertado o interesse de grandes empresas. Este é o caso do aplicativo SAPI (*Speech* API), desenvolvido pela Microsoft e integrado ao seu sistema operacional Windows 8 [3], e do JSAPI (*Java Speech* API), alternativa apresentada pela Oracle [4].

A subárea de síntese de voz, foco deste trabalho, permite a produção artificial da fala através da conversão da linguagem natural escrita em voz. Sistemas que realizam essa tarefa são tipicamente denominados TTS (*Text-to-Speech)*. A qualidade desses sistemas está atrelada à capacidade de gerar fala inteligível e natural [5]. Estas duas características, portanto, têm norteado o desenvolvimento e aprimoramento de técnicas dos sintetizadores de fala [6], representando sua principal forma de avaliação.

Fig. 1.1 Visão geral da arquitetura de um sistema TTS típico

Geralmente, tais sistemas apresentam uma arquitetura (Fig. 1.1), melhor abordada mais adiante, composta de duas partes: a primeira é destinada ao Processamento da Linguagem Natural (*Natural Language Processing* - NLP), também conhecida como *Front End,* e a segunda, à síntese de voz propriamente dita (*Back End*) [7]. Para este, existem diferentes técnicas de implementação.

Atualmente, tais técnicas têm refletido a mudança de um paradigma axiomático baseado no conhecimento para um dirigido a dados [8]. As principalmente utilizadas têm sido baseadas em formantes, em concatenação e em Modelos Ocultos de Markov (HMM - *Hidden Markov Models*) [7]. Esta última, a abordagem mais recente, teve o primeiro artigo publicado em 1995, onde apresentou-se a geração de parâmetros de fala através de HMM [9]. Esta técnica permite o treino com o uso de características de contexto, incluindo informações a nível de fones, sílabas, palavras, frases e pronúncias [10]. Quando comparada aos outros métodos, esta abordagem tem a vantagem de permitir a geração de fala sintetizada com boa prosódia, mesmo utilizando bases de treinamento pequenas. Este fato é retratado em [11], onde uma base de 80 sentenças foi utilizada. Outra vantagem da referida técnica é a possibilidade de mudança da voz do locutor sem a necessidade de gravação de uma base extensa [12, 13].

Utilizando a abordagem por HMM, diversos trabalhos vêm sendo apresentados objetivando, por exemplo, a adaptação de locutor [14-17], a interpolação de locutores [12, 18] e o desenvolvimento de aplicações para diversos idiomas [19-24].

No caso específico de síntese de voz para o português brasileiro, trabalhos pioneiros foram desenvolvidos na Universidade de Campinas (UNICAMP), através da abordagem por formantes [25], e Pontifica Universidade Católica do Rio de Janeiro (PUC-RJ), pela técnica de concatenação [26, 27], ainda na década de 1990. De acordo com [28], sistemas baseados nessas técnicas têm alcançado elevado grau de maturidade e qualidade.

Tendo em vista as vantagens citadas anteriormente, em 2005, foi desenvolvido o primeiro sintetizador baseado em HMM para o português brasileiro por [29]. Desde então, alguns trabalhos sobre este tema têm sido realizados através de grupos de pesquisa na Universidade Federal de Santa Catarina (UFSC) [30] e na Universidade Federal do Pará (UFPA) [7, 31]. Nesta última, é realizado o projeto FalaBrasil [32] que visa criar e disponibilizar recursos *open source* para pesquisadores e desenvolvedores. Hoje, o projeto disponibiliza a suite SimonBR [33] para auxiliar pessoas com necessidades especiais no uso de computadores através de comandos por voz.

A qualidade dessas aplicações, todavia, se beneficia da disponibilidade de um *corpus* robusto e com grande volume de dados [10, 34]. O primeiro registro de preocupação com o conteúdo da base de dados para o português falado no Brasil foi realizado através de um banco de 200 frases foneticamente balanceadas [35], o qual foi utilizado em [29] para o treinamento do sintetizador por HMM.

Existe uma grande quantidade de dados para diversos idiomas, como o inglês, japonês, alemão, dentre outros [36-38]. No entanto, no caso do português brasileiro, há poucos recursos acessíveis [34]. O *corpus* utilizado em [29], por exemplo, compreendeu 221 sentenças gravadas por um locutor do sexo masculino, equivalendo a 18 minutos e 48 segundos de fala.

Há diversos esforços no sentido de suprir essa carência de dados [28, 39-41], os mesmos, entretanto, apresentam dificuldades como a gravação em ambientes não controlados e/ou por diversos locutores de diferentes localidades do Brasil. Estes aspectos são de especial relevância para a síntese de voz, uma vez que é necessária uma grande quantidade de frases pronunciadas por um único locutor [41].

Tendo em vista as vantagens da abordagem de síntese por HMM e os entraves apresentados a respeito da base de dados para o português brasileiro, foram atribuídos a este trabalho os objetivos apresentados na seção a seguir.

1.2 Objetivos

Este trabalho tem o intuito de apresentar um sistema *web* que funcione como ferramenta didática no estudo e aprimoramento da conversão TTS baseada em HMM no idioma português brasileiro. Desta forma, os seguintes objetivos foram traçados:

- Processamento de uma ampla base de síntese de voz com qualidade de gravação e edição de estúdio;

- Aplicação da nova base de dados na preparação de um sistema de síntese de voz baseado em HMM;
- Inspeção do reflexo da quantidade de dados do treinamento na qualidade da voz sintetizada através de testes subjetivos realizados em plataforma *web*;
- Elaboração de tutorial de edição utilizando ProTools 10.0;
- Composição de manual para realização de treinamento do sistema de síntese de voz através da ferramenta HTS [86].

1.3 Organização

A presente dissertação está organizada em cinco capítulos. Este capítulo apresentou a motivação para o desenvolvimento deste trabalho, bem como os objetivos da pesquisa.

No Capítulo 2, é apresentado o panorama teórico em que se baseia a síntese por HMM, com ênfase nos aspectos de interesse para a compreensão da ferramenta apresentada por este trabalho.

Já no Capítulo 3, é realizada uma descrição sobre o treino do sistema de síntese, bem como sua arquitetura e algoritmos empregados. Neste âmbito também é apresentada a nova base de voz utilizada e detalhes de seu tratamento.

O Capítulo 4 descreve os testes realizados para análise da influência do tamanho da base na qualidade da síntese por HMM e para comparação com sistemas comerciais.

Por fim, o Capítulo 5 trata as conclusões do trabalho e os novos desafios que se apresentam.

Capítulo 2

Síntese de voz baseada em HMM

2.1 Introdução

A área de pesquisa em síntese de voz tem obtido grande progresso nos últimos anos, apoiada pelo avanço dos processadores e pela crescente capacidade de armazenamento de dados em dispositivos de pequeno porte. A literatura tem apontado para o desenvolvimento de novas técnicas relacionadas à síntese de voz para diversos idiomas [19-24], com diferentes estilos [12, 14-18] e emoções [42-45].

Apesar das frequentes pesquisas em síntese de voz, a técnica predominantemente empregada é a de concatenação [10]. Esse método possui como vantagem a elevada qualidade alcançada, uma vez que se utiliza da forma de onda da fala real para compor suas unidades acústicas. Todavia, para o alcance desta qualidade, esta técnica demanda a coleta de uma extensa base de dados. A essa condição também fica sujeita a viabilidade de variação de emoções e locutores.

O método baseado em HMM surgiu recentemente como alternativa estocástica e, portanto, treinável para o problema da síntese de fala. Esta técnica tem a vantagem de permitir a obtenção de boa prosódia e até a alteração de características de voz com uma base de dados pequena [13, 18, 19, 46]. Em [10], aponta-se como principal desvantagem deste método a qualidade gerada pelo *vocoder*, uma vez que consiste de um LPC (*Linear Prediction Coding*). Uma possível solução deste problema se dá pelo uso da excitação mista [47], através do uso de pulsos mistos e ruído, periódicos e aperiódicos e filtro de dispersão de pulsos, de forma a reproduzir características naturais da fala.

Este capítulo visa apresentar os aspectos teóricos envolvidos na compreensão deste trabalho. Assim, na Seção 2.2 visa contextualizar os HMM dentro da arquitetura dos sistemas TTS. A Seção 2.3 apresenta de modo mais formal os HMM e a nomenclatura utilizada neste trabalho. A Seção 2.4 busca apresentar como é realizada a geração de parâmetros para a síntese de fala a partir dos HMM. A conclusão do capítulo é exposta na Seção 2.5.

2.2 Sistemas de Conversão Texto-Fala

Sistemas de conversão texto-fala, ou como são frequentemente referidos, TTS, realizam o mapeamento de linguagem escrita em voz inteligível reproduzida por dispositivo de áudio acoplado. Tal sistema (Fig. 2.1) é composto basicamente de dois blocos principais: Processamento da Linguagem Natural (NLP - *Natural Language Processing*) e Sintetizador da Fala.

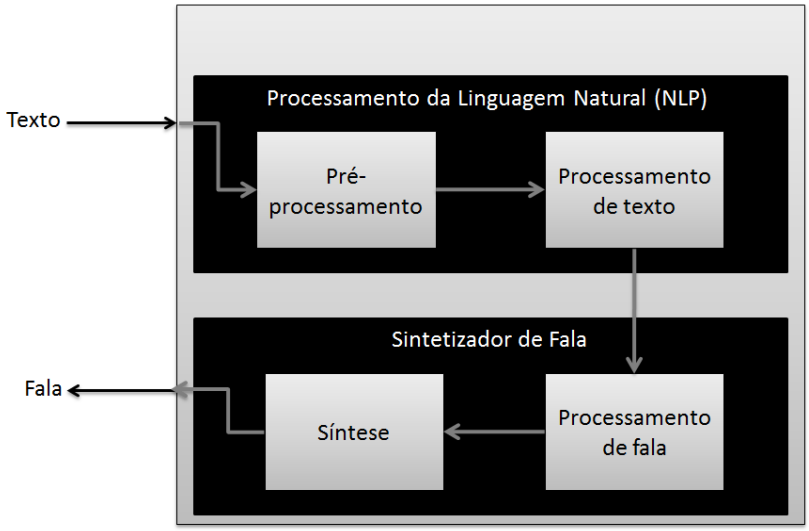


Fig. 2.1 Diagrama de Sistema de Conversão Texto-Fala

2.2.1 Processamento da Linguagem Natural

O NLP é a parte do sistema dependente da linguagem. Isso se justifica a partir de sua responsabilidade na tradução de texto (caracteres, números e abreviações) em informações de pronúncia, que diz respeito ao conhecimento das unidades acústicas a serem produzidas. Este bloco pode, ainda, ser dividido em outros dois: pré-processamento e processamento de texto.

No pré-processamento, também denominado como normalização, é feita a transcrição de abreviações, acrônimos, números, entre outros, em palavras escritas. Essas informações são introduzidas no bloco de processamento de texto, que, então, obtém as informações de pronúncia que incluem:

- Separação em sílabas: divisão da palavra em unidades menores, de acordo com regras pré-definidas conforme cada idioma;

- Conversão grafema-fonema: mapeia os caracteres do texto de entrada em fones (unidades acústicas básicas), de acordo com as regras expostas em um dicionário;
- Determinação de sílaba tônica: indica as sílabas que deverão ser pronunciadas com maior intensidade dentro das palavras;
- Determinação de função de palavra de acordo com o contexto: analisa e classifica a função morfológica, semântica e sintática da palavra dentro do contexto em que se encontra, servindo frequentemente para a desambiguação de homógrafos de pronúncias diferentes;
- Cálculo de duração das unidades acústicas: determina a duração das unidades acústicas de acordo com as informações linguísticas;
- Frequência fundamental (F0): relaciona-se com as características da voz do locutor, estando ligada à espessura e ao tamanho das pregas vocais.

2.2.2 Sintetizador de Fala

O Sintetizador de Fala gera, a partir das informações de pronúncia, a fala. Esta parte também pode ser subdividida em dois blocos: processamento de fala e síntese. O primeiro utiliza as informações de pronúncia para a geração de parâmetros. Este último é o responsável pela geração da fala propriamente dita através da geração da forma de onda.

Diversas abordagens têm sido utilizadas no desenvolvimento de síntese de voz a partir dos parâmetros gerados pelas informações de pronúncia, dentre elas, três se destacam:

- Síntese de voz por regras;
- Síntese de voz por concatenação de formas de onda;
- Síntese de voz por Modelos Ocultos de Markov.

Na síntese por regras, a produção da fala é realizada a partir do mapeamento de parâmetros selecionados por regras estabelecidas previamente em comandos para articuladores, os quais serão empregados na geração da voz [6]. A despeito da possibilidade de adaptação às vozes de diferentes locutores e de não necessitar do uso de uma base de dados extensa, a implementação de regras para esse método é uma tarefa onerosa e complexa, cuja qualidade é alcançada por sistemas de mais simples realização [29].

A síntese por concatenação de formas de onda (Fig. 2.2) realiza o encadeamento de unidades acústicas selecionadas apropriadamente a partir de uma base de dados. A seleção das unidades é realizada por funções de custo e a concatenação, por técnicas de transformação (e.g. PSOLA: *Pitch Synchronous Overlapp and Add*) das unidades acústicas presentes na base. As unidades normalmente utilizadas são difones [48], uma vez que preservam transições entre unidades e apresentam descontinuidades mínimas frente a outros tipos de unidades acústicas (palavras, sílabas e demi-sílabas). Em função da qualidade segmental obtida, esse método se tornou mais popular que aquele por regras.

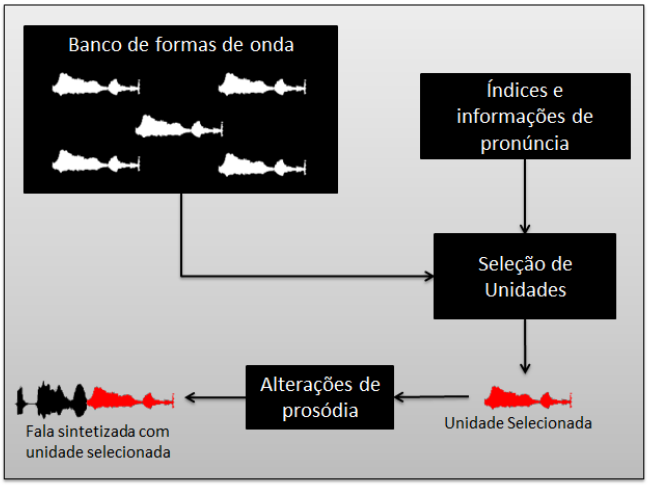


Fig. 2.2 Síntese de fala baseada em concatenação de formas de onda

Todavia, a despeito da qualidade alcançada, a técnica de concatenação possui como desvantagem marcante a reprodução do estilo de fala do banco utilizado. Assim, para uma maior variedade de características (emoções, sotaque, etc.), uma solução custosa e difícil se apresenta: a gravação de grande quantidade de dados de maneira a abranger enorme variedade de possibilidades fonéticas. Além disso, este tipo de síntese não possibilita a implementação de um sintetizador único para diferentes idiomas.

Em contraste com a técnica acima mencionada, metodologias estatísticas baseadas em HMM têm sido aplicadas como forma de tornar o processo de síntese de fala mais simples e menos dependente de uma abundante massa de dados. Nesse método, os HMM são treinados de acordo com contextos apresentados na base de dados para a geração de parâmetros que dão origem à fala sintetizada. A fala é, então, gerada pela maximização da probabilidade de observação de determinada sequência como saída do sistema. Esta técnica possui comprovada vantagem na reprodução fiel de

características prosódicas mesmo com uma base de dados pequena [11]. Ademais, a técnica de geração de voz falada por HMM viabiliza maior flexibilidade por simplificar a modificação de características da fala (e.g. adaptação de locutor, alteração de estilos, etc) pela variação de parâmetros dos modelos através de poucas gravações [12, 13]. Isso se justifica por não se tratar de um tratamento baseado em um banco de dados fixo, mas no transporte de características da fala para a modelagem matemática. A Fig. 2.3 ilustra o processo de síntese de fala por HMM. Um melhor detalhamento deste método é fornecido nas seções subsequentes.

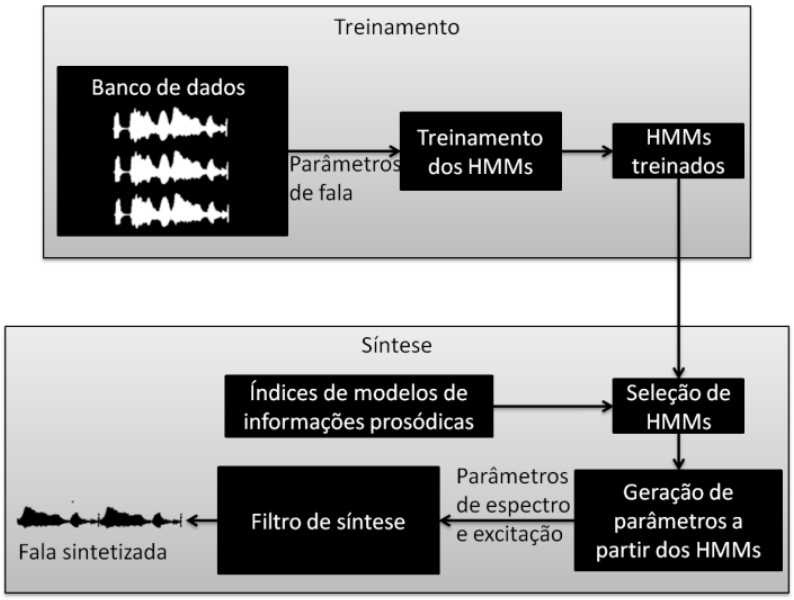


Fig. 2.3 Síntese de fala baseada em HMM

Há, ainda, métodos de síntese que utilizam outras arquiteturas de sistema, tais como a síntese articulatória e de formantes. A primeira consiste da reprodução da articulação humana através de modelagem de características físicas, anatômicas e fisiológicas dos órgãos envolvidos na produção da voz humana (lábios, língua e cordas vocais) [49]. Já a síntese de formantes faz uso da combinação de frequências submetidas a ressonadores para constituir a fala [5]. No entanto, frente a ambos os métodos, as abordagens por concatenação de forma de onda e por HMM têm apresentado maior qualidade prosódica, sendo essa a justificativa para a sua ênfase neste texto.

2.3 Modelos Ocultos de Markov

Um HMM é um autômato estocástico de estados finitos utilizado para modelar uma pronúncia de fala que pode ser uma sentença completa, uma palavra, uma sílaba ou um fone. No caso de vocabulários pequenos, é possível que um HMM modele palavras. Todavia, em vocabulários extensos, é comum que represente fones [50].

Considere a Fig. 2.4, conhecida como modelo de Markov Observável ou cadeia de Markov. Este modelo pode ser caracterizado a qualquer momento a partir de um de seus 3 estados, $\{s_1, s_2, s_3\}$.

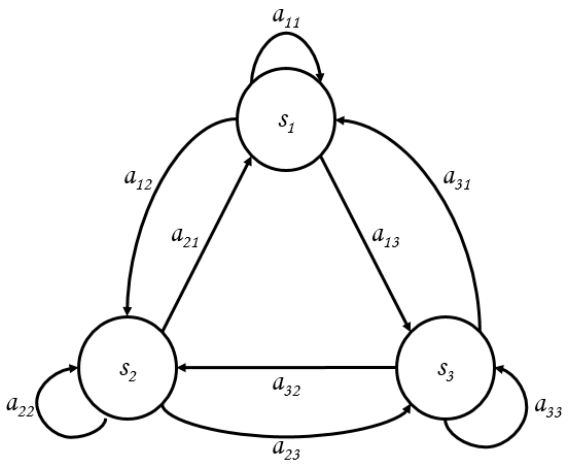


Fig. 2.4 Modelo de Markov de 1ª ordem

Assim, a cada instante de observação t, assume-se que uma transição de estados acontece. Esta transição ocorre de acordo com as probabilidades de transição entre estados. Isto é, o estado q_t é alcançado a partir do estado q_{t-1} de acordo com a probabilidade estacionária no tempo $a_{ij} = P\{q_t = s_j | q_{t-1} = s_i\}$. A caracterização das transições de estados é, em geral, fornecida através de uma matriz de transição de estados dada por

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & \cdots & a_{1S} \\ \vdots & \ddots & \vdots \\ a_{S1} & \cdots & a_{SS} \end{bmatrix}, \tag{2.1}$$

onde S representa o número de estados no modelo. No caso da Fig. 2.4, $S = 3$.

Supondo que seja necessário conhecer a probabilidade de ocorrência da sequência de estados $\boldsymbol{q} = \{q_1, q_2, q_3, q_4\} = \{s_1, s_2, s_2, s_3\}$, utilizando a matriz definida em (2.1) para $S = 3$, temos

$$P\{\boldsymbol{q}\} = P\{q_1 = s_1\} \cdot P\{q_2 = s_2\} \cdot P\{q_3 = s_2\} \cdot P\{q_4 = s_3\} = \pi_1 a_{12} a_{22} a_{23},$$

onde π_i é a probabilidade de ter o estado s_i como estado inicial, ou seja,

$$\pi_i = P\{q_t = s_i\}. \tag{2.2}$$

À semelhança da matriz de transição de estados, define-se como matriz de estados iniciais aquela dada por

$$\boldsymbol{\Pi} = \begin{bmatrix} \pi_1 \\ \vdots \\ \pi_S \end{bmatrix}. \tag{2.3}$$

As matrizes de transição e de estados iniciais definem completamente a probabilidade de se estar em um determinado estado em qualquer instante de observação.

Considere, agora, o sistema denotado na Fig. 2.5. Nela, as observações reais, representadas pela sequência de vetores $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_N\}$, são modeladas como processos estocásticos discretos no tempo. Portanto, as observações são derivadas estatisticamente da sequência de estados de saída "escondidos" $\boldsymbol{q} = \{q_1, \ldots, q_N\}$. Este modelo é conhecido como Modelo Oculto de Markov (*Hidden Markov Model* - HMM) de S estados. No caso de problemas de processamento de voz, por exemplo, os estados podem representar a posição dentro de uma palavra e as observações dizem respeito ao som correspondente.

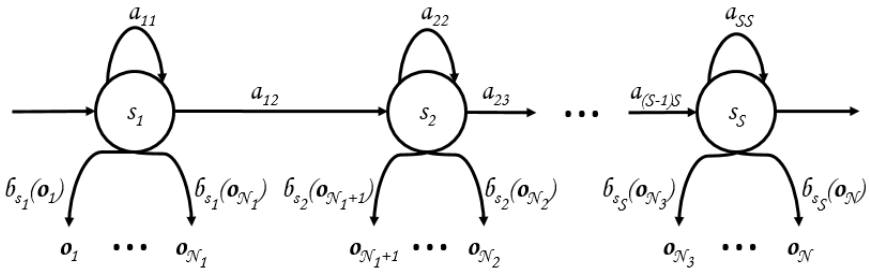


Fig. 2.5 HMM de *S* estados sem saltos da esquerda para a direita

Desse modo, as estatísticas que originam as observações a partir dos estados são dadas por $b_{s_i}(\boldsymbol{o}_t)$. Assim, um modelo λ pode ser totalmente caracterizado como [51]

$$\lambda = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\Pi}), \tag{2.4}$$

onde $\boldsymbol{A}$ é a matriz de transição de estados e $\boldsymbol{\Pi}$ é o vetor de estados iniciais, apresentadas anteriormente através das equações (2.1) e (2.3), respectivamente. $\boldsymbol{B}$ é a matriz de distribuição de probabilidades de saída caracterizado por

$$\boldsymbol{B} = \begin{bmatrix} b_{s_1}(\boldsymbol{o}_1) & \cdots & b_{s_S}(\boldsymbol{o}_1) \\ \vdots & \ddots & \vdots \\ b_{s_1}(\boldsymbol{o}_N) & \cdots & b_{s_S}(\boldsymbol{o}_N) \end{bmatrix}, \tag{2.5}$$

onde

$$b_{s_i}(\boldsymbol{o}_t) = P\{\boldsymbol{o}_t | q_t = s_i\}. \tag{2.6}$$

A Fig. 2.5 caracteriza um HMM comumente utilizado nos trabalhos desenvolvidos na área de processamento de voz, também empregado nesta dissertação. Nela, a estrutura do HMM é da esquerda para a direita e sem saltos, isto é, apenas é possível que se passe do estado atual $q_t = s_i$ para o mesmo estado $q_{t+1} = s_i$ ou um subsequente $q_{t+1} = s_{i+1}$.

Para um conhecimento mais aprofundado a respeito dos conceitos envolvidos pelos HMM sugere-se a leitura complementar de [50, 51].

2.4 Geração de parâmetros

Seja a sequência de estados $\boldsymbol{q} = \{q_1, \dots, q_N\}$, de observações $\boldsymbol{O} = \{\boldsymbol{o}_1, \dots, \boldsymbol{o}_N\}$ e um HMM λ cujo vetor de distribuição de probabilidade de saída possui componentes definidas como misturas de gaussianas. Por simplicidade, uma única gaussiana será considerada para as distribuições de saída nas demonstrações, isto é:

$$b_{q_i}(\boldsymbol{o}_t) = \mathcal{N}\left(\boldsymbol{o}_t; \boldsymbol{\mu}_{q_i}, \boldsymbol{U}_{q_i}\right) = \frac{1}{\sqrt{(2\pi)^M \left|\boldsymbol{U}_{q_i}\right|}} e^{-\frac{1}{2}\left(\boldsymbol{o}_t - \boldsymbol{\mu}_{q_i}\right)^T \boldsymbol{U}_{q_i}^{-1}\left(\boldsymbol{o}_t - \boldsymbol{\mu}_{q_i}\right)}, \tag{2.7}$$

onde M é a dimensionalidade de $\boldsymbol{o}_t$ e $\mathcal{N}\left(\cdot; \boldsymbol{\mu}_{q_i}, \boldsymbol{U}_{q_i}\right)$ é uma gaussiana multivariada de vetor de médias $\boldsymbol{\mu}$ e matriz de covariâncias $\boldsymbol{U}$. Os resultados podem ser estendidos para o caso de várias misturas [52].

Deseja-se maximizar $P\{\boldsymbol{O}|\lambda\}$ com respeito a $\boldsymbol{O}$ sob as seguintes condições:

$$\boldsymbol{o}_t = [\boldsymbol{c}_t^T \quad \Delta\boldsymbol{c}_t^{\mathrm{T}} \quad \Delta^2\boldsymbol{c}_t^T]^T, \tag{2.8}$$

$$\Delta\boldsymbol{c}_t = \sum_{\tau=-L_-^{(1)}}^{L_+^{(1)}} \boldsymbol{w}^{(1)}(\tau)\, \boldsymbol{c}_{t+\tau}, \tag{2.9}$$

e

$$\Delta^2\boldsymbol{c}_t = \sum_{\tau=-L_-^{(2)}}^{L_+^{(2)}} \boldsymbol{w}^{(2)}(\tau)\, \boldsymbol{c}_{t+\tau}. \tag{2.10}$$

onde $\boldsymbol{c}_t$ representa os coeficientes mel-cepstrais, $\Delta\boldsymbol{c}_t$ e $\Delta^2\boldsymbol{c}_t$ são os coeficientes delta e delta-delta (vetores dinâmicos de características) e $\boldsymbol{w}^{(1)}(\cdot)$ e $\boldsymbol{w}^{(2)}(\cdot)$ representam a sequência de pesos usadas para o cálculo dos coeficientes em (2.9) e (2.10). Os coeficientes $\Delta\boldsymbol{c}_t$ e $\Delta^2\boldsymbol{c}_t$ marcam a influência dos instantes mais próximos da observação atual.

Em [52] são apresentadas três alternativas para a geração de parâmetros, que são utilizadas pela ferramenta HTS [53], empregada para a síntese de voz por HMM. Estas alternativas são:

1. Maximizar $P\{\boldsymbol{O}|\boldsymbol{q},\lambda\}$ com relação a $\boldsymbol{O}$;
2. Maximizar $P\{\boldsymbol{O},\boldsymbol{q}|\lambda\}$ com relação a $\boldsymbol{O}$ e a $\boldsymbol{q}$;
3. Maximizar $P\{\boldsymbol{O}|\lambda\}$ com relação a $\boldsymbol{O}$.

A solução direta da segunda alternativa é impraticável, uma vez que seria necessário o cálculo de todas as combinações de estados para $\boldsymbol{q}$. Assim, para reduzir o custo computacional, [52] aponta para aproximações que fazem a solução do problema recair sobre a do primeiro caso. Dessa forma, a seguir, são apresentadas duas soluções para a geração de parâmetros.

2.4.1 Critério: Máxima Verossimilhança

Neste caso, a maximização é feita para uma sequência de estados fixa. Dada esta necessidade, é empregada uma solução sub-ótima para a determinação da sequência de estados $\boldsymbol{q}$ pela maximização da probabilidade de duração de estados $P\{\mathbf{q}|\lambda\}$ independente de $\boldsymbol{O}$, apresentada em [54, 55]. Como os estados $\boldsymbol{q}$ e o modelo λ são conhecidos, é possível denotar:

$$P\{\boldsymbol{O}|\boldsymbol{q},\lambda\} = b_{q_1}(\boldsymbol{o}_1)\cdot\ldots\cdot b_{q_N}(\boldsymbol{o}_N), \tag{2.11}$$

onde $b_{q_j}(\boldsymbol{o}_t)$ são definidos de acordo com (2.7). Então:

$$P\{\boldsymbol{O}|\boldsymbol{q},\lambda\} = \prod_{i=1}^{N} \frac{1}{\sqrt{(2\pi)^M |\boldsymbol{U}_{q_i}|}} e^{-\frac{1}{2}\left(\boldsymbol{o}_t-\mu_{q_i}\right)^T \boldsymbol{U}_{q_i}^{-1}\left(\boldsymbol{o}_t-\mu_{q_i}\right)}.$$

A fim de simplificar os cálculos, podemos aplicar a função logarítmica à equação anterior. Logo:

$$\ln P\{\boldsymbol{O}|\boldsymbol{q},\lambda\} = -\frac{MN}{2}\ln(2\pi) - \frac{1}{2}\sum_{i=1}^{N}|\boldsymbol{U}_{q_i}| - \frac{1}{2}\sum_{i=1}^{N}\left(\boldsymbol{o}_t-\mu_{q_i}\right)^T \boldsymbol{U}_{q_i}^{-1}\left(\boldsymbol{o}_t-\mu_{q_i}\right).$$

A equação anterior pode ser reescrita usando a definição de $\boldsymbol{O}$ e

$$\boldsymbol{M} = \left[\mu_{q_1}^T \quad \cdots \quad \mu_{q_N}^T\right]^T, \tag{2.12}$$

e

$$\boldsymbol{U} = diag\{\boldsymbol{U}_{q_1}^{-1}, \ldots, \boldsymbol{U}_{q_N}^{-1}\}. \tag{2.13}$$

obtendo-se

$$\ln P\{\boldsymbol{O}|\boldsymbol{q}, \lambda\} = -\frac{MN}{2}\ln(2\pi) - \frac{1}{2}\sum_{i=1}^{N}\left|\boldsymbol{U}_{q_i}\right| - \frac{1}{2}(\boldsymbol{O}-\boldsymbol{M})^T\boldsymbol{U}^{-1}(\boldsymbol{O}-\boldsymbol{M})$$

$$\ln P\{\boldsymbol{O}|\boldsymbol{q}, \lambda\} = -\frac{1}{2}\boldsymbol{O}^T\boldsymbol{U}^{-1}\boldsymbol{O} + \boldsymbol{O}^T\boldsymbol{U}^{-1}\boldsymbol{M} + K, \tag{2.14}$$

onde K é uma constante independente de $\boldsymbol{O}$ e

$$K = -\frac{MN}{2}\ln(2\pi) - \frac{1}{2}\sum_{i=1}^{N}\left|\boldsymbol{U}_{q_i}\right| - \frac{1}{2}\boldsymbol{M}^T\boldsymbol{U}^T\boldsymbol{M}.$$

Devido às restrições (2.8)-(2.10), podemos escrever $\boldsymbol{O}$ da seguinte forma:

$$\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}, \tag{2.15}$$

onde

$$\boldsymbol{C} = [\boldsymbol{c}_1 \quad \cdots \quad \boldsymbol{c}_N]^T, \tag{2.16}$$

$$\boldsymbol{W} = [\boldsymbol{w}_1 \quad \cdots \quad \boldsymbol{w}_N]^T, \tag{2.17}$$

$$\boldsymbol{w}_t = \left[\boldsymbol{w}_t^{(0)} \quad \boldsymbol{w}_t^{(1)} \quad \boldsymbol{w}_t^{(2)}\right]^T \tag{2.18}$$

e

$$\boldsymbol{w}_t^{(n)} = \left[\boldsymbol{O}_{MxM} \quad \cdots \quad \boldsymbol{O}_{MxM} \quad w^{(n)}\left(-L_-^{(n)}\right)\boldsymbol{I}_{MxM} \quad \cdots \quad w^{(n)}\left(L_+^{(n)}\right)\boldsymbol{I}_{MxM} \quad \boldsymbol{O}_{MxM} \quad \cdots \quad \boldsymbol{O}_{MxM}\right], \tag{2.19}$$

com $L_-^{(0)} = L_+^{(0)} = 0$ e $w^{(0)}(0) = 1$.

Substituindo (2.15) em (2.14):

$$\ln P\{\boldsymbol{O}|\boldsymbol{q},\lambda\} = -\frac{1}{2}\boldsymbol{C}^T\boldsymbol{W}^T\boldsymbol{U}^{-1}\boldsymbol{W}\boldsymbol{C} + \boldsymbol{C}^T\boldsymbol{W}^T\boldsymbol{U}^{-1}\boldsymbol{M} + K. \tag{2.20}$$

Como a função logarítmica é crescente, teremos que, se derivarmos (2.20) em relação a $\boldsymbol{C}$ e igualarmos a zero, obteremos um máximo. Assim:

$$\frac{\partial(\ln P\{\boldsymbol{O}|\boldsymbol{q},\lambda\})}{\partial \boldsymbol{C}} = -\boldsymbol{W}^T\boldsymbol{U}^{-1}\boldsymbol{W}\boldsymbol{C} + \boldsymbol{W}^T\boldsymbol{U}^{-1}\boldsymbol{M} = 0. \tag{2.21}$$

Logo, a solução da equação linear expressa em (2.21) permite que se obtenha $\boldsymbol{C}$ e, portanto, $\boldsymbol{O}$ que maximiza $P\{\boldsymbol{O}|\boldsymbol{q},\lambda\}$.

2.4.2 Critério: Expectativa-Maximização

2.4.2.1 Algoritmo de Expectativa-Maximização

O algoritmo de Expectativa-Maximização (EM), também conhecido como Baum-Welch [97], é uma ferramenta eficiente no cálculo de máxima verossimilhança no caso em que há dados ocultos. Trata-se de um procedimento iterativo que busca obter a estimativa de um modelo de parâmetros cuja relação com os dados observados seja a mais provável.

Seja $\boldsymbol{O} = \{\boldsymbol{o}_1, \dots, \boldsymbol{o}_N\}$ a sequência de vetores de observação e $\boldsymbol{q} = \{q_1, \dots, q_N\}$ sua correspondente sequência de estados para um dado HMM λ, deseja-se encontrar $\boldsymbol{O}$ tal que $P\{\boldsymbol{O}|\lambda\}$ seja máxima, ou seja, a estimativa de máxima verossimilhança para $\boldsymbol{O}$. Por facilidade matemática, define-se a função logarítmica de verossimilhança

$$L(\boldsymbol{O}) = \ln P\{\boldsymbol{O}|\lambda\}. \tag{2.22}$$

Tal função é estritamente crescente, logo, o valor de λ que a maximiza também maximizará $P\{\boldsymbol{O}|\lambda\}$.

Uma vez que se deseja obter um algoritmo capaz de estimar $\boldsymbol{O}$ tal que $L(\boldsymbol{O}') > L(\boldsymbol{O})$, onde $\boldsymbol{O}$ representa a estimativa atual e $\boldsymbol{O}'$, a nova estimativa, isto é equivalente a maximizar a função $D(\boldsymbol{O}, \boldsymbol{O}')$ definida como

$$D(\boldsymbol{O}, \boldsymbol{O}') = L(\boldsymbol{O}') - L(\boldsymbol{O}), \tag{2.23}$$

$$D(\boldsymbol{O}, \boldsymbol{O}') = \ln P\{\boldsymbol{O}'|\lambda\} - \ln P\{\boldsymbol{O}|\lambda\}. \tag{2.24}$$

Pode-se reescrever $P\{\boldsymbol{O}'|\lambda\}$ em função das sequências de estados (ocultas) como [52]

$$P\{\boldsymbol{O}'|\lambda\} = \sum_{\boldsymbol{q}} P\{\boldsymbol{O}', \boldsymbol{q}|\lambda\}$$

$$P\{\boldsymbol{O}'|\lambda\} = \sum_{\boldsymbol{q}} P\{\boldsymbol{O}'|\boldsymbol{q}, \lambda\}\, P\{\boldsymbol{q}|\lambda\}. \tag{2.24}$$

Substituindo em (2.24), temos

$$D(\boldsymbol{O}, \boldsymbol{O}') = \ln\left(\sum_{\boldsymbol{q}} P\{\boldsymbol{O}'|\boldsymbol{q}, \lambda\} P\{\boldsymbol{q}|\lambda\}\right) - \ln P\{\boldsymbol{O}|\lambda\}. \tag{2.26}$$

Por meio da desigualdade de Jensen (Apêndice A) pode-se dizer que, para constantes $k_i \geq 0$ e $\sum_{i=1}^{n} k_i = 1$,

$$\ln \sum_{i=1}^{n} k_i x_i = \sum_{i=1}^{n} k_i \ln(x_i).$$

Usando como constantes as probabilidades $P\{\boldsymbol{q}|\boldsymbol{O}, \lambda\}$, uma vez que $P\{\boldsymbol{q}|\boldsymbol{O}, \lambda\} \geq 0$ e $\sum_{\boldsymbol{q}} P\{\boldsymbol{q}|\boldsymbol{O}, \lambda\} = 1$, pode-se desenvolver (2.26) como se segue:

$$D(\boldsymbol{O}, \boldsymbol{O}') = \ln\left(\sum_{\boldsymbol{q}} P\{\boldsymbol{O}'|\boldsymbol{q}, \lambda\} P\{\boldsymbol{q}|\lambda\} \left(\frac{P\{\boldsymbol{O}'|\boldsymbol{q}, \lambda\} P\{\boldsymbol{q}|\lambda\}}{P\{\boldsymbol{q}|\boldsymbol{O}, \lambda\}}\right)\right) - \ln P\{\boldsymbol{O}|\lambda\}$$

$$D(\boldsymbol{O}, \boldsymbol{O}') = \ln\left(\sum_{\boldsymbol{q}} P\{\boldsymbol{q}|\boldsymbol{O}, \lambda\} \left(\frac{P\{\boldsymbol{O}'|\boldsymbol{q}, \lambda\} P\{\boldsymbol{q}|\lambda\}}{P\{\boldsymbol{q}|\boldsymbol{O}, \lambda\}}\right)\right) - \ln P\{\boldsymbol{O}|\lambda\}$$

$$D(\boldsymbol{O},\boldsymbol{O}') = \sum_{\boldsymbol{q}} P\{\boldsymbol{q}|\boldsymbol{O},\lambda\} \ln\left(\frac{P\{\boldsymbol{O}'|\boldsymbol{q},\lambda\}P\{\boldsymbol{q}|\lambda\}}{P\{\boldsymbol{q}|\boldsymbol{O},\lambda\}}\right) - \ln P\{\boldsymbol{O}|\lambda\}. \tag{2.27}$$

Como temos que $\sum_{\boldsymbol{q}} P\{\boldsymbol{q}|\boldsymbol{O},\lambda\} = 1$, segue que $\ln P\{\boldsymbol{O}|\lambda\} = \left(\sum_{\boldsymbol{q}} P\{\boldsymbol{q}|\boldsymbol{O},\lambda\}\right) \ln P\{\boldsymbol{O}|\lambda\}$. Então, é possível desenvolver (2.27) como

$$D(\boldsymbol{O},\boldsymbol{O}') \geq \sum_{\boldsymbol{q}} P\{\boldsymbol{q}|\boldsymbol{O},\lambda\} \ln\left(\frac{P\{\boldsymbol{O}'|\boldsymbol{q},\lambda\}P\{\boldsymbol{q}|\lambda\}}{P\{\boldsymbol{q}|\boldsymbol{O},\lambda\}P\{\boldsymbol{O}|\lambda\}}\right). \tag{2.28}$$

Por (2.23), pode-se escrever

$$L(\boldsymbol{O}') \geq L(\boldsymbol{O}) + \sum_{\boldsymbol{q}} P\{\boldsymbol{q}|\boldsymbol{O},\lambda\} \ln\left(\frac{P\{\boldsymbol{O}'|\boldsymbol{q},\lambda\}P\{\boldsymbol{q}|\lambda\}}{P\{\boldsymbol{q}|\boldsymbol{O},\lambda\}P\{\boldsymbol{O}|\lambda\}}\right). \tag{2.29}$$

Definindo

$$F(\boldsymbol{O},\boldsymbol{O}') \triangleq L(\boldsymbol{O}) + \sum_{\boldsymbol{q}} P\{\boldsymbol{q}|\boldsymbol{O},\lambda\} \ln\left(\frac{P\{\boldsymbol{O}'|\boldsymbol{q},\lambda\}P\{\boldsymbol{q}|\lambda\}}{P\{\boldsymbol{q}|\boldsymbol{O},\lambda\}P\{\boldsymbol{O}|\lambda\}}\right). \tag{2.30}$$

pode-se reescrever (2.29) como

$$L(\boldsymbol{O}') \geq F(\boldsymbol{O},\boldsymbol{O}'). \tag{2.31}$$

Por (2.31) é possível verificar que, pelo fato de $F(\boldsymbol{O},\boldsymbol{O}')$ estar limitada superiormente por $L(\boldsymbol{O}')$, se aumentar-se o valor de $F(\boldsymbol{O},\boldsymbol{O}')$, o valor de $L(\boldsymbol{O}')$ também será aumentado. Assim, o algoritmo deve selecionar $\boldsymbol{O}'$ tal que $F(\boldsymbol{O},\boldsymbol{O}')$ seja maximizado.

No entanto, observando (2.30), pode-se definir uma função $Q(\boldsymbol{O},\boldsymbol{O}')$ que representa a parcela de $F(\boldsymbol{O},\boldsymbol{O}')$ a ser maximizada, isto é, deixando à parte os termos de $F(\boldsymbol{O},\boldsymbol{O}')$ que são constantes em relação a $\boldsymbol{O}'$. Dessa forma:

$$Q(\boldsymbol{O},\boldsymbol{O}') \triangleq \sum_{\mathbf{q}} \mathrm{P}\{\mathbf{q}|\mathbf{O},\lambda\}\ln(\mathrm{P}\{\mathbf{O}'|\mathbf{q},\lambda\}\mathrm{P}\{\mathbf{q}|\lambda\}). \tag{2.32}$$

$$Q(\boldsymbol{O},\boldsymbol{O}') = \sum_{\mathbf{q}} \mathrm{P}\{\mathbf{q}|\mathbf{O},\lambda\}\ln\left(\frac{\mathrm{P}\{\mathbf{O}',\mathbf{q},\lambda\}}{\mathrm{P}\{\mathbf{q},\lambda\}}\frac{\mathrm{P}\{\mathbf{q},\lambda\}}{\mathrm{P}\{\lambda\}}\right)$$

$$Q(\boldsymbol{O},\boldsymbol{O}') = \sum_{\boldsymbol{q}} P\{\boldsymbol{q}|\boldsymbol{O},\lambda\}\ln\left(\frac{P\{\boldsymbol{O}',\boldsymbol{q},\lambda\}}{P\{\lambda\}}\right)$$

$$Q(\boldsymbol{O},\boldsymbol{O}') = \sum_{\boldsymbol{q}} P\{\boldsymbol{q}|\boldsymbol{O},\lambda\}\ln P\{\boldsymbol{O}',\boldsymbol{q}|\lambda\}$$

$$Q(\boldsymbol{O},\boldsymbol{O}') = \sum_{\boldsymbol{q}} \frac{P\{\boldsymbol{q},\boldsymbol{O}|\lambda\}}{P\{\boldsymbol{O}|\lambda\}}\ln P\{\boldsymbol{O}',\boldsymbol{q}|\lambda\}$$

$$Q(\boldsymbol{O},\boldsymbol{O}') = \frac{1}{P\{\boldsymbol{O}|\lambda\}}\sum_{\boldsymbol{q}} P\{\boldsymbol{q},\boldsymbol{O}|\lambda\}\ln P\{\boldsymbol{O}',\boldsymbol{q}|\lambda\}. \tag{2.33}$$

Como o termo $\frac{1}{P\{\boldsymbol{O}|\lambda\}}$ é constante em relação a $\boldsymbol{O}'$, é comum aproximar $Q(\boldsymbol{O},\boldsymbol{O}')$ como

$$Q(\boldsymbol{O},\boldsymbol{O}') = \sum_{\boldsymbol{q}} P\{\boldsymbol{q},\boldsymbol{O}|\lambda\}\ln P\{\boldsymbol{O}',\boldsymbol{q}|\lambda\}. \tag{2.34}$$

Observando atentamente (2.34), é possível verificar que $Q(\boldsymbol{O},\boldsymbol{O}')$ representa um cálculo de valor esperado. Dessa forma, cada iteração compreende duas etapas: expectativa (E), onde $Q(\boldsymbol{O},\boldsymbol{O}')$ é estimada em função dos parâmetros ocultos; maximização (M), na qual busca-se $\boldsymbol{O}'$ que maximize $Q(\boldsymbol{O},\boldsymbol{O}')$ [56]. Daí o nome do algoritmo: Expectativa-Maximização.

2.4.2.2 Maximização da probabilidade global

De acordo com (2.34), é possível definir uma função Q tal que

$$Q(\boldsymbol{O}, \boldsymbol{O}') = \sum_{\boldsymbol{q}} P\{\boldsymbol{q}, \boldsymbol{O}|\lambda\} \ln P\{\boldsymbol{O}', \boldsymbol{q}|\lambda\}. \tag{2.34}$$

Assim, através de manipulações algébricas, pode-se obter para $\ln P\{\boldsymbol{O}', \boldsymbol{q}|\lambda\}$ uma equação análoga à (2.21), fazendo com que possamos escrever (2.34) como a seguir [52]:

$$Q(\boldsymbol{O}, \boldsymbol{O}') = P\{\boldsymbol{O}|\lambda\}\left(-\frac{1}{2}\boldsymbol{O}'^{T}\overline{\boldsymbol{U}^{-1}}\boldsymbol{O}' + \boldsymbol{O}'^{T}\overline{\boldsymbol{U}^{-1}\boldsymbol{M}} + \overline{K}\right), \tag{2.35}$$

onde

$$\overline{\boldsymbol{U}^{-1}} = diag\left\{\overline{\boldsymbol{U}_1^{-1}}, \ldots, \overline{\boldsymbol{U}_N^{-1}}\right\}, \tag{2.36}$$

$$\overline{\boldsymbol{U}_N^{-1}} = \sum_{\boldsymbol{q}} P\{q_t = q|\boldsymbol{O}, \lambda\}\boldsymbol{U}_q^{-1}, \tag{2.37}$$

$$\overline{\boldsymbol{U}^{-1}\boldsymbol{M}} = \left[\overline{\boldsymbol{U}_1^{-1}\boldsymbol{\mu}_1}^{T} \quad \cdots \quad \overline{\boldsymbol{U}_N^{-1}\boldsymbol{\mu}_N}^{T}\right]^{T}, \tag{2.38}$$

e

$$\overline{\boldsymbol{U}_t^{-1}\boldsymbol{\mu}_t} = \sum_{\boldsymbol{q}} P\{q_t = q|\boldsymbol{O}, \lambda\}\, \boldsymbol{U}_q^{-1}\boldsymbol{\mu}_q, \tag{2.39}$$

e $\overline{K}$ é uma constante independente de $\boldsymbol{O}'$.

Analogamente a (2.15), $\boldsymbol{O}'$ pode ser reescrito da forma

$$\boldsymbol{O}' = \boldsymbol{W}\boldsymbol{C}' \tag{2.40}$$

onde $\boldsymbol{W}$ é o mesmo definido em (2.17) e $\boldsymbol{C}'$ corresponde aos coeficientes cepstrais que se relacionam com $\boldsymbol{O}'$, isto é,

$$\boldsymbol{C}' = [\boldsymbol{c}_1' \quad \cdots \quad \boldsymbol{c}_N'].$$

Daí, é possível obter $\boldsymbol{C'}$ que maximiza (2.34). Para isso, basta substituir (2.40) em (2.35):

$$Q(\boldsymbol{O}, \boldsymbol{O'}) = P\{\boldsymbol{O}|\lambda\}\left(-\frac{1}{2}\boldsymbol{C'^T W^T}\overline{\boldsymbol{U^{-1}}}\boldsymbol{W C'} + \boldsymbol{C'^T W^T}\overline{\boldsymbol{U^{-1}M}} + \overline{K}\right). \tag{2.41}$$

Derivando (2.41) em relação a $\boldsymbol{C'}$ e igualando a zero, obtemos (2.42) como uma equação linear para a maximização de $Q(\boldsymbol{O}, \boldsymbol{O'})$ [52].

$$\boldsymbol{W^T}\overline{\boldsymbol{U^{-1}}}\boldsymbol{W C'} = \boldsymbol{W^T}\overline{\boldsymbol{U^{-1}M}} \tag{2.42}$$

Desse modo, é possível, por fim, definir um algoritmo para a obtenção de $\boldsymbol{C'}$. O procedimento é resumido no fluxograma da Fig. 2.6.

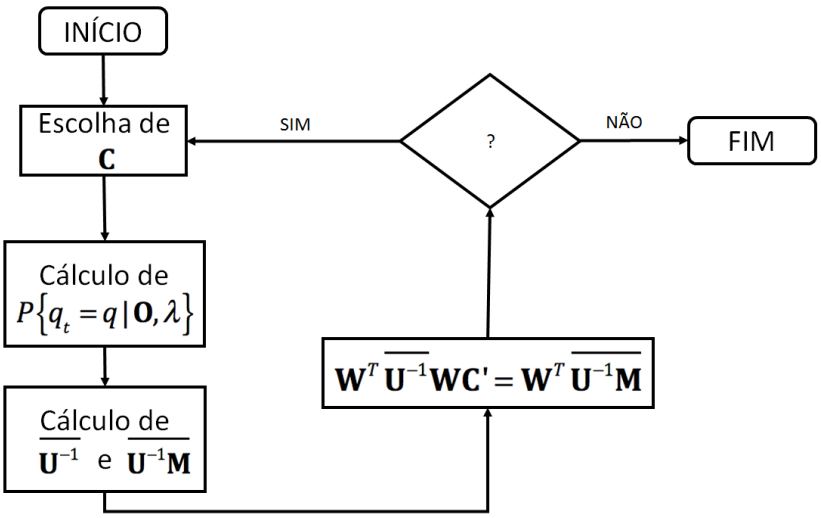


Fig. 2.6 Fluxograma do algoritmo de geração de parâmetros: "?" representa o critério de parada adotado

2.4.3 Consideração da variância global

Conforme é possível verificar através de (2.21) e (2.42), os algoritmos de geração de parâmetros apresentados consideram restrições explícitas entre as características estáticas e dinâmicas que, devido ao processo estatístico, geram parâmetros suavizados [57, 58]. Estes parâmetros, quando aplicados ao modelo fonte-filtro, acabam por produzir uma fala abafada [29]. A fim de resolver este problema, [59]

propôs uma variação para o algoritmo de geração de parâmetros baseada no critério de verossimilhança.

Nesse contexto, a variância global dos vetores estáticos de característica é definida em [59] como

$$\boldsymbol{v}(\boldsymbol{C}) = [v(1) \quad \cdots \quad v(M)]^T, \tag{2.43}$$

$$v(m) = \frac{1}{T}\sum_{t=1}^{N}\left(c_t(m) - \bar{c}(m)\right)^2 \tag{2.44}$$

e

$$\bar{c}(m) = \frac{1}{T}\sum_{\tau=1}^{N} c_\tau(m), \tag{2.45}$$

onde $\boldsymbol{C}$ é definido conforme (2.16), $\bar{c}(m)$ representa as médias dos componentes cepstrais e $v(m)$, suas variâncias.

Desse modo, o método proposto visa maximizar conjuntamente a probabilidade de observação de saída expressa em (2.21) e aquela relacionada à probabilidade de observação de saída da variância global. Isso é realizado através da maximização da variável *L* definida como

$$L = \log(P\{\boldsymbol{O}|\boldsymbol{q}, \lambda\}^{\varpi} \cdot P\{\boldsymbol{v}(\boldsymbol{C})|\lambda_v\}) \tag{2.46}$$

onde $P\{\boldsymbol{v}(\boldsymbol{C})|\lambda_v\}$ é modelado como uma única distribuição Gaussiana e λ_v consiste do seu conjunto de parâmetros, sendo $\boldsymbol{\mu}_v$ o vetor de médias e $\boldsymbol{\Sigma}_v$ a matriz de covariância para a variância global. A constante ϖ corresponde ao peso que balanceará $P\{\boldsymbol{O}|\boldsymbol{q}, \lambda\}$ e $P\{\boldsymbol{v}(\boldsymbol{C})|\lambda_v\}$. O modelo da gaussiana λ_v e o dos HMM λ são treinados de modo independente [57-59].

Para determinar $\boldsymbol{C}$ que maximiza *L*, é utilizado o método iterativo de gradiente,

$$\boldsymbol{C}^{(i+1)} = \boldsymbol{C}^{(i)} + \alpha \cdot \Delta\boldsymbol{C}^{(i)}, \tag{2.47}$$

onde α é o passo de atualização do algoritmo, $\boldsymbol{C}^{(i)}$ e $\Delta\boldsymbol{C}^{(i)}$ representam, respectivamente, $\boldsymbol{C}$ e $\Delta\boldsymbol{C}$ após a i-ésima iteração. Utilizando o algoritmo *steepest-descent* [98] com a primeira derivada, $\Delta\boldsymbol{C}^{(i)}$ pode ser obtida como

$$\Delta\boldsymbol{C}^{(i)} = \frac{\partial L}{\partial \boldsymbol{C}}\bigg|_{\boldsymbol{C}=\boldsymbol{C}^{(i)}}, \tag{2.48}$$

com

$$\frac{\partial L}{\partial \boldsymbol{C}} = \varpi(-\boldsymbol{W}^T\boldsymbol{U}^{-1}\boldsymbol{W}\boldsymbol{C} + \boldsymbol{W}^T\boldsymbol{U}^{-1}\boldsymbol{M}) + \begin{bmatrix}\boldsymbol{v}_1'^T & \cdots & \boldsymbol{v}_N'^T\end{bmatrix}^T, \tag{2.49}$$

$$\boldsymbol{v}_t' = [v_t'(1) \quad \cdots \quad v_t'(M)]^T \tag{2.50}$$

e

$$v_t'(m) = -\frac{2}{T}\boldsymbol{u}_v^{(m)^T}(\boldsymbol{v}(\boldsymbol{C}) - \boldsymbol{\mu}_v)\big(c_t(m) - \bar{c}(m)\big), \tag{2.51}$$

onde $\boldsymbol{u}_v^{(m)}$ representa o m-ésimo vetor de $\boldsymbol{\Sigma}_v^{-1}$.

Em [59], é proposta ainda uma segunda solução para a obtenção de $\Delta\boldsymbol{C}^{(i)}$ através do método de Newton-Raphson. Todavia, em função da elevada complexidade computacional, uma vez que faz uso de matrizes Hessianas no cálculo de suas segundas derivadas, este algoritmo não será apresentado neste trabalho.

Vale, ainda, ressaltar que testes subjetivos realizados em [57] demonstram que o algoritmo considerando a variância global de parâmetros gera melhorias no que diz respeito à naturalidade da fala sintetizada.

2.5 Conclusão

Este capítulo teve o intuito de apresentar a técnica dos HMM aplicada à síntese de voz. Para isso, inicialmente foi apresentada a arquitetura de um sistema TTS e foram discutidas as vantagens e desvantagens do método por HMM frente a outras técnicas utilizadas na atualidade. Na segunda parte, foi introduzido formalmente o conceito de HMM, bem como a nomenclatura utilizada nesta dissertação. Por fim, foram apresentados algoritmos que, através da técnica de HMM, realizam a geração

parâmetros que serão utilizados para a síntese da voz propriamente dita. Levando-se em consideração fatores como a naturalidade obtida e do menor custo computacional, o algoritmo apresentado na Seção 2.4.3 foi escolhido para a implementação realizada neste trabalho, que será melhor detalhada no Capítulo 3.

Capítulo 3

Treino com base de dados extensa para Português Brasileiro

3.1 Introdução

Motivada pelas inúmeras aplicações frente a um cenário globalizado e pelo consequente crescimento da importância do idioma, a síntese de voz para o português brasileiro tem alcançado significativos avanços nos últimos anos [25-27, 35, 39, 60-69]. O aspecto comum entre a maioria das contribuições é que se baseiam nas técnicas concatenativa e por formantes, as quais apresentam elevado nível de maturidade [28].

Recentemente, alguns trabalhos têm sido realizados utilizando a técnica por HMM [7, 28, 29, 48]. No entanto, mesmo com o desenvolvimento de tais sistemas, ainda não se obteve uma base de dados extensa e de elevada qualidade da qual os mesmos possam se beneficiar [34].

Deste modo, este capítulo terá como foco o sistema de síntese de voz baseado em HMM para o português brasileiro. Tal sistema se utiliza de uma base nova, extensa e de elevada qualidade, a qual foi desenvolvida através deste trabalho, em colaboração com [48, 70]. Portanto, a Seção 3.2 procura tratar a respeito das particularidades relacionadas a essa base, seu projeto e edição. A Seção 3.3 apresenta os aspectos relacionados à preparação dos dados que serão empregados na fase de treino. A Seção 3.4 descreve os parâmetros e as etapas envolvidas no treino propriamente dito.

3.2 A base de dados

3.2.1 Projeto e Gravação

A base de dados utilizada neste trabalho foi concebida em colaboração com [48, 70]. A mesma foi gravada conjuntamente com a base de reconhecimento de voz. A base de síntese de voz compreende 100 horas de gravação de um locutor do sexo feminino e de um locutor do sexo masculino, que foram escolhidos segundo critérios de

agradabilidade da voz. Aqui, foi empregada a parcela referente ao enunciador masculino.

Um sintetizador de voz deve ter a capacidade de reproduzir com naturalidade e inteligibilidade qualquer texto que se apresentar [5, 6]. Assim, faz-se necessária a gravação de uma base abrangente, visando englobar a maior variedade possível de aplicações. Logo, com o intuito de atender a essa necessidade, a base projetada inclui a gravação de frases de contexto jornalístico, instruções de GPS, previsão do tempo, textos históricos, biografias, romances, poemas, cordéis, crônicas, mensageiros web e estrangeirismos em inglês e francês. A descrição detalhada da escolha desse material pode ser encontrada em [70].

A gravação foi realizada em estúdio com equipamentos profissionais, conforme consta na Tab. 3-1, frequência de amostragem de 48 kHz e precisão de 24 bits/amostra.

Tab. 3-1 Equipamentos utilizados para gravação da base

Microfone	Neumann VA87
Pré-amplificador	Universal Audio 610-2
Conversor analógico-digital	Digi001
Software	Protools 10.0 (licenciado)
Computador	Mac Pro (2009)
Processador	Quad-Core Intel Xeon Series
Memória RAM	4 GB (PC3-8500, 1066 MHz)

3.2.2 Edição

A edição da base de dados foi realizada em estúdio através da ferramenta ProTools 10.0 (licenciado), mediante instruções prévias de técnico de som para seu uso. A edição consistiu da segmentação de toda a base, separando-a por frases, e a remoção de sons provenientes da abertura e fechamento de lábios, respiração, cliques, entre outros que não fazem parte da voz propriamente dita. Trata-se de um processo meticuloso e demorado, onde estima-se, por exemplo, que o tratamento de 15 minutos de áudio seja efetuado em 1 hora. Esta etapa foi realizada neste trabalho e em [99].

A interface de edição do ProTools é apresentada na Fig. 3.1. O processo de edição através dessa interface é descrita no Apêndice B.

A edição resultou num total de 11.456 frases segmentadas. O quantitativo de frases por conteúdo pode ser visualizado na Tab. 3-2.

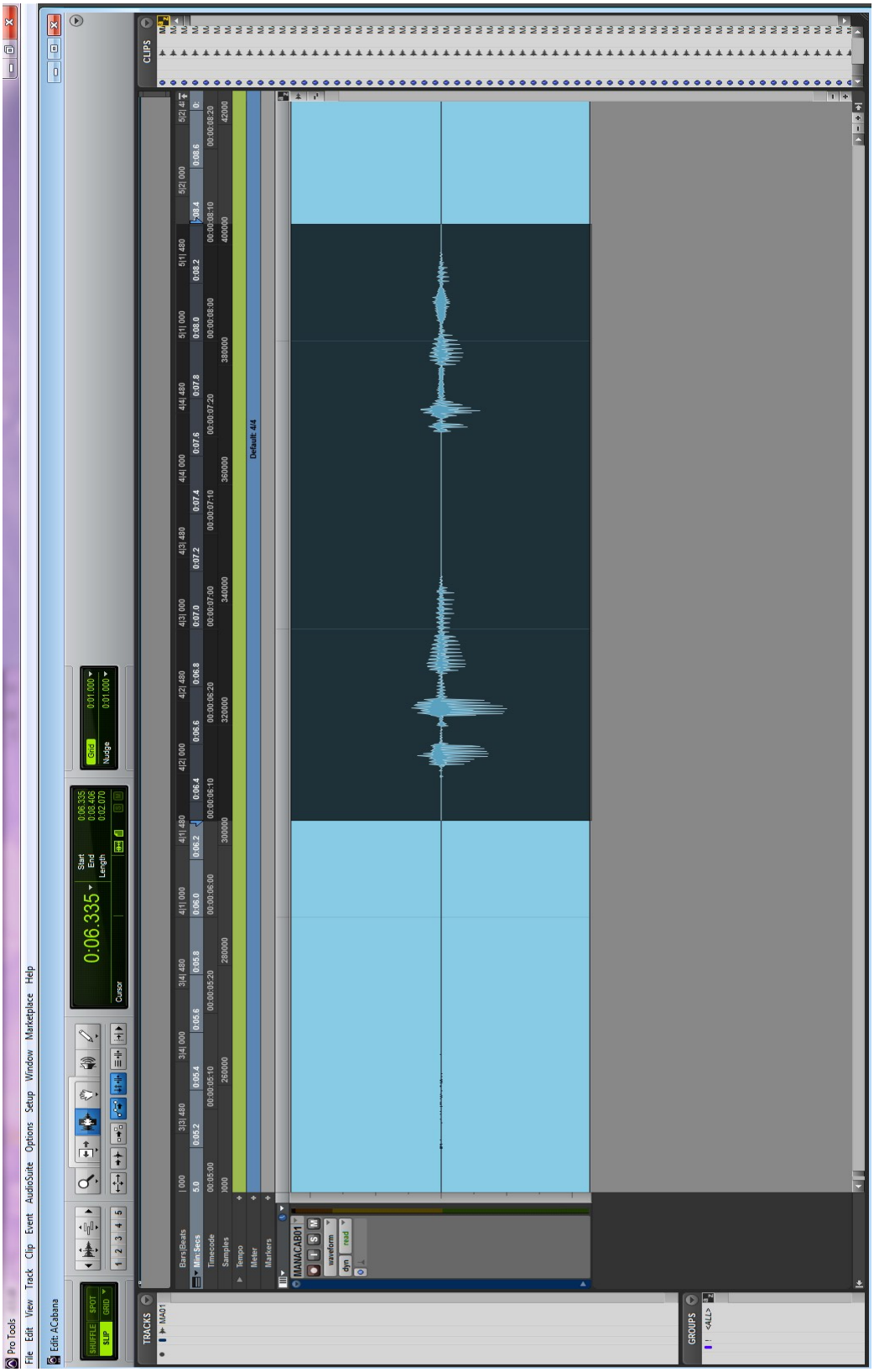


Fig. 3.1 Interface do ProTools 10.0 durante edição de áudio

Tab. 3-2 Quantitativo de frases por conteúdo

Conteúdo	Número de frases
A cabana	5.446
A assustadora história da medicina	119
Bíblia	108
Amor de gato	107
Comprometida	143
Fábulas de Esopo	143
Francês	102
Poesias – Fernando Pessoa	141
Poemas – Ferreira Gullar	35
GPS	111
MSN	114
Haicai	86
Crônicas de O Globo – Ferreira Gullar	222
Por um fio	170
Ser amigo	41
Traduzir-se	18
1808	681
A árvore generosa	61
Absurdo	34
Biografia de Da Vinci	10
Biografia de Monteiro Lobato	19
Como destruir seu casamento	156
A hora da estrela	526
Imã de geladeira	25
Inglês	551
Imaginação	24
Ironia	189
Jantar japonês	215
João Boa Morte	420
Nomes próprios em inglês	118
Números	85

Conteúdo	Número de frases
Previsão do tempo	13
Show do Fantástico	38
Sorrir	189
Translúcido	34
1000 frases	1.000

3.3 Preparação de dados

A Fig. 3.2 ilustra os passos necessários ao treinamento de HMM. Nela, é possível perceber que, para dar início a esta fase, são necessárias as formas de onda de fala (gravações) e as informações de pronúncia correspondentes. Estas últimas dizem respeito ao conhecimento básico para a fala sintetizada, isto é, são as informações mínimas a respeito da representação linguística do texto que gerará tal fala. Tais informações são compostas pela transcrição e classificação da palavra, separação silábica, indicação de sílaba tônica e transcrição fonética.

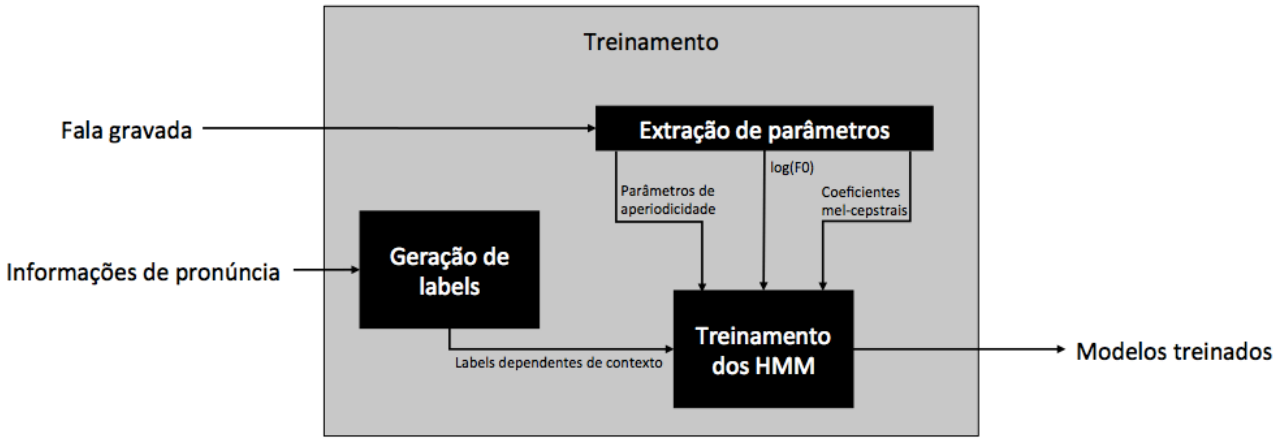


Fig. 3.2 Etapas de treinamento dos HMM

As informações requeridas são geradas através do módulo NLP, introduzido no Capítulo 2. A fim de produzi-las, o processamento de texto realizado neste módulo é dividido nas seguintes etapas, nesta sequência:

- Desambiguação de homógrafos;
- Marcação de tonicidade;
- Separação silábica;
- Conversão grafema-fone.

Esses passos são descritos brevemente nas próximas seções. Para uma visão mais aprofundada, consultar [48].

Vale dizer que antes da utilização destes módulos, é realizada uma normalização do texto[1], a fim de transcrever números e siglas que possam estar contidos no mesmo.

3.3.1 Desambiguação de homógrafos

A realização correta da transcrição fonética apresenta, dentre outras, a dificuldade de realizar a distinção entre homógrafos heterófonos[2]. Por mais que casos de homógrafos ocorram num percentual reduzido, a precisão na execução desta tarefa influencia diretamente a qualidade do sintetizador, afetando a naturalidade e inteligibilidade percebidas [71, 72].

Diversas soluções para essa questão têm sido propostas pela comunidade científica. Inicialmente estas soluções apenas tratavam o problema sob o ponto de vista morfossintático [65, 73], isto é, a desambiguação é realizada para homógrafos de classes gramaticais diferentes. Recentemente, métodos que abrangem também a abordagem semântica têm sido propostos [48, 71, 74]. A importância do tratamento semântico se dá em situações onde os homógrafos são de mesma classe gramatical.

O algoritmo empregado neste trabalho faz uso de um conjunto de 23 regras abrangendo 112 pares de homógrafos [48, 71], expostos na Tab. 3-3[3]. Tais regras são divididas de acordo com a análise: morfossintática (regras de 1 a 16) e semântica (regras 17 a 23). Dessa forma, o texto é inicialmente particionado em frases e palavras para que, através de consulta à biblioteca projetada, o homógrafo seja identificado e, então, seja empregada a regra correspondente ao seu tipo. Cada uma dessas regras possui perguntas contextuais a ela relacionada.

Como resultado da análise da desambiguação de homógrafos, o sistema gera um arquivo de texto com as ocorrências de cada um no texto. Um trecho deste arquivo é ilustrado na Fig. 3.3.

[1] Por texto-normalizado, entenda-se o texto com números, siglas e acrônimos transcritos em palavras.
[2] Palavras de escrita semelhante mas de pronúncias distintas.
[3] Nesta tabela as indicações [e] e [o] representam sons fechados e [E] e [O], sons abertos para as vogais "e" e "o", respectivamente. Essas representações serão melhor abordadas mais adiante.

Tab. 3-3 Regras aplicadas aos homógrafos

Regra	Alternância vocálica e oposição gramatical	Homógrafos
1	[e] Substantivo / [E] Verbo	acerto, apelo, aperto, apreço, começo, concerto, conserto, desemprego, desespero, emprego, enredo, erro, esmero, espeto, flagelo, gelo, governo, interesse, interesses, modelo, pego, peso, rego, selo, testo, zelo
2	[o] Substantivo / [O] Verbo	aborto, acordo, adorno, aforro, almoço, apoio, arrojo, arroto, choco, choro, conforto, consolo, contorno, controle, coro, desgosto, despojo, destroço, encosto, endosso, esforço, estorno, estorvo, folgo, gosto, jogo, logro, namoro, olho, piloto, reforço, rodo, rogo, rolo, sopro, suborno, sufoco, toco, toldo, topo, torno, troco, troço
3	[o] Substantivo / [O] Verbo	rola, rolha
4	[e] Substantivo / [E] Verbo	colher, meta
5	[e] Contração / [E] Verbo	desses, deste, destes
6	[o] Verbo / [O] Advérbio	fora
7	[e] Adjetivo ou Substantivo / [E] Verbo	seco, seca, secas
8	[o] Adjetivo ou Substantivo / [O] Verbo	boto
9	[e] Pronome demonstrativo / [E] Adjetivo ou Susbstantivo	este
10	[e] Verbo / [E] Adjetivo ou Substantivo	leste
11	[o] Preposição / [O] Verbo	sobre
12	[o] Adjetivo ou Verbo / [O] Substantivo	rota, rotas, tola, tolas
13	[o] Substantivo / [O] Verbo ou Substantivo	corte, cortes, forma, formas, molho, soco
14	[e] Preposição ou Substantivo / [E] Verbo	cerca
15	[e] Substantivo / [E] Verbo ou Substantivo	pega, pegas
16	[e] Contração ou Substantivo / [E] Verbo ou Substantivo	pelo, pela, pelas

Regra	Alternância vocálica e oposição gramatical	Homógrafos
17	[e] Substantivo / [E] Substantivo	besta, bestas
18	[e] Substantivo / [E] Substantivo	sede, sedes
19	[e] Substantivo / [E] Substantivo	medo, medos
20	[e] Substantivo ou Verbo / [E] Substantivo	termos
21	[o] Substantivo / [O] Substantivo	cor
22	[o] Substantivo / [O] Substantivo	lobo, lobos
23	[o] Substantivo / [O] Substantivo	bola, bolas

```
Arvore[1]->Ocorrencias  de acerto   =   0
Arvore[2]->Ocorrencias  de apelo    =   0
Arvore[3]->Ocorrencias  de aperto   =   0
Arvore[4]->Ocorrencias  de apreço   =   0
Arvore[5]->Ocorrencias  de começo   =   0
Arvore[6]->Ocorrencias  de concerto     =   0
Arvore[7]->Ocorrencias  de conserto     =   0
Arvore[8]->Ocorrencias  de desemprego   =   0
Arvore[9]->Ocorrencias  de desespero    =   0
Arvore[10]->Ocorrencias  de emprego     =   0
Arvore[11]->Ocorrencias  de enredo  =   0
Arvore[12]->Ocorrencias  de erro    =   0
Arvore[13]->Ocorrencias  de esmero  =   0
Arvore[14]->Ocorrencias  de espeto  =   0
Arvore[15]->Ocorrencias  de flagelo     =   0
Arvore[16]->Ocorrencias  de gelo    =   0
Arvore[17]->Ocorrencias  de governo     =   0
Arvore[18]->Ocorrencias  de interesse   =   0
Arvore[19]->Ocorrencias  de interesses  =   0
Arvore[20]->Ocorrencias  de modelo  =   0
Arvore[21]->Ocorrencias  de pego    =   0
Arvore[22]->Ocorrencias  de peso    =   0
Arvore[23]->Ocorrencias  de rego    =   0
Arvore[24]->Ocorrencias  de selo    =   0
Arvore[25]->Ocorrencias  de testo   =   0
Arvore[26]->Ocorrencias  de zelo    =   0
Arvore[27]->Ocorrencias  de aborto  =   0
```

Fig. 3.3 Trecho da análise de homógrafos da frase "Pesquisa é uma coisa que muda a toda hora."

3.3.2 Marcação de tonicidade

A tonicidade silábica representa a sílaba pronunciada mais fortemente em uma palavra. Esta característica contribui para a elevação da frequência fundamental, intensidade e duração da vogal [48, 75]. Além disso, influencia também o ritmo da fala

pela transição entre sílabas tônicas e átonas [76]. A observação desta característica contribui para a naturalidade da fala sintetizada.

Para a implementação desta funcionalidade, é realizada a análise de sequências caracteres, levando em consideração as regras de acentuação gráfica do Português Brasileiro. Essa análise se dá através de 19 regras descritas em [48], dispostas hierarquicamente. Estas regras são postas de modo mais inteligível na Tab. 3-4.

Vale ressaltar que, palavras classificadas como homógrafos não são analisadas, uma vez que sua definição de tonicidade já acontece no algoritmo apresentado na Seção 3.3.1.

Este módulo gera um arquivo de saída com a marcação das sílabas tônicas, conforme o ilustrado na Fig. 3.4.

```
pesqu[i]sa é uma c[o]isa que m[u]da a t[o]da h[o]ra.
```

Fig. 3.4 Marcação de sílaba tônica para a frase "Pesquisa é uma coisa que muda a toda hora."

Tab. 3-4 Regras para determinação da tonicidade

Regra	Descrição
1	Caso exista acento, a vogal é marcada como tônica. O acento agudo e o circunflexo tem precedência sobre o til. Exemplo: órfão.
2	Palavras terminadas em “r”, “l”, “z”, “x” ou “n”. A vogal tônica é a penúltima letra. Exemplo: feroz.
3	Palavras terminadas em “im”, “om” ou “um”. A vogal tônica será a penúltima letra. Exemplo: quindim.
4	Palavras terminadas em “ins”, “ons” ou “uns” (plural da regra 3). A vogal tônica será a antepenúltima. Exemplo: quindins.
5	Palavras terminadas em “qui”, “qüi”, “gui” ou “güi”. A vogal tônica será a última. Exemplo: aqui.
6	Palavras terminadas em “quis”, “qüis”, “guis” ou “güis” (plural da regra 5). A vogal tônica será a antepenúltima. Exemplo: caquis.
7	Palavras terminadas em “i” ou “u”. Caso penúltima letra seja vogal diferente de “u”, esta será a tônica. Caso contrário, a última letra será a tônica. Exemplo: caiu.
8	Palavras terminadas em “is” ou “us” e antepenúltima letra é vogal, esta será a tônica. Exemplo: degraus.
9	Palavras terminadas em “is” ou “us” e antepenúltima letra não é vogal. A tônica será a penúltima letra. Exemplo: javalis.
10	Palavras terminadas em “porque”. A tônica será a última letra. Exemplo: porque.
11	Palavras terminadas em “que” ou “gue”. Se a letra que precede essa terminação for vogal, ela será a tônica. Caso contrário, a letra que precede esta última será a vogal tônica. Exemplos: bosque, justifique.
12	Palavras terminadas em “ques” ou “gues”. Se a letra que precede essa terminação for vogal, ela será a tônica. Caso contrário, a letra que precede esta última será a vogal tônica. (plural da regra 11) Exemplos: bosques, justifiques.
13	Palavras terminadas com a sequência: vogal + “i” + vogal. A vogal tônica será a antepenúltima letra. Exemplo: correio.
14	Palavras terminadas com a sequência: letra diferente de “q” ou “g” + vogal + “i” ou “u” + consoante + vogal. A vogal tônica será a anterior à antepenúltima. Exemplos: mangueira, doido.
15	Palavras terminadas com a sequência: letra diferente de “q” ou “g” + vogal + “i” ou “u” + consoante + vogal + “s”. A vogal tônica será a anterior à antepenúltima. (plural da regra 14) Exemplos: mangueiras, doidos.
16	Palavras terminadas com a sequência: vogal + "in" ou “un” + consoante + “a”, “e” ou “o”. A vogal tônica será a anterior à antepenúltima. Exemplos: saindo, oriundo.
17	Palavras com penúltima vogal sendo “i” ou “u” sucedido de consoante e precedido de vogal precedida de letra diferente de “q” ou “g”. A vogal tônica será aquela que precede o “i” ou “u”. Exemplos: fausto.
18	Palavras terminadas em “quem”. A vogal tônica será a penúltima. Exemplo: quem.
19	A vogal tônica será a penúltima vogal da palavra. Exemplos: guerra.

3.3.3 Separação silábica

Por se tratar de um idioma *syllable-timed*, isto é, cuja unidade responsável pela marcação do ritmo da fala é a sílaba, a separação silábica se revela de especial

relevância para o caso do Português Brasileiro [77]. Logo, este é também um dos fatores que influenciam a prosódia e, portanto, a naturalidade da fala sintetizada.

Assim, o algoritmo implementado para separação silábica visa realizar a análise por caracteres de cada palavra. O mesmo leva em consideração o fato de cada sílaba ter como base uma vogal [78]. Vale ressaltar que este método analisa a expressão fonética das palavras de maneira que esta precede sobre a ortografia. De maneira semelhante aos algoritmos anteriormente apresentados, este dispõe de 25 regras dispostas hierarquicamente para proceder a separação silábica.

A Fig. 3.5 mostra a saída deste módulo para uma frase de exemplo.

```
pesquisa é uma coisa que muda a toda hora pes qui sa é u ma coi sa que mu da a to da ho ra
```

Fig. 3.5 Separação silábica para a frase "Pesquisa é uma coisa que muda a toda hora."

3.3.4 Conversão grafema-fone

A conversão grafema-fone (G2P) também é conhecida como transcrição fonética das palavras. A realização desta etapa conclui a etapa de NLP, tendo especial importância por ser o primeiro passo para a geração dos rótulos contribuintes para a concatenação dos modelos.

Para proceder a transcrição fonética, um alfabeto apropriado deve ser empregado. Neste caso, o escolhido foi o SAMPA (*Speech Assessment Methods Phonetic Alphabet*) [79], que é ilustrado na Tab. 3-5.

O algoritmo G2P utilizado realiza a conversão através de regras posicionadas em graduação. Na maioria dos casos, as regras são acionadas a cada letra detectada. Todavia, há casos onde essa conversão resulta em saltos de grafemas, como, por exemplo, na palavra *pulam* cujo "a" recai em uma regra que desconsidera a transcrição do "m" posterior. Além disso, há situações em que um só grafema pode ser traduzido como uma sequência de dois fones, como o "x" que se torna na sequência [k s] em *oxítona.*

Vale, ainda, ressaltar que o algoritmo não considera ditongos como unidades acústicas independentes (caso do sistema de síntese de voz Festival [80]), mas os trata como uma sequência de difones, como é o caso de [a w] no ditongo formado pela sequência "al" na palavra *original*.

Ao final da transcrição, o G2P gera um arquivo *.utt com informações por fone a respeito da sílaba em que está inserido, a tonicidade, a palavra e a classe desta última. O formato deste arquivo é ilustrado na Fig. 3.6. As informações nele contidas serão utilizadas no módulo de treinamento conforme veremos a seguir.

Tab. 3-5 Alfabeto fonético SAMPA

Fones	Exemplos
Vogais orais	
a	lápis, ábaco, cabeça, caça
E	é, médico, pele, ferro, velho
e	capacete, resolver
i	justiça, país, ele, pele
O	ópio, cópia, jogos, sozinho
o	jogo, bolo, cor
u	baiacu, Raul, cururu
Vogais nasais	
a~	avião, andar, cama
e~	então, consciência, bem
i~	ninho, tinta, importa
o~	onda, homem, fronha
u~	um, muito, umbigo
Semi-vogais	
w	Natal, eu, quase
j	fui, pai, sei, foi
w~	não, cão
j~	muito, bem, parabéns
Fricativas não vozeadas	
f	festa, afta, fanfarrão
s	sapo, lápis, tórax, capaz, desço
S	chá, xaveco, cachorro
Fricativas vozeadas	
z	casa, quase, exato
v	vovó, avião, vamos
Z	geladeira, trovejar
Plosivas	

Fones	Exemplos
b	barba, absinto
d	dados, cidade, administrar
t	todos, constituinte
k	casa, quero, quanto
g	guerra, gato, aguentar, agnóstico
p	papai, apto
Líquidas	
l	laranja, leitão
L	calhar, colheita, melhor
R	carro, rua, carga
X	casar, certo, arpa
r	carona, frango, por exemplo
Consoantes nasais	
m	mamãe, marmota
n	nome, atenuar, encanação
J	casinha, galinha

```
phone       syll    stress      word        class
sil
p           pes       0         peskiza     content
e
s
k           ki        1
i
z           za        0
a
E           Eu~       0         Eu~ma       content
u~
m           ma        0
a
k           koj       0         kojza       function
o
j
z           za        0
a
k           ki        1         ki          content
i
m           mu        0         muda        function
u
d           da        0
a
a           a         1         a           content
t           to        0         toda        function
o
d           da        0
a
o           o         1         ora         content
r           ra        0
a
sil
sil
sil
sil
```

Fig. 3.6 Rótulos gerados pela conversão grafema-fonema para a sentença "Pesquisa é uma coisa que muda a toda hora."

3.4 Treinamento

Conforme ilustrado na Fig. 3.2, a realização do treinamento envolve três etapas básicas: geração de *labels*, extração de parâmetros da fala e o treinamento propriamente dito dos HMM. Cada uma dessas fases é descrita a seguir. Detalhes práticos sobre o procedimento de treino estão no Apêndice C

3.4.1 Geração de *labels*

Nesta etapa, as informações de fala obtidas do NLP são convertidas em *labels* contextuais para HMM. Estes *labels* contém informações dependentes de contexto que são de especial relevância para a reprodução natural da prosódia.

Para a determinação desses fatores são considerados dados teóricos a respeito do idioma como também análise empírica para ajuste de características [10]. Assim, fatores considerados importantes para o idioma podem ser incluídos e outros de pequena relevância excluídos. Os fatores contextuais empregados neste trabalho são listados na Tab. 3-6. Os mesmos são derivados de [55] e adaptados conforme [10].

Desse modo, para cada fone da elocução, haverá uma *label* correspondente com todo o conjunto de informações contextuais, expressas na Tab. 3-6, relacionadas a ele. A representação dessas *labels* possui formato próprio que é utilizado para introduzir as informações em arquivos *lab*, conforme Fig. 3.7 [29]. Estes arquivos serão utilizados como entrada no módulo de árvores contextuais no treino dos HMM.

Ademais, para a melhor estimativa dessas informações, os arquivos com as transcrições fonéticas foram ajustados manualmente antes do treinamento para corrigir eventuais erros de algoritmo persistentes da fase de NLP.

Tab. 3-6 Informações nos arquivos de *label*

nível de fone	fone {pré-anterior, anterior, atual, seguinte, pós-seguinte}
	posição do fone na sílaba atual
nível de sílaba	quando ou não sílaba {anterior, atual, seguinte} é tônica
	número de fones na sílaba {anterior, atual, seguinte}
	posição da sílaba atual na palavra atual
	número de sílabas tônicas na frase atual {antes, depois} da sílaba atual
	número de sílabas (contando da anterior tônica até a sílaba atual)
	número de sílabas (contando da atual até a próxima tônica)
nível de palavra	classificação POS da palavra {anterior, atual, seguinte}
	número de sílabas na palavra {anterior, atual, seguinte}
	posição da palavra atual na frase atual
	número de palavras de conteúdo na frase atual {antes, depois} da palavra atual
	número de palavras contando da palavra de conteúdo anterior até a palavra atual
	número de palavras contando da atual até a próxima palavra de conteúdo
nível de frase	número de {sílabas, palavras} na frase {anterior, atual, seguinte}
	posição da frase atual na elocução atual
nível de elocução	número de {sílabas, palavras, frases} na expressão

```
y^y-sil+p=e/M2:y_y/U:15_$8_&1
y^sil-p+e=s/M2:1_3/S1:y_@y-0_@3+1_@2/S2:1_3/S3:1_15/S4:0_4/S5:0_2/S6:e/W1:y_#y-content_#3+content_#2/W2:1_8/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
sil^p-e+s=k/M2:2_2/S1:y_@y-0_@3+1_@2/S2:1_3/S3:1_15/S4:0_4/S5:0_2/S6:e/W1:y_#y-content_#3+content_#2/W2:1_8/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
p^e-s+k=i/M2:3_1/S1:y_@y-0_@3+1_@2/S2:1_3/S3:1_15/S4:0_4/S5:0_2/S6:e/W1:y_#y-content_#3+content_#2/W2:1_8/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
e^s-k+i=z/M2:1_2/S1:0_@3-1_@2+0_@2/S2:2_2/S3:2_14/S4:0_3/S5:0_7/S6:i/W1:y_#y-content_#3+content_#2/W2:1_8/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
s^k-i+z=a/M2:2_1/S1:0_@3-1_@2+0_@2/S2:2_2/S3:2_14/S4:0_3/S5:0_7/S6:i/W1:y_#y-content_#3+content_#2/W2:1_8/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
k^i-z+a=E/M2:1_2/S1:1_@2-0_@2+0_@2/S2:3_1/S3:3_13/S4:1_3/S5:2_6/S6:a/W1:y_#y-content_#3+content_#2/W2:1_8/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
i^z-a+E=u~/M2:2_1/S1:1_@2-0_@2+0_@2/S2:3_1/S3:3_13/S4:1_3/S5:2_6/S6:a/W1:y_#y-content_#3+content_#2/W2:1_8/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
z^a-E+u~=m/M2:1_2/S1:y_@y-0_@2+0_@2/S2:1_2/S3:4_12/S4:1_3/S5:3_5/S6:u~/W1:content_#3-content_#2+function_#2/W2:2_7/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
a^E-u~+m=a/M2:2_1/S1:y_@y-0_@2+0_@2/S2:1_2/S3:4_12/S4:1_3/S5:3_5/S6:u~/W1:content_#3-content_#2+function_#2/W2:2_7/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
E^u~-m+a=k/M2:1_2/S1:0_@2-0_@2+0_@3/S2:2_1/S3:5_11/S4:1_3/S5:4_4/S6:a/W1:content_#3-content_#2+function_#2/W2:2_7/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
u~^m-a+k=o/M2:2_1/S1:0_@2-0_@2+0_@3/S2:2_1/S3:5_11/S4:1_3/S5:4_4/S6:a/W1:content_#3-content_#2+function_#2/W2:2_7/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
m^a-k+o=j/M2:1_3/S1:y_@y-0_@3+0_@2/S2:1_2/S3:6_10/S4:1_3/S5:5_3/S6:o/W1:content_#2-function_#2+content_#1/W2:3_6/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
a^k-o+j=z/M2:2_2/S1:y_@y-0_@3+0_@2/S2:1_2/S3:6_10/S4:1_3/S5:5_3/S6:o/W1:content_#2-function_#2+content_#1/W2:3_6/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
k^o-j+z=a/M2:3_1/S1:y_@y-0_@3+0_@2/S2:1_2/S3:6_10/S4:1_3/S5:5_3/S6:o/W1:content_#2-function_#2+content_#1/W2:3_6/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
o^j-z+a=k/M2:1_2/S1:0_@3-0_@2+1_@2/S2:2_1/S3:7_9/S4:1_3/S5:6_2/S6:a/W1:content_#2-function_#2+content_#1/W2:3_6/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
j^z-a+k=i/M2:2_1/S1:0_@3-0_@2+1_@2/S2:2_1/S3:7_9/S4:1_3/S5:6_2/S6:a/W1:content_#2-function_#2+content_#1/W2:3_6/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
z^a-k+i=m/M2:1_2/S1:y_@y-1_@2+0_@2/S2:1_1/S3:8_8/S4:1_2/S5:7_4/S6:i/W1:function_#2-content_#1+function_#2/W2:4_5/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
a^k-i+m=u/M2:2_1/S1:y_@y-1_@2+0_@2/S2:1_1/S3:8_8/S4:1_2/S5:7_4/S6:i/W1:function_#2-content_#1+function_#2/W2:4_5/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
k^i-m+u=d/M2:1_2/S1:y_@y-0_@2+0_@2/S2:1_2/S3:9_7/S4:2_2/S5:2_3/S6:u/W1:content_#1-function_#2+content_#1/W2:5_4/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
i^m-u+d=a/M2:2_1/S1:y_@y-0_@2+0_@2/S2:1_2/S3:9_7/S4:2_2/S5:2_3/S6:u/W1:content_#1-function_#2+content_#1/W2:5_4/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
m^u-d+a=a/M2:1_2/S1:0_@2-0_@2+1_@1/S2:2_1/S3:10_6/S4:2_2/S5:3_2/S6:a/W1:content_#1-function_#2+content_#1/W2:5_4/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
u^d-a+a=t/M2:2_1/S1:0_@2-0_@2+1_@1/S2:2_1/S3:10_6/S4:2_2/S5:3_2/S6:a/W1:content_#1-function_#2+content_#1/W2:5_4/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
d^a-a+t=o/M2:1_1/S1:y_@y-1_@1+0_@2/S2:1_1/S3:11_5/S4:2_1/S5:4_4/S6:a/W1:function_#2-content_#1+function_#2/W2:6_3/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
a^a-t+o=d/M2:1_2/S1:y_@y-0_@2+0_@2/S2:1_2/S3:12_4/S4:3_1/S5:2_3/S6:o/W1:content_#1-function_#2+content_#2/W2:7_2/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
a^t-o+d=a/M2:2_1/S1:y_@y-0_@2+0_@2/S2:1_2/S3:12_4/S4:3_1/S5:2_3/S6:o/W1:content_#1-function_#2+content_#2/W2:7_2/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
t^o-d+a=o/M2:1_2/S1:0_@2-0_@2+1_@1/S2:2_1/S3:13_3/S4:3_1/S5:3_2/S6:a/W1:content_#1-function_#2+content_#2/W2:7_2/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
o^d-a+o=r/M2:2_1/S1:0_@2-0_@2+1_@1/S2:2_1/S3:13_3/S4:3_1/S5:3_2/S6:a/W1:content_#1-function_#2+content_#2/W2:7_2/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
d^a-o+r=a/M2:1_1/S1:y_@y-1_@1+0_@2/S2:1_2/S3:14_2/S4:3_0/S5:4_0/S6:o/W1:function_#2-content_#2+y_#y/W2:8_1/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
a^o-r+a=sil/M2:1_2/S1:1_@1-0_@2+y_@y/S2:2_1/S3:15_1/S4:4_0/S5:2_0/S6:a/W1:function_#2-content_#2+y_#y/W2:8_1/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
o^r-a+sil=y/M2:2_1/S1:1_@1-0_@2+y_@y/S2:2_1/S3:15_1/S4:4_0/S5:2_0/S6:a/W1:function_#2-content_#2+y_#y/W2:8_1/W3:0_0/W4:0_0/P1:y_!y-15_!8+y_!y/P2:1_1/U:15_$8_&1
r^a-sil+sil=sil/M2:y_y/U:15_$8_&1
a^sil-sil+sil=sil/M2:y_y/U:15_$8_&1
sil^sil-sil+sil=y/M2:y_y/U:15_$8_&1
sil^sil-sil+y=y/M2:y_y/U:15_$8_&1
```

Fig. 3.7 Exemplo de arquivo *lab* para a frase "Pesquisa é uma coisa que muda a toda hora."

3.4.2 Extração de parâmetros de fala

Nesta etapa são extraídas as seguintes informações a partir das frases gravadas no banco de treino:

- Logaritmos das frequências fundamentais $\{\log(F0_1), \dots, \log(F0_N)\}$ obtidas numa base *short-time*, onde *N* é o número total de *frames* de todas as elocuções presentes no banco de treino. Caso $F0 = 0$, o sinal é considerado não vozeado.
- Vetores de coeficientes mel-cepstrais, que representam o envelope espectral da fala, $\{c_1, \dots, c_N\}$ [81], onde c_i é um vetor coluna M-dimensional e i representa o número do frame. Para a obtenção destes parâmetros, é considerada a sequência de $\log(F0)$ para remover a periodicidade do sinal [82].
- Sequência de vetores de coeficientes de aperiodicidade, $\{d_1, \dots, d_N\}$, onde d_ié um vetor coluna de dimensão 5 [10].

Os procedimentos para a extração destes parâmetros são descritos em [82].

3.4.3 Treinamento dos HMM

3.4.3.1 Observações

Para o treinamento dos HMM, foram utilizados HMM de *S* estados sem pulos da esquerda para a direita, conforme modelo apresentado na Fig. 2.5. O vetor de saída para o i-ésimo *frame* é dado por $\boldsymbol{o}_i = [\boldsymbol{o}_{i1}^{\boldsymbol{T}} \quad \cdots \quad \boldsymbol{o}_{i5}^{\boldsymbol{T}}]^T$, onde:

- Vetor o_{i1}: composto dos coeficientes mel-cepstrais $\{c_{i0}, \dots, c_{iM}\}$, os componentes delta $\{\Delta c_{i0}, \dots, \Delta c_{iM}\}$ e delta-delta $\{\Delta^2 c_{i0}, \dots, \Delta^2 c_{iM}\}$ correspondentes, conforme uma distribuição de probabilidades contínua.
- Vetores o_{i2}, o_{i3} e o_{i4}: compostos do logaritmo da frequência fundamental, $\log(F0_i)$, os componentes delta $\Delta \log(F0_i)$ e delta-delta $\Delta^2 \log(F0_i)$ correspondentes, cada um modelado por uma distribuição de probabilidades multiespaço.
- Vetor o_{i5}: composto dos coeficientes de aperiodicidade $\{d_{i1}, \dots, d_{i5}\}$, os componentes delta $\{\Delta d_{i1}, \dots, \Delta d_{i5}\}$ e delta-delta $\{\Delta^2 d_{i1}, \dots, \Delta^2 d_{i5}\}$ correspondentes, de acordo com uma distribuição de probabilidades contínua.

A sequência de observações $\boldsymbol{O} = \{\boldsymbol{o}_1, \dots, \boldsymbol{o}_N\}$ é gerada conforme o exposto no fluxograma da Fig. 3.8. Nela, tem-se que a observação $\boldsymbol{o}_i$ é obtida através da

distribuição de probabilidades de saída $b_{s_j}(\boldsymbol{o}_i)$ no estado $q_i = s_j$. Esta distribuição é dada por

$$b_{s_j}(\boldsymbol{o}_i) = \prod_{k=1}^{K}\left[\sum_{l=1}^{L_k} w_{s_jkl}\mathcal{N}\left(\boldsymbol{o}_{ik}; \boldsymbol{\mu}_{s_jkl}, \boldsymbol{\Sigma}_{s_jkl}\right)\right]^{\gamma_k}, \tag{3.1}$$

onde:

- K é o número de vetores de componentes de $\boldsymbol{o}_i$. Neste caso $K = 5$;
- L_k é o número de componentes da mistura para o k-ésimo componente de o_i. Neste caso $L_1 = L_5 = 1$, já que o_{i1} e o_{i5} são modelados por distribuições Gaussianas contínuas de uma mistura, e $L_2 = L_3 = L_4 = 2$, pois o_{i2}, o_{i3} e o_{i4} possuem modelos como distribuições Gaussiadas multiespaço com duas componentes [84];
- γ_k é o peso para o k-ésimo componente de o_i;
- w_{s_jkl} é o peso do l-ésimo componente da mistura para o k-ésimo componente de o_i;
- $\mathcal{N}(\cdot; \mu, \Sigma)$ é uma distribuição Gaussiana multivariada de vetor de médias μ e matriz de covariâncias Σ.

De modo análogo, a sequência de estados $\boldsymbol{q} = \{q_1, \ldots, q_N\}$ é obtida a partir das probabilidades de transição a_{kj}[1].

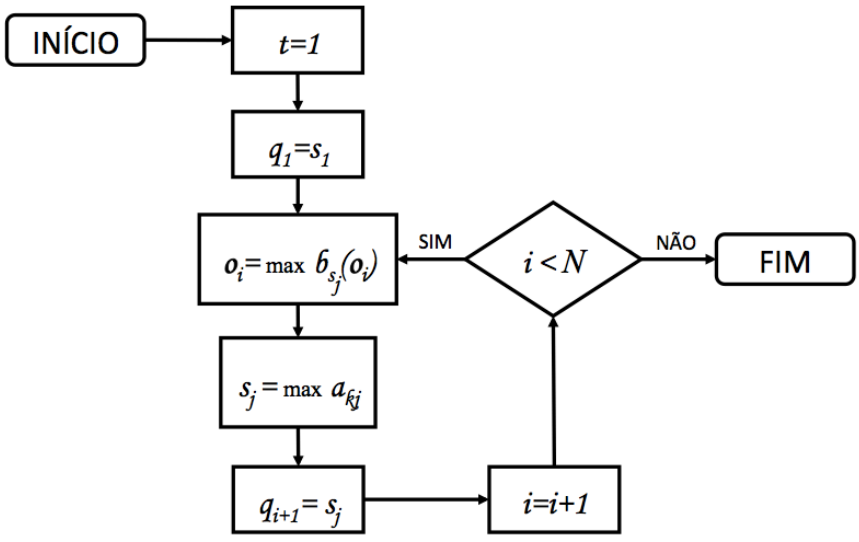


Fig. 3.8 Geração de observações

3.4.3.2 Duração

A incorporação de densidade de duração de estados dos HMM é refletida na significativa melhoria da qualidade da modelagem [51]. As durações dos S estados de cada HMM são, desse modo, tratadas como vetores da forma

$$\boldsymbol{d}_k = [d_{k1} \quad \cdots \quad d_{kS}]^T \tag{3.2}$$

onde d_{ks} é a duração do s-ésimo estado do k-ésimo HMM. Assim, as durações modificam a modelagem dos HMM, retirando as transições de um estado para si mesmo, mostradas na Fig. 2.5.

Cada vetor de duração é modelado conforme uma distribuição Gaussiana de uma única mistura de dimensão S [51]. A reestimativa das probabilidades de saída de cada duração é realizada através do algoritmo Baum-Welch, de modo análogo à estimativa de parâmetros apresentada no Capítulo 2 [85].

3.4.3.3 Árvores de decisão

Conforme foi visto na Seção 3.4.1, cada fone é representado por uma *label* que descreverá seu contexto. Observe que haverá uma variedade muito grande de contextos e portanto, de *labels*, que podem gerar uma quantidade impraticável de HMM a serem treinados. Além disso, deve-se levar em consideração que a especificidade extrema de um modelo pode acarretar na rara coincidência entre aquele e as informações contextuais geradas durante a síntese [29]. Ou seja, o sistema acaba sendo onerado computacionalmente em prol de avanço mínimo.

Assim, a fim de contornar esta dificuldade, é empregada a técnica de árvores de decisão [83], que permite a generalização de modelos e a construção de outros não observados durante o treinamento. Para a criação dessas árvores, inicialmente cada um dos estados $\{s_1, \dots, s_S\}$ são colocados em S grupos. Para cada grupo, é realizada uma pergunta contextual escolhida de acordo com a maior probabilidade de saída gerada, dados os estados no grupo. Tal pergunta tem como respostas possíveis “sim” ou “não, que acarretam a subsequente divisão do grupo em dois. Para cada grupo resultante, este procedimento é repetido e nova pergunta é realizada. A repetição deste processo é feita de forma que um limiar de probabilidade, definido empiricamente, seja alcançado.

Quando o processo é encerrado, as árvores de decisão são obtidas. O processo é ilustrado na Fig. 3.9.

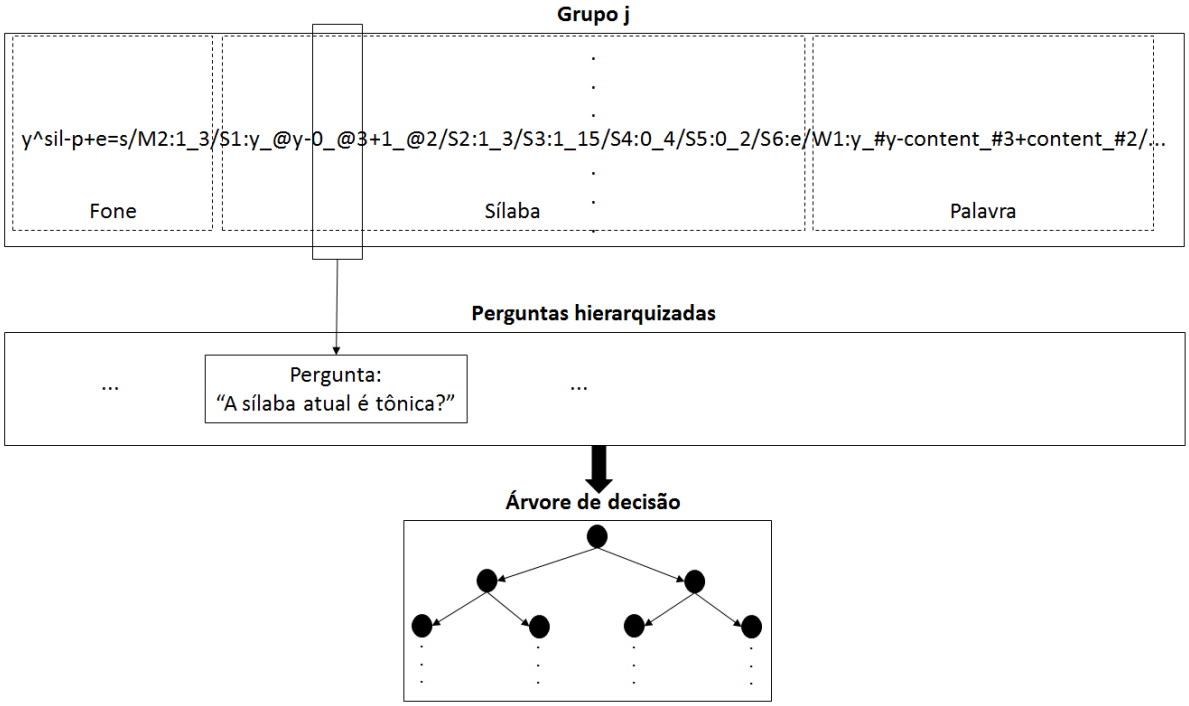


Fig. 3.9 Obtenção da árvore de decisão para o grupo *j*

As árvores de decisão são formadas para cada um dos parâmetros empregados na modelagem dos HMM. Assim, serão geradas, ao final do processo, $3S + 1$ árvores acústicas:

- S árvores para os coeficientes mel-cepstrais;
- S árvores para o logaritmo da frequência fundamental;
- S árvores para os parâmetros de aperiodicidade;
- uma árvore para a duração dos estados.

Tais árvores determinarão, dessa forma, cada modelo que será treinado.

Cabe ressaltar que as características espectrais e prosódicas da fala sintetizada são diretamente influenciadas por esse processo de agrupamento de modelos. Assim, a qualidade da síntese depende da determinação apropriada das perguntas realizadas para a geração das árvores. No caso do sistema implementado, foram utilizadas as perguntas desenvolvidas em [10] para o Português Brasileiro. Tais perguntas são derivadas de cada um dos fatores contextuais expostos na Tab. 3-6, características fonéticas (e.g. fones vozeados, consoantes, vogais e semivogais), ditongos e fones contextos

específicos (e. g. vogais no final de falas, precedidas de paradas ou fricativas e seguidas de silêncio).

3.5 Conclusão

Este capítulo teve o intuito de dar detalhes a respeito do treinamento do sistema de síntese de voz baseado em HMM. Para tal, o mesmo apresentou como foi realizado o projeto, edição e gravação da extensa base para o Português Brasileiro, composta de 11.456 elocuções. Além disso, foram apresentadas as etapas realizadas para o treinamento do sistema, cujos resultados serão apresentados no próximo capítulo.

Capítulo 4

Avaliações subjetivas

4.1 Introdução

A qualidade de sistemas de síntese de voz é associada à sua capacidade de gerar fala inteligível e natural [5]. Frequentemente, testes subjetivos são utilizados como forma de avaliar estes quesitos [17,47,52]. O sistema implementado foi, assim, submetido a tais avaliações como forma de estimar a influência do tamanho da base de dados na qualidade percebida da voz sintetizada. Além disso, também foi realizada sua comparação com sistemas comerciais a fim de verificar seu grau de aceitabilidade frente aos usuários desse tipo de tecnologia.

Neste capítulo, serão apresentados detalhes dos testes realizados e discutidos os resultados. Para tal, o mesmo está organizado da seguinte maneira: na Seção 4.2, trata dos testes através da descrição dos métodos adotados, da base de dados, da definição e da plataforma; na Seção 4.3, apresenta os perfis dos avaliadores, detalha as técnicas para tratamento e análise dos dados e discute os resultados; por fim, a Seção 4.4, apresenta uma breve conclusão do capítulo.

4.2 Testes

4.2.1 Método de avaliação

Conforme mencionado anteriormente, testes subjetivos são empregados na avaliação da qualidade percebida de sistemas de síntese de voz. Alguns dos métodos mais utilizados são o MOS (*Mean Opinion Score*) e o A *versus* B.

No MOS, os indivíduos são convidados a aplicar pontuações às sentenças conforme uma escala discretizada de 1 (ruim) a 5 (excelente). Tal escala é conhecida pelo nome do próprio método [87]. Já o método A *versus* B inclui a confrontação de pares de sentenças dentre os quais os testadores devem escolher aquelas que considerarem melhor [88].

Para a verificação mais direta dos objetivos apontados na seção 4.1, referentes à influência do tamanho da base de dados e ao comparativo com sistemas comerciais, foi

adotada a técnica por preferência de A *versus* B. Este método possui como vantagem adicional a independência de uma escala cuja graduação é intrínseca ao usuário.

Na utilização da técnica de A *versus* B, admitiu-se, além das opções de preferência entre os dois áudios avaliados, a de "empate", caso os indivíduos não percebessem diferença quanto a ambos.

4.2.2 Base de dados

Para a realização dos testes, foram utilizados quatro grupos diferentes de vinte frases foneticamente balanceadas escolhidas de [89]. A importância do balanceamento fonético é permitir a variedade de unidades num conjunto de dados reduzido. As Tab. 4-1 a Tab. 4-4 relacionam os grupos de frases empregados por teste.

Cabe ainda dizer que as frases empregadas para as avaliações, embora tenham sido gravadas na base, não foram utilizadas nos treinamentos a fim de que fossem inéditas para a geração do sintetizador, conferindo uma situação mais real de emprego.

As frases geradas foram gravadas em formado *wave*, taxa de amostragem de 16 kHz e precisão de 16 bits/amostra, conforme indicação no fórum do HTS [100].

Tab. 4-1 Conjunto de frases foneticamente balanceadas aplicadas à pesquisa 1 [89]

Frase 1	Há algumas coisas que não podem deixar de serem vistas em Paris.
Frase 2	Há demanda por real, não por dólar, cuja cotação cai.
Frase 3	Há cento e setenta bilhões de moedas de um centavo em circulação.
Frase 4	Cada uma delas custa oito décimos de um centavo.
Frase 5	O mercado fica de alto risco a curto prazo.
Frase 6	A perspectiva continua otimista para o médio e longo prazos.
Frase 7	O desafio agora é controlar, com medicamentos, essa secreção.
Frase 8	O jogo contra a Suécia deixou todo o país preocupado.
Frase 9	Está instalado na casa do avô de Lúcia Flexa de Lima.
Frase 10	Funcionários do governo do estado dão orientação técnica aos voluntários.
Frase 11	O ministério diz que não foi consultado sobre a transação.
Frase 12	Este ano a empresa deve perder cerca de sete milhões de dólares.
Frase 13	Ela começou a tirar fotos ontem em Buenos Aires.
Frase 14	Ele se comprometeu a ler e responder parte da minha correspondência.
Frase 15	Os brasileiros no exterior ganharam um poder aquisitivo sem precedentes.
Frase 16	O objetivo é abordar todas as fases da ópera.
Frase 17	O curso é diário e tem duração de três meses.
Frase 18	Os jogadores santistas deixaram o gramado sob vaias da torcida.
Frase 19	A torcida voltou a criticar a atual administração e não economizou críticas.
Frase 20	A Infraero ainda não sabe em qual dos dois aeroportos Alexandre embarcou.

Tab. 4-2 Conjunto de frases foneticamente balanceadas aplicadas à pesquisa 2 [89]

Frase 1	São essas qualidades que inspiraram o plano real desde a sua criação.
Frase 2	Todos os batizados são consagrados a Deus e devem tender à santidade.
Frase 3	"O homem Sem Qualidades" não é um livro comum.
Frase 4	Os problemas surgem nas importações diretas via catálogos, por exemplo.
Frase 5	Lula chegou ao Rio com duas horas de atraso.
Frase 6	No ano, a taxa é de três vírgula sete por cento.
Frase 7	Batizado de Heitor, o trabalho traz treze faixas compostas pelo guitarrista.
Frase 8	Já Viola levou o terceiro cartão amarelo domingo passado.
Frase 9	Inflação volta a subir no Rio e em São Paulo.
Frase 10	CINCO: frequentemente responde perguntas antes que elas sejam concluídas.
Frase 11	Quero estar no grupo e ajudar sempre que necessário.
Frase 12	Expedito Andrelino de Souza, cinquenta e um anos, encarregado de almoxarifado.
Frase 13	Isso mostra que nossa flutuação cambial suja e voluntariosa foi apenas tolice.
Frase 14	Nós somos uma espécie de contingente escolhido do povo brasileiro.
Frase 15	Não é o primeiro carregamento de armas para Angola.
Frase 16	O dinheiro para pagamento dos policiais militares sairia de verba da prefeitura.
Frase 17	Até agora, na televisão, eles já somam quatro ministros.
Frase 18	Pode ser o início de uma campanha de contestação da vitória.
Frase 19	Da mesma forma, empataram nas finalizações: um a um.
Frase 20	O jogador do Palmeiras foi o único que lançou três vezes.

Tab. 4-3 Conjunto de frases foneticamente balanceadas aplicadas à pesquisa 3 [89]

Frase 1	Quanto ao seu próprio time o diagnóstico é menos generoso.
Frase 2	Não há desculpas para o mau jogo que fizemos contra a Arábia.
Frase 3	Salário real médio do varejo aumenta doze por cento no mês passado
Frase 4	Em Tóquio, o dólar fechou cotado a noventa e oito ienes.
Frase 5	Os sobreviventes cremaram seus mortos e procuram reconstruir suas vidas.
Frase 6	Os dois times perderam uma chance atrás da outra.
Frase 7	A geração de Zico não seria campeã do mundo.
Frase 8	O curso é gratuito e tem a duração de seis meses.
Frase 9	Assaltava desde os oito anos e meus amigos estão quase todos presos.
Frase 10	O mercado de ações vive momentos de plena euforia.
Frase 11	A imprensa não se incomodou, nem denunciou esse detalhe.
Frase 12	Primeiro arredondou a idade do percussionista para cinqüenta anos.
Frase 13	Os médicos ainda não sabem quanto tempo ele deverá ficar no hospital.
Frase 14	O feminino fica em quarto, sua melhor colocação em olimpíadas.
Frase 15	O modelo foi reestilizado e ganhou linhas mais agressivas.
Frase 16	Seu interior, no entanto, lembra o de automóveis japoneses.
Frase 17	Sete membros do grupo que se opõe à paz continuam presos.
Frase 18	Os jogadores reclamam do calor, que consideram excessivo dentro do estádio.
Frase 19	O limite anterior para estes títulos bancários era de noventa dias.
Frase 20	Também preparam a transição para o real no caso das aplicações financeiras.

Tab. 4-4 Conjunto de frases foneticamente balanceadas aplicadas à pesquisa 4 [89]

Frase 1	Pesquisa é uma coisa que muda a toda hora.
Frase 2	No total, serão chamados vinte e seis mil candidatos.
Frase 3	O número de convocados por vaga é de doze candidatos.
Frase 4	Atualmente, esse abatimento é limitado a setenta por cento dos gastos.
Frase 5	Sandra Regina Machado: acho que ela enfim criou juízo.
Frase 6	Eles estão colocando armadilhas nas fazendas onde já ocorreram os ataques.
Frase 7	Dessas, somente umas trezentas e vinte foram inauguradas em território americano.
Frase 8	No total, sete mísseis foram disparados contra o encrave.
Frase 9	Em Florianópolis, foi registrado dois graus Celsius na manhã de domingo.
Frase 10	As situações ditas embaraçosas são resolvidas com os dados.
Frase 11	Itamar tem razão de estar exultante como nunca desde que virou presidente.
Frase 12	A mãe de todas as reformas é a reforma política.
Frase 13	Conseguiram eliminar áreas supérfluas ou que antes eram desperdiçadas.
Frase 14	Uma lata de leite em pó integral vale um ingresso.
Frase 15	A maioria dos passageiros do barco naufragado era de crianças.
Frase 16	A provável causa do acidente foi excesso de lotação a bordo.
Frase 17	Não prometo nada, porque não adianta eu prometer e não cumprir.
Frase 18	Se for eleito, vocês vão ver o meu trabalho.
Frase 19	Ele era um dos poucos atores negros que tinham espaço.
Frase 20	A secretaria estadual de saúde distribuirá cem mil preservativos no carnaval.

4.2.3 Descrição

Os testes foram dispostos em quatro avaliações. As duas primeiras possuíram o intuito de avaliar o impacto do tamanho da base utilizada no treino do sistema na qualidade percebida pelo usuário. Para tal, foram confrontados áudios gerados através treinos com base de 1.000 contra 5.000 frases e de 3.000 contra 5.000 frases, respectivamente. Os áudios correspondentes a cada pesquisa foram gerados a partir dos conteúdos das Tab. 4-1 e Tab. 4-2.

Nessas avaliações, as frases foram dispostas aleatoriamente a fim de evitar qualquer viés na seleção por preferência.

A segunda etapa das avaliações consistiu da comparação do sistema treinado com 5.000 frases com dois outros sistemas comerciais. Tais sistemas foram escolhidos em função de seu emprego na área de ensino de idiomas, onde a naturalidade e a correta pronúncia do vocabulário é prezada. Vale ressaltar que os mesmos utilizam-se de vozes femininas, ao passo que o sistema deste trabalho faz uso de locutor masculino.

Os áudios empregados nas pesquisas comparativas com sistemas comerciais foram aqueles gerados a partir das frases das Tab. 4-3 e Tab. 4-4.

Em função do crescente período de tempo necessário para o treinamento do sistema conforme o aumento do tamanho da base, agravado pela disponibilidade de máquinas que pudessem executá-lo, optou-se por utilizar sistemas treinados com 1.000, 3.000 e 5.000 frases. O treinamento com essas bases levou, respectivamente, em torno

de 8 horas, 30 horas e 58 horas em máquinas com processadores *Intel Core i7* de 8 núcleos de *3,4 GHZ*, HD magnético e memória RAM de *8 GB*.

A base completa foi treinada em aproximadamente 150 horas, em um servidor mais robusto com processador *Intel Xeon* de 4 núcleos, HD com tecnologia SSD (*Solid-State Drive*) e memória RAM de *16 GB*. O sintetizador treinado com esta base está disponível na plataforma *web*, conforme será apresentado na Seção 4.2.4. Este sintetizador não foi utilizado nos testes.

Vale dizer que obteve-se a redução do tempo necessário de processamento pela diminuição do intervalo despendido na tarefa de geração de árvores de decisão, ponto crítico do treino. Isto foi alcançado pela simplificação do detalhamento de contexto envolvido nos arquivos de *labels*, através da mudança de sua geração para que fosse feita a partir dos arquivos *mono* obtidos durante a fase de NLP. Todavia, testes informais demonstraram que embora a redução de tempo de treinamento tenha sido significativa (4 horas para o caso da base de 1.000 frases), a degradação na qualidade do áudio acompanhou sensivelmente este resultado, o que levou à persistência no uso de mais informações sobre a pronúncia.

4.2.4 Plataforma

Os testes realizados foram feitos através de um sistema *web*, a fim de alcançar maior número de participações. Para isso, foi utilizado um servidor com plataforma *Linux* na distribuição *Ubuntu 14.04*, processador *Intel Core i7 3,4 GHz x 8*, memória RAM de *8 GB*. Os navegadores utilizados foram *Safari*, *Google Chrome* e *Mozilla Firefox*, uma vez que o *Internet Explorer* não possibilita a reprodução de áudio em formato *wave*. A plataforma de testes foi desenvolvida em *Java* com o *framework web Groovy Grails 2.3.7*.

A entrada no sistema contou com a utilização de *login* com senha, cuja interface é ilustrada na Fig. 4.1. Essa característica de *login* permitiu a configuração dos testes de modo que os usuários poderiam salvar seu avanço por etapa de avaliação e prosseguir conforme sua conveniência. Todavia, essa funcionalidade acarretou que alguns usuários não realizassem todos os estágios de apreciação.

Fig. 4.1 Interface de entrada para acesso aos testes

Para o cadastro do *login* e posterior análise do público envolvido nos testes subjetivos, necessitou-se de registro prévio através de formulário (Fig. 4.2). Esse registro foi realizado pelos próprios avaliadores antes de seu primeiro acesso ao sistema.

Fig. 4.2 Formulário para registro de novo avaliador

Após a entrada no sistema, a tela inicial para seleção dos testes a serem efetuados é apresentada, conforme a Fig. 4.3. Vale dizer que foi permitido aos avaliadores proceder os testes na ordem de sua conveniência.

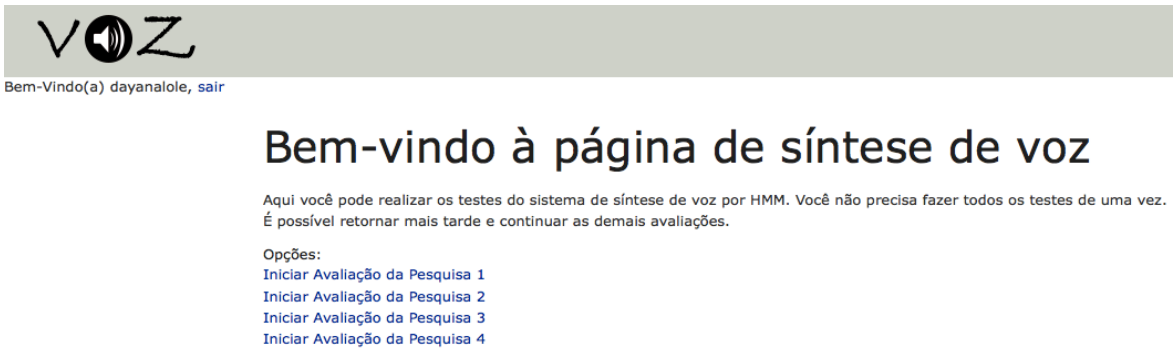


Fig. 4.3 Tela inicial dos testes

Durante as avaliações, os áudios apresentados aos usuários estavam sem transcrição, para evitar quaisquer distrações. Foi permitido que cada usuário reproduzisse cada som quantas vezes julgasse necessário.

As instruções para os testes estavam disponíveis no topo de cada página de avaliação, onde os usuários eram aconselhados a portar fones de ouvido. A Fig. 4.4 ilustra a página de avaliações.

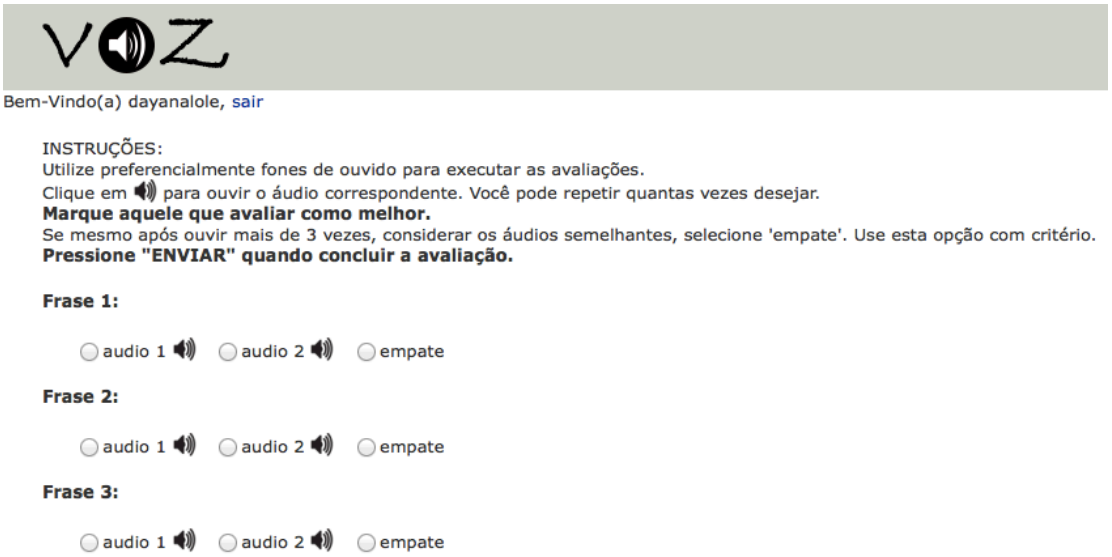


Fig. 4.4 Parte da página de avaliação com instruções ao usuário

Ademais, não foi autorizado aos usuários o envio de avaliações incompletas, isto é, com itens não respondidos. De forma semelhante, a alteração das respostas apenas foi possível antes do envio dos testes pelo avaliador.

Por fim, para usuários com permissões administrativas, é possível visualizar na tela inicial uma opção para o sintetizador (Fig. 4.5). Esta opção viabiliza o acesso ao

sintetizador com opções de escolha para base treinada com 1.000, 3.000, 5.000 e 11.456 frases (base completa). Esta interface pode ser vista na Fig. 4.6.

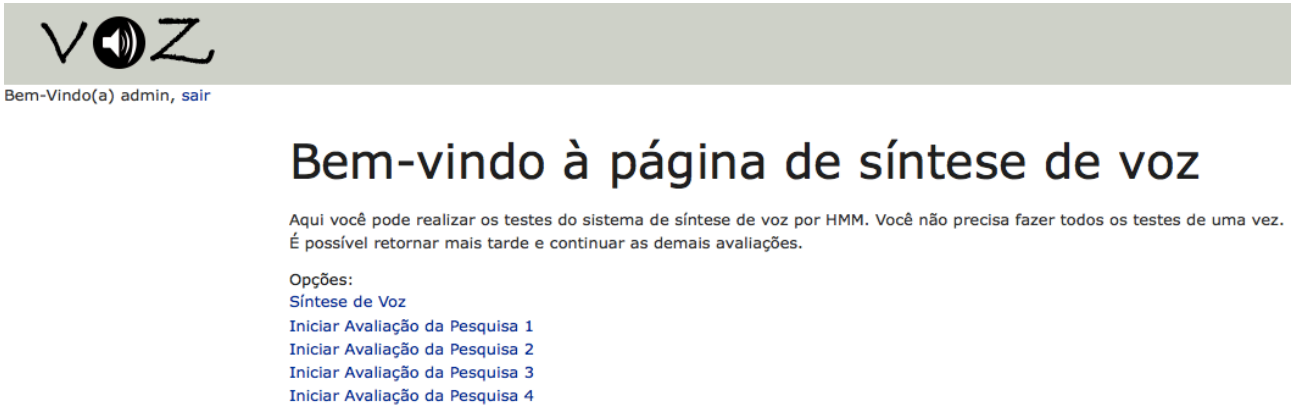


Fig. 4.5 Tela inicial para usuário administrador com opção "Síntese de voz"

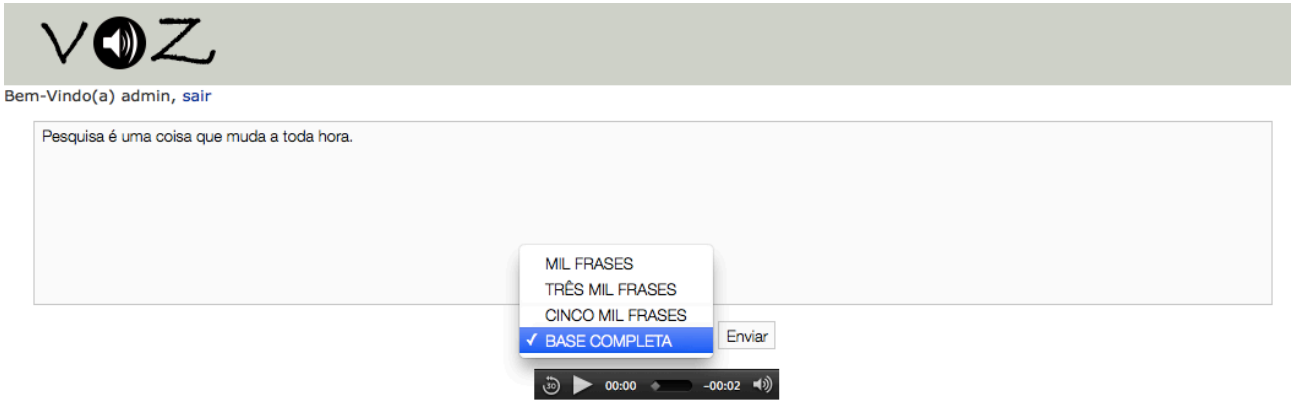


Fig. 4.6 Interface do sintetizador

4.3 Resultados

4.3.1 Perfil dos avaliadores

A seguir, é apresentado o perfil dos avaliadores de acordo com cada teste realizado, uma vez que o quantitativo de participantes foi variável devido às características do sistema expostas na seção anterior.

Os totais de participantes das avaliações, compostos em sua maioria de público não especialista na área de processamento de voz, estão listados na Tab. 4-5.

Tab. 4-5 Totais de participantes por avaliação

Avaliação	Total de participantes
1 (base de 1.000 *versus* 5.000 frases)	68
2 (base de 3.000 *versus* 5.000 frases)	57
3 (sistema HMM-PB *versus* sistema comercial I)	53
4 (sistema HMM-PB *versus* sistema comercial II)	55

Os usuários apresentaram idades no intervalo de 18 a 57 anos, concentrando-se na faixa etária mais jovem. A distribuição dos mesmos pode ser observada através dos gráficos das Fig. 4.7 a Fig. 4.10.

Quanto ao grau de escolaridade, a maioria dos avaliadores é declarado como possuidor de nível superior completo, conforme pode ser atestado pelas distribuições expostas nas Fig. 4.11 a Fig. 4.14.

No que diz respeito ao sexo, os avaliadores do sexo masculino estão em maior número. As distribuições por sexo estão dispostas nas Fig. 4.15 e Fig. 4.16.

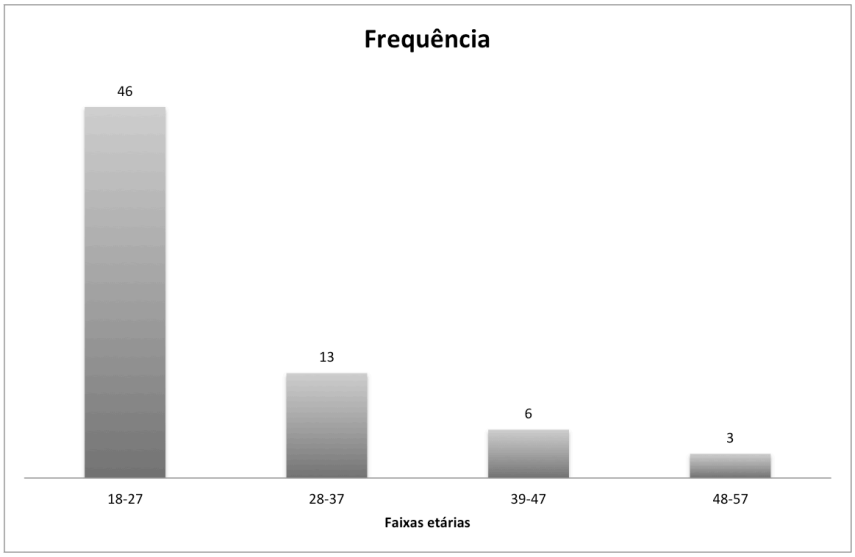


Fig. 4.7 Distribuição de avaliadores da pesquisa 1 por faixa etária

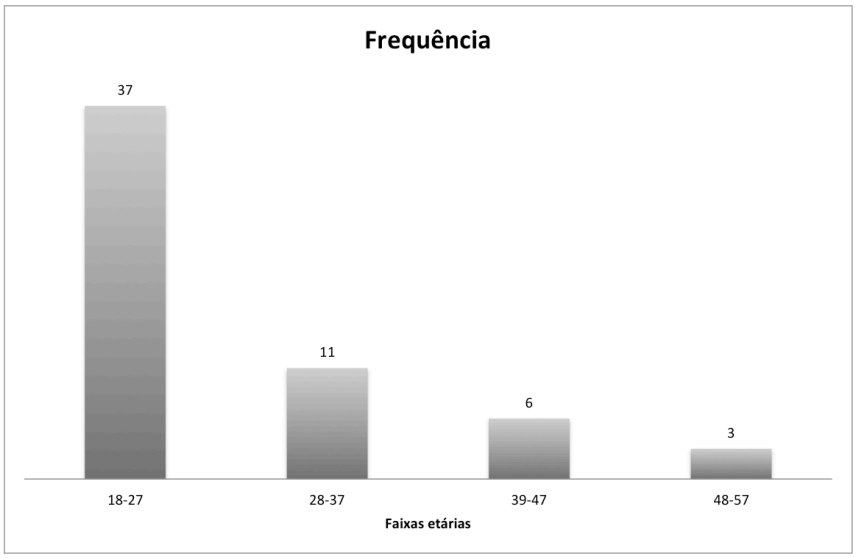


Fig. 4.8 Distribuição de avaliadores da pesquisa 2 por faixa etária

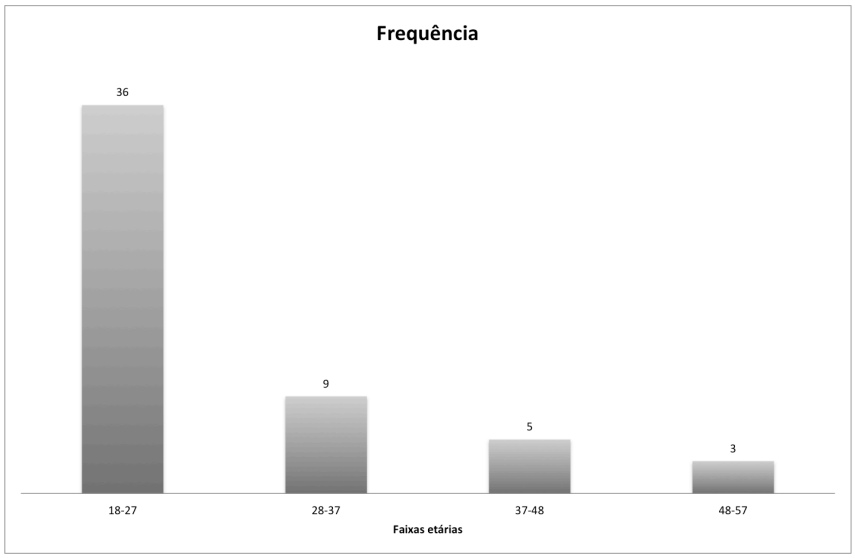


Fig. 4.9 Distribuição de avaliadores da pesquisa 3 por faixa etária

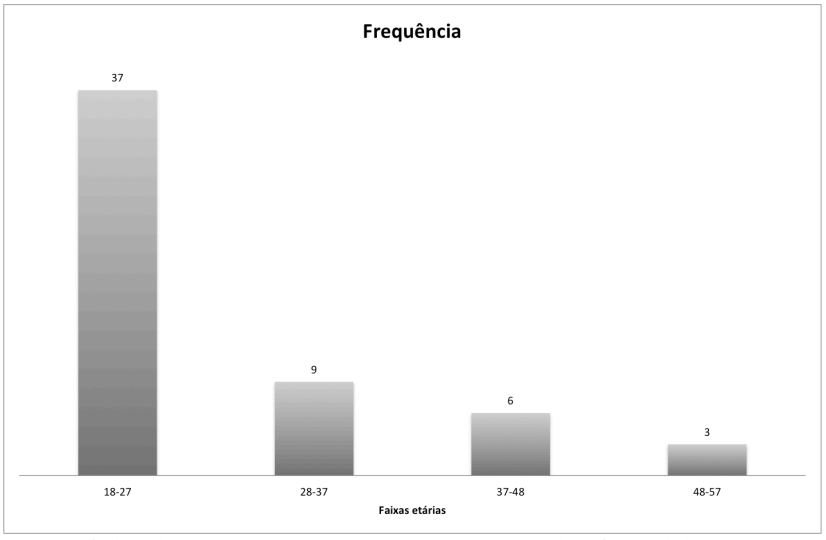


Fig. 4.10 Distribuição de avaliadores da pesquisa 4 por faixa etária

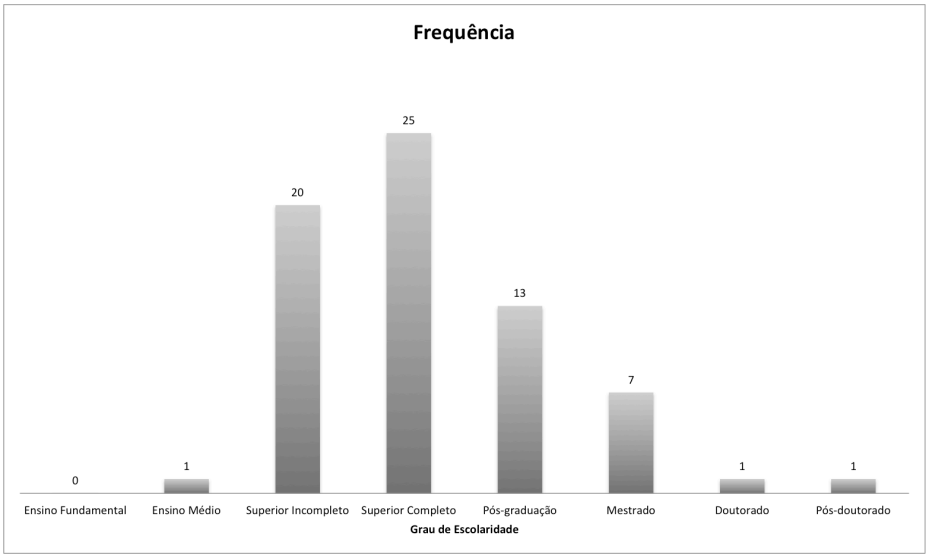


Fig. 4.11 Distribuição de avaliadores da pesquisa 1 por grau de escolaridade

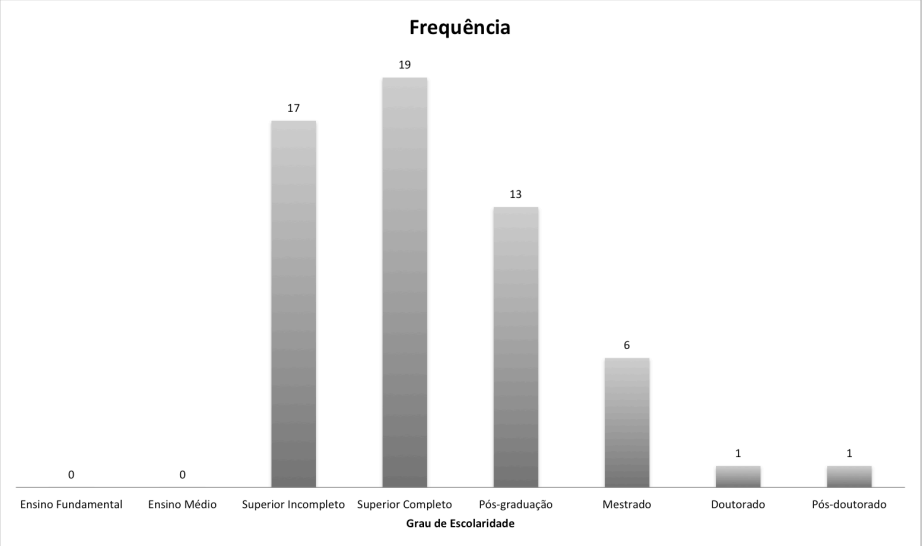


Fig. 4.12 Distribuição de avaliadores da pesquisa 2 por grau de escolaridade

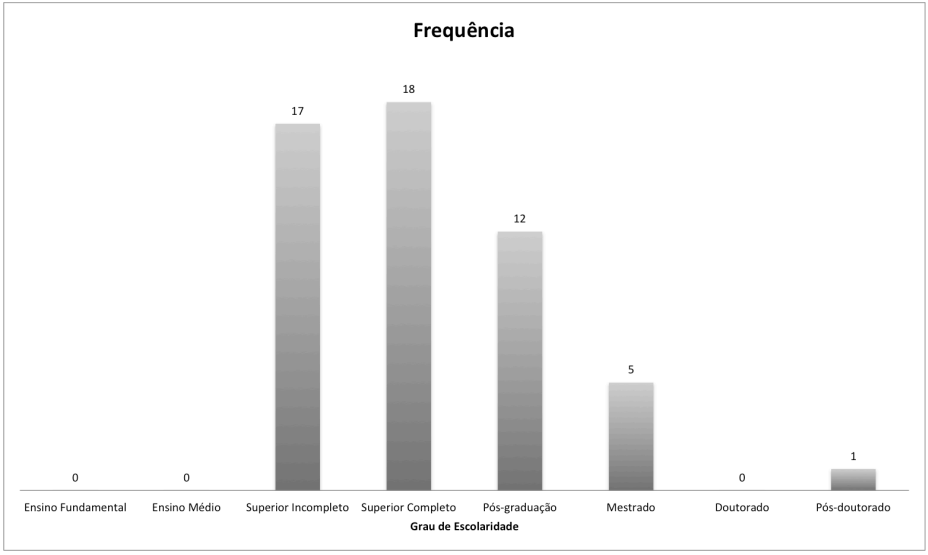


Fig. 4.13 Distribuição de avaliadores da pesquisa 3 por grau de escolaridade

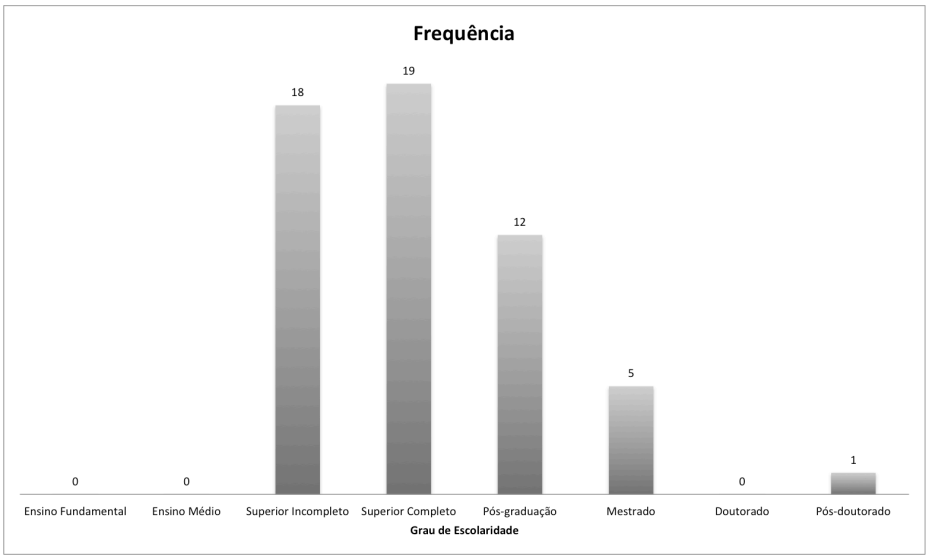
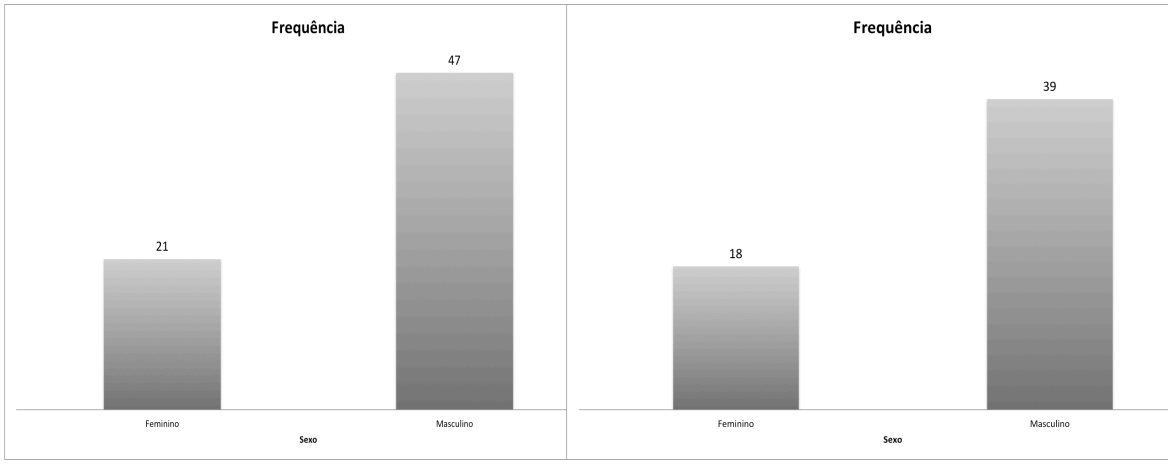
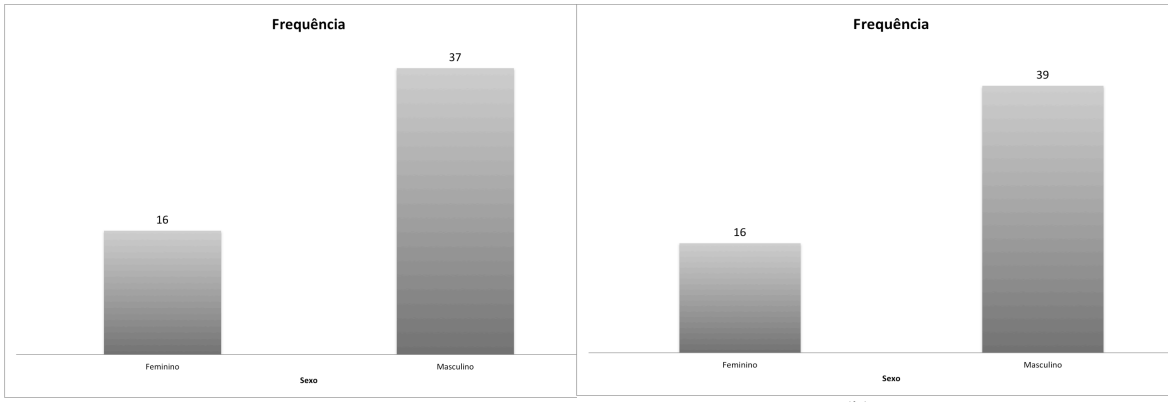


Fig. 4.14 Distribuição de avaliadores da pesquisa 4 por grau de escolaridade



(a) (b)

Fig. 4.15 Distribuição de avaliadores das pesquisas 1 (a) e 2 (b) por sexo



(a) (b)

Fig. 4.16 Distribuição de avaliadores das pesquisas 3 (a) e 4 (b) por sexo

4.3.2 Métricas

Para efeito de análise, foram utilizadas as médias aritméticas por respostas nas quatro avaliações realizadas. A definição da média aritmética é dada por

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \tag{4.1}$$

onde x_i representa a i-ésima amostra e n diz respeito ao total de amostras.

Além disso, também foi empregado o desvio-padrão populacional para cada medida efetuada. Este desvio é descrito como

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2}, \tag{4.2}$$

cujos parâmetros x_i e n estão de acordo com a definição anterior e $\bar{x}$ é a média aritmética definida em (4.1).

4.3.3 Remoção de *outliers*

4.3.3.1 Método de *Z-score*

Como forma de refinar os resultados, foram utilizadas técnicas de identificação de *outliers* para posterior remoção. O método conhecido como *Z-score* é bastante difundido, onde a identificação dos *outliers* é realizada através da medida da razão [90]

$$Z_i = \frac{x_i - \bar{x}}{\sigma}, \tag{4.3}$$

onde Z_i é a medida do *Z-score* da i-ésima amostra. Em geral, valores de $Z_i > 3$ são considerados como referentes a *outliers*.

De acordo com [90], o valor máximo para essa pontuação depende do número de amostras e é dado por

$$Z_{i_{max}} = \frac{(n-1)}{\sqrt{n}}, \tag{4.4}$$

em que n é, conforme definido anteriormente, o número de amostras.

4.3.3.2 Método de *Z-score* modificado

Uma desvantagem no uso da técnica de *Z-score* é a sensibilidade da média e do desvio-padrão a valores de *outliers* extremos. Assim, discrepâncias intermediárias podem ser mascaradas com o uso do método.

Com o intuito de contornar essa dificuldade, [91] propôs uma alteração da técnica substituindo a média e o desvio padrão pela mediana e o desvio absoluto da mediana. Esta modificação se justifica em função da menor sensibilidade da mediana em relação a *outliers* extremos.

Assim, a técnica *Z-score* modificada é expressa matematicamente como

$$M_i = \frac{0{,}6745(x_i - \tilde{x})}{MAD}, \tag{4.5}$$

onde $\tilde{x}$ é a mediana e

$$MAD = mediana(|x_i - \tilde{x}|). \tag{4.6}$$

Neste caso, assume-se como *outliers* casos onde $|M_i| > 3{,}5$.

Vale ressaltar que, neste trabalho, os *outliers* foram representados por usuários cujas respostas apresentaram-se distantes do perfil da maioria para uma dada pesquisa de acordo com a técnica de *Z-score* modificado.

4.3.4 Análise dos resultados

Conforme apresentado na seção 4.2.3, a primeira pesquisa envolveu a confrontação do sistema de síntese baseado em HMM com bases de 1.000 e 5.000 frases. Os resultados das avaliações podem ser resumidos através da Tab. 4-6. A Fig. 4.17 demonstra a frequência de respostas de acordo com cada frase.

Tab. 4-6 Resultados da pesquisa 1

	Base 1000 frases	Base 5000 frases	Empate
Total de respostas	260	736	364
Média	3,82	10,82	5,35
Desvio-Padrão	2,22	3,21	3,49
Média %	19,12	54,12	26,76
Desvio-padrão %	11,11	16,04	17,46

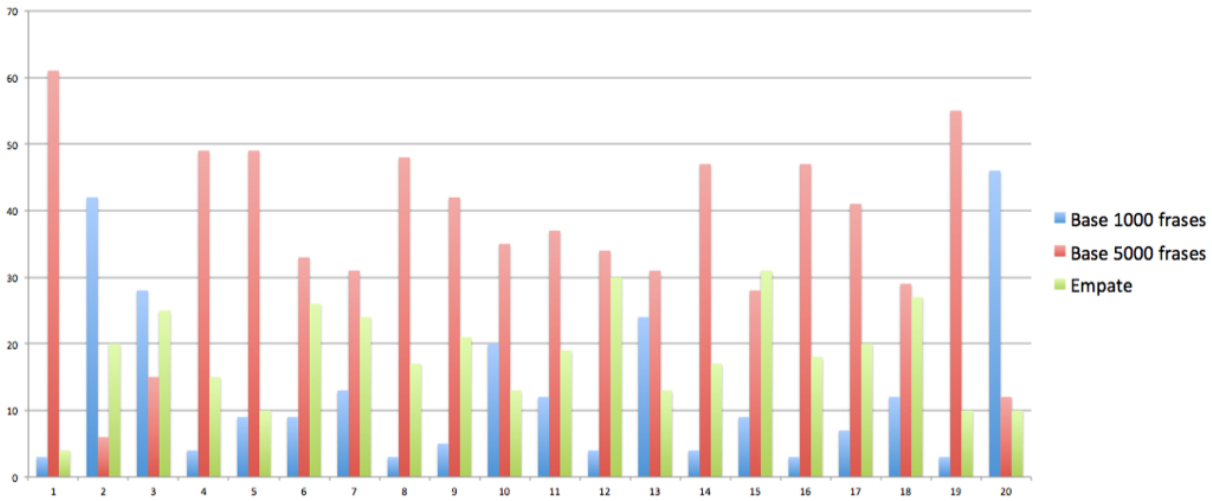


Fig. 4.17 Frequência de respostas considerando cada frase para a pesquisa 1

Mediante observação dos resultados desta pesquisa, pode-se verificar que há sensível diferença na qualidade percebida pelos avaliadores, onde a maioria considera o sistema treinado com 5.000 frases superior àquele com menor número de frases.

No segundo teste houve um estreitamento na diferença do quantitativo de frases utilizado em cada treino. Os resultados para esta situação são apresentados na Tab. 4-7 e na Fig. 4.18.

Tab. 4-7 Resultados da pesquisa 2

	Base 3000 frases	Base 5000 frases	Empate
Total de respostas	290	499	351
Média	5,09	8,75	6,16
Desvio-Padrão	2,50	2,81	3,69
Média %	25,44	43,77	30,79
Desvio-padrão %	12,51	14,03	18,44

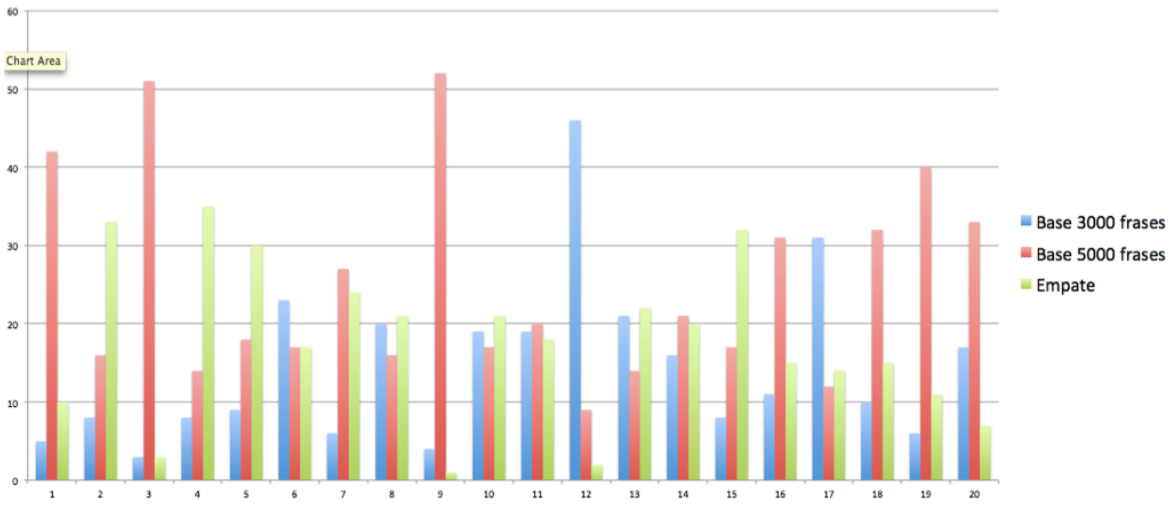


Fig. 4.18 Frequência de respostas considerando cada frase para a pesquisa 2

Com a aproximação dos tamanhos das bases utilizadas no treinamento, notou-se que, mesmo tendo-se mantido a preferência pelo treino com 5.000 frases, a distinção entre as qualidades da síntese foi reduzida.

A análise conjunta de ambos os testes reflete que o aumento do número de frases no treino influenciou na apreciação de qualidade do áudio gerado. Pode-se dizer que estes resultados refletem a elevação do tamanho das árvores de decisão à medida que a quantidade de elocuções é acrescida. Para o caso de $\log(F0)$, por exemplo, o treinamento com 1.000 frases gera árvores da ordem de 6.000 nós, enquanto o de 3.000 frases produz na ordem de 11.000 nós e o de 5.000 frases, 15.000 nós.

De acordo com [10], a ausência de dados prejudica a qualidade, enquanto que o contrário possibilita aos HMM o rastreio mais apropriado dos atributos das diversas *labels* contextuais derivadas do banco utilizado para treino. Este fato é mais sensível no caso do Português Brasileiro devido a sua variedade fonética.

Todavia, é importante observar que os resultados sugerem que esta percepção tende a um valor limite, onde a elevação do tamanho da base empregada no treinamento não mais a influencia.

As avaliações seguintes envolveram a comparação com sintetizadores comerciais, doravante denominados Sistema Comercial I (pesquisa 3) e Sistema Comercial II (pesquisa 4). Os resultados destas consultas são mostrados nas Tab. 4-8 e Fig. 4.19, para a aferição com o primeiro sistema comercial, e Tab. 4-9 e Fig. 4.20, com o segundo.

Tab. 4-8 Resultados da pesquisa 3

	HMM-PB	Sistema Comercial I	Empate
Total de respostas	556	437	87
Média	10,30	8,09	1,61
Desvio-Padrão	6,26	5,57	2,63
Média %	51,48	40,46	8,06
Desvio-padrão %	31,29	27,86	13,14

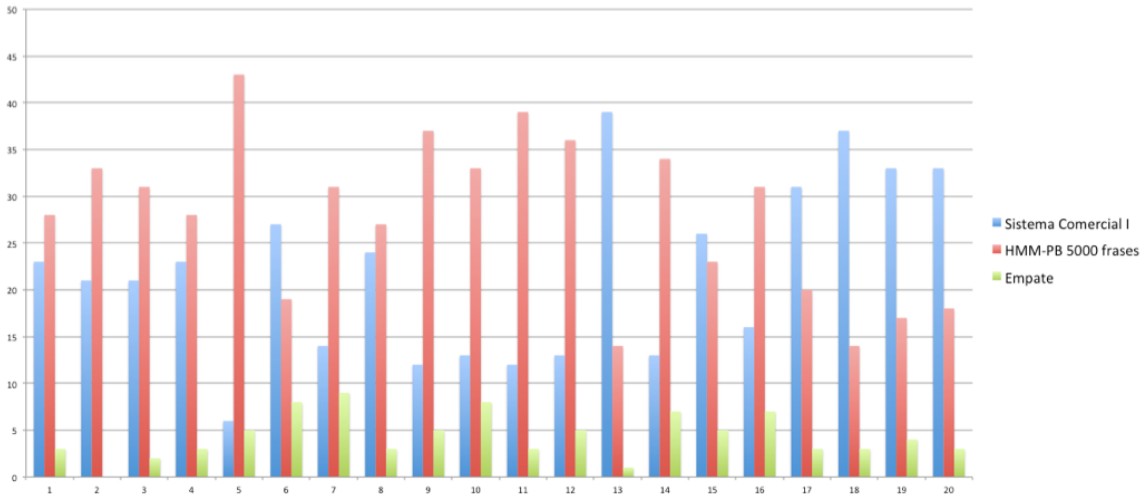


Fig. 4.19 Frequência de respostas considerando cada frase para a pesquisa 3

Tab. 4-9 Resultados da pesquisa 4

	HMM-PB	Sistema Comercial II	Empate
Total de respostas	458	410	192
Média	8,64	7,74	3,62
Desvio-Padrão	4,16	3,65	3,66
Média %	43,21	38,68	18,11
Desvio-padrão %	20,81	18,23	18,31

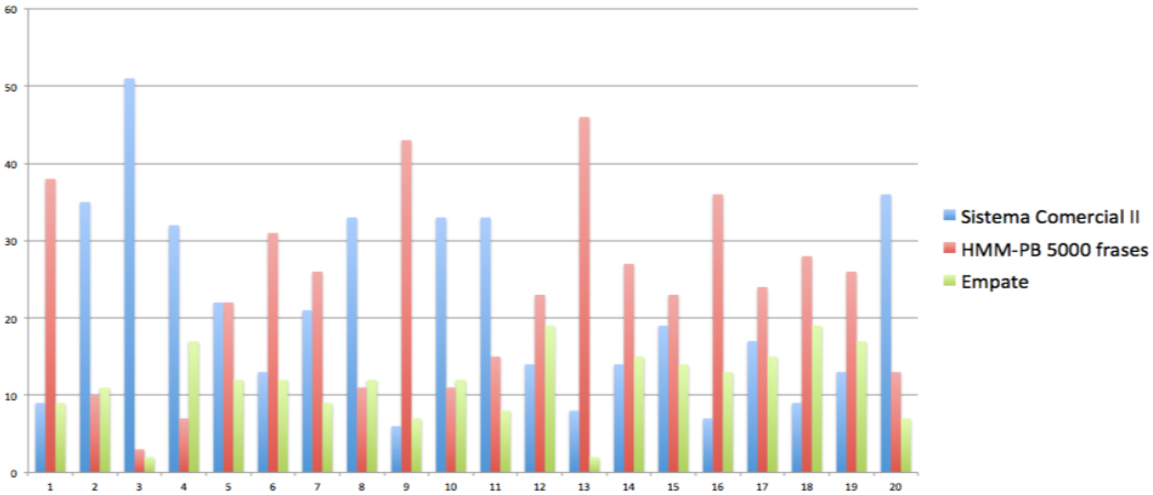


Fig. 4.20 Frequência de respostas considerando cada frase para a pesquisa 4

Observando as Fig. 4.19 e Fig. 4.20, percebe-se a ocorrência de casos extremos, com menos de 5 votos para um dos sistemas comerciais e quase totalidade para o sintetizador baseado em HMM. Verificou-se posteriormente que nas frases onde tal ocorreu, houve erros de pronúncia dos sistemas comerciais (e.g. palavra “cremaram”, para o Sistema Comercial I) e acerto do sistema desenvolvido neste trabalho. O oposto também aconteceu. Um exemplo dessa ocorrência foi a palavra “Celsius” pronunciada de maneira incorreta pelo sintetizador baseado em HMM.

Mediante observação geral dos resultados, é possível perceber que o sistema treinado neste trabalho obteve, na média, preferência entre os avaliadores. Entretanto, observando-se a parcela de empate e as diferenças entre os percentuais de respostas médios, verifica-se uma maior aproximação entre o mesmo e o Sistema Comercial II.

Considerando os desvios-padrões obtidos, tem-se que há um empate técnico entre ambos os sistemas comerciais e aquele desenvolvido através da metodologia dos HMM com uma base de 5.000 frases. Cabe ressaltar que os desvios-padrões elevados também podem manifestar a preferência dos usuários pela voz que consideraram mais agradável, uma vez que a proximidade da qualidade da síntese entre os sistemas comparados passa a não ser um fator de considerável discrepância.

Os resultados demonstram a capacidade da técnica dos HMM em obter prosódia, naturalidade e inteligibilidade equiparáveis a sistemas utilizados pelo público.

4.4 Conclusão

Este capítulo teve o intuito de detalhar os testes, apresentar e discutir os resultados obtidos. Para tal, foram esboçadas as decisões de projeto de testes, bem como de tratamento de dados. Mediante a análise dos testes realizados, foi possível perceber que o aumento da base empregada no treinamento possui reflexo positivo na percepção de qualidade pelos usuários. Todavia, esta qualidade parece aumentar até um limiar. Além disso, obteve-se como o resultado a equiparação do sistema baseado em HMM treinado com base de 5.000 frases àqueles empregados comercialmente.

Capítulo 5

Conclusão e Trabalhos Futuros

5.1 Conclusão

Este trabalho realizou o tratamento de uma nova base de dados de elevada qualidade para o Português Brasileiro e aplicou-a a um sistema de síntese de voz baseado HMM utilizando a ferramenta HTS, uma extensão do *toolkit* HTK.

Para isso, o Capítulo 1 apresentou uma perspectiva da área de síntese de voz, contextualizando a técnica por HMM e alertando para a dificuldade encontrada na ausência de amplas bases de qualidade para o caso do idioma Português Brasileiro – principais temas desta dissertação. Além disso, foram abordados os objetivos envolvidos neste projeto.

O Capítulo 2 tratou da apresentação da arquitetura de sistemas de conversão texto-fala, bem como da introdução ao tema de HMM. Além disso, foram apresentadas as aproximações matemáticas empregadas na geração de parâmetros implementadas no HTS. Neste ponto, ficou explícita a opção pelo algoritmo de geração de parâmetros baseada no critério de verossimilhança considerando a variância global.

O Capítulo 3 descreveu, primeiramente, a nova base de dados utilizada neste trabalho, desde o seu projeto até a sua edição. A base concebida conta, em sua formação, com 11.456 elocuções. Em seguida, foram abordadas as etapas de preparação dos dados, que constituem, em última análise, ao NLP, e aquelas envolvidas no treinamento do sistema propriamente dito.

O Capítulo 4 discorreu sobre as avaliações subjetivas, o método empregado, a base de dados e a plataforma *web* desenvolvida. Ademais, foram avaliados os resultados mediante os testes procedidos com mais de 50 usuários, compostos de maioria leiga no assunto de processamento de voz. As avaliações foram realizadas através do método A *versus* B e envolveram o teste da variação de tamanho da base utilizada no treino e a comparação com dois sistemas comerciais. Os dados obtidos foram analisados conforme as métricas estatísticas de média e desvio-padrão e refinados conforme o método de *Z-score* modificado.

Os resultados do teste do sistema treinado com bancos de tamanhos diferentes confirmaram a expectativa de que uma base maior e mais robusta favorece a qualidade

do sistema de síntese através de HMM. Todavia, também foi possível observar que há um limiar para a influência do tamanho do banco de frases empregado no treino na qualidade percebida pelos ouvintes.

Já as avaliações comparativas com sistemas comerciais revelaram que o sintetizador por HMM treinado com base de 5.000 frases corresponde e até supera, na média, a qualidade dos mesmos.

5.2 Trabalhos Futuros

Levando-se em consideração as contribuições deste trabalho, alguns passos seguintes são propostos a seguir:

- Proposta e gravação de frases que possuam fones cuja qualidade ficou degradada na síntese;
- Elaboração de plataforma de treinamento do sistema de síntese em servidor com elevada capacidade de processamento e acesso remoto;
- Adaptação do sintetizador a sotaques variados do Português Brasileiro, aproveitando a capacidade de modificação de características de voz provida pelos HMM;
- Uso da base completa para o aprimoramento da síntese com emoções apresentada por [48].

Referências Bibliográficas

[1] ORCA. (27/11/2014). Projeto ORCA. Disponível em: http://live.gnome.org/Orca

[2] DOSVOX. Projeto DOSVOX. Disponível em: www.intervox.nce.ufrj.br/dosvox

[3] Microsoft. (27/11/2014). Speech API. Disponível em: http://www.microsoft.com/speech

[4] Oracle. (27/11/2014). Java Speech API. Disponível em: www.oracle.com/technetwork/java/jsapifaq-135248.html

[5] S. Pearson, H. Moran, K. Hata, and F. Holm, "Combining Concatenation and Formant Synthesis for Improved Intelligibility and Naturalness in Text-to-Speech Systems," Second ESCA/IEEE Workshop on Speech Synthesis, Nova Iorque, 1994.

[6] D. Klatt, "The Klattalk Text-to-Speech Conversion System," Acoustics, Speech, and Signal Processing, IEEE International Conference ICASSP'82, 1982.

[7] I. Couto, N. Neto, V. Tadaiesky, A. Klautau, and R. Maia, "An Open Source HMM-based Text-to-Speech System for Brazilian Portuguese," 7th International Telecommunications Symposium, Manaus, 2010.

[8] R. Damper, Data-Driven Techniques in Speech Synthesis: Kluwer Academic Publishers, 2001.

[9] K. Tokuda, T. Kobayashi, and S. Imai, "Speech parameter generation from HMM using dynamic features," Acoustics, Speech, and Signal Processing, Detroit, MI, 1995.

[10] R. Maia, H. Zen, K. Tokuda, T. Kitamura, and F. Resende Jr., "An HMM-based Brazilian Portuguese speech synthesizer and its characteristics," IEEE Journal of Communication and Information Systems, vol. 21, pp. 58-71, 2006.

[11] R. Maia, H. Zen, K. Tokuda, T. Kitamura, and F. Resende Jr., "Towards the development of a Brazilian Portuguese text-to-speech system based on HMM," European Conference on Speech Communications and Technology (EUROSPEECH), 2003, pp. 2465-2468.

[12] T. Yoshimura, T. Masuko, K. Tokuda, and T. Kitamura, "Speaker interpolation in HMM-based speech synthesis system," European Conference on Speech Comunication and Technology (EUROSPEECH), Grécia, 1997, pp. 2523-2526.

[13] T. Yamagashi, M. Tachibana, T. Masuko, and T. Kobayashi, "Speaking style adaptation using context clustering decision tree for HMM-based speech synthesis," IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2004, pp. 5-8.

[14] M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi, "Adaptation of pitch and spectrum for HMM-based speech synthesis using MLLR," IEEE International Conference on Acoustics, Speech, and Signal Processing, Salt Lake City, UT, 2001.

[15] T. Yamagashi, M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi, "A Training Method of Average Voice Model for HMM-based Speech Synthesis," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, vol. E86-A, pp. 1956-1963, 2003.

[16] T. Yamagashi and T. Kobayashi, "Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training," IEICE Transactions on Information and Systems, vol. E90-D, pp. 533-543, 2007.

[17] T. Yamagashi, T. Kobayashi, Y. Nakano, K. Ogata, and J. Isogai, "Analysis of Speaker Adaptation Algorithms for HMM-based Speech Synthesis and a Constrained SMAPLR Adaptation Algoritm," IEEE Transactions on Acoustics, Speech, Signal Processing, vol. 17, pp. 66-83, 2009.

[18] T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speaker interpolation for HMM-based speech synthesis system," Journal of the Acoustical Society of Japan (E), vol. 21, pp. 199-206, 2000.

[19] R. Maia, H. Zen, K. Tokuda, T. Kitamura, and F. Resende Jr., "Towards the development of a Brazilian Portuguese text-to-speech system based on HMM," EUROSPEECH, 2003, pp. 2465-2468.

[20] C. Weiss, R. Maia, K. Tokuda, and W. Hess, "Low resource HMM-based synthesis applied to German," Proc. of Electronic Speech Signal Processing (ESSP), 2005.

[21] O. Abdel-Hamid, S. Abdou, and M. Rashwan, "Improving Arabic HMM-based speech synthesis quality," Interspeech, 2006, pp. 1332-1335.

[22] S. Kim, J. Kim, and M. Hahn, "Implementation and Evaluation of an HMM-based Korean Speech Synthesis System," IEICE Transactions on Information and Systems, vol. E89-D, pp. 1116-1119, 2006.

[23] J. Latorre, K. Iwano, and S. Furui, "New approach to the polyglot speech generation by means of an HMM-based speaker adaptable synthesizer," Speech Communication, vol. 48, pp. 1227-1242, 2006.

[24] Y. Qian, F. Soong, Y. Chen, and M. Chu, "An HMM-based Mandarin Chinese text-to-speech system," International Symposium on Chinese Spoken Language Processing (ISCSLP), 2006.

[25] L. Gomes, E. Nagle, and J. Chiquito, "Text-to-speech conversion system for Brazilian Portuguese using a formant-based synthesis technique," SBT/IEEE International Telecomunications Symposium, São Paulo, 1998.

[26] E. Albano and P. Aquino, "Linguistic criteria for building and recording units for concatenative speech synthesis in Brazilian Portuguese," European Conference on Speech Comunication and Technology (EUROSPEECH), Rhodes, Grécia, 1997, pp. 725-728.

[27] P. Barbosa, F. Violaro, E. Albano, F. Simes, P. Aquino, S. Madureira, et al., "Aiuruete: a high quality concatenative text-to-speech system for Brazilian Portuguese with demisyllabic analysis-based units and hierarchical model of rythm

production," European Conference on Speech Comunication and Technology (EUROSPEECH), Budapeste, Hungria, 1999, pp. 2059-2062.

[28] E. Costa, A. Monte, N. Neto, and A. Klautau, "Um Sintetizador de Voz Baseado em HMMs livre: Dando Novas Vozes para Aplicações Livres no Português do Brasil," Workshop de Software Livre, 2012.

[29] R. Maia, "Speech synthesis and phonetic vocoding for Brazilian Portuguese based on parameter generation from Hidden Markov Models," Doctor of Philosophy, Department of Computer Science and Engineering, Instituto de Tecnologia de Nagoya, Nagoya, 2005.

[30] M. Nicodem, S. Kafka, R. Seara Jr., and R. Seara, "Refinamento da Segmentação Fonética em Aplicações de Síntese de Fala," Simpósio Brasileiro de Telecomunicações, Recife, Pernambuco, 2007.

[31] N. Sampaio Neto, "Ferramentas e Recursos Livres para Reconhecimento e Síntese de Voz em Português Brasileiro," Doutorado, Programa de Pós-graduação em Engenharia Elétrica, Universidade Federal do Pará, Belém, Pará, 2011.

[32] UFPA. (29/11/2014). Projeto FalaBrasil. Disponível em: http://www.laps.ufpa.br/falabrasil

[33] UFPA. (29/11/2014). SimonBR. Disponível em: http://www.laps.ufpa.br/falabrasil/simonbr.php

[34] N. Sampaio Neto, E. Sousa, V. Macedo, A. Adami, and A. Klautau, "Desenvolvimento de Software Livre Usando Reconhecimento e Síntese de Voz: O Estado da Arte para o Português Brasileiro," Workshop de Software Livre, 2005.

[35] A. Alcaim, J. Solemicz, and J. de Morais. (1992) Frequência de ocorrência de fones e listas de frases foneticamente balanceadas para o português falado no Rio de Janeiro. Revista da Sociedade Brasileira de Telecomunicações 7. 23-41.

[36] Universidade de Carnegie Mellon. CMU_ARCTIC. Disponível em: http://festvox.org/cmu_arctic/

[37] Universidade de Mons. The MBROLA Project. Disponível em: http://tcts.fpms.ac.be/synthesis/mbrola.html

[38] Universidade da Pensilvânia. (29/11/2014). LDC: Linguistic Data Consortium.

[39] Universidade de Ciência e Saúde do Óregon. (29/11/2014). The Spoltech Brazilian Portuguese v1.0. Disponível em: http://www.cslu.ogi.edu/corpora/spoltech/

[40] C. Ynoguti and F. Violaro, "A Brazilian Portuguese Speech Database," Simpósio Brasileiro de Telecomunicações, Rio de Janeiro, Brasil, 2008.

[41] V. Serrani and L. Uebel. (2011) Bancos de Fala para o Português Brasileiro. LinguaMÁTICA. 69-75.

[42] M. Schröder, "Emotional speech synthesis: a review," European Conference on Speech Communication and Technology (EUROSPEECH), Aalborg, Dinamarca, 2001.

[43] A. Black, "Unit selection and emotional speech," European Conference on Speech Communication and Technology (EUROSPEECH), Geneva, Suíça, 2003, pp. 1649-1652.

[44] N. Campbell, "Towards synthesizing expressive speech: designing and collective expressive speech data," European Conference on Speech Communication and Technology (EUROSPEECH), Madri, Espanha, 2003, pp. 1637-1640.

[45] E. Douglas-Cowie, N. Campbell, R. Cowie, and P. Roach. (2003) Emotional speech: Towards a new generation of databases. Speech Communication. 33-60.

[46] K. Shichiri, A. Sawabe, T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, et al., "Eigenvoices for HMM-based speech synthesis," International Conference on Spoken Language Processing (ICSLP), 2002, pp. 1269-1272.

[47] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Mixed-excitation for HMM-based speech synthesis," European Conference on Speech Communication and Technology (EUROSPEECH), Escandinávia, 2001.

[48] D. Silva, "Algoritmos de processamento da linguagem e síntese de voz com emoções aplicados a um conversor texto-fala baseado em HMM," Doutorado, Programa de Engenharia Elétrica, Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia (COPPE/UFRJ), Rio de Janeiro, 2011.

[49] A. Teixeira, R. Martinez, L. Silva, L. Jesus, J. Príncipe, and F. Vaz, "Simulation of Human Speech Production Applied to the Study and Synthesis of European Portuguese," EURASIP Journal on Applied Signal Processing, vol. 9, pp. 1435-1448, 2005.

[50] J. Deller, J. Proakis, and J. Hansen, Discrete-time processing of speech signals. EUA: Macmillan, 1993.

[51] L. Rabiner and B. Juang, Fundamentals of Speech Recognition. New Jersey: Prentice Hall, 1993.

[52] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2000, pp. 1315-1318.

[53] K. Oura. (2011, List of modifications made in HTS (for version 2.2). Disponível em: http://hts.sp.nitech.ac.jp/archives/2.2/HTS_Document.pdf

[54] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," European Conference on Speech Communication and Technology (EUROSPEECH), 1999, pp. 2347-2350.

[55] K. Tokuda, H. Zen, and A. Black, "An HMM-based speech synthesis system applied to English," IEEE Speech Synthesis Workshop, 2002.

[56] A. Singh. (2005, 08/12/2014). The EM Algorithm. Disponível em: http://www.cs.mu.edu/~awm/10701/assignments/EM.pdf

[57] T. Toda, K. Tokuda, and A. Black, "Speech parameter generation algorithm considering global variance for HMM-based speech synthesis," European Conference on Speech Communication and Technology (EUROSPEECH), Lisboa, 2005, pp. 2801-2804.

[58] T. Toda and K. Tokuda, "A Speech Parameter Generation Algorithm considering Global Variance for HMM-Based Speech Synthesis," IEICE Transactions on Information and Systems, vol. E90-D, pp. 816-824, 2007.

[59] T. Toda, A. Black, and K. Tokuda, "Spectral conversion based on maximum likelihood estimation considering global variance of converted parameter," International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Filadélfia, 2005, pp. 9-12.

[60] J. Solewicz, A. Alcaim, and J. Moraes, "Text-to-speech system for brazilian portuguese using a reduced set of synthesis unit," International Symposium on Speech, Image Processing and Neural Networks, Hong Kong, 1994, pp. 579-582.

[61] F. Egashira and F. Violaro, "Conversor texto-fala para a língua portuguesa," Simpósio Brasileiro de Telecomunicações, 1995, pp. 71-76.

[62] P. Barbosa, "A model of segment (and pause) duration generation for Brazilian Portuguese text-to-speech synthesis," European Conference on Speech Communication and Technology (EUROSPEECH), 1997, pp. 2655-2658.

[63] J. de Morais, "Vowel nasalization in Brazilian Portuguese," European Conference on Speech Communication and Technology (EUROSPEECH), 1997, pp. 733-736.

[64] S. Kafka, F. Pacheco, I. Seara, S. Klein, and R. Seara, "Utilização de segmentos transicionais homorgânicos em síntese de fala concatenativa," Congresso Brasileiro de Automática (CBA), 2002, pp. 2742-2747.

[65] I. Seara, S. Kafka, S. Klein, and R. Seara. (2002) Alternância vocálica das formas verbais e nominais do português brasileiro para aplicação em conversão texto-fala. Revista da Sociedade Brasileira de Telecomunicações. 79-85.

[66] F. Barbosa, G. Pinto, F. Resende Jr., C. Gonçalves, R. Monserrat, and R. Rosa, "Grapheme-phone transcription algorithm for a Brazilian Portuguese TTS," Workshop on Computational Processing of Written and Spoken Portuguese (PROPOR), 2003.

[67] F. Barbosa, R. Maia, and F. Resende Jr., "Análise comparativa do impacto da classe gramatical em sistemas TTS baseados em HMMs," Simpósio Brasileiro de Telecomunicações (SBrT), 2004.

[68] R. Seara Jr., I. Seara, S. Kafka, F. Pacheco, R. Seara, and S. Klein, "Parâmetros linguísticos utilizados para a geração automática de prosódia em sistemas de síntese de fala," Simpósio Brasileiro de Telecomunicações (SBrT), 2004.

[69] I. Seara, M. Nicodem, R. Seara, and R. Seara Jr., "Classificação sintagmática focalizando a síntese de fala: Regras para o Português Brasileiro," Simpósio Brasileiro de Telecomunicações (SBrT), 2007, pp. 1-6.

[70] R. Martins, "Projeto, gravação e edição de base de voz para aplicações em Síntese e Reconhecimento da Fala," Trabalho de Conclusão de Curso, Departamento de Engenharia Eletrônica e de Computação, Universidade Federal do Rio de Janeiro, 2011.

[71] D. Silva, D. Braga, and F. Resende Jr., "Conjunto de regras para desambiguação de homógrafos heterófonos do Português Brasileiro," Simpósio Brasileiro de Telecomunicações, Blumenau, 2009, pp. 1-6.

[72] C. Shulby, G. Mendonça, and V. Marquiafável, "Automatic disambiguation of homographic heterophone pairs containing open and closed mid vowels," in Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana, Fortaleza, CE, 2013, pp. 126-137.

[73] I. Seara, S. Kafka, S. Klein, and R. Seara, "Considerações sobre os problemas da alternância vocálica das formas verbais do Português falado no Brasil para aplicação em um sistema de conversão texto-fala," Simpósio Brasileiro de Telecomunicações, 2001.

[74] L. Ferrari, F. Barbosa, and F. Resende Jr., "Construções gramaticais e sistemas de conversão texto-fala: o caso dos homógrafos," International Conference on Cognitive Linguistics, 2003.

[75] P. Arantes and P. Barbosa, "Secondary stress in Brazilian Portuguese: the interplay between production and perception studies," Speech Prosody, Dresden, 2006, pp. 73-76.

[76] J. Teixeira, E. Paulo, D. Freitas, and M. Pinto, "Acoustical characterisation of accented syllable in Portuguese; a contribution to the naturalness of speech synthesis," European Conference on Speech Communication and Technology, Hungria, 1999.

[77] M. O'Dell, M. Lennes, S. Werner, and T. Nieminen, "Looking for rhythm in conversational speech," in International Congress of Phonetic Sciences, Saarbrücken, 2007, pp. 1201-1204.

[78] E. Bechara, Moderna Gramática Portuguesa. Rio de Janeiro, RJ, Brasil: Lucerna, 2002.

[79] (11/12/2014). "Speech Assessment Methods Phonetic Alphabet (SAMPA). Disponível em: http://www.phon.ucl.ac.uk/home/sampa/index.html

[80] (11/12/2014). The Festival Speech Synthesis System. Disponível em: http://www.festvox.org/festival/

[81] T. Fukada, K. Tokuda, T. Kobayashi, and S. Imai, "An adaptive algorithm for mel-cepstral analysis of speech," International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 1992.

[82] H. Kawahara, I. Masuda-Katsuse, and A. Cheveigné. (1999) Reconstructing speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: possible role of a repetitive structure in sounds. Speech Communication. 187-207.

[83] J. Odell, "The use of context in large vocabulary speech recognition," PhD, Universidade de Cambridge, 1995.

[84] K. Tokuda, N. Miyazaki, and T. Kobayashi, "Multi-space probability distribution HMM," IEICE Transactions on Information and Systems, vol. E85-D, pp. 455-464, 2002.

[85] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Hidden semi-Markov model based speech synthesis," International Conference on Spoken Language Processing (ICSLP), 2004.

[86] Instituto de Tecnologia de Nagoya. (17/01/2015). HMM-based speech synthesis system (HTS). Disponível em: http://hts.sp.nitech.ac.jp

[87] Cisco. (17/01/2015). Understanding Codecs: Complexity, Hardware Support, MOS, and Negotiation. Disponível em: http://www.cisco.com/c/en/us/support/docs/voice/h323/14069-codec-complexity.html#mos

[88] S. Mukherjee and S. Mandal, "Generation of F_0 contour using deep Boltzmann Machine in twin Gaussian process hybrid model for Bengali Language," Interspeech, 2014, pp. 2445-2449.

[89] R. Cirigliano, C. Monteiro, F. Barbosa, F. Resende Jr., L. R. Couto, J. A. Moraes, "Um Conjunto de 1000 Frases Foneticamente Balanceadas para o Português Brasileiro Obtido Utilizando a Abordagem de Algoritmos Genéticos", Simpósio Brasileiro de Telecomunicações (SBrT), Campinas, Brasil, 2005, pp. 544-549.

[90] R. Shiffler, "Maximum Z score and outliers", The American Statistician, Vol. 42, No. 1, pp. 79-80, 1988.

[91] B. Iglewicz, and D. Hoaglin, How to detect and handle outliers, EUA: ASQC Quality Press, 1993.

[92] Ubuntu. (20/01/2015). Ubuntu. Disponível em: http://www.ubuntu.com

[93] Universidade de Cambridge. (20/01/2015). HTK. Disponível em: http://htk.eng.cam.ac.uk

[94] Instituto de Tecnologia de Nagoya. (20/01/2015). SPTK. Disponível em: http://sp-tk.sourceforge.net

[95] Instituto de Tecnologia de Nagoya. (20/01/2015). HTS Engine. Disponível em: http://hts-engine.sourceforge.net

[96] Active State. (20/01/2015). Active Tcl. Disponível em: http://www.activestate.com/activetcl

[97] L. Baum, “An equality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes,” Inequalities, vol. 3, pp. 1-8, 1972.

[98] G. Arfken, Mathematical Methods for Physicists, EUA: Academic Press, 1985.

[99] L. Vecchietti, “Processamento de uma nova base de voz com aplicação em Síntese da Fala utilizando Modelos Ocultos de Markov,” Projeto Final em andamento.

[100] Instituto de Tecnologia de Nagoya. (24/01/2015). HTS-Users Mailing List Archive. Disponível em: http://hts.sp.nitech.ac.jp/hts-users

Apêndice A

Prova da desigualdade de Jensen

Este apêndice tem o intuito de apresentar a prova por indução da desigualdade de Jensen.

A.1 Enunciado

Seja f uma função convexa definida num intervalo D. Se $\{x_i\}_{i=1}^{N} \in D$ e $\{c_i\}_{i=1}^{N}$, tal que $c_i \geq 0$ para todo $1 \leq i \leq N$, e $\sum_{i=1}^{N} c_i = 1$. Tem-se:

$$f\left(\sum_{i=1}^{N} c_i x_i\right) \leq \sum_{i=1}^{N} c_i f(x_i) \tag{a.1}$$

A.2 Prova

Utiliza-se a prova por indução. Assim:

a) Para $N = 1$:

Substituindo $N = 1$ em (a.1), tem-se:

$$f(c_1 x_1) \leq c_1 f(x_1)$$

onde, pelas restrições expressas sobre as constantes c_i, teremos $c_1 = 1$ e, portanto, a expressão (trivial) verdadeira:

$$f(x_1) \leq f(x_1).$$

b) Para $N = 2$:

Substituindo $N = 2$ em (a.1), tem-se:

$$f(c_1 x_1 + c_2 x_2) \leq c_1 f(x_1) + c_2 f(x_2) \tag{a.2}$$

onde, pelas restrições expressas sobre as constantes c_i, teremos que $c_1 + c_2 = 1$ e $c_1, c_2 \geq 0$. A expressão acima juntamente com as restrições expostas compõem a definição de convexidade. Como f é convexa, tem-se que a proposição é verdadeira para $N = 2$.

c) Para $N = n$:

Neste ponto, assume-se que

$$f\left(\sum_{i=1}^{n} c_i x_i\right) \leq \sum_{i=1}^{n} c_i f(x_i) \tag{a.3}$$

é verdadeira.

d) Para $N = n + 1$:

Fazendo a substituição $N = n + 1$ no termo à esquerda de (a.1), tem-se:

$$f\left(\sum_{i=1}^{n+1} c_i x_i\right) = f\left(c_{n+1} x_{n+1} + \sum_{i=1}^{n} c_i x_i\right)$$

$$f\left(\sum_{i=1}^{n+1} c_i x_i\right) = f\left(c_{n+1} x_{n+1} + (1 - c_{n+1}) \frac{1}{(1 - c_{n+1})} \sum_{i=1}^{n} c_i x_i\right) \tag{a.4}$$

Adotando $x_1 = x_{n+1}$ e $x_2 = \frac{1}{(1-c_{n+1})} \sum_{i=1}^{n} c_i x_i$ em (a.2), pode-se estabelecer a seguinte relação a partir de (a.4):

$$f\left(\sum_{i=1}^{n+1} c_i x_i\right) = f\left(c_{n+1} x_{n+1} + (1 - c_{n+1}) \frac{1}{(1 - c_{n+1})} \sum_{i=1}^{n} c_i x_i\right)$$

$$f\left(\sum_{i=1}^{n+1} c_i x_i\right) \leq c_{n+1} f(x_{n+1}) + (1 - c_{n+1}) f\left(\frac{1}{(1 - c_{n+1})} \sum_{i=1}^{n} c_i x_i\right)$$

$$f\left(\sum_{i=1}^{n+1} c_i x_i\right) \leq c_{n+1} f(x_{n+1}) + (1 - c_{n+1}) f\left(\sum_{i=1}^{n} \frac{c_i}{(1 - c_{n+1})} x_i\right) \tag{a.5}$$

Assim, chamemos c'_i os coeficientes do somatório presente no termo mais à direita de (a.5), onde $c'_i = \frac{c_i}{(1-c_{n+1})}$. Note que os "novos" coeficientes c'_i cumprem a restrição de que $c'_i \geq 0$ para $1 \leq i \leq n$ e $\sum_{i=1}^{n} c'_i = 1$. Assim, por (a.3) pode-se estabelecer a seguinte relação:

$$f\left(\sum_{i=1}^{n+1} c_i x_i\right) \leq c_{n+1} f(x_{n+1}) + (1 - c_{n+1}) f\left(\sum_{i=1}^{n} c'_i x_i\right)$$

$$f\left(\sum_{i=1}^{n+1} c_i x_i\right) \leq c_{n+1} f(x_{n+1}) + (1 - c_{n+1}) \sum_{i=1}^{n} c'_i f(x_i)$$

$$f\left(\sum_{i=1}^{n+1} c_i x_i\right) \leq c_{n+1} f(x_{n+1}) + \sum_{i=1}^{n} (1 - c_{n+1}) c'_i f(x_i)$$

$$f\left(\sum_{i=1}^{n+1} c_i x_i\right) \leq c_{n+1} f(x_{n+1}) + \sum_{i=1}^{n} c_i f(x_i)$$

$$f\left(\sum_{i=1}^{n+1} c_i x_i\right) \leq \sum_{i=1}^{n+1} c_i f(x_i) \tag{a.6}$$

Daí, temos que (a.1) é válida para $N = n + 1$, concluindo, assim, a prova de que a desigualdade de Jensen é verdadeira.

Vale, ainda, ressaltar que, como a definição de função côncava é semelhante à de convexa, apenas com o sinal de desigualdade invertido, isto é, as funções que cumprem

$$f(c_1 x_1 + c_2 x_2) \geq c_1 f(x_1) + c_2 f(x_2),$$

onde $c_1 + c_2 = 1$ e $c_1, c_2 \geq 0$, é possível aplicar a desigualdade de Jensen a elas através da inversão do sinal de desigualdade. Desse modo, para funções côncavas, a desigualdade se torna

$$f\left(\sum_{i=1}^{N} c_i x_i\right) \geq \sum_{i=1}^{N} c_i f(x_i) \tag{a.7}$$

para $\{c_i\}_{i=1}^{N}$, tal que $c_i \geq 0$ para todo $1 \leq i \leq N$, e $\sum_{i=1}^{N} c_i = 1$.

Apêndice B

Edição de áudio utilizando ProTools 10.0

Este apêndice tem o intuito de apresentar como foi realizada a edição de áudio através do software ProTools 10.0. Para tal, pressupõe-se que o referido software esteja instalado na máquina do operador, bem como estejam disponíveis a interface de áudio e as caixas de som necessárias à sua utilização.

B.1 Criação de Sessão

Ao proceder a abertura do ProTools, imediatamente apresenta-se a caixa de diálogo ilustrada na Fig. B.1, onde, para o caso de criação de sessão, deve ser selecionado o campo "Create Blank Session…". Se desejar prosseguir trabalho anteriormente iniciado, basta selecionar "Open Session…".

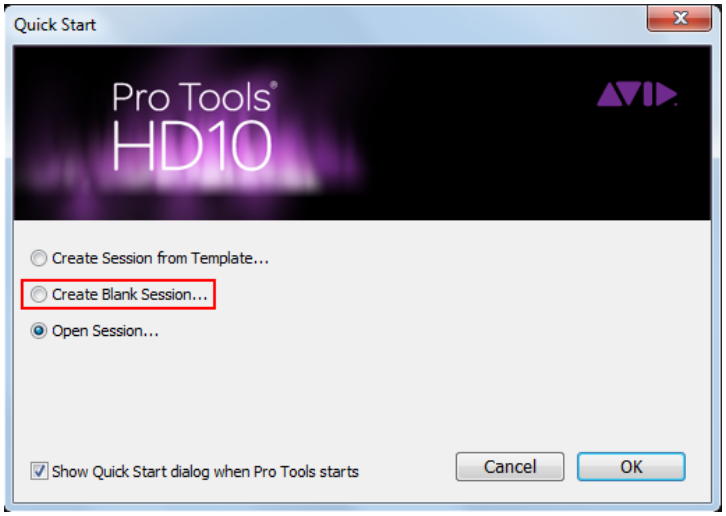


Fig. B.1 Caixa de diálogo inicial

Quando marcada a opção, basta clicar em "OK". Para uma nova sessão, será solicitada a inserção do nome da mesma (Fig. B.2).

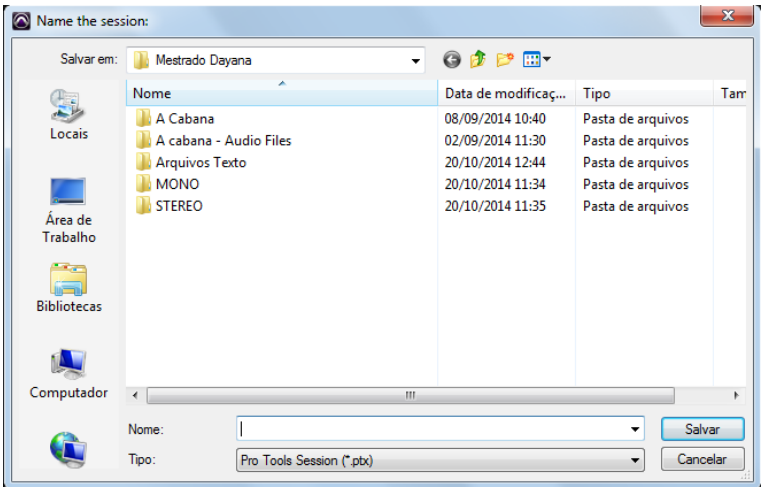


Fig. B.2 Caixa de diálogo para salvar nome de nova sessão

Após salvar o nome da sessão, abrir-se-á uma janela de edição conforme Fig. B.3.

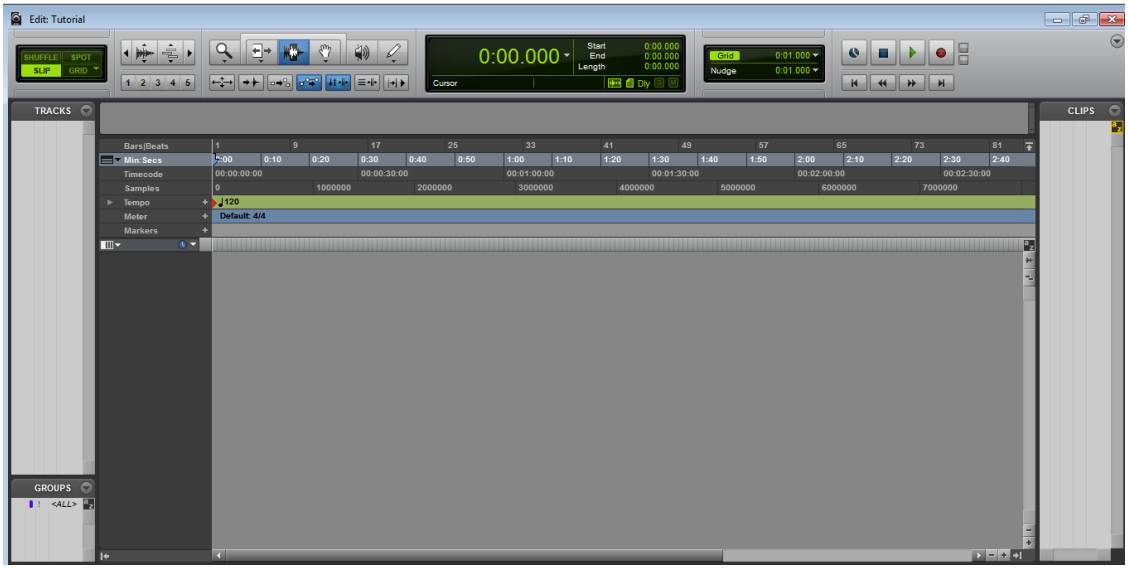


Fig. B.3 Janela de edição

Para a inclusão de áudio na sessão, deve-se ir até o menu “File”, selecionar “Import” e, a seguir, “Audio” (Fig. B.4).

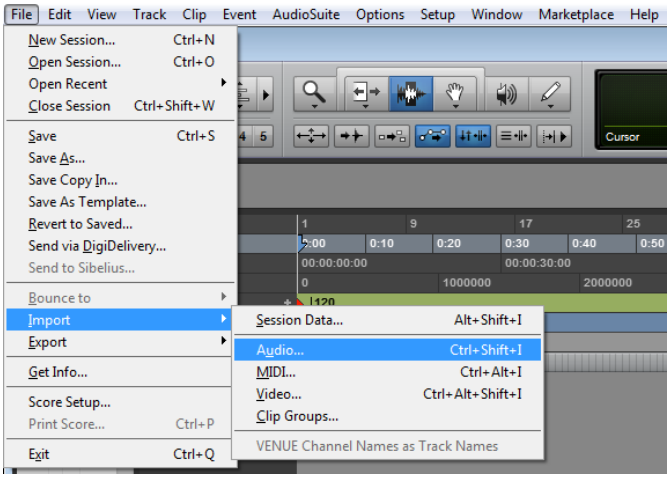


Fig. B.4 Inclusão de áudio na sessão

A seguir, será apresentada uma caixa de diálogo (Fig. B.5), onde é possível navegar entre as pastas do computador através do campo “Examinar”. O conteúdo das pastas é visualizado na coluna mais à esquerda. Após duplo clique no conteúdo desejado (passo “1” da Fig. B.5), o áudio é apresentado na coluna “clip” do meio da caixa. As informações de conteúdo (início, fim e comprimento do áudio) são detalhadas abaixo da referida coluna. Para inclusão do arquivo na trilha de edição, basta selecioná-lo nesta coluna e clicar em “Add clip” (passo “2” da Fig. B.5). O título do arquivo, então, será copiado para a coluna “clip” à direita. O processo pode ser repetido quantas vezes for necessário. Para remover áudio incluso na trilha, deve-se selecioná-lo na coluna mais à direita e clicar em “Remove”. Ao término da seleção, clicar em “Done” e serão vistas as opções de importação de áudio (Fig. B.6). Marcar “New Track” para a adição de uma nova trilha para cada áudio e, no campo “Location”, selecionar “Session Start” para que a(s) mesma(s) seja(m) apresentadas na sessão inicial. Por fim, confirmar com “OK”.

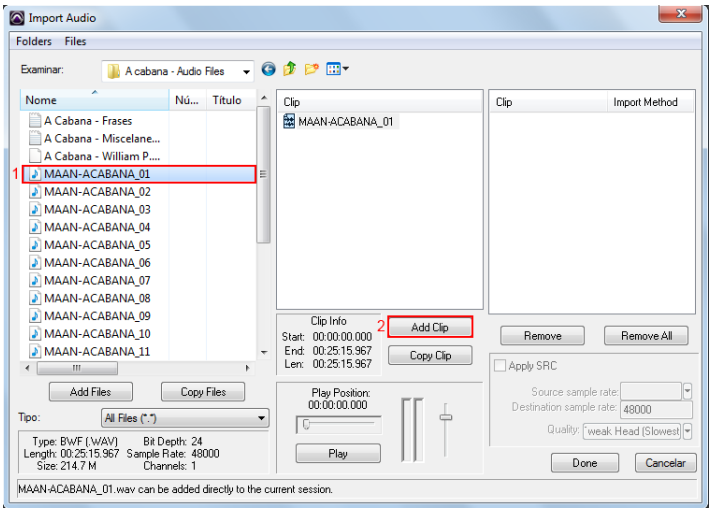


Fig. B.5 Caixa de diálogo para importação de áudio

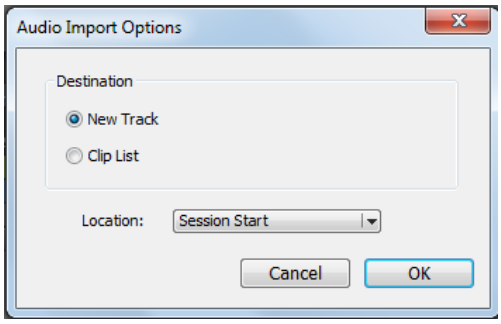


Fig. B.6 Opções de importação de áudio

O(s) áudio(s) selecionado(s) pode(m) ser visto(s), em seguida, na janela de edição (Fig. B.7).

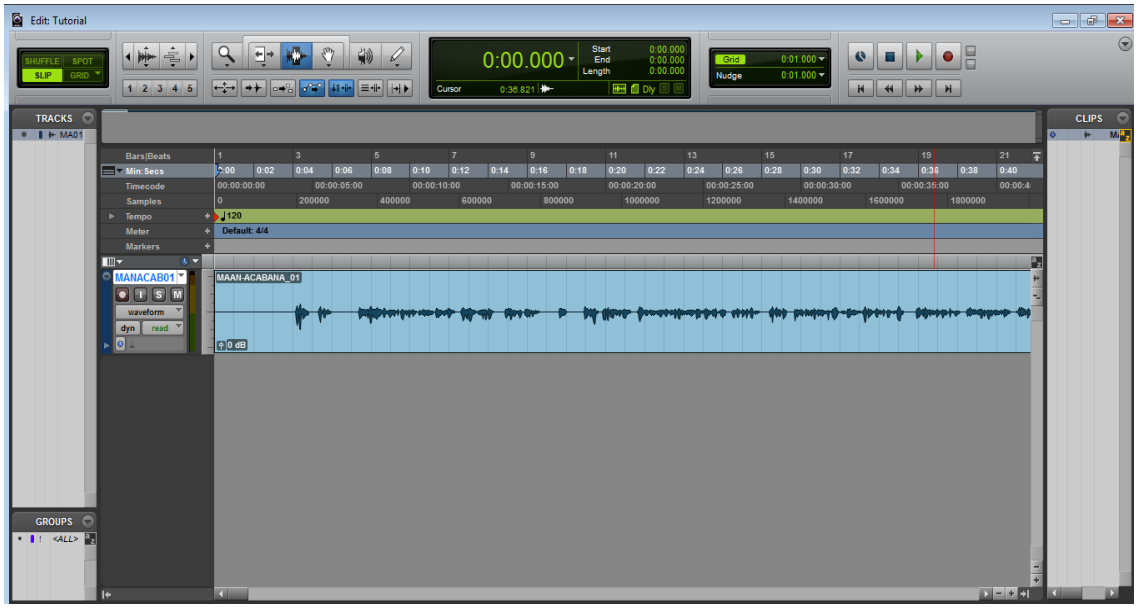


Fig. B.7 Janela de edição com nova trilha de áudio

Ao pressionar a barra de espaço ou clicar no botão de *play*, típico de tocadores de áudio, localizado no canto superior direito da tela de edição, o áudio é tocado. A

localização na trilha do momento correspondente à audição em determinado instante pode ser acompanhada através da barra vertical que "anda" à medida que o som é tocado. Esta barra é indicada na Fig. B.8.

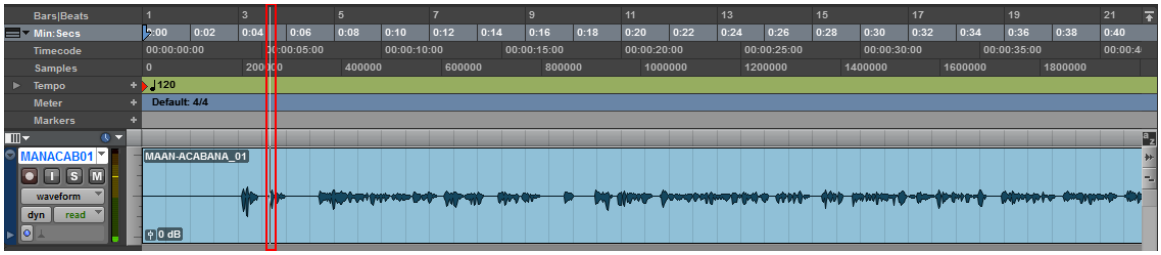


Fig. B.8 Barra na trilha para acompanhamento do áudio tocado

B.2 Segmentação das frases

Para a edição de áudio realizada neste trabalho, foram utilizadas ferramentas básicas do ProTools. A primeira delas é a ferramenta de seleção de trecho (Fig. B.9), localizada no menu acima das trilhas. Para utilizá-la, basta acioná-la e marcar o trecho desejado arrastando o cursor sobre a trilha com o *mouse* pressionado. Um exemplo de trecho selecionado é exibido na Fig. B.10.

Fig. B.9 Menu com ferramenta de seleção de trecho destacada

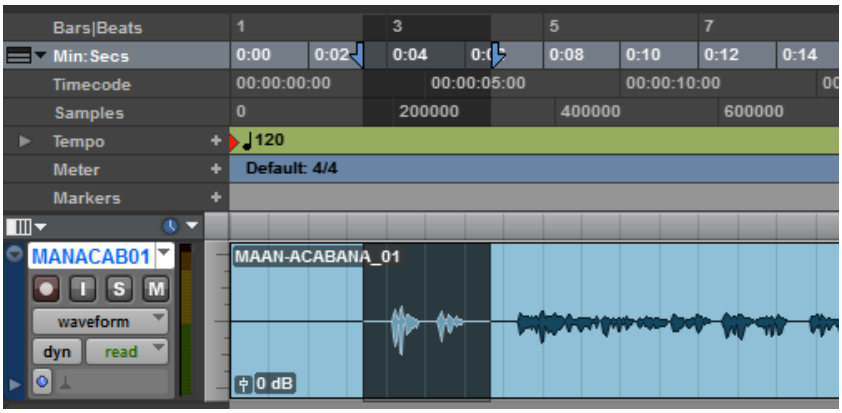


Fig. B.10 Trecho selecionado na trilha (fundo mais escuro)

Para segmentar as frases do texto lido continuamente e gravado em um único arquivo, a ferramenta de seleção de trecho foi utilizada em conjunto com a exportação do mesmo. Assim, após a marcação do intervalo desejado, deve-se ir em "File", no menu principal do ProTools, e selecionar "Bounce to" seguido de "Disk…" (Fig. B.11).

A seguir, é aberta uma caixa de diálogo (Fig. B.12) intitulada “Bounce”. Nela, escolhe-se “de HD Audio 1 (Mono)” como “Bounce Source” e “Mono (summed)” como “Format”. Os demais valores são mantidos como padrão, isto é, formato de arquivo (“File Type”) “wav”, precisão (“Bit Depth”) de 24 bits e taxa de amostragem (“Sample Rate”) de 48 kHz. Para confirmar o preenchimento, basta clicar em “Bounce”.

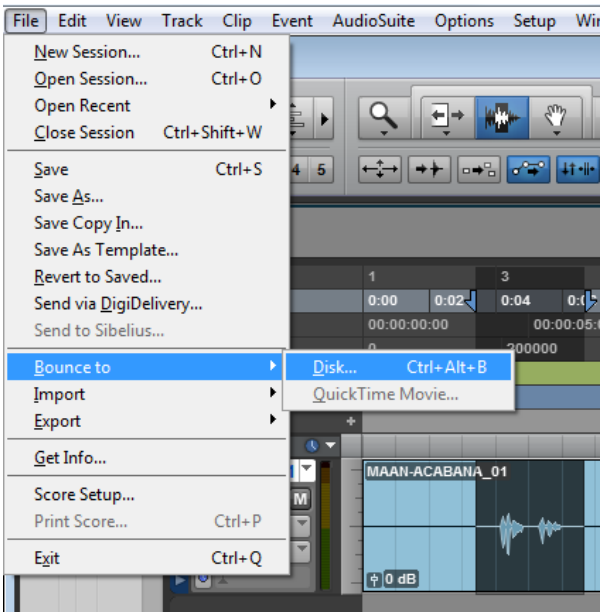


Fig. B.11 Utilização da ferramenta para exportação de trecho

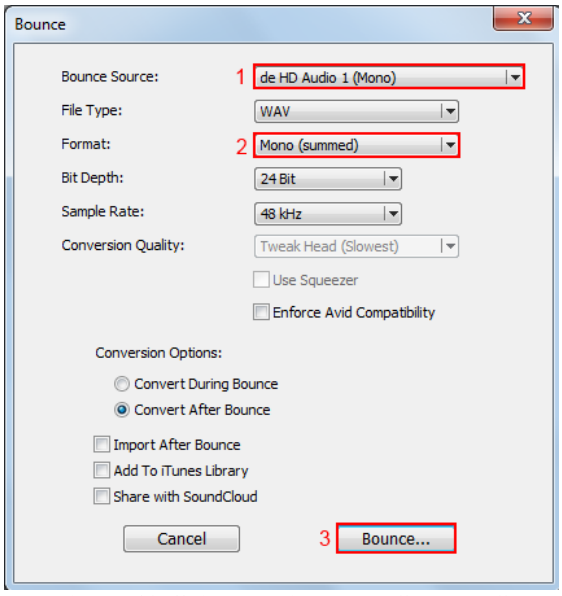


Fig. B.12 Caixa de diálogo de "Bounce"

Será, então, aberta uma janela (Fig. B.13) onde é solicitada a inserção de um nome para o arquivo exportado. Ao incluí-lo, clicar em “Salvar”.

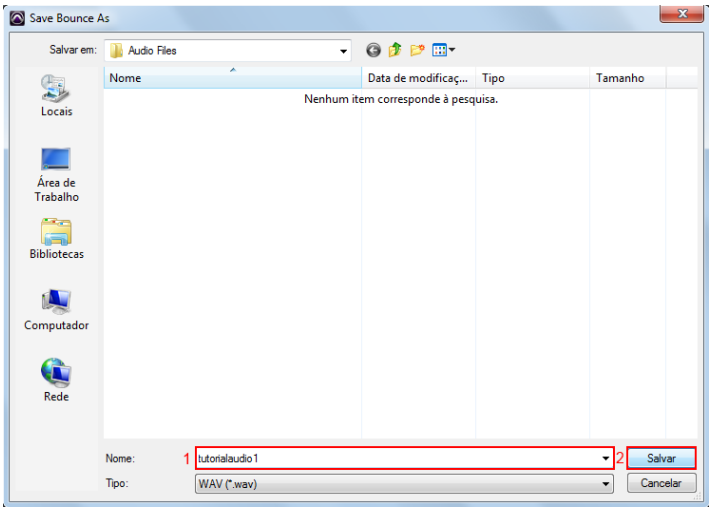


Fig. B.13 Inserção de nome de arquivo a ser exportado

B.3 Edição de áudio

Com frequência, durante a segmentação de frases gravadas continuamente, é possível identificar trechos de ruído. Esse ruído pode ser oriundo do ambiente externo, da respiração do locutor, palavras repetidas durante a leitura, entre outros fatores. Para contornar esse problema, faz-se uso de ferramentas de edição que permitem excluir trechos isolados onde haja presença exclusiva do ruído.

A primeira delas é a o modo de edição “SLIP” (Fig. B.14), localizada no menu do canto superior esquerdo da janela de edição. Com ele, pode-se criar trechos de silêncio onde antes havia ruído, sendo interessante para os casos onde há presença da respiração do locutor ou influências do meio externo quando não há fala envolvida.

Fig. B.14 Menu onde é possível selecionar o modo de edição "SLIP"

O outro modo de edição de interesse é o “SHUFFLE” (Fig. B.15). Neste caso, é possível apagar trechos e unir seus extremos, como numa “emenda”. É possível utilizar este modo em situações onde o locutor, por exemplo, duplicou palavras ou trechos durante a sua fala.

Fig. B.15 Modo "SHUFFLE" selecionado

Para uso destes modos de edição, deve-se selecionar aquele desejado e, a seguir, destacar o trecho que se deseja remover com o auxílio da ferramenta de seleção vista na seção anterior. Por fim, pressione o botão “DELETE” de seu teclado. Este procedimento é ilustrado na Fig. B.16. O resultado para modo “SLIP” está na Fig. B.17 e para “SHUFFLE” na Fig. B.18.

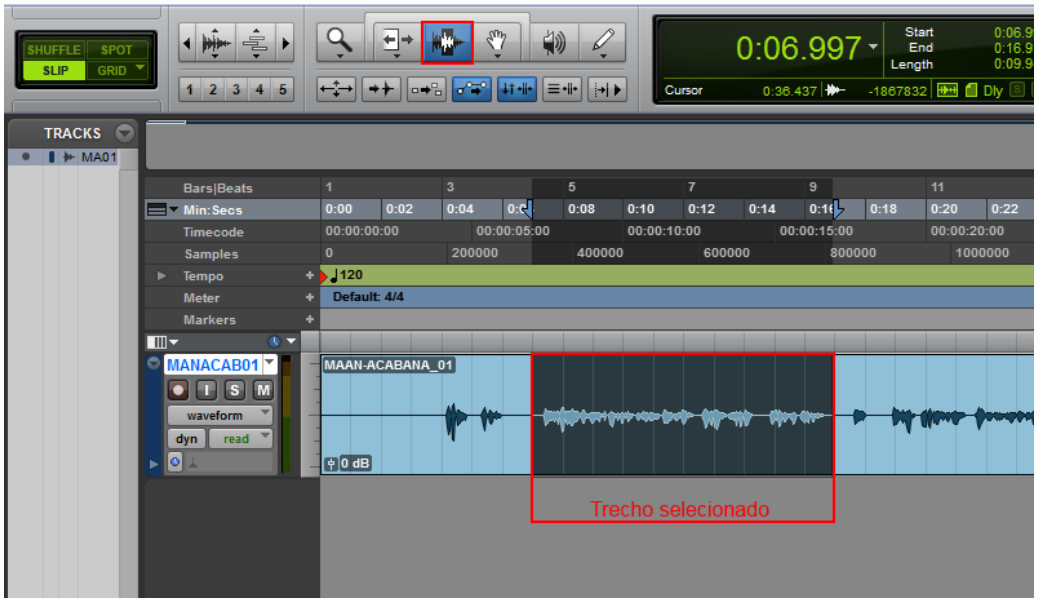


Fig. B.16 Edição utilizando modo "SLIP"

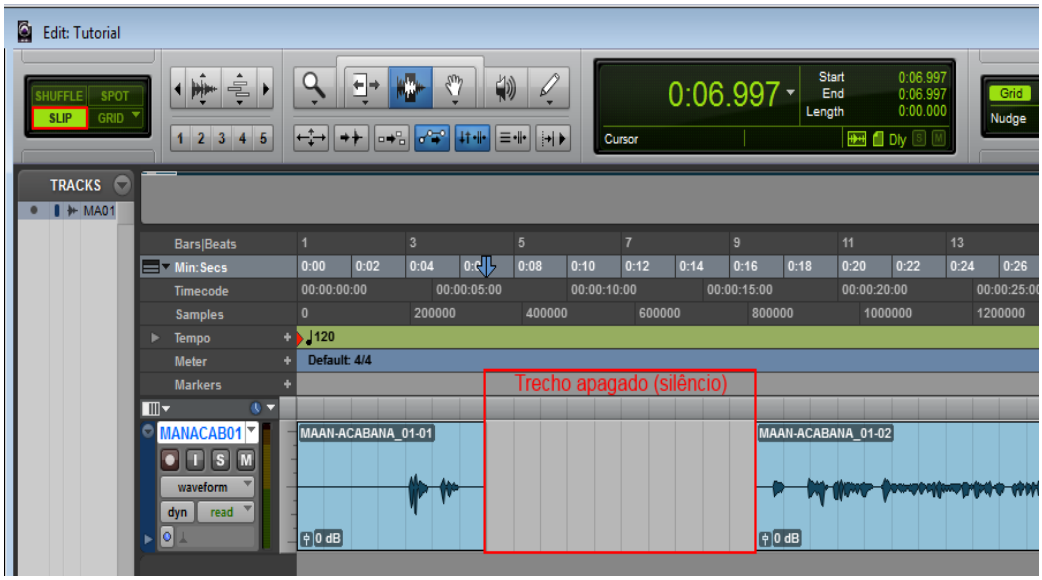


Fig. B.17 Resultado da edição em modo "SLIP"

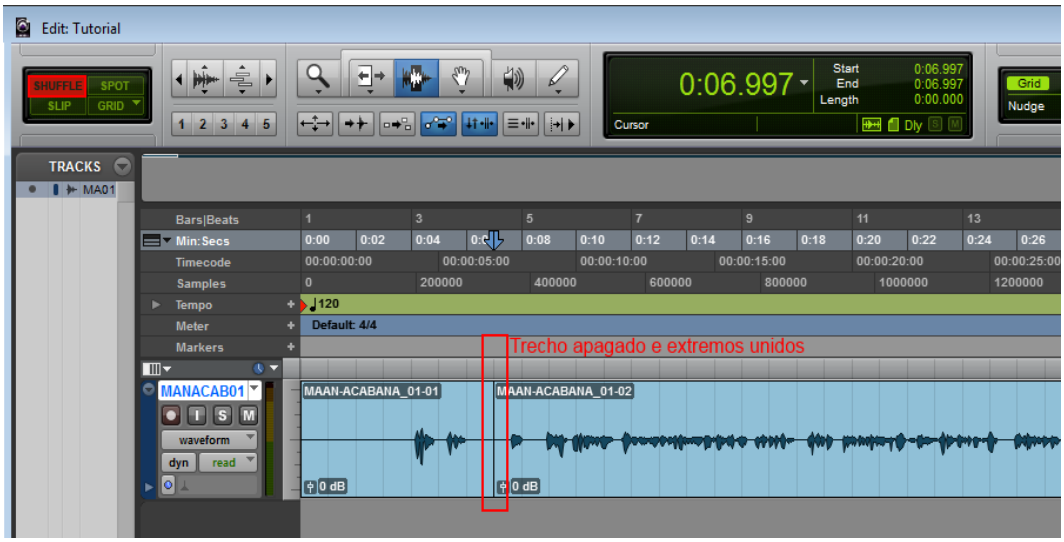


Fig. B.18 Resultado da edição em modo "SHUFFLE"

B.4 Funcionalidades úteis

Neste ponto são apresentadas, de modo sucinto, algumas funcionalidades úteis durante a edição de áudio no ProTools.

Navegação na trilha: funcionalidade realizada através da barra de rolagem localizada no extremo inferior da janela de edição (Fig. B.19).

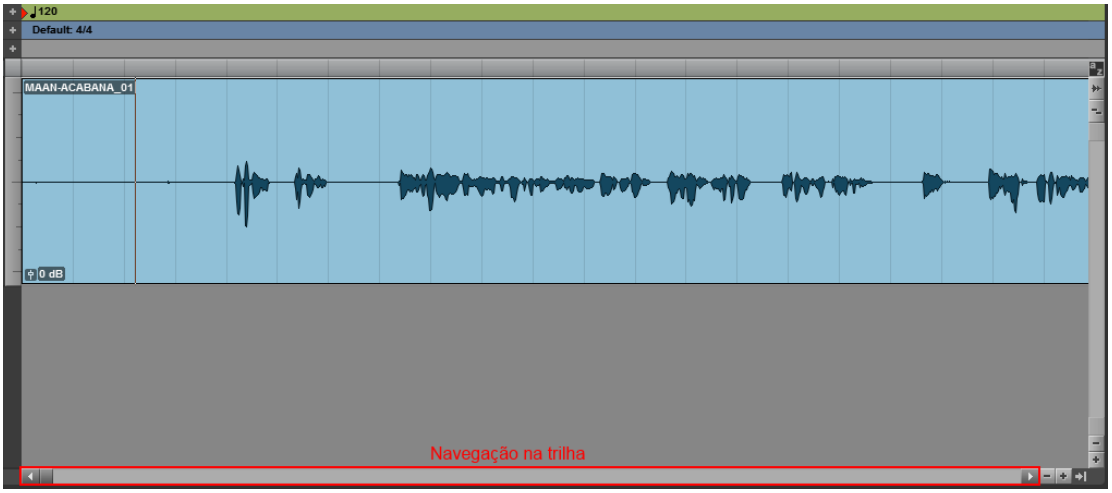


Fig. B.19 Navegação na trilha de edição

Zoom horizontal: é a ferramenta que viabiliza o alongamento gráfico da linha do tempo (eixo horizontal, permitindo melhor visualização de detalhes devido ao maior espaçamento entre as unidades de tempo. Esta ferramenta pode ser ativada pela lupa no menu superior à trilha (Fig. B.20) ou através dos botões de "+" e "-" localizados à direita da barra de rolagem (Fig. B.21).

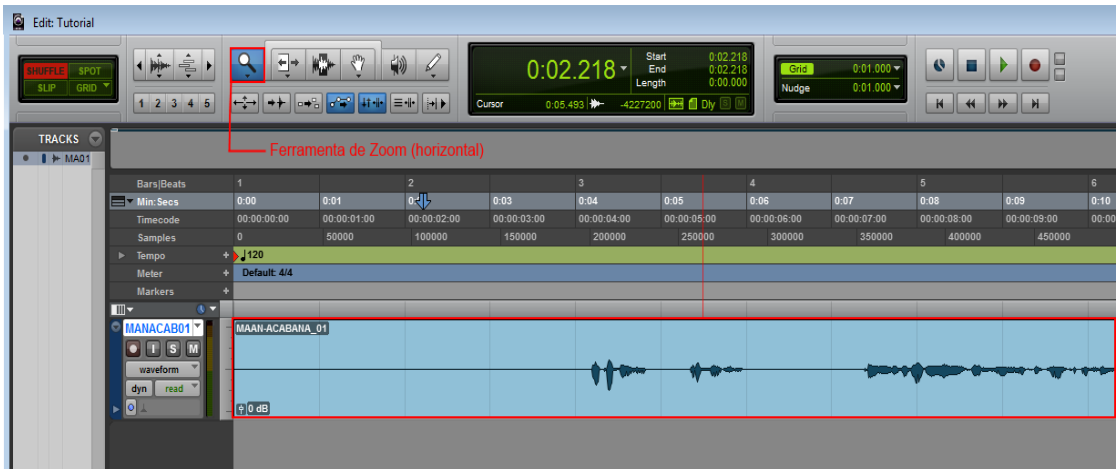


Fig. B.20 Ferramenta de *zoom* horizontal

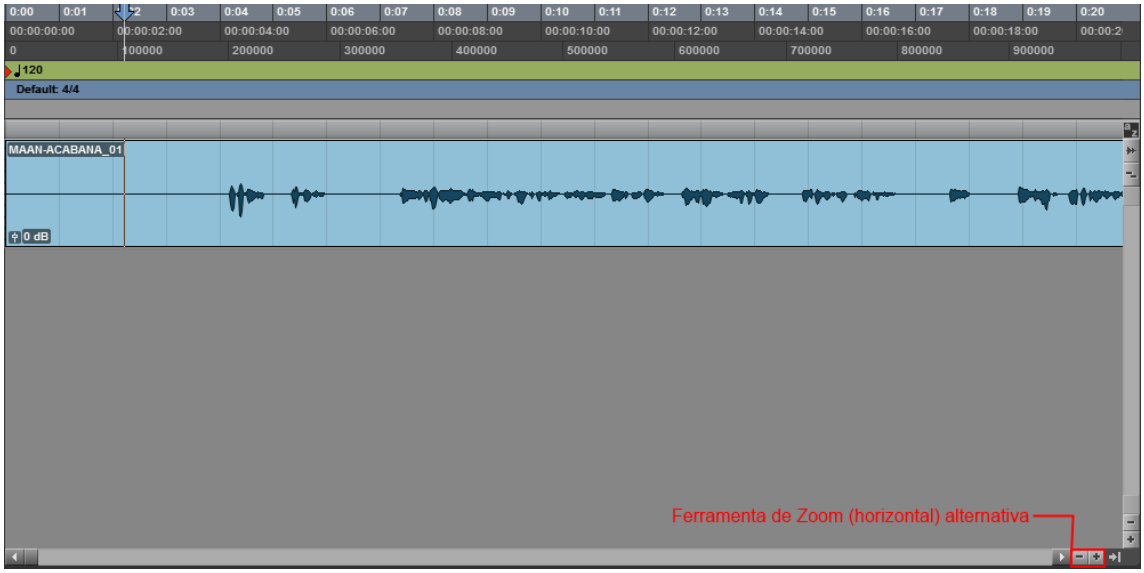


Fig. B.21 Alternativa da ferramenta de *zoom* horizontal

Zoom vertical: análoga ao *zoom* horizontal, possibilitando observar detalhes do sinal de áudio, através de sua ampliação gráfica na direção do eixo vertical. Seu uso é feito através dos botões de "+" e "-" localizados conforme a Fig. B.22.

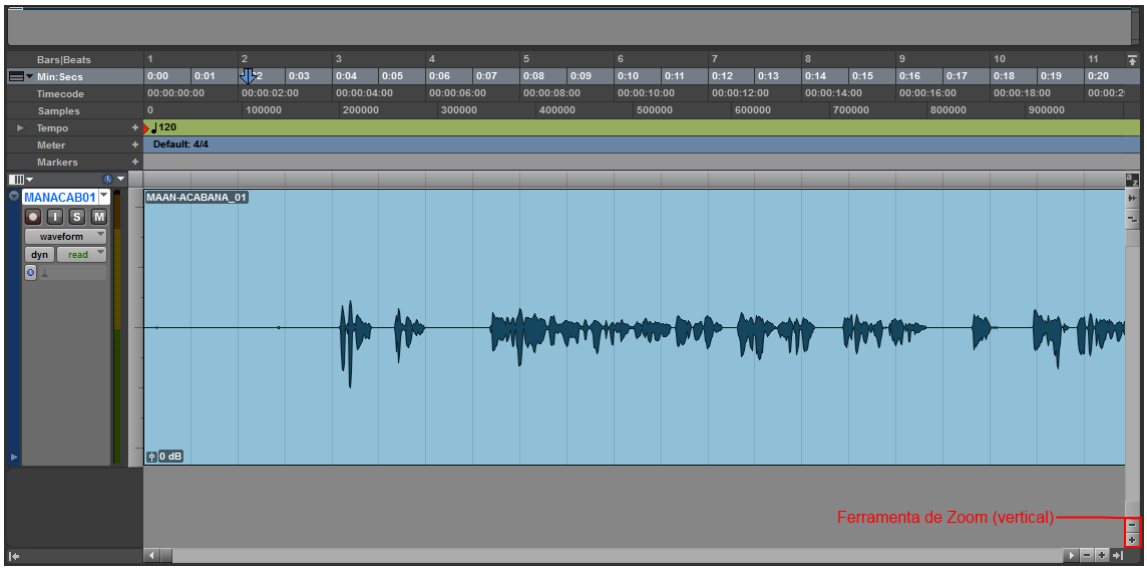


Fig. B.22 Ferramenta de *zoom* vertical

Hide e *Make Inactive:* estas funcionalidades podem ser acessadas através do menu de contexto (Fig. B.23), clicando com o botão direito do *mouse* sobre o nome da trilha. A função *hide* permite tornar a trilha "invisível" na tela de edição. Todavia, mesmo com uso deste modo, quando a função *play* é ativada, o áudio é tocado normalmente.

Para evitar que isso ocorra, aplica-se a ferramenta *make inactive* que faz a trilha ser desativada e o seu áudio não ser ouvido. Quando essa função está em uso, o menu de contexto apresenta em seu lugar a função *make active*, que é usada para revertê-la.

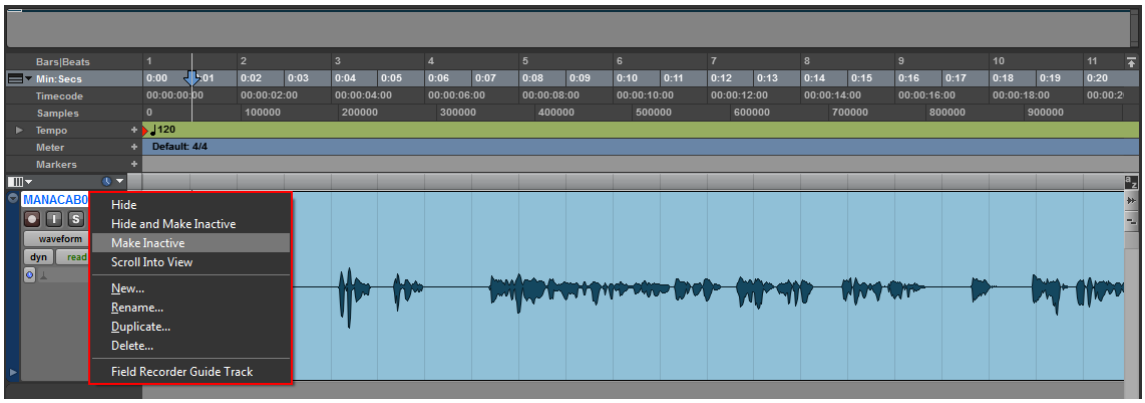


Fig. B.23 Menu de contexto

Apêndice C

Treino do sintetizador utilizando HTS

Neste apêndice, são detalhados os procedimentos relativos ao treino do sistema TTS, desde a preparação de dados até o treinamento propriamente dito.

C.1 Preparação do ambiente

O treino do sistema baseia-se na ferramenta HTS [86], adaptada para o Português Brasileiro. As ferramentas utilizadas na preparação do ambiente para o treino estão dispostas na Tab. C-1.

Tab. C-1 Ferramentas empregadas na preparação do ambiente

Sistema Operacional	*Linux* distribuição *Ubuntu 14.04 LTS* [92]
***Toolkit* de reconhecimento de voz**	*HTK 3.4.1* [93]
***Toolkit* de processamento de sinais de fala**	*SPTK 3.4.1* [94]
Motor de síntese	*HTS Engine 1.05* [95]
Interpretador de Tcl	*Active Tcl 8.4* [96]
Sistema de síntese de voz	*HTS 2.2 [86]*

C.2 Preparação de dados

C.2.1 Criação de dicionário fonético

Esta primeira etapa consiste da criação de uma lista ordenada de palavras do vocabulário que será empregado no treinamento. A fim de criar um modelo acústico apropriado, o HTK necessita de um dicionário balanceado foneticamente, onde haja a ocorrência de um mesmo fonema de 3 a 5 vezes [93].

Assim, num primeiro passo serão criados três arquivos:

- `prompts.txt`: realiza a associação do arquivo de áudio com a transcrição da frase correspondente;
- `wlist.txt`: contém a lista de todas as palavras contidas no dicionário ordenadas alfabeticamente, acrescida das utilizadas para marcação de início (*SENT-START*) e fim (*SENT-END*) de sentenças e pausa (*SENT-PAUSE*);

- `lexicon.lex`: possui a lista de todas as palavras dispostas no `wlist.txt` transcritas foneticamente. Neste trabalho, após a criação deste arquivo, o mesmo foi editado manualmente a fim de corrigir algumas transcrições.

Para isso, é utilizado o *script* `get_lex.pl`, que recebe como entrada um arquivo `wavlist.txt`, que contém a listagem de todos os arquivos de áudio *wave* que serão utilizados no treino, e o arquivo `frases.txt`, que possui o texto de cada frase transcrito por linha, sendo ordenado conforme os áudios. O comando empregado para uso deste é *script* é

```
perl get_lex.pl wavlist.txt frases.txt [DIRETÓRIO DO TTS],
```

onde o campo `[DIRETÓRIO DO TTS]` representa a localização dos executáveis do NLP.

Uma vez gerados os arquivos, deve-se acrescentar manualmente, respeitando o ordenamento alfabético, os marcadores *SENT-START*, *SENT-END* e *SENT-PAUSE* ao `wlist.txt` e ao `lexicon.lex`.

A seguir, será utilizado novo comando para a geração do dicionário propriamente dito que é

```
HDMan -A -D -T 1 -m -w wlist.txt -n monophones1 -i -l dlog dict
                        lexicon.lex.
```

Com este comando, são gerados os arquivos:

- `dlog`: apresenta o *log* da operação;
- `monophones1`: possui a lista de monofones presentes no dicionário;
- `dict`: é o dicionário em si.

O arquivo de *log* permite verificar o balanceamento fonético do dicionário, uma vez que apresenta o número de ocorrência de cada fone. Um exemplo desse tipo de arquivo está apresentado na Fig. C.1.

```
WARNING: no script file /home/dayana/HTS-10984Frases-Nova/data/segmentation/lexicon.lex.ded

Dictionary Usage Statistics
---------------------------
  Dictionary    TotalWords WordsUsed  TotalProns PronsUsed
 lexicon.lex     16470      16470      16470      16470
        dict     16470      16470      16470      16470

16470 words required, 0 missing

New Phone Usage Counts
----------------------
  1. a     : 15170
  2. sp    : 16468
  3. b     :  1613
  4. f     :  1522
  5. d     :  4902
  6. v     :  2453
  7. j     :   615
  8. S     :   587
  9. a~    :  2886
 10. n     :  2774
 11. u     :  7121
 12. X     :  2019
 13. ow    :   592
 14. l     :  2599
 15. o~    :  1266
 16. o     :  4467
 17. r     :  5804
 18. w~    :  1217
 19. s     :  8582
 20. t     :  5295
 21. i     :  9276
 22. ej    :   512
 23. e     :  7969
 24. i~    :  2022
 25. m     :  3311
 26. e~    :  2241
 27. tS    :  1442
 28. dZ    :   943
 29. k     :  4368
 30. E     :   366
 31. L     :   320
 32. js    :  1248
 33. w     :  1140
 34. R     :  2399
 35. j~    :   208
 36. z     :  1704
 37. e~j~  :   278
 38. Z     :   981
 39. g     :  1459
 40. p     :  3238
 41. u~    :   470
 42. J     :   406
 43. O     :   393
 44. j~s   :     5
 45. sil   :     2
 46. pau   :     1

Dictionary /home/dayana/HTS-10984Frases-Nova/data/segmentation/dict created
```

Fig. C.1 Exemplo de arquivo de *log*

C.2.2 Criação de arquivos de transcrição de palavras

O *toolkit* do HTK não processa diretamente o arquivo `prompts.txt`, sendo necessário o particionamento da transcrição em palavras. Para tal, utiliza-se o *script* `prompts2mlf` para gerar um arquivo `words.mlf`. Este último será composto do nome dos arquivos de *label*, os quais serão gerados posteriormente, seguidos de cada palavra separada por linha, conforme o formato exposto na Fig. C.2.

```
#!MLF!#
"*/PTBR_MAAN_00000.lab"
QUEM
NÃO
DUVIDARIA
AO
OUVIR
UM
HOMEM
AFIRMAR
QUE
PASSOU
UM
FIM
DE
SEMANA
INTEIRO
COM
DEUS
.
"*/PTBR_MAAN_00001.lab"
E
SENT-PAUSE
AINDA
MAIS
SENT-PAUSE
EM
UMA
CABANA
.
```

Fig. C.2 Exemplo de parte de arquivo *words.mlf*

A execução do *script* é dada por

```
perl prompts2mlf words.mlf prompts.txt.
```

A seguir, é realizada a expansão do arquivo de transcrição de palavras `words.mlf` para fones (`phones0.mlf`), onde os fonemas estarão separados por linha de arquivo. Vale dizer que o arquivo gerado não conterá a pausa curta (`sp`) entre as palavras transcritas. As pausas curtas contrastam com o modelo de silêncio (`sil`), que possui duração mais longa e acontece ao final das sentenças.

Assim, o primeiro passo nesse sentido é a criação de um arquivo de *script* `mkphones0.led` cujo conteúdo deve ser conforme o disposto na Fig. C.3.

```
EX
IS sil sil
DE sp
```

Fig. C.3 Conteúdo do arquivo *mkphones0.led*, que deve conter uma linha em branco no final

A seguir, deve-se executar o comando `HLEd` do HTS como

```
HLEd –A –D –T 1 -l '*' -d dict -i phones0.mlf mkphones0.led words.mlf,
```

que criará o arquivo `phones0.mlf`.

Neste momento, será criado o arquivo de fones que incluirá as pausas curtas. À semelhança do realizado anteriormente, precisa-se de um arquivo de *script* `mkphones1.led` cujo conteúdo deve estar de acordo com a Fig. C.4.

```
EX
IS sil sil
```

Fig. C.4 Conteúdo do arquivo *mkphones1.led*, que deve conter uma linha em branco no final

Posteriormente, basta gerar o arquivo `phones1.mlf` pelo uso do comando `HLEd` novamente, substituindo `phones0.mlf` por `phones1.mlf` e `mkphones0.led` por `mkphones1.led`.

C.2.3 Codificação de áudio

O HTK trabalha internamente com arquivos de extensão *mfcc*. Para a obtenção dos mesmos é necessária a conversão dos arquivos *wave* para este formato. Dessa forma, inicialmente cria-se um arquivo denominado `codetrain.scp` que faz o mapeamento dos áudios em arquivos no novo formato. Isso é efetuado através do comando

```
perl prompts2codetrain.pl prompts.txt codetrain.scp train.scp
```

que gerará o arquivo `codetrain.scp` e `train.scp`. Este último conterá as informações sobre a localização dos arquivos *mfcc*.

A conversão propriamente dita de arquivos *wave* para *mfcc* é procedida através do comando `HCopy`, que necessita da configuração prévia contida no arquivo `config`. O arquivo utilizado neste trabalho está apresentado na Fig. C.5. Nele, é possível verificar que o reconhecedor fonético empregado faz uso de taxa de *frame* (`TARGETRATE`) de 10 ms e janelamento de Hamming (`WINDOWSIZE`) de 25 ms. Além disso, totaliza o emprego de 39 coeficientes: 12 coeficientes mel-cepstrais (`NUMCEPS`), o coeficiente c_0 (que concentra toda a energia do *frame*), 13 coeficientes delta e 13 coeficientes delta-delta (`MFCC_0_D_A`).

```
SOURCEFORMAT = WAV
TARGETKIND = MFCC_0_D_A
TARGETRATE = 100000.0
SAVECOMPRESSED = T
SAVEWITHCRC = T
WINDOWSIZE = 250000.0
USEHAMMING = T
PREEMCOEF = 0.97
NUMCHANS = 26
CEPLIFTER = 22
NUMCEPS = 12
```

Fig. C.5 Conteúdo do arquivo *config*

O comando então pode ser efetuado como se segue

```
HCopy –A –D –T 1 –C config –S codetrain.scp
```

que culminará na criação de arquivos *mfcc* conforme listados no `codetrain.scp`.

C.2.4 Segmentação automática

Nesta fase, será realizada uma tarefa de relevante importância para a qualidade da síntese: o alinhamento do conteúdo fonético dos áudios. Em outras palavras, é realizada a correspondência entre as unidades da transcrição fonética das elocuções e os trechos de áudio.

Este processo é iniciado através da estimativa de um conjunto de HMM baseado em monofones independentes de contexto. Os modelos possuem 3 estados emitentes sem saltos e com transição da esquerda para a direita, sendo cada estado representado por uma gaussiana. Além disso, a inicialização dos HMM é realizada pela técnica de *flat start*.

Assim, para a primeira estimativa dos HMM de cada fone são necessários o modelo denominado `proto` (Fig. C.6) e o arquivo de configuração `config2` (Fig. C.7).

```
~o <vecSize> 39 <MFCC_0_D_A>
~h "proto"
<BEGINHMM>
 <NUMSTATES> 5
 <STATE> 2
  <MEAN> 39
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
  <VARIANCE> 39
   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 <STATE> 3
  <MEAN> 39
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
  <VARIANCE> 39
   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 <STATE> 4
  <MEAN> 39
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
  <VARIANCE> 39
   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 <TRANSP> 5
  0.0 1.0 0.0 0.0 0.0
  0.0 0.6 0.4 0.0 0.0
  0.0 0.0 0.6 0.4 0.0
  0.0 0.0 0.0 0.7 0.3
  0.0 0.0 0.0 0.0 0.0
<ENDHMM>
```

Fig. C.6 Conteúdo do arquivo *proto*

```
TARGETKIND = MFCC_0_D_A
TARGETRATE = 100000.0
SAVECOMPRESSED = T
SAVEWITHCRC = T
WINDOWSIZE = 250000.0
USEHAMMING = T
PREEMCOEF = 0.97
NUMCHANS = 26
CEPLIFTER = 22
NUMCEPS = 12
```

Fig. C.7 Conteúdo do arquivo *config2*

Neste ponto, deve-se criar uma pasta `hmm0` e realizar a estimativa inicial dos modelos de HMM através do comando

```
HCompV -A -D -T 1 -C config2 -f 0.01 -m -S train.scp -M hmm0 proto,
```

que gerará dentro de `hmm0` dois novos arquivos: `vFloors` e outra versão de `proto`.

Observe que são utilizados o arquivo `train.scp`, criado nos procedimentos da seção anterior, a fim de informar a localização dos arquivos *mfcc*.

A seguir, cria-se, na pasta `hmm0`, um arquivo denominado `hmmdefs` (Fig. C.8) com o conteúdo do `monophones1` sem o fone `sp` (pausa curta) adaptado da seguinte forma:

- Cada fone colocado entre aspas duplas e antecedido de '`~h` ';
- Copiar após cada um desses fones o conteúdo localizado entre as `<BEGINHMM>` e `<ENDHMM>` do arquivo `hmm0/proto`.

```
~h "a"
<BEGINHMM>
<NUMSTATES> 5
<STATE> 2
<MEAN> 39
 -6.448639e+00 -1.399141e+00 5.063490e+00 -3.644217e-02 -4.658403e+00 -1.722829e+00 -9.693022e+00 -1.468678e+00 -2.336308e+00
-5.104821e-01 1.148050e+00 -5.914116e+00 5.736514e+01 4.120281e-03 -1.937155e-04 2.583198e-03 1.730618e-03 3.707967e-03 -9.882323e-04
-2.335669e-03 2.386020e-04 2.051294e-03 6.102481e-04 7.940540e-04 -1.474211e-03 4.717657e-03 -7.459765e-04 1.514689e-03 -7.640736e-04
1.760556e-04 3.836304e-04 4.993323e-04 1.358736e-03 7.926145e-05 -7.410717e-04 -6.230447e-04 -4.121614e-04 4.821779e-04 -4.507719e-03
<VARIANCE> 39
 1.184532e+02 5.272099e+01 7.626372e+01 5.504166e+01 7.164722e+01 8.257607e+01 6.029518e+01 7.031299e+01 4.923815e+01 4.235046e+01
4.525887e+01 6.166288e+01 1.517871e+02 6.223657e+00 3.636537e+00 3.364627e+00 3.841260e+00 4.764065e+00 4.362070e+00 4.113756e+00
4.478717e+00 3.689177e+00 3.266164e+00 3.634533e+00 3.948331e+00 6.011727e+00 9.258169e-01 6.168156e-01 5.310647e-01 6.744997e-01
8.004351e-01 7.681108e-01 7.539262e-01 8.003576e-01 6.804072e-01 6.177183e-01 6.845239e-01 6.945533e-01 9.099500e-01
<GCONST> 1.406855e+02
<STATE> 3
<MEAN> 39
 -6.448639e+00 -1.399141e+00 5.063490e+00 -3.644217e-02 -4.658403e+00 -1.722829e+00 -9.693022e+00 -1.468678e+00 -2.336308e+00
-5.104821e-01 1.148050e+00 -5.914116e+00 5.736514e+01 4.120281e-03 -1.937155e-04 2.583198e-03 1.730618e-03 3.707967e-03 -9.882323e-04
-2.335669e-03 2.386020e-04 2.051294e-03 6.102481e-04 7.940540e-04 -1.474211e-03 4.717657e-03 -7.459765e-04 1.514689e-03 -7.640736e-04
1.760556e-04 3.836304e-04 4.993323e-04 1.358736e-03 7.926145e-05 -7.410717e-04 -6.230447e-04 -4.121614e-04 4.821779e-04 -4.507719e-03
<VARIANCE> 39
 1.184532e+02 5.272099e+01 7.626372e+01 5.504166e+01 7.164722e+01 8.257607e+01 6.029518e+01 7.031299e+01 4.923815e+01 4.235046e+01
4.525887e+01 6.166288e+01 1.517871e+02 6.223657e+00 3.636537e+00 3.364627e+00 3.841260e+00 4.764065e+00 4.362070e+00 4.113756e+00
4.478717e+00 3.689177e+00 3.266164e+00 3.634533e+00 3.948331e+00 6.011727e+00 9.258169e-01 6.168156e-01 5.310647e-01 6.744997e-01
8.004351e-01 7.681108e-01 7.539262e-01 8.003576e-01 6.804072e-01 6.177183e-01 6.845239e-01 6.945533e-01 9.099500e-01
<GCONST> 1.406855e+02
<STATE> 4
<MEAN> 39
 -6.448639e+00 -1.399141e+00 5.063490e+00 -3.644217e-02 -4.658403e+00 -1.722829e+00 -9.693022e+00 -1.468678e+00 -2.336308e+00
-5.104821e-01 1.148050e+00 -5.914116e+00 5.736514e+01 4.120281e-03 -1.937155e-04 2.583198e-03 1.730618e-03 3.707967e-03 -9.882323e-04
-2.335669e-03 2.386020e-04 2.051294e-03 6.102481e-04 7.940540e-04 -1.474211e-03 4.717657e-03 -7.459765e-04 1.514689e-03 -7.640736e-04
1.760556e-04 3.836304e-04 4.993323e-04 1.358736e-03 7.926145e-05 -7.410717e-04 -6.230447e-04 -4.121614e-04 4.821779e-04 -4.507719e-03
<VARIANCE> 39
 1.184532e+02 5.272099e+01 7.626372e+01 5.504166e+01 7.164722e+01 8.257607e+01 6.029518e+01 7.031299e+01 4.923815e+01 4.235046e+01
4.525887e+01 6.166288e+01 1.517871e+02 6.223657e+00 3.636537e+00 3.364627e+00 3.841260e+00 4.764065e+00 4.362070e+00 4.113756e+00
4.478717e+00 3.689177e+00 3.266164e+00 3.634533e+00 3.948331e+00 6.011727e+00 9.258169e-01 6.168156e-01 5.310647e-01 6.744997e-01
8.004351e-01 7.681108e-01 7.539262e-01 8.003576e-01 6.804072e-01 6.177183e-01 6.845239e-01 6.945533e-01 9.099500e-01
<GCONST> 1.406855e+02
<TRANSP> 5
 0.000000e+00 1.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
 0.000000e+00 6.000000e-01 4.000000e-01 0.000000e+00 0.000000e+00
 0.000000e+00 0.000000e+00 6.000000e-01 4.000000e-01 0.000000e+00
 0.000000e+00 0.000000e+00 0.000000e+00 7.000000e-01 3.000000e-01
 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
<ENDHMM>
~h "b"
<BEGINHMM>
<NUMSTATES> 5
<STATE> 2
```

Fig. C.8 Exemplo de parte do conteúdo do arquivo *hmmdefs*

É preciso também gerar um arquivo chamado `macros` (Fig. C.9) com o conteúdo de `vFloors` antecedido das 3 primeiras linhas de `proto` (de `~o` e `<DIAGC>`).

```
~o
<STREAMINFO> 1 39
<VECSIZE> 39<NULLD><MFCC_D_A_0><DIAGC>
~v varFloor1
<Variance> 39
 1.184532e+00 5.272099e-01 7.626371e-01 5.504166e-01 7.164722e-01 8.257606e-01
6.029518e-01 7.031299e-01 4.923815e-01 4.235046e-01 4.525886e-01 6.166288e-01 1.517871e+00
6.223657e-02 3.636536e-02 3.364627e-02 3.841260e-02 4.764065e-02 4.362069e-02 4.113756e-02
4.478717e-02 3.689177e-02 3.266164e-02 3.634533e-02 3.948331e-02 6.011727e-02 9.258169e-03
6.168156e-03 5.310647e-03 6.744997e-03 8.004352e-03 7.681108e-03 7.539262e-03 8.003577e-03
6.804071e-03 6.177183e-03 6.845239e-03 6.945533e-03 9.099500e-03
```

Fig. C.9 Exemplo de arquivo *macros*

Na sequência, são criadas 9 pastas numeradas consecutivamente como `hmm1` a `hmm9`. Os monofones são reestimados através do comando `HERest`, que carrega todos os modelos de `hmm0` e os arquivos *mfcc* neste processo. A execução desta tarefa é feita como

```
HERest -A -D -T 1 -C config2 -I phones0.mlf -t 250.0 150.0 1000.0 -S
    train.scp -H hmm0/macros -H hmm0/hmmdefs -M hmm1 monophones0,
```

que geram os arquivos `hmmdefs` e `macros` na pasta `hmm1`. O arquivo `monophones0` é composto pelo conteúdo de `monophones1` sem o fone `sp`.

Este processo é repetido para os conjuntos de modelos em `hmm2` e `hmm3`. Para hmm2, o comando é

```
HERest -A -D -T 1 -C config2 -I phones0.mlf -t 250.0 150.0 1000.0 -S
    train.scp -H hmm1/macros -H hmm1/hmmdefs -M hmm2 monophones0,
```

enquanto para hmm3 é

```
HERest -A -D -T 1 -C config2 -I phones0.mlf -t 250.0 150.0
1000.0 -S train.scp -H hmm2/macros -H hmm2/hmmdefs -M hmm3 monophones0.
```

No próximo passo é incluído o modelo de `sp`, que está associado ao estado central do de silêncio. Para fazer essa associação entre os modelos, é utilizada a ferramenta `HHED`. Desse modo, preliminarmente copia-se o conteúdo de `hmm3` para `hmm4` e gera-se um novo modelo de `sp` no arquivo `hmmdefs` pela replicação do `sil`, remoção dos estados 2 e 4 do mesmo e realização das seguintes mudanças:

- `<NUMSTATES>` para 3;
- `<STATE>` para 2;
- `<TRANSP>` para 3;
- Matriz `<TRANSP>` para $\begin{bmatrix} 0,0 & 1,0 & 0,0 \\ 0,0 & 0,9 & 0,1 \\ 0,0 & 0,0 & 0,0 \end{bmatrix}$.

Deve-se, então, criar o arquivo de configuração `sil.hed` conforme a Fig. C.10.

```
AT 2 4 0.2 {sil.transP}
AT 4 2 0.2 {sil.transP}
AT 1 3 0.3 {sp.transP}
TI silst {sil.state[3],sp.state[2]}
```

Fig. C.10 Conteúdo do arquivo *sil.hed*

Por fim, executa-se o comando

```
HHEd -A -D -T 1 -H hmm4/macros -H hmm4/hmmdefs -M hmm5 sil.hed
                        monophones1
```

desta vez utilizando `monophones1`, que inclui o modelo de `sp`.

Sendo criados os arquivos `hmmdefs` e `macros` em `hmm5`. Na sequência, os modelos são reestimados através de `HERest`. Para `hmm6`,

```
HERest -A -D -T 1 -C config  -I phones1.mlf -t 250.0 150.0 3000.0 -S
    train.scp -H hmm5/macros -H  hmm5/hmmdefs -M hmm6 monophones1
```

e, para hmm7,

```
HERest -A -D -T 1 -C config  -I phones1.mlf -t 250.0 150.0 3000.0 -S
    train.scp -H hmm6/macros -H  hmm6/hmmdefs -M hmm7 monophones1.
```

É, portanto, utilizada a ferramenta `HVite`, que realiza o realinhamento dos dados de treino gerando o arquivo `aligned.mlf` (Fig. C.11). O comando é descrito como

```
HVite -A -D -T 1 -l '*' -o SWT -b SENT-END -C config -H hmm7/macros -H
  hmm7/hmmdefs -i aligned.mlf -m -t 250.0 150.0 1000.0 -y lab -a -I
      words.mlf -S train.scp dict monophones1> HVite_log.
```

```
#!MLF!#
"*/PTBR_MAAN_00000.lab"
0 200000 sil
200000 600000 k
600000 1500000 e~
1500000 1800000 m
1800000 1800000 sp
1800000 2100000 n
2100000 2400000 a~
2400000 3200000 w~
3200000 3200000 sp
3200000 3500000 d
3500000 4000000 u
4000000 4700000 v
4700000 5100000 i
5100000 5700000 d
5700000 7000000 a
7000000 7400000 r
7400000 9300000 i
9300000 9600000 a
9600000 9600000 sp
9600000 9900000 a
9900000 10300000 w
10300000 10300000 sp
10300000 11800000 ow
11800000 12800000 v
12800000 13800000 i
13800000 14100000 X
14100000 14100000 sp
14100000 14700000 u~
14700000 15400000 m
15400000 15400000 sp
15400000 16900000 o~
16900000 17500000 m
17500000 18400000 e~j~
18400000 18400000 sp
18400000 19000000 a
19000000 20000000 f
20000000 20700000 i
20700000 21500000 R
21500000 22200000 m
22200000 23200000 a
23200000 23800000 X
23800000 23800000 sp
23800000 24600000 k
24600000 24900000 i
24900000 24900000 sp
24900000 25700000 p
25700000 26400000 a
26400000 27700000 s
27700000 28300000 ow
28300000 28300000 sp
28300000 29300000 u~
29300000 29600000 m
29600000 29600000 sp
```

Fig. C.11 Exemplo de parte do conteúdo do arquivo *aligned.mlf*

É, ainda, produzido um *log*, denominado `HVite_log`, que deve ser analisado para verificação de eventuais erros. Em caso de ausência de erros, é possível prosseguir com a reestimativa dos modelos. Para `hmm8`,

```
HERest -A -D -T 1 -C config -I aligned.mlf -t 250.0 150.0 3000.0 -S
    train.scp -H hmm7/macros -H hmm7/hmmdefs -M hmm8 monophones1
```

e, para `hmm9`,

```
HERest -A -D -T 1 -C config -I aligned.mlf -t 250.0 150.0 3000.0 -S
    train.scp -H hmm8/macros -H hmm8/hmmdefs -M hmm9 monophones1.
```

Ao término desta última estimativa, deve-se executar `HVite` novamente na pasta de `hmm9`. O produto final desta execução será o novo arquivo `aligned.mlf`.

Por fim, são empregados dois *scripts* para a geração dos arquivos de *label*. Estes arquivos são criados segundo dois formatos distintos: *mono* e *full*. O primeiro deles possui a correspondência entre o áudio e a transcrição fonética. Já o segundo, possui, além dessa informação, os dados de pronúncia concernentes ao contexto de cada fone. Os comandos dos *scripts* são

```
perl realinhaLabel.pl aligned.mlf [DIRETÓRIO DE DESTINO DOS ARQUIVOS
                                   MONO],
```

para a geração dos *labels mono*, e

```
perl realinhaFullLabel.pl aligned.mlf frases.txt [DIRETÓRIO DE DESTINO
                  DOS ARQUIVOS FULL] [DIRETÓRIO DO TTS],
```

para os *labels full*.

C.3 Treinamento

Para o treinamento, é considerada a estrutura de pastas da C.12, que pode ser obtida através de *download* em [86].

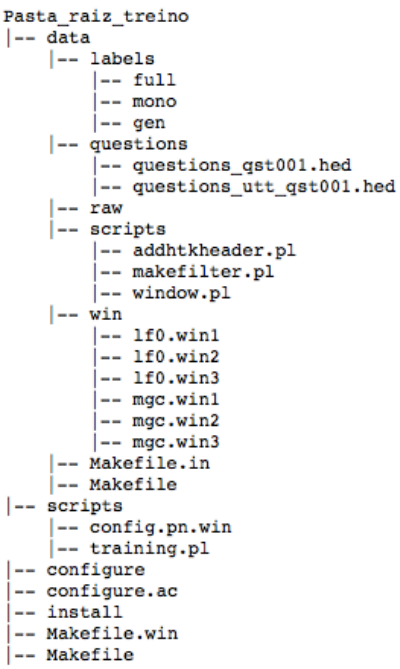


Fig. C.12 Estrutura de arquivos para o treinamento

Os arquivos de *label* gerados na seção anterior são adicionados à pasta da estrutura de Fig. C.12 de mesmo nome. Já os arquivos de áudio no formato *wave* são convertidos para *raw* por meio do *script* `wav2raw.pl`, efetuando o comando

```
perl wav2raw.pl [DIRETÓRIO DE ORIGEM] [DIRETÓRIO DE DESTINO] [NÚMERO
                                    DE WAVES].
```

A seguir, os arquivos criados são incluídos na pasta correspondente dentro da estrutura da Fig. C.12.

Feito isso, basta executar o comando de configuração do treino `configure`:

```
./configure --with-tcl-search-path=[DIRETÓRIO BIN DO ACTIVE TCL] \
                --with-sptk-search-path=[DIRETÓRIO BIN DO SPTK] \
                 --with-hts-search-path=/[DIRETÓRIO BIN DO HTS] \
               --with-hts-engine-search-path=[DIRETÓRIO BIN DO  HTS
                                    ENGINE] \
          MGCORDER=24 GAMMA=0 FREQWARP=0.42 SPEAKER=[IDENTIFICAÇÃO
DO LOCUTOR] DATASET=PTBR SAMPFREQ=16000 FFTLEN=1024 FRAMESHIFT=80 \
                     FRAMELEN=400 LOWERF0=40 UPPERF0=400
```

que originará um arquivo de `makefile`. Para iniciar o treinamento, deve proceder o comando `make`.