TESE DE MESTRADO

RECONHECIMENTO DE LOCUTORES UTILIZANDO MODELOS DE MARKOV ESCONDIDOS CONTÍNUOS

EVANDRO DAVID SILVA PARANAGUÁ

INSTITUTO MILITAR DE ENGENHARIA

RECONHECIMENTO DE LOCUTOR UTILIZANDO MODELOS DE MARKOV ESCONDIDOS CONTÍNUOS

POR

EVANDRO DAVID SILVA PARANAGUÁ

TESE SUBMETIDA

COMO REQUISITO PARCIAL

PARA A OBTENÇÃO DO GRAU DE

MESTRE EM CIÊNCIAS

EM ENGENHARIA ELÉTRICA

Assinatura do Orientador da Tese

TC QEM Sidney Cerqueira Bispo dos Santos - M.C.

Rio de Janeiro - RJ

Maio de 1997

Tese apresentada por:

_____

EVANDRO DAVID SILVA PARANAGUÁ

e aprovada pelos Srs.:

_____

SIDNEY CERQUEIRA BISPO DOS SANTOS - TC. QEM - M.C. - (IME)
Orientador

_____

ANTÔNIO CARLOS GAY THOMÉ - Cel. R/1 - Ph.D. - (UFRJ)

_____

ROBERTO MISCOW FILHO - Cel. R/1 - M.C. (IME)

IME, RIO DE JANEIRO - RJ, 06 DE MAIO DE 1997

"Dedico esta obra aos meus pais, Orivaldo e Elizabeth, que sempre me incentivaram a transpor os obstáculos desta caminhada.
E também, a minha namorada, Aninha, pela imensurável compreensão e carinho me proporcionado neste período."

AGRADECIMENTOS

Ao TC QEM Sidney Cerqueira Bispo dos Santos pela indicação do tema desta tese, pela orientação segura e pelo companheirismo transmitido no decorrer deste trabalho.

Aos professores Cel. R/1 Roberto Miscow Filho e Cel. R/1 Antônio Carlos Gay Thomé pelos ensinamentos transmitidos e pela valiosa colaboração nas correções técnicas do compêndio desta tese.

Aos alunos da graduação do IME, em especial ao Ten. Dirceu Gonzaga da Silva pela colaboração na implementação dos algoritmos.

À todos os locutores e locutoras que participaram do desenvolvimento do sistema. Em especial à Ana Claudia Paulino Cáo, que apesar de suas inúmeras tarefas profissionais, dedicou-se nos finais de semana às gravações.

Aos colegas da área de processamento de sinais Suelaine dos Santos Diniz e Ricardo Honório Guedes de Souza agradeço pelo ambiente de cooperação desinteressada e interesse pelo estudo que souberam criar.

Ao Departamento de Engenharia Elétrica do IME (DE-3), agradeço o permanente apoio e cooperação que me foram oferecidos pela Chefia, pelo corpo docente e funcionários. Em especial, aos técnicos dos laboratórios pela imprescindível ajuda em equipamentos.

À CAPES pelo apoio financeiro proporcionado.

RESUMO

É proposto um estudo dos parâmetros que melhor modelem um sistema de reconhecimento automático do locutor utilizando Modelos de Markov Escondidos Contínuos cujo objetivo é encontrar um modelo representativo de cada locutor que permita realizar as tarefas de verificação e identificação, dependente do texto, a partir de uma elocução teste.

Para a realização deste estudo foi necessário a obtenção de um conjunto de locutores e a seleção de uma base de dados a partir de suas gravações. Utilizou-se dois conjuntos de locutores, um masculino com 5 locutores para treinamentos e testes e 3 somente para testes e um feminino contendo 4 locutores para treinamentos e testes e 3 somente para testes. Para uma melhor comparação entre os modelagem do sistema com relação ao conteúdo acústico, utilizou-se duas frases distintas, selecionadas considerando-se estudos fonéticos / fonológicos. Obteve-se para cada frase uma base de dados através da gravação de 70 elocuções, 50 para treinamento e 20 para teste de cada locutor utilizado para treinamento, e mais 20 gravações, realizadas para cada locutor destinado a testes.

Os vetores das características da voz, ou seja, as observações, foram construídas utilizando-se 12 coeficientes mel-cepestro, 12 coeficientes delta mel-cepestro, log- energia e delta log-energia.

Foram analisadas diversas combinações de misturas de Gaussianas e número de estados. A configuração utilizando uma mistura de 5 grupos e 12 estados foi a que apresentou melhor resultado, para as duas frases e os dois conjuntos de locutores. O conjunto feminino apresentou maiores dificuldades na convergência do modelo e um desempenho pior que o conjunto masculino. O conjunto feminino foi o único que apresentou erros de falsa aceitação para o limiar utilizado.

Após a realização dos testes verificou-se que na maioria dos casos, o total dos erros eram de falsa rejeição do locutor treinado. Concluiu-se que os limiares estavam um pouco abaixo do valor ideal para o sistema, o que pode ser atribuído ao pequeno número de elocuções utilizadas no cálculo. Um pequeno ajuste nos limiares proporcionou uma mudança da taxa de aceitação para aproximadamente 100%.

ABSTRACT

It is proposed a study of the parameters for the Continuous Density Hidden Markov Models, which better model a system for Speaker's Automatic Recognition through his voice.

For training and tests, two set of speakers were used, a male and a female ones and two sentences: "O prazo tá terminando" and "Amanhã ligo de novo".

The configuration with 5 mixtures and 12 states was which presented best result for both sets. Meanwhile, the set of female speakers has got greater difficulties in the model's convergence and its performance was worse than the set of male speakers.

It was verified that, in most part of tests, the total of errors were of false rejection of the experienced speaker. It is concluded that the thresholds were a little below the ideal value for the system, what may be attributed to the short number of elocutions used in the calculus. A simple adjust in threshold supplied a changing of acception rate for approximately 100%.

SUMÁRIO

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SÍMBOLOS

HMM	"Hidden Markov Models" — Modelos de Markov Escondidos.
IAL	Identificação Automática do Locutor
VAL	Verificação Automática de Locutor
RASTA	"Relative Spectral"
MFCC	"Mel Frequency Cepstral Coeficients"
PLP	"Perceptually Weigthed Linear Prediction"
RAL	Reconhecimento Automático de Locutor
ANN	"Artificial Neural Network"
MDD	"Mixture Decomposition Discrimination"
A/D	Analógico - Digital
RSR	Relação Sinal Ruído
LPC	"Linear Predictive Coefficients"
MCLPC	Mel- Cepstro Derivado do LPC
DFT	"Discrete Fourier Transform"
IDFT	"Inverse Discrete Fourier Transform"
LIT	Linear Invariante no Tempo
QV	Quantização Vetorial
ML	"Maximum Likelihood"
CDHMM	"Continuous Densities Hidden Markov Models"
CM	Conjunto Masculino
CF	Conjunto Feminino
USP	Universidade de São Paulo
PC	"Personal Computer"
IME	Instituto Militar de Engenharia
FIR	"Finite-Duration Impulse Response Systems"
AR	"Autoregressive"
LID	Sistema Linear Invariante Discreto
FA	Falsa Aceitação
FR	Falsa Rejeição
TA	Taxa de Aceitação
CDMC	Coeficientes Delta-Mel-Cepestro

CAPÍTULO 1

INTRODUÇÃO

Atualmente, a gigantesca e flexível rede mundial de telecomunicações torna possível o acesso do pesquisador às pesquisas mais recentes. Várias destas pesquisas surgem de idéias originais, porém os artigos ou "tutoriais" publicam apenas a teoria básica. As partes mais importantes só chegam ao domínio público quando já existem outras mais avançadas. Assim, cumpre aos pesquisadores brasileiros o papel de romper o isolamento e iniciar pesquisas em temas recentes, trazendo ao seu domínio e adaptando às suas condições.

Com esta visão, iniciou-se na linha de Pesquisa de Processamento de Sinais de Voz do IME, pesquisas utilizando métodos estatísticos baseados em Modelos de Markov Escondidos (Hidden Markov Models - HMM) no reconhecimento de locutor.

Este modelo foi introduzido e estudado nas décadas de 60 e 70, progressivamente tornando-se popular nos últimos anos. O interesse está no fato de que o HMM é muito eficaz na utilização das propriedades estatísticas dos fenômenos.

Pesquisas recentes, feitas com sinais de voz em várias línguas estrangeiras, têm demonstrado que HMM possui altas taxas de acertos no reconhecimento automático; isso sugere a existência de uma gama de estudos para os pesquisadores que operam com sinais de voz em português.

1.1 - O RECONHECIMENTO DE LOCUTOR POR PESSOAS

As pessoas podem identificar com precisão vozes familiares. Cerca-se de 2 a 3s (segundos) da voz são suficientes para a identificação, apesar da performance diminuir com a

não familiariedade. O reconhecimento de locutor é uma área da Inteligência Artificial onde o desempenho da máquina pode superar o desempenho de seres humanos: usando curtas locuções de testes e um grande número de locutores, a acurácia da Verificação Automática de Locutor (**VAL**) ou da Identificação Automática de Locutor (**IAL**) frequentemente excede a dos seres humanos. Verifica-se isto, especialmente para locutores não familiares, onde o "tempo de treinamento" para os seres humanos aprenderem a nova voz é maior quando comparado ao tempo da máquina[1].

Um estudo com 29 conhecidos entre si obteve 31%, 66% e 83% de reconhecimento com uma palavra, uma sentença e 30s de voz, respectivamente[1]. Em um outro estudo, porém, usando vozes de 45 pessoas famosas, com 2s de elocução, obteve-se somente 27% de reconhecimento no teste em que o ouvinte tinha a liberdade da escolha do suposto locutor da voz do teste, e 70% de reconhecimento em que o ouvinte seleciona 1 dentre 6 locutores determinados[2]. Observa-se que a habilidade do reconhecimento humano decresce muito quando diminui o tempo da elocução teste. Em contra partida, num estudo utilizando somente as relações entre as características mel-cepestrais da voz do locutor calculadas a partir de segmentos de voz de muito curta duração, de aproximadamente 120 ms, obteve-se para o conjunto de 10 locutores, uma taxa de 100% de acerto para a IAL e de 99% para a VAL[3].

Desse modo, com a tecnologia em reconhecimento automático do locutor avançando, espera-se a instalação de serviços utilizando a voz, tais como: transações bancárias, controle de acesso a áreas restritas e atividades forenses.

1.2 - PRINCÍPIOS DE RECONHECIMENTO DE LOCUTOR

1.2.1 - Classificação da Tecnologia de Reconhecimento de Locutor

Reconhecimento de Locutores é um termo genérico que se refere à tarefa de discriminar pessoas baseando-se apenas nas características da voz, a Figura 1.1 apresenta as formas de reconhecimento. Divide-se em **identificação do locutor** e **verificação do locutor**. A identificação do locutor é um processo no qual é determinado a qual dos N locutores treinados é, uma dada elocução teste. A identificação ainda pode ser: com rejeição ou sem rejeição. Com rejeição, admite-se um limiar para cada locutor, classifica-se o locutor verdadeiro se a medida de similaridade da elocução teste for maior que algum limiar, caso contrário considera-se o locutor como falso. A Verificação do Locutor é o processo de aceitar ou rejeitar a identidade pretensa de um locutor teste. Dois tipos de erros podem ocorrer na verificação do locutor: a) *falsa aceitação*, que consiste na aceitação de um locutor impostor ou mímico e b) *falsa rejeição*, rejeição do locutor verdadeiro[4,5].

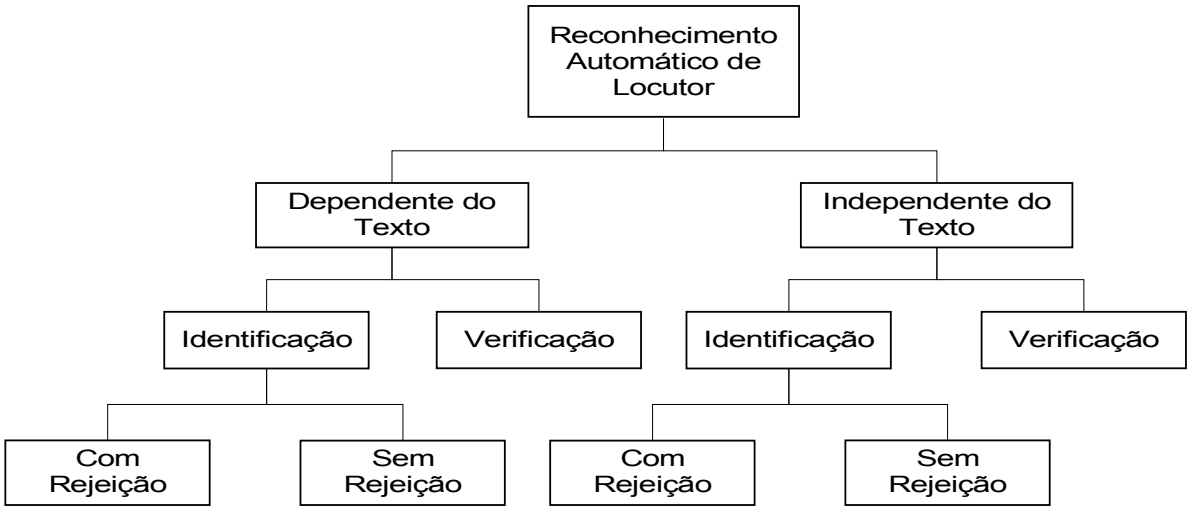


FIGURA 1.1: Tipos de Reconhecimento de Locutor

A diferença fundamental entre identificação e verificação é o número de decisões a serem tomadas. Na identificação, o número de decisões é igual ao tamanho da população (mais 1 quando com rejeição), e, na verificação há duas, aceitação ou rejeição, indiferente do tamanho da população. Portanto, a performance da identificação diminui com o aumento da população, e a performance na verificação se aproxima de uma constante, independente do tamanho da população[4,5].

Os métodos de reconhecimento de locutor podem ser divididos em método dependente do texto e método independente do texto. O primeiro método utiliza o mesmo texto para o treinamento e reconhecimento, ao passo que o último não especifica o texto. A estrutura do sistema de reconhecimento dependente do texto é, portanto, mais simples.

1.2.2 - Estrutura Básica dos Sistemas de Reconhecimento de Locutores

A Figura 1.2 mostra a estrutura básica do sistema de reconhecimento de locutor. Na identificação, a elocução de um locutor desconhecido é analisada e comparada com o modelo dos locutores conhecidos. O locutor desconhecido é identificado como o locutor de cujo modelo melhor corresponder a elocução de entrada.

Na verificação do locutor, a elocução de um locutor supostamente conhecido é comparada com o modelo pretendido; se o resultado for acima de um limiar, a identidade é aceita. Um limiar alto dificulta a falsa aceitação pelo sistema, mas aumenta o risco de ocorrer a falsa rejeição. E, ao contrário, um limiar baixo garante a aceitação de todos os locutores verdadeiros mas aumenta o risco de falsa aceitação[4,5].

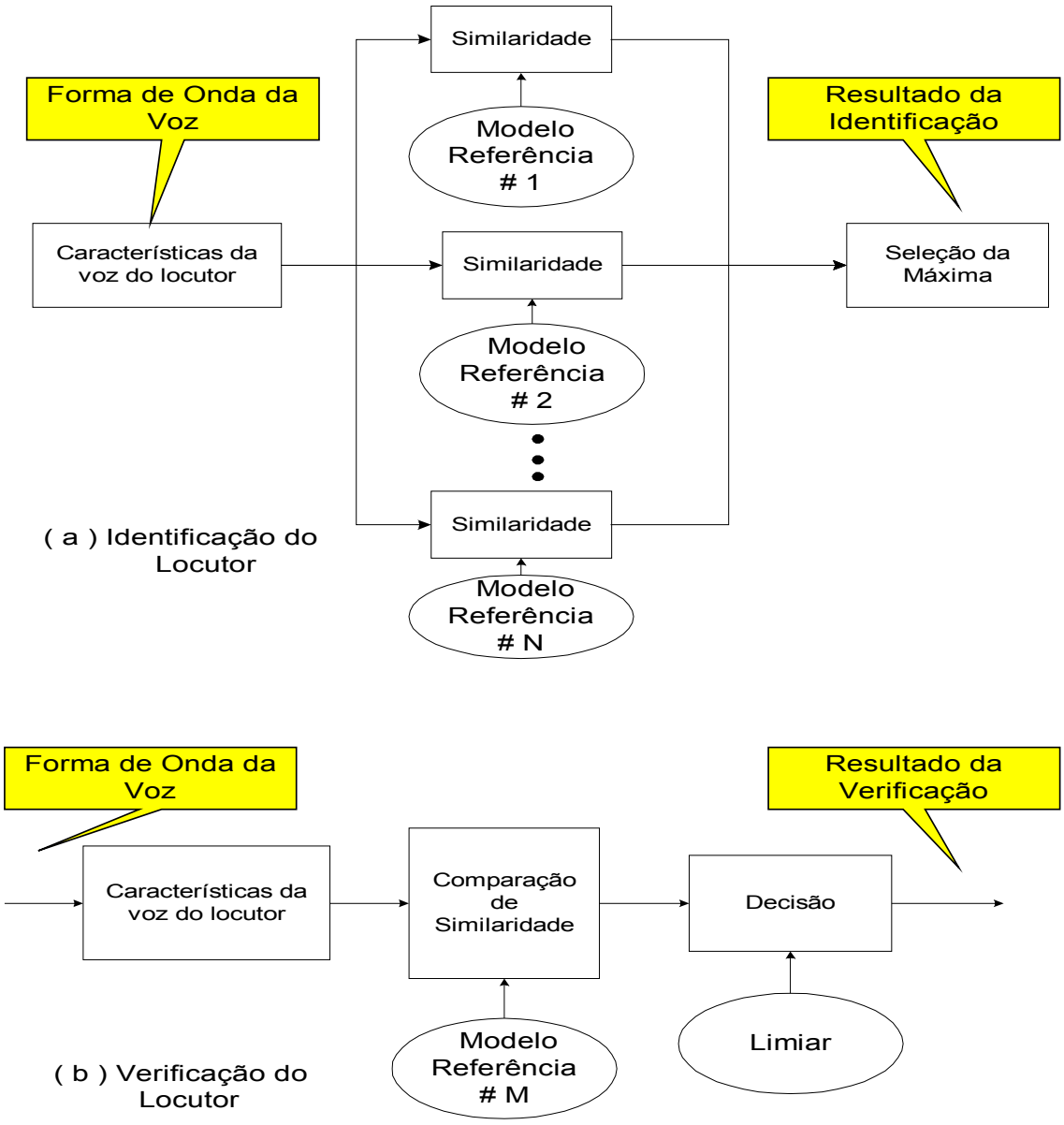


FIGURA 1.2: Estrutura básica do Sistema de Reconhecimento do Locutor

1.3 - ESTADO ATUAL DA ARTE EM RECONHECIMENTO DE LOCUTOR

Atualmente, a vasta área de processamentos de sinais de voz pode ser dividida em outras sub-áreas, de acordo com sua aplicação e tecnologia[6]. Estas áreas são:

1) Síntese da Voz

2) Codificação da Voz

3) Reconhecimento da Voz

4) Reconhecimento do Locutor

5) Tradução da Linguagem Falada

6) Identificação da Linguagem Falada

A sub-área Reconhecimento do Locutor é o processo de reconhecer um locutor através de sua voz. Este processo é dividido em duas partes principais:

1) Extração das características da voz do locutor

2) Reconhecimento de padrões

Na extração das características busca-se obter impressões da voz do locutor que sejam inerentes a ele. Estas características devem ser robustas ao ruído, isto porque existem diferenças entre as condições de treinamento (laboratório) e o ambiente de testes (pode ser ruidoso). Em reconhecimento de padrões almeja-se a separação entre os padrões falsos e os verdadeiros, através de algum modelo matemático[6].

Os, modelos matemáticos utilizados em reconhecimento de voz conseguem obter bons resultados quando os sinais treinados pelo modelo não são ruidosos. Entretanto, como esta situação não é sempre possível[7] pesquisam-se algoritmos de extração de características que reduzam o ruído de fundo somado ao sinal de voz, como por exemplo, RASTA[8] — "relative spectral", MFCC[13] — "mel frequency cepstral coeficients", PLP[9] — "perceptually weigthed linear prediction", características combinadas como MFCC com PLP-RASTA[10] e algorítmos de extração do mel-cepstrum com espaçamento linear e logarítmo do filtro banda passante[11].

Atualmente, as pesquisas em Reconhecimento Automático de Locutor (**RAL**) tem utilizado: HMM (Hidden Markov Models - Modelos de Markov Escondidos)[12,13,14] devido aos ótimos resultados obtidos e a facilidade do seu uso na modelagem do sinal de voz ; ANN (Artificial Neural Network - Redes Neurais Artificiais)[15,16]; modelos híbridos, tais como:

HMM com ANN (Artificial Neural Network - Redes Neurais Artificiais)[17,18] os quais combinam a modelagem temporal do HMM com a capacidade em classificação de padrões da ANN; HMM híbrido com MDD (Mixture Decomposition Discrimination - Discriminação da Decomposição das Misturas)[19] que analisa as diferenças entre as misturas dos estados, entre outros modelos.

O uso do verificação do locutor tem o propósito de restringir o acesso à informação, à redes (computadores, PABX), ou o acesso a locais restritos. Com a ampliação da redes de computadores e outros sistemas de telecomunicações, a necessidade de segurança tem aumentado o ponto de tornar imperativa a verificação do locutor, como por exemplo:

1) Nos serviços de PBX, contra o uso impróprio por pessoas não autorizadas.

2) Nas edes de serviços, onde a verificação do locutor provê acesso a um grande número de serviços de telecomunicações.

3) Nos sistemas de computadores onde é necessário senhas por voz para o acesso.

Portanto, as interfaces por comandos de voz estão se tornando comuns, e a voz é uma escolha indicada para o estabelecimento da identidade pessoal, para prevenção de fraudes, ou para o reconhecimento de fraudadores (na área forense)[6].

1.4 - OBJETIVO DA TESE

O objetivo desta tese foi a obtenção dos parâmetros (número de grupos e número de estados) dos Modelos de Markov Escondidos (HMM) que pudessem melhor representar locutores femininos e masculinos na tarefa de reconhecimento.

1.5 - ORGANIZAÇÃO DO COMPÊNDIO

Este compêndio compõe-se de 7 (sete) capítulos. O Capítulo 1 é uma introdução com ênfase no estado atual da arte. No Capítulo 2 inicia-se um estudo das características ("features") extraídas das elocuções "O prazo tá terminando" e "Amanhã ligo de novo" e, utilizada para o treinamento do modelo. Os Capítulos 3 e 4 tratam da teoria do HMM, sendo que o Capítulo 4 descreve, em fases, o algoritmo implementado na pesquisa. O Capítulo 5, trata do desenvolvimento do "software" implementado e da técnica de obtenção do limiar de decisão empregado. No Capítulo 6 são descritos os resultados e analisado o desempenho do sistema e, por último, no Capítulo 7, são apresentadas as conclusões e sugestões para a continuação da pesquisa realizada.

CAPÍTULO 2

PARÂMETROS DA VOZ

2.1 - INTRODUÇÃO

Em um sistema de Reconhecimento Automático de Locutor (RAL) o pré-processamento da voz, isto é: extração dos parâmetros relevantes do sinal de voz (vetor de características da voz) é o primeiro estágio e consiste, basicamente, em uma compressão de dados com o objetivo de reduzir a dimensão do espaço em que são definidas as elocuções, ou seja: consiste em realizar um mapeamento das elocução de dimensão M, número de amostras do sinal de voz, em um espaço dimensional N, número dos parâmetros (características) extraídos deste sinal (sendo N<<M).

Algumas destas técnicas de análise surgiram recentemente, procurando uma representação paramétrica da "percepção" do sinal de voz: parâmetros que sejam correlatos ao percebido pelo sistema de audição humano.

Deseja-se que estes parâmetros, no Reconhecimento do Locutor ou da Fala, sejam robustos às dificuldades provocadas pelas variabilidades, que podem ser incluídas em quatro categorias[6]:

1- Variabilidade dos sons (mesmo locutor ou diferentes locutores)

2- Variabilidade do canal de gravação (tipo de microfone utilizado).

3- Variabilidade devida à adição do ruído ambiente

4- Variabilidade no modo de falar (hesitação ao falar, ruído de respiração, pigarro)

O fato é que estas variabilidades, geralmente, não podem ser eliminadas. Por isso, as tecnologias de reconhecimento de voz devem executar duas tarefas básicas[6]:

1- Deteção da voz (retirar o ruído antes do início e depois do fim da elocução)

2- Reconhecer a sentença falada baseada em reconhecimento de padrões determinístico (ou estatístico) ou métodos fonético-acústicos.

Para o teste do locutor, o desempenho é sensível a muitos fatores. Entre eles pode-se citar[6]:

1- O microfone usado na gravação.

2- O canal de transmissão.

3- O ruído de fundo.

4- As condições físicas do locutor.

5- As condições do uso do sistema (fone sem fio, telefone celular)

Várias soluções técnicas têm sido proposta para este problema, entretanto, não há uma solução que forneça um sistema robusto com alto desempenho neste campo.

Um estágio de pré-processamento típico de um sistema de RAL pode ser divido em três operações básicas: modelagem espectral, análise espectral e transformação paramétrica. Na Figura 2.1 mostra-se a seqüência destas etapas.

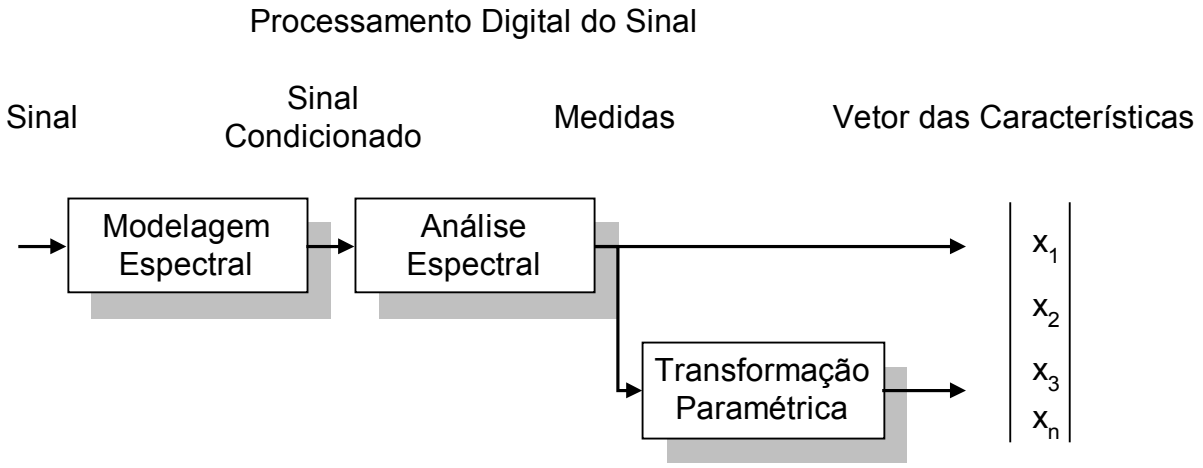


FIGURA 2.1: Um Sistema típico de Pré-Processamento do sinal de voz. A análise espectral fornece o vetor das características ao qual pode-se acrescentar outras por meio da transformações paramétricas (coeficientes delta).

Dado um sinal de voz, o processo da modelagem espectral produz um sinal tratado pela pre-ênfase de modo a realçar as freqüências a que o sistema auditivo é mais sensível e que é convertido em medidas pelo processo da análise espectral gerando um vetor de características. Finalmente, a transformação paramétrica é aplicada sobre essas medidas obtendo-se novos parâmetros (chamados de "parâmetros delta") os quais capturam a dinâmica espectral, ou mudanças do espectro com o tempo; estes parâmetros são acrescentados ao vetor de características[20].

2.2 - MODELAGEM ESPECTRAL

A modelagem espectral envolve duas operações básicas: conversão A/D - conversão do sinal analógico para sinal digitalizado; e filtragem digital, isto é: pré-ênfase — realçando determinadas freqüências do sinal[20,21]. Como mostra a Figura 2.2.

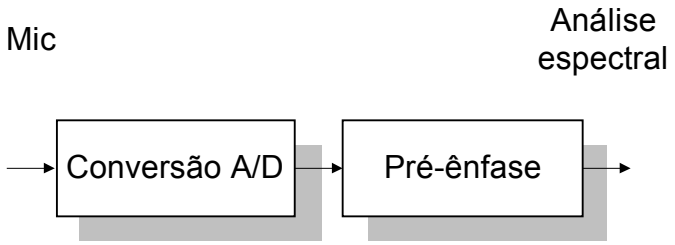


FIGURA 2.2: Seqüência de operações na conversão do sinal analógico para digital.

Um propósito do processo da digitalização do sinal é produzir uma representação dos dados amostrados do sinal de voz com uma relação sinal ruído (RSR) tão alta quanto possível. Em reconhecimento de voz é comum utilizar um valor aproximadamente igual a 30 dB[20].

Após a conversão A/D, utiliza-se o *filtro de pré-ênfase* (filtro digital):

$$H_{pre}(z) = 1 + a_{pre} * z^{-1} \qquad (2.1)$$

com valores típicos para a_{pre} entre -1,0 a -0,4. A pré-ênfase proporciona um ganho no espectro do sinal de aproximadamente 20 dB/dec[20]. Há duas razões para seu uso:

1- as partes vozeadas do sinal de voz possuem uma atenuação espectral natural de aproximadamente 20 dB/dec, devida as características fisiológicas do sistema de produção da voz[22,23]. O filtro de pré-ênfase compensa esta atenuação antes da análise espectral[20].

2- o sistema auditivo é mais sensível às freqüências em torno de 1kHz do espectro. O filtro de pré-ênfase amplifica esta área, fornecendo ao algoritmo de análise espectral uma modelagem das percepções mais importantes do espectro da voz[20].

2.3 - ANÁLISE ESPECTRAL

2.3.1- Janelamento

O primeiro passo para a obtenção das características relevantes do sinal de voz é a divisão do sinal em intervalos curtos de tempo. Isto porque, sendo o sinal de voz um processo estocástico, em geral não estacionário, e sabendo-se que o trato vocal muda de forma muito lentamente na voz contínua, muitas partes da onda acústica podem ser supostas estacionárias, num intervalo de curta duração; este intervalo caracteriza o tamanho da janela a ser usada, cuja duração — de 10 a 40ms (milisegundos) — permite considerar o processo estacionário[15,23,24]. O janelamento do sinal tem o objetivo de amortecer o efeito do "fenômeno Gibbs"[23,24] ("ripple" em amplitude na resposta em freqüência da janela — surge devido a descontinuidade das janelas). Utiliza-se janelas que possuam no domínio da freqüência, um lóbulo principal o mais estreito possível e uma grande diferença de amplitudes entre o lóbulo principal e o primeiro lóbulo lateral.

Algumas janelas comumente usadas são a Kaiser, a Hamming e a Hanning[22,23,24]. Em reconhecimento de locutor normalmente é usada a janela de Hamming, que proporciona maior atenuação fora da banda passante. A equação da janela Hamming é:

$$w[n] = 0.54 - 0.46 \cos(2\pi n/N), \qquad 0 \le n \le N \qquad (2.2)$$

A Figura 2.3 mostra a forma de onda de um segmento (1024 amostras) do fonema / a / e a Figura 2.4 o sinal após a multiplicação por uma janela de Hamming.

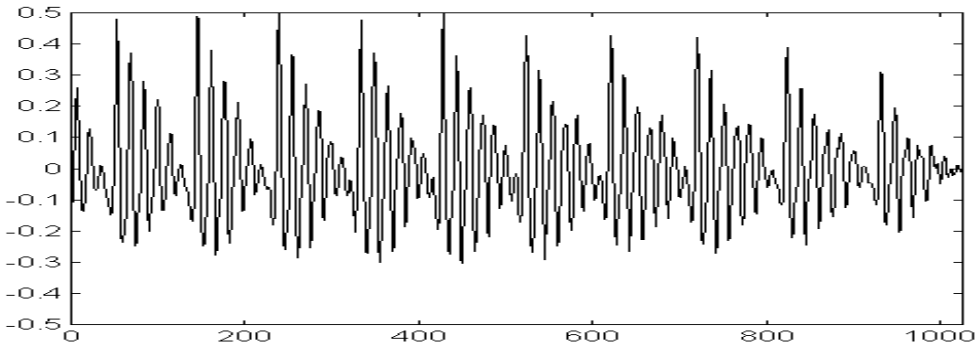


FIGURA 2.3: Forma de onda de um segmento do fonema / a / no domínio do tempo

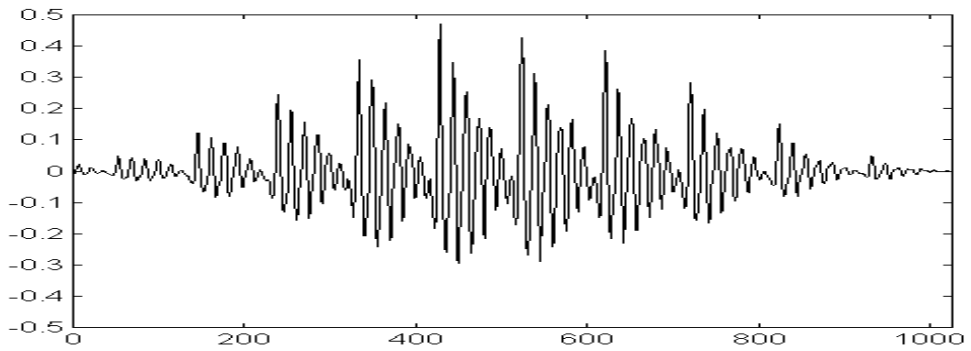


FIGURA 2.4: Forma de onda do fonema / a / janelado por uma janela de Hamming

Realiza-se o janelamento com ou sem superposição parcial entre janelas consecutivas. A superposição aumentará a correlação entre as janelas próximas evitando

variações bruscas entre as características extraídas das janelas adjacentes, entretanto o tempo de processamento será maior.

A Figura 2.5 mostra o procedimento de janelamento, onde T_f é a duração do quadro, definido como a duração, em segundos, na qual o conjunto de parâmetros é válido. T_w é a duração da janela; corresponde ao número de amostras com os quais se obtêm os parâmetros.

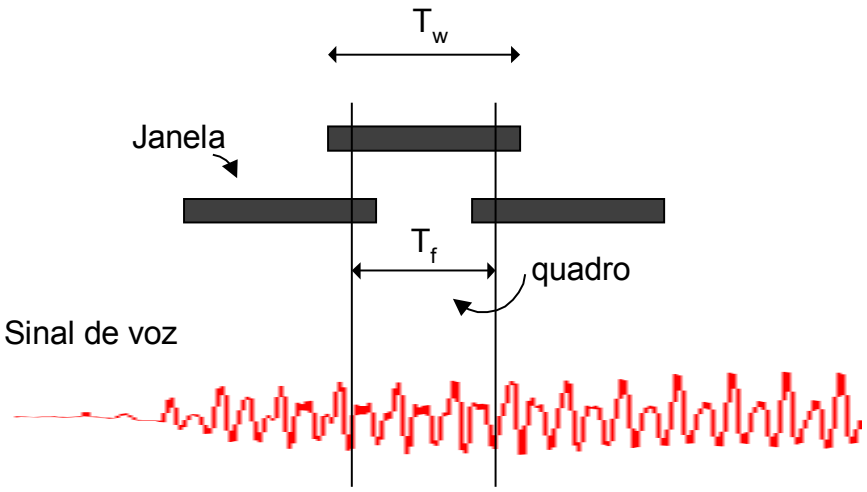


FIGURA 2.5: Procedimento para a obtenção das janelas e quadros.

O valor da superposição é proporcional à diferença T_w-T_f. E a porcentagem da superposição é dada por[22,24]:

$$\% \text{ de Superposição} = (T_w - T_f) * 100 / T_w \qquad (2.3)$$

Um sinal de voz resultante do janelamento, pode ser expresso como:

$$\hat{s}(n,m) = s(n)\, w(n - m * N_f), \qquad 0 \le m \le M - 1 \quad e \quad 0 \le n \le N - 1 \qquad (2.4)$$

onde M representa o número de quadros e N é o número de amostras. N_f é a duração do quadro T_f expresso em amostras. Como exemplo, de valores dos parâmetros M, N e Nf, obtidos na extração das características de uma repetição da frase "O prazo tá terminando"

pronunciada por uma locutora feminina, obteve-se M igual a 124 quadros, N igual a 13782 amostras e N_f igual a 220 amostras por quadro.

2.3.2- Características mais Relevantes do Sinal de Voz

Após o sinal ter sido janelado, o vetor das características ("features") é calculado para cada janela. Um grande número de características podem ser extraídas, para o uso no RAL. Algumas destas, tais como: taxa de cruzamento de zero[25], freqüência fundamental da voz[15,25] ainda são usadas, mas torna-se comum o uso dos parâmetros espectrais[12,20,26,27]. A razão desta popularidade não é unicamente o poder discriminatório da específica informação, mas também a existência de algoritmos robustos para a estimação destas características. Basicamente, as características espectrais podem ser divididas em sete classes, como mostra a Figura 2.6[20].

Pesquisas recentes tem incluído no vetor das características informações dinâmicas, tais como as derivadas de primeira e segunda ordem (derivadas no domínio do tempo — coeficientes de regressão) e parâmetros que simulam o comportamento do sistema de percepção e audição humanas (baseados na escala mel). Sendo:

(a) ***Derivada no domínio do tempo***

As representações mais populares incluem as derivadas das características cepestrais de primeira ordem (delta cepestro) e segunda ordem (delta delta cepestro) e do log energia de primeira e segunda ordem (delta log-energia e delta delta log-energia, respectivamente)[8,9,12,13,20,26-33,59].

$$\frac{\log 1000}{\log x} = \frac{\log 4000}{3{,}06}$$

(b) ***Escala mel***

A escala mel baseia-se no sistema de audição humano, cuja sensibilidade aos sinais de voz se processa em uma escala não linear de freqüências. Este processo baseou-se em estudos no campo da psicoacústica [20,34-37], o qual estuda a percepção auditiva humana.

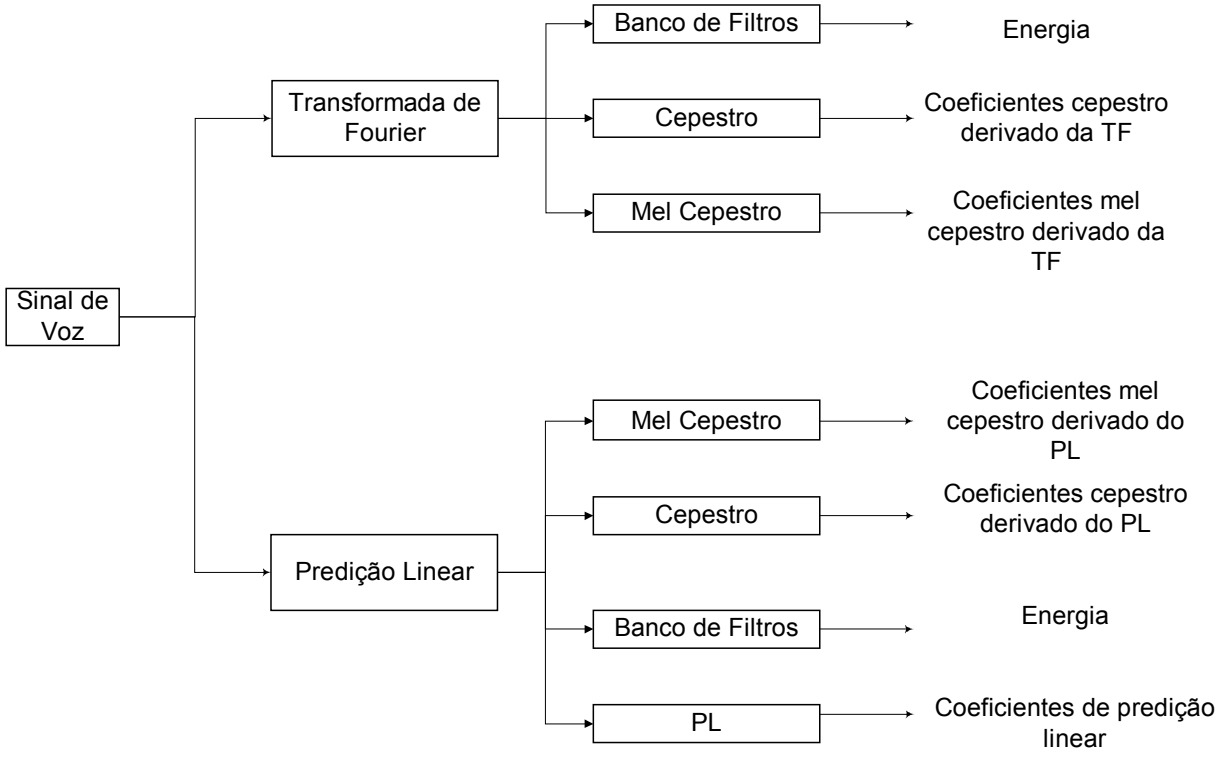


FIGURA 2.6: Tipos de características do sinal de voz que podem ser obtidas. As mais utilizadas são os parâmetros mel-cepestro obtidos ou da predição linear ou da transformada de Fourier. O método da transformada de Fourier é considerado como robusto no caso de ambiente muito ruidoso.

A Equação 2.5 mostra uma aproximação utilizada para mapear a escala da freqüência acústica, f, para a escala da freqüência de "percepção", f_{mel}, conhecida como *escala mel* (f_{mel})[36]. A Figura 2.7 mostra a relação da escala mel com a escala de percepção.

$$f_{mel} = 2595 \log_{10} (1 + f / 700) \qquad (2.5)$$

O mel é a unidade de medida do tom, isto é, de uma freqüência única percebida pelo ouvinte. Stevens e Volkman (1940), em seu experimento, determinaram um mapeamento entre a escala da freqüência real (Hz) e a escala da freqüência percebida (mels)[13].

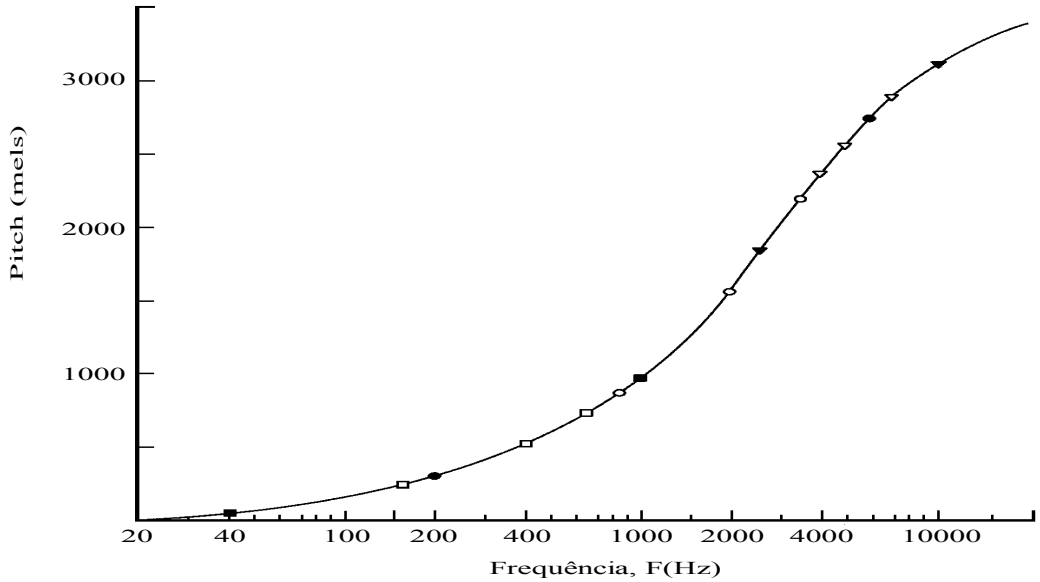


FIGURA 2.7: Mapeamento entre a escala real e a escala mel

O mapeamento é aproximadamente linear abaixo de 1kHz e logarítmico acima (Koenig, 1949). Isso ocorre porque a percepção de uma *determinada freqüência* f_o, pelo sistema auditivo, é influenciada pela energia dentro de uma banda crítica em torno dessa *freqüência* f_o [13]. Além disso, a largura da banda crítica varia com a freqüência, começando em torno dos 100Hz para freqüências abaixo de 1kHz, e aumentando logaritmicamente acima de 1kHz. Assim, as freqüências centrais abaixo de 1kHz são espaçadas em 100Hz, e as freqüências acima de 1kHz são obtidas de acordo com o espaçamento calculado pela Equação 2.6[13] e mostrada na Figura 2.8.

FIGURA 2.8: Relação entre a escala linear de 1000 Hz a 4000 Hz e a escala logarítmica de log(1000) a log(4000).

$$\log_{10} x = 3 + L^* 0.6/10 \tag{2.6}$$

onde L é o passo que varia de acordo com o números de filtros na escala logaritma, neste caso usou-se 10 filtros. Obtém-se assim, um banco de filtros triangulares passa faixa — filtros de banda crítica — com a freqüência central de cada filtro igual a

$$F_{c,i} = k_i \cdot \frac{f_s}{N} \tag{2.7}$$

onde f_s é a freqüência de amostragem e N é o número de pontos usado no cálculo da transformada discreta de Fourier (DFT — Discrete Fourier Transform) e k_i é o ponto da DFT correspondente à freqüência central de cada filtro. A Figura 2.9 mostra a escala logarítmica das freqüências centrais.O gráfico dos filtros é obtido utilizando as Equações 2.7 a 2.11. Para o primeiro filtro, X são os pontos na DFT que variam de 1 a 19:

a) reta à esquerda da freqüência central k_i

$$F(X) = \frac{X}{k_i(1)} \qquad 1 \le X \le 10 \tag{2.8}$$

b) reta à direita da freqüência central k_i

$$F(X) = \frac{X - k_i(2)}{k_i(1) - k_i(2)} \qquad 11 \le X \le 19 \tag{2.9}$$

Para os filtros seguintes:

a) reta à esquerda da freqüência central k_i

$$F(X) = \frac{X - k_i(i)}{k_i(i) - k_i(i-1)} + 1 \tag{2.10}$$

b) reta à direita da freqüência central k_i

$$F(X) = \frac{X - k_i(i)}{k_i(i) - k_i(i+1)} + 1 \tag{2.11}$$

onde:

X são pontos da DFT pertecentes ao filtro k_i.

$k_i(i)$ número da DFT referente a freqüência central k_i, $2 \leq i \leq 20$.

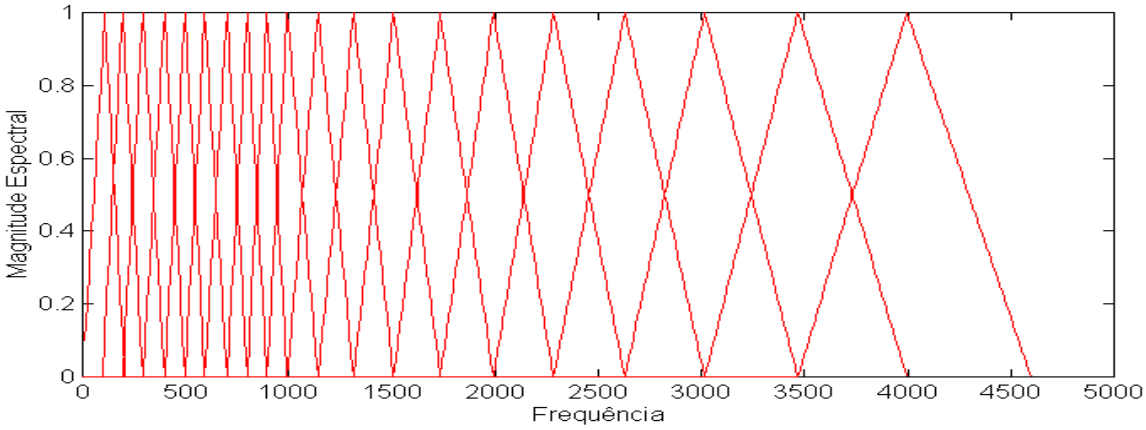


FIGURA 2.9: Espaçamento entre as freqüências centrais dos filtros triangulares.

Como o tempo computacional exigido para o processamento da extração das características e treinamento do modelo HMM é diretamente proporcional a dimensão do vetor extraído das janelas, a seleção das características mais relevantes (discriminantes) do sinal de voz é fundamental. Esta pesquisa utilizou parâmetros da voz que possuem melhores discriminação intra-locutores[8,9,12,13,20,26-33,38].

Atualmente os parâmetros mais utilizados são os *coeficientes mel-cepestro derivados do LPC* (acrônimo de Linear Predictive Coefficients - Coeficientes da Predição Linear) e seu coeficiente de regressão *delta mel-cepestro,* os *parâmetros log-energia* e seu coeficiente de regressão *delta log-energia*.

Estas medidas serão descritas nos itens a seguir.

2.3.3- Coeficientes Mel-Cepestro Derivados do LPC (MCLPC)

Para a definição deste coeficiente torna-se necessário o conhecimento prévio dos conceitos da análise cepestral e LPC.

(a) Análise Cepestral

A voz pode ser representada como a saída de um *sistema linear variante no tempo* e que possui propriedades que variam lentamente com o tempo. Considera-se o princípio básico de análise da voz o qual diz que curtos segmentos do sinal da voz podem efetivamente ser modelados[23,24] como tendo sido gerados por um *sistema linear invariante no tempo* (LIT) excitado por um trem de impulsos quase-periódicos ou por um sinal de ruído aleatório (ver Figura 2.3 que mostra a invariância do sinal em tempo curto).

A excitação e[n] e a resposta ao impulso θ[n] de um sistema LIT são combinadas por uma convolução, como mostra:

$$s[n] = e[n] \otimes \theta\ [n] \tag{2.12}$$

onde e[n] é a excitação do sinal, θ[n] é a resposta ao impulso do trato vocal, ⊗ a operação convolução (operação linear) e s[n] o sinal de voz obtido no domínio do tempo. Faz-se a deconvolução[23,24] para remover a alta freqüência (representada por e[n]) e obter a envoltória do sinal θ[n], isto é, a informação do trato vocal. Portanto, aplica-se o conceito de sistemas homomórficos para a deconvolução dos sinais; onde a operação de entrada é uma convolução e a operação de saída é uma adição (Figura 2.10).

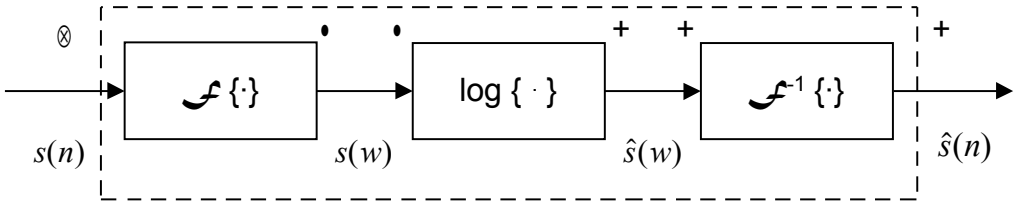


FIGURA 2.10: Representação das características do sistema para a deconvolução

Utilizando-se um operador linear, a Transformada de Fourier Discreta ("DFT") ℱ {•}, transforma a Equação 2.12 para o domínio da freqüência, cuja inversa é $\mathcal{F}^{-1}$ {•}, obtendo:

$$\mathcal{F}\{s[n]\} = s(w) = \mathcal{F}\{e[n] \otimes \theta\ [n]\} \tag{2.13}$$

$$= \mathcal{F}\{e[n]\} \bullet \mathcal{F}\{\theta\ [n]\} \tag{2.14}$$

$$= E(w) \bullet \theta\ (w) \tag{2.15}$$

Sabendo-se que o logaritmo de um produto é definido como a soma dos logaritmos dos termos individuais, tal que:

$$\hat{s}(w) = \log[s(w)] = \log[E(w) \bullet \theta\ (w)] \tag{2.16}$$

$$= \log[E(w)] + \log[\theta\ (w)] \tag{2.17}$$

Obtém-se do sistema, saídas lineares, ou seja, as componentes representativas do sinal tornam-se linearmente combinadas. Aplicando-se a DFT inversa (IDFT) obtém-se o cepestro. (O termo "cepstrum" — cepestro — foi introduzido por Borget entre outros[39] e é aceita como a terminologia para a transformada inversa de Fourier do logaritmo do espectro de potência do sinal[23]). As equações para a determinação dos coeficientes cepestros são:

$$C_s(n) = \mathcal{F}^{-1}\{\log|\mathcal{F}\{s[n]\}|\} \tag{2.18}$$

$$= \mathcal{F}^{-1}\{\log[E(w)] + \log[\theta\ (w)]\} \tag{2.19}$$

$$= \mathcal{F}^{-1}\{\log[E(w)]\} + \mathcal{F}^{-1}\{\log[\theta\ (w)]\} \tag{2.20}$$

$$= \mathcal{F}^{-1}\{\log|\mathcal{F}\{e[n]\}|\} + \mathcal{F}^{-1}\{\log|\mathcal{F}\{\theta\ [n]\}|\} \tag{2.21}$$

$$C_s(n) = C_e(n) + C_\theta(n) \tag{2.22}$$

ou seja, o cepestro representa uma operação linear no domínio "quefrency"[23,40] (anagrama da palavra "frequency") dos coeficientes $C_e(n)$ da excitação, responsável pelas variações rápidas espectrais, com os coeficientes $C_\theta(n)$ do trato vocal, responsável pelas lentas variações

espectrais. Com o objetivo de analisar apenas os componentes $C_\theta(n)$, ou seja, a resposta ao impulso do trato vocal, aplica-se o processo de "liftering" ("filtering" no domínio da freqüência) para remover a componente $C_e(n)$ de $C_s(n)$ e, neste caso, utiliza-se um "low-time lifter" (análogo ao filtro passa baixas no domínio da freqüência)[23,40].

Este procedimento de análise cepestral é mostrado em diagrama na Figura 2.11. Na Figura 2.12 mostra-se os coeficientes $C_\theta(n)$ e $C_e(n)$ do fonema / a / (ver Figura 2.3).

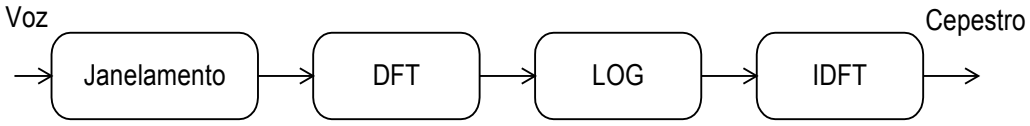


FIGURA 2.11: Análise Cepestral

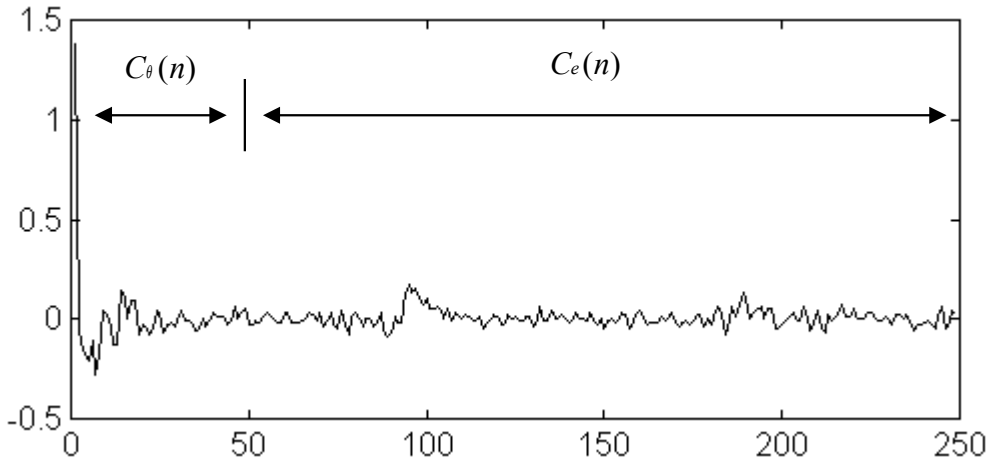


FIGURA 2.12: Coeficientes cepestro $C_s(n)$, no domínio "quefrency"

(b) Coeficientes de Predição Linear - ("LPC")

Este método tem se tornado uma ferramenta para a estimação dos parâmetros básicos da voz, tal como a "pitch", formantes, cepestro, PLP e funções relativa a área do trato vocal.

A idéia fundamental da análise da predicão linear é que a amostra atual da voz pode ser aproximada por uma combinação linear das amostras passadas. O cálculo dos parâmetros do modelo é baseado na teoria do erro médio quadrática mínimo[20].

Dado um sinal s(n), define-se o modelo de predição linear como[20]:

$$s(n) = \sum_{i=1}^{N_{LP}} a_{LP}(i)\, s(n-i) + e(n) \tag{2.23}$$

onde N_{LP} representa o número de coeficientes do modelo (ordem da predição), os $\{a_{LP}\}$ os coeficientes de predição linear (coeficientes do preditor) e e(n) o erro do modelo (a diferença entre o valor predito e o valor medido).

As equações da predição linear são vistas como um filtro quando obtida a transformada Z do erro:

$$e(n) = s(n) - \sum_{i=1}^{N_{LP}} a_{LP}(i)\, s(n-i) \tag{2.24}$$

$$E(z) = S(z) \bullet A(z) \tag{2.25}$$

onde

$$A(z) = 1 - \sum_{i=1}^{N_{LP}} a_{LP}(i)\, z^{-i} \tag{2.26}$$

e 1/A(z) é o modelo *só de pólos*.

Uma virtude deste modelo é sua capacidade de predizer o valor atual do sinal baseado sobre o conjunto de medidas passados[20]. O termo *e(n)* da Equação 2.24 permite uma medida da eficácia do modelo.

Existem três métodos básicos para o cálculo dos coeficientes do preditor: método da covariância que baseia-se sobre a matriz covariância, método da autocorrelação e método "lattice" (ou harmônico)[4,20,22,23]. Em reconhecimento de locutor, o método da autocorrelação é mais utilizado[15,22], com as seguintes vantagens: 1) disponibilidade de um

algoritmo eficiente para a sua implementação: o *algoritmo de Durbin* [41]; 2) os coeficiente gerados por este método resultam em filtros garantidamente estáveis; 3) possibilidade da verificação da estabilidade do sistema pela análise dos coeficientes intermediários[15,20,41].

(c) Coeficientes Mel-Cepestro Derivados do LPC

De acordo com a definição de cepestro, os coeficientes podem ser obtidos *simplesmente* com o uso da análise de Fourier. Isto porque é admitido que o sistema de produção da voz é invariante no tempo — para curtos segmentos de voz. Como desvantagem, o uso da técnica de Fourier aumenta o tempo de processamento em duas vezes em relação ao tempo utilizado pela análise da predição linear [20].

O cepestro derivado da DFT produz, em reconhecimento de locutor, o mesmo desempenho quando comparado ao cepestro derivado da LPC embora em ambientes ruidosos o cepestro calculado pela DFT obtenha um melhor desempenho[20]. Consequentemente a maioria dos pesquisadores utilizam o coeficiente cepestro derivado do LPC[17,29,31,32]

Na Equação (2.26) o logaritmo do inverso do filtro pode ser expresso como uma *série de potências* em z^{-1}, da seguinte forma[20]:

$$\log|A(z)| = C(z) = \sum_{n=1}^{\infty} c_{LP}(i)\, z^{-n} \qquad (2.27)$$

onde

$z = \exp(jwt)$,

w é a freqüência em radianos,

$t = nT$,

T o intervalo de amostragem,

c_{LP} é a amplitude no n-ésimo instante amostrado,

Diferenciando-se ambos os lados da expressão com respeito a z^{-1}, e igualando-se os coeficientes resultantes do polinômio, obtém-se a seguinte equação:

$$c_{LP}(n) = \frac{1}{N_s} \sum_{k=0}^{N_s - 1} \log_{10} |H(k)| \, e^{(2\pi / N_s)kn} \qquad 0 \le n \le N_s - 1 \tag{2.28}$$

onde N_s é o número de amostras da janela selecionada e os coeficientes $\{c_{LP}\}$ são definidos como os coeficientes cepestro derivados do LPC.

Como os coeficientes são, de fato, a transformada inversa de Fourier das respostas ao impulso do modelo LP, o modelo LP do sinal é um *filtro de resposta ao impulso infinito*, assim, pode-se em teoria, calcular infinitos coeficientes cepestro. Entretanto, nas pesquisas se utiliza calcular 12 coeficientes cepestro[17,29,31,32].

Os parâmetros mel-cepestro são obtidos calculando-se a energia E_k dos k filtros aplicado no espectro, com freqüência central espaçados linearmente até 1kHz e logaritmicamente acima, ou seja a escala mel (Figura 2.9), e aplicando-se na Equação 2.28. Desde que a seqüência de energias é par, pode-se substituir a exponêncial por um coseno, e desenvolvendo a equação encontra-se para o cálculo dos coeficientes mel-cepestro:

$$MCLPC_i = \sum_{k=1}^{20} E_k \cos[i(k - 0.5)\pi / 20], \qquad i = 1,2,\ldots,M \tag{2.29}$$

onde *i* é o número de coeficientes cepestro e E_k, k = 1,2,...,20, representa as saídas da log energia do k-ésimo filtro.

2.3.4 - Coeficientes Delta Mel-Cepestro Derivados do LPC

Após a obtenção dos parâmetros MCLPC extrai-se novas características chamadas de *parâmetros delta* (de acordo com a Figura 2.1).

Estes parâmetros são as derivadas de primeira e segunda ordem do MCLPC. São usados para representar as mudanças dinâmicas no espectro da voz. Uma alternativa para a representação das derivadas no domínio do tempo são os coeficientes de regressão. Este coeficientes representam uma aproximação das derivadas de primeira e segunda ordem do vetor das caracteríscas, nesta seqüência.

As aproximações mais populares são [20,24]:

$$\dot{s}(n) \equiv \frac{d}{dt}s(n) \approx s(n) - s(n-1) \tag{2.30}$$

$$\dot{s}(n) \equiv \frac{d}{dt}s(n) \approx s(n+1) - s(n) \tag{2.31}$$

$$\dot{s}(n) \equiv \frac{d}{dt}s(n) \approx \sum_{m=-N_d}^{N_d} ms(n+m) \tag{2.32}$$

Os parâmetros de segunda ordem (parâmetros delta delta) são obtidos reaplicando-se as equações sobre os resultados.

Estes parâmetros detectam variações bruscas dentro do espectro da voz e aumentam a robustez do sistema de reconhecimento[20].

2.3.5 - Log Energia

A energia de tempo curto é definida como:

$$E(n) = \sum_{m=0}^{N-1} (x(m) \bullet w(n-m))^2 \tag{2.33}$$

onde x(n) é a seqüência obtida a partir do sinal de voz amostrado e w(n) é uma das janelas apresentadas no Item 2.3.1. Comumente se usa a janela de Hamming.

Tradicionalmente utiliza-se o logaritmo sobre os parâmetros da energia no reconhecimento de voz com o objetivo de para obter uma compressão entre a baixa energia e a alta energia[12,13], conforme mostra a Figura 2.12. Sadaoki Furui em seu artigo[27] afirma que

aplicando-se a transformação logarítmica na energia obtém-se uma melhor aproximação com a escala de percepção humana.

2.3.6 - Coeficientes Delta Log Energia

Estes parâmetros são calculados utilizando as Equações 2.30 a 2.32 aplicada aos coeficientes da energia.

De acordo com alguns pesquisadores[12,13], a utilização destas características em conjunto com os coeficientes log energia, coeficientes mel-cepestral e delta mel-cepestral aumenta a taxa de reconhecimento.

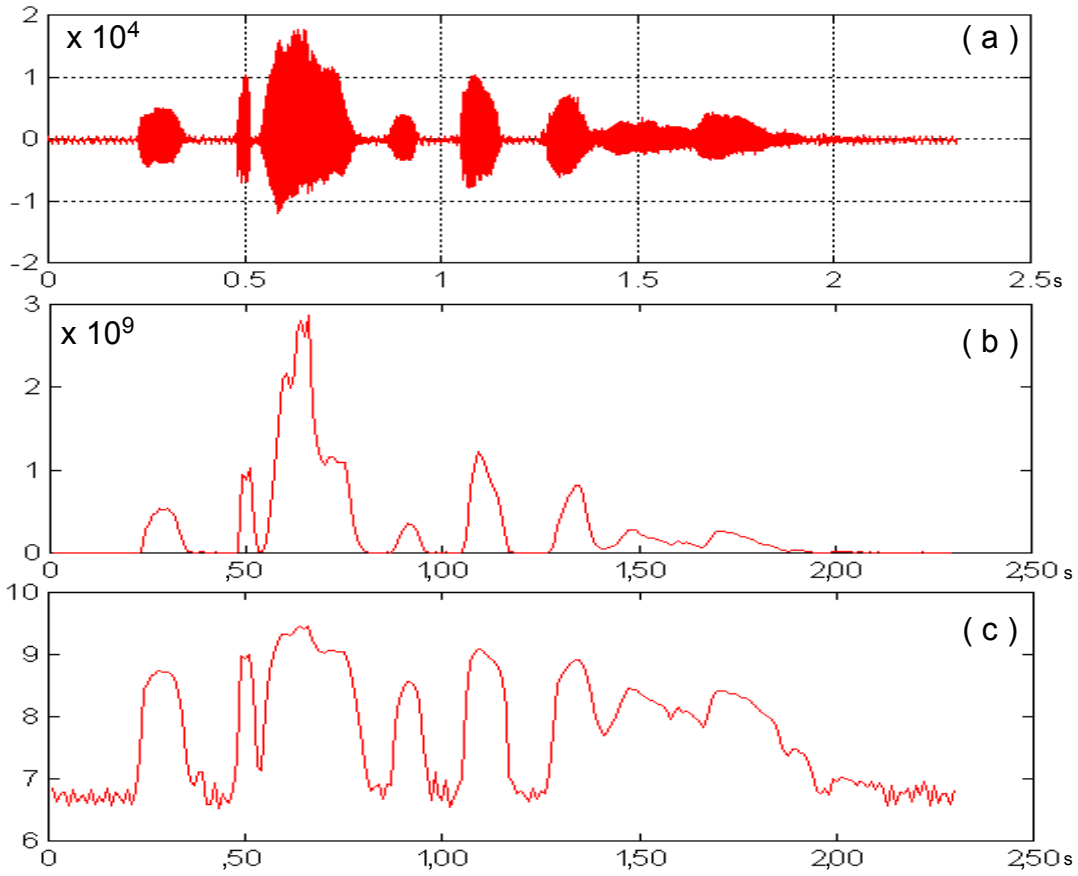


FIGURA 2.13: Gráficos mostrando a necessidade de utilizar o operador logaritmo. Em (a) observa-se a forma de onda da frase "O prazo tá terminando" no domínio do tempo; (b) a evolução da energia de tempo curto com janelas de Hamming; (c) a evolução da energia de tempo curto com janelas de Hamming, mas com o uso do logaritmo aplicado em sua amplitude.

CAPÍTULO 3

TEORIA DOS MODELOS DE MARKOV ESCONDIDOS

3.1 - INTRODUÇÃO

Na década de 70, Baker (1975) e Jelinek (1976) aplicaram independentemente os Modelos de Markov Escondidos (HMM - Hidden Markov Models), no reconhecimento da voz, utilizando a forma moderna de desenvolvimento matemático, proposto por Baum entre outros (Baum & Eagon, 1967; Baum & Sell,1968; Baum, 1972; Baum, Petrie, et alii., 1970), baseados nos conceitos de cadeia de Markov.

O Modelo de Markov Escondido é, atualmente, o mais popular e a melhor aproximação estocástica para o reconhecimento da voz. Esta popularidade deve-se à existência de um algoritmo eficiente e robusto para o Treinamento e Reconhecimento.

No treinamento do modelo, o conjunto dos parâmetros acústicos do sinal de voz do locutor (parâmetros estes extraídos em janelas de intervalo de tempo curto), também chamado de seqüência das observações, é modelada por uma seqüência de estados (cadeia de Markov de primeira ordem) de acordo com a variação temporal da voz.

No reconhecimento, a seqüência das observações da elocução em teste é aceita como verdadeira se possuir alguma medida de similaridade (verossimilhança) acima de um limiar estipulado.

3.2 - DEFINIÇÃO DOS MODELOS DE MARKOV ESCONDIDOS

O Modelo de Markov Escondido constitui-se de um conjunto de N estados cujas transições são governadas pela distribuição das probabilidades de transições. Associada a cada

estado há uma função de densidade de probabilidade discreta (HMM Discreto) ou contínua (HMM Contínuo), que fornece a densidade de probabilidade da observação pertencer àquele estado.

Portanto, um Modelo de Markov Escondido é definido como um par de processos estocásticos (X,Y). O processo X é uma cadeia de Markov de primeira ordem, não sendo diretamente observável. Enquanto, que o processo Y é uma seqüência de variáveis aleatórias no espaço dos parâmetros acústicos (observações)[42].

Para se definir um HMM completamente são necessários os seguintes parâmetros: N, M, A, B e Π_i assim definidos[12]:

1) N, representa número de estados do modelo. Os estados são representados como $S=\{S_1, S_2, ...,S_N\}$ e $q_t(i)$ indica estar no estado S_i, no tempo t.

2) M, representa número de símbolos no alfabeto, quando o espaço é definido por uma função de densidade probabilidade (fdp) discreta ou número de grupos quando for fdp contínua.

3) A, matriz de probabilidades de transições entre estados, $A=\{a_{ij}\}$.

$$a_{ij} = P(q_t=j / q_{t-1}=i) \qquad 1\leq i, j\leq N \tag{3.1}$$

em que q_t indica o estado atual. A matriz A deve satisfazer as condições estocásticas:

$$a_{ij} \geq 0, \qquad 1 \leq i, \quad j \leq N \tag{3.2}$$

e

$$\Sigma\, a_{ij} =1, \qquad 1\leq i \leq N. \tag{3.3}$$

Os valores das probabilidades de transições definem o tipo de topologia do HMM, através das restrições de avanço ou recuo entre estados.

4) B, representa a distribuição da probabilidade de observação em cada estado, $B = \{b_j(k)\}$. Para o HMM discreto, seus elementos são do tipo

$$b_j(k) = p\{o_t = v_k / q_t = j\},\ 1 \le j \le N,\quad 1 \le k \le M \tag{3.4}$$

onde $b_j(k)$ é a probabilidade da variável aleatória o_t (observação) pertencer ao estado j e v_k representa o k-ésimo símbolo observado no alfabeto. As condições estocásticas, abaixo, devem ser satisfeitas:

$$b_j(k) \ge 0, \qquad 1 \le j \le N, \qquad 1 \le k \le M \tag{3.5}$$

e

$$\sum_{k=1}^{M} b_j(k) = 1, \qquad 1 \le j \le N \tag{3.6}$$

No HMM contínuo como será visto na Item 3.4, o conjunto de observações pertencentes a cada estado é dividido em M grupos (através de um algoritmo de agrupamento), onde cada qual possui um vetor média e uma matriz covariância associados (Gaussiana). A densidade de probabilidade em cada estado é, então, calculada através de uma soma das M distribuições Gaussianas $\mathcal{N}$, ponderada por c_{jm}, ou seja uma mistura de Gaussianas.

$$b_j(o_t) = \sum_{m=1}^{M} c_{jm} \mathcal{N}(o_t, u_{jm}, U_{jm}) \tag{3.7}$$

onde: c_{jm} = coeficiente de ponderação.

u_{jm} = vetor média.

U_{jm} = matriz covâriancia.

c_{jm} deve satisfazer as condições estocásticas,

$$c_{jm} \ge 0, \qquad 1 \le j \le N, \qquad 1 \le m \le M \tag{3.8}$$

e

$$\sum_{m=1}^{M} c_{jm} = 1, \qquad 1 \le j \le N \tag{3.9}$$

5) Π_i, distribuição do estado inicial

$$\Pi = \{ \Pi_i \} \tag{3.10a}$$

$$\Pi_i = P \{ q_1 = i \}, \qquad 1 \leq i \leq N \tag{3.10b}$$

A notação usual para representar o modelo de Markov é através da forma compacta

$$\lambda = (A, B, \Pi) \tag{3.11}$$

3.3 - INFLUÊNCIA DAS PROBABILIDADES DAS TRANSIÇÕES

As probabilidades das transições definem o processo de Markov e sua ordem. Quando a transição feita para o estado atual não depender da ocorrência de todos os estados anteriores, mas, somente do estado imediatamente anterior, é caracterizado um Processo de Markov de Primeira Ordem. E se as variáveis aleatórias da seqüência S assumirem somente valores discretos (valores correspondendo aos estados do modelo), o processo de Markov então será chamado Cadeia de Markov. Portanto, a seqüência de estados do HMM é uma Cadeia de Markov, dado que as suas variáveis aleatórias assumem somente valores inteiros correspondendo aos estados do modelo[13].

De acordo com a matriz de transição A, a Cadeia de Markov assume uma topologia.

A topologia, ou estrutura de um HMM, é determinada pelas transições que ocorrem entre estados. Entre as estruturas mais utilizadas no HMM para o reconhecimento da voz estão o modelo ergódico e o modelo esquerda-direita ("left-right model")[12].

O **Modelo Ergódico** não restringe nenhuma transição entre estados, portanto, qualquer estado pode ser alcançado a partir de qualquer outro estado, isto é

$$a_{ij} \geq 0, \quad \text{para todos } i \text{ e } j \tag{3.12}$$

A Figura 3.1(a) mostra exemplo do modelo ergódico. Com esta estrutura obtém-se uma maior flexibilidade na geração das observações, embora ela não permita uma ótima representação da voz. A principal desvantagem é uma dificuldade em modelar a seqüência

temporal dos eventos acústicos em cada estado[13], além de aumentar o risco da convergência em um máximo local no processo de treinamento do HMM. Quando esta estrutura é usada para o treinamento da voz, as probabilidades das transições de retorno obtidas são próximas a zero[13].

Para o modelo com N=4 estados, da Figura 3.1(a), a sua matriz de probabilidade de transição A, será:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}$$

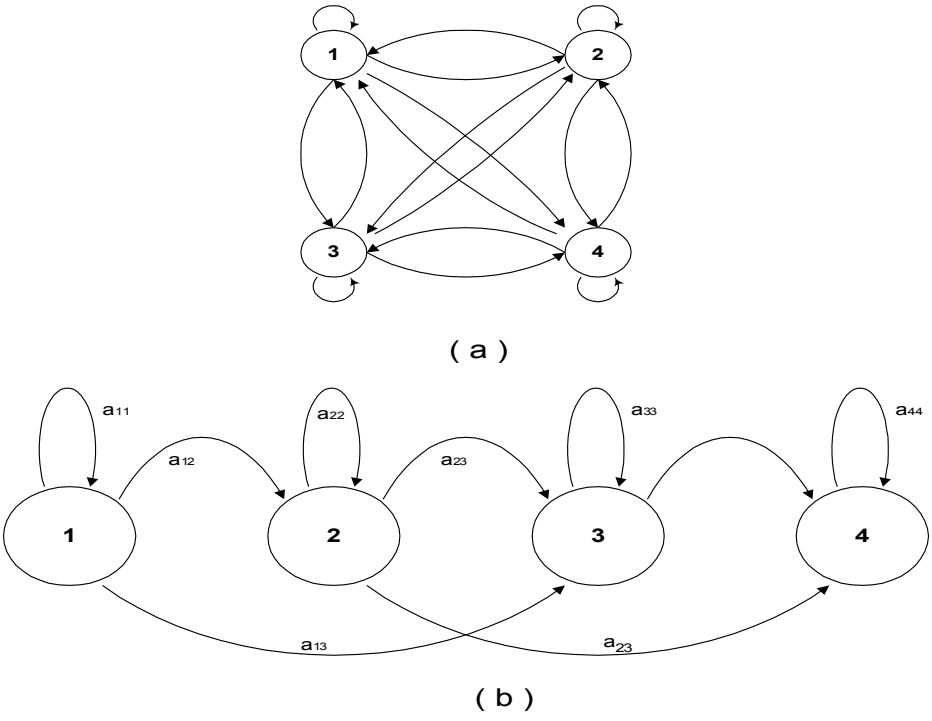


FIGURA 3.1: (a) Modelo ergódico com 4 estados; (b) Modelo esquerda-direita com 4 estados

Uma estrutura que melhor representa a variação temporal das características da voz deve possuir suas transições de retorno iguais a zero. Esta estrutura é chamada de **Modelo Esquerda-Direita,** como mostra a Figura 3.1(b), com a propriedade de:

$$a_{ij} = 0 \quad \text{para todo } j < i, \tag{3.13}$$

isto é, nenhuma transição será permitida para estados cujo índice seja menor do que o atual. A probabilidade do estado inicial será igual a:

$$\Pi i = \begin{cases} 0, & i \neq 1 \\ 1, & i = 1 \end{cases} \tag{3.14}$$

Frequentemente, são impostas restrições nas transições do modelo, atribuindo-se a cada transição um número máximo de estados que pode ser alcançado, ou seja,

$$a_{ij} = 0, \quad \text{para todos} \quad i > j+\Delta. \tag{3.15}$$

Para o exemplo da Figura 3.1(b) o valor de Δ é igual a 2, ou seja, permite a transição até o segundo estado a frente. Esta é uma particularidade do modelo esquerda-direita chamado de **Modelo Bakis** (Bakis, 1976). A forma da matriz de transição de estados do exemplo será:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 \\ 0 & a_{22} & a_{23} & a_{24} \\ 0 & 0 & a_{33} & a_{34} \\ 0 & 0 & 0 & a_{44} \end{bmatrix}$$

E o último estado, observado na matriz, terá a probabilidade de transição igual:

$$\begin{aligned} a_{NN} &= 1 \\ a_{Ni} &= 0, \quad i < N \end{aligned} \tag{3.16}$$

Existem outras variações de estruturas utilizadas em HMM, entretanto, as mais comuns são modelo ergódico, modelo esquerda-direita ou suas combinações. A Figura 3.2 ilustra combinações do modelo esquerda direita.

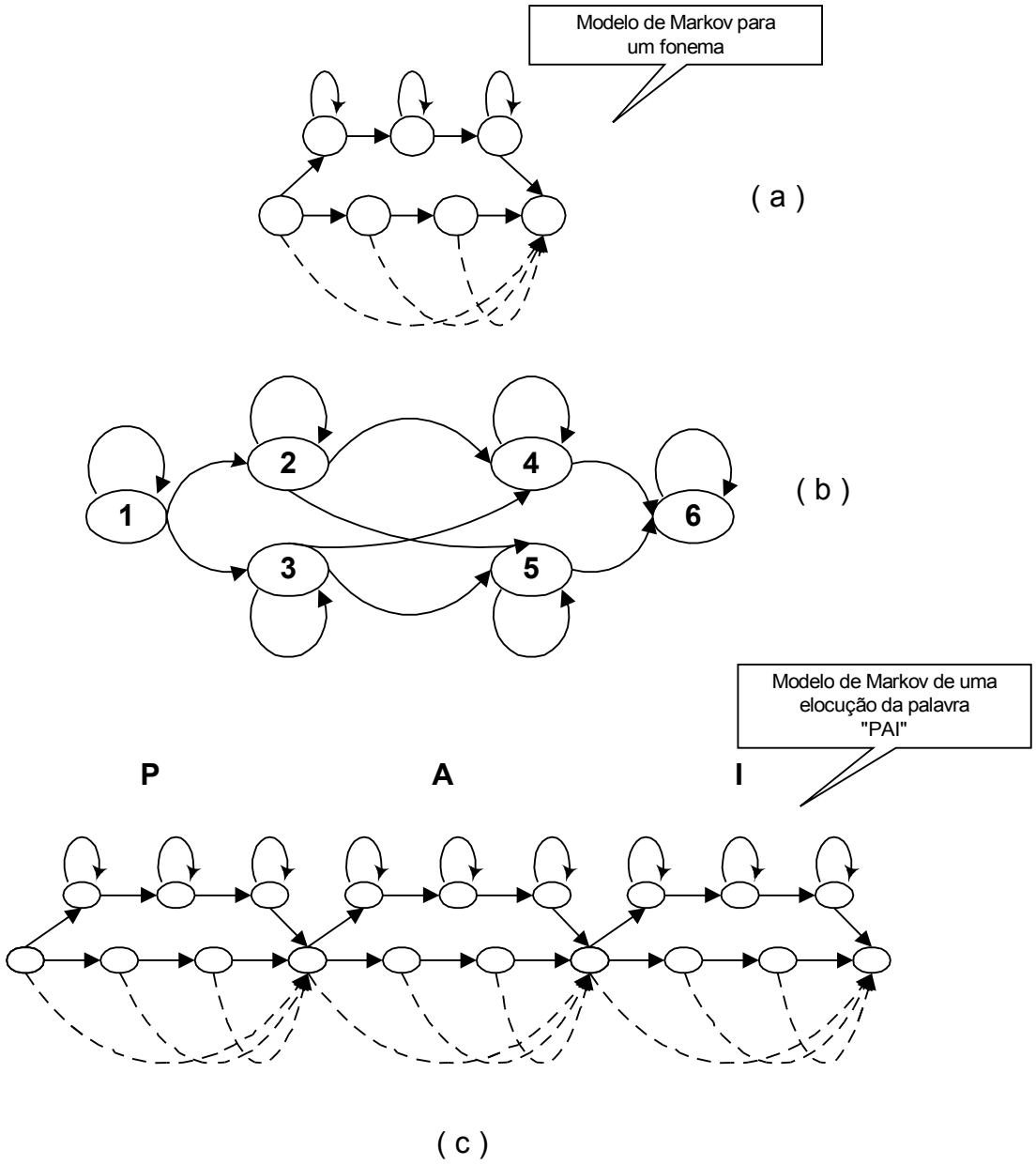


FIGURA 3.2: Exemplo de (a) modelo de um fonema, (b) modelo esquerda-direita paralelo com 6 estados e (c) modelo da palavra "PAI"

3.4 - INFLUÊNCIA DA DISTRIBUIÇÃO DE PROBABILIDADE DAS OBSERVAÇÕES.

Um HMM é definido por duas seqüências: estados, $S=\{S_1, S_2, S_3, ..., S_N\}$, e observações, $O=\{ o_1, o_2, o_3, ...,o_T\}$. Sabe-se que o modelo de Markov representa uma

modelagem estocástica da voz. Durante o treinamento, cada observação é associada a um estado de acordo com as probabilidades de transições entre estados e as probabilidades das observações dado o estado, apresentado na Figura 3.3(b,c). Como as observações são os vetores das características relevantes da voz, a sequência temporal (Figura 3.3(a)) é segmentada pelos estados obedecendo a algum evento acústico que será função da estrutura do modelo utilizada[12,13,14,30].

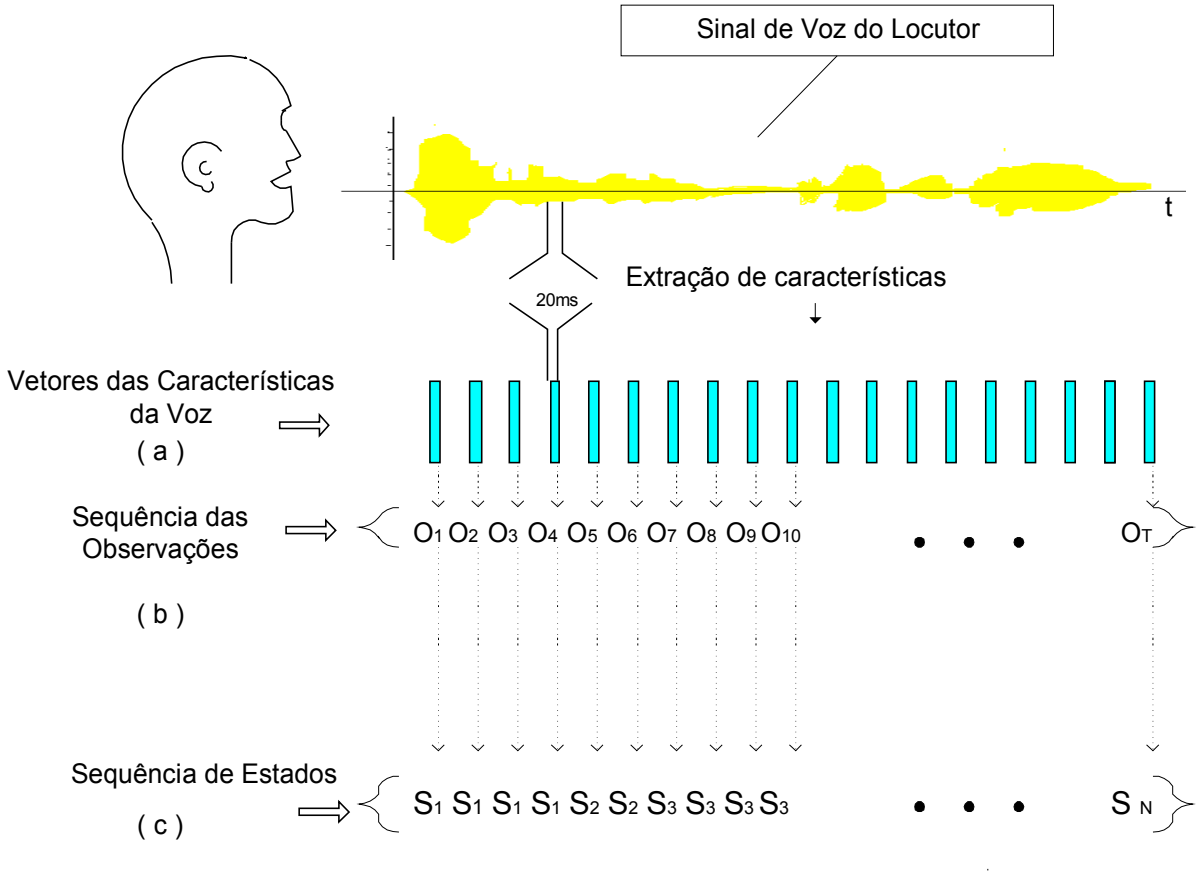


FIGURA 3.3: (a) Vetores das características da voz do locutor; (b) Seqüência das observações; (c) Seqüência de estados obtida durante o treinamento

No exemplo da Figura 3.4, três elocuções de uma mesma frase de um locutor, gravadas em horários diferentes foram treinadas em um HMM com cinco estados utilizando o modelo Bakis. Observa-se que, no treinamento, as seqüências das observações são segmentadas nos

estados conforme sua semelhança acústica (velocidade, entonação, emoção do falante, variação de início e fim das palavras, etc.) medida por uma distância entre padrões da voz. Esta medida representa o quão próximo um padrão está de um grupo, ou seja, de algum evento acústico. Após o treinamento, de todas as elocuções, as observações ficaram separadas em estados, levando em consideração as variações temporais das sequências das observações.

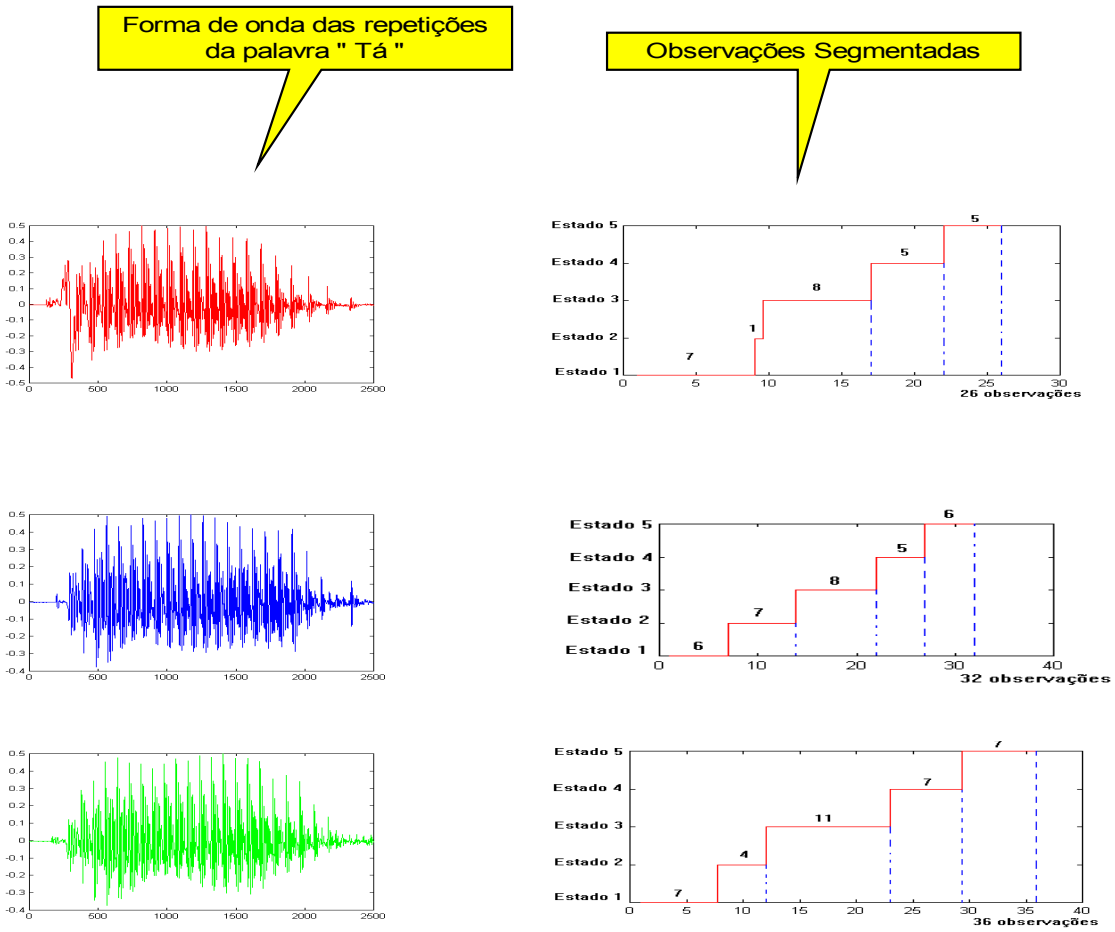
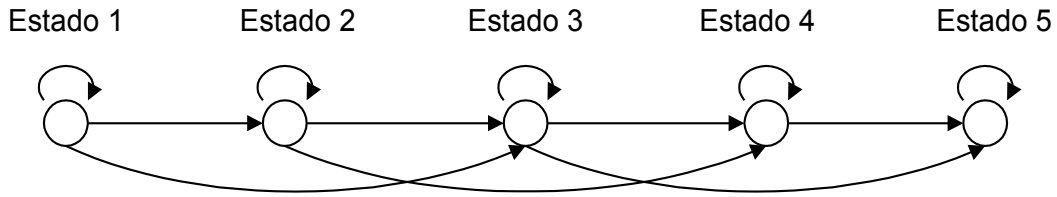


FIGURA 3.4: Mostra a variação do número de observações nos estados, para o mesmo tipo de palavra nas suas três repetições feitas por um mesmo locutor.

Observando-se a porcentagem acumulada das elocuções, na Figura 3.5, verifica-se que os mesmos fenômenos acústicos (observações) das repetições estão separados nos mesmos estados. E que, devido a existência de variabilidades acústicas, os vetores de características da voz podem apresentar uma distribuição multimodal. Assim, costuma-se, representar o espaço

espectral de cada estado[12,13,30,43], por uma densidade de probabilidade discreta (isto é, uma distribuição sobre todo o dicionário espectral), ou contínua (exemplo, mistura de fdp gaussianas) ou ainda semi-contínua (isto é, densidade contínua sobre um dicionário de formas espectrais comuns).

Sequência de Estados



Observações segmentadas nos Estados

	Estado 1	Estado 2	Estado 3	Estado 4	Estado 5
Locução 1	**O1 O2 O3 O4 O5 O6 O7**	**O8**	**O9 O10 O11 O12 O13 O14 O15 O16**	**O17 O18 O19 O20 O21**	**O22 O23 O24 O25 O26**
% acumulado	26,9%	30,77%	61,54%	80,77%	100%
Locução 2	**O1 O2 O3 O4 O5 O6**	**O7 O8 O9 O10 O11 O12 O13**	**O14 O15 O16 O17 O18 O19 O20 O21**	**O22 O23 O24 O25 O26**	**O27 O28 O29 O30 O31 O32**
% acumulado	18,75%	40,63%	65,63%	81,26%	100%
Locução 3	**O1 O2 O3 O4 O5O6 O7**	**O8 O9 O10 O11**	**O12 O13 O14 O15 O16 O17 O18 O19 O20 O21 O22**	**O23 O24 O25 O26 O27 O28 O29**	**O30 O31 O32 O33 O34 O35 O36**
% acumulado	19,44%	30,54%	61,13%	80,57%	100%

FIGURA 3.5: Cadeia de Markov utilizada no treinamento de três elocuções e resultados da segmentação das observações nos estados. Observa-se pelas percentagens, que as distribuições das observações foram as mesmas — mostrando uma tendência do HMM em separar os mesmos fenômenos acústicos nos mesmos estados.

Estas representações podem ser divididas em[12,13,30]:

1- Medição Acústica Não Paramétrica, Figura 3.6(a)

a) - Quantização Vetorial (QV)

2- Medição Acústica Paramétrica, Figura 3.6(b)

a) - Distribuição Gaussiana

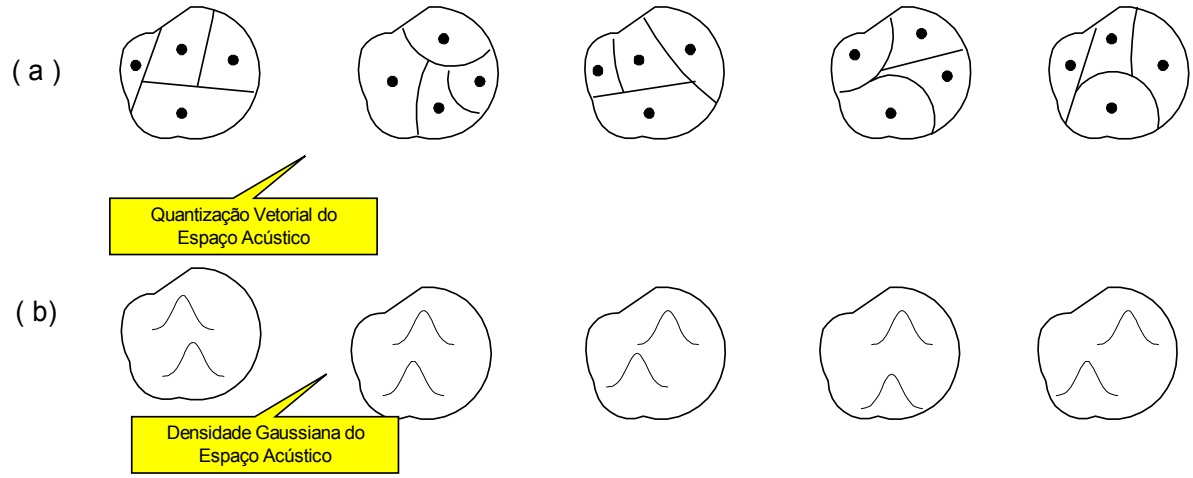


FIGURA 3.6: (a) Representação do espaço acústico por uma Quantização Vetorial
(b) Representação do espaço acústico por uma Mistura de Gaussianas;

3.4.1- Medição Acústica Não Paramétrica

DHMM - Discrete Hidden Markov Models.

Neste caso, a observação é associada a símbolos discretos, escolhidos de um alfabeto finito, sendo por isso, uma densidade discreta[12]. Assim,

$$b_j(k) = \text{Prob}(v_k \text{ em } t \,/\, q_t = S_j) \quad 1 \le j \le N, \quad 1 \le k \le M \qquad (3.17)$$

isto é, a probabilidade de observar o símbolo v_k (de uma célula k) dado o estado S_j, onde

q_t = estado no tempo t

v_k = vetor pertencente a célula quantizada k

Para se obter as probabilidades das observações pertencente a um determinado estado é necessário fazer a quantização vetorial, Figura 3.6(b), que representa um particionamento do espaço acústico em células não sobrepostas, normalmente baseado em um

critério de distorção espectral mínima. A Figura 3.7 exemplifica um o espaço acústico dividido em células.

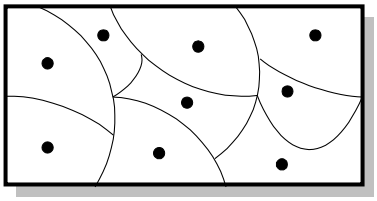


FIGURA 3.7: Particionamento do espaço acústico em N células.

Um dicionário pode ser constituído por um conjunto de vetores de características (considerando a seqüência de treinamento), aplicando-se algoritmos de agrupamento ("clustering algorithms"). Há atualmente duas aproximações utilizadas: o algoritmo "Kmeans"[44,45] (Figura 3.16) e o algoritmo Linde - Buzo - Gray[46]. Ambos algoritmos geram um espaço dimensional reduzido pela substituição de grupos de palavras códigos similares por uma única palavra código representando o centróide do grupo[12,13,14].

O vetor das características extraído do sinal de voz é atribuído a uma palavra código que produz a menor distorção, como demonstra a Figura 3.8. A medida de distorção mais usada é a distância ponderada Euclideana[12,13,14].

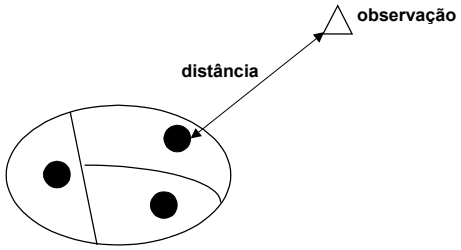


FIGURA 3.8: Espaço Acústico quantizado com 3 centróides e uma medida de distorção mínima entre um vetor e o centróide.

A Figura 3.9 mostra uma distorção típica versus o tamanho do dicionário[12].

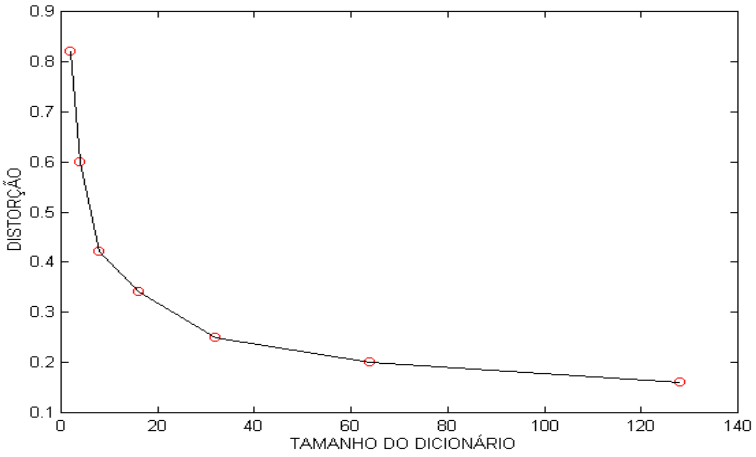


FIGURA 3.9: Gráfico típico da distorção versus o tamanho do dicionário

Devido ao fato de que o número de observações (vetores de características) classificado em cada estado normalmente é pequeno e que o erro de quantização pode provocar uma degradação da modelagem acústica do estado, duas dificuldades devem ser ponderadas[30]:

1- Um dicionário muito grande pode diminuir a performance do reconhecedor, uma vez que torna difícil a estimação e a interpolação dos parâmetros discretos do HMM (treinamento).

2- Um dicionário muito pequeno pode resultar numa performance não desejada do reconhecedor devido ao erro acrescentado oela quantização.

Um tamanho de dicionário razoável é de 256 níveis para as duas aproximações citadas acima, pois, melhora a resolução acústica e facilita a estimação dos parâmetros[30].

3.4.2- Medição Acústica Paramétrica

CDHMM - Continuous Density Hidden Markov Models.

O cálculo da densidade de probabilidade das observações é feito diretamente das observações, evitando a distorção acumulada que pode ocorrer no processo da QV. A

aproximação mais comum é o uso de uma função de densidade de probabilidade elipticamente simétrica[12], podendo ser uma distribuição Gaussiana que modele o espaço acústico das observações no estado[47,48,49], Figura 3.6(b). Então, a probabilidade de uma observação pertencer a um determinado estado no instante t é igual a[12] :

$$b_j(o_t) = \mathcal{N}\ (\ o_t\ ,\ u_j\ ,\ U_j\ ) \tag{3.18}$$

onde $\mathcal{N}$ representa uma distribuição Gaussiana, u_j a média e U_j a covariância das observações no estado j e o_t é o vetor de características do sinal da voz.

De acordo com a Equação 3.18, o espaço acústico do estado é modelado apenas por uma Gaussiana. Todavia, para o treinamento do HMM são necessárias várias repetições da mesma elocução, e, como existem muitas diferenças na pronúncia das repetições feitas por um mesmo locutor, a distribuição das observações dentro de um estado passa a ser multimodal. Uma forma bastante popular de aproximação desta fdp é a utilização de uma mistura de Gaussianas[12,13,14,47,48,49].

A mistura é uma soma linearmente ponderada de diferentes densidades Gaussianas que representa a distribuição de probabilidade das observações no estado.

$$b_j(o_t) = \sum_{m=1}^{M} c_{jm}\ N(o_t,\ u_{jm},\ U_{jm}) \tag{3.19}$$

sendo c_{jm} o coeficiente de ponderação, u_{jm} a média e U_{jm} a covariância do m-ésimo grupo.

Como a pesquisa da tese foi desenvolvida com o uso de mistura de funções de densidade contínuas, ou seja, misturas de Gaussianas, apenas este caso será tratado nos capítulos seguintes.

3.5 - SUPOSIÇÕES NECESSÁRIAS PARA A UTILIZAÇÃO DO HMM NO RECONHECIMENTO DO LOCUTOR.

Para tornar viável o emprego do HMM no Reconhecimento do Locutor, as seguintes suposições são admitidas[12,13,14,50,51].

(1) Suposição de Markov:

Como apresentado na Item 3.2, a probabilidade de transição é definida como

$$a_{ij} = p\left(q_{t+1} = j / q_t = i\right) \tag{3.20}$$

Ou seja, supõe-se que o próximo estado é dependente somente do estado atual, e o modelo resultante torna-se um HMM de primeira ordem. Entretanto, geralmente o próximo estado pode depender dos k estados passados e é possível obter um tal modelo, chamado HMM de k-ésima ordem, definindo as probabilidades de transições como se segue:

$$a_{i_1 i_2 i_3 i_4 j} = p\left(q_{t+1} = j / q_t = i_1, q_{t-1} = i_2 \ \ldots \ \ q_{t-k+1} = i_k\right) \quad 1 \le i_1, i_2, \ldots, i_k, j \le N \tag{3.21}$$

Normalmente não se utiliza Processo de Markov de maior ordem devido ao aumento no desempenho ser pequeno em relação ao aumento na complexidade no algoritmo.

(2) Suposição de estacionariedade.

É suposto que as probabilidades de transições entre estados são independentes do instante na qual ocorre, ou seja, para qualquer t_1 e t_2

$$p\left(q_{t1+1} = j / q_{t1} = i\right) = p\left(q_{t2+1} = j / q_{t2} = i\right) \tag{3.22}$$

(3) Suposição das observações independentes.

Supõe-se que a observação corrente é estatisticamente independente das anteriores, ou seja, considerando-se uma seqüência de observações

$$O = \{o_1, o_2, \ldots, o_T\} \tag{3.23}$$

$$p(O / q_1, q_2, \ldots, q_T, \lambda) = \prod_{t=1}^{T} p(o_t / q_t, \lambda) \qquad (3.24)$$

significa que nem os estados passados da cadeia de Markov nem as observações passadas influenciam na observação presente, se a última transição na cadeia é especificada.

3.6 - MODELOS DE MARKOV ESCONDIDOS

Para o desenvolvimento do algoritmo do modelo as seguintes fases são necessárias:

Fase 1 : Inicialização

a) - estrutura do modelo

b) - parâmetros fixos

c) - parâmetros variáveis

Fase 2 : Treinamento

a) - estimação dos parâmetros

b) - reestimação dos parâmetros

Fase 3 : Reconhecimento

As Figuras 3.10 e 3.11 mostram os fluxogramas das fases de treinamento e de reconhecimento.

3.6.1 - Inicialização

Na fase inicial do modelo utiliza-se dois tipos de parâmetros: fixos e variáveis. Os parâmetros fixos são as quantidades de estados e de fdp's da misturas, que não se alteram durante o treinamento e o reconhecimento. Os parâmetros variáveis são a matriz de

probabilidade de transição, o vetor média e a matriz covariância dos grupos. Ao contrário dos fixos, são apenas valores aleatórios ou estimativas iniciais que são ajustados durante o treinamento.

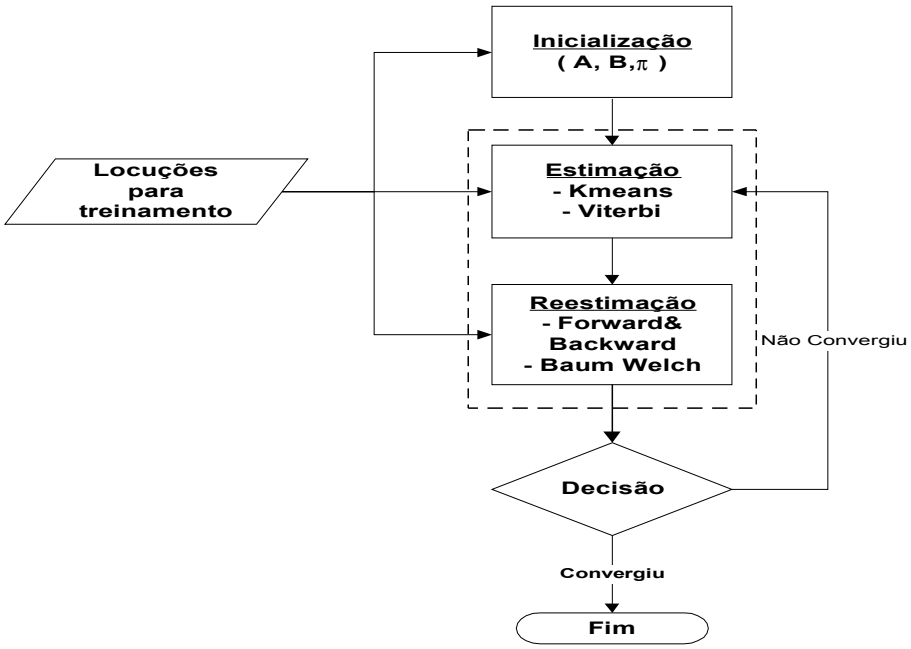


FIGURA 3.10: Fase de treinamento

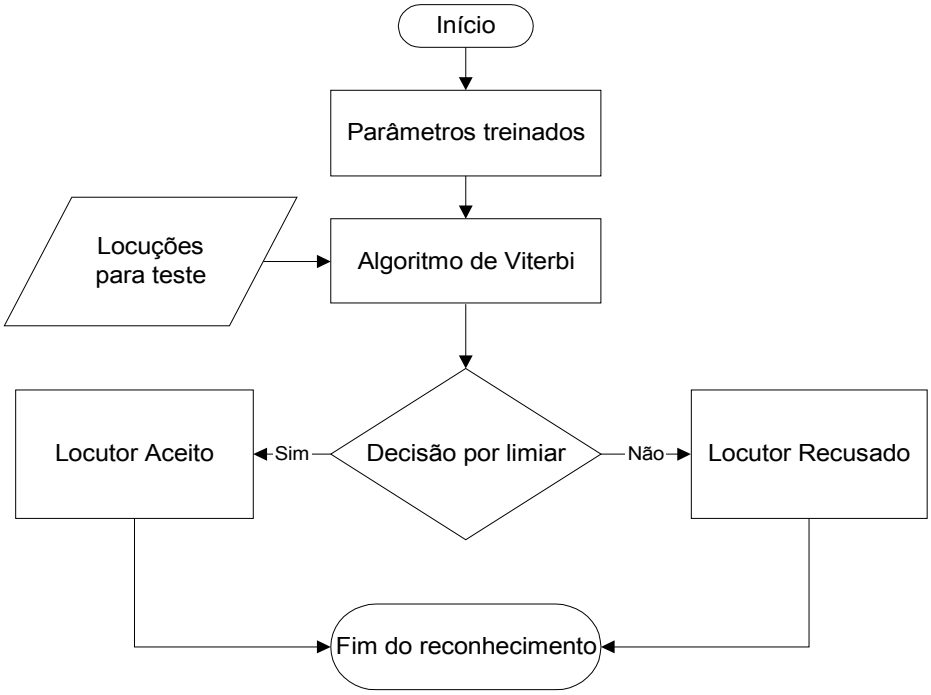


FIGURA 3.11: Fase de reconhecimento

3.6.1.1 - Estrutura do Modelo (Topologia)

Esta escolha é feita dependendo do sinal que será modelado[12]. O uso do modelo esquerda-direita é mais apropriado do que o modelo ergódico, devido à possibilidade de associação da evolução temporal do sinal com os estados do modelo, isto é, a possibilidade de sons distintos da elocução serem modelados nos estados de acordo com sua seqüência de observações[12].

3.6.1.2 - Parâmetros Fixos

N : número de estados

A escolha do número de estados pode estar relacionada à quantidade de eventos acústicos. Normalmente utiliza-se 2 a 10 estados por fonema[12], ou, aproximadamente o número médio de observações nas elocuções[12,13,14].

A Figura 3.12 mostra a relação da quantidade de estados versus convergência do modelo de uma palavra[12]. Observa-se que a quantidade N=6 pode ser um valor apropriado.

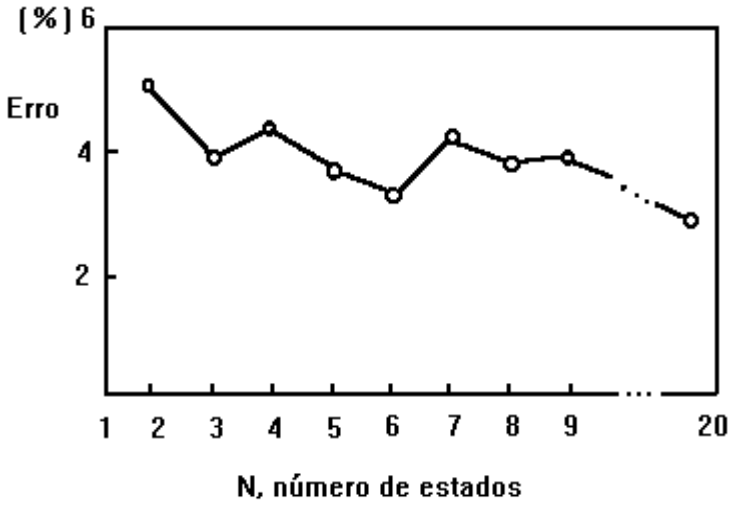


FIGURA 3.12:Taxa de erro médio versus o número de estados

M : número de grupos (Gaussianas)

Foi observado através de pesquisas[12], que o número de grupos em cada estado deve ser maior que a unidade. Pois, quando se constrói histogramas[12] das observações pertencentes a um determinado estado, verifica-se que possuem uma fdp multimodal, mostrando assim a necessidade de utilizar mais de uma distribuição Gaussiana (grupos) para o mapeamento das observações do estado.

Π_i : Probabilidade Inicial

Esta probabilidade é fixada igual a 1 para o primeiro estado nos modelos esquerda-direita, como visto no Item 3.3.

3.6.1.3 - Parâmetros Variáveis

A_{ij} : Probabilidade Inicial das Transições

Os valores atribuídos a matriz A podem ser inicialmente aleatórios. Observa-se nos resultados práticos, que esta atribuição não influencia os resultados finais, e que, depois de alguns ciclos de estimação e reestimação o mesmo resultado final pode ser obtido independentemente das condições iniciais utilizadas para a matriz A[12].

Para o modelo Bakis a matriz A possui a restrição de:

$$a_{ij} = 0, \text{ para todos } \quad i > j+2 \tag{3.25}$$

C_{jm} : Coeficientes de Misturas

O coeficiente de mistura é uma ponderação estocástica no somatório das Gaussianas. A restrição para a atribuição dos valores iniciais é a condição estocástica.

$$\sum_{m=1}^{N} c_{jm} = 1, \qquad 1 \le j \le N . \tag{3.26}$$

U_{jm} , μ_{jm} ,: Covariância e Média dos Grupos de Observações

O cálculo da densidade de probabilidade das observações influencia na performance do treinamento do modelo, deste modo não se atribui valores aleatórios para a média e a covariância.

Obtém-se uma distribuição uniforme e sequencial com a divisão das T observações de cada repetição pelo número de estados N. Após esta divisão, agrupa-se as observações de cada estado, separadamente, em M grupos através do algoritmo "Kmeans"[12]. Em seguida, obtém-se para cada grupo a média μ_{jm} e a covariância U_{jm}.

Em vários trabalhos foi utilizada a matriz covariância diagonal, ao invés da matriz completa. A razão para esta escolha foi a dificuldade de reestimar os elementos fora da diagonal da matriz, devido a limitação na quantidade de repetições para treinamento[12,13,14,29,43,52,53].

3.6.2 -Treinamento

Estimação e Reestimação

A partir dos parâmetros iniciais do modelo λ_i, estima-se o novo modelo, $\bar{\lambda}$, usando a seqüência de treinamento $O = \{o_1, o_2, \ldots, o_{T-2}, o_{T-1}, o_T\}$, tal que

$$\bar{\lambda} = \arg\max_{\lambda} P(O/\lambda) \tag{3.27}$$

Com boas estimativas iniciais dos parâmetros variáveis pode-se conseguir uma rápida convergência, utilizando-se um dos procedimentos iterativos da reestimação descritos nesta seção. Contudo, para uma dada seqüência de entrada, O, $P(O/\lambda)$ é geralmente uma função não linear dos parâmetros do modelo. Esta função por conseqüência terá muitos

máximos locais em um espaço multidimensional. A idéia é mostrada em duas dimensões na Figura 3.13.

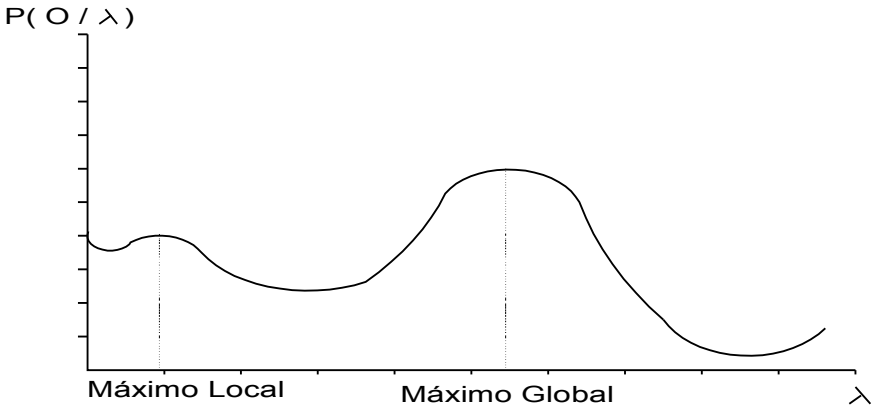


FIGURA 3.13: Conceituação da Verossimilhança do HMM como função dos parâmetros do modelo

O modelo ótimo, λ^*, corresponde ao máximo global da função, e a convergência neste ponto será quando $\bar{\lambda} = \lambda^*$. Através de um procedimento iterativo na reestimação onde $P(O/\bar{\lambda}) > P(O/\lambda)$ e os parâmetros são atualizados, se consegue convergir para um ponto de máximo local. Todavia, esta reestimação pode não convergir para o melhor modelo possível, λ^* (máximo global). Assim, é comum a prática de iniciar o algoritmo várias vezes com diferentes conjuntos de parâmetros iniciais e obter como modelo treinado aquele que possuir o maior valor de $P(O/\bar{\lambda})$ [13].

Os algoritmos mais utilizados nas pesquisas atualmente são: **Algoritmo de Reestimação Baum - Welch** (baseado no algoritmo Forward & Backward)[12,13], o **Procedimento de Viterbi** (baseado no algoritmo de Viterbi)[12,13,14,] e o **"Segmental K-means"**[12].

3.6.2.1- Método de Baum - Welch

O Método de Baum - Welch[12,13,54] é um procedimento iterativo, onde escolhido o modelo inicial, $\lambda_i = (A, B, \pi)$, encontra-se a máxima verossimilhança (ML - Maximum Likelihood) dos parâmetros do modelo. O método Baum - Welch basea-se no conceito estatístico da esperança do número de transições entre estados e da esperança do número de ocorrência das observações nos estados. Usa-se o número esperado porque esta ferramenta estatística é a média sobre uma grande quantidade de dados.

O método basea-se no cálculo das variáveis "Forward" e "Backward", obtidas através do **Procedimento "Forward-Backward"**. Para compreensão deste método utiliza-se uma treliça de estados (Figura 3.14), assim composta: na vertical os estados e na horizontal as observações.

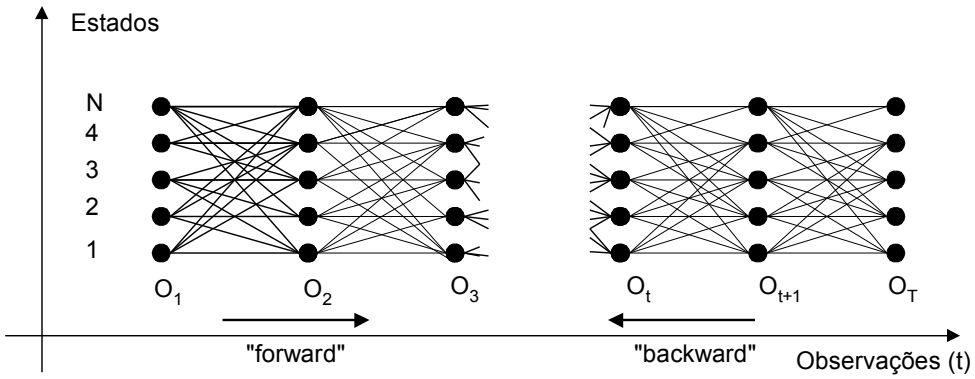


FIGURA 3.14: Implementação do cálculo das variáveis "Forward" e "Backward".

Verifica-se que cada observação possui uma probabilidade inicial de ocorrência em cada estado para t = 1 dada pela probabilidade conjunta do evento, observação o_1 e estado i ($1 \leq i \leq N$). Para t > 1, a probabilidade de alcançar um estado j qualquer será o

somatório do produto da probabilidade de estar no estado i no instante t com a probabilidade de transição, a_{ij}, do estado i para o estado j em t+1com $1 \leq i \leq N$, como mostra a Figura 3.15.

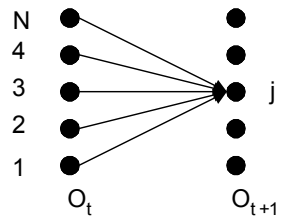


FIGURA 3.15: Probabilidade conjunta de ocorrência da observação o_{t+1} e o estado j.

Com a recursão, este cálculo é efetuado para todas as observações de 1 a T. O somatório de todas as verossimilhanças dos caminhos possíveis (para $1 \leq j \leq N$) fornece a máxima verossimilhança da seqüência de observações, dado o modelo λ.

As variáveis "forward" e "backward" são obtidas como o somatório das verossimilhanças dos caminhos que poderiam ter gerado as seqüências parciais O_1^t e O_t^T respectivamente, definidos como[12]:

a) $\alpha_t(i) \rightarrow$ a variável "forward" representa a probabilidade conjunta da seqüência parcial $O_1^t = [o_1, o_2, \ldots, o_t]$ e a observação o_t ocorrer no estado i no tempo t dado o modelo λ.

$$\alpha_t(i) = P(o_1 o_2 \ldots o_t, q_t / \lambda) \tag{3.28}$$

b) $\beta_t(i) \rightarrow$ a variável "backward" representa a probabilidade conjunta da seqüência parcial $O_{t+1}^T = [o_{t+1}, o_{t+2}, \ldots, o_T]$ dado a observação o_t pertence ao estado i e o modelo λ.

$$\beta_t(i) = P(o_{t+1} o_{t+2} \ldots o_T / q_t = i, \lambda) \tag{3.29}$$

No tempo t=1 (a primeira faixa na treliça) calcula-se os valores iniciais de $\alpha_1(i)$, $1 \leq i \leq N$ (α - variável forward). Nos tempos t=2,3,..., T os valores de $\alpha_t(j)$ são calculados recursivamente para $1 \leq j \leq N$.

O algoritmo é sumarizado a seguir[12]:

1- Procedimento "Forward"

1- início

$$\alpha_1(i) = \pi_i \cdot b_i(O_1) \qquad 1 \leq i \leq N \tag{3.30}$$

2 - recursão

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) \cdot a_{ij} \right] \cdot b_j(O_{t+1}) \qquad 1 \leq j \leq N \quad 1 \leq t \leq T-1 \tag{3.31}$$

3 - fim

$$P(O/\lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{3.32}$$

este último passo fornece o cálculo de $P(O/\lambda)$, como o somatório das variáveis "forward" $\alpha_T(i)$. Por definição

$$\alpha_T(i) = P(o_1 o_2 \ldots o_T, q_T = i/\lambda) \tag{3.33}$$

e então $P(O/\lambda)$ é exatamente a soma dos $\alpha_T(i)$'s.

2 - Procedimento "Backward"[12]

1 - início

$$\beta_T(i) = 1 \qquad 1 \leq i \leq N \tag{3.34}$$

2 - recursão

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} \cdot b_j(O_{t+1}) \cdot \beta_{t+1}(j) \quad 1 \leq i \leq N \quad t = T-1, T-2, \ldots, 1 \tag{3.35}$$

3 - fim

$$P(O/\lambda) = \sum_{i=1}^{N} \beta_1(i) \tag{3.36}$$

Após o cálculo de α e β, calcula-se a probabilidade a posteriori definida por

$$\gamma_t(i,k) = P(q_t = i, m = k / O, \lambda) \tag{3.37}$$

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)} \right] \left[\frac{c_{jk} \mathcal{N}(o_t, \mu_{jk}, U_{jk})}{\sum_{m=1}^{M} c_{jm} \mathcal{N}(o_t, \mu_{jm}, U_{jm})} \right] \tag{3.38}$$

ou seja, a probabilidade conjunta de estar no estado i e na mistutra k dado o modelo e a seqüência de observações.

Os valores dos parâmetros estimados do HMM serão[12]:

1- probabilidade inicial

$\hat{\pi}_i \rightarrow$ número esperado de vezes no estado S_i no tempo t = 1

$$\hat{\pi}_i = \alpha_1(i)\beta_1(i) \tag{3.39}$$

2- probabilidade de transição

$$\hat{a}_{ij} \rightarrow \frac{\text{número esperado de transições do estado Si para Sj}}{\text{número esperado de transições do estado Si}}$$

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1} \alpha_t(i) \cdot a_{ij} \cdot b_j(O_{t+1}) \cdot \beta_{t+1}(j)}{\sum_{t=1}^{T-1} \alpha_t(i) \cdot \beta_t(i)} \tag{3.40}$$

3- probabilidade da observação dado o estado

$$\hat{b}_j(O_t) = \sum_{k=1}^{M} c_{jk} \mathcal{N}(o_t, \mu_{jm}, U_{jm}) \qquad 1 \le j \le N \tag{3.41}$$

onde seus parâmetros serão calculados de acordo com:

$\hat{c}_{jk}$ → número esperado de ocorrer a Gaussiana k no estado j / número esperado de ocorrer o estado Sj

$$\hat{c}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)}{\sum_{t=1}^{T}\sum_{k=1}^{M} \gamma_t(j,k)} \quad \text{coeficiente de mistura} \tag{3.42}$$

$\hat{\mu}_{jk}$ → número esperado de ocorrer a Gaussiana k no estado j ponderada pela observação o_t / número esperado de ocorrer o estado Sj e na mistura k

$$\hat{\mu}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) \cdot o_t}{\sum_{t=1}^{T} \gamma_t(j,k)} \quad \text{média da mistura} \tag{3.43}$$

$\hat{U}_{jk}$ → número esperado de ocorrer a Gaussiana k no estado j ponderado pela matriz covariância / número esperado de estar no estado Sj e na mistura k

$$\hat{U}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) \cdot [(o_t - \mu_j)(o_t - \mu_j)^T]}{\sum_{t=1}^{T} \gamma_t(j,k)} \quad \text{covariância da mistura} \tag{3.44}$$

Deste modo, define-se o modelo reestimado $\hat{\lambda} = (\hat{A}, \hat{B}, \hat{\pi})$ determinado a partir do modelo anterior $\lambda_i = (A_i, B_i, \pi_i)$.

Baseado no procedimento acima, usa-se $\hat{\lambda} = (\hat{A}, \hat{B}, \hat{\pi})$ no lugar de λ_i, repete-se o cálculo da reestimação e encontra-se um ponto limite quando não houver melhoria em $P(O/\hat{\lambda})$ (ou seja, no terceiro passo do algoritmo forward, Equação 3.32).

Foi provado por Baum e seus auxiliares que[12]:

1- o modelo inicial λ define um ponto crítico da função de verossimilhança no caso que $\hat{\lambda} = \lambda$.

2- o modelo $\hat{\lambda}$ é mais provável do que o modelo λ no sentido que $P(O/\hat{\lambda}) > P(O/\lambda)$.

3.6.2.2- Procedimento de Viterbi

O Procedimento de reestimação Baum-Welch pode ser substituído pelo Procedimento otimizado de Viterbi, baseado no algoritmo de Viterbi[14,55]. Os parâmetros de λ são reestimados, e ao contrário, do Procedimento de Baum-Welch onde se calcula o número esperado dos eventos, no Procedimento de Viterbi soma-se as transições e as observações pertencentes a cada estado, de acordo com o melhor caminho (seqüência) de estados.

Portanto, torna-se necessário para o desenvolvimento do Procedimento de Viterbi, a apresentação do seu algoritmo. Define-se a variável:

$$\delta_t(i) = \max_{q_1,q_2,\ldots,q_{t-1}} P[q_1 q_2 \ldots q_{t-1}, q_t = i, o_1 o_2 \ldots o_t / \lambda] \tag{3.45}$$

onde, $\delta_t(i)$ é a maior probabilidade ao longo do melhor caminho da seqüência, até o tempo t, para as t primeiras observações finalizando no estado i.

O algoritmo de Viterbi é sumarizado abaixo:

1) início

$$\delta_1(i) = \pi_i b_i(o_1) \qquad 1 \le i \le N \tag{3.46}$$

$$\psi_1(i) = 0 \qquad 1 \le i \le N \tag{3.47}$$

2) recursão

$$\delta_t(i) = \max_{1 \le i \le N} [\delta_{t-1}(i)\, a_{ij}]\, b_j(o_t) \qquad 1 \le j \le N \quad 2 \le t \le T \tag{3.48}$$

$$\psi_t(i) = \arg\max_{1 \le i \le N} [\delta_{t-1}(i)\, a_{ij}] \qquad 1 \le j \le N \quad 2 \le t \le T \tag{3.49}$$

3) finalização

$$P = \max_{1 \le i \le N} \delta_T(i) \tag{3.50}$$

$$q_T = \arg\max_{1 \le i \le N} \delta_T(i) \tag{3.51}$$

4) O caminho ótimo é obtido retornando

$$q_t = \psi_{t+1}(q_{t+1}) \qquad t = T-1, T-2, \ldots, 1 \tag{3.52}$$

Após a segmentação das observações, de cada repetição, pelo algoritmo de Viterbi, consegue-se a melhor seqüência de estados. Por exemplo, na tabela abaixo, onde o_T^S é observação no tempo T e estado S, admite-se que os caminhos ótimos, dado pelo algoritmo de Viterbi, das quatro repetições de treinamento sejam:

Rep 1 [	o_1^1 o_2^1	o_3^2 o_4^2	o_5^3	o_6^4	....o_T^N]
Rep 2 [	o_1^1	o_2^2 o_3^2	o_4^3 o_5^3 o_6^3		....o_T^N]
Rep 3 [	o_1^1	o_2^2 o_3^2 o_4^2	o_5^3 o_6^3		....o_T^N]
Rep 4 [	o_1^1	o_2^2	o_3^3 o_4^3 o_5^3	o_6^4	....o_T^N]

Os parâmetros $\hat{a}_{ij}$ reestimados, são obtidos através da contagem do número das transições do estado i para o estado j, dividido pelo número de todas as transições feitas a partir do estado i (inclusive)

$$\hat{a}_{ij} \rightarrow \frac{\text{número de transições do estado Si para Sj}}{\text{número de transições do estado Si}},$$

do exemplo acima

$$a_{12} = \frac{4}{5}.$$

Os parâmetros média, covariância e coeficiente de mistura são obtidos para cada estado, após o agrupamento dos vetores de observações em M grupos (algoritmo "Kmeans"[45], Figura 3.16). A média é simplesmente estimada pela média de todas as

observações pertencentes àquela Gaussiana, o mesmo para a covariância. O coeficiente de misturas (quando M > 1) será igual ao número de observações classificado no grupo dividido pelo número total de observações classificado naquele estado.

Assim, para

$$\hat{b}_j(o_t) = \sum_{k=1}^{M} c_{jk} \mathcal{N}(o_t, \mu_{jm}, U_{jm}) \qquad 1 \le j \le N \tag{3.53}$$

a) a média reestimada será

$$\hat{\mu}_{jm} = \frac{1}{N_{jm}} \sum_{i=1}^{Njm} o_i \tag{3.54}$$

b) a covariância reestimada será

$$\hat{U}_{jm} = \frac{1}{N_{jm}} \sum_{i=1}^{Njm} (o_i - \mu_{jm})(o_i - \mu_{jm})^T \tag{3.55}$$

onde o_i é a i-ésima observação associado ao estado j e Gaussiana m, a qual possui Njm observações classificadas.

c) o coeficiente de mistura reestimado será

$$c_{jm} = \frac{N_{jm}}{Nj} \tag{3.56}$$

onde Nj é o número de observações no estado j e N_{jm} número de observações na m-ésima mistura do estado j.

A Figura 3.16 mostra o algoritmo[45] "Modified K-means". Um algoritmo que agrupa os padrões (ou vetores das características da voz do locutor) das repetições da elocução, classificadas em um estado, em M grupos tal que dentro de cada grupo os padrões sejam bastante similares.

No fluxograma, denota-se o i-ésimo grupo de um grupo solução j, na k-ésima iteração, como $(w_j^i)^{(k)}$, onde i = 1,2 , ...,j, e k = 0, 1, 2, ..., $K_{máx}$ (onde $K_{máx}$ é o contador

de iteração máxima). Os valores de j vai de 1 (único grupo) até $J_{máx}$ (máximo número de grupos).

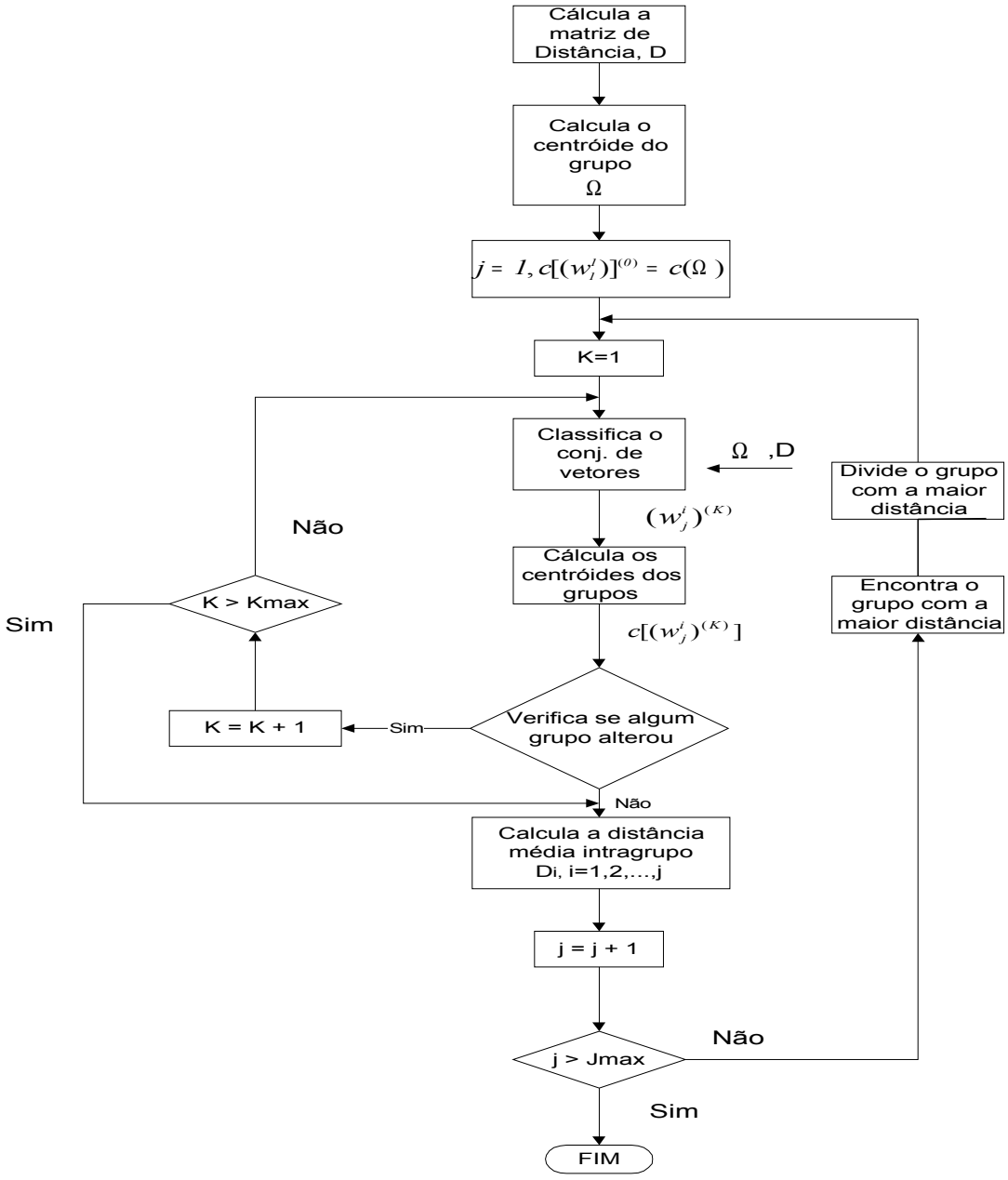


FIGURA 3.16: Fluxograma do Algoritmo "Kmeans" Modificado (MKM)

3.6.2.3- "Segmental K-means"

Como a convergência, dos algoritmos apresentados, é muito sensível aos valores iniciais de $b_j(o)$, Rabiner[56] apresentou um algoritmo de treinamento, o "Segmental K-means" (Figura 3.17) usado para estimar os valores dos parâmetros do modelo λ.

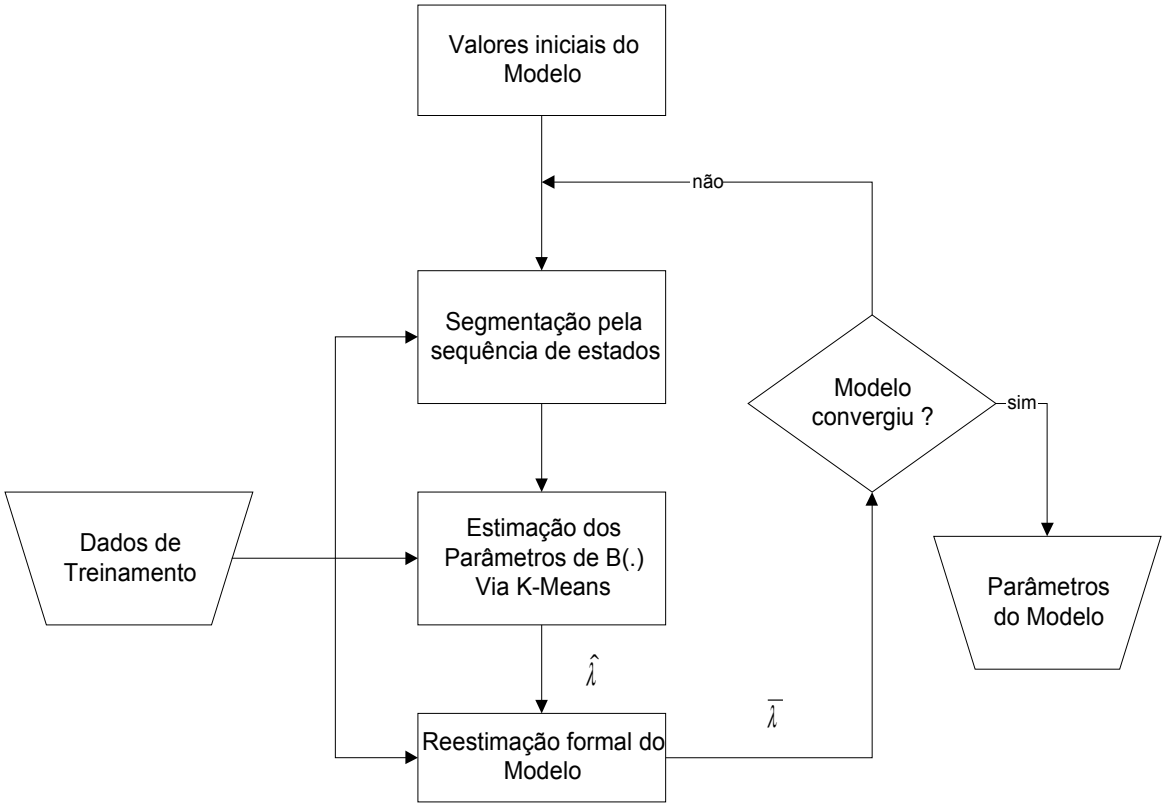


FIGURA 3.17: Algoritmo "Segmental Kmeans"

De acordo com a Figura 3.17, a partir dos valores iniciais dos parâmetros do modelo λ_i, aplica-se o Procedimento de Viterbi (Item 3.6.2.2), e obtém-se então o modelo $\hat{\lambda} = (\hat{A}, \hat{B}, \hat{\pi})$. Estes parâmetros são utilizados na reestimação formal, através do Método Baum-Welch (Item 3.6.2.1), onde se reestima todos os parâmetros.

O modelo resultante é comparado com o modelo anterior através do valor da verossimilhança, que pode ser obtida pelo algoritmo de Viterbi. Se o valor de verosimilhança exceder um limiar,

$$P(O/\hat{\lambda}) - P(O/\lambda) \leq \theta \qquad (3.57)$$

os parâmetros anteriores serão substituídos pelos atuais e todo o treinamento será repetido. Caso contrário, terá convergido e os parâmetros são admitidos como do modelo treinado.

3.6.3 - Reconhecimento

Para reconhecer se uma elocução teste pertence ou não a um determinado locutor (verificação), calcula-se a verossimilhança da elocução P(O/λ), de ter sido gerada por este modelo. Aceita se for maior ou igual a um determinado limiar.

Na existência de vários modelos (identificação), a medida de verossimilhança obtida em todos os modelos dos locutores provê uma comparação relativa entre os mesmos, aceitando-se aquele que possuir a maior.

Para o cálculo da verossimilhança utiliza-se o algoritmo de Viterbi, o qual fornece o caminho de máxima verossimilhança, da elocução teste, em pertencer ao modelo treinado. Este valor é comparado a um limiar, obtido por meio de algum método de decisão.

Um dos métodos utilizados para o cálculo do limiar é o método de Bayes[60], Figura 3.18. Assume-se que as distribuições obtidas com os valores das verossimilhanças para o locutor treinado e para os locutores não treinados são Gaussianas. Como mostra a Figura 3.18, calcula-se as médias e as variâncias intra-locutor e inter-locutores, obtendo $\mathcal{N}_{verd}(\mu_1, \sigma_1)$ e $\mathcal{N}_{falso}(\mu_2, \sigma_2)$.

Igualam-se as duas distribuições e encontra-se o ponto médio (limiar) θ.

$$\mathcal{N}_{verd} = \mathcal{N}_{falso} \qquad (3.58)$$

$$\frac{1}{\sigma_1}\exp[\frac{-(\theta-\mu_1)^2}{2\sigma_1^2}] = \frac{1}{\sigma_2}\exp[\frac{-(\theta-\mu_2)^2}{2\sigma_2^2}] \quad (3.59)$$

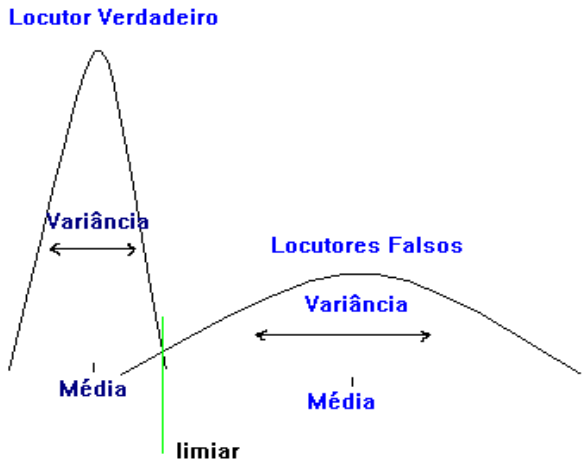


FIGURA 3.18: Obtenção do limiar através do Método de Bayes

A Figura 3.19 mostra a área de superposição entre as duas distribuições e um ponto de intersecção entre as curvas. Esta área indica o quanto há de incerteza entre aceitar um locutor falso (falsa aceitação) ou rejeitar um locutor verdadeiro (falsa rejeição). O ponto de intersecção, será o limiar entre as duas suposições. Podendo ser ajustado até encontrar um valor ótimo para os dados de teste.

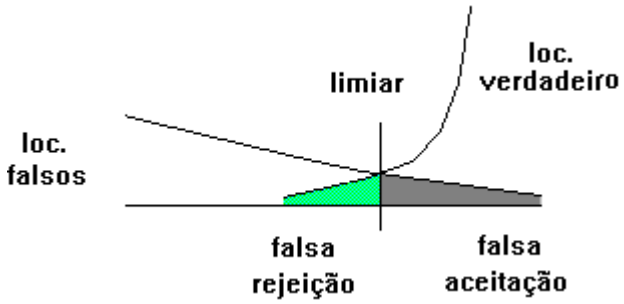


FIGURA 3.19: Limiar de reconhecimento

Na verificação do locutor, a elocução de um locutor supostamente conhecido é comparado com o modelo pretendido, se a verossimilhança for um valor acima do limiar θ, a

identidade é verificada como verdadeira. Um limiar alto dificulta a falsa aceitação pelo sistema, mas aumenta o risco de ocorrer falsa rejeição. E, ao contrário, um limiar baixo garante a aceitação de todos os locutores verdadeiros mas aumenta o risco da falsa aceitação.

A Figura 3.20 apresenta a identificação de um determinado sinal de voz como pertencente a um dos modelos treinados.

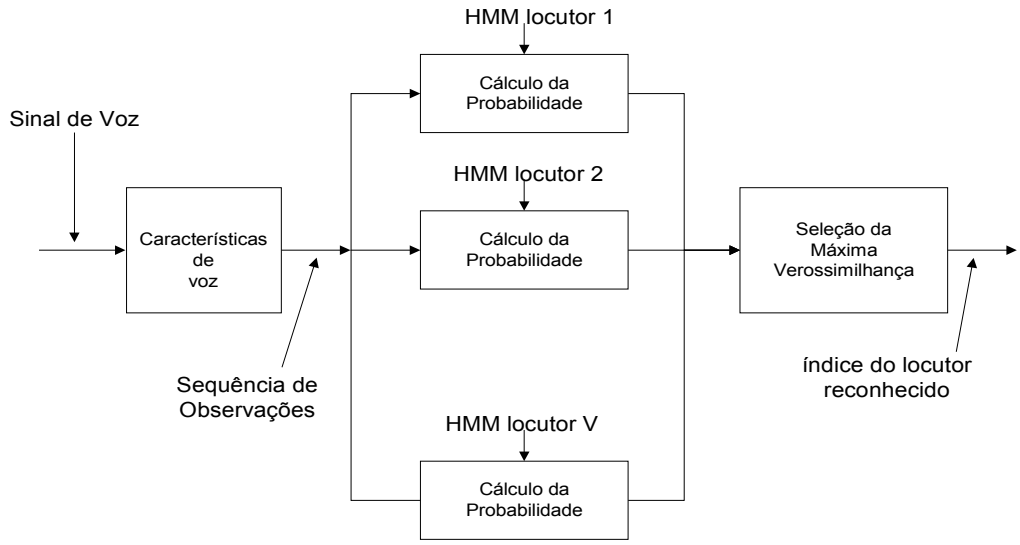


FIGURA 3.20: Procedimento para identificação de um locutor

Quando existem V modelos treinados e uma elocução pretensa, pode-se fazer a identificação. Na identificação compara-se a verossimilhança da seqüência de observações (calculado pelo algoritmo de Viterbi) com todas as verossimilhanças dos modelos treinados, $P(O/\bar{\lambda}_v)$, $1 \leq v \leq V$ e seleciona-se o locutor referente ao HMM com a maior verossimilhança (Figura 3.20). Desta forma,

$$v' = \arg\max_{1 \leq v \leq V} P(O/\bar{\lambda}_v) \tag{3.60}$$

Na identificação com rejeição, após a seleção do modelo com maior verossimilhança, faz-se a comparação desta com o limiar de aceitação associado. Se for menor que este limiar, o locutor era considerado falso.

CAPÍTULO 4

DESENVOLVIMENTO DO ALGORITMO "SEGMENTAL KMEANS"

4.1- INTRODUÇÃO

De acordo com o Item 3.6.1.3, o modelo é muito sensível aos parâmetros iniciais, principalmente os valores iniciais das probabilidades das observações. Utiliza-se, então, o artifício de dividir as observações, sequencialmente, por estados e obter a probabilidade inicial da observação, o_t, em pertencer aos estados. Com esta divisão os valores das probabilidades serão assumidos de acordo com a Figura 4.1, onde a segmentação das observações pelo número de estados proporciona valores de probabilidades maiores para as observações pertencentes ao estado mais próximo. E, durante o treinamento os parâmetros serão reestimados.

Este procedimento de treinamento é apresentado na Figura 4.2 onde mostra o algoritmo "Segmental K-means"[12]. Através dos parâmetros iniciais estima-se novos valores pelo Procedimento de Viterbi (Item 3.6.2.2), e, por ser um procedimento iterativo, os parâmetros estimados serão reestimados pelo método de Baum-Welch[12,57]. Verifica-se a convergência do modelo (Equação 3.57) e, caso este não ocorra, os parâmetros $\bar{\lambda}$ reestimados serão estimados em uma nova iteração.

No Item 3.6.2.3 foi apresentado o modelo de treinamento de forma resumida. Neste capítulo será apresentado a estrutura do algoritmo admitindo-se o uso de várias repetições da locução para o treinamento. Divide-se em duas partes[12]: treinamento (algoritmo "Segmental Kmeans") e reconhecimento (algoritmo de Viterbi).

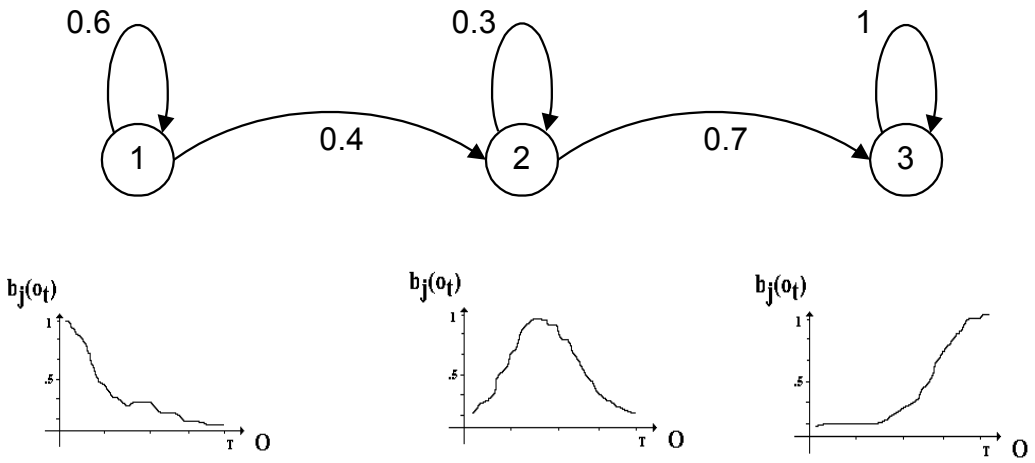


FIGURA 4.1: Exemplo de valores iniciais que podem ser atribuídos às probabilidades das observações.

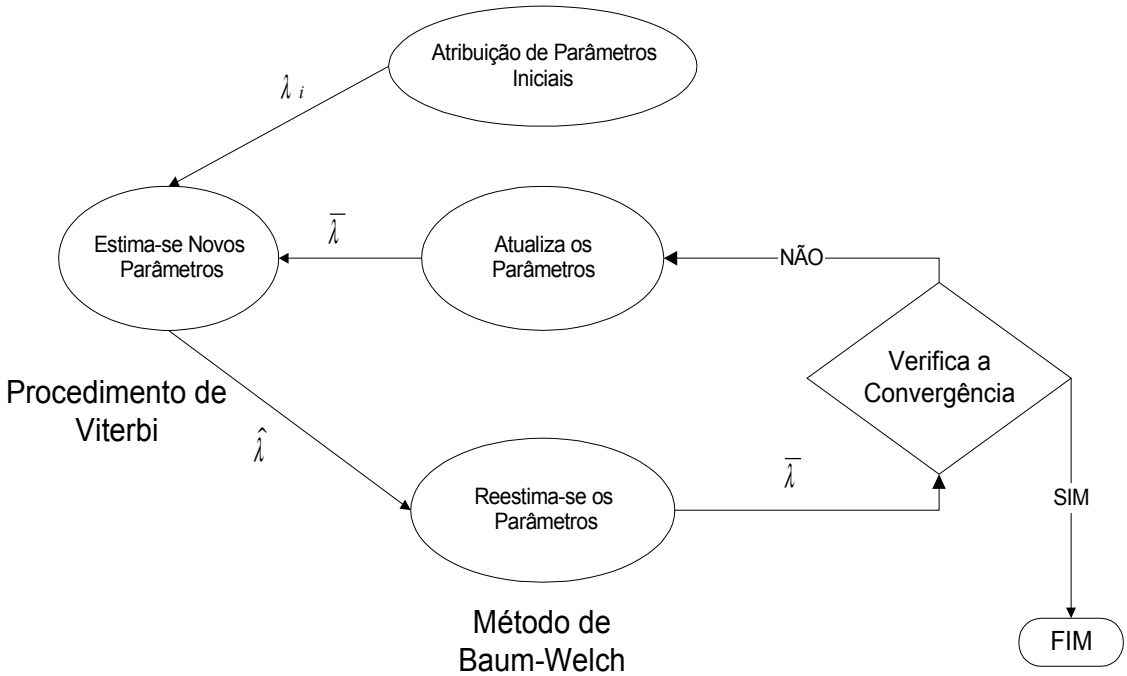


FIGURA 4.2: Sequência do algoritmo "Segmental Kmeans"

4.2- TREINAMENTO

Para o treinamento, escolhe-se os parâmetros iniciais, dentro de algum critério, e aplica-se o algoritmo "Segmental Kmeans", dividido em estimação e reestimação.

4.2.1- Parâmetros Iniciais $\lambda_i = (A_i, B_i, \pi_i)$

Para iniciar o modelo, como visto no Item 3.6.1, utilizam-se dois tipos de parâmetros: os fixos e os variáveis. A escolha dos valores dos parâmetros fixos (número de estados e número de Gaussianas) não são comentados aqui, pois consistem em um dos objetos da pequisa. A topologia do modelo (também parâmetro fixo) usada foi o Modelo Bakis (um caso particular da estrutura esquerda-direita).

Os parâmetros variáveis serão apresentados em forma de algoritmo comentado.

1) $A_i \rightarrow$ **Probabilidades Iniciais de Transições**

Mostra-se[12,13] que a atribuição de uma matriz de probabilidades de transições iniciais aleatória não influencia no resultado final dos parâmetros. Desta forma, usa-se uma matriz com valores randômicos entre zero e um, respeitando-se a restrição estocástica e do avanço e retorno entre estados do modelo Bakis.

2) $B_i \rightarrow$ **Probabilidade das Observações**

De acordo com os autores Rabiner, Deller, os valores iniciais das probabilidades das observações influenciam na convergência do treinamento do modelo. Assim, apresenta-se abaixo um algoritmo para estimar valores iniciais.

Algoritmo para o cálculo das Probabilidades das Observações[12]:

- Para cada repetição, as T observações serão divididas pelo número de estados (N).

- Agrupa-se as observações nos estados de acordo com a divisão: número de observações por número de estados (T/N).
- Utiliza o algoritmo K-means, em cada estado, para agrupar as observações em M grupos.
- Para cada grupo encontra-se a média μ_{jk}, a covariância U_{jk}, e o coeficiente de mistura c_{jk} de acordo com as equações 3.54, 3.55, 3.56.

início

para cada repetição

para cada estado j

para cada grupo

para cada vetor de característica O_t

$$b_j(o_t) = \sum_{m=1}^{M} c_{jm} \mathcal{N}(o_t, \mu_{jm}, U_{jm}) \quad (4.1)$$

fim

fim

fim

fim

Deste modo, obtém-se todos os parâmetros iniciais do modelo

$$\lambda_i = \left(A_i, B_i\left(c_{jm}, \mu_{jm}, U_{jm}\right), \pi_i \right) \quad (4.2)$$

4.2.2- Estimação dos Parâmetros $\hat{\lambda} = (\hat{A}, \hat{B}, \hat{\pi}_i)$

Para a estimação e reestimação usa-se o algoritmo "Segmental K-means", apresentado por Rabiner[12], Figura 4.3.

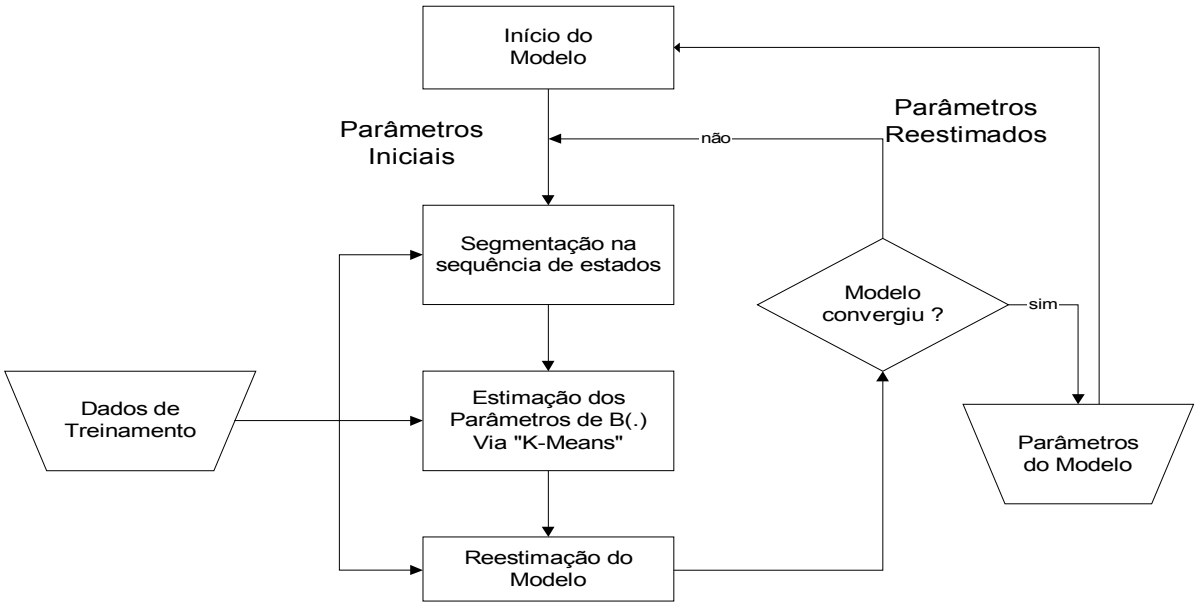


FIGURA 4.3: Procedimento "Segmental K-means" usado para estimar os valores dos parâmetros do HMM.

Utilizam-se os parâmetros iniciais, ou os parâmetros reestimados (como mostra a Figura 4.3), calculando-se os novos parâmetros do modelo, λ, de acordo com o algoritmo abaixo:

- $\lambda_{anterior}$
- Para cada repetição

  segmenta-se as sequências de Observações nos estados

  utliza-se o algoritmo de Viterbi para encontrar a melhor seqüência de estados:

  $$P(O, S / \lambda_{anterior})$$

  agrupam-se os vetores nos estados de acordo com a segmentação de Viterbi
- Fim
- Para cada estado

  utiliza-se o algoritmo "K-means" para agrupar os vetores em M grupos.

  obtém-se de cada grupo a média μ_{jk}, a covariância U_{jk} e o coeficiente de mistura c_{jk}

de acordo com as fórmulas 3.54, 3.55, 3.56 respectivamente.

calcula-se a matriz de probabilidades de transições de acordo com a segmentação de Viterbi, isto é, a melhor sequência de estados,

$$\hat{a}_{ij} \rightarrow \frac{\text{número de transições do estado Si para Sj}}{\text{número de transições do estado Si}} \tag{4.3}$$

- Fim

Conclui-se com a obtenção de todos os parâmetros do modelo estimado

$$\hat{\lambda} = (\hat{A},\ \hat{B}(\hat{c}_{jm}, \hat{\mu}_{jm}, \hat{U}_{jm}), \pi_i) \tag{4.4}$$

4.2.3- Reestimação dos Parâmetros $\overline{\lambda} = (\overline{A}, \overline{B}, \overline{\pi})$

Conforme o Item 3.6.2.3, Rabiner[12] utilizou, na reestimação do algoritmo, o método de Baum-Welch[59] (conhecido também como EM - expectation maximization). Este basea-se nos procedimentos "Forward" e "Backward".

(a) Definição de $\xi_t(i,j)$ ***e*** $\gamma_t(i,k)$ [12,13,58,59]

a.1 - Define-se $\xi_t(i,j)$ como sendo a probabilidade de estar no estado i no tempo t e no estado j no tempo t+1, dado o modelo e a sequência de observação.

$$\xi_t(i,j) = P(q_t = i, q_{t+1} = j / O, \lambda) \tag{4.5}$$

Sendo ilustrado na Figura 4.4:

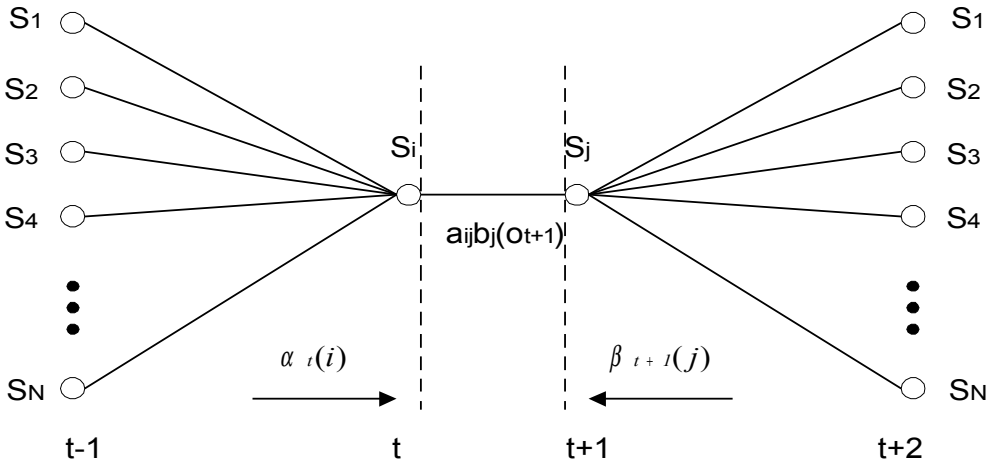


FIGURA 4.4: Ilustração da sequência de operações requerida para o cálculo do evento conjunto do sistema estar no estado i no tempo t e no estado j no tempo t+1.
Nesta figura ilustra-se todas as seqüências de estados possíveis entre o estado Si e o estado Sj.

Verifica-se, então, que $\xi_t(i,j)$ é o valor esperado do número de transições do estado Si para Sj.

Escrevendo $\xi_t(i,j)$ em função de α e β, tem-se

$$\xi_t(i,j) = \frac{\alpha_t(i)\cdot a_{ij}\cdot b_j(o_{t+1})\cdot \beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)\cdot a_{ij}\cdot b_j(o_{t+1})\cdot \beta_{t+1}(j)} \tag{4.6}$$

como $\alpha_t(i) = P(o_1 o_2 \ldots o_t, q_t / \lambda)$ e $\beta_t(i) = P(o_{t+1} o_{t+2} \ldots o_T / q_t = i, \lambda)$, então

$$P(O/\lambda) = \sum_{i=1}^{N}\alpha_t(i)\cdot\beta_t(i) \tag{4.7}$$

ou seja

$$P(O/\lambda) = \sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)\cdot a_{ij}\cdot b_j(o_{t+1})\cdot\beta_{t+1}(j) \tag{4.8}$$

a.2 - A outra variável a ser definida é a probabilidade a posteriori[57]

$$\gamma_t(i,k) = P(q_t = i, m = k / O, \lambda) \tag{4.9}$$

como sendo a probabilidade de estar no estado i e no grupo k no tempo t, dado a sequência de observação e o modelo. A Figura 4.5 apresenta as seqüencias possíveis dado o estado S_i.

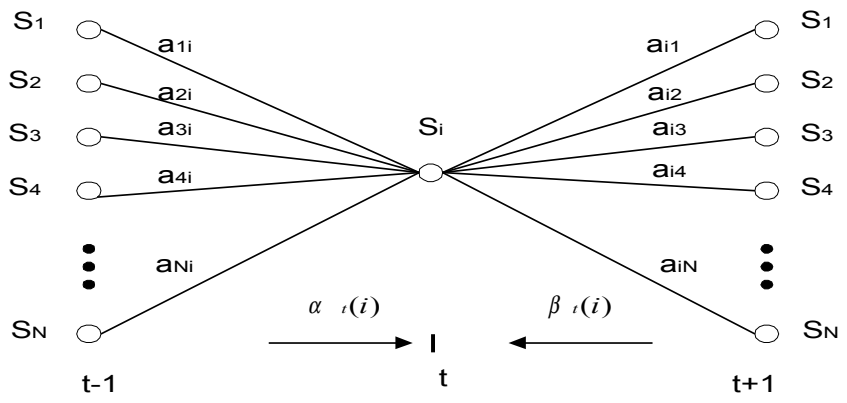


FIGURA 4.5: Sequência das operações requerida para a computação da variável $\gamma_t(i,k)$

Observa-se que esta figura apresenta todas as possíveis seqüências de estados, com o estado S_i incluído, observando uma determinada Gaussiana. Ou seja, $\gamma_t(i,k)$ é o cálculo do valor esperado do número de vezes que uma específica observação ocorre em uma determinada Gaussiana k e estado S_i.

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)} \right] \left[\frac{c_{jk}\mathcal{N}(o_t, \mu_{jk}, U_{jk})}{\sum_{m=1}^{M} c_{jm}\mathcal{N}(o_t, \mu_{jm}, U_{jm})} \right] \tag{4.10}$$

e para uma grupo

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)} \right] \tag{4.11}$$

Observa-se a Equação 4.12, que ao se somar todas as transições feitas do estado i para o estado j, tem-se a probabilidade de estar no estado i no tempo t.

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j) \tag{4.12}$$

Somando-se a variável $\gamma_t(i)$ para todo t, encontra-se o número esperado de transições a partir do estado i, como mostrado abaixo[12,58]:

$\sum_{t=1}^{T-1} \gamma_t(i) \rightarrow$ número esperado das transições que partem do estado i.

$\sum_{t=1}^{T-1} \xi_t(i,j) \rightarrow$ número esperado das transições que partem do estado i para j.

Assim, o cálculo das probabilidades das transições se torna:

$$\overline{a}_{ij} \rightarrow \frac{\text{número esperado de transições do estado Si para Sj}}{\text{número esperado de transições do estado Si}}$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \tag{4.13}$$

(b) Normalização[12,49,57,58]

Como os números usados estão na faixa entre 0 a 1 (valores probabilísticos) a multiplicação destes rapidamente leva o computador a um "underflow". Usa-se um fator de escala multiplicado a α e a β, independente do estado, e dependente somente de t, com o objetivo de manter $\alpha_t(i)$ e $\beta_t(i)$ num valor dentro da faixa de precisão do computador para $1 \leq t \leq T$. Sendo cancelado no final dos cálculos.

Calcula-se, então, o valor do coeficiente, fazendo o inverso do somatório de todas variáveis "forward's", referentes aos estados em um determinado tempo t:

b.1 - Encontra-se o coeficiente de normalização para as variáveis $\alpha_t(i)$

$$c_t = \frac{1}{\sum_{i=1}^{N} \alpha_t(i)} \tag{4.14}$$

normalizando as variáveis:

$$\tilde{\alpha}_t(i) = c_t \alpha_t(i) \qquad 1 \leq i \leq N \tag{4.15}$$

ou seja

$$\tilde{\alpha}_t(i) = \frac{\alpha_t(i)}{\sum_{j=1}^{N} \alpha_t(i)} \tag{4.16}$$

e utilizando a Equação 3.30, faz-se a recursão :

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \tilde{\alpha}_t(i) \cdot a_{ij} \right] \cdot b_j(o_{t+1}) \qquad 1 \leq j \leq N \tag{4.17}$$

b.2 - Coeficiente de escala para as variáveis $\beta_t(i)$

$$c_t = \frac{1}{\sum_{j=1}^{N} \beta_t(i)} \tag{4.18}$$

normalizando as variáveis:

$$\tilde{\beta}_t(i) = c_t \beta_t(i) \qquad 1 \le i \le N \tag{4.19}$$

ou

$$\tilde{\beta}_t(i) = \frac{\beta_t(i)}{\sum_{j=1}^{N} \beta_t(j)} \tag{4.20}$$

e utilizando a Equação 3.35 faz-se a recursão:

$$\beta_{t-1}(j) = \sum_{i=1}^{N} a_{ji} \cdot b_i(o_t) \cdot \tilde{\beta}_t(i) \qquad 1 \le j \le N \tag{4.21}$$

Assim, em termos das variáveis "forward" e "backward" normalizados, a fórmula para o cálculo de a_{ij} torna-se[12,13,58]:

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \tilde{\alpha}_t(i) \cdot a_{ij} \cdot b_j(o_{t+1}) \cdot \tilde{\beta}_{t+1}(j)}{\sum_{t=1}^{T-1} \sum_{j=1}^{N} \tilde{\alpha}_t(i) \cdot a_{ij} \cdot b_j(o_{t+1}) \cdot \tilde{\beta}_{t+1}(j)} \tag{4.22}$$

Na fórmula de a_{ij}, os termos do fator de escalamento se cancelam, como demonstra-se:

$$\tilde{\alpha}_t(i) = c_t \alpha_t(i) \tag{4.23}$$

$$\tilde{\beta}_{t+1}(i) = c_{t+1} \beta_{t+1}(i) \tag{4.24}$$

substituindo 4.23 e 4.24 em 4.22

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} c_t \alpha_t(i) \cdot a_{ij} \cdot b_j(o_{t+1}) \cdot c_{t+1} \beta_{t+1}(j)}{\sum_{t=1}^{T-1} \sum_{j=1}^{N} c_t \alpha_t(i) \cdot a_{ij} \cdot b_j(o_{t+1}) \cdot c_{t+1} \beta_{t+1}(j)} \tag{4.25}$$

fazendo

$$c_t \cdot c_{t+1} = C_T \tag{4.26}$$

tem-se

$$\bar{a}_{ij} = \frac{C_T \sum_{t=1}^{T-1} \alpha_t(i) \cdot a_{ij} \cdot b_j(o_{t+1}) \cdot \beta_{t+1}(j)}{C_T \sum_{t=1}^{T-1} \sum_{j=1}^{N} \alpha_t(i) \cdot a_{ij} \cdot b_j(o_{t+1}) \cdot \beta_{t+1}(j)} \tag{4.27}$$

cancelando C_T.

(c) Número de sequências para treinamento[12,13,14,57,58]

Até este ponto, o cálculo de a_{ij} refere-se apenas a uma sequência de observação. Adiante, verifica-se que um problema associado com treinamento do HMM, é que o número de sequências de observações usadas para o treinamento é finita. Então, pode existir uma baixa probabilidade da ocorrência de um certo evento (ex. baixa ocorrência da observação dentro de um estado) impossibilitando o algoritmo de obter boas estimativas dos parâmetros do modelo. Assim, para uma melhor obtenção de estimativas são utilizadas sequências múltiplas de observações (várias repetições da locução).Isto é,

$$O = [O^{(1)}, O^{(2)}, O^{(3)}, \ldots, O^{(R)}] \tag{4.28}$$

sendo R o número de repetições da locução usada para o treinamento. Supõe-se que as repetições são independentes entre si:

$$P(O / \lambda) = \prod_{o=1}^{R} P(O^{(O)} / \lambda) \tag{4.29}$$

(d) Parâmetros Reestimados[12,13,14,57,58]

d.1 - Com $\alpha^o_t(j)$, $\beta^o_t(j)$ representando os parâmetros da o-ésima locução, a **probabilidade de transição** finalmente se torna:

$$\overline{a}_{ij} = \frac{\sum_{o=1}^{R}\sum_{t=1}^{To-1} \tilde{\alpha}^o_t(i)\cdot a_{ij}\cdot b_j(o^o_{t+1})\cdot \tilde{\beta}^o_{t+1}(j)}{\sum_{o=1}^{R}\sum_{t=1}^{T-1}\sum_{j=1}^{N} \tilde{\alpha}^o_t(i)\cdot a_{ij}\cdot b_j(o^o_{t+1})\cdot \tilde{\beta}^o_{t+1}(j)} \tag{4.30}$$

e a sua *fórmula simplificada* igual a

$$\overline{a}_{ij} = \frac{\sum_{o=1}^{R}\sum_{t=1}^{To-1} \tilde{\alpha}^o_t(i)\cdot a_{ij}\cdot b_j(o^o_{t+1})\cdot \tilde{\beta}^o_{t+1}(j)}{\sum_{o=1}^{R}\sum_{t=1}^{T} \tilde{\alpha}^o_t(i)\ \tilde{\beta}^o_t(j)/c_t} \tag{4.31}$$

d.2 - A **probabilidade inicial** se mantém:

$$\pi_1 = 1 \text{ e } \pi_i = 0 \qquad com \qquad i \neq 1 \tag{4.32}$$

d.3 - **Probabilidade das Observações**

Na reestimação dos parâmetros média, covariância e coeficiente dos grupos para o cálculo da probabilidade de observação, utiliza-se o valor esperado da frequência do vetor de características (observação) em cada estado, estando este em um dos k grupos.

Para o obter esta probabilidade, calcula-se todas as seqüências de estados que inclua o estado S_j, como visto na Figura 4.5, e observando este vetor. A Equação 4.33 mostra esta probabilidade,

$$\gamma_t(j,k) = \frac{\left[\sum_{i=1}^{N} \alpha_{t-1}(i)\cdot a_{ij}\right]\cdot b_j(o_t)\cdot \beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)} \left[\frac{c_{jk}\,\mathcal{N}(o_t,\mu_{jk},U_{jk})}{\sum_{m=1}^{M} c_{jm}\,\mathcal{N}(o_t,\mu_{jm},U_{jm})}\right] \tag{4.33}$$

ou seja, todos os caminhos possíveis incluindo o estado S_j observando-se a observação o_t sobre todos os caminhos possíveis, em conjunto com a densidade de probabilidade da observação estar na Gaussiana k da mistura de Gaussianas. Sendo representada pela Figura 4.6:

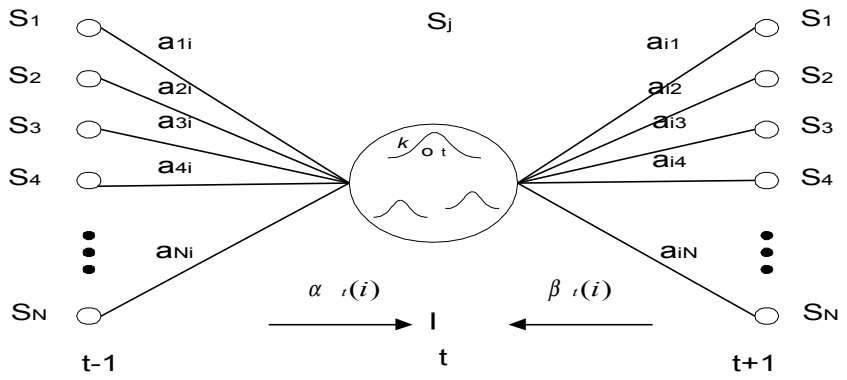


FIGURA 4.6: Observação o_t no estado j e no grupo k.

Sabendos-se que a variável "forward" é igual a:

$$\alpha_t(j) = \left[\sum_{i=1}^{N} \alpha_{t-1}(i) \cdot a_{ij} \right] \cdot b_j(O_t) \qquad 1 \le j \le N \quad 1 \le t \le T-1 \tag{4.34}$$

e utilizando a Equação 4.34 na Equação 4.33, obtém-se

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)} \right] \left[\frac{c_{jk} \mathcal{N}(o_t, \mu_{jk}, U_{jk})}{\sum_{m=1}^{M} c_{jm} \mathcal{N}(o_t, \mu_{jm}, U_{jm})} \right] \tag{4.35}$$

Com o valor de $\gamma_t(j,k)$ calculado, pode-se obter os valores para o coeficiente, a média e a covariância reestimadas dos grupos. Portanto, as fórmulas serão[12,13,58]:

1) Coeficiente do grupo K no estado j sobre o número esperado de estar no estado S_j

$$c_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)}{\sum_{t=1}^{T}\sum_{k=1}^{M} \gamma_t(j,k)} \tag{4.36}$$

2) Média do grupo k no estado j ponderado pela observação sobre o número esperado de estar no estado S_j e na Gaussiana k

$$\mu_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) \cdot o_t}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{4.37}$$

3) Covariância do grupo k no estado j ponderado pela covariância do vetor sobre o número esperado de permanecer no estado S_j e na grupo k.

$$U_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) \cdot [(o_t - \mu_{jk})(o_t - \mu_{jk})^T]}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{4.38}$$

Como observado no cálculo do a_{ij} (Equação 4.30), utilizam-se R locuções (múltiplas locuções) para o treinamento do HMM. Para obtenção das fórmulas para a reestimação, devem ser feitas algumas simplificações, mostradas nas equações seguintes onde $P^o(O/\lambda)$, $\alpha^o_t(j)$, $\beta^o_t(j)$ representam os parâmetros da o-ésima locução.

Da Equação 4.10 faz-se

$$D^o{}_{jk} = \left[\frac{c_{jk} \mathcal{N}(o^o{}_t, \mu_{jk}, U_{jk})}{\sum_{m=1}^{M} c_{jm} \mathcal{N}(o^o{}_t, \mu_{jm}, U_{jm})} \right] \tag{4.39}$$

então

$$\gamma_t(j,k) = \left[\frac{\alpha^o{}_t(j)\beta^o{}_t(j)}{\sum_{j=1}^{N} \alpha^o{}_t(j)\beta^o{}_t(j)} \right] D^o{}_{jk} \tag{4.40}$$

e sabe-se que

$$P(O/\lambda) = \sum_{i=1}^{N} \alpha_t(i) \cdot \beta_t(i) \tag{4.41}$$

portanto,

$$\bar{c}_{jk} = \frac{\sum_{o=1}^{R} \frac{1}{P^o(O/\lambda)} \left[\sum_{t=1}^{T} \alpha^o_t(j) \beta^o_t(j) \, D^o_{jk} \right]}{\sum_{o=1}^{R} \frac{1}{P^o(O/\lambda)} \left[\sum_{t=1}^{T} \alpha^o_t(j) \beta^o_t(j) \left[\sum_{k=1}^{M} D^o_{jk} \right] \right]} \tag{4.42}$$

neste cálculo observa-se que se pode eliminar o termo do somatório das probabilidades e que

$$\sum_{k=1}^{M} D_{jk} = 1. \tag{4.43}$$

Logo, as equações para o cálculo da **probabilidade de observação** serão:

a) **coeficiente reestimado**

- forma expandida

$$\bar{c}_{jk} = \frac{\sum_{o=1}^{R} \sum_{t=1}^{T-1} \sum_{i=1}^{N} \tilde{\alpha}^o_t(j) \cdot a_{ji} \cdot b_i(o^o_{t+1}) \cdot \tilde{\beta}^o_{t+1}(j) \, D^o_{jk}}{\sum_{o=1}^{R} \sum_{t=1}^{T-1} \sum_{i=1}^{N} \tilde{\alpha}^o_t(j) \cdot a_{ji} \cdot b_i(o^o_{t+1}) \cdot \tilde{\beta}^o_{t+1}(j)} \tag{4.44}$$

- forma simplificada

$$\bar{c}_{jk} = \frac{\sum_{o=1}^{R} \sum_{t=1}^{T} \tilde{\alpha}^o_t(j) \, \tilde{\beta}^o_t(j) / c_t \, D^o_{jk}}{\sum_{o=1}^{R} \sum_{t=1}^{T} \tilde{\alpha}^o_t(j) \, \tilde{\beta}^o_t(j) / c_t} \tag{4.45}$$

b) **média reestimada**

- forma expandida

$$\bar{\mu}_{jk} = \frac{\sum_{o=1}^{R} \sum_{t=1}^{T-1} \sum_{i=1}^{N} \tilde{\alpha}^o_t(j) \cdot a_{ji} \cdot b_i(o^o_{t+1}) \cdot \tilde{\beta}^o_{t+1}(j) \, D^o_{jk} \cdot o^o_t}{\sum_{o=1}^{R} \sum_{t=1}^{T-1} \sum_{i=1}^{N} \tilde{\alpha}^o_t(j) \cdot a_{ji} \cdot b_i(o^o_{t+1}) \cdot \tilde{\beta}^o_{t+1}(j) \, D^o_{jk}} \tag{4.46}$$

- forma simplificada

$$\overline{\mu}_{jk} = \frac{\sum_{o=1}^{R}\sum_{t=1}^{T} \tilde{\alpha}_t^o(j) \cdot \tilde{\beta}_t^o(j) / c_t \cdot D^o{}_{jk} \cdot o^o{}_t}{\sum_{o=1}^{R}\sum_{t=1}^{T} \tilde{\alpha}_t^o(j) \cdot \tilde{\beta}_t^o(j) / c_t \cdot D^o{}_{jk}} \tag{4.47}$$

c) **covariância reestimada**

- forma expandida

$$\overline{U}_{jk} = \frac{\sum_{o=1}^{R}\sum_{t=1}^{T-1}\sum_{i=1}^{N} \tilde{\alpha}_t^o(j) \cdot a_{ji} \cdot b_i(o_{t+1}^o) \cdot \tilde{\beta}_{t+1}^o(j) \cdot D^o{}_{jk} \cdot (o^o{}_t - \mu_{jk})\,(o^o{}_t - \mu_{jk})^T}{\sum_{o=1}^{R}\sum_{t=1}^{T-1}\sum_{i=1}^{N} \tilde{\alpha}_t^o(j) \cdot a_{ji} \cdot b_i(o_{t+1}^o) \cdot \tilde{\beta}_{t+1}^o(j)\, D^o{}_{jk}} \tag{4.48}$$

- forma simplificada

$$\overline{U}_{jk} = \frac{\sum_{o=1}^{R}\sum_{t=1}^{T} \tilde{\alpha}_t^o(j) \cdot \tilde{\beta}_t^o(j) / c_t \cdot D^o{}_{jk} \cdot (o^o{}_t - \mu_{jk})\,(o^o{}_t - \mu_{jk})^T}{\sum_{o=1}^{R}\sum_{t=1}^{T} \tilde{\alpha}_t^o(j) \cdot \tilde{\beta}_t^o(j) / c_t \cdot D^o{}_{jk}} \tag{4.49}$$

Assim sendo, obtém-se desta maneira para cada estado e grupo os parâmetros do modelo reestimado

$$\overline{\lambda} = (\overline{A}, \overline{B}(\overline{c}_{jm}, \overline{\mu}_{jm}, \overline{U}_{jm}), \overline{\pi}_i) \tag{4.50}$$

4.3- RECONHECIMENTO

O diagrama da fase reconhecimento, Figura 4.5, ilustra a simplicidade do algoritmo. Com a utilização, somente do algoritmo de Viterbi[12,13,14] e dos parâmetros do modelo treinado λ, obtém-se a verossimilhança $P(O/\lambda)$ para um dado vetor de entrada (teste).

O procedimento para reconhecimento de uma locução de teste com vários HMM foi exposto no capítulo anterior Figura 3.18, no Item 3.6.3.1.

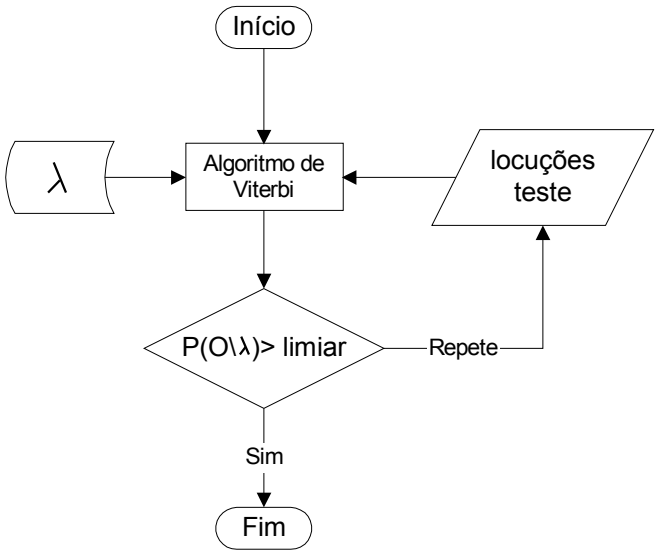


FIGURA 4.5: Procedimento de Reconhecimento da locução teste utilizando o algoritmo de Viterbi

O algoritmo de Viterbi[12] (Item 3.6.2.2) utiliza a multiplicação entre seus termos, os quais são valores probabilísticos. Por conseguinte, obtém-se resultados com magnitudes extremamente pequenas podendo ocorrer "underflow". Uma solução[12,14] é o uso da implementação alternativa de Viterbi. Aplica-se o logaritmo nos parâmetros do modelo, evitando desta forma a necessidade de qualquer multiplicação.

O algoritmo se torna:

- pré-processamento

$$\tilde{\pi}_i = \log(\pi_i) \qquad 1 \le i \le N \tag{4.51}$$

$$\tilde{b}_i(o_t) = \log[b_i(o_t)] \qquad 1 \le i \le N \quad 1 \le t \le T \tag{4.52}$$

$$\tilde{a}_{ij} = \log(a_{ij}) \qquad 1 \le i \le N \tag{4.53}$$

- início

$$\tilde{\delta}_1(i) = \log(\delta_1(i)) = \pi_i + b_i(o_1) \qquad 1 \le i \le N \tag{4.54}$$

$$\psi_1(i) = 0 \qquad 1 \le i \le N \tag{4.55}$$

- recursão

$$\tilde{\delta}_t(i) = \log(\delta_t(i)) = \max_{1 \le i \le N}[\tilde{\delta}_{t-1}(i) + \tilde{a}_{ij}] + \tilde{b}_j(o_t) \qquad 1 \le j \le N \quad 2 \le t \le T \tag{4.56}$$

$$\psi_1(j) = \arg\max_{1 \le i \le N}[\tilde{\delta}_{t-1}(i)\,\tilde{a}_{ij}] \qquad 1 \le j \le N \quad 2 \le t \le T \tag{4.57}$$

- finalização

$$\tilde{P} = \max_{1 \le i \le N}[\tilde{\delta}_T(i)] \tag{4.58}$$

$$q_T = \arg\max_{1 \le i \le N}[\tilde{\delta}_T(i)] \tag{4.59}$$

- retorno para obter o melhor caminho

$$q_t = \psi_{t+1}(q_{t+1}) \qquad t = T-1, T-2, \ldots, 1 \tag{4.60}$$

A medida resultante calculada para o modelo λ, de uma sequência de observações, O, será igual a

$$P(O, Q / \lambda) \tag{4.61}$$

onde Q é a sequência dos estados mais provável.

A medida não é exatamente uma probabilidade, mas é evidente que ela é uma magnitude de verossimilhança[14], com a qual se permite uma comparação relativa entre os modelos. Note-se que, tomando o logaritmo de $b_j(o_t)$ será equivalente a produzir uma máxima verossimilhança (ou, quase equivalente, a distância de Mahalanobis) da distância da

observação o_t (locução de teste) até a média das observações treinadas, assumindo ser uma densidade Gaussiana[14].

CAPÍTULO 5

IMPLEMENTAÇÃO DO SISTEMA

5.1 - INTRODUÇÃO

Este capítulo tem por objetivo descrever os principais procedimentos utilizados na obtenção das elocuções, para treinamento e teste dos modelos dos locutores, os algoritmos utilizados para a extração das características e a implementação do sistema de reconhecimento automático de locutor (**RAL**) dependente do texto utilizando HMM contínuo — apresentado nos Capítulos 3 e 4.

5.2 - SISTEMA PROPOSTO

O sistema **RAL** proposto foi desenvolvido no Laboratório de Processamento de Sinais do IME utilizando-se um microcomputador 486 DX4 100 MHz, equipado com uma placa Sound Blaster 16 (Sound Blaster é marca registrada da "Creative Labs, Inc.").

Este sistema utiliza os Modelos de Markov Escondidos Contínuos (**CDHMM — Continuous Densities Hidden Markov Models**) na fase de treinamento e teste, os quais permitem uma modelagem estatística do sinal de voz[12,13] na representação dos locutores. O sistema RAL é um processo de reconhecer um locutor pretenso, como verdadeiro ou falso, através das informações obtidas do sinal de sua voz. Tem duas tarefas distintas: *verificação* e *identificação*.

Na tarefa de verificação, o limiar de aceitação θ de cada HMM treinado foi obtido pelo método Bayes (Figura 3.16). O locutor era considerado verdadeiro se a verossimilhança,

obtida pela aplicação do algoritmo de Viterbi[12], da seqüência das observações teste ficasse acima do limiar de aceitação do HMM correspondente àquele locutor.

Na identificação, após a aplicação da elocução teste em todos HMM, o locutor era associado ao modelo que produzisse a maior verossimilhança. Na identificação com rejeição, após a seleção do modelo com maior verossimilhança, fez-se a comparação desta com o limiar de aceitação associado. Se fosse menor que este limiar, o locutor era considerado falso.

O trabalho em laboratório dividiu-se em 3 partes principais:

1 - Aquisição das elocuções;

- gravação

2 - Processamento da Base de Dados

- pré-processamento dos sinais a serem analisados;
- extração das características do sinal de voz;

3 -Projeto do Sistema de Decisão

- atribuição dos parâmetros fixos dos HMMs;
- treinamento e teste dos HMMs;

O procedimento utilizado é descrito no diagrama em bloco apresentado na Figura 5.1.

5.3 - AQUISIÇÃO DOS DADOS

Utilizou-se a placa Sound Blaster 16 da Creative, Inc. para a gravação das elocuções. As principais características da placa Sound Blaster são: 1) realiza amostragens (conversão A/D) de sinais de áudio em mono ou estéreo com taxas de amostragem de 11, 22 ou 44 Khz; 2) grava e lê arquivos em disco com extensão *WAV* (padrão de arquivos de áudio amostrado padronizado internacionalmente e adotado pela Microsoft) e 3) obtém amostras do sinal em 8

ou 16 bits, ou seja, o sinal digital obtido pode conter 256 ou 65535 níveis de quantização, possibilitando a análise de sinais de áudio com faixa dinâmica de 21 e 45 dB respectivamente.

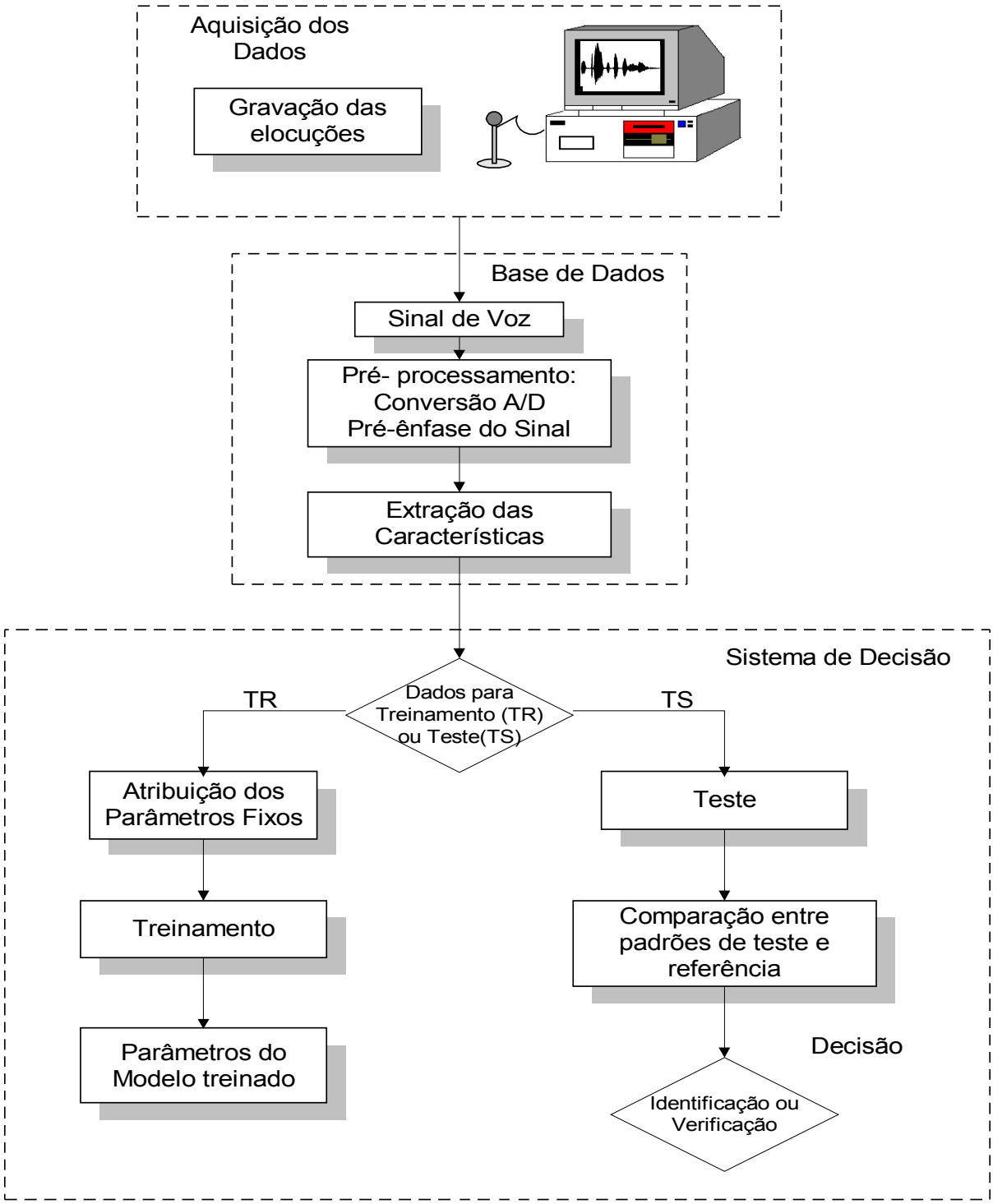


FIGURA 5.1: Diagrama em bloco mostrando as etapas desenvolvidas no sistema proposto

Utilizou-se a taxa de amostragem igual a 11025 Hz (valor mínimo fornecido pela placa Sound Blaster). Como a maior freqüência da banda passante da voz é igual a 5 kHz o sinal não sofre "aliasing".

Após a realização das gravações, criam-se arquivos com extensão *.WAV*. Estes arquivos são separados em dois conjuntos: masculino e feminino, denominados L*ax*E*y*R*z*.WAV, onde *a* pode ser *M* conjunto masculino ou *F* conjunto feminino, *x* é o número do locutor, *y* é o número da elocução e *z* é o número da repetição desta frase. Como exemplo, o arquivo LM1e2r3.WAV representa a repetição número 3 da elocução número 2 feita pelo locutor masculino 1.

As sessões de gravações foram realizadas em uma sala sem isolamento acústico, utilizando um PC com placa Sound Blaster 16. Em cada sessão gravou-se duas repetições de cada frase em horários e dias diferentes.

5.4 - BASE DE DADOS

Para treinamento e teste do sistema foram utilizados dois conjuntos de locutores: o masculino (**CM**) com 5 locutores para treinamento e teste e 3 somente para teste e o feminino (**CF**) com 4 locutores para treinamento e teste e 3 somente para teste.

Foram utilizadas as frases "O prazo tá terminando" (**fr1**) e "Amanhã ligo de novo" (**fr2**) considerada as mais adequadas conforme as análises fonéticas/fonológicas do Departamento de Lingüística da USP em um conjunto de frases para Reconhecimento de Locutores[15].

Realizaram-se para cada locutor 70 gravações de cada frase (50 para treinamento e 20 para teste). Foram realizadas também, 20 gravações de outros locutores que não participaram

do treinamento, totalizando assim, 360 locuções teste do conjunto CM e 290 do conjunto CF para cada frase. A cada seção de gravação realizaram-se apenas 2 gravações das elocuções fr1 e fr2, com a intenção de capturar a variabilidade intra-locutor.

Os dados foram pré-processados com filtro de pré-ênfase

$$H(z)=1- 0.952z^{-1} \quad (5.1)$$

e utilizaram-se janelas Hamming de 20 ms com superposição de 50%. O vetor de características, de acordo com o Item 2.3.2, é composto de 12 coeficientes mel-cepestro, 12 coeficientes delta-mel-cepestro, log-energia e delta log-energia[13].

5.5 - EXTRAÇÃO DAS CARACTERÍSTICAS DO SINAL DE VOZ

Após a obtenção do sinal de v Ponto Inicial sto no Itei Ponto Final a extração das características mais relevantes do sinal de voz do locutor. O sistema é, então, treinado para torná-lo discriminante em relação às locuções de outros locutores.

Estas característcas foram escolhidas com base em trabalhos publicados[12,13,29-33,60] recentemente. A seguir, são apresentados os procedimentos e as considerações utilizadas nas implementações.

5.5.1 - Consideração Sobre o Nível de Ruído

As gravações foram realizadas no Laboratório de Processamentos de Sinais em uma sala sem isolamento acústico. As Figuras 5.2 e 5.3 apresentam exemplos de elocuções, no domínio do tempo das frases fr1 "O prazo tá terminando", e fr2 "Amanhã ligo de novo", respectivamente, ambas pronunciadas por um locutor do sexo masculino. Fez-se o cálculo da

relação sinal ruído (RSR), obtendo-se para fr1 uma RSR máxima igual a 32 dB e a média igual a 21.99 dB. Para a fr2 a RSR máxima foi de 29.43 dB e a média de 22.22 dB.

Como todas as gravações foram realizadas no mesmo equipamento e no mesmo ambiente, o nível de ruído manteve-se aproximadamente constante nos valores citados. Deve-se ressaltar que a baixa relação sinal ruído não invalida o processo de identificação tendo em vista que o sistema foi proposto, treinado e testado com locuções gravadas em situações idênticas.

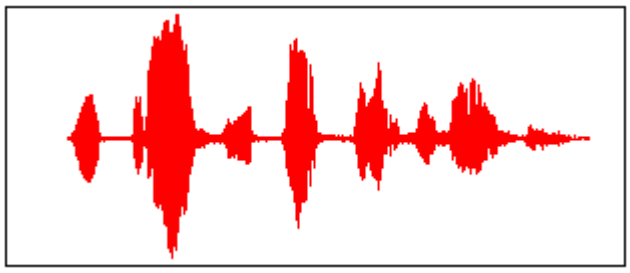


FIGURA 5.2: Exemplo da elocução fr1 ("O prazo tá terminando") dita por um locutor masculino

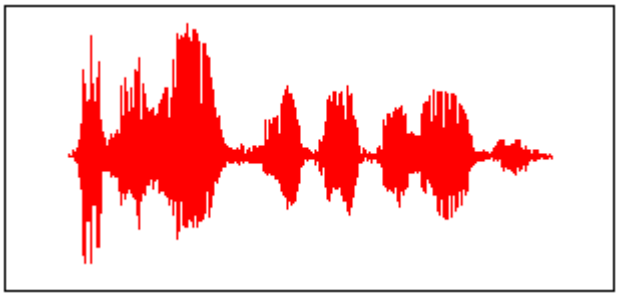


FIGURA 5.3: Exemplo da elocução fr2 ("Amanhã ligo de novo") dita por um locutor masculino

5.5.2 - Consideração Sobre a Determinação dos Pontos Extremos

Para o reconhecimento de locutores se faz a detecção dos pontos extremos de cada elocução com o objetivo de separar a região falada da região ruidosa, isto é, a localização

do início e do fim de cada locução. Desse modo evita-se a extração de características nas regiões de silêncios anteriores e posteriores às elocuções a serem analisadas.

Utilizou-se o algoritmo de Rabiner e Sambur para a determinação dos pontos extremos que exige uma relação sinal ruído mímina, pelo menos, de 30 dB. Tal algoritmo se baseia em duas medidas do sinal de voz: a energia e a taxa de cruzamento do zero obtidas em janelas de 10 ms de duração do sinal. Um intervalo de 100 ms no início da elocução (10 janelas) é utilizada para efetuar uma estatística do ruído de fundo. Este algoritmo caracteriza-se por ser adaptativo às condições do ambiente acústico. Como as elocuções obtiveram, em geral, taxas médias inferiores a este valor, da ordem de 22 dB (Item 5.5.1), fizeram-se modificações nos limiares superior e inferior de energia dos sinais utilizados, obtendo-se resultados satisfatórios porém susceptíveis a erros.

Portanto, os limiares foram heurísticos e, mesmo assim, não se obteve 100% de frases extraídas corretamente, como mostra a Figura 5.4.

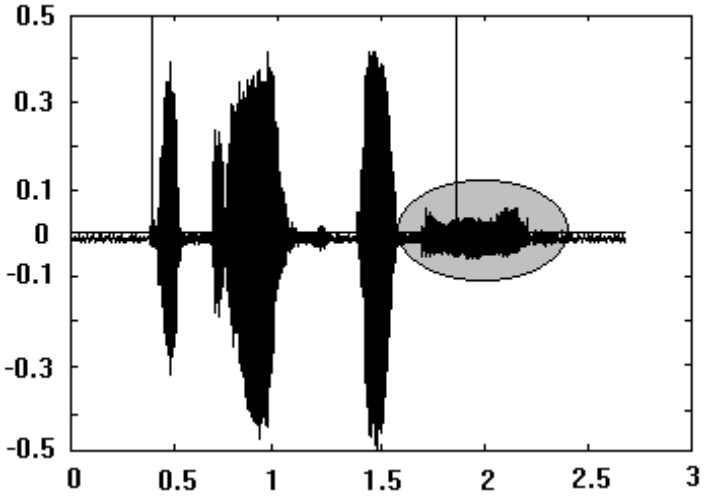


FIGURA 5.4: Elocução em que os pontos extremos foram extraídos incorretamente.

Por causa disso, após a extração dos pontos extremos de todas as elocuções de cada locutor fez-se uma verificação auditiva e visual das elocuções.

5.5.3- Coeficientes Mel-Cepestro

As formas mais comum de se obter os coeficientes mel-cepestro são: 1) utilizando a transformada de Fourier (mostrado no Item 2.3.3(a)); 2) por meio dos parâmetros LPC (mostrados no Item 2.3.4(c)). Pelos motivos apresentados no Item 2.3.3(c) optou-se por calcular os parâmetros do mel cepestro utilizando a técnica LPC (obtido pelo método da autocorrelação).

Para se obter os coeficientes mel-cepestro do trato vocal $C_\theta(n)$ deve-se estimar a densidade espectral do trato vocal, para cada sinal janelado, ponderada pela escala da freqüência mel e, em seguida, obter o valor da log energia, no domínio da freqüência, dentro de uma banda crítica em torno da freqüência mel central em cada um dos 20 filtros[13]. Conclui-se com o cálculo dos 12 coeficientes mel-cepestro aplicando-se a Equação 5.8 para cada filtro.

Supondo-se que o sinal da voz seja estacionário em intervalos de tempo curto, ou seja, suas características se mantêm constantes e sabendo-se que o sinal de voz é a convolução do sinal da excitação (um trem de impulsos quase periódicos) com a resposta ao impulso do filtro do trato vocal, sendo a excitação suposta ser um processo estocástico gaussiano branco, pode-se considerar que as propriedades do trato vocal permanecem constante e consequentemente representá-lo como um filtro digital, invariante no tempo, com as amostras do sinal de voz admitidas como saídas do filtro. Representa-se o filtro digital, em cada intervalo de tempo da janela, pelo conjunto dos coeficientes da resposta ao impulso de duração finita (FIR).

Um modelo de filtro bastante usado, para o sinal de voz é o modelo autoregressivo (AR), apresentado na Figura 5.5. Supondo-se que o processo aleatório que esta sendo analisado y(n), saída do filtro, possua uma entrada espectralmente invariante[23,59], estima-se então a densidade de potência espectral da saída y(n).

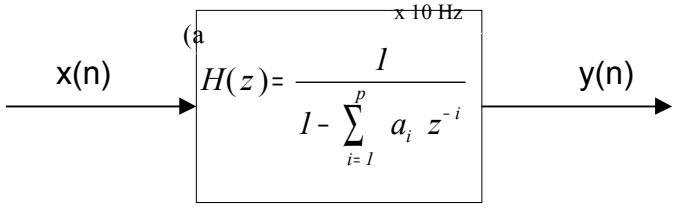


FIGURA 5.5: O processo y(n) pode ser modelado como a saída de um filtro FIR com função de transferência do tipo "só-de-pólos" excitado por processo de ruído branco (amostras de um processo estocástico gaussiano estacionário e ergódico)

Analizando-se no círculo unitário do plano Z, ou seja, fazendo-se $z=e^{-j\lambda}$, para uma entrada impulsional com distribuição Gaussiana, a densidade espectral da saída de um sistema linear invariante discreto (LID) é dada por

$$f_y = \left|H(e^{-j\lambda})\right|^2 \cdot f_x \tag{5.2}$$

sendo $\lambda=w\cdot\Delta t$, e T o período de amostragem. Portanto a densidade espectral de potência de saída, $f(\lambda)$, do sistema LID da Figura 5.5 será

$$f(\lambda) = P_x(n) \cdot \frac{1}{\left|1 - \sum_{i=1}^{p} a_i z^{-i}\right|^2} \tag{5.3}$$

onde $P_x(n)$ é a densidade espectral de potência da entrada e p o número de pólos necessários à aproximação da densidade espectral da voz. Normalmente se utilizam 14 pólos para modelagem da região entre 0 a 5KHz[23].

Supondo-se que a entrada x(n) é um processo estocástico gaussiano estacionário, então

$$P_x(n) = \sigma^{2}_{x(n)} \tag{5.4}$$

com $\sigma^{2}_{x(n)}$ sendo a variância do sinal de entrada e a densidade espectral f(λ) será expressa da seguinte maneira:

$$f(\lambda) = \frac{1}{2\pi} \frac{\sigma^{2}_{x(n)}}{A_o + 2\sum_{i=1}^{p} A_i \cos i\lambda} \tag{5.5}$$

com a variância igual a

$$\sigma^{2}_{x} = A_o R(O) + 2 \cdot \sum_{i=1}^{p} A_i R(i) \tag{5.6}$$

sendo R(•) igual à autocorrelação de y(n)

$$A_i = \sum_{j=1}^{p-i} a_j \; a_{j+i} \; . \tag{5.7}$$

Neste ponto, obtém-se a densidade espectral f(λ) do sinal da saída do sistema LID. Para que o eixo de freqüências do espectro de y [n] corresponda à escala mel, ou seja, linear até 1000 Hz e logarítmica acima (ver Figura 2.7), faz-se a filtragem, ponto a ponto, do espectro de y(n) utilizando os filtros triangulares da escala mel obtidos a partir das Equações 2.7 a 2.11.

Após a filtragem, calcula-se a energia de tempo curto do sinal de voz, no domínio da freqüência para cada filtro. Algumas pesquisas tem sugerido o uso do log energia dentro da banda crítica em torno da freqüência mel[13]. Através da Equação 5.8 calcula-se o log energia

$$E(i) = \sum_{k=0}^{N} \log|y(n)|^2 \cdot H_i(k \cdot \frac{2\pi}{N'}) \tag{5.8}$$

onde $H_i(\bullet)$ é o filtro triangular para o qual se está calculando a energia dos k pontos da DFT e N' é o número de pontos usado no cálculo da DFT. Portanto, o log energia total em cada filtro k_i fica igual a

$$E(k) = \begin{cases} E(i), & k = k_i \\ 0, & outros\, k \in [0, N' - 1] \end{cases}$$

A Tabela 5.1 mostra as freqüências de corte e a largura de banda crítica, de acordo com a Equação 2.6. Este cálculo pode ser visto como a filtragem da seqüência y(n) por um filtro passa baixa com freqüência de corte dada por: $Fc=kFs / N$, onde Fs é a freqüência de amostragem da seqüência y(n), N é o número de amostras do sinal contido em cada janela e k é o número do filtro. A Figura 5.6(a) mostra a densidade espectral obtida para a primeira janela de 20 ms da elocução "O prazo tá terminando", calculada utilizando N igual a 1024 pontos para o cálculo da densidade espectral.

TABELA 5.1: Valores das freqüências de corte dos 20 filtros ($F_{c,i}$) e seu respectivo ponto da DFT.

Faixa Linear			Faixa Logarítmica		
i-ésimo Filtro	$F_{c,i}$	k_i	i-ésimo Filtro	$F_{c,i}$	k_i
1	100	10	11	1148	107
2	200	19	12	1318	123
3	300	28	13	1514	141
4	400	38	14	1737	162
5	500	47	15	1995	186
6	600	56	16	2292	213
7	700	66	17	2692	245
8	800	75	18	3020	281
9	900	84	19	3467	323
10	1000	93	20	4000	372

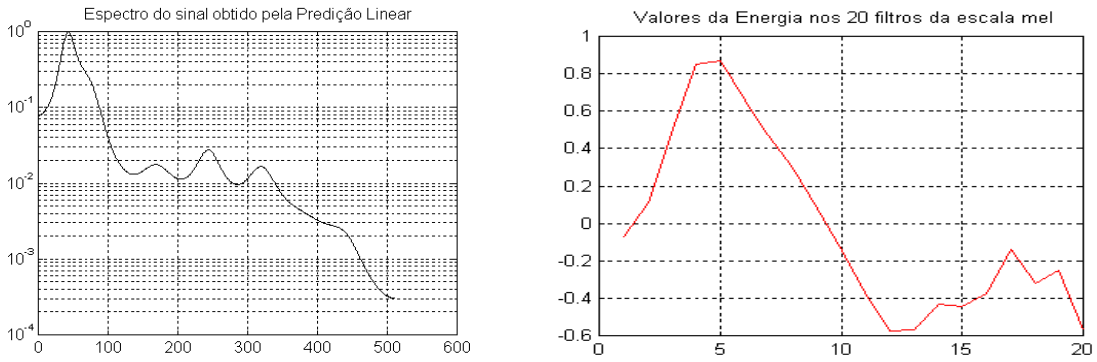


FIGURA 5.6: (a) Gráfico da Densidade Espectral, y(n), obtida a partir dos 14 coeficientes LPC do filtro digital, representando o trato vocal, para a primeira janela de 20 ms do sinal de voz; (b) Evolução da Energia dos 20 filtros da escala mel, obtido a partir do espectro do sinal y(n).

De posse da evolução da energia do sinal (do trato vocal) obtida a partir dos filtros da escala mel faz-se o cálculo dos coeficientes mel-cepestro. Utiliza-se a Equação 5.9 para o cálculo dos coeficientes derivados do LPC (MCLPC):

$$MCLPC_i = \sum_{k=1}^{20} E_k \cos[i(k - 0.5)\pi / 20], \qquad i = 1,2,\ldots,M \tag{5.9}$$

onde i é o número de coeficientes cepestro e E_k, k = 1,2,...,20, representa as saídas da log energia do k-ésimo filtro. A Figura 5.7 apresenta o gráfico da evolução dos 12 primeiros coeficientes mel-cepestro para o quadro da Figura 5.6.

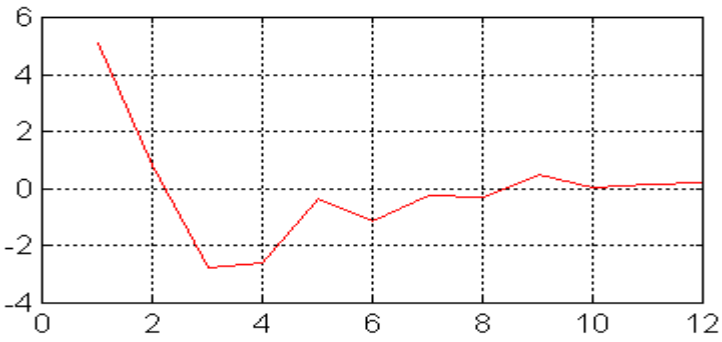


FIGURA 5.7: Evolução dos *12 parâmetros mel cepestro derivados do LPC* calculados utilizando o logaritmo da densidade espectral.

A Figura 5.8 apresenta o algoritmo para obtenção dos coeficientes mel cepestro a partir dos coeficientes do LPC.

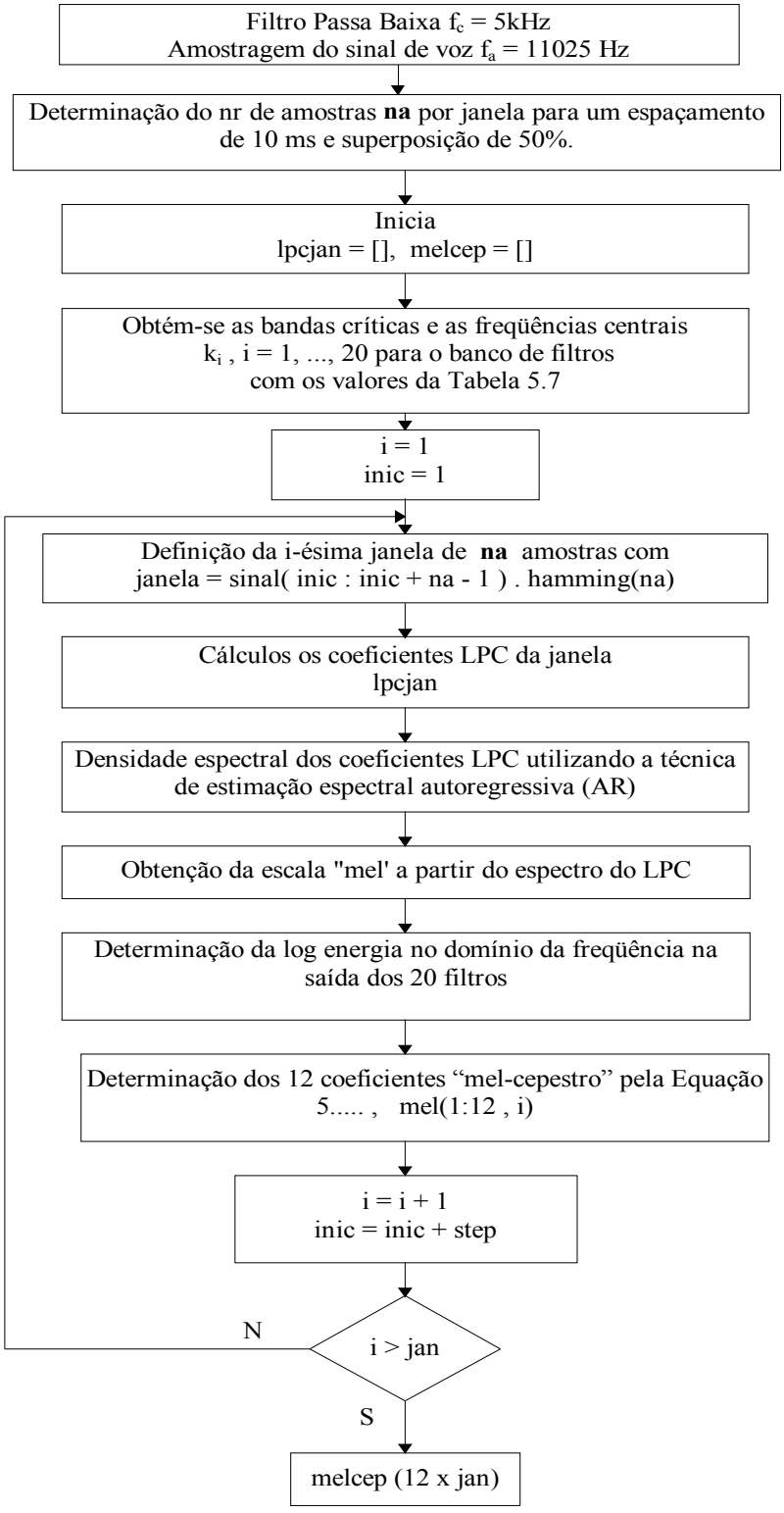


FIGURA 5.8: Algoritmo para a obtenção dos coeficientes mel-cepestro.

5.5.4 - Coeficiente Delta-Mel-Cepestro (CDMC)

Para representar as variações dinâmicas do espectro da voz, por exemplo as variações encontradas nos fonemas consonantais, e aumentar a robustez do sistema utilizam-se os coeficientes delta-mel-cepestros[12,30,60]. Estes parâmetros são obtidos por meio da diferença entre os coeficientes mel-cepestro de δ janelas a frente com os coeficientes mel-cepestro de δ janelas anteriores.

Admitindo-se que os coeficientes mel-cepestro (MCLPC) de uma dada janela estão dispostos como um vetor, a Equação 2.29 torna-se:

$$Delta_i(n) = MCLPC_i(\delta+n) - MCLPC_i(\delta-n) \qquad 1 \leq i \leq M \qquad (5.10)$$

onde n é a n-ésima janela e M é igual ao número dos coeficientes cepestro. O valor atribuído a δ neste trabalho foi 2.

A Figura 5.9 apresenta a evolução dos 12 coeficientes delta-mel-cepestro obtido para a primeira janela da vogal da elocução fr1.

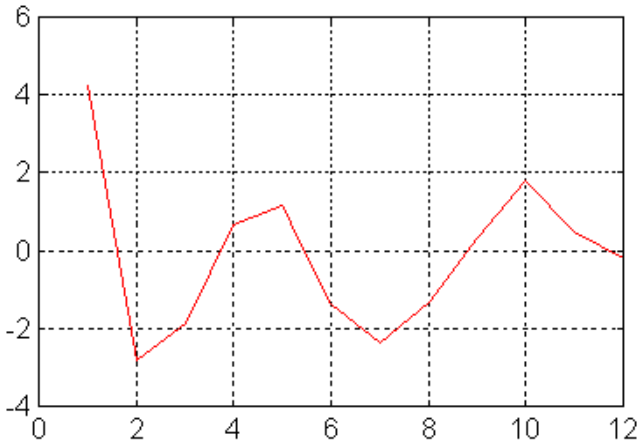


FIGURA 5.9: Evolução dos coeficientes delta-mel-cepestro obtidos através da diferença entre coeficientes mel-cepestro de janelas deslocadas por δ igual a 2.

5.5.5 - Log-Energia

Conforme apresentado no Item 5.5.3, para o cálculo dos coeficientes mel-cepestro, utilizou-se a energia de tempo curto do sinal de voz[30], calculada no domínio da frequência, para cada filtro. Aplicando-se o somatório na Equação 5.8 para todos os filtros triangulares, encontra-se a energia total do segmento de 20 ms do sinal de voz (sinal janelado). Utilizou-se o logaritmo dos parâmetros da energia para se obter uma compressão da sua faixa dinâmica, a qual é expressiva[12,13].

5.5.6 - Delta Log-Energia

Para a extração dos coeficientes delta-log-energia[12,30,60] aplica-se os mesmos procedimentos do Item 5.5.5 e utiliza-se a Equação 5.10, tornando-se:

$$\text{Delta\_log}(n) = E(\delta+n)-E(\delta-n) \qquad (5.11)$$

onde δ é igual a 2.

5.6 - CONSIDERAÇÕES SOBRE A IMPLEMENTAÇÃO DOS HMM's.

A estrutura do modelo a ser usada é a esquerda-direita, uma particularidade do modelo Bakis[12] e o procedimento de treinamento "Segmental-Kmeans"[12] (Capítulo 4).

Os parâmetros variáveis iniciais são selecionados conforme descrito no Capítulo 4.

1) probabilidade inicial (π): será fixa durante todo o treinamento. De acordo com o modelo Bakis,

$$\pi_1 = 1 \quad e \quad \pi_n = 0 \quad para \; n \neq 1 .$$

2) valores da matriz inicial das probabilidades das transições (**A**): aleatória, respeitando-se as restrições do modelo Bakis.

3) valores da matriz inicial da densidade de probabilidade de saída das observações (**B**): o treinamento do modelo mostrou-se mais sensível aos valores atribuídos inicialmente a matriz B. Assim, visando obter um estimativa dos valores iniciais que satisfaça a evolução temporal da seqüência de observações nos estados segmenta-se as seqüências de observações nos N estados[12]. Em cada estado utiliza-se o algoritmo "kmeans" modificado para separar as observações em M grupos e obtém-se valores das probabilidades de observações por meio da mistura de M Gaussianas. Testou-se dois algoritmos distintos para a separação das observações. O primeiro, executa a divisão das R seqüências de observações por N estados, sendo o resto da divisão somado ao último estado. A Figura 5.10 exemplifica esta divisão.

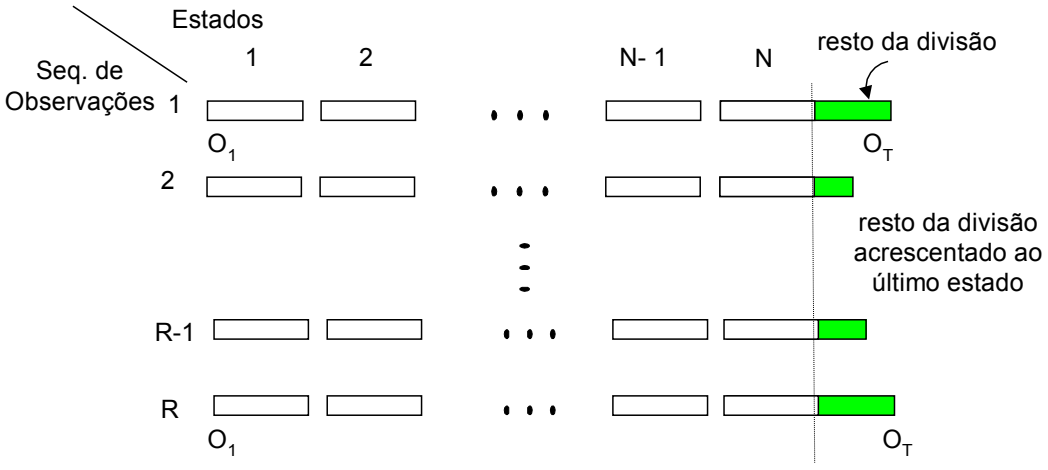


FIGURA 5.10: Apresenta as R seqüências de observações divididas por N estados e o resto de cada divisão somado ao último estado.

O segundo algoritmo, proposto e testado neste trabalho, faz a divisão da primeira seqüência de observações pelo número N de estados e o resto da divisão é somado ao primeiro estado. Para a segunda seqüência o resto é somado ao segundo estado e assim por diante para

todas as elocuções. Após a N-ésima seqüência, o resto da seqüência número N+1 obtido pela divisão é acrescentado ao primeiro estado, retornando-se ao processo iterativo anterior. Desse modo, o número de observações dentro dos estados tende a ser mais homogêneo. A Figura 5.11 mostra esta divisão para N estados e R repetições ($1 \leq r \leq R$).

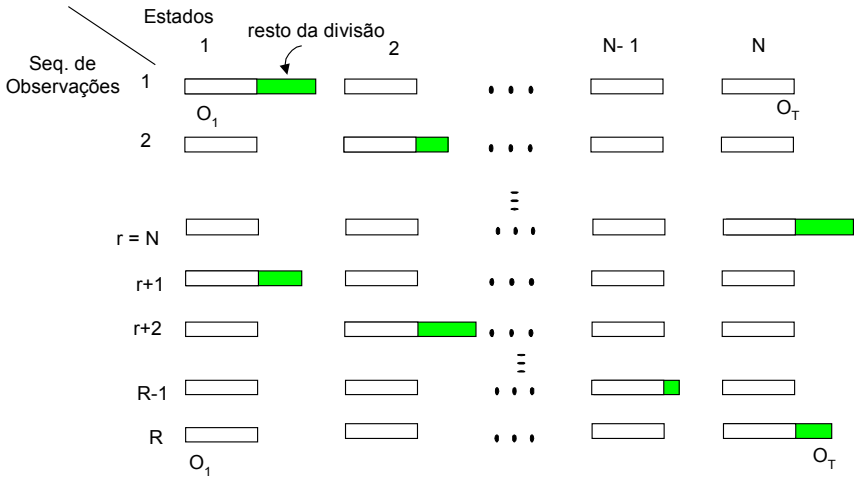


FIGURA 5.11: Apresenta a divisão das R seqüências de observações por N estados e o resto acrescentado seqüencialmente a cada estado até r igual a N. A partir de r +1 acrescenta-se o resto ao primeiro estado e assim sucessivamente até R seqüências.

A escolha dos valores dos parâmetros fixos iniciais, N (número de estados) e M (número de grupos), para o primeiro experimento, foram efetuadas do seguinte modo[12]:

1) N: *aproximadamente* 3 estados por fonema. Desse modo, a frase 1 foi treinada inicialmente com 36 estados e a frase 2 com 42 estados.

2) M: 5 grupos para cada estado.

Os parâmetros escolhidos para os demais testes foram empíricos. A Tabela 5.2 relaciona os parâmetros utilizados nos treinamentos e testes, onde LMfr1 é o locutor masculino pronunciando uma repetição da frase 1 e g5s12 significa uma mistura de 5 Gaussianas e uma estrutura com 12 estados.

TABELA 5.2: Parâmetros fixos utilizados nos treinamentos e testes.

CM		CF	
LMfr1	LMfr2	LFfr1	LFfr2
g5s12	g5s12	g5s12	g3s12
g5s36	g5s36	g5s36	g3s24
g5s60	g5s42	g5s42	g3s36
g10s12	g5s60	g5s60	g5s12
g10s36	g10s36	g10s36	g5s36
	g10s42	g10s60	g5s42
			g5s60
			g10s36
			g10s60

5.6.1 - Considerações Sobre o Treinamento do HMM

Por causa do número limitado de sequências de observações para treinar o modelo, pode ocorrer, durante o treinamento, uma probabilidade de observação nula para determinado estado. Causado por as observações que possuem uma pequena probabilidade de ocorrer neste estado em relação ao estado seguinte da cadeia de Markov — portanto, obtendo média e covariância nulas. Entretanto, o locutor treinado pode pronunciar uma elocução que possua esta probabilidade de observação não nula. Assim, admite-se como valor mínimo de 0.0001 para os parâmetros associados à mistura de Gaussianas[12].

Assume-se a convergência do modelo no treinamento quando a diferença entre as verossimilhanças anterior e atual for menor do que 1%. Caso a verossimilhança λ^* obtida na reestimação seja menor que a estimada λ acrescenta-se, então, um valor empírico de 10% sobre os valores dos parâmetros λ, da seguinte maneira:

```
A=Aaux*.01+Aaux;                % (matriz de transições)
```

```
media=mediaux*.01+mediaux;      % (vetor média )
covar=covaraux*.01+covaraux;    % (matriz covariância)
coef=coefaux*.01+coefaux;       % (vetor dos coeficientes de mistura)
```

onde A, media, covar e coef são os parâmetros do modelo atualizados com 10% e Aaux, mediaux, covaraux e coefaux são os parâmetros covariância e coeficiente reestimados. E prossegue-se com nova reestimação.

5.6.2 - Teste do Sistema

Na fase de reconhecimento, aplica-se o algoritmo de Viterbi numa elocução teste obtendo-se um valor de verossimilhança. esta verossimilhança pode ser interpretada como uma distância da elocução ao modelo treinado. Verifica-se que este valor é influenciado pelo número de observações da elocução —número de janelas. Portanto, uma elocução falsa *pode* ser admitida como *verdadeira* se seu comprimento for suficientemente grande tal que o algoritmo de Viterbi forneça uma verossimilhança acima do limiar.

Consequentemente, torna-se necessário uma normalização da verossimilhança da elocução teste pelo seu número de janelas. Como exemplo a Figura 5.12 apresenta um gráfico contendo no eixo X o número da elocução testada e no eixo Y o valor de sua verossimilhança.

Observa-se que os valores das verossimilhanças não normalizadas dos locutores falsos e locutor verdadeiro, na Figura 5.12, se misturam dificultando a determinação de um limiar de separação. A Figura 5.13, mostra o gráfico com as mesmas elocuções do gráfico anterior, porém, com a verossimilhança normalizada.

Na fase de teste, para cada elocução apresentada ao modelo λ, obtém-se como resposta um valor que ao ser comparado com um limiar — obtido pelo método de Bayes — determinará se a mesma é falsa ou verdadeira. A Figura 5.14 apresenta as distribuições das verossimilhanças dos locutores falsos e locutor verdadeiro e um limiar (teta) igual a -17,47 para um modelo treinado e testado com 5 grupos e 60 estados.

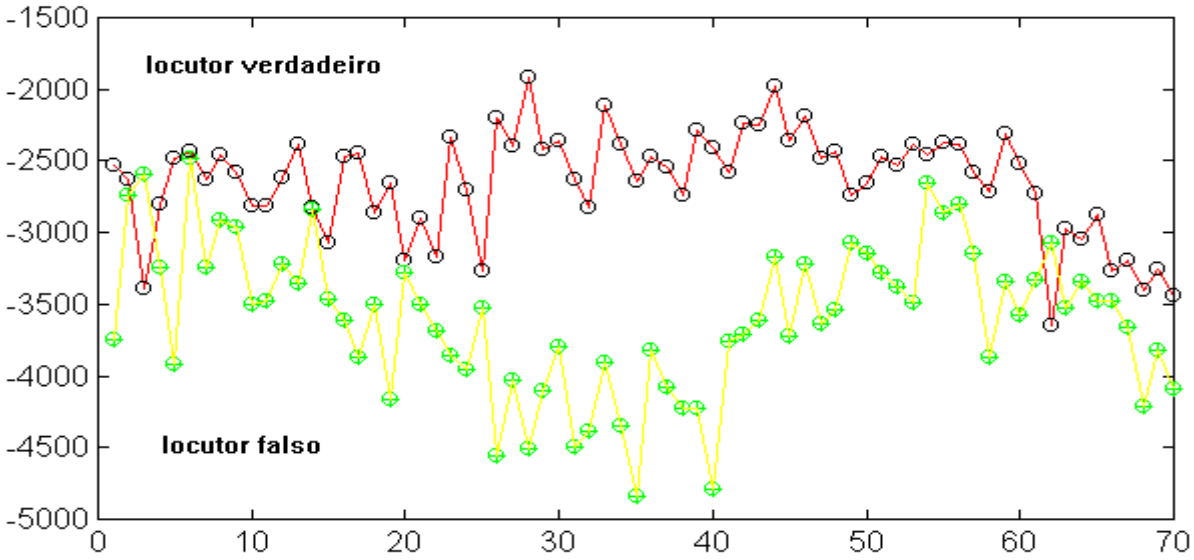


FIGURA 5.12: Esta figura apresenta os valores das verossimilhanças sem normalização obtidos das 70 elocuções do locutor verdadeiro e locutor falso. Representados por: ⊕ elocuções do locutor falso e ***o*** elocuções do locutor verdadeiro.

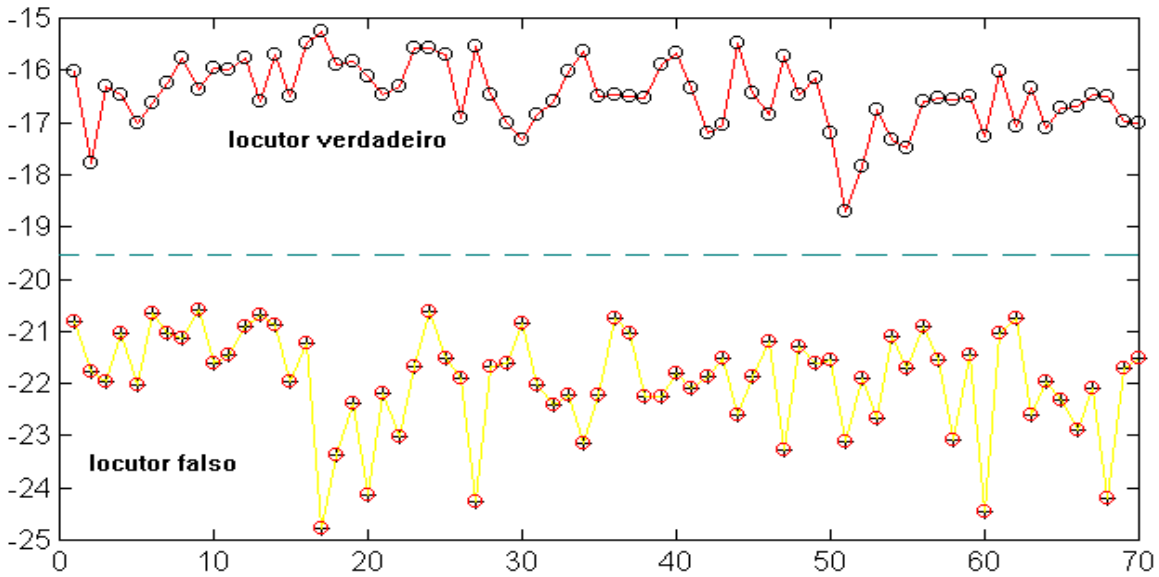


FIGURA 5.13: Após a normalização das verossimilhanças das elocuções observa-se uma maior separação entre as falsas e as verdadeiras. Neste, caso o limiar de separação poderia ser -20.

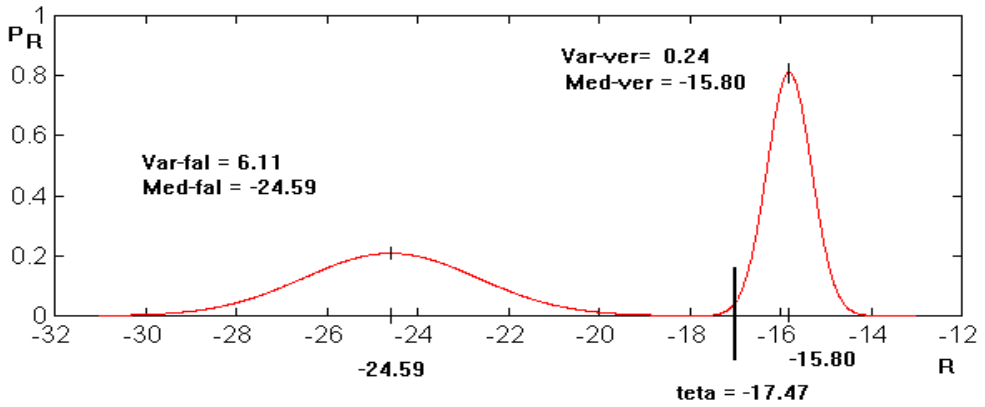


FIGURA 5.14: Distribuições das verossimilhanças das elocuções dos locutores falsos e elocuções do locutor verdadeiro separadas pelo limiar obtido pelo método Bayes, onde var-fal, med-fal, var-ver e med-ver são a variância e a média da distribuição dos locutores falsos e a variância e a média da distribuição do locutor verdadeiro, respectivamente. Com o eixo x mostrando os valores das verossimilhanças das elocuções obtidas no modelo.

Pelo fato do cálculo do limiar ser obtido sobre valores das verossimilhança das elocuções treinadas, então, pode acontecer erro de falsa rejeição quando se apresenta elocuções verdadeiras para teste, como é verificado na Figura 5.15 onde a área sob a linha tracejada representa a probabilidade de ocorrência das *elocuções verdadeiras não treinadas* (teste) e a área sob a linha contínua representa a probabilidade de ocorrência das *elocuções treinadas.* Nota-se que para este modelo, treinado com g5s60, o limiar distingue as elocuções verdadeiras das falsas mas não separa totalmente as elocuções verdadeiras utilizadas no teste, ocorrendo falsa rejeição.

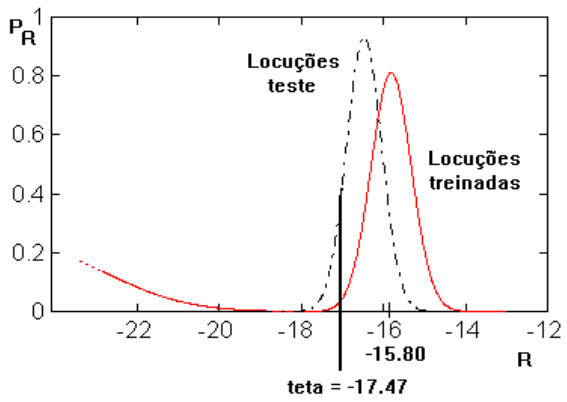


FIGURA 5.15: Comparação entre a distribuição das verossimilhanças das elocuções verdadeiras para treino e a distribuição das verossimilhanças das elocuções verdadeiras para testes.

A utilização de outros parâmetros no modelo λ altera as distribuições das verossimilhanças. Na utilização de 5 grupos e 12 estados as distribuições são como apresentam na Figura 5.16. Verifica-se que nestes parâmetros houve uma melhor aceitação das elocuções verdadeiras de teste pelo limiar, atribuindo valor zero para falsa aceitação e rejeição.

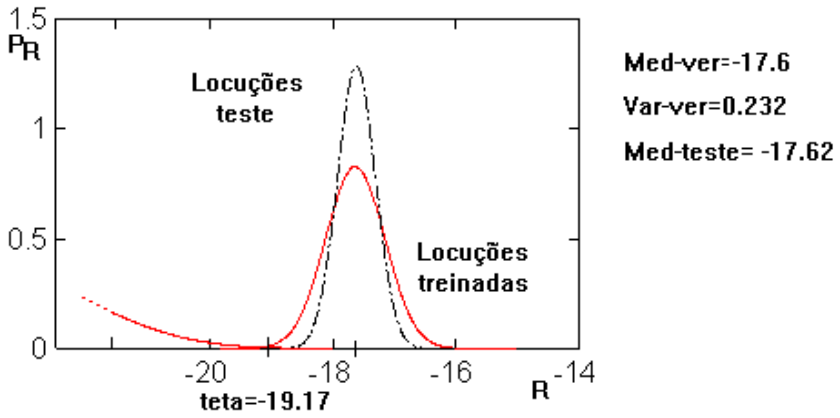


FIGURA 5.16: Distribuições das verossimilhanças das elocuções verdadeiras treinadas (linha contínua) e testadas (linha tracejadas).

5.7 - VERIFICAÇÃO DO LOCUTOR

O objetivo da verificação é verificar se uma dada elocução de teste pertence ou não ao locutor com o qual o modelo foi treinado, esta decisão é baseada em algum limiar previamente estipulado (Item 5.6.2).

Pode acontecer dois tipos de erros:

- *falsa rejeição:* quando o sistema rejeita uma elocução do locutor verdadeiro, ou seja, sendo a entrada uma elocução do locutor verdadeiro, a verossimilhança desta elocução fica abaixo do limiar estabelecido;

- *falsa aceitação:* quando o sistema reconhece um locutor estranho como verdadeiro, ou seja, sendo a entrada uma elocução pertencente a um locutor qualquer (diferente do utilizado no treinamento), a verossimilhança obtida fica acima do limiar estabelecido.

Quanto mais próximo está o limiar da média da distribuição das elocuções verdadeiras maior será a probabilidade de ocorrer um erro de *falsa rejeição*, e por outro lado, quanto mais distante estiver maior será a probabilidade de ocorrer erro de *falsa aceitação*.

5.8 - IDENTIFICAÇÃO DO LOCUTOR

A tarefa da identificação é sub-dividida entre identificação do locutor e identificação do locutor com rejeição. Na identificação, após a obtenção do valor da verossimilhança da elocução teste, para todos os modelos, escolhe-se a maior verossimilhança e atribui-se a elocução àquele modelo. Na identificação com rejeição, uma vez obtida a verossimilhança da elocução para todos os modelos (na Figura 5.17: v1, v2, v3, v4), seleciona-se aquele que forneceu a maior. Faz-se uma comparação desta verossimilhança com o limiar de verificação referente àquele modelo (Figura 5.17). Aceita-se quando maior e recusa-se caso contrário.

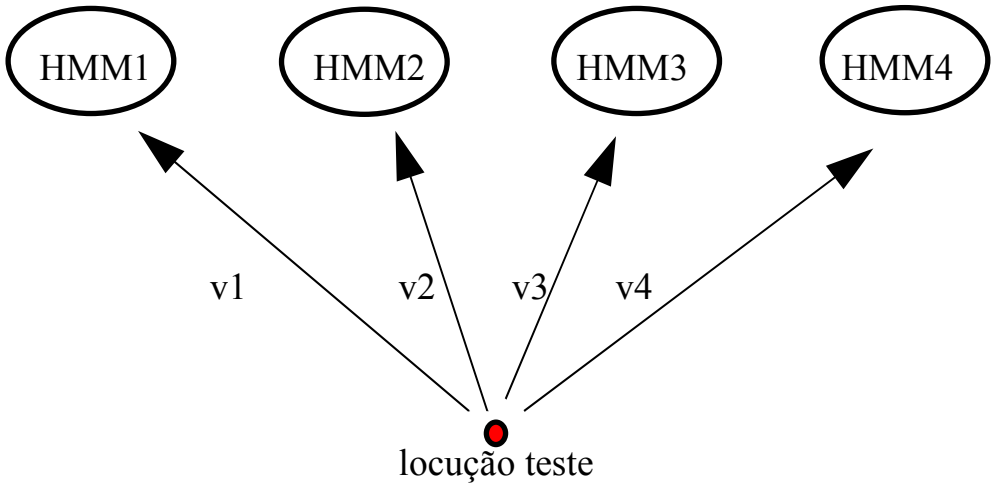


FIGURA 5.17: Ilustração da identificação e identificação com rejeição. No primeiro caso, aceita-se a elocução como verdadeira para um determinado modelo (HMM) se a verossimilhança (v) fornecida por ele for a maior entre todas. No segundo caso, após a seleção deste modelo (identificação) faz-se uma comparação com o limiar de verificação, aceitando-a ou rejeitando-a.

Poderão ocorrer por ocasião do teste, 3 tipos de erros:

- *falsa rejeição:* quando o sistema rejeita uma elocução pertecente a um dos locutores utilizados no treinamento, ou seja, quando a verossimilhança obtida com uma elocução teste (de um dos locutores utilizados no treinamento) fica abaixo do limiar estabelecido;

- *falsa aceitação:* quando o sistema atribui a elocução de um locutor estranho a um dos locutores utilizados no treinamento, ou seja, quando a verossimilhança obtida com uma elocução de teste de um locutor não utilizado no treinamento, ultrapassa o limiar estabelecido para qualquer um dos locutores utilizados no treinamento;

-*falsa identificação:* quando a verossimilhança de uma elocução de um locutor utilizado no treinamento ultrapassar o limiar estabelecido, porém de outro locutor utilizado no treinamento.

5.9 - PROGRAMAS DESENVOLVIDOS

Para a elaboração deste trabalho foram desenvolvidos vários programas utilizando o software MATLAB 4.2 for Windows da The MathWorks, Inc.

As principais rotinas desenvolvidas e utilizadas nesta tese fora as seguintes:

a) extração das características

endpoint.m	Programa que extrai os pontos extremos das elocuções.
melceps.m	Programa que faz o janelamento na elocução e extrai de cada janela os

12 mel cepestro, 12 delta mel cepestro, log-energia e delta log-energia.

b) treinamento e teste do HMM

spker.m	Programa principal que realiza o treinamento do HMM para um locutor
ini.m	Rotina que faz a inicialização para treinamento das elocuções.
kmeans.m	Rotina que realiza o agrupamento e obtém deles a média, covariância e coeficiente de misturas.
kmeans.exe	Programa escrito em linguagem C para agrupamento das observações pertencente a um determinado estado em k grupos.
matriza.m	Programa que gera uma matriz das probabilidades de transições aleatória.
h_prob.m	Rotina que calcula a matriz das probabilidades de observações.
viterbi.m	Programa que fornece a seqüência de estados mais provável, para uma determinada sequência de observações, e o valor de sua verossimilhança, dado o modelo λ.
reest.m	Rotina que realiza a reestimação dos parâmetros do modelo.
forward.m	Rotina que calcula as variáveis "forward".
backward.m	Rotina que calcula as variáveis "backward".
reconh.m	Programa que faz o reconhecimento das elocuções dado os modelos. Fornece duas matrizes com os valores das verossimilhanças de todos os locutores.
verif.m	Programa que a partir das matrizes fornecidas por "reconh.m" faz a verificação, a identificação e a identificação com rejeição.

CAPÍTULO 6

RESULTADOS OBTIDOS E AVALIAÇÃO DO SISTEMA

6.1 - INTRODUÇÃO

Os resultados obtidos no trabalho da Tese estão relatados e analisados neste capítulo. Desde a decisão sobre a escolha dos algoritmos até o desempenho geral do sistema. Para tal, serão abordados os seguintes tópicos:

- algoritmo de inicialização dos modelos
- tempo gasto para treinamento dos modelos
- resultados obtidos no reconhecimento
- estudo do limiar utilizado no reconhecimento
- avaliação do sistema

6.2 - ALGORITMO DE INICIALIZAÇÃO DO MODELO

No Item 3.6.1 mostrou-se a necessidade de estimar parâmetros iniciais que estejam próximos a um máximo da função de verossimilhança. Para a matriz A, algumas experiências mostraram que a estimação de valores aleatórios ou uniformes não influência na convergência do modelo. Entretanto, o treinamento do modelo mostrou-se mais sensível aos valores atribuídos inicialmente a matriz B. Assim, de acordo com o Item 5.6 onde apresentam os algoritmos de inicialização que tem como função dividir as seqüências de observações pelo número de estados do modelo e estimar as probabilidades iniciais das observações, faz-se uma avaliação de qual dos dois algoritmos de inicialização, dos parâmetros de B, apresenta um

melhor desempenho em relação à convergência do modelo. Realizaram-se vários testes com o conjunto CF, utilizando a frase fr1 e o modelo g5s36.

A Tabela 6.1 apresenta os resultados da divisão de 50 repetições (seqüências de observações) da elocução fr1 por 36 estados. Por motivos práticos, apenas os estados número 1, 35 e 36 são apresentados. Observa-se que o primeiro algoritmo possui, inicialmente, uma tendência em deixar o estado N com maior número de observações e que durante o treinamento as observações serão melhor modeladas e distribuídas de acordo com as probabilidades dos parâmetros variáveis reestimados. Já o segundo algoritmo faz uma distribuição mais homogênea das observações dentro dos estados na inicialização, facilitando a modelagem durante a reestimação.

TABELA 6.1: Valores dos números de observações obtidos pelos algoritmo 1 e 2, para a CF2fr1. A tabela mostra as quantidades de amostras (observações) separadas por estados.

	Início		**1° Iteração**		**2° Iteração**		**3° Iteração**	
Estados	**Alg 1**	**Alg 2**	**Alg 1**	**Alg 2**	**Alg 1**	**Alg 2**	**Alg 1**	**Alg 2**
1	224	258	191	210	194	204	189	202
35	224	228	139	200	370	213	396	213
36	1174	224	1016	456	654	442	537	445
Algoritmo 1			**Verossimilhança do modelo = -1.482 x 10^5 Tempo de processamento 14 horas e 29 min**					
Algoritmo 2			**Verossimilhança do modelo = -1.471 x 10^5 Tempo de processamento 14 horas e 35 min**					

Da Tabela 6.1 verificou-se que a distribuição das observações na inicialização pelo algoritmo 2 aproximou da distribuição obtida na convergência do modelo. Entretanto, para o algoritmo 1, a distribuição das observações na inicialização foi, para o estado 36, 637 observações acima do valor na convergência do modelo.

A Tabela 6.2 apresenta os valores das verossimilhanças, obtidas durante os treinamentos dos modelos para os dois algoritmos de inicialização. Foi utilizado o modelo com 36 estados e 5 grupos por estados para treinar o conjunto feminino usando a frase 1.

TABELA 6.2: Valores das verossimilhanças obtidas durante a fase de treinamento dos HMM's, para o conjunto de locutoras, com 36 estados e 5 grupos. Observa-se nos valores em negrito que o algoritmo 1 proporcionou uma melhor convergência somente para o HMM da locutora 1.

Locutor / alg	**Início**	**1° Iteração**	**2° Iteração**	**3° Iteração**
locutor 1 / 1	*-1.6256*	*-1.4819*	*-1.4652*	***-1.4583***
locutor 1 / 2	-1.6486	-1.4803	-1.4671	-
locutor 2 / 1	*-1.6423*	*-1.5071*	*-1.4864*	*-1.4824*
locutor 2 / 2	-1.6728	-1.4935	-1.4777	***-1.4707***
locutor 3 / 1	*-1.4637*	*-1.3265*	*-1.3165*	-
locutor 3 / 2	-1.4777	-1.3306	-1.3140	***-1.3017***
locutor 4 / 1	*-1.4799*	*-1.3726*	*-1.3505*	*-1.3382*
locutor 4 / 2	-1.5013	-1.3490	-1.3298	***-1.3248***
				1.0e+005

O algoritmo 2 proporcionou uma melhor separação das observações nos estados para todos os modelos, exceto para o locutor 1, o qual convergiu com 3 iterações e, mesmo assim, a diferença entre as verossimilhanças dos modelos foi de 0.0088. Para o locutor 3 e algoritmo 2, observa-se que apesar da ocorrência de 3 iterações e do maior tempo de treinamento (8h e 09min gasto pelo modelo do alg.1 e 13h e 40min pelo modelo do alg. 2), o valor da verossimilhança foi superior ao do algoritmo 1 já na segunda iteração. Observa-se também que os modelos locutor 1 (alg. 2) e locutor 3 convergiram na segunda iteração. Assim, das Tabelas 6.1 e 6.2, concluiu-se por meio da divisão das observações pelo algoritmo 2, ou seja uma divisão mais homogênea das observações, obteve-se um valor de convergência melhor na maioria dos modelos.

6.3 - TEMPO GASTO NO TREINAMENTO DO MODELO

6.3.1 - Tempo Relativo ao Programa Computacional Implementado

Os programas desenvolvidos para treinamento e teste dos modelos foram escritos na linguagem MATLAB (Item 5.9), por ser a linguagem muito utilizada nos meios acadêmicos e de fácil implementação. Entretanto, por ser uma linguagem interpretada, a execução dos programas ou rotinas torna-se lenta.

Fizeram-se, então, testes para verificar o tempo gasto no treinamento dos modelos. Realizaram-se dois tipos de teste. No primeiro treinaram-se 10 modelos, com 8 estados e 5 grupos, utilizando 30 elocuções de cada digíto de 0 a 9, pronunciadas por um locutor do sexo masculino e empregando-se o algoritmo "k-means" escrito em linguagem MATLAB. No segundo, realizou-se o mesmo procedimento descrito acima, entretanto, empregando-se o algoritmo "k-means" escrito em linguagem C. Os resultados obtidos estão apresentados na Tabela 6.3.

TABELA 6.3: Tempos gastos para treinar modelos que representem os digitos de 0 a 9. Na primeira coluna são mostrados os tempos obtidos quando utilizou-se o algoritmo "k-means" escrito em linguagem MATLAB, e na outra coluna os tempos obtidos quando utilizou-se o algoritmo "k-means" escrito em linguagem C.

#	TEMPO	
modelo	**Rotina em Matlab**	**Rotina em C**
0	2 horas e 49 min	0 hora e 32 min
1	0 hora e 43 min	0 hora e 15 min
2	2 horas e 31 min	0 hora e 32 min
3	2 horas e 33 min	0 hora e 30 min
4	3 horas e 9 min	0 hora e 32 min
5	3 horas e 16 min	0 hora e 35 min
6	2 horas e 54 min	0 hora e 34 min
7	3 horas e 0 min	0 hora e 35 min

8	2 horas e 50 min	0 hora e 30 min
9	2 horas e 43 min	0 hora e 33 min

Analisando os dados obtidos no teste, verificou-se que o tempo médio gasto para treinamento do modelo do dígito, empregando a linguagem MATLAB, foi de 2 horas e 38 minutos enquanto que para a linguagem C foi de 30 minutos, isto é, aproximadamente 5 vezes menos. Para treinamento do modelo do dígito, cada seqüência de observações possuía em média 67,20 observações, e, para as 30 repetições utilizadas somaram-se 2016 observações que foram dividas em N estados.

No treinamento de um modelo de um locutor, porém, cada seqüência de observações possui em média 164,71 observações — média das elocuções fr1 mais fr2 pronunciadas pelos locutores masculinos e femininos — e para as 50 repetições de uma elocução utilizadas somam-se 8235,5 observações a serem divididas em N estados e agrupadas em k grupos pelo algorítmo "k-means". Neste caso, o tempo de processamento, quando todas as rotinas são escritas em MATLAB, se tornará muito alto, dada a quantidade de dados a serem processados.

Portanto, nos treinamentos realizados para os modelos dos locutores utilizou-se programas híbridos em MATLAB e C.

6.3.2 - Tempo Relativo aos Valores dos Parâmetros Fixos do Modelo Utilizado

Para uma análise real do tempo gasto durante o treinamento dos modelos dos locutores, apresenta-se a Tabela 6.4, que mostra estes tempos para o conjunto feminino utilizando a frase 1 e a frase 2. Comparou-se os modelos g5s36, g10s36, g5s42 e g10s42 (onde g é o número de grupos e s o número de estados).

Nesta tabela, observa-se que para a frase 1 o aumento do número de estados ou de grupos provocou um aumento no tempo de treinamento. Enquanto que, para a frase 2, determinados locutores (L1, L3 e L4) tiveram seu tempo de treinamento reduzido quando se aumentou o número de estados.

TABELA 6.4: Tempos obtidos nos treinamentos dos conjuntos dos locutores femininos para os modelos g5s36, g10s36, g5s42 e g10s42. Entre parênteses representa-se a quantidade de iterações necessárias para a convergência..

		g5s36	**g10s36**	**g5s42**	**g10s42**
frase 1	**loc 1**	13h 27m (3)	15h 54m (2)	17h 30m (3)	18h 11m (2)
	loc 2	14h 29m (3)	15h 23m (2)	16h 55m (3)	26h 5m (3)
	loc 3	8h 10m (2)	13h 42m (2)	15h 23m (3)	16h 10m (2)
	loc 4	14h 30m (3)	14h 15m (2)	15h 53m (3)	24h 25m (3)
frase 2	**loc 1**	18h 31m (3)	29h 49m (3)	16h 28m (3)	25h 19m (3)
	loc 2	12h 3m (2)	19h 22m (2)	15h 29m (3)	23h 52m (3)
	loc 3	11h 5m (2)	26h 29m (3)	14h 14m (3)	22h 14m (3)
	loc 4	10h 19m (2)	18h 32m (3)	8h 53m (2)	13h 52m (2)

Tempo médio das frases para o conjunto feminino (s)	**frase 1**				**frase 2**			
	L1	**L2**	**L3**	**L4**	**L1**	**L2**	**L3**	**L4**
	1,862	1,796	1,693	1,743	1,763	1,622	1,487	1,391

A Figura 6.1 mostra, para CFfr2 (conjunto feminino utilizando a frase 2), as variações dos tempos ocorridos nos treinamentos e entre parênteses o número de iterações obtidos em cada modelo. O número de grupos é mantido constante para efeito de comparação com a Tabela 6.4.

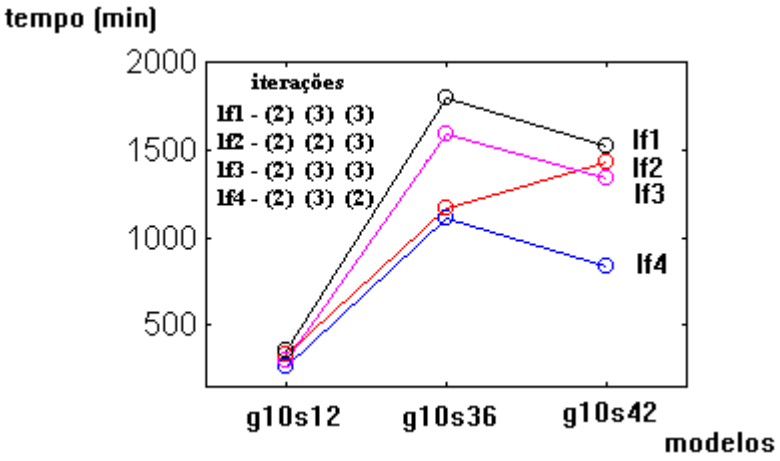


FIGURA 6.1: Variação do número de estados para o CFfr2 para um número de grupos fixo. O tempo é dado em minutos.

Entretanto, de acordo com a Tabela 6.4, observa-se que nada se pode concluir sobre um melhor desempenho global das frases fr1 e fr2 em relação ao *tempo de treinamento*. Para a Figura 6.1, em primeira análise, observa-se que o modelo g10s12 obteve, para todos os locutores do conjunto CFfr2, um melhor desempenho em relação ao tempo.

As Figuras 6.2, 6.3, 6.4 e 6.5 mostram o tempo de convergência no treinamento dos modelos g5s12, g5s36, g10s36 e g5s60 para os locutores CMfr1, CMfr2, CFfr1 e CFfr2.

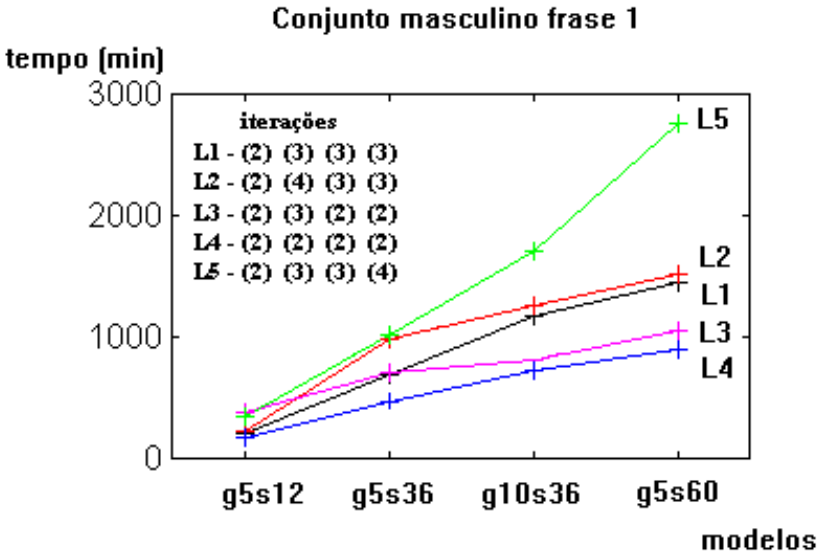


FIGURA 6.2: Conjunto masculino treinado com a frase 1. Entre parênteses representa-se a quantidade de iterações necessárias para a convergência.

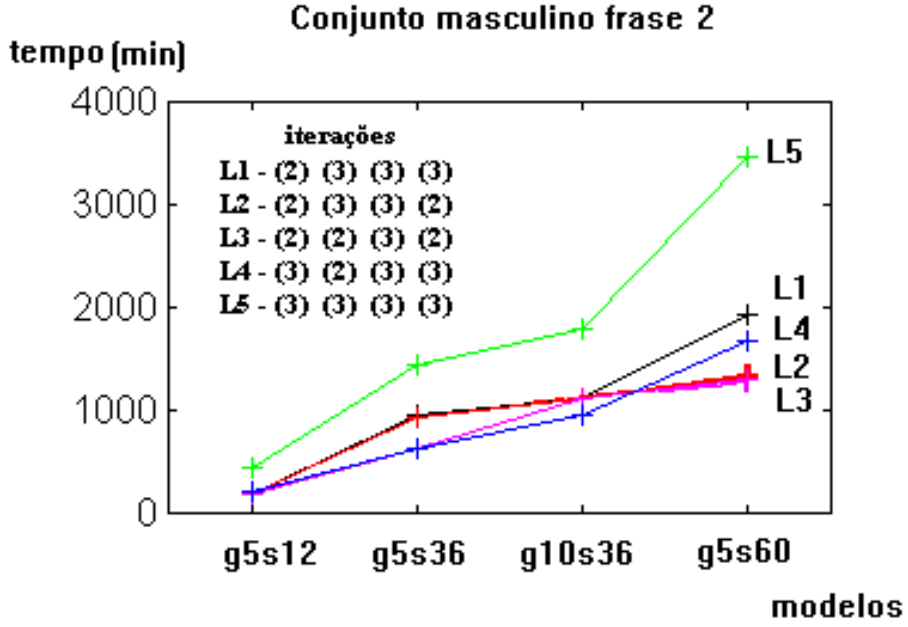


FIGURA 6.3: Conjunto masculino treinado com a frase 2. Entre parênteses representa-se a quantidade de iterações necessárias para a convergência.

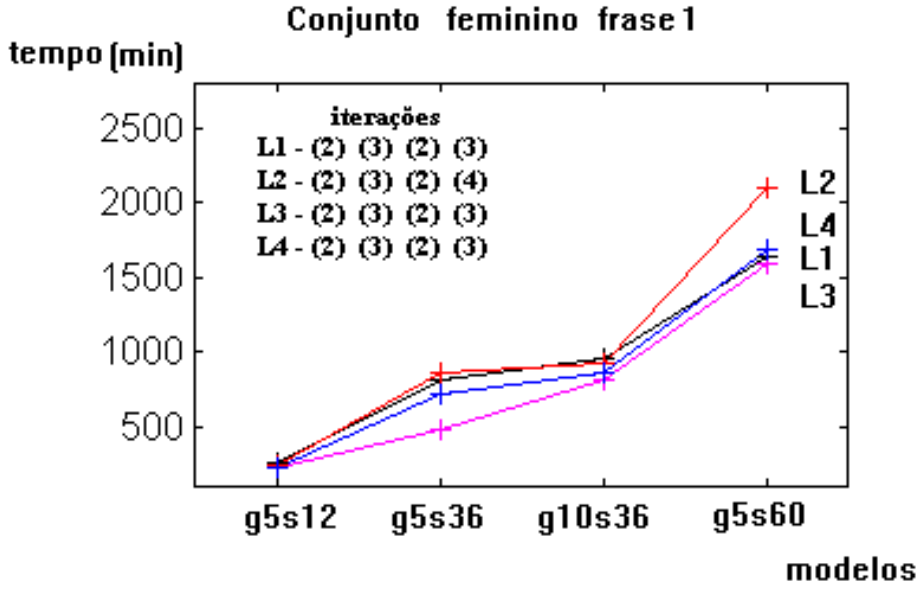


FIGURA 6.4: Conjunto feminino treinado com a frase 1. Entre parênteses representa-se a quantidade de iterações necessárias para a convergência.

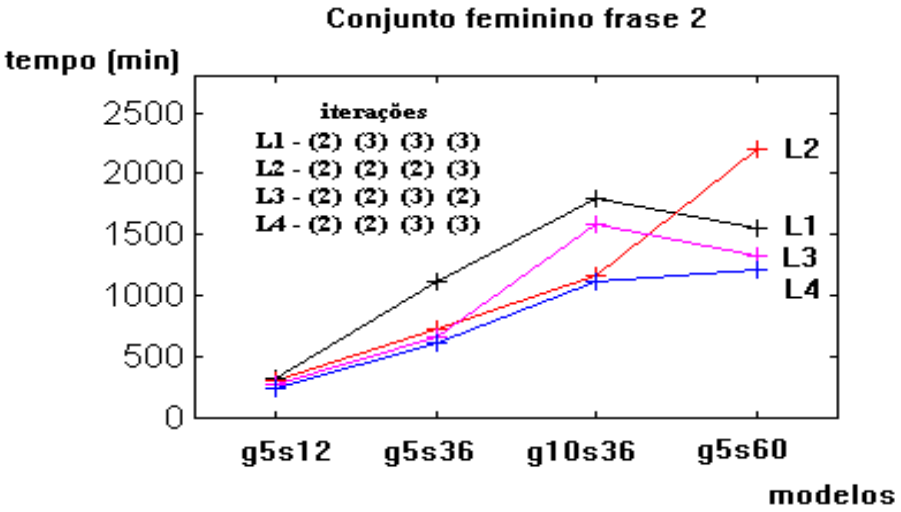


FIGURA 6.5: Conjunto feminino treinado com a frase 2. Entre parênteses representa-se a quantidade de iterações necessárias para a convergência.

Embora não seja claramente discernível, do ponto de vista do tempo da convergência, de qual frase apresentou um razoável desempenho no treinamento do modelo, algumas conclusões são evidentes:

- não existiu nenhuma das duas que apresentasse um melhor desempenho em todos os treinamentos dos modelos para conjuntos CM e CF, em relação ao tempo de treinamento.

- verificou-se que o aumento do número de estados não proporcionou diminuição do tempo de treinamento. Baseando-se nas Figuras 6.1 e 6.5, que representam o conjunto CFfr2, nota-se uma tendência da redução do tempo quando se aumenta o número de estados. Entretanto, pelas Figuras 6.2, 6.3 e 6.4, que representam os conjuntos CMfr1, CMfr2 e CFfr1 respectivamente, vê-se que não ocorreu nenhuma redução.

- constatou-se que o modelo que apresentou um melhor desempenho, em relação ao tempo de treinamento, para todos os locutores e frases utilizadas foi o modelo *g5s12*.

6.4 - RESULTADOS OBTIDOS NO RECONHECIMENTO

6.4.1 - Resultados Obtidos na Verificação do Locutor

Realizaram-se 137 treinamentos envolvendo locutores masculinos (conjunto masculino CM) e femininos (conjunto feminino CF). Após o treinamento, foram realizados dois testes: o primeiro, utilizando as elocuções empregadas no treinamento do modelo (**treino**), e o segundo, com as elocuções não utilizadas (**teste**), sendo essas independentes dos modelos treinados.

Os resultados obtidos na tarefa de verificação do locutor constam nas Tabelas 6.5 a 6.13, e estão divididas em *tabela treino* e *tabela teste* para cada conjunto de locutores — CM e CF — e frase — fr1 e fr2. Nas tabelas é mostrado o número de Falsa Rejeição (**FR**), Falsa Aceitação (**FA**), N_{total} — número total de locuções testadas — e Taxas de Aceitação (**TA**) que é definida pela seguinte equação:

$$TA = \frac{N_{total} - FA - FR}{N_{total}} \tag{6.1}$$

Observa-se que, os testes de verificação dos modelos do conjunto CM, utilizando as frases fr1 e fr2, não apresentaram nenhum erro de falsa aceitação.

Quando se realizou o teste com as elocuções treinadas, verificou-se que para o conjunto CMfr2 nos modelos g5s60, g10s36 e g10s42 não foi obtido nenhuma falsa rejeição, e, para o conjunto CMfr1 foram obtidas falsas rejeições em todos os testes dos modelos. Cabe ressaltar que todos os modelos do locutor 3 do CMfr1 rejeitaram a elocução nº 49.

No conjunto CF obteve-se falsa aceitação para a frase fr1 no modelo g3s12 e para a fr2 nos modelos g3s12 e g5s12. Praticamente em todos os modelos, com as excessões

de g10s60 (lf1) e g10s42 (lf2), apresentaram-se falsas rejeições. Nestes testes não foi verificado o desempenho do sistema no reconhecimento de locutores mímicos.

TABELA 6.5: Resultados da *verificação* do CM utilizando a frase 1.

CMfr1 Tabela treino (%)								N_{total}=50
# locutor	**g5s12**		**g5s36**		**g5s60**			
	TA	**FR**	**TA**	**FR**	**TA**	**FR**		
1	98	1	98	1	100	-		
2	100	-	100	-	100	-		
3	98	1	98	1	98	1		
4	100	-	100	-	100	-		
5	98	1	100	-	100	-		
Tabela teste (%)								N_{total}=360
# locutor	**g5s12**			**g5s36**			**g5s60**		
	TA	**FA**	**FR**	**TA**	**FA**	**FR**	**TA**	**FA**	**FR**
1	99,72	-	1	99,44	-	2	98,89	-	4
2	100	-	-	99,72	-	1	99,44	-	2
3	99,72	-	1	98,89	-	4	98,33	-	6
4	97,78	-	8	96,11	-	14	95,56	-	16
5	100	-	-	100	-	-	100	-	-

TABELA 6.6: Continuação do resultado da *verificação* do CM utilizando a frase 1.

CMfr1 Tabela treino (%)					N_{total}=50
# locutor	**g10s12**		**g10s36**		
	TA	**FR**	**TA**	**FR**	
1	100	-	100	-	
2	100	-	100	-	
3	98	1	98	1	
4	100	-	100	-	
5	100	-	100	-	
Tabela teste (%)					N_{total}=360
# locutor	**g10 s12**			**g10 s36**		
	TA	**FA**	**FR**	**TA**	**FA**	**FR**
1	99,44	-	2	98,61	-	5
2	100	-	-	99,17	-	3
3	99,17	-	3	98,33	-	6
4	96,67	-	12	95,56	-	16
5	100	-	-	100	-	-

TABELA 6.7: Resultado da *verificação* do CM utilizando a frase 2.

CMfr2 Tabela treino (%)								N_{total}=50
# locutor	g5s12		g5s36		g5s42		g5s60	
	TA	FR	TA	FR	TA	FR	TA	FR
1	100	-	100	-	100	-	100	-
2	100	-	100	-	100	-	100	-
3	100	-	100	-	100	-	100	-
4	98	1	100	-	100	-	100	-
5	100	-	98	1	98	1	100	-

Tabela teste (%)												N_{total}=360
# locutor	g5s12			g5s36			g5s42			g5s60		
	TA	FA	FR	TA	FA	FR	TA	FA	FR	TA	FA	FR
1	100	-	-	100	-	-	99,72	-	1	99,44	-	2
2	100	-	-	100	-	-	100	-	-	100	-	-
3	99,72	-	1	99,72	-	1	99,17	-	3	99,72	-	1
4	97,50	-	9	97,78	-	8	96,44	-	11	99,44	-	11
5	99,72	-	1	98,61	-	5	98,61	-	5	98,06	-	7

TABELA 6.8: Continuação do resultado da *verificação* do CM utilizando a frase 2.

CMfr2 Tabela treino (%)					N_{total}=50
# locutor	g10s36		g10s42		
	TA	FR	TA	FR	
1	100	-	100	-	
2	100	-	100	-	
3	100	-	100	-	
4	100	-	100	-	
5	100	-	100	-	

Tabela teste (%)						N_{total}=360
# locutor	g10s36			g10s42		
	TA	FA	FR	TA	FA	FR
1	99,17	-	3	99,44	-	2
2	99,17	-	3	100	-	-
3	98,89	-	4	97,22	-	10
4	96,11	-	14	95,56	-	16
5	96,44	-	11	95,83	-	15

TABELA 6.9: Resultado da *verificação* do CF utilizando a frase 1.

CFfr1 Tabela treino (%)								N_{total}=50
# locutor	g5s12		g5s36		g5s42		g5s60	
	TA	FR	TA	FR	TA	FR	TA	FR
1	100	-	100	-	100	-	100	-
2	98	1	98	1	100	-	98	1

3	98	1	100	-	100	-	100	-
4	98	1	98	1	100	-	100	-

Tabela teste (%)											**N_{total}=290**	
#	**g5s12**			**g5s36**			**g5s42**			**g5s60**		
locutor	**TA**	**FA**	**FR**	**TA**	**FA**	**FR**	**TA**	**FA**	**FR**	**TA**	**FA**	**FR**
1	98,97	-	4	97,59	-	7	97,59	-	7	96,21	-	11
2	98,28	3	2	98,97	-	3	99,31	-	2	98,62	-	4
3	97,93	2	4	98,62	-	4	98,62	-	4	97,59	-	7
4	97,93	1	5	97,24	-	8	97,59	-	7	97,24	-	8

TABELA 6.10: Continuação do resultado da *verificação* do CF utilizando a frase 1.

CFfr1 Tabela treino (%)					**N_{total}=50**
#	**g10s36**		**g10s60**		
locutor	**TA**	**FR**	**TA**	**FR**	
1	100	-	100	-	
2	100	-	100	-	
3	100	-	100	-	
4	98	1	100	-	

Tabela teste (%)							**N_{total}=290**
#	**g10s36**			**g10s60**			
locutor	**TA**	**FA**	**FR**	**TA**	**FA**	**FR**	
1	97,59	-	7	94,83	-	15	
2	97,93	-	6	95,86	-	12	
3	98,28	-	5	97,24	-	8	
4	96,90	-	9	94,14	-	17	

TABELA 6.11: Resultado da *verificação* do CF utilizando a frase 2.

CFfr2 Tabela treino (%)							**N_{total}=50**
#	**g3s12**		**g3s24**		**g3s36**		
locutor	**TA**	**FR**	**TA**	**FR**	**TA**	**FR**	
1	96	2	96	2	96	2	
2	98	1	98	1	98	1	
3	100	-	100	-	100	-	
4	98	1	100	-	100	-	

Tabela teste (%)									**N_{total}=290**
#	**g3s12**			**g3s24**			**g3s36**		
locutor	**TA**	**FA**	**FR**	**TA**	**FA**	**FR**	**TA**	**FA**	**FR**

1	98,97	-	3	98,28	-	5	97,93	-	6
2	97,93	6	-	99,66	-	1	98,97	-	3
3	97,59	6	1	98,97	-	3	98,97	-	3
4	98,62	4	-	100	-	-	100	-	-

TABELA 6.12: Continuação do resultado da *verificação* do CF utilizando a frase 2.

CFfr2 Tabela treino (%)								**N_{total}=50**
#	**g5s12**		**g5s36**		**g5s42**		**g5s60**	
locutor	**TA**	**FR**	**TA**	**FR**	**TA**	**FR**	**TA**	**FR**
1	96	2	100	-	98	1	100	-
2	98	1	98	1	100	-	100	-
3	98	1	98	1	98	1	98	1
4	100	-	100	-	100	-	100	-

Tabela teste (%)											**N_{total}=290**
#	**g5s12**			**g5s36**			**g5s42**			**g5s60**	
locutor	**TA**	**FA**	**FR**	**TA**	**FA**	**FR**	**TA**	**FA**	**FR**	**TA**	**FA**	**FR**
1	98,97	-	3	97,59	-	7	97,93	-	6	96,21	-	11
2	100	-	-	97,59	-	7	96,55	-	10	97,24	-	8
3	98,28	2	3	98,97	-	3	98,97	-	3	98,28	-	5
4	100	-	-	100	-	-	99,66	-	1	99,66	-	1

TABELA 6.13: Continuação do resultado da *verificação* do CF utilizando a frase 2.

CFfr2 Tabela treino (%)				**N_{total}=50**
#	**g10s12**		**g10s42**	
locutor	**TA**	**FR**	**TA**	**FR**
1	100	-	100	-
2	98	1	100	-
3	98	1	100	-
4	100	-	100	-

Tabela teste (%)						**N_{total}=290**
#	**g10s12**			**g10s42**		
locutor	**TA**	**FA**	**FR**	**TA**	**FA**	**FR**
1	98,62	-	4	96,21	-	11
2	99,66	-	1	96,90	-	9
3	98,97	-	3	97,24	-	8
4	100	-	-	99,31	-	2

Conforme apresentaram as Tabelas 6.5 a 6.13, o aumento do número de estados tanto para o conjunto CM quanto CF provocou uma especialização dos modelos às elocuções utilizadas no treinamento. Porquanto, às elocuções para teste são rejeitadas em maior número com o aumento do número de estados ou Gaussianas por misturas.

Verifica-se que devido ao número fixo de observações, ou seja uma mesma quantidade de repetições da frase para treinamento de todos os modelos, uma melhor reestimação dos parâmetros não é permitida via algoritmo "forward-backward", que utiliza para cálculo das probabilidades das transições e das observações o valor esperado das ocorrências dos eventos das observações nos estados.

6.4.2 - Resultados Obtidos na Identificação do Locutor

(a) *Identificação sem Rejeição*

Na identificação sem rejeição todos os conjuntos, em todos os testes, obtiveram 100% de acerto, com excessão do conjunto CMfr2 que identificou a elocução de número 20 pertecente ao locutor 4 como sendo do locutor 1, em todos os testes.

A Figura 6.6 mostra um gráfico que apresenta os valores das verossimilhanças obtidas por cada elocução da frase fr2 do locutor masculino 4 quando testadas em seu modelo.

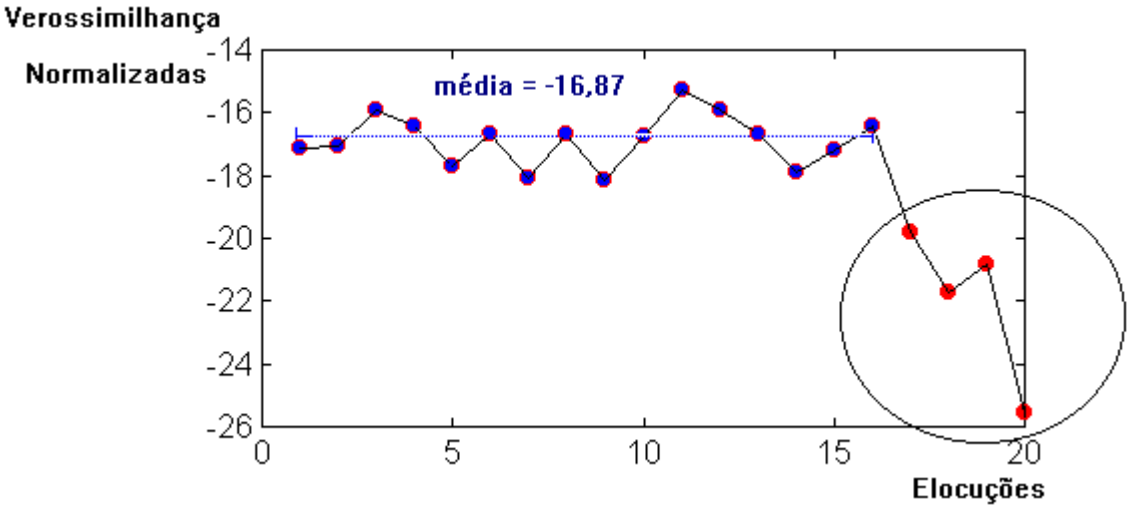


FIGURA 6.6: Valores das verossimilhanças obtidas pelas elocuções teste (frase 2) do locutor 4 no modelo λ (g5s36).

Verifica-se, na Figura 6.6, que as elocuções 17 a 20 obtiveram suas verossimilhanças bem abaixo da média das elocuções de 1 a 16. A Figura 6.7 mostra as verossimilhanças obtidas pelas elocuções do locutor verdadeiro (locutor 4) e as obtidas pelas elocuções dos falsos locutores (locutores 1, 2, 3, e 5) no modelo do locutor 4. Observam-se dois grupos distintos. Entretanto, a elocução de número 20 apresenta-se como pertencente ao grupo dos falsos locutores. Assim, na tarefa de identificação sem rejeição essa elocução foi atribuída ao locutor 1, como mostra a Figura 6.8 onde a maior verossimilhança foi obtida pela elocução 20 do locutor 4 no modelo do locutor 1.

Este resultado, apresentado pela Figura 6.6, mostra que apesar das elocuções 17,18,19 e 20 estarem com os pontos extremos bem delimitados entre o sinal de voz e o ruído, possui uma diferença nas verossimilhanças obtidas em relação às outras elocuções verdadeiras. Portanto, pode-se concluir que foram pronunciadas de forma diferente, ou seja, contendo eventos acústicos não apresentados nas elocuções de treinamento.

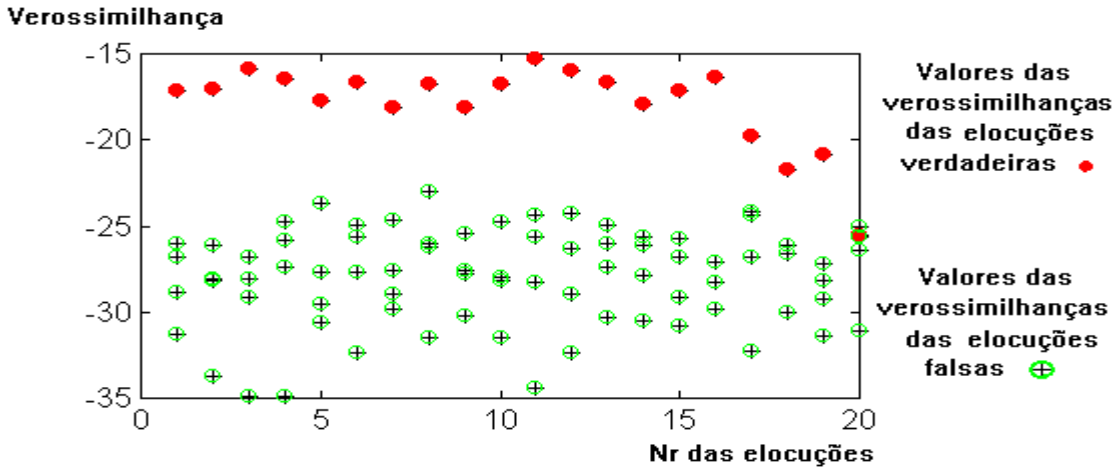


FIGURA 6.7: Apresenta os valores das verossimilhanças obtidas pelas elocuções verdadeiras e falsas no modelo treinado do locutor masculino 4.

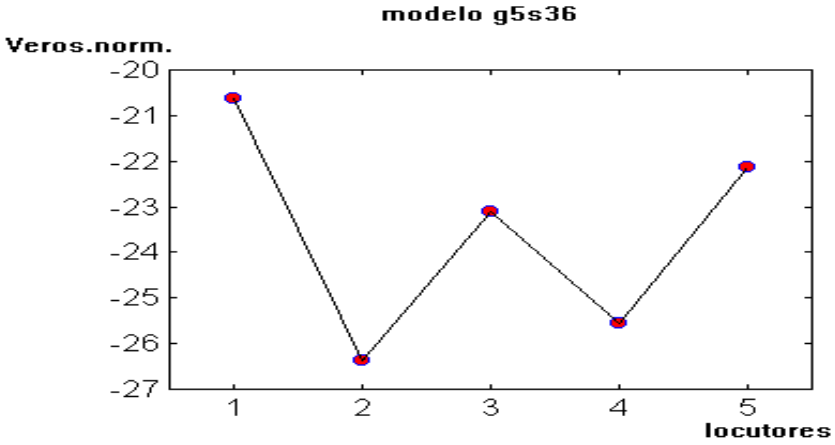


FIGURA 6.8: Valores das verossimilhanças da elocução 20, pertencente ao locutor 4, obtidos nos modelos dos 5 locutores do conjunto CMfr2. Neste caso, ocorre uma falsa aceitação com a identificação do locutor 1 como verdadeiro.

Como pôde ser verificado na Figura 6.8, a elocução 20 pertencente ao locutor 4 é identificado como do locutor 1 Entretanto, no caso da identificação com rejeição, este erro não ocorre porque após a seleção do modelo com a maior verossimilhança compara-se esta com o limiar do modelo. Neste caso, o valor para o locutor 1, utilizando o modelo g5s36, é igual a -18.6730.

(b) *Identificação com Rejeição*

Na realização dos testes de identificação com rejeição verificou-se que apenas os locutores utilizados para teste (Item 5.4) obtiveram elocuções falsamente identificadas por

algum modelo. E, dos resultados da verificação nota-se que apenas o conjunto CF obteve erro de falsa aceitação. Portanto, as Tabelas 6.14 e 6.15 mostram os valores obtidos na identificação dos conjuntos CFfr1 e CFfr2, respectivamente.

TABELA 6.14: Resultados do teste de identificação para o modelo g5s12 do conjunto CFfr1

Tabela teste (%) de CFfr1			**N_{total}=290**
# locutor	**g5s12**		
	TA	**FA**	**FR**
1	98,62	-	4
2	98,31	-	2
3	97,62	-	4
4	97,93	1 (locutor 7)	5

TABELA 6.15: Resultados do teste de identificação para o modelo g3s12 do conjunto CFfr2

Tabela teste (%) de CFfr2			**N_{total}=290**
# locutor	**g3s12**		
	TA	**FA**	**FR**
1	98,97	-	3
2	98,62	4 (locutor 5)	-
3	98,97	2 (locutor 5)	1
4	98,97	3 (locutor 5)	-

Apesar do modelo g5s12 do conjunto CFfr2 ter obtido resultados de FA, na identificação estes resultados foram atribuídos corretamente porque as elocuções pertenciam a locutores utilizados para treinamentos. Assim, a taxa de aceitação para este modelo será conforme a Tabela 6.16.

TABELA 6.16: Resultados do teste de identificação para o modelo g5s12 do conjunto CFfr2

Tabela teste (%) de CFfr2			**N_{total}=290**
# locutor	**g5s12**		
	TA	**FA**	**FR**
1	98,97	-	3
2	100	-	-
3	98,97	-	3

4	100	-	-

Para os locutores de treinamento, fez-se comparação da verossimilhança, e ocorreu somente "falça rejeição".

6.5 - ESTUDO DO LIMIAR UTILIZADO NO RECONHECIMENTO

Após a realização dos testes verificou-se que a quase totalidade dos erros observados foram de falsa rejeição do locutor treinado. Conclui-se que os limiares estavam um pouco abaixo do valor ideal para o sistema, o que pode ser atribuído ao pequeno número de elocuções utilizadas no cálculo dos limiares (ver Figura 5.11 e 5.12).

Um pequeno ajuste dos limiares levou a um resultado sensivelmente melhor. O ajuste consistiu em verificar um limiar que proporcionasse um menor erro de falsa aceitação e de falsa rejeição.

No teste de verificação foram obtido os seguintes resultados:

- o conjunto CMfr1 obteve 100% de acerto em todos os modelos dos locutores à exceção dos modelos do locutor 4 que recusaram as elocuções testes nºs 18 e 19, em todos os testes.

- o conjunto CMfr2 obteve da mesma forma 100% de acerto para todos os locutores à exceção dos modelos do locutor 4 que recusaram a elocução teste nº 20, em todos os testes.

- o conjunto CFfr1 apresentou o pior desempenho para as elocuções teste em todos os modelos:

  - os modelos do locutor 2 recusaram a elocução nº 20.

  - os modelos do locutor 3 recusaram a elocução nº 10.

  - os modelos do locutor 3 recusaram as elocução nºs 8, 13, 14, 18, 19 e 20.

- o conjunto CFfr2 obteve 100% de acerto para todos os locutores à exceção dos modelos do locutor 3 que recusaram as elocuções testes n^{os} 3, 8 e 17 em todos os testes.

No teste da verificação utilizando o novo limiar, não ocorreu nenhuma FA. Portanto, para o teste de identificação com rejeição os resultados foram os mesmos do teste da verificação.

6.6 - AVALIAÇÃO DO SISTEMA

Em relação ao tempo gasto para treinamento do modelo, nenhuma das duas *frases* apresentou melhor desempenho para os conjuntos CM e CF. Apenas, constatou-se que o *modelo* com melhor desempenho, em relação ao tempo de treinamento, para todos os locutores e frases utilizadas foi o modelo *g5s12*.

A partir dos resultados obtidos nas tabelas de teste de verificação observa-se que aumentando-se o número de estados ou de grupos por mistura no modelo, o número de vetores pertecentes a cada grupo de cada estado diminui, o que torna a estimação dos parâmetros menos precisa, ocasionando uma queda no desempenho.

Assim, aumentando-se o número de estados e de grupos por mistura ocorre maior separação entre as médias das Gaussianas. Entretanto, suas variâncias aumentam. De acordo com os resultados obtidos, verifica-se que a probabilidade de aceitação é melhor para as locuções treinadas e diminui para as locuções não treinadas, ou seja: existe um condicionamento dos parâmetros para as locuções treinadas e o poder de generalização do HMM diminui.

O método de Bayes, utilizado no cálculo do limiar, não fornece boas estimativas quando os HMM são treinados com muitos estados e grupos por mistura. Por exemplo, para

os modelos g10s42 (CMfr2) e g5s60 (CMfr1), um limiar heurístico adequadamente escolhido, utilizando-se os resultados de teste, mostrou que as TA podem alcançar 100%. É provável que este resultado ocorra devido ao fato de que as distribuições das verosimilhanças dos HMM não sejam exatamente Gaussianas, como foi assumida no cálculo do limiar de Bayes.

Não houve uma frase que apresentasse um resultado global melhor em todos os treinamentos e testes. Entretanto, para o conjunto CM a frase que apresentou melhores taxas foi a fr1 e para o conjunto CF foi a fr2.

O conjunto de locutores que obteve o melhor desempenho foi o conjunto CM quando utilizado com a frase 1. Neste caso a taxa de falsa aceitação foi zero.

O modelo composto por 5 grupos e 12 estados foi o que apresentou melhor desempenho global. Entretanto, para os locutores femininos foi o que apresentou um maior número de falsa aceitação. É importante ressaltar que estes modelos foram os únicos, nos dois conjuntos, que apresentaram este tipo de erro.

O CF necessitou de um maior número de treinamentos com valores variados de parâmetros fixos para obter um modelo que melhor o representassem. Observando, assim, que as locutoras femininas utilizando a frase fr2 possuem uma maior dificuldades em ser distingüidas entre si.

CAPÍTULO 7

CONCLUSÕES E SUGESTÕES

Ao término deste trabalho conclui-se que os Modelos de Markov Escondidos Contínuos apresentaram um bom desempenho em relação a TA no reconhecimento. Obteve-se para a maioria dos modelos treinados uma TA acima de 98%. Entretanto, para as elocuções utilizadas no teste (não-treinadas) houve um decréscimo da TA nos modelos com maior número de estados ou grupos por misturas. A explicação para estes resultados está no número fixo (igual a 50) de seqüências de observações utilizadas para o treinamento de todos os modelos. Como o algoritmo "Segmental kmeans" utiliza o procedimento Baum - Welch, que obtém o valor esperado das ocorrências das observações nos estados, um baixo número de observações não proporciona uma reestimação eficiente dos parâmetros do modelo.

Apesar das TA serem valores menores para as elocuções de testes, verificou-se que a maioria de erros foram de falsa rejeição do locutor treinado. Pode-se concluir que o limiar utilizado foi abaixo do valor ideal para o sistema. Para provar esta teoria, fez-se uma alteração do limiar, obtido pelo método de Bayes, encontrando-se uma TA igual a 100% para quase todos os modelos. Desta forma, este estudo mostrou que o método de Bayes não fornece boas estimativas quando os HMM são treinados com muitos estados e grupos por misturas. É provável que este resultado ocorra devido ao fato de que a distribuição das verossimilhaças dos HMM não sejam exatamente Gaussianas, como foi assumida no cálculo do limiar de Bayes.

Uma sugestão para futuras pesquisas utilizando os modelos HMM's é avaliar os resultados obtidos por um outro método de cálculo do limiar.

Neste trabalho fez-se a normalização da verossimilhança da elocução teste obtida no reconhecimento pelo comprimento de sua seqüência de observações, portanto todos os resultados da TA foram obtidos a partir desse método de normalização. Seria interessante a utilização de outros métodos de normalização e sua comparação com o limiar, como uma outra forma de avaliação para trabalhos futuros,

Uma outra sugestão seria o treinamento do modelo utilizando poucos estados um número maior de Gaussianas por misturas, e de repetições o qual poderá proporcionar uma melhor representação das observações dentro do estado.

REFERÊNCIAS BIBLIOGRÁFICAS

1. LADEFOGED, P. & LADEFOGED, J. The Ability of Listeners to Identify Voices, UCLA Working Papers in Phonetics, **1980**, nº 49, pp 43-51.

2. VAN LANCKER, D.; KREIMAN, J.; EMMOREY K. Familiar Voice Recognition: Patterns and Parameters - Recognition of Backward Voices, Journal Phonetics, **1985**, nº 13, pp 19-38.

3. SORIA, R. A. B. & CABRAL Jr., E. F. Comparison of Different Neural Paradigms in a Speaker Recognition Task Using Mel-Frequency Cepstral Coefficients Correlations, XIV Simpósio Brasileiro de Telecomunicações, Jul. **1996**, Vol. 2, pp. 521-526.

4. ATAL, B. S. Automatic Recognition of Speakers from Their Voices, Proceedings of IEEE, Abril **1976**, vol 64, nº 04, pp 460-475.

5. ROSEMBERG, A. E. Automatic Speaker Verification: A Review, Proceedings of IEEE, Abril **1976**, vol. 64, nº 04, pp 475-487.

6. RABINER, L. R. Applications of Voice Processing to Telecommunications, Proceedings of IEEE, february **1994**, vol. 82, nº 2, pp197-228.

7. MAMMONE, R. J. et alii. Robust Speaker Recognition - A Feature-Based Approch, IEEE Signal Processing Magazine, september **1996**, pp 58-71.

8. HERMANSKY, H. & MORGAN, N. RASTA Processing of Speech, IEEE Trans. Speech, Audio Processing, October **1994**, vol 2, pp 578-589.

9. HERMANSKY, H. Perceptual Linear Predection (PLP) Analysis of Speech, Journal Acoustic Soc. Am., Abril **1990**, vol 87, nº 4, pp 1738-1752.

10. OPENSHAW, J. P. et alii. A Comparison of Composite Features Under Degraded Speech in Speaker Recognition, IEEE, **1993**, vol. II, pp 371-374.

11. MIHELIC, F. et alii. Comparison of Features and Classification Rules for Acoustic-Phonetic Transcrition of Slovene Speech, Digital Signal Processing — Processing of the International Conference Florence, Italy, **1991**, pp 453-457.

12. RABINER, L. R. & JUANG, B.H. **Fundamentals of Speech Recognition**, Prentice Hall, Inc., Englewood Clifs, Nova Jersey, 1993.

13. DELLER Jr., J. R. et alii. **Discrete-Time Processing of Speech Signals**, Macmillian Publishing Company, Nova Iorque, 1993.

14. PICONE, J. Continuous Speech Recognition Using Hidden Markov Models, IEEE

Acoustics, Speech and Signal Processing Magazine, Julho **1990**, vol. 07, pp.26-41.

15. BEZERRA, M. R. **Reconhecimento Automático de Locutor para Fins Forenses, Utilizando Técnicas de Redes Neurais**, Tese de Mestrado, IME, Rio de Janeiro, 1994.

16. BARNARD, E. et alii. Real-World Speech Recognition with Neural Networks, CSLU, **1995**, Oregon Graduate Institute,USA.

17. ZHU, X. Y. A Combined Neural Network and Hidden Markov Model Approach to Speaker Recognition, IEEE Region 10 Internat Conference on Computers, Communications and Automation, **1993**, vol. 2, pp 19-21.

18. BOURLARD, H. Merging Multilayer Perceptrons and Hidden Markov Models: Some Experiment in Continuous Speech Recognition, Neural Network Advances and Applications, **1991**, North-Holland, The Netherlands, pp. 215-239.

19. SETLUR, A. R. et alii. Speaker Verification Using Mixture Likelihood Profiles Extracted from Speaker Independent Hidden Markov Models, IEEE , **1996**, pp 109-112.

20. PICONE, J. Signal Modeling Techiques in Speech Recognition, Proceedings of the IEEE, Set. **1993**, vol.81, nº 9, pp 1215-1246.

21. PROAKIS, J. G. **Digital Communications**, McGraw-Hill, New York, 2º ed, 1989.

22. MARKEL, J. & GRAY, A. H. **Linear Prediction of Speech**, Springer-Verlag, New York, 1980.

23. RABINER, L. R. & SCHAFER, R. W. **Digital Processing of Speech Signal**, Prentice-Hall, Englawood Clifs, New Jersey, 1978.

24. OPPENHEIM, A. V. & SCHAFER, R.W. **Digital-Time Signal Processing**, Prentice-Hall, Inc, Englewood Clifs, New Jersey, 1978.

25. / SOUSA, R. H. G. **Estudo de Características Relevantes do Sinal de Voz Para o Reconhecimento Automático do Locutor Desprevenido, Independente do Texto**, Tese de Mestrado, IME, Rio de Janeiro, 1996.

26. FURUI, S. Comparison of Speaker Recognition Methods Using Statistical Features ans Dynamic Features, IEEE Transaction on Acoustics, Speech and Signal Processing, Junho **1981**, vol. 29, nº 3, 448-449.

27. FURUI, S. Cepstral Analysis Technique for Automatic Speaker Verification, IEEE Transactions on Acoustics, Speech and Signal Processing, Abril **1981**, vol. 29, nº 2, pp 254-272.

28. REYNOLDS, D. A. Experimental Evaluation of Features for Robust Speaker Identification, IEEE Transaction Speech Audio Process, Out. **1994**, vol. 2, pp 639-643.

29. WEBB, J. J. & RISSANEN, E. L. Speaker Identification Experiments Using HMM's, ICASSP,**1993**, vol. 2, pp 387-390.

30. LEE, C. H. L. et alii. Speaker Independent Continuous Speech Recognition Using Continuous Density Hidden Markov Models, Proc. NATO-ASI, Speech Recognition and Understanding: Recent Advances, Trends and Applications, P. Laface and R. DeMori, Eds., Springer-Verlag, Cetaro, Italia, **1992**, pp. 135-163.

31. TSENG, B. L. et alii. Continuous Probabilistic Acoustic MAP for Speaker Recognition, ICASSP - IEEE Int. on Acoustics, Speech and Signal Processing, **1992**, vol. 2, pp 161-164.

32. NAIK, J. Field Trial of a Speaker Verification Service for Caller Identity Verification in the telephone Network, IEEE 2º Workshop on Interactive voice Technology for Telecommunication (IVTTA 94), **1994**, pp 125-128.

33. JACOBS, T. & SETLUR, A. A Field Study of Performance Improvements in HMM Based Speaker Verification, IEEE 2º Workshop on Interactive voice Technology for Telecommunication (IVTTA 94), **1994**, pp 121-124.

34. PICKLES, J. O. **An Introduction to the Physiology of Hearing**, Academic Press, New York, 1988.

35. MOLLER, A. R. **Auditory Physiology**, Academic Press, New York, 1983.

36. O'SHAUGHNESSY, D. **Speech Communication: Human and Machine**, Addison-Wesley, New York, 1987.

37. ALLEN, J. B. Cochlear Modeling, IEE ASSP Mag., Set. **1985**, vol. 3, nº 3, pp 3- 29.

38. LEE, K. F. **Automatic Speech Recognitin — The Development of the SPHINX System**, Kluwer Academic Publisher, Boston, 1989.

39. BOGERT, B.; TUKEY J.; HEALY M. The Quefrency Analysis of Time Series for Echoes, Proc. Symp. on Time Series Analysis, M. Rosenblatt, Ed., Cap 15, pp 209-243, J. Wiley, New York, **1963**.

40. SCHAFER, R. W. & RABINER, L. R. Digital Representations of Speech Signals, Proceedings of IEEE, Abril **1975**, vol. 63, nº 4, pp 662-667.

41. MAKHOUL, J. Efficient Acoustic Parameter for Speaker Recognition,The Journal of the Acoustical Society of America, **1972**, vol. 51, nº 6, parte 2.

42. COLE, R. A. **Survey of the State of the Art in Human Language Technology**, "Center for Spoken Languagem Understanding, Oregon Graduate Institute", Publicações Técnicas, Nov. 1995.

43. BAHL, L. R. et alii. Speech Recognition with Continuous-Parameter Hidden Markov

Models, IBM Thomas J. Watson Research Center, NY, Nov. **1987,** Publicações Técnicas.

44. ANDERBERG, M. R. **Cluster Analysis For Applications**, Academic Press, Nova York, 1973.
45. WILPON, J. G. & RABINER, L. R. A Modified K-Means Clustering Algorithm for Use in Isolated Work Recognition, IEEE Transaction on Acoustics, Speech and Signal Processing, Jun. **1985**, vol. ASSP-33, nº 3, pp. 587-594.

46. LINDE, Y; BUZO, A.; GRAY, R. M. An Algorithm For Vector Quantizer Design, IEEE Transactions On Communications, Jan. **1980**, vol. 28, nº 1, pp. 84-95.

47. LIPORACE, L. A. Maximum Likelihood Estimation for Multivariate Observations of Markov Source, IEEE Trans. Informations Theory, **1982**, vol. IT-28, nº 5, pp. 729-734.

48. JUANG, B. H. Maximum Likelihood Estimation for Mixture Multivariate Stochastic Observations of Markov Chains, AT&T Tech. Journal, Jul-Ago **1985**, vol. 64, nº 6, pp. 1235-1249.

49. JUANG, B. H. et alii. Maximum Likelihood Estimation for Multivariate Mixture Observations of Markov Chains, IEEE Trans. Informations Theory, Mar. **1986**, vol. IT-32, nº 2, pp. 307-309.

50. RENALS, S. & MORGAN, N. Connections Probability estimation in HMM Speech Recognition, Technical Research of Internationa Computer Science Institute, Dez. **1992**, TR-92-081.

51. GALES, M. J. F. & YOUNG, S. J. The Theory of Segmental Hidden Markov Models, Technical Research of Cambridge University Engineering Department, Jun. **1993**, CUED/ F-INFENG/ TR 133.

52. FALAVIGNA, D. Comparison of Different HMM Based Methods for Speaker Verification, IRST - Istituto per la Ricerca Scientifica e Tecnologica, I-38050 Povo (Trento), **1994**, Italy.

53. ANGELINI, B. et alii. Speaker Independent Continuous Speech Recognition Using an Acoustic-Phonetic Italian Corpus, IRST - Istituto per la Ricerca Scientifica e Tecnologica, I-38050 Povo (Trento), **1994**, Italy.

54. BAUM, L. E. et alii. A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains, Annals of Mathematical Statistics, **1970**, vol. 41, pp. 164-171.

55. FORNEY Jr., G. D. The Viterbi Algorithm, Proceedings of the IEEE, Mar. **1973**, vol.61, nº 3, pp. 268-278.

56. RABINER, L. R. et alii. Recognition of Isolated Digits Using Hidden Markov Models with Continuous Mixture Densities, AT&T Tech. J. Jul.-Ago. **1985**, 64(6):1211-1234.

57. LEVINSON, S. E. et alii. An Introduction to the Application Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition, Bell system Tech. J., Abr. 1983, 62(4), pp. 1035-1074.

58. RABINER, L. R. A Tutorial on Hidden Markov Modls and Selected Applications in Speech Recognition, Proc. IEEE, Fev. **1989**, Vol. 77 (2):257-286.

59. DEMPSTER, A. P. et alii. Maximum Likelihood from Incomplete Data via the EM Algorithm, J. Roy. Stat. Soc, **1977**, Vol. 39, pp. 1-38.

60. SANTOS, S. C. B. **Poder Discriminatório dos Fonemas Nasalados na Verificação Automática de Locutores**, Tese de Mestrado, IME, Dez. 1989.