LISELENE DE ABREU BORGES

SISTEMAS DE ADAPTAÇÃO AO LOCUTOR UTILIZANDO AUTOVOZES

Dissertação apresentada à Escola
Politécnica da Universidade de
São Paulo para a obtenção do Título
de Mestre em Engenharia.

SÃO PAULO
2001

LISELENE DE ABREU BORGES

SISTEMAS DE ADAPTAÇÃO AO LOCUTOR UTILIZANDO AUTOVOZES

Dissertação apresentada à Escola
Politécnica da Universidade de
São Paulo para a obtenção do Título
de Mestre em Engenharia.

Área de concentração:
Engenharia Elétrica

Orientador:
Professor Livre-Docente
Miguel Arjona Ramírez

SÃO PAULO
2001

FICHA CATALOGRÁFICA

BORGES, LISELENE DE ABREU

SISTEMAS DE ADAPTAÇÃO AO LOCUTOR UTILIZANDO AUTOVOZES
MUNICÍPIO DE SÃO PAULO, SÃO PAULO. SÃO PAULO 2001.
65p.

DISSERTAÇÃO (MESTRADO) – ESCOLA POLITÉCNICA DA UNIVERSIDADE DE
SÃO PAULO. DEPARTAMENTO DE ENGENHARIA ELÉTRICA.

1.RECONHECIMENTO DE VOZ 2.ADAPTAÇÃO AO LOCUTOR 3.AUTOVOZES
I. UNIVERSIDADE DE SÃO PAULO. ESCOLA POLITÉCNICA. DEPARTAMENTO DE
ENGENHARIA ELÉTRICA – SISTEMAS ELETRÔNICOS
II. t.

Dedico este trabalho ao meu pai e
à minha mãe pelo incentivo,
apoio e compreensão.

AGRADECIMENTOS

Ao Prof. Dr. Miguel Arjona Ramirez, meu orientador, pela orientação prestada.

À minha amiga Bibliotecária Letícia Prestes Caldas pela hospedagem, compreensão, amizade e paciência.

Ao Prof. Dr. Rubem Dutra Ribeiro Fagundes pelo incentivo e apoio à execução do mestrado.

Ao Eng^o. Gustavo Nascimento da Cunha, pelo seu apoio, carinho e paciência.

Ao Eng^o. Daniel Barros Júnior pelas trocas de informações.

À Prof. Dr^a. Denise Consonni pelo trabalho feito junto ao programa PAE.

Ao colega de PAE, Eng^o. Marcelo Bender Perotoni pela sua ajuda.

Àquelas pessoas que contribuíram gentilmente cedendo amostra de suas vozes

RESUMO

O presente trabalho descreve duas técnicas de adaptação ao locutor para sistemas de reconhecimento de voz utilizando um volume de dados de adaptação reduzido. Regressão Linear de Máxima Verossimilhança (MLLR) e Autovozes são as técnicas trabalhadas. Ambas atualizam as médias das Gaussianas dos modelos ocultos de Markov (HMM). A técnica MLLR estima um grupo de transformações lineares para os parâmetros das médias das Gaussianas do sistema. A técnica de Autovozes baseia-se no conhecimento prévio das variações entre locutores. Para obtermos o conhecimento prévio, que está contido nas autovozes, utiliza-se a análise em componentes principais (PCA). Fizemos os testes de adaptação das médias em um sistema de reconhecimento de voz de palavras isoladas e de vocabulário restrito. Contando com um volume grande de dados de adaptação (mais de 70% das palavras do vocabulário) a técnica de autovozes não apresentou resultados expressivos com relação aos que a técnica MLLR apresentou. Agora, quando o volume de dados reduzido (menos de 15% das palavras do vocabulário) a técnica de Autovozes apresentou-se superior à MLLR.

ABSTRACT

This present work describe two speaker adaptation technique, using a small amount of adaptation data, for a speech recognition system. These techniques are Maximum Likelihood Linear Regression (MLLR) and Eigenvoices. Both re-estimates the mean of a continuous density Hidden Markov Model system. MLLR technique estimates a set of linear transformations for mean parameter of a Gaussian system. The eigenvoice technique is based on a previous knowledge about speaker variation. For obtaining this previous knowledge, that are retained in eigenvoices, it necessary to apply principal component analysis (PCA). We make adaptation tests over an isolated word recognition system, restrict vocabulary. If a large amount of adaptation data is available (up to 70% of all vocabulary) Eigenvoices technique does not appear to be a good implementation if compared with MLLR the technique. Now, when just a small amount of adaptation data is available (less than 15 % of all vocabulary), Eigenvoices technique get better results than MLLR technique.

SUMÁRIO

Resumo

Abstract

Sumário

Lista de figuras

Lista de tabelas

1	Introdução.....	1
2	Princípios gerais de reconhecimento de voz	3
2.1	Introdução	3
2.2	Características do sinal de voz.....	3
2.3	Extração de parâmetros	4
2.3.1	Coeficientes de frequência Mel Cepstrais	4
2.3.2	Coeficiente de Energia.....	6
2.4	O reconhecedor de voz de palavras isoladas.....	6
2.5	Modelos Ocultos de Markov.....	8
2.5.1	Probabilidades de saída.....	11
2.5.2	Estimativa de Baum-Welch.....	12
2.6	Conclusão.....	15
3	Sistemas de adaptação ao locutor	16
3.1	Introdução	16
3.2	Princípios de adaptação ao locutor.....	16
3.3	Tipos de adaptação ao locutor	17
3.4	Parâmetros de um sistema de adaptação ao locutor	18
3.5	Estimativa de Máxima Verossimilhança.....	19
3.6	Conclusão.....	23
4	Regressão linear de máxima verossimilhança.....	24
4.1	Introdução	24
4.2	Idéia Geral.....	24

4.3	Definições da técnica MLLR.....	25
4.4	Classes de regressão na adaptação MLLR.....	27
4.5	Conclusão.....	28
5	Autovozes.....	29
5.1	Introdução	29
5.2	Noções básicas.....	29
5.2.1	Análise em componentes principais.....	30
5.3	Estimação do Autoespaço	31
5.3.1	Definição do número de componentes de autovozes	32
5.4	Técnicas de estimação dos Autovozes	33
5.4.1	Projeção	33
5.4.2	Auto-decomposição de máxima verossimilhança (MLED)	35
5.5	Conclusão.....	38
6	Metodologia	40
6.1	Introdução	40
6.2	Definições	40
6.3	Ferramenta HTK.....	41
6.3.1	Análise do sinal	42
6.3.2	Treinamento do sistema.....	43
6.3.3	Medidas de desempenho	45
6.4	Adaptação MLLR.....	46
6.5	Adaptação Autovozes.....	48
6.6	Conclusão.....	50
7	Avaliação Experimental	51
7.1	Introdução	51
7.2	Experimentos	51
7.2.1	Sistema adaptado utilizando MLLR	51
7.2.2	Sistema adaptado utilizando Autovozes	54
7.3	Conclusões	59
8	Conclusão	60

9	Referências Bibliográficas	62
Anexo A –	Estimativa da matriz de transformação	65

LISTA DE FIGURAS

Figura 1.1 - Sistemas dependente de locutor adaptado combinando informações gerais de voz de um sistema independente e as informações específicas do novo locutor obtidas através dos dados de adaptação.....	1
Figura 2.1- Gráfico de relação entre frequência física e frequência percebida.....	5
Figura 2.2 - Banco de filtros ao longo da escala Mel	6
Figura 2.3 - Reconhecedor de voz de palavras isoladas	7
Figura 2.4 - Modelo HMM de 5 estados.....	9
Figura 3.1 – Processo de adaptação ao locutor de um sistema de reconhecedor de voz independente de locutor.....	16
Figura 3.2 - Ilustração da adaptação por mapeamento espectral.....	17
Figura 3.3 - Modo de adaptação em bloco	18
Figura 3.4 - Modo de adaptação instantâneo.....	19
Figura 3.5 - Modo de adaptação incremental.....	19
Figura 4.1 - Ilustração das classes de regressão utilizadas pela técnica MLLR.....	28
Figura 5.1 - Ilustração da projeção do novo locutor no espaço gerado pelas autovozes.	33
Figura 7.1 - Curva que relaciona o número de palavras de adaptação com a taxa de reconhecimento adaptado resultante, utilizando a técnica MLLR.....	53
Figura 7.2 – Variação cumulativa das autovozes.....	55
Figura 7.3 - Taxa de reconhecimento para a técnica Autovozes variando o número de componentes de autovozes para os métodos MLED e projeção.....	56
Figura 7.4 - Taxa de reconhecimento independente e adaptado para diferentes N ^{os} de locutores Base do modelo SI.....	56
Figura 7.5 – Resultado da adaptação Autovozes variando o volume de dados de adaptação, para as 3 e 6 primeiras componentes de autovozes.....	59

LISTA DE TABELAS

Tabela 6.1- Vocabulário do sistema de reconhecimento adotado.....	41
Tabela 6.2 - Configuração do sistema de reconhecimento adotado.....	41
Tabela 7.1 - Resultados da adaptação utilizando a técnica MLLR e uma palavra na adaptação.....	52
Tabela 7.2 - Resultados da adaptação MLLR variando o n° de palavras diferentes na adaptação.....	53
Tabela 7.3 - Resultados da adaptação Autovozes utilizando todas as palavras do vocabulário , para diferentes dimensões do subespaço das Autovozes (K).....	54
Tabela 7.4 - Resultados da adaptação Autovozes, utilizando uma palavra como dados de adaptação, para as 3, 6 e 9 primeiras componentes de autovozes	58
Tabela 7.5 – Resultado da adaptação Autovozes variando o número de palavras diferentes de adaptação para as 3 e 6 primeiras componentes de autovozes	58

1 INTRODUÇÃO

Figura 1.1 - Sistemas dependente de locutor adaptado combinando informações gerais de voz de um sistema independente e as informações específicas do novo locutor obtidas através dos dados de adaptação

O sistema de reconhecimento de voz dependente de locutor (SD), normalmente, apresenta o melhor desempenho em aplicações que envolvem um único locutor. Para isso, no entanto, é necessário se dispor de um volume de dados de treinamento suficiente para treinar o sistema de forma robusta. Porém uma vez que o tamanho do vocabulário aumente, torna-se cada vez mais difícil dispor do volume de dados necessário. Uma solução é treinar um sistema independente de locutor (SI), utilizando dados de diferentes locutores. Todavia, experimentos mostram que a utilização deste tipo de sistema em geral apresenta menor desempenho com relação ao SD. Uma solução para este problema é a utilização de um sistema de adaptação ao locutor. Um sistema adaptado, Figura 1.1, não é um SD treinado por completo, mas sim um sistema que combina o conhecimento geral dado por um sistema SI com informações mais específicas do novo locutor, obtidas através de seus dados de adaptação.

Os sistemas de adaptação ao locutor estão sendo cada vez mais empregados. Pois eles apresentam uma resposta desejável quando não se dispõe de um volume grande de locutores e locuções para se efetuar um bom treinamento do sistema. No presente trabalho, serão abordadas duas técnicas em particular, a técnica de regressão linear de

máxima verossimilhança (MLLR) e a técnica de Autovozes. Ambas as técnicas modificam os parâmetros dos modelos previamente treinados. Será mostrado, através dos experimentos, qual das técnicas é a mais apropriada para o nosso sistema proposto, que é de reconhecimento de voz, de palavras isoladas e vocabulário restrito.

A técnica MLLR gera as médias adaptadas multiplicando o vetor de médias originais por uma matriz de transformação. A técnica de Autovozes baseia-se no conhecimento prévio das variações entre locutores para se estimar os novos modelos. Para obtermos o conhecimento prévio, contidos nas autovozes, utiliza-se a análise em componentes principais (PCA).

Nesta dissertação será apresentado primeiramente um sistema de reconhecimento de voz de palavras isoladas. Após haverá um capítulo dedicado aos sistemas de adaptação ao locutor, onde serão apresentados os diversos tipos de adaptação. Serão detalhadas, em dois capítulos subseqüentes, as técnicas utilizadas, MLLR e Autovozes. Em seguida haverá um capítulo dedicado à metodologia de trabalho utilizada para a extração dos resultados, que estarão contidos num capítulo à parte e no capítulo final serão apresentadas as conclusões.

2 PRINCÍPIOS GERAIS DE RECONHECIMENTO DE VOZ

2.1 Introdução

O propósito deste capítulo é descrever as principais tarefas necessárias para que se desenvolva um sistema de reconhecimento de voz de palavras isoladas. Primeiramente serão apresentadas as características do sinal de voz. Após ter-se-á extração dos parâmetros característicos do sinal. E por fim será introduzida a idéia do sistema de reconhecimento de voz de palavras isoladas. Em [1] [2] encontram-se interpretações mais didáticas para o uso dos modelos ocultos de Markov, enquanto [4] faz apresentação mais completa sobre a produção do sinal de voz e a extração de parâmetros. Na referência [3] a ferramenta de modelagem de modelos ocultos de Markov HTK é apresentada.

2.2 Características do sinal de voz

Um reconhecedor de voz é, basicamente, um comparador de padrões acústicos de voz associados a uma seqüência de símbolos que codificam alguma mensagem. Quando se fazem comparações de padrões, espera-se que o sinal de entrada a ser comparado não apresente muitas variações espúrias, a fim de facilitar o processo. Contudo, os sinais de voz não são tão bem comportados como gostaríamos. Eles apresentam diversas variações, que podem ser classificadas em duas classes: variações interlocutores e intralocutores.

Um mesmo locutor pode apresentar variações na pronúncia do seu sinal de voz. Estas variações podem ser originadas de diversas fontes de uma locução. O contexto da frase, o estado emocional do locutor, a velocidade com que enuncia a frase, a dicção e o grau de clareza com que o locutor pronuncia as palavras. Estes tipos de variações são as chamadas variações intralocutor, pois se refere às variações inerentes a um locutor apenas.

Existem também as variações associadas às diferenças fisiológicas entre locutores, como sexo e idade e às diferenças culturais, como o sotaque. Estas são as chamadas variações interlocutores, pois se referem às variações entre os diversos locutores.

2.3 Extração de parâmetros

O primeiro contato com sinal de voz é através de sua aquisição [4] utilizando um microfone. Após, o sinal deve ser submetido a um filtro de pré-ênfase, buscando amenizar os efeitos da radiação dos lábios do locutor.

Na seqüência de tarefas de extração de parâmetros [16], o sinal deve ser dividido em janelas de mesma duração, utilizando normalmente a janela de Hamming, possibilitando assim considerar o sinal estacionário neste intervalo de tempo. De cada segmento de sinal extrai-se a sua representação paramétrica, que passa a ser a observação para os modelos de palavras. Assume-se que o conjunto de todas as observações representa fielmente o sinal de voz original. Para o presente trabalho, as representações paramétricas a serem extraídos são os coeficientes de frequência mel cepstrais (MFCC) e o de energia (E) de cada janela.

2.3.1 Coeficientes de frequência Mel Cepstrais

A escala Mel foi criada a partir de um estudo que relacionou as frequências físicas, que são medidas em Hz, e as frequências percebidas pelo ouvido humano, que são medidas em Mel. No gráfico da Figura 2.1, pode-se verificar a correspondência entre mel e Hz, que matematicamente pode ser expressa pela equação abaixo:

$$f_{mel}(f_{Hz}) = 2595 \log_{10} \left(1 + \frac{f_{Hz}}{700} \right) \quad (2.1)$$

Figura 2.1- Gráfico de relação entre frequência física e frequência percebida

Para o cálculo dos MFCC, são tomados os valores da magnitude do cálculo da transformada de Fourier do segmento de sinal de voz (segmento delimitado pela janela de hamming). Estes coeficientes de magnitude são extraídos da correlação destes coeficientes com cada um dos filtros triangulares. Aqui, extrair significa que as magnitudes dos coeficientes da Fourier são multiplicadas pelo ganho do filtro correspondente e estes resultados são acumulados. Então, cada extração contém uma soma ponderada que representa a magnitude do espectro daquele canal do banco de filtros, Figura 2.2. Os coeficientes MFCC são calculados, Equação (2.2), aplicando a ele transformada discreta do co-seno (DCT) ao logaritmo das magnitudes do banco de filtros, m_k . O método de cálculo dos MFCCs foi proposto por [5] em 1980.

$$MFCC(i) = \sqrt{\frac{2}{P} \sum_{k=1}^P m_k \cos\left(\frac{\pi i}{P}(k - 0,5)\right)} \quad i = 1, 2, \dots, M \quad (2.2)$$

onde M é o número de coeficientes Mel-cepstrais desejados, m_k , $k=1, 2, \dots, P$, representa o logaritmo de energia da saída do k -ésimo filtro e P é o número de canais do banco de filtros.

Figura 2.2 - Banco de filtros ao longo da escala Mel

2.3.2 Coeficiente de Energia

Para completar o conjunto de parâmetros que serão extraídos do sinal é necessária uma explicação sobre o coeficiente de energia. O coeficiente de energia nada mais é que o logaritmo da energia do sinal dentro do período da janela. Ele é calculado da seguinte forma:

$$E = \log_{10} \sum_{n=1}^N x_n^2 \cdot w_n \quad (2.3)$$

onde x_n são as amostras do sinal de voz e w_n é uma janela de Hamming de comprimento N .

2.4 O reconhecedor de voz de palavras isoladas

Um sistema de reconhecimento de voz de palavras isoladas é apresentado na Figura 2.3. A principal tarefa do reconhecedor de voz de palavras isoladas é escolher, dentre um grupo de modelos previamente treinado, qual deles representa melhor o sinal de voz pronunciado pelo locutor.

Como foi visto anteriormente, o sinal de voz a ser reconhecido deve ser primeiramente parametrizado (no caso os parâmetros são os MFCC mais o de energia). Estes

parâmetros são as chamadas observações. Então o sinal de voz pronunciado por um locutor pode ser representado pela seqüência de observações $\mathbf{O}=\{\mathbf{o}_1,\mathbf{o}_2,...,\mathbf{o}_T\}$, onde $\mathbf{o}_t = [\text{MFCC1}_t \text{MFCC2}_t \dots \text{MFCC12}_t E_t]$ é a observação correspondente ao tempo t .

Dados os N modelos de um sistema de reconhecimento de voz, λ_i para $i=1,2,...,N$, a tarefa do reconhecedor de voz de palavras isoladas, é de determinar, dentre os N modelos, qual deles representa melhor a seqüência de observações $\mathbf{O}$. Para isso é necessário se calcular a probabilidade de cada um dos modelos serem gerados pela seqüência de observações $\mathbf{O}$, ou seja:

$$P(\lambda_i | \mathbf{O}) \quad (2.4)$$

A palavra reconhecida é indicada pelo argumento da maior verossimilhança do seu modelo dadas as observações, como mostra a Equação (2.5).

Figura 2.3 - Reconhecedor de voz de palavras isoladas

$$i = \arg \max_{1 \leq i \leq N} \{P(\lambda_i | \mathbf{O})\} \quad (2.5)$$

Utilizando a formula de Bayes [18], a Equação (2.4) pode ser reescrita como:

$$P(\lambda_i | \mathbf{O}) = \frac{P(\mathbf{O} | \lambda_i)P(\lambda_i)}{P(\mathbf{O})} \quad (2.6)$$

Então, para um dado grupo de probabilidades iniciais $P(\lambda_i)$, o cálculo da probabilidade de ocorrer cada palavra, dada a seqüência de observação $\mathbf{O}$, depende somente do cálculo de $P(\mathbf{O}|\lambda_i)$. Como a seqüência de observações $\mathbf{O}$ normalmente possui um número de termos $\mathbf{O} = \{o_1 o_2 \dots o_T\}$, o cálculo direto desta probabilidade condicional conjunta, $P(o_1, o_2, \dots, o_T | \lambda_i)$, é impraticável. Foi procurando solucionar este problema que se pensou na utilização dos modelos ocultos de Markov para a modelagem de um sistema de reconhecimento de voz.

Nos sistemas de reconhecimento de voz de palavras isoladas, cada palavra é representada por um modelo oculto de Markov (HMM).

2.5 Modelos Ocultos de Markov

Um HMM é uma máquina de estados finita, que muda seu estado atual i para o estado futuro j a cada unidade de tempo t e associado a esse estado j no instante t , um vetor de observações $\mathbf{o}_t$ é gerado, com densidade de probabilidade de saída $b_j(\mathbf{o}_t)$. Os estados iniciais e finais, que são estados considerados não-emitentes, não possuem probabilidade de saída associada¹. Além do mais a transição de um estado i para j é também probabilística e por sua vez comandada pela probabilidade discreta $a_{ij} = P(q_t = j | q_{t-1} = i)$ $1 \leq i, j \leq N$, que é representada abaixo pela matriz de transição A :

$$a_{ij} = P(q_t = j | q_{t-1} = i) \quad 1 \leq i, j \leq N \quad (2.7)$$

¹ Não ter probabilidade de saída associada implica que estes estados não representam nenhum som, eles apenas demarcam o início e o fim do modelo.

onde q_t é o estado no tempo t , com $a_{ij} \geq 0$ e $\sum_{j=1}^N a_{ij} = 1$, pois a_{ij} , $j=1,2,\dots,N$ representam todas as possibilidades de transição a partir do estado i , onde N é o número de estados do modelo. Um HMM é graficamente apresentado na Figura 2.4 e analiticamente apresentado pela Equação (2.8).

Figura 2.4 - Modelo HMM de 5 estados

$$\lambda = (\mathbf{A}, \mathbf{B}, \boldsymbol{\pi}) \quad (2.8)$$

onde $\mathbf{A}$ é a matriz de transição de estados, $\mathbf{B}$ é a matriz de distribuição de saída e $\boldsymbol{\pi}$ é o vetor de probabilidade inicial do modelo

$$\pi_i = P(q_1 = i) \quad \text{para } 1 \leq i \leq N \quad (2.9)$$

Normalmente $\sum_i^N \pi_i = 1$.

A probabilidade da sequência de observações $\mathbf{O}$ ser gerada por um HMM através de uma sequência de estados fixa $\mathbf{q}=(q_1,\dots,q_T)$ é dada por:

$$P(\mathbf{O} | \mathbf{q}, \lambda) = \prod_{t=1}^T P(o_t | q_t, \lambda) \quad (2.10)$$

assumindo independência estatística[19] das observações, então teremos:

$$P(\mathbf{O} \mid \mathbf{q}, \lambda) = b_{q_1}(\mathbf{o}_1) \cdot b_{q_2}(\mathbf{o}_2) \cdots b_{q_T}(\mathbf{o}_T) \quad (2.11)$$

Por sua vez, a probabilidade de ocorrência da seqüência de estados $\mathbf{q}$ é dada por:

$$P(\mathbf{q} \mid \lambda) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \cdots a_{q_{T-1} q_T} \quad (2.12)$$

A probabilidade conjunta de $\mathbf{O}$ e $\mathbf{q}$, ou seja, a probabilidade de $\mathbf{O}$ e $\mathbf{q}$ ocorrerem simultaneamente é dada por:

$$P(\mathbf{O}, \mathbf{q} \mid \lambda) = P(\mathbf{O} \mid \mathbf{q}, \lambda) P(\mathbf{q} \mid \lambda) \quad (2.13)$$

A probabilidade de $\mathbf{O}$ (dados o modelo) é obtida pela soma das probabilidades conjuntas sobre todos as seqüências de estados possíveis, isto é:

$$P(\mathbf{O} \mid \lambda) = \sum_{\text{todos } \mathbf{q}} P(\mathbf{O} \mid \mathbf{q}, \lambda) P(\mathbf{q} \mid \lambda) \quad (2.14)$$

$$= \sum_{q_1, q_2, \dots, q_T} \pi_{q_1} b_{q_1}(\mathbf{o}_1) a_{q_1 q_2} b_{q_2}(\mathbf{o}_2) \cdots a_{q_{T-1} q_T} b_{q_T}(\mathbf{o}_T) \quad (2.15)$$

Para o cálculo da Equação (2.15), utiliza-se um processo de recorrência, para efetuar este cálculo eficientemente. Este processo é conhecido como o algoritmo de Viterbi [1]. Em todos os cálculos acima assumiu-se que os parâmetros do modelo já foram previamente treinados, ou seja, que as probabilidades de transição, a_{ij} , e de saída, $b_j(\mathbf{o}_t)$, já são conhecidas.

No processo de treinamento dos modelos, é necessário um grupo de diferentes exemplos de sinais de voz que correspondam a cada dada palavra, para que se possa estimar de maneira eficiente e robusta cada modelo do sistema.

2.5.1 Probabilidades de saída

Para cada estado do modelo HMM que seja emitente, uma densidade de probabilidade de saída está associada. Para o caso dos HMM, do tipo contínuo, estas densidades de probabilidade são misturas de gaussianas multivariáveis, que são descritas pelas equações abaixo:

$$b_j(\mathbf{o}_t) = \sum_{m=1}^M c_{jm} N(\mathbf{o}_t; \boldsymbol{\mu}_{jm}; \boldsymbol{\Sigma}_{jm}) \quad (2.16)$$

onde M é o número de misturas de gaussianas por estado, c_{jm} é a ponderação de cada mistura, satisfazendo $\sum_{m=1}^M c_{jm} = 1$, para cada estado $2 \leq j \leq N-1$, onde N é o número de estados do modelo.

$$N(\mathbf{o}; \boldsymbol{\mu}; \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^E |\boldsymbol{\Sigma}|}} e^{-\frac{1}{2}(\mathbf{o}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{o}-\boldsymbol{\mu})} \quad (2.17)$$

onde $\boldsymbol{\mu}$ é uma distribuição gaussiana de o vetor das médias e $\boldsymbol{\Sigma}_{jm}$ a matriz de covariância da mistura m do estado j , M é o número de misturas de gaussianas e c_{jm} é o peso de cada mistura, satisfazendo $\sum_{m=1}^M c_{jm} = 1$, para cada estado $2 \leq j \leq N-1$, onde N é o número de estados do modelo.

2.5.2 Estimativa de Baum-Welch

Para se determinar os parâmetros de um modelo HMM, primeiramente é necessário que se faça uma previsão do que ele deve ser. No caso, deve-se fazer uma inicialização do modelo. Uma vez feito isso, uma estimativa mais robusta dos parâmetros pode ser obtida utilizando a re-estimativa de Baum-Welch [1].

Essencialmente o que se quer fazer é uma estimativa das médias e das covariâncias do modelo HMM, onde, para o caso de uma única mistura de gaussiana multivariável por estado, tem-se:

$$b_j(\mathbf{o}_t) = \frac{1}{\sqrt{(2\pi)^E |\boldsymbol{\Sigma}_j|}} e^{-\frac{1}{2}(\mathbf{o}_t - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1} (\mathbf{o}_t - \boldsymbol{\mu}_j)} \quad (2.18)$$

Se houvesse somente um estado j no modelo HMM, a estimativa dos parâmetros seria fácil. As estimações de máxima verossimilhança de $\boldsymbol{\mu}_j$ e $\boldsymbol{\Sigma}_j$ seriam simplesmente dadas em função de médias, ou seja:

$$\hat{\boldsymbol{\mu}}_j = \frac{1}{T} \sum_{t=1}^T \mathbf{o}_t \quad (2.19)$$

$$\hat{\boldsymbol{\Sigma}}_j = \frac{1}{T} \sum_{t=1}^T (\mathbf{o}_t - \boldsymbol{\mu}_j)(\mathbf{o}_t - \boldsymbol{\mu}_j)^T \quad (2.20)$$

Na prática, porém não é tão simples assim, a começar por que sempre irão existir múltiplos estados. E também não há nenhuma ligação direta entre as observações e os estados individuais, visto que a sequência de estados é oculta.

Uma vez que a total verossimilhança de cada sequência de observações é baseada num somatório sobre todas as sequências de estados possíveis, cada vetor de observações $\mathbf{o}_t$ contribui para o cálculo de máxima verossimilhança dos valores dos parâmetros de cada

estado j . Isto é, ao invés de se dizer que cada vetor de observação pertence a um estado em especial, diz-se a priori que cada observação pertence a todos os estados com uma certa proporção associada de verossimilhança. Portanto, se $\gamma_j(t)$ é a probabilidade de ocupação, sendo definida como a probabilidade de a observação no tempo t pertencer ao estado j , as novas estimações de média e variância são as seguintes:

$$\hat{\boldsymbol{\mu}}_j = \frac{\sum_{t=1}^T \gamma_j(t) \mathbf{o}_t}{\sum_{t=1}^T \gamma_j(t)} \quad (2.21)$$

$$\hat{\boldsymbol{\Sigma}}_j = \frac{\sum_{t=1}^T \gamma_j(t) (\mathbf{o}_t - \boldsymbol{\mu}_j)(\mathbf{o}_t - \boldsymbol{\mu}_j)'}{\sum_{t=1}^T \gamma_j(t)} \quad (2.22)$$

onde o somatório sobre todas as probabilidade de ocupação, $\sum_{t=1}^T \gamma_j(t)$, é incluído a fim de se efetuar uma normalização. As Equações (2.21) e (2.22) são as fórmulas de estimativa de Baum-Welch [1]. Pode-se notar que é necessário o cálculo da probabilidade de ocupação e isto é feito utilizando-se o algoritmo *forward-backward* [1].

Considere a variável *forward*, $\alpha_i(t)$, definida como:

$$\alpha_i(t) = P(\mathbf{o}_1, \dots, \mathbf{o}_t, q_t = i \mid \lambda) \quad (2.23)$$

ou seja, $\alpha_j(t)$ é a probabilidade de ocorrer a seqüência de observações parcial, $\mathbf{o}_1, \dots, \mathbf{o}_t$, e o estado j no tempo t , dado o modelo λ . Podemos resolver a equação de $\alpha_j(t)$ através dos seguintes passos:

1) inicialização

$$\alpha_i(1) = \pi_i b_i(\mathbf{o}_1), \quad 1 \leq i \leq N \quad (2.24)$$

2)recorrência

$$\alpha_j(t+1) = \left[\sum_{i=1}^N \alpha_i(t) a_{ij} \right] b_j(\mathbf{o}_{t+1}), \quad 1 \leq t \leq T-1$$

$$1 \leq j \leq N \quad (2.25)$$

3)finalização

$$P(\mathbf{O} | \lambda) = \sum_{i=1}^N \alpha_i(T) \quad (2.26)$$

De maneira similar, vamos agora definir a variável *backward*, $\beta_i(t)$, como sendo:

$$\beta_i(t) = P(\mathbf{o}_{t+1}, \dots, \mathbf{o}_T | q_t = i, \lambda) \quad (2.27)$$

ou seja, $\beta_i(t)$ é a probabilidade de ocorrer a sequência de observações parciais de $t+1$ até o fim, dado que estado no tempo t seja igual a i e que o modelo seja λ . Analogamente podemos resolver $\beta_i(t)$ através dos seguintes passos:

1)inicialização

$$\beta_i(T) = 1, \quad 1 \leq i \leq N \quad (2.28)$$

2)recorrência

$$\beta_i(t) = \sum_{j=1}^N a_{ij} b_j(\mathbf{o}_{t+1}) \beta_j(t+1), \quad t = T-1, T-2, \dots, 1$$

$$1 \leq j \leq N \quad (2.29)$$

Por fim, a probabilidade de ocupação por estado pode ser eficientemente calculada utilizando as variáveis *forward* e *backward* definidas anteriormente, então teremos:

$$\gamma_j(t) = P(q_t = j | \mathbf{O}, \lambda) \quad (2.30)$$

$$= \frac{P(q_t = j, \mathbf{O} | \lambda)}{P(\mathbf{O} | \lambda)} = \frac{\alpha_j(t) \beta_j(t)}{\sum_{j=1}^N \alpha_j(t) \beta_j(t)} \quad (2.31)$$

2.6 Conclusão

Neste capítulo falamos sobre os sistemas de reconhecimento de voz. Foram mostradas as características de um sinal de voz e como são extraídos os parâmetros a serem utilizados. E por fim detalhamos o tipo de sistema que iremos utilizar na implementação do projeto, o sistema de reconhecimento de voz de palavras isoladas, detalhando o modelo oculto de Markov e como são estimados seus parâmetros. No próximo capítulo introduziremos as idéias básicas dos sistemas de adaptação.

3 SISTEMAS DE ADAPTAÇÃO AO LOCUTOR

3.1 Introdução

Neste capítulo vamos descrever as técnicas de adaptação ao locutor para os sistemas de reconhecimento de voz. Primeiramente vamos falar sobre os tipos de adaptação possíveis de se efetuar [12]. No tópico seguinte vamos falar sobre as propriedades do sistema de adaptação e por fim vamos falar da estimativa de máxima verossimilhança, que constitui a base do desenvolvimento dos métodos de adaptação propostos e estudados no presente trabalho.

3.2 Princípios de adaptação ao locutor

O sistema de reconhecimento de voz com adaptação ao locutor busca modificar o sistema independente de locutor (SI) previamente treinado, de forma que ele se aproxime do sistema dependente de locutor (SD) ideal para este novo locutor, utilizando somente um conjunto reduzido de dados na adaptação [15]. Um SD normalmente apresenta melhores resultados do que um SI. Um diagrama ilustrativo do processo de adaptação é apresentado abaixo.

Figura 3.1 – Processo de adaptação ao locutor de um sistema de reconhecedor de voz independente de locutor

3.3 Tipos de adaptação ao locutor

Existem basicamente duas maneiras de se efetuar a adaptação em um sistema de reconhecimento de voz, para sistemas baseados em HMM. Estas duas maneiras são [8] a modificação dos parâmetros de entrada do novo locutor, o chamado mapeamento espectral e a modificação dos modelos do sistema, o chamado mapeamento do modelo.

Figura 3.2 - Ilustração da adaptação por mapeamento espectral

O que se faz no **mapeamento espectral** é basicamente mapear as características espectrais do novo locutor, para um espaço previamente treinado. Isto é feito de tal maneira que a diferença entre os dados de entrada do novo locutor e do locutor de referência seja minimizada. Este método é baseado em que as variações entre locutores causam diferenças no conteúdo espectral do sinal de voz, que se refletem diretamente nos parâmetros que representam o mesmo.

Já no **mapeamento dos modelos** são feitas modificações nos parâmetros dos HMMs previamente treinados. Normalmente se faz uma re-estimativa dos valores das médias dos modelos. Mas também podem ser feitas re-estimativas da matriz de covariâncias, dos pesos das misturas de gaussianas e também dos elementos das matrizes de transição. Esta maneira de implementação divide-se em três famílias distintas, que são: a família de adaptações MAP (*Maximum a Posteriori*), a família dos métodos de agrupamentos de locutores e a família das transformações lineares.

3.4 Parâmetros de um sistema de adaptação ao locutor

Agora que já sabemos um pouco sobre as maneiras de se fazer adaptação em sistemas de reconhecimento de voz, vamos falar sobre os modos como podem ser implementadas. Pode-se implementar a adaptação de dois modos. Quando os dados de adaptação são conhecidos, estamos diante da adaptação é do tipo supervisionada. Por outro lado, quando os dados não são conhecidos então estamos diante da adaptação não-supervisionados.

Também é necessário definir quando será feita a adaptação: *online* se feita juntamente com o reconhecimento das palavras, ou *offline* se feita anteriormente ao reconhecimento das palavras.

Por fim, ainda temos que definir de que maneira será feita a adaptação: se de uma só vez, a chamada adaptação em bloco, Figura 3.3, se ao mesmo tempo, em paralelo ao reconhecimento do sistema, a chamada adaptação incremental, Figura 3.5, ou se ambos, o reconhecimento e a adaptação serão efetuados ao mesmo tempo, a chamada adaptação instantânea, Figura 3.4.

Figura 3.3 - Modo de adaptação em bloco

Figura 3.4 - Modo de adaptação instantâneo

Figura 3.5 - Modo de adaptação incremental

3.5 Estimativa de Máxima Verossimilhança

A adaptação em um sistema de reconhecimento de voz é considerada uma etapa de treinamento adicional. O que se busca é maximizar a verossimilhança de novas observações, dado o HMM. As equações que se utilizam para buscar estas estimativas são semelhantes às utilizadas durante as etapas de treinamento padrão.

Um locutor é considerado uma máquina de produzir locuções. Dada nossa produção vocal, o que se procura é otimizar os HMMs atuais para obter a máxima verossimilhança das observações [14]. Dado o HMM atual, assumindo que os dados do novo locutor são desconhecidos, objetiva-se otimizar o HMM. Este processo de adaptação pode ser equacionado por:

$$\hat{\lambda} = \arg \max_{\lambda \in \Omega} P(\mathbf{O} | \lambda) \quad (3.1)$$

onde $\mathbf{O}$ é o conjunto de observações que representa a locução de adaptação e Ω é o conjunto dos modelos do sistema. Para calcularmos o novo modelo estimado, da Equação (3.1), vamos aplicar as definições do algoritmo de Máxima verossimilhança (ML). Este algoritmo transforma a função objetiva $P(\mathbf{O}|\lambda)$ na função auxiliar de Baum, $Q(\lambda, \hat{\lambda})$, e depois maximiza a função com relação a $\hat{\lambda}$. A função Q essencialmente mede a divergência entre o modelo inicial, λ , e o modelo estimado, $\hat{\lambda}$. A função Q é definida como sendo:

$$Q(\lambda, \hat{\lambda}) = \sum_{\mathbf{q}} P(\mathbf{O}, \mathbf{q} | \lambda) \log P(\mathbf{O}, \mathbf{q} | \hat{\lambda}) \quad (3.2)$$

onde $\mathbf{q}=(q_1, \dots, q_T)$ é a sequência de estados possíveis para um dado conjunto de observação $\mathbf{O}$. Baum *et al.* provou que maximizando a função auxiliar Q conseguimos aumentar a verossimilhança, isto é:

$$\max_{\hat{\lambda}} [Q(\lambda, \hat{\lambda})] \Rightarrow P(\mathbf{O} | \hat{\lambda}) \geq P(\mathbf{O} | \lambda) \quad (3.3)$$

Estas fórmulas de re-estimação podem ser interpretadas como uma implementação do algoritmo de *Expectation-Maximization* (EM), onde o passo de *Expectation* é o cálculo

da função auxiliar de Baum Q e o passo de *Maximization* é a maximização com relação ao modelo estimado $\hat{\lambda}$.

Por simplicidade vamos derivar os cálculos considerando somente uma gaussiana por estado, uma vez que uma mistura de gaussianas pode ser expandida em um grupo de gaussianas simples em paralelo, visto que uma mistura de Gaussianas é uma soma ponderada de Gaussianas simples, como mostra a equação abaixo:

$$b_j(\mathbf{o}_t) = c_{j1}N(\mathbf{o}_t; \boldsymbol{\mu}_{j1}; \boldsymbol{\Sigma}_{j1}) + c_{j2}N(\mathbf{o}_t; \boldsymbol{\mu}_{j2}; \boldsymbol{\Sigma}_{j2}) + \dots + c_{jM}N(\mathbf{o}_t; \boldsymbol{\mu}_{jM}; \boldsymbol{\Sigma}_{jM}) \quad (3.4)$$

onde j é o estado do HMM e M é o número de misturas por estado. Vamos então reescrever o logaritmo da probabilidade como sendo:

$$\log P(\mathbf{O}, \mathbf{q} | \hat{\lambda}) = \log[\hat{a}_{q_T N} \prod_{t=1}^T \hat{a}_{q_{t-1}q_t} \hat{b}_{q_t}(\mathbf{o}_t)] \quad (3.5)$$

$$= \log \hat{a}_{q_T N} + \sum_{t=1}^T \log \hat{a}_{q_{t-1}q_t} + \sum_{t=1}^T \log \hat{b}_{q_t}(\mathbf{o}_t) \quad (3.6)$$

Substituindo agora a Equação (3.6) na função auxiliar Q, Equação (3.2), tem-se:

$$Q(\lambda, \hat{\lambda}) = \sum_{\mathbf{q}} P(\mathbf{O}, \mathbf{q} | \lambda) \left\{ \log \hat{a}_{q_T N} + \sum_t \log \hat{a}_{q_{t-1}q_t} + \sum_t \log \hat{b}_{q_t}(\mathbf{o}_t) \right\} \quad (3.7)$$

Como nos interessa somente saber os valores das médias adaptadas, fazendo $\hat{a} = a$ e reescrevemos a Equação (3.7) como:

$$Q(\lambda, \hat{\lambda}) = Q_a(\lambda, \{a_{ij}\}) + Q_b(\lambda, \hat{b}_j) \quad (3.8)$$

onde $Q_a(\lambda, \{a_{ij}\})$ é uma constante de valor:

$$Q_a(\lambda, \{a_{ij}\}) = \sum_{q \in \mathbf{q}} P(\mathbf{O}, q_t = i | \lambda) \log \hat{a}_{iN} + \sum_{q \in \mathbf{q}} \sum_{t=1}^T \sum_{j=1}^N P(\mathbf{O}, q_{t-1} = i, q_t = j | \lambda) \log \hat{a}_{ij} \quad (3.9)$$

$$Q_b(\lambda, \hat{b}_j) = \sum_{\mathbf{q} \in \mathbf{Q}} P(\mathbf{O}, \mathbf{q} | \lambda) \sum_t \log \hat{b}_j(\mathbf{o}_t) \quad (3.10)$$

com

$$\log \hat{b}_j(\mathbf{o}_t) = -\frac{n}{2} [\log(2\pi) - \log |\boldsymbol{\Sigma}_j| - (\mathbf{o}_t - \hat{\boldsymbol{\mu}}_j)^T (\boldsymbol{\Sigma}_j)^{-1} (\mathbf{o}_t - \hat{\boldsymbol{\mu}}_j)] \quad (3.11)$$

Pela definição da probabilidade total de ocupação da observação no tempo t , temos:

$$\gamma_j(t) = \frac{1}{P(\mathbf{O} | \lambda)} \sum_{\mathbf{q} \in \mathbf{Q}} P(\mathbf{O}, q_t = j | \lambda) \quad (3.12)$$

onde $\mathbf{Q}$ é o conjunto de todas as seqüências $\mathbf{q}$ possíveis. Então podemos reescrever a Equação (3.10) da seguinte forma:

$$Q_b(\lambda, \hat{b}_j) = -\frac{n}{2} P(\mathbf{O} | \lambda) \sum_t \gamma_j(t) [\log(2\pi) + \log |\boldsymbol{\Sigma}_j| + (\mathbf{o}_t - \hat{\boldsymbol{\mu}}_j)^T (\boldsymbol{\Sigma}_j)^{-1} (\mathbf{o}_t - \hat{\boldsymbol{\mu}}_j)] \quad (3.13)$$

Então a função auxiliar a ser maximizada é reescrita como sendo:

$$Q(\lambda, \hat{\lambda}) = -\frac{n}{2} P(\mathbf{O} | \lambda) \sum_j \sum_t \gamma_j(t) [\log(2\pi) + \log |\boldsymbol{\Sigma}_j| + (\mathbf{o}_t - \hat{\boldsymbol{\mu}}_j)^T (\boldsymbol{\Sigma}_j)^{-1} (\mathbf{o}_t - \hat{\boldsymbol{\mu}}_j)] \quad (3.14)$$

onde:

$\mathbf{o}_t$ é o vetor de observações;

$(\Sigma_j)^{-1}$ é a inversa da matriz de covariância do estado j .

$\hat{\mu}_j$ é o vetor de médias adaptadas estimadas para o estado j .

$\gamma_j(t)$ é a probabilidade de ocupação da observação no tempo t no estado j .

3.6 Conclusão

Neste capítulo falamos dos sistemas de adaptação ao locutor utilizados em sistemas de reconhecimento de voz e suas propriedades. Por fim apresentamos a estimativa de máxima verossimilhança. Esta estimativa é utilizada em ambas as técnicas de adaptação a serem apresentadas tendo-se estabelecido assim uma base comum de notação e de técnicas que auxiliarão nos seus desenvolvimentos.

4 REGRESSÃO LINEAR DE MÁXIMA VEROSSIMILHANÇA

4.1 Introdução

Neste capítulo vamos apresentar a técnica de adaptação de regressão linear de máxima verossimilhança (MLLR) [8][9]. Esta é uma técnica de adaptação ao locutor bem conhecida e utilizada. Primeiramente vamos descrever a técnica, explicando seu funcionamento usando os conceitos básicos apresentados no capítulo anterior. Por fim vamos apresentar o cálculo da estimativa da matriz de transformação que é aplicada aos dados.

4.2 Idéia Geral

A técnica MLLR de adaptação ao locutor, desenvolvida por [5], assume que o sistema inicial de reconhecimento de voz é independente de locutor, e que é um sistema baseado em modelos de Markov do tipo contínuo.

Esta é uma técnica de adaptação do modelo que estima um grupo de transformações lineares para os parâmetros de médias e de covariâncias, das Gaussianas do sistema HMM. O efeito destas transformações é de deslocar as componentes de média e depois as variâncias do sistema inicial fazendo com que cada estado no sistema HMM seja mais provável para se gerar os dados de adaptação.

Assume-se que a quantidade de dados de adaptação é reduzido com relação ao vocabulário do sistema, por esse motivo seria impossível adaptar todos os modelos HMM individualmente. Para solucionar este problema, são criadas as classes de regressão, onde cada classe tem uma matriz de transformação associada. Os critérios de seleção de cada classe serão explicados mais adiante.

A adaptação das probabilidades de transição e dos pesos das misturas, para casos onde se utilizam misturas de gaussianas, apresentam muito pouca interferência nos resultados finais da adaptação [9]. Já segundo Rabiner [19] a adaptação das variâncias, quando se

possui um volume muito restrito de dados de adaptação, não apresenta melhora na taxa de reconhecimento, por esse motivo optamos por adaptar somente as médias dos HMMs. Uma vez que vamos adaptar somente as médias dos HMMs, para definirmos os novos HMMs adaptados, mantemos o restante dos parâmetros, variâncias (matriz de transição) os mesmos do SI.

4.3 Definições da técnica MLLR

Dado o vetor de média estendido da distribuição de saída do estado j de um dado modelo HMM, como sendo $\boldsymbol{\mu}_j = [\omega \ \mu_1 \ \dots \ \mu_N]^T$, onde ω^2 é o termo de “offset” (normalmente $\omega = 1$) e E é a dimensão das observações. O vetor de média adaptado é então dado por:

$$\hat{\boldsymbol{\mu}}_j = \mathbf{W}_j \cdot \boldsymbol{\mu}_j \quad (4.1)$$

onde $\mathbf{W}_j$ é uma matriz de transformação linear de dimensão $(E \times (E+1))$ que otimiza a máxima verossimilhança do modelo, através da função auxiliar de Baum Q, Equação (3.14), apresentada no capítulo 3.

Se R estados $\{j_1, j_2, \dots, j_R\}$ compartilhem uma mesma matriz de transformação $\mathbf{W}_j$, maximizando a Equação 3.12, Anexo A, teremos:

$$\sum_{t=1}^T \sum_{r=1}^R \gamma_{j_r}(t) \boldsymbol{\Sigma}_{i_r}^{-1} \mathbf{o}_t \hat{\boldsymbol{\mu}}_{j_r}^T = \sum_{t=1}^T \sum_{r=1}^R \gamma_{j_r}(t) \boldsymbol{\Sigma}_{i_r}^{-1} \mathbf{W}_j \hat{\boldsymbol{\mu}}_{j_r} \hat{\boldsymbol{\mu}}_{j_r}^T \quad (4.2)$$

Reescrevendo a equação acima de uma forma mais usual, temos:

² O termo de “offset” é uma constante utilizada para efetuar um deslocamento fixo nos parâmetros a serem adaptados.

$$\sum_{t=1}^T \sum_{r=1}^R \gamma_{j_r}(t) \Sigma_{i_r}^{-1} \mathbf{o}_t \hat{\boldsymbol{\mu}}_{j_r}^T = \sum_{r=1}^R \mathbf{V}^{(r)} \mathbf{W}_j \mathbf{D}^{(r)} \quad (4.3)$$

onde $\mathbf{V}^{(r)}$, é uma matriz (N x N), é a matriz de covariância inversa da distribuição do estado ponderada por sua probabilidade de ocupação do estado,

$$\mathbf{V}^{(r)} = \sum_{t=1}^T \gamma_{j_r}(t) \Sigma_{j_r}^{-1} \quad (4.4)$$

e $\mathbf{D}^{(r)}$ é o produto interno do vetor estendido das médias:

$$\mathbf{D}^{(r)} = \hat{\boldsymbol{\mu}}_{j_r} \hat{\boldsymbol{\mu}}_{j_r}^T \quad (4.5)$$

Seja o lado esquerdo da Equação (4.3) definido como sendo $\mathbf{Z}$ e os elementos individuais de $\mathbf{V}^{(r)}$, $\mathbf{W}_j$ e $\mathbf{Z}$ definidos por $v_{ik}^{(r)}$, w_{ik} , $d_{ik}^{(r)}$ e z_{ik} , respectivamente, então o lado direito da Equação (4.3) pode ser utilizado para designar os elementos individuais de z_{ik} como sendo

$$z_{ik} = \sum_{p=1}^N \sum_{q=1}^{N+1} w_{pq} \left[\sum_{r=1}^R v_{ip}^{(r)} v_{qk}^{(r)} \right] \quad (4.6)$$

Se todas as matrizes de covariância são diagonais e sabendo que $\mathbf{D}$ é uma matriz simétrica

$$\sum_{r=1}^R v_{ip}^{(r)} d_{qk}^{(r)} = \begin{cases} \sum_{r=1}^R v_{ip}^{(r)} d_{qk}^{(r)} & \text{quando } i = p \\ 0 & \text{quando } i \neq p \end{cases} \quad (4.7)$$

$$z_{ik} = \sum_{q=1}^{N+1} w_{iq} g_{kq}^{(i)} \quad (4.8)$$

onde $g_{kq}^{(i)}$ são os elementos da matriz $\mathbf{G}^{(i)}$ de dimensão $((E+1) \times (E+1))$ dada por:

$$g_{kq}^{(i)} = \sum_{r=1}^R v_{ii}^{(r)} d_{kq}^{(r)} \quad (4.9)$$

Usando a Equação (4.8) como um sistema de equações simultâneas é gerado

$$\mathbf{w}_i^T = \mathbf{G}^{(i-1)} \mathbf{z}_i^T \quad (4.10)$$

onde $\mathbf{w}_i$ e $\mathbf{z}_i$ são as i -ésimas linhas de $\mathbf{W}_j$ e $\mathbf{Z}$ respectivamente. Resolvendo-se linha por linha teremos a matriz $\mathbf{W}_j$ completa.

Todas as derivações foram feitas considerando somente uma mistura de Gaussiana multivariável por estado. O cálculo estendido, para um número de misturas maior que um, pode ser verificado em [23].

4.4 Classes de regressão na adaptação MLLR

A classe de regressão, Figura 4.1, define grupos de misturas de componentes. Deve-se salientar que uma componente de mistura de um mesmo estado pode, em geral, estar vinculado a duas diferentes classes de regressão. Cada classe de regressão i possui a sua matriz $\mathbf{W}_j$ associada.

Para que a adaptação seja suficientemente efetiva contando, somente, com uma quantidade pequena de dados, cada matriz de regressão é associada a um número de estados de distribuição e estimada a partir da combinação destes dados. Esta idéia de matriz de regressão, gera uma transformação linear geral para todas as componentes da classe e assim, os parâmetros que não foram apresentados nos dados de treinamento,

podem ser atualizados utilizando-se a mesma matriz de regressão da classe. Por isso, até métodos não supervisionados de adaptação podem ser utilizados de maneira robusta.

As classes de regressão normalmente são definidas de acordo com o volume de dados de adaptação. Quando se utiliza classe de regressão, existem duas maneiras básicas de se definir as divisões das classes: de acordo com as características fonéticas ou de acordo com medidas de distâncias.

Figura 4.1 - Ilustração das classes de regressão utilizadas pela técnica MLLR

As divisões das classes podem ser feitas de maneira estática ou dinâmica. A divisão das classes é estática uma vez que seja definida previamente ao processo de adaptação. Por sua vez, a divisão é dinâmica quando ela pode ser definida, de acordo com o volume e o tipo dos dados no decorrer do processo de adaptação.

4.5 Conclusão

Neste capítulo apresentamos, partindo dos conceitos apresentados no capítulo anterior, a técnica de adaptação mais utilizada em sistemas de reconhecimento de voz, a técnica de adaptação MLLR. Visto que esta técnica é amplamente utilizada, ela servirá como método de referência para sistemas de adaptação. Para que com isso possamos compará-la com a técnica a ser apresentada no próximo capítulo.

5 AUTOVOZES

5.1 Introdução

Neste capítulo vamos falar sobre a técnica de adaptação de Autovozes [10][14]. Esta técnica não é tão utilizada quanto a MLLR, mas ela é muito eficiente para se fazer adaptação rápida. Primeiramente vamos dar uma visão geral da técnica, em seguida vamos falar da estimação do espaço de Autovozes, que é obtida utilizando a análise em componentes principais (PCA)[21]. Por fim, vamos falar da estimação dos coeficientes de Autovozes, apresentando as técnicas de projeção e auto decomposição de máxima verossimilhança (MLE)[10].

5.2 Noções básicas

As técnicas de Autovozes foram apresentadas em [8] e [14], mas elas tiveram origem no processamento de imagem[13], como a técnica de Autofaces, onde eram utilizadas como um método de compressão. A idéia central das Autovozes é de reduzir a dimensão dos dados, que normalmente é um grande número de variáveis inter-relacionadas, de forma que mantenha ainda o máximo possível da variação presente nos dados. [21].

A redução dos dados é obtida através da transformação para um novo grupo de variáveis, as componentes principais, que são descorrelacionadas e que são ordenadas de tal forma que as primeiras retenham a maior variação presente em todas as variáveis originais.

Segundo [17] “As Autovozes formam uma base de um subespaço do espaço dos modelos acústicos da voz e são escolhidas de tal maneira que possam modelar as variações entre os locutores”. Dado um grupo de parâmetros dos diferentes SDs, a análise em componentes principais define as direções lineares que contêm a maior variação dos dados. Chamamos estas direções lineares de componentes principais, ou Autovozes.

5.2.1 Análise em componentes principais

O método de análise em componentes principais (PCA)[21] também é conhecido como método de Karhunen-Loève [11]. A PCA é uma transformação ortogonal que reduz um grande número de variáveis correlacionadas em um pequeno número de componentes descorrelacionadas. A PCA garante que para os dados originais, o erro quadrático médio introduzido pela redução da dimensão é minimizado.

Algebricamente, as componentes principais são combinações lineares de D variáveis aleatórias $\mu_1, \mu_2, \dots, \mu_D$ sendo nossas variáveis aleatórias neste caso as médias dos HMMs. Geometricamente, estas combinações lineares representam a seleção de um novo sistema de coordenadas obtido pela rotação do sistema original dado seus eixos de coordenadas $\mu_1, \mu_2, \dots, \mu_D$. Estes novos eixos representam as direções com a máxima variabilidade.

Seja o vetor aleatório das médias dos modelos HMM dado por $\boldsymbol{\mu} = [\mu_1, \mu_2, \dots, \mu_D]^T$. Definimos a primeira componente principal ξ_1 como sendo a combinação linear dos μ_j , tal que a variância seja maximizada e definimos a i -ésima componente principal ξ_i , para $i > 0$, como sendo a combinação linear dos μ_j , tal que a variância seja maximizada e que ξ_i seja não-correlacionado com $\xi_1, \dots, \xi_{i-1}$.

$$\begin{aligned}\xi_1 &= \boldsymbol{\varphi}^{(1)T} \boldsymbol{\mu} = \varphi_1^{(1)} \mu_1 + \varphi_2^{(1)} \mu_2 + \dots + \varphi_D^{(1)} \mu_D \\ \xi_2 &= \boldsymbol{\varphi}^{(2)T} \boldsymbol{\mu} = \varphi_1^{(2)} \mu_1 + \varphi_2^{(2)} \mu_2 + \dots + \varphi_D^{(2)} \mu_D \\ &\vdots \\ \xi_D &= \boldsymbol{\varphi}^{(D)T} \boldsymbol{\mu} = \varphi_1^{(D)} \mu_1 + \varphi_2^{(D)} \mu_2 + \dots + \varphi_D^{(D)} \mu_D\end{aligned}\tag{5.1}$$

onde $\boldsymbol{\varphi}^{(j)}$, $j=1,2,\dots,D$, é um vetor de D constantes. Imagine que, a partir dos dados de adaptação de T locutores, nós tenhamos estimado T supervetores de parâmetros de média, $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_T$, cada um de dimensão D , para estimarmos a matriz de covariância, $\Sigma_{\boldsymbol{\mu}}$, de dimensão $D \times D$. Dada a matriz de covariância, extraímos sua matriz de

autovetores, $\mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_D]$, onde $\mathbf{e}_i = \sum_{k=0}^{D-1} \phi_k^{(i)} \boldsymbol{\mu}_k$. As componentes principais são então dadas por $\xi_i = \sum_{k=0}^{D-1} \phi_k^{(i)} \boldsymbol{\mu}_k$, ou seja, o novo sistema de coordenadas é dado pela matriz de autovetores, $\mathbf{E}$, da matriz de covariância $\Sigma_{\boldsymbol{\mu}}$.

Apesar de a análise em componentes principais ser definida com relação à matriz de covariância de $\boldsymbol{\mu}_j$, outros tipos de produtos podem ser utilizados, como a matriz de correlação ou mesmo $\boldsymbol{\mu}\boldsymbol{\mu}^T$. Nós optamos por trabalhar com a matriz $\boldsymbol{\mu}\boldsymbol{\mu}^T$ [20].

5.3 Estimação do Autoespaço

Quando se utiliza a técnica de Autovozes a fim de fazer adaptação ao locutor, o primeiro passo a ser dado é a criação do Autoespaço. Previamente deve ser treinado o SD de cada um dos T locutores base do sistema. De cada um dos T SDs extraímos os parâmetros a serem adaptados. No nosso caso os parâmetros a serem extraídos são todas as médias das Gaussianas dos HMMs.

As médias de cada um dos T modelos SD são então copiadas em forma de um vetor, o chamado supervetor. A ordem com que as médias são copiadas no supervetor é indiferente, porém deve ser a mesma para todos os T locutores. Cada supervetor tem dimensão D, onde D é o número de médias a serem adaptadas. Com os T supervetores de dimensão D, geramos então uma matriz $\mathbf{M}$ de dimensão (D x T).

$$\mathbf{M} = [\boldsymbol{\mu}^{(1)} \quad \boldsymbol{\mu}^{(2)} \quad \dots \quad \boldsymbol{\mu}^{(T)}] \quad (5.2)$$

onde $\boldsymbol{\mu}^{(t)}$ é o supervetor de dimensão (D x 1) que contém todas as médias do locutor t, com $t=1,2,\dots,T$. Essas médias são os parâmetros a serem adaptados para os novos locutores. Por exemplo, no nosso caso, onde o nosso sistema de reconhecimento de voz tem um vocabulário de 20 palavras, cada palavra é representada por um modelo HMM de 6 estados (o primeiro e o último estados são não-emitentes), cada estado possui uma

distribuição de saída associada de uma gaussiana de 13 variáveis. O número de médias, no total, a serem adaptadas são: $D=20(\text{palavras}) \times 4(\text{estados emitentes por palavra}) \times 13(\text{parâmetros das médias das Gaussiana}) = 1040(\text{parâmetros de médias})$.

5.3.1 Definição do número de componentes de autovozes

Apesar de D componentes serem necessárias para se reproduzir a total variabilidade do sistema, quase sempre a maior parte desta variabilidade está concentrada em um pequeno número, K , de componentes principais. Ou seja, a maior concentração de informações estão nas K componentes principais. Então as K componentes principais podem substituir as D variáveis originais e o grupo original de dados constituído por T observações das D variáveis, é reduzido a um grupo de T medidas das K componentes principais. O valor de K deve ser menor que o posto da matriz, que é dado por $T=\text{posto}(\mathbf{M})$ e como T normalmente é menor que a dimensão D da matriz $\mathbf{M}$, então, $K < T < D$.

O valor de K pode ser definido de diversas formas [22]. Nós aqui vamos trabalhar com a porcentagem da variação cumulativa. Variação cumulativa é a relação entre o autovalor associado a cada autovetor com a soma total de todos os autovalores da matriz. Em geral, se consideram as K autovozes que contribuem em conjunto em torno de 80 a 90% da variação total. O cálculo da variação cumulativa é dado pela Equação (5.3).

$$\%VarCum_K = 100 \cdot \frac{\sum_{k=1}^K \lambda_k}{\sum_{d=1}^D \lambda_d} \quad (5.3)$$

onde λ_k é o autovalor associado ao autovetor $\mathbf{e}_k$. O nosso novo espaço de Autovozes é então dado por:

$$\tilde{\mathbf{E}} = [\mathbf{e}_1 \quad \mathbf{e}_2 \quad \cdots \quad \mathbf{e}_K] \quad (5.4)$$

onde as últimas (D-K) autovozes foram ignoradas, pois são os autovetores que apresentam a menor variação associada.

5.4 Técnicas de estimação dos Autovozes

Existem várias técnicas que podem ser utilizadas para se estimar a adaptação de Autovozes. No presente trabalho, utilizamos duas técnicas, que são a de projeção e a de auto-decomposição de máxima verossimilhança (MLE). A partir de agora, a fim de simplificar a notação, vamos fazer $\mathbf{E} = \tilde{\mathbf{E}}$.

5.4.1 Projeção

Seja o locutor L representado pelas coordenadas do ponto $\hat{\boldsymbol{\mu}}$ no espaço K-dimensional. Sejam $\mathbf{e}_1, \dots, \mathbf{e}_K$ K Autovozes e, por sua vez, $\mathbf{E} = [\mathbf{e}_1 \ \mathbf{e}_2 \ \dots \ \mathbf{e}_K]$, uma matriz de dimensão D x K. Para se efetuar a adaptação de Autovozes utilizando a técnica de projeção é necessário se treinar um modelo SD do locutor a ser adaptado. Do modelo SD são então extraídas as suas médias e copiadas no supervetor, que chamaremos de $\boldsymbol{\mu}$, de dimensão (D x 1), que será projetado no espaço K-dimensional para obtermos o ponto $\hat{\boldsymbol{\mu}}$. Considere a seguinte projeção:

Figura 5.1 - Ilustração da projeção do novo locutor no espaço gerado pelas autovozes

Onde $\hat{\boldsymbol{\mu}}$ é o vetor de média adaptado, $\boldsymbol{\mu}$ é o vetor de média original, $\boldsymbol{\varepsilon}_k$ é o vetor do erro inserido devido a redução das dimensão e $\mathbf{e}_k$ é o autovoz. Seja:

$$\boldsymbol{\mu} = \hat{\boldsymbol{\mu}} + \boldsymbol{\varepsilon}_k \quad (5.5)$$

o erro contido no subespaço das (D-K) autovozes desprezadas. Projetando a Equação (5.5) no espaço das autovozes, $\mathbf{E}$, temos:

$$\mathbf{e}_k^T \boldsymbol{\mu} = \mathbf{e}_k^T \hat{\boldsymbol{\mu}} + \mathbf{e}_k^T \boldsymbol{\varepsilon}_k \quad (5.6)$$

Mas como,

$$\hat{\boldsymbol{\mu}} = v_k \mathbf{e}_k \quad (5.7)$$

e como $\mathbf{e}_k^T \mathbf{e}_k = 1$, obtém-se:

$$\mathbf{e}_k^T \boldsymbol{\mu} = v_k \mathbf{e}_k^T \mathbf{e}_k \quad (5.8)$$

Portanto, temos:

$$v_k = \frac{\mathbf{e}_k^T \boldsymbol{\mu}}{\mathbf{e}_k^T \mathbf{e}_k} \quad (5.9)$$

e como

$$\mathbf{v} = \sum_{k=1}^K v_k \mathbf{e}_k, \quad (5.10)$$

$$\mathbf{v} = \sum_{k=1}^K \frac{\mathbf{e}_k^T \boldsymbol{\mu}}{\mathbf{e}_k^T \mathbf{e}_k} \mathbf{e}_k, \quad (5.11)$$

como $\mathbf{e}_k^T \mathbf{e}_k = 1$ então teremos a projeção final dada por:

$$\hat{\boldsymbol{\mu}} = \mathbf{E} \mathbf{E}^T \boldsymbol{\mu} \quad (5.12)$$

Determinando assim as coordenadas do ponto $\hat{\mu}$. Podemos, então, agora gerar o modelo adaptado para o locutor L. Como os D parâmetros em $\hat{\mu}$ representam somente as médias das gaussianas, o restante dos parâmetros dos modelos, como as variâncias, as probabilidades de transição e os pesos das misturas das gaussianas do modelo HMM, serão extraídos dos modelos do SI.

Esta técnica de projeção tem a desvantagem de que para a estimativa dos parâmetros, todo o vocabulário deve ser utilizado como dados de adaptação.

5.4.2 Auto-decomposição de máxima verossimilhança (MLEM)

Definimos o vetor de médias adaptadas como sendo:

$$\hat{\mu} = \mathbf{E} \cdot \mathbf{v} = \sum_{k=1}^K v(k) \mathbf{e}_k \quad (5.13)$$

E por sua vez, definimos o modelo estimado da média do estado j de uma palavra como sendo:

$$\hat{\mu}(j) = \sum_{k=1}^K v(k) \mathbf{e}_k(j), \quad j=1,2,\dots,D \quad (5.14)$$

Onde D é o número total de parâmetros a serem adaptados, $\mathbf{e}_k$ é um vetor coluna e $v(k)$, com $k=1,2,\dots,K$, são os K novos valores das componentes principais, que combinados com os seus respectivos Autovalores estimam os novos valores das médias. Como para estimar estes novos autovalores nós também buscamos maximizar a verossimilhança dos dados de adaptação com relação ao modelo, iremos também maximizar a função auxiliar Q, Equação (3.12):

$$Q(\lambda, \hat{\lambda}) = \frac{1}{2} P(\mathbf{O} | \lambda) \sum_j \sum_t \gamma_j(t) [E \log(2\pi) + \log |\Sigma_j| + h(\mathbf{o}_t, j)] \quad (5.15)$$

onde

$$h(\mathbf{o}_t, j) = (\mathbf{o}_t - \hat{\mu}_j)^T (\Sigma_j)^{-1} (\mathbf{o}_t - \hat{\mu}_j) \quad (5.16)$$

e $P(\mathbf{O} | \lambda)$, γ_j , $\hat{\mu}_j$, Σ_j , $\mathbf{o}_t$ são respectivamente, a probabilidade do conjunto de observações $\mathbf{O}$ dado o modelo da palavra, a probabilidade de ocupação da observação no tempo t no estado j , a média do estado j , a matriz de covariância do estado j e $\mathbf{o}_t$ vetor de observações no tempo.

Para o cálculo dos coeficientes de Autovozes, devemos maximizar a função auxiliar Q com relação a $v(k)$, ou seja:

$$\frac{\partial Q}{\partial v(k)} = 0 \quad (5.17)$$

para $k=1,2,...,K$. E por sua vez, temos:

$$\frac{\partial Q}{\partial v(k)} = -\frac{n}{2} P(\mathbf{O} | \lambda) \sum_j \sum_t \gamma_j(t) \frac{\partial}{\partial v(k)} h(\mathbf{o}_t, j) = 0 \quad (5.18)$$

Vamos definir a derivada da média estimada como sendo:

$$\frac{\partial}{\partial v(k)} \hat{\mu}_j = \frac{\partial}{\partial v(k)} \sum_{i=1}^K v(i) \mathbf{e}_j(i) = \mathbf{e}_j(k) \quad (5.19)$$

As próximas equações vão detalhar os passos para se efetuar a derivada da Equação (5.16).

$$\frac{\partial}{\partial v(k)} h(\mathbf{o}_t, j) = \frac{\partial}{\partial v(k)} (\mathbf{o}_t - \hat{\mathbf{\mu}}_j)^T (\boldsymbol{\Sigma}_j)^{-1} (\mathbf{o}_t - \hat{\mathbf{\mu}}_j) \quad (5.20)$$

$$= \frac{\partial}{\partial v(k)} \left\{ \mathbf{o}_t^T \boldsymbol{\Sigma}_j^{-1} \mathbf{o}_t - 2\mathbf{o}_t^T \boldsymbol{\Sigma}_j^{-1} \hat{\mathbf{\mu}}_j + \hat{\mathbf{\mu}}_j^T \boldsymbol{\Sigma}_j^{-1} \hat{\mathbf{\mu}}_j \right\} \quad (5.21)$$

Separando o diferenciador entre as parcelas, tem-se:

$$\frac{\partial}{\partial v(k)} h(\mathbf{o}_t, j) = -2\mathbf{o}_t^T \boldsymbol{\Sigma}_j^{-1} \left(\frac{\partial}{\partial v(k)} \hat{\mathbf{\mu}}_j \right) + \frac{\partial}{\partial v(k)} (\hat{\mathbf{\mu}}_j^T \boldsymbol{\Sigma}_j^{-1} \hat{\mathbf{\mu}}_j) \quad (5.22)$$

Agora vamos substituir $\frac{\partial}{\partial v(k)} \hat{\mathbf{\mu}}_j$ pela Equação (5.19) e também aplicar algumas propriedades básicas da diferenciação.

$$\frac{\partial}{\partial v(k)} h(\mathbf{o}_t, j) = -2\mathbf{o}_t^T \boldsymbol{\Sigma}_j^{-1} \mathbf{e}_j(k) + \left(\frac{\partial}{\partial v(k)} \hat{\mathbf{\mu}}_j^T \right) \boldsymbol{\Sigma}_j^{-1} \hat{\mathbf{\mu}}_j + \hat{\mathbf{\mu}}_j^T \boldsymbol{\Sigma}_j^{-1} \left(\frac{\partial}{\partial v(k)} \hat{\mathbf{\mu}}_j \right) \quad (5.23)$$

Uma vez que a transposta de um escalar é ele mesmo, temos:

$$\frac{\partial}{\partial v(k)} h(\mathbf{o}_t, j) = -2\mathbf{o}_t^T \boldsymbol{\Sigma}_j^{-1} \mathbf{e}_j(k) + 2 \left(\frac{\partial}{\partial v(k)} \hat{\mathbf{\mu}}_j^T \right) \boldsymbol{\Sigma}_j^{-1} \hat{\mathbf{\mu}}_j \quad (5.24)$$

por fim temos que:

$$h(\mathbf{o}_t, j) = 2[-\mathbf{o}_t^T \boldsymbol{\Sigma}_j^{-1} \mathbf{e}_j(k) + \mathbf{e}_j^T(k) \boldsymbol{\Sigma}_j^{-1} \sum_{i=1}^K v(i) \mathbf{e}_j(i)] \quad (5.25)$$

Agora, voltando à diferenciação da função auxiliar Q, Equação (5.15), temos:

$$\sum_j \sum_t \gamma_j(t) \left\{ -\mathbf{e}_j^T(k) \boldsymbol{\Sigma}_j^{-1} \mathbf{o}_t + \sum_{i=1}^K v(i) \mathbf{e}_j^T(i) \boldsymbol{\Sigma}_j^{-1} \mathbf{e}_j(k) \right\} = 0 \quad (5.26)$$

De onde tiramos o sistema de equações lineares:

$$\sum_j \sum_t \gamma_j(t) \mathbf{e}_j^T(k) \boldsymbol{\Sigma}_j^{-1} \mathbf{o}_t = \sum_j \sum_t \gamma_j(t) \sum_{i=1}^K v(i) \mathbf{e}_j^T(i) \boldsymbol{\Sigma}_j^{-1} \mathbf{e}_j(k) = 0 \quad (5.27)$$

para $k=1, \dots, K$.

Reescrevendo a Equação (5.27) da forma matricial temos:

$$\mathbf{v} = \mathbf{Z}^{-1} \mathbf{y} \quad (5.28)$$

onde os coeficientes do vetor $\mathbf{y}$, de dimensão $(K \times 1)$ são definidos por:

$$y(i) = \sum_j \sum_t \gamma_j(t) \mathbf{e}_j^T(i) \boldsymbol{\Sigma}_j^{-1} \mathbf{o}_t \quad \text{para } i = 1, 2, \dots, K \quad (5.29)$$

e os coeficientes da matriz $\mathbf{Z}$ de dimensão $(K \times K)$ são definidos por:

$$z_{ki} = \sum_j \sum_t \gamma_j(t) \sum_{i=1}^K v(i) \mathbf{e}_j^T(i) \boldsymbol{\Sigma}_j^{-1} \mathbf{e}_j(k) \quad (5.30)$$

5.5 Conclusão

Neste capítulo apresentamos a técnica de adaptação de Autovozes e como ela é implementada. Esta é uma técnica pouco conhecida, porém apresenta uma boa

performance como adaptação rápida a sistemas de reconhecimento de voz, como veremos no capítulo 7.

6 METODOLOGIA

6.1 Introdução

Neste capítulo vamos descrever a metodologia utilizada nas fases de implementação do sistema. Vamos falar do software Hidden Markov Model Toolkit (HTK) e suas configurações. Vamos apresentar as configurações utilizadas no sistema e de que maneira foram feitas as adaptações. Também vamos falar como foram feitas as medidas de desempenho que serão apresentadas no capítulo 6.

6.2 Definições

O presente trabalho foi desenvolvido no período de janeiro a setembro de 2001. Para o desenvolvimento do sistema, foi utilizada uma máquina Compaq Celeron 600MHz, com sistema operacional Windows Me.

Os softwares utilizados para desenvolver o sistema foram o HTK[3] e o Matlab[6]. No software HTK foram desenvolvidos os sistemas de reconhecimento de voz base. O mesmo também foi utilizado para testar os sistemas adaptados bem como efetuar a adaptação do tipo MLLR. A ferramenta Matlab foi utilizada para desenvolver a adaptação de Autovozes, assim como gerar scripts de edição dos modelos, entre outros necessários.

A ferramenta HTK foi escolhida devido a sua credibilidade com relação à modelagem de sistemas de reconhecimento de voz. Por sua vez, a ferramenta Matlab foi escolhida devido a sua acessibilidade quanto a manipulação de matrizes.

Para efetuar os treinamentos e testes foi utilizada a base de dados do inglês TIMIT[7].

O sistema proposto é de reconhecimento de voz de palavras isoladas e vocabulário restrito. O vocabulário é mostrado na Tabela 6.1.

she	greasy	ask	oily
had	wash	me	rag
your	water	to	like
dark	year	carry	that
suit	dont	an	in

Tabela 6.1- Vocabulário do sistema de reconhecimento adotado

Na Tabela 6.2 são apresentadas as configurações utilizadas durante a fase de análise do sinal e também as configurações utilizadas no modelo HMM.

Numero de estados do modelo	6 estados
Parâmetros característicos	12 coeficientes MFCC e 1 de energia; um total de 13 parâmetros.
Número de filtros no banco de filtros mel	22
Tipo de Janela	Hamming
Duração da Janela	8 ms
Superposição entre janelas	6 ms
Coeficiente de Pré-ênfase	0,97
Frequência de amostragem do sinal de entrada	16 kHz

Tabela 6.2 - Configuração do sistema de reconhecimento adotado

Para treinar o sistema independente de locutor foram utilizados vinte locutores, dentre os quais sete são mulheres e treze são homens.

6.3 Ferramenta HTK

O HTK[3] é uma ferramenta que se utiliza para modelar um sistema de reconhecimento de voz baseado em modelos de Markov do tipo oculto (HMM). Ela possui diversas rotinas desde a análise do sinal até a fase de cálculo de desempenho.

6.3.1 Análise do sinal

Para se fazer análise do sinal, que é a fase no qual são extraídos os parâmetros relevantes a serem utilizados nas próximas fases do sistema, é utilizada a função Hcopy, que é evocada pela seguinte linha de comando:

» hcopy -C config -S script.scp

O arquivo *config* contém as configurações utilizadas pelo procedimento. Abaixo será apresentado o arquivo de configuração utilizado.

Arquivo config:

```
SOURCEKIND = WAVEFORM
SOURCEFORMAT = WAV
SOURCERATE = 625
TARGETKIND = MFCC_E
TARGETFORMAT = HTK
TARGETRATE = 20000
ZMEANSOURCE = TRUE
WINDOWSIZE = 80000.0
USEHAMMING = TRUE
PREEMCOEF = 0.97
NUMCHANS = 24
CEPLIFTER = 22
NUMCEPS = 12
```

Os parâmetros SOURCEKIND e SOURCEFORMAT definem o formato do arquivo de entrada. Da mesma maneira, os parâmetros TARGETKIND e TARGETFORMAT definem o formato do arquivo de saída. Os valores WAVEFORM, WAV definem que o arquivo de entrada é um sinal e codificado do tipo WAV windows. Os valores , MFCC_E e HTK definem que o arquivo de saída é do tipo Mel-Cepstrais com a adição da energia e codificado no formato próprio do HTK.

O valor de 625 do parâmetro SOURCERATE representa a período de amostragem do sinal de entrada, na escala de 100ns, que é de 16KHz. Já o valor de 20000 do parâmetro TARGETRATE define que o deslocamento adotado para a obtenção da próxima janela é

de 2ms. O parâmetro ZMEANSOURCE define que deverá ser extraído das amostras do sinal o seu valor DC.

Já a variável WINDOWSIZE define o tamanho da janela a ser utilizada, que no nosso caso, será de 8ms. Visto que o deslocamento entre janelas consecutivas é de 2ms, a superposição entre janelas será de 6ms. O parâmetro USEHAMMING define que a janela a ser utilizada é do tipo Hamming. Por fim parâmetro PREEMCOEF define o coeficiente do filtro de pré ênfase que será 0.97.

O parâmetro NUMCHANS define o número de canais do banco de filtros cepstrais, CEPLIFTER define o número de filtros do banco de filtros cesptrais e NUMCEPS define o número de coeficientes de saída do banco de filtros cepstrais.

Por sua vez, o arquivo script.scp deve conter um script com os nomes dos arquivos de entrada seguido pelos nomes dos arquivos de saída. Um exemplo é apresentado abaixo.

Arquivo script.scp:

dark.wav	dark.mfc
greasy.wav	greasy.mfc
had.wav	had.mfc
in.wav	in.mfc

6.3.2 Treinamento do sistema

Para se fazer o treinamento do sistema, primeiramente devemos inicializar os modelos HMM, utilizando a função *Hinit*. A chamada desta função é apresentada abaixo:

```
» hinit -m 1 -S tr_she -M hmm1 she
```

onde -m 1 significa que será utilizado no mínimo um arquivo de entrada no treinamento. O arquivo tr_she contém o caminho dos arquivos a serem utilizados no treinamento. A chave -M indica onde será salvo o novo modelo, e she é o nome do protótipo do modelo

HMM para a palavra She. Um exemplo dos arquivos tr_she e she serão apresentados abaixo.

Arquivo tr_she:

fsah0/she.mfc
fsjk1/she.mfc
fsma0/she.mfc

Arquivo she (protótipo do modelo HMM)

```
~o
<STREAMINFO> 1 13
<VECSIZE> 13<NULLD><MFCC_E>
~h "she"
<BeginHMM>
<NumStates> 6
<State> 2
  <Mean> 13
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
  <Variance> 13
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
<State> 3
  <Mean> 13
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
  <Variance> 13
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
<State> 4
  <Mean> 13
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
  <Variance> 13
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
<State> 5
  <Mean> 13
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
  <Variance> 13
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
<TransP> 6
  0.0 1.0 0.0 0.0 0.0 0.0
  0.0 0.5 0.5 0.0 0.0 0.0
  0.0 0.0 0.5 0.5 0.0 0.0
  0.0 0.0 0.0 0.5 0.5 0.0
  0.0 0.0 0.0 0.0 0.5 0.5
  0.0 0.0 0.0 0.0 0.0 0.0
<EndHMM>
```


Após efetuar a inicialização dos modelos, faremos o refinamento dos mesmos utilizando a estimativa de Baum-Welch, que é executada pela função *Hrest*. A chamada da função é apresentada abaixo:

```
» hrest -m 1 -S tr_she -M hmm1 hmm1/she
```

onde *tr_she* é o mesmo arquivo apresentado na descrição da função *Hinit*. Já o arquivo *she* é o arquivo de saída da função *Hinit*, ou seja, o arquivo já inicializado do modelo HMM da palavra *she*.

6.3.3 Medidas de desempenho

Para se fazer alguma medida de desempenho do sistema, faremos primeiramente o reconhecimento das palavras, dados os modelos HMM previamente treinados. É necessária a criação de um arquivo que contenha todos os modelos HMM, que nós chamamos de *hmmset*.

O reconhecimento das palavras é feito utilizando o algoritmo de Viterbi. Este algoritmo é implementado na função chamada *Hvite*. Esta função é executada como mostra a seguinte linha de comando:

```
» hvite -S dados_teste.scp -H hmm1\hmm_set -i results.mlf -w wdnnet dict hmmlist
```

onde *dados_teste.scp* é um arquivo que contém o caminho dos arquivos com as locuções a serem reconhecidas. O arquivo *results.mlf* será o arquivo de saída, contendo o resultado do reconhecimento. O arquivo *wdnnet* contém o modelo de linguagem, que vai orientar no processo de busca. O arquivo *dict* é o arquivo que contém o vocabulário do sistema. E o arquivo *hmmlist* contém a lista de palavras de acordo com a sequência de modelos HMM dispostas no arquivo *hmmset*.

De posse do resultado obtido utilizando a função *Hvite*, o arquivo *results.mlf*, vamos então executar a função que vai medir o desempenho do sistema. Esta função é a chamada *Hresults*. Ela é executada de acordo com a seguinte linha de comando.

```
» hresults -I refs hmmlist results.mlf
```

Esta função faz a comparação das palavras identificadas, contidas no arquivo *results.mlf*, com o respectivo arquivo de transcrição da palavra, contido no arquivo *refs*. Ela faz um alinhamento entre a palavra reconhecida e a sua transcrição correspondente e com isso calcula a taxa de acerto das palavras reconhecidas.

6.4 Adaptação MLLR

A adaptação MLLR foi também efetuada utilizando a ferramenta HTK. Para se executar a adaptação, antes de tudo, tem que se criar o arquivo com as estatísticas do sistema, chamado de *stats*. Este arquivo é gerado pela função *Herest*, como mostra a seguinte linha de comando:

```
» herest -I labels.mlf -s stats -S tr -H hmm0/hmmset -M hmm1 hmmlist
```

onde o arquivo *labels.mlf* possui o caminho e a descrição das palavras que foram utilizadas durante o treinamento. E o arquivo *tr* contém toda a lista dos arquivos de treinamento.

Arquivo labels.mlf:

```
#!MLF!#
"../timit/d_*/she.mfc"
she
.
"../timit/d_*/had.mfc"
had
```

```

.
"/timit/d_*/your.mfc"
your
.
...(continua)

```

Arquivo tr:

```

../timit/d_fcjf0/she.mfc
../timit/d_fcjf0/had.mfc
../timit/d_fcjf0/your.mfc
...(continua)

```

Em seguida é necessário se definir o número de classes de regressão a serem utilizadas, isto é definido no arquivo *regtree.reg*. Este arquivo é apresentado abaixo:

Arquivo regtree.reg:

```

RN "md"
LS "stats"
RC 1 "rtree"

```

De posse deste arquivo, executaremos, então, a modificação no arquivo *hmmset* para definir quais gaussianas pertencem a cada classe de regressão. Este procedimento é executado pela função *Hhed*, que edita os modelos HMM.

```
» hhed -T 1 -H hmm1/hmmset -M hmm2 regtree.hed hmmlist
```

Após definir o número de classes e quais gaussianas pertencem a qual classe, podemos então aplicar a rotina de adaptação. Esta rotina é executada pela seguinte linha de comando:

```
» Headapt -C config_a -g -S adapt.scp -I adaptwords.mlf -H hmmset -K global.tmf
hmmlist
```

onde *config_a* é o arquivo que possui as configurações necessárias para executar a adaptação. O arquivo *adapt.scf* contém a lista dos dados de adaptação. *Adaptwords.mlf* contém a descrição desta lista de dados. E o arquivo *global.tmf* contém a estimativa da matriz de transformação *W*. O uso de *-g* força que a adaptação seja do tipo global.

6.5 Adaptação Autovozes

A adaptação de Autovozes foi feita utilizando o Matlab. Primeiramente treinamos um modelo SD para cada um dos 20 locutores base. Feito isso criamos a rotina chamada de *Autovozes.m*, que executa as seguintes tarefas descritas abaixo.

Rotina Autovozes.m:

```

Inicializa o vetor mu_ e a matriz mu
Abre o arquivo que contém o nome dos locutores base
Para todos os locutores fazer
    {
        Abrir o arquivo do modelo SD respectivo
        Salvar as médias no vetor auxiliar mu_
        Copiar o vetor mu_ para a matriz mu
    }
Fecha o arquivo com o nome dos locutores
Calcula a matriz  $X = \mu \mu'$ 
Calcula os autovalores (D) e autovetores (U) da matriz X
Salva os dados em disco
Fim

```

Para se fazer a adaptação utilizando o método de projeção, foi criada a função *proj.m*. São entradas desta rotina, o nome do locutor a ser adaptado e a base de Autovozes **U**. Esta rotina é descrita abaixo:

Rotina proj.m:

Cria-se o modelo SD deste novo locutor
 Lê-se deste modelo as suas médias que são armazenadas no vetor **mu**

Para o valor de $k=x$

Calcula-se o ponto $p=U(:,1:k)*U(:,1:K)'\cdot\mathbf{mu}$
 Salva as novas médias no novo arquivo com os modelos HMM

Fim

Por fim, para se efetuar a adaptação utilizando o método MLED, criaram-se diversas rotinas, sendo as duas principais: *adapt.m* e *coefs_av.m*. Os parâmetros de entrada da função *adapt* são o nome do novo locutor, os dados de adaptação e a base de Autovozes **U**. Nesta rotina preparam-se todos os dados necessários para se efetuar o cálculo dos coeficientes de Autovozes. Isto é, são calculadas as probabilidades de ocupação por estado, de acordo com os dados de adaptação que são lidos também dos arquivos de parâmetros MFCC das dadas palavras de adaptação e também o arquivo dos modelos HMM. Com os dados já preparados, executamos então a rotina de cálculo dos coeficientes, que é apresentada abaixo:

Rotina coefs_av.m:

Inicializa-se o vetor **v** e o vetor **q**
K := dimensão do autovoz desejado
j=número de estados do modelo
t=tempo

$$\text{Calcule: } z_{ki} = \sum_j \sum_t \gamma_j(t) \sum_{i=1}^K v(i) \mathbf{e}_j^T(i) \mathbf{\Sigma}_j^{-1} \mathbf{e}_j(k) \quad , k = 1, 2, \dots, K$$

$$\text{Calcule: } y_k = \sum_j \sum_t \gamma_j(t) \mathbf{e}_j^T(k) \mathbf{\Sigma}_j^{-1} \mathbf{o}_t \quad , k = 1, 2, \dots, K$$

Calcule: $\mathbf{q} = \text{inv}(\mathbf{Z}) \cdot \mathbf{Y}$;
 Calcula as novas médias: $\mathbf{MU_A} = \mathbf{U}(:, 1:K) \cdot \mathbf{q}$;
 Salva as novas médias no novo arquivo com os modelos HMM

Fim

6.6 Conclusão

Neste capítulo foram apresentadas as metodologias utilizadas na fase de implementação do sistema. Foram dadas breves noções da ferramenta HTK e suas configurações. Por fim, falamos das implementações dos métodos de adaptação feitos em Matlab, onde fizemos breves descrições das rotinas desenvolvidas. No próximo capítulo apresentaremos os resultados obtidos através dos testes feitos para as técnicas de adaptação propostas.

7 AVALIAÇÃO EXPERIMENTAL

7.1 Introdução

O propósito do trabalho é verificar a eficácia de duas diferentes técnicas de adaptação ao locutor: MLLR e Autovozes. Os experimentos feitos seguiram as definições apresentadas no Capítulo 6. Os treinamentos dos SIs e SDs foram desenvolvidos utilizando a ferramenta Hidden Markov Models Toolkit (HTK)[3]. A técnica de adaptação MLLR foi testada utilizando também a ferramenta HTK. Já a técnica de Autovozes foi desenvolvida utilizando a ferramenta Matlab[6].

7.2 Experimentos

Para as configurações já apresentadas no capítulo anterior, Tabela 6.1 e Tabela 6.2, foi obtida uma taxa média de reconhecimento de voz para o SI de 84,33%, utilizando, para isso, 15 locutores de testes, dentre os quais seis são mulheres e nove homens.

Como mencionado no capítulo anterior, os treinamentos do SI e SDs foram feitos utilizando a ferramenta HTK. Os testes com a técnica MLLR foram também feitos utilizando a ferramenta HTK, uma vez que já existe uma rotina desenvolvida para este algoritmo nesta ferramenta. A rotina do algoritmo do Autovozes foi desenvolvida utilizando a ferramenta Matlab. Os testes utilizando esta técnica de adaptação foram efetuados utilizando, também, a ferramenta HTK.

7.2.1 Sistema adaptado utilizando MLLR

Para se efetuar os testes com a técnica MLLR, não se utilizaram as classes de regressão. Todos os testes de adaptação foram feitos utilizando a adaptação global, pois o sistema de reconhecimento é de palavras isoladas e vocabulário restrito.

Primeiramente buscamos verificar a eficiência do método MLLR utilizando como dado de adaptação um vocabulário muito reduzido, de apenas uma palavra.

Palavra	Taxa de acerto (%)
she	15,33
had	7,33
your	8,00
dark	13,33
suit	8,33
in	5,67
greasy	8,67
wash	17,00
water	8,97
year	10,67
dont	6,33
ask	10,67
me	10,00
to	9,67
carry	7,33
an	8,33
oily	8,67
rag	7,00
like	8,00
that	10,33

Tabela 7.1 - Resultados da adaptação utilizando a técnica MLLR e uma palavra na adaptação.

Através dos dados da Tabela 7.1 percebe-se que para um vocabulário muito restrito, a técnica MLLR, não deve ser empregada. Isto se deve ao fato da pouca informação contida em apenas uma palavra, que não é suficiente para se estimar uma transformação linear para este tipo de sistema.

Já que uma palavra era insuficiente para se efetuar a adaptação, fizemos então alguns testes variando o número de palavras de adaptação. Como se esperava, para a técnica MLLR, o aumento do número de palavras implica em um aumento da taxa de reconhecimento, pois um volume maior de dados, permite uma melhor estimativa da matriz de transformação linear. Porém, como utilizamos a adaptação do tipo global, para um grupo de 16 palavras o sistema já apresenta a sua melhor performance. Quando é possível o aumento do volume de dados de adaptação, é então aconselhável que se

trabalhe com classes de regressão. A Tabela 7.2 apresenta os resultados obtidos variando o número de palavras de adaptação. Este resultados também podem ser visualizados de forma gráfica na Figura 7.1.

Nº de palavras	Taxa de acerto (%)
2	18,67
3	15,00
4	20,67
5	33,00
6	44,00
7	67,33
8	77,67
9	82,33
10	86,00
11	86,67
12	91,67
13	93,67
14	94,67
15	96,00
16	97,00
17	96,33
18	96,00
19	96,67
20	96,67

Tabela 7.2 - Resultados da adaptação MLLR variando o nº de palavras diferentes na adaptação.

Figura 7.1 - Curva que relaciona o número de palavras de adaptação com a taxa de reconhecimento adaptado resultante, utilizando a técnica MLLR.

7.2.2 Sistema adaptado utilizando Autovozes

O primeiro teste efetuado foi a verificação do efeito da variação da dimensão do subespaço das Autovozes. Isso foi feito variando-se o número de componentes de autovozes, que se faz variando k . Pela Figura 7.2 a maior variação cumulativa, Equação (5.3), está concentrada em torno do subespaço primeiras dimensões 3.

Para verificar a variação cumulativa na prática, foram utilizadas duas diferentes técnicas de implementação dos Autovozes: a técnica de projeção e a técnica de auto-decomposição de máxima verossimilhança (MLED). Para o propósito do teste, utilizamos todas as palavras do vocabulário como dados de adaptação. A técnica de projeção necessariamente deve ser feita utilizando todos os dados de treinamento. Os resultados destas adaptações são apresentados na Tabela 7.3 e também através da Figura 7.3. O espaço das autovozes foi calculado utilizando a matriz de correlação das amostras de médias dos 20 locutores base. A dimensão D de cada supervetor foi de 1040 (treze parâmetros característicos por estado vezes quatro estados emitentes por palavra vezes vinte palavras).

Valor de K	MLED(%)	Projeção(%)
1	85,33	79,67
2	90,67	85,67
3	90,67	85,67
4	89,67	86,67
5	90,33	87,00
6	89,67	87,33
7	90,67	88,00
8	91,00	88,67
9	91,33	89,00
10	91,67	89,67
11	92,00	90,33
12	91,67	90,33
13	91,33	89,33
14	91,00	89,00
15	91,00	88,67
16	91,33	87,67
17	91,00	88,00

Tabela 7.3 - Resultados da adaptação Autovozes utilizando todas as palavras do vocabulário , para diferentes dimensões do subespaço das Autovozes (K)

Para a técnica de MLED, o resultado obtido utilizando o subespaço principais de dimensão $K=3$ já é satisfatório. Já para a técnica de projeção um resultado equivalente ao do MLED só foi conseguindo utilizando o subespaço principal de dimensão $K=11$. Cada dimensão incorpora uma porcentagem da variação cumulativa do autoespaço. Com o aumento do volume de dados, aumenta também a variação dos mesmos e com isso os novos parâmetros serão estimados de forma mais precisa.

Figura 7.2 – Variação cumulativa das autovozes

Figura 7.3 - Taxa de reconhecimento para a técnica Autovozes variando o número de componentes de autovozes para os métodos MLED e projeção.

Figura 7.4 - Taxa de reconhecimento independente e adaptado para diferentes N°s de locutores Base do modelo SI.

Também foram feitos alguns testes variando o número de locutores do sistema independente. Pela Figura 7.4 fica claro que o número de locutores base é indiferente

para a melhora que o sistema vai apresentar. A melhora dos sistemas adaptados com relação aos seus sistemas originais SI é constante e em torno de 10%.

Na Tabela 7.4, são apresentados os testes que foram feitos utilizando somente uma palavra como dado de adaptação, utilizando três subespaços principais de dimensões $K=3, 6$ e 9 . De acordo com os resultados, o sistema foi mais bem adaptado utilizando a palavra *wash*. Por sua vez, as palavras *your* e *suit* apresentaram os menores resultados para o sistema adaptado. Para o caso de adaptação utilizando apenas uma palavra, não é interessante a utilização de uma dimensão k muito grande. Isto porque uma única palavra contém pouca informação, o que impossibilitaria uma estimativa mais precisa dos coeficientes das dimensões maiores. Por esse motivo, a modelagem, neste caso mais refinada, tende a piorar a taxa de reconhecimento.

O último teste foi feito variando o volume de dados de adaptação. Estes resultados são apresentados na Tabela 7.5 e também na Figura 7.5. A partir destes resultados ficou claro que o aumento da taxa de reconhecimento, utilizando esta técnica, independe do volume de dados de adaptação. A taxa de reconhecimento para uma palavra chegou a ser de 90% e a taxa utilizando todas as palavras como dados de adaptação foi de 91%. Claro que para algumas palavras o sistema utilizando apenas uma palavra como dado de adaptação não chegou a esta taxa de 90%. Por isso é necessário se fazer uma boa escolha dos dados a serem utilizados. A escolha deve ser de tal forma que a palavra escolhida contenha uma grande variabilidade de informação associada.

Dados de adaptação	K=3 (%)	K=6 (%)	K=9 (%)
she (1)	88,33	82,67	77,00
had (2)	88,67	84,67	81,33
your (3)	83,33	76,00	64,67
dark (4)	89,67	82,00	79,67
suit (5)	84,00	81,00	79,67
in (6)	88,67	82,33	72,67
greasy (7)	88,00	85,00	75,00
wash (8)	90,00	87,33	78,67
water (9)	88,67	86,67	77,33
year (10)	88,00	81,67	76,00
dont (11)	87,33	85,00	80,00
ask (12)	87,33	89,00	84,00
me (13)	88,67	86,67	83,33
to (14)	86,67	78,00	68,67
carry (15)	88,67	82,67	80,33
an (16)	89,33	84,00	81,33
oily (17)	89,00	78,33	75,67
rag (18)	89,00	82,67	77,33
like (19)	88,67	83,33	76,33
that (20)	88,00	77,00	76,67

Tabela 7.4 - Resultados da adaptação Autovozes, utilizando uma palavra como dados de adaptação, para as 3, 6 e 9 primeiras componentes de autovozes

Dados de adaptação	K=3 (%)	K=6(%)
palavra(1)	88,33	82,67
palavras (1-2)	88,67	86,67
palavras (1-3)	87,33	88,00
palavras (1-4)	88,33	89,67
palavras (1-5)	88,33	88,33
palavras (1-6)	88,67	89,33
palavras (1-7)	89,00	89,00
palavras (1-8)	89,33	89,33
palavras (1-9)	89,00	89,33
palavras (1-10)	90,00	90,00
palavras (1-11)	90,33	90,00
palavras (1-12)	90,67	90,33
palavras (1-13)	90,67	90,33
palavras (1-14)	90,33	90,00
palavras (1-15)	90,33	91,00
palavras (1-16)	90,67	90,67
palavras (1-17)	91,00	90,00
palavras (1-18)	90,67	90,00
palavras (1-19)	90,67	89,67
palavras (1-20)	90,67	89,67

Tabela 7.5 – Resultado da adaptação Autovozes variando o número de palavras diferentes de adaptação para as 3 e 6 primeiras componentes de autovozes

Figura 7.5 – Resultado da adaptação Autovozes variando o volume de dados de adaptação, para as 3 e 6 primeiras componentes de autovozes

7.3 Conclusões

Neste capítulo foram apresentados todos os resultados obtidos durante a fase de testes do projeto. Ficou claro que a técnica que mais se aplica para o nosso tipo de sistema é a técnica de Autovozes. Por outro lado a técnica MLLR seria muito bem empregada em sistemas maiores, onde possuíssemos um volume de dados de adaptação maior.

8 CONCLUSÃO

O objetivo do trabalho aqui apresentado foi de pesquisar técnicas de adaptação ao locutor. Para isso, utilizou-se um sistema de reconhecimento de voz, de palavras isoladas, baseado em modelados HMM. Foram feitos testes utilizando a técnica MLLR e com a técnica de Autovozes. A técnica MLLR foi utilizada como técnica de referência para compararmos com a técnica de Autovozes

Primeiramente pensávamos em trabalhar com a língua portuguesa, utilizando uma base de dados que iríamos construir. Porém, infelizmente, devido à diversidade das fontes ficou difícil de se gerar uma base de dados confiável. Para podermos validar a proposta de trabalho optamos então por trabalhar com a língua inglesa. Para isso utilizamos a já conhecida e consagrada base de dados TIMIT.

Em todos os testes feitos com a técnica MLLR optamos por implementar a adaptação do tipo global, esta escolha está baseada no fato de o nosso vocabulário ser restrito.

Dos resultados obtidos utilizando a técnica MLLR ficou claro que este tipo de adaptação não se adequou para o nosso tipo de sistema. Pois se consegue uma melhora com um volume de dados de adaptação muito elevado, se comparado com o volume de dados que utilizamos técnica de Autovozes.

Quando trabalhamos com a técnica de Autovozes, a definição mais importante, afora o volume de dados, é a definição do número de autovozes K , que iremos utilizar para modelar os novos dados. Durante os testes ficou claro que quanto menor é o volume de dados de adaptação, menor será a dimensão do espaço. E por sua vez, quanto maior o volume de dados maior o número de autovozes (K) necessárias para representa-las. Isto porque quanto menor volume de dados, menor é o volume de informações. Como cada autovoz k , com $k=1,2,...,K$, representa uma proporção da variação dos modelos, se utilizarmos uma dimensão maior, estaríamos fazendo estimativas péssimas para os

novos parâmetros incorretamente. Porém com o aumento do volume de dados e do espaço K , não implica em uma melhora no sistema. O sistema de Autovozes utilizou um volume de dados de aproximadamente 70% do seu volume de dados total, já alcança a máxima performance.

Na maioria dos sistemas de adaptação de Autovozes já implementados, sempre se trabalhou com um volume de mais de 100 locutores base. No nosso caso trabalhamos somente com 20 locutores e mesmo assim já obtivemos bons resultados.

Para esta técnica ainda não se conhece nenhum teste feito com uma base de dados maior, em torno de 1000 palavras. Acredito que para se utilizar esta técnica com um vocabulário deste tipo seria necessário o gasto de muito tempo de processamento para a definição dos Autovozes. Concluí que esta técnica tem grande eficácia para sistemas pequenos e que requerem uma adaptação rápida. Já para casos onde o vocabulário é amplo o uso da técnica MLLR é mais adequado.

9 REFERÊNCIAS BIBLIOGRÁFICAS

- [1] RABINER, L. *A tutorial on Hidden Markov Models and selected Applications in Speech Recognition*. Proceedings of the IEEE, vol 77. N°2, 1989, p.257-284.

- [2] JUANG, B.; RABINER, L. *Issues in Using Hidden Markov Models for Speech Recognition*. In: Advances in Speech Signal Processing. Ed. Furui, S.; Sondhi, M.; New York: Marcel Dekker, 1992, p.509-553.

- [3] YOUNG, S.; ET. AL. *The HTK book*. Microsoft Corporation. Version 3.0, <http://htk.eng.cam.ac.uk/>, 2000.

- [4] DELLER, J.; PROAKIS, J.; HANSEN, J. *Discrete-Time processing of Speech Signals*. Macmillan, New York, p63-66, 1993.

- [5] DAVIS, B.; MERMELSTEIN, P. *Comparision of parametric representations for monosyllabic word recognition in continuous spoken sentences*. . IEEE Acoustics, Speech and Signal Proceeding, Vol. 28, p357-366, 1980.

- [6] THE MATHWORKS, INC. *Matlab 5.3*, <http://www.mathworks.com/>, 1999.

- [7] NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY (NIST). *The DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus*, Virginia, <http://www.nist.gov/>, 1990.

- [8] CHRISTENSEN, H. *Speaker Adaptation of Hiden Markov Models using Maximum Likelihood Linear Regression*. Thesis, Aalborg University, Denmark, 1996.

- [9] LEGGETER, C. WOODLAND, P. *Speaker Adaptation of HMMs using Linear Regression*. Cambridge University, Technical Report, Cambridge, UK, 1994.

- [10] KUHN, H. ,et. al., *Eigenvoices for speaker adaptation*. Proc. of ICSLP-98, p1771-1774, Sydney, Australia, 1998.

- [11] CHEN C.S., HUO, K.S., *Karhunen-Loève method for data compression and speech synthesis*, IEEE Proceeding-I, vol 138, N°5, p377-380, 1991.
- [12] WOODLAND, P., *Speaker Adaptation: Techniques and Challenges*, Proceedings IEEE Automatic Speech Recognition and Understanding Workshop, p85-90, Colorado, 2000.
- [13] KUHN, H. ,*et. al.*, *Eigenfaces and Eigenvoices: Dimensionality reduction for specialized Pattern Recognition*. IEEE workshop on multimedia Signal Processing, California , 1998.
- [14] KUHN, H. ,*et. al.*, *Eigenvoices for speaker adaptation*. Internal technical report, STL, California, 1997.
- [15] FURUI, S., *Speaker-Independent and Speaker-Adaptive Recognition Techniques*, In: Advances in Speech Signal Processing. Ed. Furui, S., Sondhi, M., New York: Marcel Dekker, p.597-621,1992.
- [16] FAGUNDES, R., *Reconhecimento de voz, linguagem continua, usndo modelos de Markov*, Dissertação de Mestrado, Universidade de São Paulo, São Paulo, 1993.
- [17] WESTWOOD, R., *Speaker Adaptation using Eigenvoices*, Thesis, Cambridge University, Cambridge, 1999.
- [18] PAPOULIS, A., *Probability, random variables, and stochastic processes*, Singapore, McGraw-Hill, 1984.
- [19] RABINER, L. JUANG, B. *Fundamentals of speech recognition*. Prentice Hall, New York, 1993.
- [20] GREEN, P., CARROLL, P., *Analyzing Multivariate Data*, cap. 8-9. The Dryden Press, 1978.
- [21] JOLLIFFE, T., *Principal Component Analysis*, Springer-Verlag, New York, 1986
- [22] JOHNSON, R., WICHERN, D., *Applied Multivariate statical analysis*, Prentice Hall, Texas, 1988.
- [23] LEGGETER, C. *Improved Acoustic Modelling for HMMs using Linear Transformations*. PhD thesis, Cambridge University, Cambridge, 1995.

ANEXO A – ESTIMATIVA DA MATRIZ DE TRANSFORMAÇÃO

Para efetuarmos o cálculo da matriz de transformação que é necessária para aplicarmos a adaptação utilizando a técnica MLLR, é necessário otimizar a função auxiliar, Equação 3.14:

$$Q(\lambda, \hat{\lambda}) = \frac{1}{2} P(\mathbf{O} | \lambda) \sum_j \sum_t \gamma_j(t) \left[E \log(2\pi) \log |\boldsymbol{\Sigma}_j| + (\mathbf{o}_t - \hat{\boldsymbol{\mu}}_j)^T (\boldsymbol{\Sigma}_j)^{-1} (\mathbf{o}_t - \hat{\boldsymbol{\mu}}_j) \right] \quad (\text{A.1})$$

Sabendo que as médias adaptadas utilizando a técnica MLLR são dadas por:

$$\hat{\boldsymbol{\mu}}_j = \mathbf{W}_j \cdot \boldsymbol{\mu}_j \quad (\text{A.2})$$

onde $\boldsymbol{\mu}_j = [\omega \ \mu_1 \ \dots \ \mu_N]^T$ é o vetor de médias estendido. Como queremos maximizar a Equação (A.1), então devemos diferencia-la com relação a $\mathbf{W}_j$ e igualar a zero, ou seja:

$$\frac{\partial Q(\lambda, \hat{\lambda})}{\partial \mathbf{W}_j} = 0 \quad (\text{A.3})$$

Então

$$\frac{\partial}{\partial \mathbf{W}_j} Q(\lambda, \hat{\lambda}) = \frac{1}{2} P(\mathbf{O} | \lambda) \frac{\partial}{\partial \mathbf{W}_j} \sum_{t=1}^T \gamma_j(t) \left[E \log(2\pi) + \log |\boldsymbol{\Sigma}_j| + h(\mathbf{o}_t, j) \right] = 0 \quad (\text{A.4})$$

onde

$$h(\mathbf{o}_t, j) = (\mathbf{o}_t - \mathbf{W}_j \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1} (\mathbf{o}_t - \mathbf{W}_j \boldsymbol{\mu}_j) \quad (\text{A.5})$$

Diferenciando a Equação (A.5) temos:

$$\frac{\partial}{\partial \mathbf{W}_j} h(\mathbf{o}_t, j) = 2 \boldsymbol{\Sigma}_j^{-1} [\mathbf{o}_t - \mathbf{W}_j \boldsymbol{\mu}_j] \boldsymbol{\mu}_j^T \quad (\text{A.6})$$

Reescrevendo a Equação (A.4) temos:

$$\frac{\partial}{\partial \mathbf{W}_j} Q(\lambda, \hat{\lambda}) = P(\mathbf{O} | \lambda) \sum_{t=1}^T \gamma_j(t) \boldsymbol{\Sigma}_j^{-1} [\mathbf{o}_t - \mathbf{W}_j \boldsymbol{\mu}_j] \boldsymbol{\mu}_j^T = 0 \quad (\text{A.7})$$

e por fim teremos:

$$\sum_{t=1}^T \gamma_j(t) \boldsymbol{\Sigma}_j^{-1} \mathbf{o}_t \boldsymbol{\mu}_j^T = \sum_{t=1}^T \gamma_j(t) \boldsymbol{\Sigma}_j^{-1} \mathbf{W}_j \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T \quad (\text{A.8})$$

A Equação (A.8) é a forma geral de otimização da matriz $\mathbf{W}_j$, para o caso de cada distribuição ter a sua matriz de transformação linear associada. Já para o caso de uma adaptação global tem-se:

$$\sum_{t=1}^T \sum_{r=1}^R \gamma_{j_r}(t) \boldsymbol{\Sigma}_{j_r}^{-1} \mathbf{o}_t \boldsymbol{\mu}_{j_r}^T = \sum_{t=1}^T \sum_{r=1}^R \gamma_{j_r}(t) \boldsymbol{\Sigma}_{j_r}^{-1} \mathbf{W} \boldsymbol{\mu}_{j_r} \boldsymbol{\mu}_{j_r}^T \quad (\text{A.9})$$

Onde R é o número total de estados dos HMMs do sistema.