DISSERTAÇÃO DE MESTRADO

VERIFICAÇÃO AUTOMÁTICA DO LOCUTOR, DEPENDENTE DO TEXTO, UTILIZANDO SISTEMAS HÍBRIDOS MLP / HMM

MARCOS PAULO BARROS OLIVEIRA

INSTITUTO MILITAR DE ENGENHARIA

VERIFICAÇÃO AUTOMÁTICA DO LOCUTOR, DEPENDENTE DO TEXTO, UTILIZANDO SISTEMAS HÍBRIDOS MLP / HMM

POR

MARCOS PAULO BARROS OLIVEIRA

DISSERTAÇÃO SUBMETIDA

COMO REQUISITO PARCIAL

PARA A OBTENÇÃO DO GRAU DE

MESTRE EM CIÊNCIAS

EM ENGENHARIA ELÉTRICA

Assinatura do Orientador da Dissertação

_____

Cel. QEM Sidney Cerqueira Bispo dos Santos – D.C.

Assinatura do Co-orientador da Dissertação

_____

Cel. R/1 Roberto Miscow Filho – M.C.

Rio de Janeiro - RJ

Agosto de 2001

Dissertação apresentada por:

_____
Marcos Paulo Barros Oliveira

E aprovada pelos Srs.:

_____
Cel. QEM Sidney Cerqueira Bispo dos Santos – D.C.

_____
Cel. R/1 Roberto Miscow Filho – M.C.

_____
Cel. R/1 Antônio Carlos Gay Thomé – Ph.D.

_____
TC. QEM José Antônio Apolinário Júnior – D.C.

_____
Sérgio Lima Netto – Ph.D.

IME, RIO DE JANEIRO – RJ, 08 de agosto de 2001.

“Dedico esta obra aos meus pais, Remi e Maria, pela formação ética e moral que me passaram.

À Vanessa, minha mulher, por todo carinho, atenção e compreensão.

E também, à pequena Isabelle, minha sobrinha, que veio ao mundo para torná-lo mais alegre”

AGRADECIMENTOS

A Deus, que ilumina minha vida e me dá forças, sempre, para continuar minha jornada.

Ao Cel. QEM Sidney Cerqueira Bispo dos Santos – D.C. pela indicação do tema da dissertação, pela orientação segura e pelos conhecimentos que me transmitiu.

Ao Cel. R/1 Roberto Miscow Filho – M.C. pela experiência e ensinamentos transmitidos e pela correção técnica do compêndio desta dissertação.

A Remi Ribeiro Oliveira, meu pai, pelos conselhos, pelo apoio, pela alegria e felicidade que sempre me transmitiu.

À Maria Barros Oliveira, minha mãe, pelo apoio e carinho transmitidos.

À Vanessa, minha mulher, por estar sempre ao meu lado.

Aos meus irmãos Ricardo, Daniela e Viviane pela amizade e incentivo.

À toda minha família que torceu por mim.

Ao colega da área de processamento de sinais Cap. QEM Dirceu Gonzaga da Silva pela cooperação e sugestões na elaboração deste trabalho

À professora Maria de Fátima Santos Farias – D.C. e ao professor José Ivan Carnaúba Accioly – M.C. pelo apoio e incentivo.

Ao amigo Mário Jorge pelo companheirismo e incentivo.

A todos os locutores que participaram do desenvolvimento do sistema.

Ao departamento de Engenharia Elétrica do IME (DE-3), agradeço a pronta cooperação oferecida pelo corpo docente e funcionários.

À CAPES pelo apoio financeiro proporcionado.

RESUMO

É proposto um sistema híbrido MLP/HMM para verificação automática do locutor dependente do texto.

A base de dados é formada por 41 locutores, 24 para treinamento e 17 para teste, a frase utilizada foi: "Amanhã Ligo de Novo".

As características de voz utilizadas foram 12 coeficientes mel-cepestrais, log-energia e seus deltas, totalizando 26 características.

Desenvolveram-se dois sistemas híbridos: no primeiro utilizaram-se as MLP's para gerar as probabilidades de saída para o HMM; no segundo, a duração de estados e a verossimilhança, fornecidas pelo HMM, foram usadas para treinar uma MLP, obtendo-se melhor resultado neste último sistema, onde o EER foi de 4,72. O desempenho deste sistema superou tanto a MLP quanto o HMM em todos os testes realizados.

ABSTRACT

It is proposed a hybrid architecture MLP/HMM for text dependent speaker verification.

The database is formed for 41 (Brazilian portuguese) speakers, 24 for training and 17 for test. The used sentence was “Amanhã Ligo de Novo”.

The features used were 12 mel-cepstra coefficients, log energy and their temporal derivates (delta), totaling 26 features.

Two hybrid systems were developed: in the first system a MLP was used to estimate the emisson probabilities for the HMM; in the second system the state duration and de likelihood, supplied by HMM, were used to train a MLP, and the results were better in this last case, where the EER was 4,72. The performance for this hybrid was higher than both the MLP and the HMM for all tests.

SUMÁRIO

LISTA DE ABREVIATURAS E SÍMBOLOS

A/D	Analógico – Digital
ANN	"Artificial Neural Network"
DFT	"Discrete Fourier Transform"
DTW	“Dynamic Time Warping”
EER	“Equal Error Rate”
FA	Falsa Aceitação
FR	Falsa Rejeição
HMM	"Hidden Markov Models"
IAL	Identificação Automática do Locutor
MLP	“Multilayer Perceptrons”
RAL	Reconhecimento Automático de Locutor
RNA	Redes Neurais Artificiais
RSR	Razão Sinal Ruído
VAL	Verificação Automática de Locutor
LPC	"Linear Predictive Coefficients"
MLE	"Maximum Likelihood Estimates"
MSE	“Mean Squared Error”
IME	Instituto Militar de Engenharia
RBF	“Radial Basis Function”
MAP	“Maximum *a Posteriori* Probabilities”

CAPÍTULO 1

INTRODUÇÃO

1.1 PRINCÍPIOS DE RECONHECIMENTO DE LOCUTOR

O crescente desenvolvimento da computação de alta velocidade, a grande capacidade de armazenamento e o uso de algoritmos eficientes têm possibilitado o desenvolvimento de técnicas cada vez mais sofisticadas na área de processamento de voz e em especial em *reconhecimento automático do locutor* (RAL).

No RAL determina-se a identidade de uma pessoa através da voz, com o propósito de restringir o acesso a redes, computadores, base de dados bem como restringir informações confidenciais às pessoas não autorizadas dentre outras aplicações. Um sistema que trabalha com RAL calcula a similaridade entre as características da voz do locutor que se deseja reconhecer com as características armazenadas previamente pelo sistema. O RAL divide-se em *identificação automática do locutor* (IAL) e *verificação automática do locutor* (VAL). Na IAL é determinado a qual dos N locutores treinados pertence uma dada elocução teste. A identificação pode ocorrer com rejeição ou sem rejeição. Neste último classifica-se o locutor como verdadeiro se a similaridade da elocução teste for maior que um limiar; caso contrário considera-se o locutor como falso. Na verificação do locutor aceita-se ou rejeita-se a identidade pretensa de um locutor teste. Podemos observar, então, que a rejeição é uma característica intrínseca da verificação. Podem ocorrer dois tipo de erros na verificação: *falsa aceitação* (FA), aceitação de um locutor impostor ou mímico e

falsa rejeição (FR), rejeição do locutor verdadeiro (ATAL, 1976), (ROSEMBERG, 1976).

A Figura 1.1 mostra as diversas formas de reconhecimento.

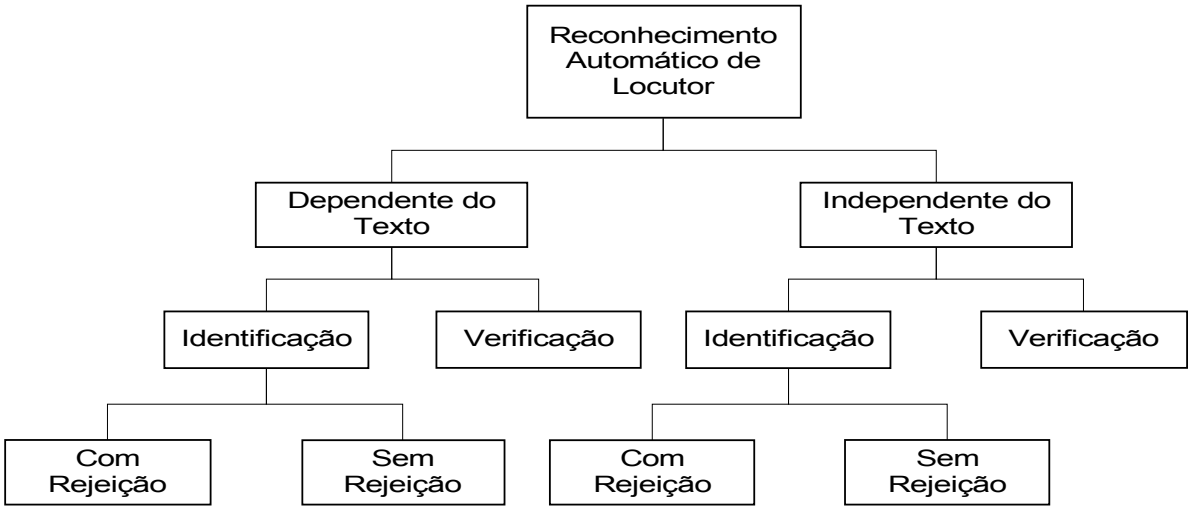


FIGURA 1.1: Diversas formas de Reconhecimento de Locutor (PARANAGUÁ, 1997).

Uma diferença importante existente entre identificação e verificação é o número de decisões a serem tomadas. Na identificação, o número de decisões (saídas do sistema de reconhecimento) é igual ao número de locutores que se deseja identificar (mais 1 devido à rejeição); na verificação existem somente duas: aceitação ou rejeição. Com isso, o desempenho da identificação diminui com o aumento da população, e o desempenho na verificação se aproxima de uma constante, independente do tamanho da população (ATAL, 1976), (ROSEMBERG, 1976).

O RAL pode ser realizado utilizando-se locuções de texto fixo (dependentes do texto) ou locuções de texto livre (independentes do texto) (BEZERRA, 1994). O desempenho do reconhecimento torna-se melhor em aplicações com locuções dependentes do texto. O modelo de reconhecimento para este caso é mais simples, porque não há necessidade de lidarmos com a variabilidade adicionada pela diferença entre as locuções desconhecidas e de referência (ATAL, 1976).

No RAL um conhecimento específico das características da voz (padrões) é imprescindível para obtermos resultados satisfatórios, podemos dizer que aquelas que procuram simular atividades orgânicas, ou seja, que se aproximam da audição humana, têm um desempenho melhor na tarefa do reconhecimento (RABINER, 1993).

Neste compêndio serão utilizados três modelos para verificação do locutor bastante difundidos no meio científico. São eles: HMM ("Hidden Markov Models" - Modelos de Markov Escondidos), ANN ("Artificial Neural Networks" - Redes Neurais Artificiais), utilizou-se nesse compêndio as MLP's ("Multilayer Perceptrons"), e ANN / HMM (Modelo Híbrido).

1.2 OBJETIVOS DA DISSERTAÇÃO

Os objetivos desta dissertação foram: fazer uma comparação do desempenho das MLP's e do HMM, para a tarefa de verificação automática do locutor dependente do texto, identificando os pontos favoráveis e desfavoráveis de cada um deles; utilizar este dois modelos para implementar um terceiro, sistema híbrido, que tenha um desempenho melhor que as MLP's e o HMM para esta mesma tarefa.

1.3 DIVISÃO DO COMPÊNDIO

Este compêndio compõe-se de sete capítulos. O Capítulo 1 fornece uma visão geral de RAL. No Capítulo 2 são descritos os métodos de extração das características das elocuções. No Capítulo 3 descrevem-se os princípios básicos do HMM e das MLP's. No Capítulo 4 é realizado um estudo sobre o primeiro sistema híbrido implementado. O Capítulo 5 trata da forma de implementação dos sistemas. No Capítulo 6 são mostrados os resultados. O Capítulo 7 apresenta as conclusões e propostas para pesquisas futuras.

CAPÍTULO 2

ATRIBUTOS UTILIZADOS NO RAL

2.1 – INTRODUÇÃO

A extração de atributos da voz, para obtermos resultados satisfatórios em RAL, deve fornecer um conjunto de parâmetros que represente informações da voz que são dependentes do locutor (ATAL, 1976). Um conjunto de características desejáveis dos parâmetros para o reconhecimento apropriado do locutor está relacionado abaixo (WOLF, 1972):

1 – Eficiente na representação de informações dependentes do locutor;

2 – Fácil de medir;

3 – Estável no tempo;

4 – Ocorrer naturalmente e freqüentemente na voz;

5 – Pouca mudança de um ambiente de gravação para outro;

6 – Não susceptível à mímica.

Atualmente já se consegue satisfazer a estas exigências para os parâmetros da voz. Existem, porém, outros fatores importantes que limitam o desempenho do RAL, dentre os quais podemos citar as variabilidades encontradas nas elocuções. Essas variabilidades apresentam-se da seguinte forma (RABINER, 1994):

1 – Variabilidade dos sons, tanto em elocuções de um mesmo locutor como em elocuções de diferentes locutores;

2 – Variabilidade do canal de gravação, devido, por exemplo, ao tipo de microfone utilizado;

3 – Variabilidade devida ao ruído ambiente;

4 – Variabilidade na produção da fala, por exemplo, estalos labiais, ruído da respiração, hesitação ao falar.

Estas fontes de variabilidade geralmente não podem ser eliminadas, logo precisam ser modeladas pela tecnologia de reconhecimento. Baseado nisso existem duas tarefas principais que um sistema de reconhecimento deve realizar (RABINER, 1994):

1 – Detecção da voz, isto é, cortar os silêncios de fundo antes e depois da entrada de voz fornecida pelo locutor;

2 – Reconhecimento da voz de entrada, baseada em tecnologia de reconhecimento de padrões (incluindo métodos determinísticos e estatísticos), métodos fonéticos-acústicos ou métodos baseados em redes neurais.

O processamento digital do sinal de voz, para extração de suas características, pode ser dividido em três operações básicas: modelamento espectral, análise espectral e transformações paramétricas (PICONE, 1993), conforme mostra a Figura 2.1.

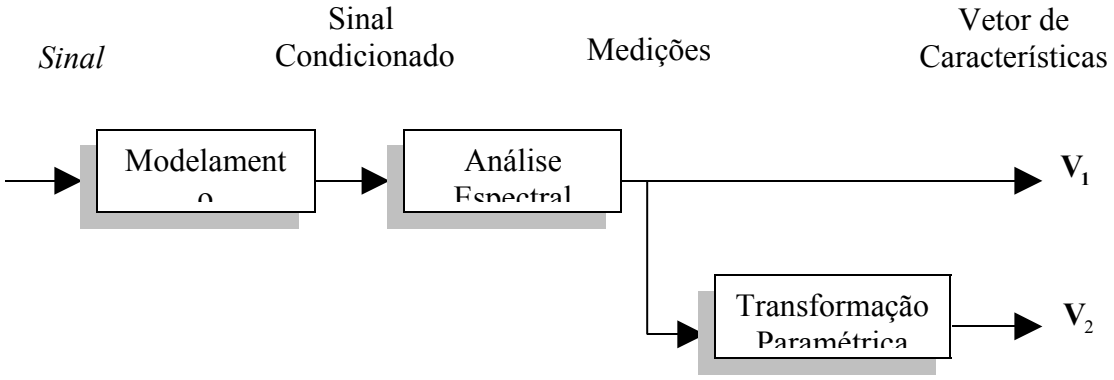


FIGURA 2.1: Sistema de processamento digital do sinal de voz para extração de suas características.

Na modelagem espectral há uma filtragem digital que enfatiza componentes de freqüências importantes. Na análise espectral são retiradas as caraterísticas (vetor $\mathbf{V_1}$) e finalmente na transformação paramétrica é capturada a dinâmica espectral (PICONE, 1993) de cada características do vetor $\mathbf{V_1}$ resultando, então, no vetor $\mathbf{V_2}$. A seguir descreveremos cada uma das três operações básicas.

2.2 – MODELAGEM ESPECTRAL

2.2.1 Conceitos Básicos

Há duas operações básicas no modelamento espectral: *conversão A/D* que implica converter um sinal analógico em digital; e *filtragem digital* que enfatiza importantes freqüências do sinal (MAKHOUL, 1975), Figura 2.2.

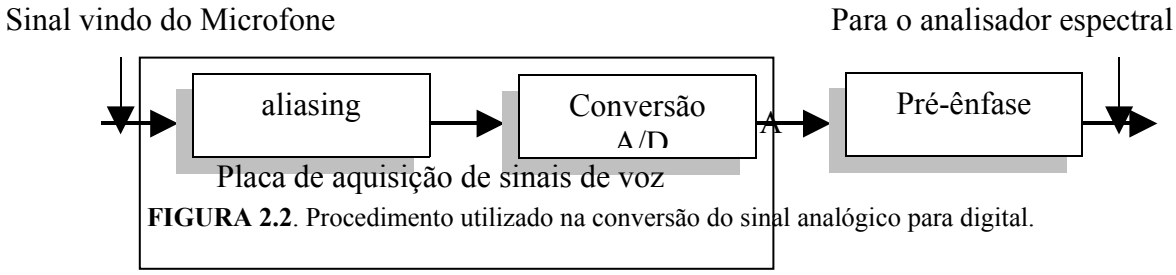


FIGURA 2.2. Procedimento utilizado na conversão do sinal analógico para digital.

Na filtragem anti-aliasing o sinal é limitado em freqüência evitando-se, assim, a ocorrência de aliasing (superposição espectral) na representação espectral do sinal de voz no domínio do tempo discreto. Na conversão A/D o sinal é amostrado (obedecendo-se, é claro, a taxa de Nyquist) e quantizado; com isso, o sinal de voz já pode ser tratado por uma máquina digital. O principal propósito do processo de digitalização é produzir uma representação digital do sinal com uma relação sinal-ruído (RSR) a mais alta possível, que para aplicações em reconhecimento de voz e locutor deve ser de aproximadamente 30 dB, suficiente para não prejudicar o desempenho do reconhecedor (PICONE, 1993).

O próximo passo agora é realizar uma filtragem digital, através de um filtro conhecido como filtro de *pré-ênfase*, cuja a resposta em freqüência é (PICONE, 1993):

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

os valores típicos de a_{pre} variam entre –1.0 e –0.4. A pré-ênfase proporciona um ganho de aproximadamente 20 dB por década (PICONE, 1993). Há dois motivos para usarmos este filtro: como o sinal de voz tem uma atenuação de aproximadamente 20 dB por década, devido a características fisiológicas do sistema de produção de voz, o filtro de pré-ênfase serve para compensar esta atenuação antes da análise espectral, melhorando então a eficiência da análise (RABINER, 1978); sendo a audição mais sensível a freqüências acima de 1 kHz do espectro, a pré-ênfase amplifica esta área do espectro, auxiliando os algoritmos de análise espectral na modelagem do aspectos mais perceptualmente importante do espectro da voz (RABINER, 1978).

2.2.2 Determinação dos Pontos Extremos

Após a conversão A/D é realizada a determinação dos pontos extremos ("endpoints"), ponto A na Figura 2.2. Este procedimento visa fazer a detecção da voz, ou seja: encontrar, no arquivo, onde começa e onde termina a voz. As principais vantagens em se determinar os "endpoints" são:

- Redução do tempo de processamento, já que só o sinal de voz será processado pelo sistema de reconhecimento;

- Evitar que o ruído de fundo que ocorra antes e depois do sinal de voz prejudique o desempenho do reconhecimento (RABINER, 1993).

O método de determinação dos pontos extremos utilizado no trabalho descrito neste compêndio baseou-se no fato de que os atributos retirados de janelas adjacentes do sinal de voz (coeficientes Mel-Ceps ou Mel-PLP, por exemplo) têm uma variação maior do que os retirados de janelas adjacentes do ruído de fundo. São calculados as médias e os desvios

padrões de cada atributo ao longo das cinqüenta janelas iniciais de 20 milisegundos; considera-se que estas janelas sejam formadas apenas por ruído de fundo. Com esses valores chega-se a um limiar determinado pela distância euclidiana entre um vetor *M+ (M mais),* formado pelas médias mais os desvios, e um vetor *M- (M menos)* formado pelas médias menos os respectivos desvios padrões (MARKEL, 1980). Caso a distância entre duas janelas do restante da gravação seja maior que este limiar, estas janelas serão candidatas a representar sinais relevantes de voz. Este procedimento é realizado no começo da gravação (determinação do ponto inicial) e no final da gravação (determinação do ponto final). Para maiores detalhes deste método de determinação dos pontos extremos consulte (ANDRADE, 1999).

2.3 – ANÁLISE ESPECTRAL

2.3.1 Janelamento

Um passo muito importante antes de iniciarmos a extração das características da voz é o janelamento. O janelamento do sinal consiste em multiplicarmos o sinal por uma janela como mostra equação 2.2:

$$s_w[n] = s[n] \bullet w[n] \qquad (2.2)$$

onde, $s[n]$ é o sinal de voz, $w[n]$ a janela e $s_w[n]$ o sinal janelado, todos em tempo discreto (o uso de colchetes e parênteses será usado para representar um sinal ou seqüência em tempo discreto e contínuo, respectivamente). A janela mais usada em reconhecimento de voz é a janela de Hamming (PICONE, 1993) dada pela equação abaixo:

$$w[n] = \begin{cases} 0.54 - 0.46\cos[2\pi n / M], & 0 \le n \le M, \\ 0, & \text{caso contrário} \end{cases} \qquad (2.3)$$

onde, *M*+1 é o tamanho da janela. Pode-se perceber, pela equação 2.3, que o janelamento extrai um pequeno trecho (de largura *M*+1) do sinal de voz. A largura da janela deve estar no intervalo de 10 a 40 ms; será citado mais adiante o motivo para tal. A Figura 2.3 mostra uma janela de Hamming com 220 amostras (M = 220). Analisando a Figura 2.3 nota-se que esta janela atenua suavemente o sinal em direção às suas bordas (à medida que afasta-se do ponto central *M*/2). As principais vantagens do janelamento do sinal de voz são as seguintes:

- Como o sinal de voz é um processo estocástico não estacionário e o trato vocal muda lentamente na voz contínua (OPPENHEIM, 1989), o trecho de voz, obtido pelo janelamento, pode ser considerado um processo estacionário, se esse trecho tiver uma duração de 10 a 40 milisegundos.

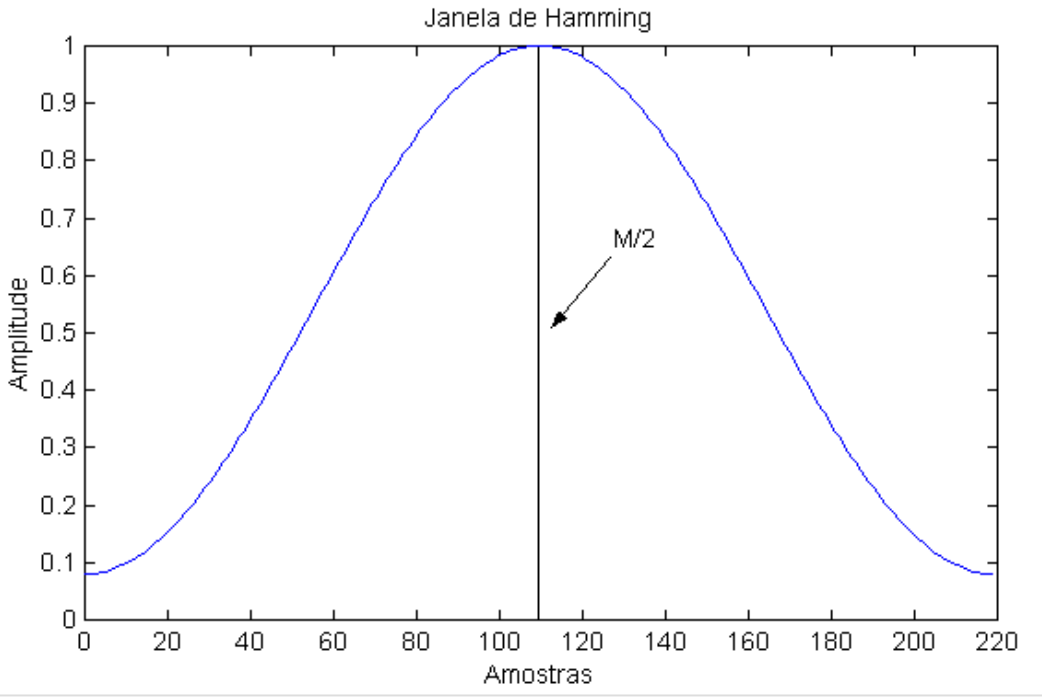


FIGURA 2.3: Janela de Hamming com 220 amostras (M = 220).

- O truncamento abrupto do sinal iria causar o fenômeno de Gibbs ("ripple" em amplitude na resposta em freqüência da janela) (OPPENHEIM, 1989), (STRUM, 1988); a janela de Hamming atenua este problema.

- A superposição entre janelas aumentará a correlação entre janelas sucessivas, evitando variações bruscas entre "features" extraídas de janelas adjacentes (RABINER, 1978).

2.3.2 – Escala Mel

Experimentos no campo da psicoacústica mostraram que a percepção humana de componentes de freqüência de sons, de tons puros ou de sinais de voz, não seguem uma escala linear (RABINER, 1993). Então, para cada freqüência *f*, medida em Hz na escala linear, faz-se um mapeamento em uma escala chamada *escala mel* que corresponde à real percepção do sistema auditivo humano. Como ponto de referência, uma freqüência de 1 kHz, 40 dB acima do limiar da percepção auditiva, é definida como 1000 mels (RABINER, 1993).

Outra importante conclusão da psicoacústica é a de que freqüências de um som dentro de certa largura de faixa (banda) de uma freqüência não podem ser individualmente identificadas. Quando uma dessas componentes de freqüência do som sai fora desta banda, esta componente pode ser individualmente distinguida. Esta banda é conhecida como banda crítica (PICONE, 1993).

A equação 2.4, abaixo, realiza o mapeamento da freqüência acústica *f* para a freqüência mel. A Figura 2.4 mostra este mapeamento.

$$f_{mel} = 2595 \bullet \log_{10}(1 + f / 700) \qquad (2.4)$$

A escala mel é aproximada como escala linear de 0 a 1000 Hz, e logarítmica além de 1000 Hz.

A escala mel pode ser considerada, pois, uma transformação da escala de freqüência real em uma escala perceptual que se aproxima do modo como as freqüências são percebidas pelo ouvido humano. A utilização da escala mel melhora o desempenho do sistema de reconhecimento (DAVIS, 1980).

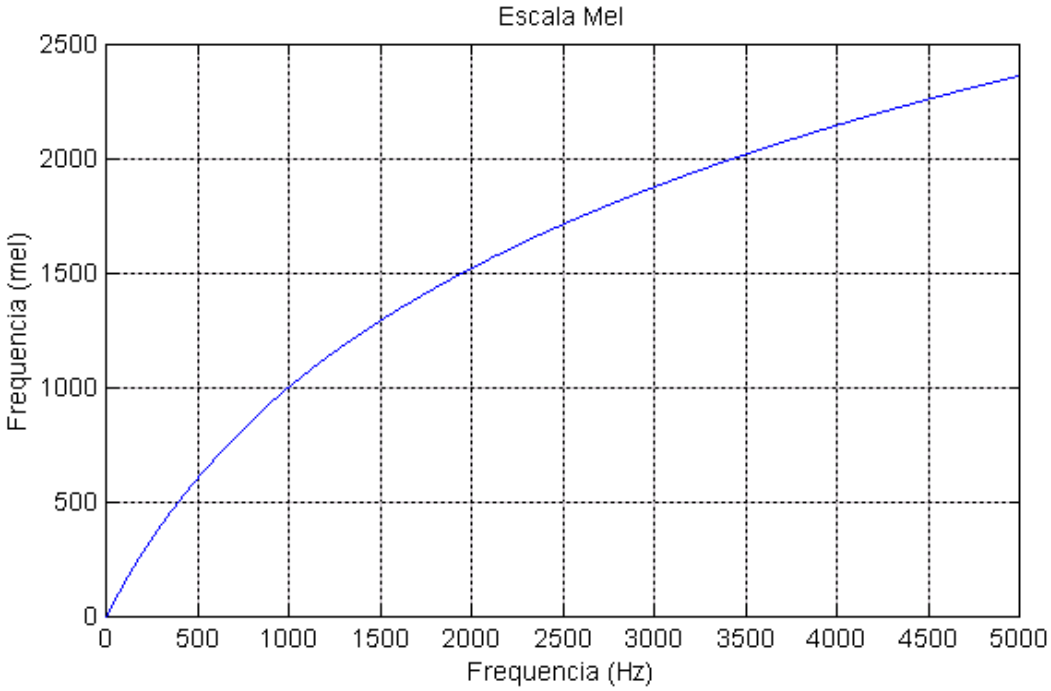


FIGURA 2.4: Mapeamento entre a escala de freqüência real e a escala mel.

A escala mel é implementada com um *banco de filtros de banda crítica* (filtros triangulares passa faixa) cuja larguras de faixa são escolhidas para serem iguais a larguras de faixa da banda crítica para a correspondente freqüência central (PICONE, 1993).

A freqüência central de cada filtro triangular é obtida pela seguinte equação:

$$F_{c,i} = k_i \cdot \frac{f_s}{N} \qquad (2.5)$$

onde, f_s é a freqüência de amostragem, N o número de pontos usado no cálculo da transformada discreta de Fourier (DFT) e k_i é o ponto da DFT correspondente à freqüência central de cada filtro.

Nas implementações feitas para esta dissertação utilizaram-se 20 filtros triangulares passa-faixa.

2.3.3 – Coeficientes Cepestrais

Sistemas homomórficos são muito úteis em processamento de voz porque possibilitam a separação da fonte de excitação (trem de impulsos quase periódicos ou sinal de ruído aleatório) da forma do trato vocal (OPPENHEIM, 1989), (PICONE, 1993). No modelo de produção da voz o espectro da voz, obtido através da transformada de Fourier, consiste no sinal de excitação filtrado por um filtro linear variante no tempo representando a forma do trato vocal, como mostra Figura 2.5.

Fonte de excitação
p[n]

Sinal de voz
s[n]

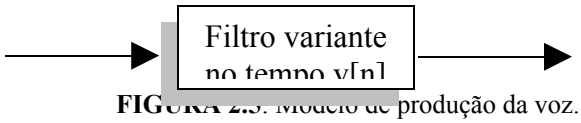


FIGURA 2.5: Modelo de produção da voz.

Como a forma do trato vocal varia de maneira relativamente lenta no tempo, então o trato pode ser modelado por um filtro também variando lentamente no tempo (OPPENHEIM, 1989). Assim, o sinal de voz janelado, pode ser modelado por um filtro invariante no tempo para intervalos da ordem de 10 à 40 ms.

O processo de separação das duas componentes é chamado de *deconvolução*, porque o sinal de voz é formado pela convolução da fonte de excitação com o trato vocal.

Sendo assim, podemos escrever um sinal de voz $s[n]$ formado pela convolução de $p[n]$ e $v[n]$ como:

$$s[n] = p[n] * v[n] \tag{2.12}$$

onde, $p[n]$ é o sinal de excitação, $v[n]$ é a resposta ao impulso do trato vocal e "$*$" representa a convolução.

Aplicando a transformada de Fourier na equação 2.12, temos:

$$S(w) = P(w) \cdot V(w) \tag{2.13}$$

Aplicando o logaritmo complexo em ambos os lados, temos

$$\begin{aligned} \log(S(w)) &= \log(P(w) \cdot V(w)) \\ &= \log(P(w)) + \log(V(w)) \end{aligned} \tag{2.14}$$

Logo, a excitação e o trato vocal tornaram-se linearmente combinadas, podendo ser agora mais facilmente separadas.

Aplicando, agora, a transformada inversa de Fourier, temos

$$\begin{aligned} \mathrm{F}^{-1}\{\log(S(w))\} &= \mathrm{F}^{-1}\{\log(P(w))\} + \mathrm{F}^{-1}\{\log(V(w))\} \\ C_s[n] &= C_p[n] + C_v[n] \end{aligned} \tag{2.15}$$

ou seja: os coeficientes, $C_p[n]$, da excitação e do trato vocal, $C_v[n]$, encontram-se separados no domínio da "quefrency" (SCHAFER, 1975), podendo-se, então, separá-los com o uso de um "lifter" $l[n]$, análogo a um filtro no domínio da freqüência. $C_v[n]$ é composta de componentes que variam lentamente devido à resposta ao impulso do trato vocal, logo, responsáveis pela produção de coeficientes de baixa "quefrency", e $C_p[n]$ é composta de componentes que variam rapidamente devido à função de excitação, responsáveis pela produção dos coeficientes de alta "quefrency" (SHAFER, 1975). A Figura 2.6 mostra o procedimento descrito acima para a análise cepestral.

Na Figura 2.6 (a) temos 330 amostras, aproximadamente 30 ms de duração, de um sinal de voz correspondente ao fonema /*a*/ amostrado a 11025 Hz. A Figura 2.6 (b) mostra o sinal janelado por uma janela de Hamming de 330 pontos. Na Figura 2.6 (c) temos os coeficientes $C_s[n]$ e finalmente na Figura 2.6 (d) temos os coeficientes $C_v[n]$ obtidos pelo "Low-Time Lifter" (liftragem passa-baixo no domínio da "quefrency") de 12 pontos aplicado no cepestro. Observa-se na Figura 2.6 (a) a natureza estacionária do sinal de voz em um curto intervalo de tempo (30 ms). Os coeficientes $C_s[n]$ encontrados por meio das equações 2.12 à 2.15 são conhecidos como coeficientes cepestrais derivados da transformada de Fourier.

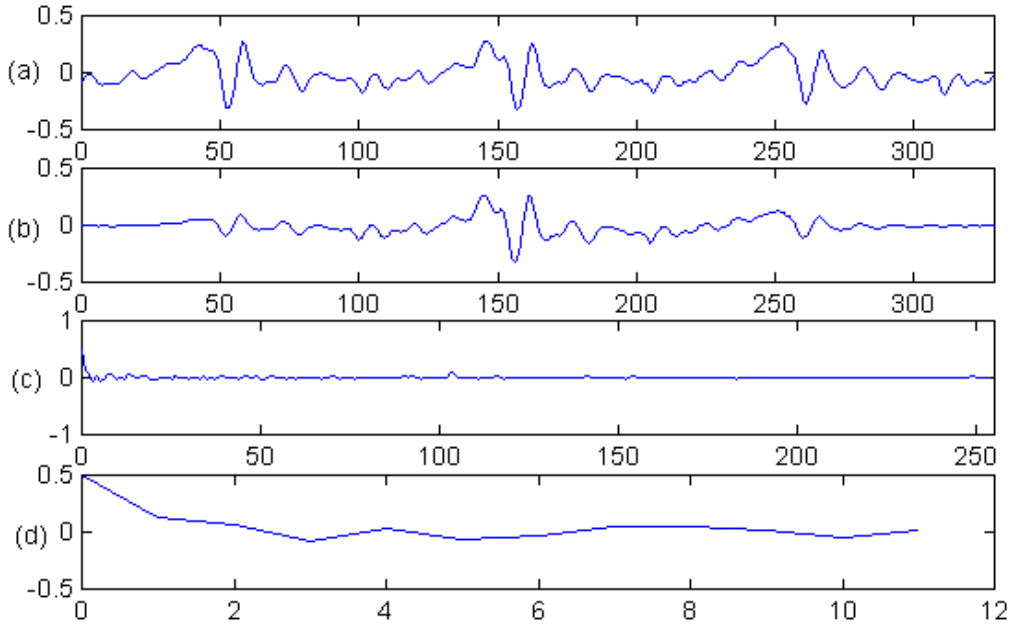


FIGURA 2.6: Análise Cepestral.**(a)** Sinal de Voz. **(b)** Sinal de voz janelado por um janela de Hamming. **(c)** Coeficientes $C_s[n]$. **(d)** Coeficientes $C_v[n]$.

2.3.4 – Coeficientes LPC

A idéia básica no modelo LPC é que um sinal de voz, $s[n]$, pode ser aproximado como uma combinação linear de suas p amostras passadas (RABINER, 1993), (PICONE, 1993), (MAKHOUL, 1975), ou seja:

$$s[n] = \sum_{i=1}^{p} a_i s[n-i] + e[n], \qquad (2.16)$$

onde, p representa o número de coeficientes do modelo (ordem do preditor), $\{a_i\}$ é o conjunto dos coeficientes de predição linear e $e(n)$ representa o erro no modelo (a diferença entre o valor predito e o valor real do sinal) (PICONE, 1993). Uma característica importante do modelo de predição linear é que este modela o espectro do sinal como um espectro suavizado (SANTOS, 1997).

Aplicando a transformada Z na equação 2.16 temos:

$$E(z) = A(z)S(z) \qquad (2.17)$$

onde, $E(z)$ e $S(z)$ são as transformadas Z do erro e do sinal de voz, respectivamente, sendo $A(z)$ dado por:

$$A(z) = \frac{E(z)}{S(z)} = 1 - \sum_{i=1}^{p} a_i z^{-i} \qquad (2.18)$$

sendo $1/A(z)$ um modelo só de pólos.

O objetivo básico do método LPC é achar um conjunto de coeficientes do preditor que minimize o erro médio quadrático da predição sobre um segmento curto de sinal de voz (sinal janelado) (RABINER, 1993). O LPC aplicado em processamento de voz possibilita uma boa modelagem do sinal de voz, porque o modelo só de pólos fornece uma boa aproximação da envoltória espectral do trato vocal (RABINER, 1993).

Existem três maneiras básicas de se calcular os coeficientes do preditor: método da covariância, método da autocorrelação e método treliça. Em reconhecimento de voz o método da autocorrelação é o mais utilizado (PICONE, 1993) devido à existência de algoritmos eficientes para sua implementação (algoritmo de Levinson-Durbin) e devido ao fato de que os coeficientes gerados por este método resultam em filtros estáveis (PICONE, 1993).

2.3.5 – Coeficientes Mel-Cepestrais e Delta Mel-Cepestrais Derivados do LPC

Em reconhecimento de locutor os coeficientes cepestrais derivados da DFT têm o mesmo desempenho que os derivados da análise LPC (PICONE, 1993). Apesar dos coeficientes cepestrais calculados pela DFT serem mais robustos em ambientes ruidosos do que os calculados pela LPC, um bom motivo para usarmos a técnica de predição linear em vez da transformada de Fourier, para o cálculo dos coeficientes mel-cepestrais, é que esta última gasta o dobro do tempo para o cálculo (PICONE, 1993).

Após obtermos a energia E_k, seção 2.3.6, dos k filtros aplicados no espectro, os coeficientes mel-cepestrais são encontrados utilizando-se a equação abaixo (RABINER, 1993):

$$MCLPC_i = \sum_{k=1}^{20} E_k \cos[\frac{i(k - 0.5)\pi}{20}], \qquad i = 1,2,...,M \tag{2.19}$$

Após obtermos os coeficientes mel-cepestrais aplicamos a equação 2.22 para obter os parâmetros delta de cada um dos coeficientes mel-cepestrais calculados.

2.3.6 – Energia

A energia de tempo de curto de um sinal é definida como :

$$E(n) = \sum_{m=0}^{N-1} (x[m] \bullet w[n - m])^2 \tag{2.20}$$

onde $x[m]$ é um trecho do sinal de voz e w[n-m] é uma janela, no nosso caso uma janela de Hamming, de largura N. O somatório do quadrado na equação 2.20 torna-a muito sensível a largas variações no nível do sinal em amostras consecutivas. Este problema pode ser evitado utilizando-se a função magnitude média, definida como (SHAFER, 1975):

$$M(n) = \sum_{m=0}^{N-1} | x[m] \bullet w[n - m] | \tag{2.21}$$

A magnitude média é computada como o somatório dos valores absolutos das amostras de $x[m]$. A magnitude média é utilizada em RAV para distinguirmos segmentos sonoros dos surdos no sinal de voz.

Utiliza-se o logaritmo sobre os parâmetros da energia no reconhecimento de voz para obter uma compressão entre a baixa energia e a alta energia (RABINER, 1993).

Os parâmetros delta energia são calculados aplicando-se a equação 2.22 nos coeficientes da energia. A Figura 2.7 mostra a energia e a magnitude de tempo curto para um sinal de voz calculados utilizando-se 170 janelas de Hamming com duração de 20 ms. Na Figura 2.7 (a) temos a forma de onda da frase “Amanhã ligo de novo”, e nas Figuras 2.7 (b) e (c) energia e magnitude, respectivamente, de tempo curto das 170 janelas.

2.4 – TRANSFORMAÇÕES PARAMÉTRICAS

As derivadas no tempo dos parâmetros (transformações paramétricas) do sinal possibilitam uma melhor caracterização das variações temporais no sinal (PICONE, 1993).

Em processamento digital de sinais a derivada de primeira ordem pode ser aproximada da seguinte forma:

$$\hat{s}[n] \equiv \frac{d}{dt} s[n] \approx s[n] - s[n-1] \qquad (2.22)$$

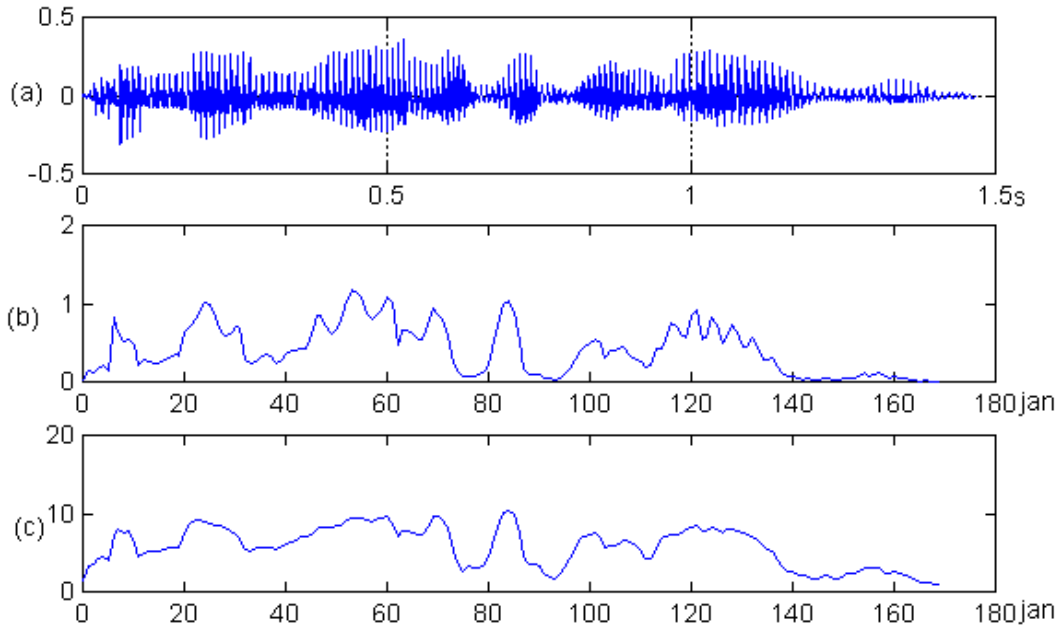


FIGURA2.7: Energia e magnitude de tempo curto para um sinal de voz. **(a)** forma de onda da frase "Amanhã ligo de novo". **(b)** e **(c)** energia e magnitude de tempo curto das 170 janelas de Hamming.

Este processo de diferenciação, derivada de primeira ordem, é conhecido como parâmetro delta. A derivada de segunda ordem (parâmetro delta-delta) é obtida reaplicando a equação 2.22 na saída do diferenciador de primeira ordem.

2.5 – SELEÇÃO DOS ATRIBUTOS MAIS RELEVANTES

Um aspecto muito importante no RAL e RAV é a escolha das características mais relevantes do sinal de voz. Um conjunto de características se torna eficiente na discriminação de locutores se as distribuições das elocuções dos diferentes locutores no espaço de características estão concentradas em localizações largamente afastadas (ATAL, 1976). A seleção de características visa a diminuição do esforço computacional com perda mínima de informação.

A razão *F* foi utilizada para a tarefa de seleção de características e é definida como:

$$F = \frac{\text{variância entre locutores}}{\text{variância intra locutores}} = \frac{\left\langle \left[\mu_i - \bar{\mu} \right]^2 \right\rangle_i}{\left\langle \left[x_\alpha^i - \mu_i \right]^2 \right\rangle_{\alpha,1}} \qquad (2.23)$$

onde, x_α^i é o valor do parâmetro da α -ésima repetição de uma elocução falada pelo i-ésimo locutor, $\langle \ \rangle_i$ indica média sobre os locutores, $\langle \ \rangle_\alpha$ indica média sobre diferentes elocuções de um locutor,

$$\mu_i = \left\langle x_\alpha^i \right\rangle_\alpha \qquad (2.24)$$

é a média estimada do parâmetro para o i-ésimo locutor, e

$$\bar{\mu} = \left\langle \mu_i \right\rangle_i \qquad (2.25)$$

é a média estimada sobre o valor médio do parâmetro para todos os locutores.

De acordo com a equação 2.23, quanto maior a razão *F* (maior variância entre locutores e menor variância intra locutores), para uma característica, maior será seu poder de discriminação de locutores.

CAPÍTULO 3

HMM E REDES NEURAIS ARTIFICIAIS (RNA)

3.1 – INTRODUÇÃO

Este capítulo abordará os princípios básicos de HMM e Redes Neurais colocando em evidência suas principais vantagens e desvantagens em reconhecimento de voz.

3.2 – HMM

3.2.1 – Introdução

Modelos de Markov Escondidos (HMM) são largamente usados no reconhecimento automático da voz e do locutor porque manipulam muito bem os aspectos estatísticos e seqüências do sinal de voz. Seu uso causou uma inovação na área do RAV (BOURLARD, 1990). O HMM modela a seqüência de *vetores característicos* extraídos de intervalos de duração curta da voz. Isto é, uma elocução é modelada como uma sucessão de estados, com transição instantânea entre esses estados. Neste caso, o HMM é definido como uma máquina estocástica de estados construída a partir de um conjunto de K estados $Q = \{q_1, q_2, ...q_K\}$ de onde apenas as transições para o mesmo estado e transições esquerda-direita, para o caso de VAL dependente do texto, entre estados são permitidas, dada a característica seqüencial da voz. A abordagem define dois processos estocásticos: a seqüência de estados (modelando a *seqüência temporal* da voz) e o conjunto de processos

de saída dos estados (modelando as *características acústicas* do sinal de voz). O HMM é chamado de modelo "escondido" de Markov porque um dos processos (seqüência de estados) não é observado mas afeta a seqüência de eventos observados. É chamado de "Markov" porque a probabilidade de um estado corrente depende somente dele mesmo e do estado anterior (processo de Markov de primeira ordem) (MORGAN, 1995).

O modelamento temporal permite que o HMM faça um ajuste, no tempo, das elocuções de uma mesma palavra, já que é raro pronunciamos palavra repetidas com o mesmo tempo de duração.

No modelamento acústico é suposto que os vetores acústicos sejam descorrelacionados de seus vizinhos, o que não é verdade, porque, devido à coarticulação, a elocução de um fonema é influenciada pelos seus vizinhos mais próximos. Com essa suposição o modelamento contextual se torna muito fraco ou quase inexistente. No HMM existem duas fases importantes: *fase de Reconhecimento* - Dado uma seqüência de observações e os modelos, a fase de reconhecimento consiste em encontrarmos uma correspondente seqüência de estados que maximize as verossimilhanças das observações através dos modelos; *fase de Treinamento* - Consiste em ajustarmos os parâmetros do modelo para maximizar as verossimilhanças das observações do locutor verdadeiro que está sendo modelado.

3.2.2 – HMM em Reconhecimento de Voz

O problema básico no HMM aplicado à voz é apresentar como saída o modelo mais provável de gerar uma dada sequência de observações acústicas. É escolhido o modelo *M*

para o qual a probabilidade P(M|**X**) é máxima, onde **X** é uma seqüência de N vetores acústicos $\mathbf{X} = \{x_1, x_2, ...x_N\}$.

Como não se consegue estimar diretamente P(M|**X**) utiliza-se a regra de Bayes obtendo-se a seguinte expressão:

$$P(M \mid \mathbf{X}) = \frac{P(\mathbf{X} \mid M)P(M)}{\mathrm{P}(\mathbf{X})} \tag{3.1}$$

A probabilidade P(M|**X**) é dividida, então, em duas partes: *modelagem acústica,* na qual a probabilidade P(**X**|M)/P(**X**) é estimada, e *modelagem da linguagem*, na qual as probabilidades a priori dos modelos P(M) são estimadas. Então podemos tratar a modelagem acústica e modelagem da linguagem independentemente.

Utiliza-se o critério da máxima verossimilhança ("Maximum Likelihood Estimates - MLE") para a estimação do modelo acústico. Neste processo estima-se apenas P(**X**|M) já que P(**X**) é suposto ser igual para todos os modelos.

3.2.3 – Elementos de um HMM

Os elementos utilizados para se definir um HMM são os seguintes (RABINER, 1993):

1) N, representa o número de estados do modelo.

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

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

$$a_{ij} = \mathrm{P}(q_t = j \mid q_{t-1} = i) \qquad 1 \le i, j \le N \tag{3.2}$$

em que q_t indica o estado atual. A matriz A deve satisfazer às restrições estocásticas:

$$a_{ij} \ge 0, \qquad 1 \le i, \quad j \le N \tag{3.3}$$

e

$$\Sigma\, a_{ij} = 1, \qquad 1 \le i \le N. \tag{3.4}$$

Os valores das probabilidades de transições, que restringem o avanço e o recuo entre os estados, definem o tipo de topologia do HMM.

4) B, representa a distribuição da probabilidade de observação em cada estado, $B = \{b_j(k)\}$.

- Para o HMM discreto, seus elementos são do tipo

$$b_j(k) = p\{x = v_k / q_t = j\}, \quad 1 \le j \le N, \quad 1 \le k \le M \tag{3.5}$$

onde, $b_j(k)$ é a probabilidade da variável aleatória x (observação) pertencer ao estado j e v_k representa o k-ésimo símbolo observado no alfabeto. As restrições estocásticas são as seguintes:

$$b_j(k) \ge 0, \qquad 1 \le j \le N, \qquad 1 \le k \le M \tag{3.6}$$

e

$$\sum_{k=1}^{M} b_j(k) = 1, \quad 1 \le j \le N \tag{3.7}$$

- No HMM contínuo o conjunto de observações segmentado em cada estado é dividido em M grupos, onde cada um possui um vetor média e uma matriz covariância associados (gaussiana). A densidade de probabilidade em cada estado é, então, calculada através de uma soma das M distribuições gaussianas N, ponderada por c_{jm},, ou seja, uma mistura de gaussianas (RABINER, 1993). A probabilidade $B = \{b_{ij}\}$, então, apresenta-se da seguinte forma:

$$b_j(x) = \sum_{m=1}^{M} c_{jm} N(x, \mu_{jm}, U_{jm}) \tag{3.8}$$

onde: N = distribuição gaussianas

M = número de distribuições gaussianas

c_{jm} = coeficiente de ponderação.

μ_{jm} = vetor média.

U_{jm} = matriz covariância.

c_{jm} deve satisfazer às restrições estocásticas,

$$c_{jm} \geq 0, \qquad 1 \leq j \leq N, \qquad 1 \leq m \leq M \tag{3.9}$$

e

$$\sum_{m=1}^{M} c_{jm} = 1, \qquad 1 \leq j \leq N \tag{3.10}$$

5) Π_i, distribuição do estado inicial

$$\Pi_i = P\{ q_1 = i \}, \qquad 1 \leq i \leq N \tag{3.11}$$

A notação para representar um modelo de Markov é a seguinte:

$$\lambda = (A, B, \Pi) \tag{3.12}$$

Um HMM simples é ilustrado na Figura 3.1. Cada frase falada é associada com um particular modelo de Markov construído a partir dos Q estados de acordo com a topologia pré-definida. HMM, para reconhecimento de voz contínua, são construídos simplesmente pela concatenação de unidades elementares da fala. A unidade básica da fala que iremos trabalhar daqui para frente será o fonema.

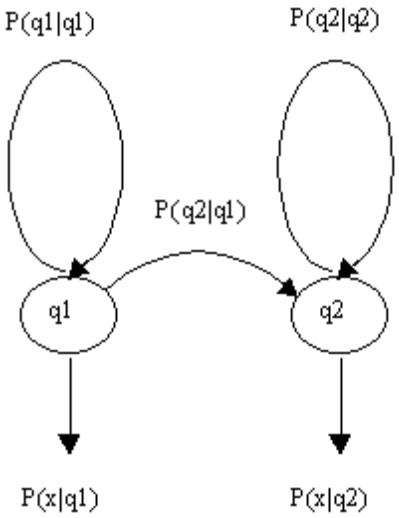


FIGURA 3.1: Esquema de um HMM de dois estados com topologia esquerda-direita.

3.2.4 – Suposições Adotadas para o HMM

Existem algumas suposições adotadas para simplificar o treinamento dos HMM's, são elas (MORGAN, 1995), (RABINER, 1993), (PICONE, 1990):

- *Suposição de Markov*: o próximo estado do HMM depende somente do estado atual, o modelo resultante torna-se, então, um HMM de primeira ordem.
- *Suposição de estacionaridade*: as probabilidades de transição de um estado para outro não se alteram durante o tempo.
- *Suposição das observações independentes*: uma dada observação corrente é estatisticamente independente das observações anteriores e posteriores, ou seja, não há correlação entre observações adjacentes. Com isso, o HMM desconsidera o efeito da coarticulação.

3.2.5 – Treinamento

O treinamento consiste em ajustarmos o modelo λ para obtermos um novo modelo $\bar{\lambda}$, usando uma seqüência de treinamento $\mathbf{X} = \{x_1, x_2, ...x_N\}$, tal que a probabilidade $P(\mathbf{X}|\lambda)$ seja máxima. Como $P(\mathbf{X}|\lambda)$ é uma função não linear dos parâmetros do modelo, apresentando muitos máximos locais, utiliza-se, então, um procedimento iterativo para encontrarmos o melhor modelo possível.

Os algoritmos utilizados em HMM são (RABINER, 1993): no treinamento do modelo, o algoritmo *Baum-Welch*, o algoritmo de *Viterbi* e o algoritmo *Segmental K-means* e no reconhecimento, o algoritmo de Viterbi.

O algoritmo de *Baum-Welch* encontra a máxima verossimilhança dos parâmetros do modelo, baseando-se no conceito estatístico da esperança do número de transições entre estados e da esperança do número de ocorrência das observações nos estados

(PARANAGUÁ, 1997). Utilizam-se, neste algoritmo, as variáveis α e β (progressiva e regressiva) para compor uma terceira variável, $\gamma_t(i)$, chamada de variável de probabilidade *a posteriori*, correspondente a probabilidade de estar no estado *i* no tempo *t* dada a seqüência de observações **X**, representada pela relação (RABINER, 1993):

$$\gamma_t(i) = P(q_t = i \mid \mathbf{X}, \lambda). \tag{3.13}$$

Então com as três variáveis e os parâmetros do modelo inicial λ a ser ajustado, os parâmetros do novo modelo discreto $\bar{\lambda}$ são dados por:

$$\bar{\pi}_i = \text{número esperado de vezes do estado } q_i \text{ no tempo } t \tag{3.22}$$

$$\bar{a}_{ij} = \frac{\text{número esperado de transições do estado } i \text{ para o estado } j}{\text{número esperado de transições do estado } i} \tag{3.23}$$

$$\bar{b}_j(x_K) = \frac{\text{número esperado de vezes que } x_K \text{ é observado em } q_j}{\text{número esperado de transições pelo estado } j} \tag{3.24}$$

Para o caso de λ ser um modelo contínuo, o novo modelo $\bar{\lambda}$ terá c_{jm}, μ_{jm}, e U_{jm} ajustados por:

$$\bar{c}_{jm} = \frac{\text{número esperado de ocorrer uma Gaussiana } m \text{ no estado } j}{\text{nº esperado de ocorrências do estado } q_j} \tag{3.25}$$

$$\bar{\mu}_{jm} = \frac{\text{número esp. de ocorrer uma gaussiana } m \text{ em } q_j \text{ ponderada por } o_t}{\text{nº esperado de ocorrer } q_j \text{ e na mistura m}} \tag{3.26}$$

$$\bar{U}_{jm} = \frac{\text{número esperado de ocorrência de uma Gaussiana } m \text{ em } q_j \text{ ponderado pela matriz covariância}}{\text{número esperado de estar no estado } q_j \text{ e na mistura } m} \tag{3.27}$$

Dados estes parâmetros, a probabilidade da observação dado o estado é calculada utilizando a equação 3.8. Lembrar que há *M* gaussianas.

Substitui-se λ por $\bar{\lambda}$, e repete-se o cálculo da reestimação até encontrar-se um ponto limite quando não houver melhoria em $P(\mathbf{X}/\bar{\lambda})$.

O algoritmo de *Baum-Welch* pode ser substituído pelo algoritmo de *Viterbi* (RABINER, 1993). Neste último somam-se as transições e as observações pertencentes a cada estado através do melhor caminho (seqüência) de estados.

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

$$\hat{a}_{ij} \rightarrow \frac{\text{número de transições do estado } q_i \text{ para } q_{jj}}{\text{número de transições do estado } q_i} \quad (3.28),$$

Os parâmetros média, covariância e coeficiente de mistura são obtidos para cada estado, após o agrupamento dos vetores de observações em *M* grupos através do algoritmo *K-means Modificado* (WILPON, 1985). A média é estimada pela média de todas as observações pertencentes a um dos *M* grupos de gaussianas de cada estado, sendo o mesmo feito para a covariância. O coeficiente de misturas será igual ao número de observações classificadas no grupo dividido pelo número total de observações classificadas naquele estado. Então, os parâmetros reestimados são dados por:

$$\bar{\mu}_{jm} = \frac{1}{N_{jm}} \sum_{i=1}^{Njm} x_i \quad (3.29)$$

$$\hat{U}_{jm} = \frac{1}{N_{jm}} \sum_{i=1}^{Njm} (x_i - \mu_{jm})(x_i - \mu_{jm})^T \quad (3.30)$$

$$c_{jm} = \frac{N_{jm}}{Nj} \quad (3.31)$$

onde x_i é a *i*-ésima observação associado ao estado *j* e gaussiana *m*, a qual possui N_{jm} observações classificadas e *Nj* é o número de observações no estado *j* e N_{jm} o número de observações na *m*-ésima mistura do estado *j*.

O algoritmo *Segmental K-means*, Figura 3.2, ameniza a sensibilidade do HMM aos valores iniciais de $b_j(x)$ sendo usado para estimar os valores dos parâmetros do modelo λ. Neste algoritmo, primeiro dividem-se as observações pelos estados de forma seqüencial, aplica-se o algoritmo de Viterbi obtendo-se o modelo $\bar{\lambda}$. Todos os parâmetros deste modelo são reestimados, logo em seguida, pelo algoritmo de Baum-Welch. A verossimilhança do modelo resultante é comparada com a do modelo anterior, obtida pelo

algoritmo de Viterbi. Se o valor de verossimilhança exceder um limiar, os parâmetros

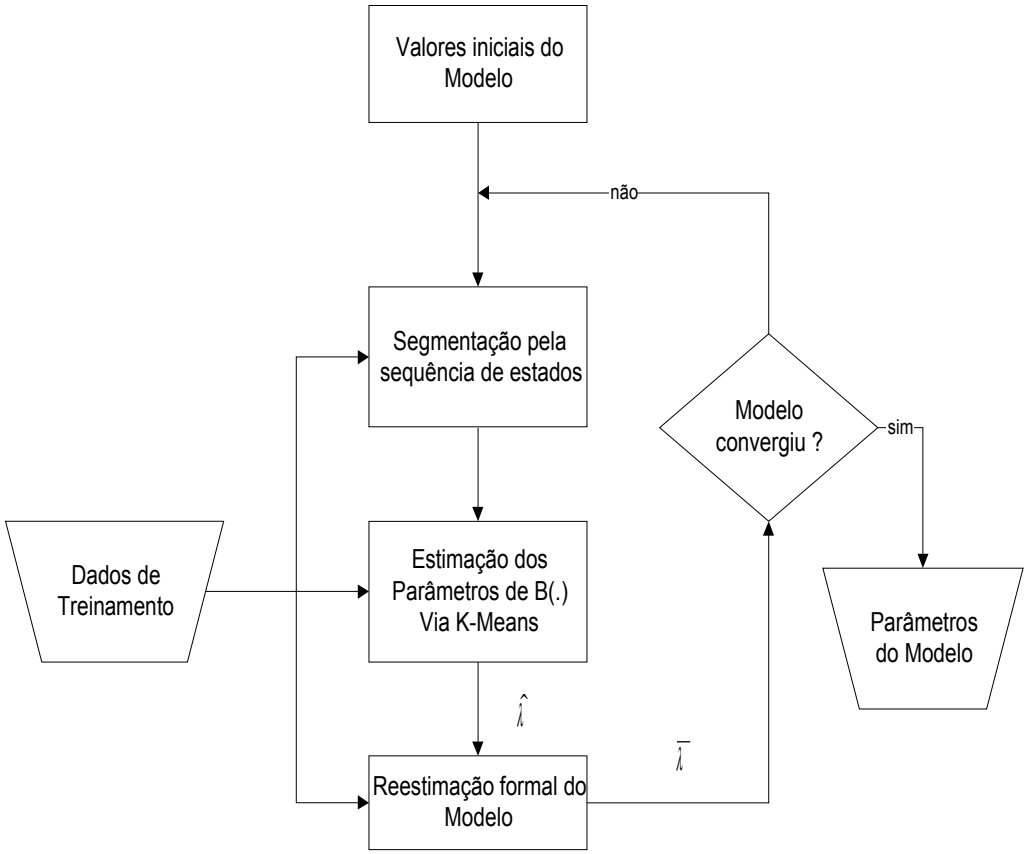


FIGURA 3.2: Algoritmo *Segmental K-means* (RABINER, 1993).

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

Uma descrição mais detalhada dos diversos algoritmos de treinamento do HMM pode ser encontrada na literatura de referência (RABINER, 1993), (PICONE, 1990), (RABINER, 1989), (RENALS, 1994).

3.2.6 – Reconhecimento

Na fase de reconhecimento, no caso da verificação, deve-se decidir se uma elocução teste pertence ou não a um determinado locutor. Calcula-se a verossimilhança da elocução $P(O|\lambda)$ ter sido gerada pelo modelo treinado, aceitando-a se for igual ou maior que um limiar. Utiliza-se o algoritmo de Viterbi para o cálculo da verossimilhança.

Na escolha do limiar devemos levar em consideração que um limiar alto dificulta a falsa aceitação, porém aumenta a falsa rejeição. Já um limiar baixo possibilita a aceitação de todos os locutores verdadeiros mas aumenta o risco da falsa aceitação.

3.2.7 – Desvantagens do HMM

Apesar dos excelentes resultados obtidos com o HMM, este possui as seguintes desvantagens:

- Discriminação fraca devido ao algoritmo de treinamento, que maximiza a verossimilhança do modelo do locutor verdadeiro e não minimiza a dos modelos dos locutores falsos, ou seja, cada modelo é treinado independentemente dos demais.
- A hipótese de Markov de primeira ordem, que diz que todas as probabilidades dependem somente do estado corrente, resultando na dificuldade dos HMM's em modelar coarticulações.
- A hipótese da independência das observações que não leva em conta a informação contextual; logo, as possíveis correlações das observações sucessivas são desprezadas.
- Os modelos de densidade de probabilidade dos HMM's (discreto, contínuo) têm um desempenho sub-ótimo, especialmente o HMM discreto, que sofre de erros de quantização. No contínuo, há um casamento pobre entre o modelo estatístico estimado e a verdadeira densidade das observações (SILVA, 1997).

3.3 – REDES NEURAIS ARTIFICIAIS (RNA)

3.3.1 – Introdução

As Redes Neurais Artificiais (RNA), ou Neurônios Artificiais, têm sido bastante usadas na resolução de problemas difíceis em reconhecimento de padrões (MORGAN, 1995), (HAYKIN, 1994). Na forma mais geral, uma rede neural é uma máquina projetada para modelar a maneira pela qual o cérebro realiza uma determinada tarefa ou função. RNA´s empregam células de funcionalidade simples, altamente interconectadas e trabalhando maciçamente em paralelo, chamadas de *neurônios* (HAYKIN, 1994). O conhecimento é adquirido por uma rede neural através do processo de "aprendizagem" que consiste no ajustes dos pesos da rede de tal maneira que esta atinja o objetivo desejado. Este processo é chamado de treinamento.

Dentre as principais características de RNA podemos citar: aproximação universal; aplicações em tempo real; tolerância a falhas; habilidade em "aprender" e adaptar-se ao seu ambiente; habilidade de generalização que se refere a produção de uma saída razoável para entradas não encontradas no treinamento; e informação contextual, onde cada neurônio na rede é afetado pela atividade global de todos os outros neurônios na rede (HAYKIN, 1994).

3.3.2 – Neurônio Artificial

Dois neurônios, com e sem polaridade, com uma simples entrada escalar são representados esquematicamente na Figura 3.3

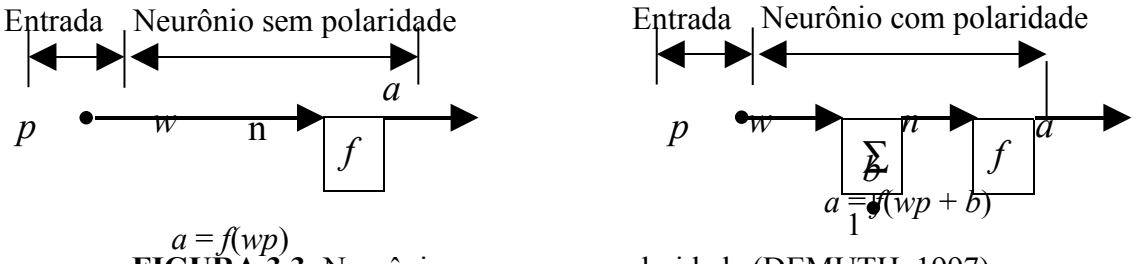


FIGURA 3.3: Neurônios com e sem polaridade (DEMUTH, 1997).

No neurônio sem polaridade a entrada escalar *p* é transmitida através de uma conexão que a multiplica pelo peso escalar *w*, formando o produto *wp*, novamente um escalar. A entrada ponderada *wp* é o argumento *n*, logo *n* = *wp*, da função de transferência *f*, que produz uma saída escalar *a*. O neurônio da direita possui polaridade, escalar *b*, que é somada ao argumento de *f*, ocasionando, então, um deslocamento da função *f* correspondente ao valor de *b*. A polaridade *b* é como um peso, exceto que a entrada conectada a esta polaridade é constante e igual a 1 (PARANAGUÁ, 1997). A entrada da função de transferência *n*, novamente um escalar, é a soma da entrada ponderada *wp* e a polaridade *b*. Esta soma é o argumento *n*, neste caso *n* = *wp* + *b*, da função de transferência *f* que produz a saída *a*. Os parâmetros ajustáveis são os escalares *w* e *b* do neurônio. Como dito na Seção 3.3.1, nós podemos treinar a rede para realizar um determinado trabalho ajustando os pesos e polaridades de acordo com valores alvos fornecidos pelo usuário, ou então a própria rede irá ajustar esses parâmetros para produzir uma saída desejada (DEMUTH, 1997).

As funções de transferências utilizadas no trabalho descrito neste compêndio foram:

- Linear:

$$f = n \tag{3.32}$$

- Logarítmica Sigmoidal:

$$f = \frac{1}{1 + e^{-(n)}} \tag{3.33}$$

- Tangente Sigmoidal:

$$f = \frac{e^{(n)} - e^{-(n)}}{e^{(n)} + e^{-(n)}} \tag{3.34}$$

- Softmax:

$$f = \frac{e^{(n_i)}}{\sum_{m=1}^{K} e^{(n_m)}} \tag{3.35}$$

onde, *i,* na função *softmax*, é a entrada corrente do neurônio e *K* é o total de saídas da camada à qual o neurônio pertence.

3.3.3 – Estrutura das Redes Neurais

Podemos citar dois tipos principais de redes neurais: *feedforward* e as *redes recorrentes*.

- Redes Feedforward: são redes onde os neurônios são organizados em forma de camadas. Os dados de entrada são captados pelos neurônios da camada de entrada produzindo uma resposta na camada de saída, podendo ainda existir camadas intermediarias chamadas de camadas escondidas. O fluxo da resposta de cada neurônio só ocorre no sentido entrada-saída. Todas as saídas dos neurônios de uma camada são conectadas com todos os neurônios da camada posterior, rede completamente conectada, sendo que não há conexões entre neurônios de uma mesma camada. A Figura 3.4 representa uma rede feedforward com quatro camadas.

- Redes Recorrentes: Estas redes diferenciam-se das feedforward pela presença de realimentações ou retardos. Com isso, um neurônio pode ser retroalimentado pela sua própria saída ou pela saída de um outro neurônio pertencente a qualquer camada. Nas redes recorrente não existe um sentido único para o fluxo das saídas dos neurônios. A Figura 3.5 apresenta uma rede recorrente.

3.3.4 – Aprendizado

Como já mencionado, o aprendizado consiste no ajuste dos pesos e das polaridades da rede de tal forma que esta possa realizar uma determinada tarefa. Existem dois tipos de aprendizado:

- *Supervisionado*: É fornecido um conjunto de funções alvos (vetores ou escalares) que é comparado com a saída da rede. A diferença entre a saída da rede e as funções alvos gera um erro que irá determinar a mudança nos pesos da rede. A mudança nos pesos é realizada de forma a minimizar esse erro.
- *Não Supervisionado*: Não são fornecidas funções alvos à rede, que trabalha somente com as entradas, organizando-se de modo a classificá-las mediante algum critério de semelhança.

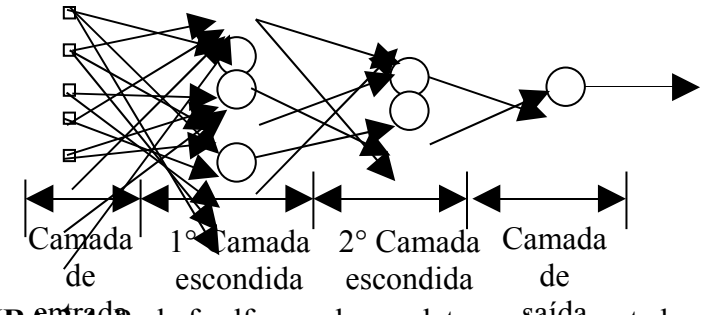


FIGURA 3.4: Rede feedforward completamente conectada com quatro camadas (HAYKIN, 1994).

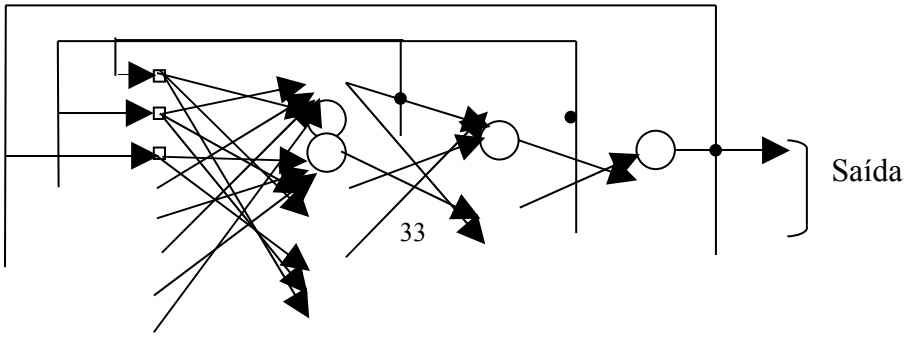


FIGURA 3.5: Rede Recorrente (HAYKIN, 1994).

3.3.5 – Dinâmica de Treinamento

O ajuste dos parâmetros da rede podem ser feitos da seguinte forma:
- "Batch": Os parâmetros são ajustados após a apresentação de todos os dados de entrada da rede (epoch). O treinamento é mais estável e menos influenciado pela ordem de apresentação dos dados de entrada, porém o aprendizado (convergência da rede) é mais lento.
- Incremental: os parâmetros são ajustados após a apresentação de cada dado de entrada. O tempo de treinamento é influenciado pela ordem de apresentação dos dados de entrada. O aprendizado nesta abordagem é mais rápido.

3.3.6 – Redes Multilayer Perceptron (MLP) ou *Backpropagation*

O algoritmo *backpropagation* foi criado pela generalização da regra de aprendizagem Widrow-Hoff para redes multicamadas e com função de transferência não linear e diferenciável em todos os pontos. O "backpropagation" padrão é um algoritmo de gradiente descendente, assim como a regra de aprendizagem Widrow-Hoff. O termo "backpropagation" refere-se à maneira pela qual o gradiente é computado em redes multicamadas não-lineares (DEMUTH, 1997). Este algoritmo ajusta os pesos dos neurônios da rede de acordo com o erro, de forma a encontrar um conjunto de pesos e polarizações que minimizem a função erro, equação 3.36. O treinamento destas redes é do tipo supervisionado.

$$E = \frac{1}{2}\sum_{x=1}^{Q}\sum_{i=1}^{S}(d_{x,i} - a_{x,i})^2 \qquad (3.36)$$

onde, Q é o número de padrões ou vetores de entrada, S o número de neurônios de saída, $d_{x,i}$ a saída desejada no i-ésimo neurônio, quando o x-ésimo padrão é apresentado e $a_{x,i}$ a saída obtida pela rede no i-ésimo neurônio, quando o x-ésimo padrão é apresentado.

Dois parâmetros são utilizados para acelerar o treinamento:

- *Taxa de aprendizado*: Influência na magnitude da variação dos pesos; quanto maior a taxa de aprendizado, maior será a variação. Se a taxa de aprendizado for muito grande, o algoritmo irá se tornar instável. Se a taxa de aprendizado for muito

pequena, o algoritmo irá levar muito tempo para convergir, podendo parar em um mínimo local.

- *Momento*: Possibilita que a rede ignore as variações da alta freqüência na superfície de erro. Sem o momento a rede pode parar em um mínimo local.

O ajuste dos pesos $W_{i,j}$ é calculado da seguinte forma:

$$\Delta W_{i,j} = - \eta \frac{\partial E}{\partial W_{i,j}} \tag{3.37}$$

onde, η é a taxa de aprendizagem e $\frac{\partial E}{\partial W_{i,j}}$ o gradiente que corresponde à derivada parcial de primeira ordem do erro em relação ao peso da respectiva conexão. Uma restrição importante na minimização do erro no sentido do gradiente descendente é que a função de transferência do neurônio tem que ser monotônica e diferenciável em qualquer ponto (HAYKIN, 1994), (RENALS, 1994).

Existem outras variantes no algoritmo básico do backpropagation que são baseadas em outras técnicas, tais como: gradiente conjugado e métodos de Newton. Ambos usam o vetor gradiente (derivada de primeira-ordem) e a matriz hessiana (derivada de segunda ordem) para o treinamento da rede (HAYKIN, 1994).

Como o algoritmo "backpropagation" básico ajusta os pesos na direção do gradiente descendente, e embora o erro, MSE, da função de desempenho decaia mais rapidamente ao longo do gradiente negativo, este não produz necessariamente uma convergência mais rápida. A solução através deste método pode seguir um caminho em "zigzag". O método do gradiente conjugado evita este problema estabelecendo uma relação complexa entre o vetor de direção e o vetor gradiente (DEMUTH, 1997), (HAYKIN, 1994). O método do gradiente conjugado é descrito na próxima seção.

3.3.7 – Método do Gradiente Conjugado

Sendo **p**(n) o vetor direção na interação n, os pesos **w** são atualizados da seguinte forma:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta(n)\mathbf{p}(n) \tag{3.38}$$

onde, $\eta(n)$ é a taxa de aprendizagem. O vetor de direção inicial **p**(0), correspondente a primeira interação, é ajustado para ser igual ao negativo do vetor gradiente inicial, **g**(0), isto é,

$$\mathbf{p}(0) = -\mathbf{g}(0) \tag{3.39}$$

Cada vetor de direção subsequente é então computado como uma combinação linear do vetor gradiente corrente e do vetor direção anterior, da seguinte forma:

$$\mathbf{p}(n+1) = -\mathbf{g}(n+1) + \beta(n)\mathbf{p}(n) \tag{3.40}$$

onde, $\beta(n)$ é um parâmetro que depende dos vetores gradientes **g**(n) e **g**(n+1). As várias versões do gradiente conjugado distinguem-se da maneira na qual $\beta(n)$ é computado. Para a atualização Fletcher-Reeves (FLETCHER, 1964), (HAGAN, 1996) o procedimento é:

$$\beta(n) = \frac{\mathbf{g}^T(n+1)\mathbf{g}(n+1)}{\mathbf{g}^T(n)\mathbf{g}(n)} \tag{3.41}$$

A computação da taxa de aprendizagem $\eta(n)$ na equação 3.38 envolve uma pesquisa linear, o propósito desta pesquisa é achar um valor particular de η para qual o erro quadrático médio (MSE), $E_{AV}[\mathbf{w}(n) + \eta\,\mathbf{p}(n)]$, seja minimizado, dados valores fixos de **w**(n) e **p**(n). Isto é, $\eta(n)$ é definido por:

$$\eta(n) = \arg\min_{\eta}\{E_{AV}[\mathbf{w}(n) + \eta\,\mathbf{p}(n)]\} \tag{3.42}$$

A precisão na pesquisa de linha exerce influência no desempenho do método do gradiente conjugado (HAYKIN, 1994).

O algoritmo “backpropagation” baseado no método do gradiente conjugado requer um menor número de “epochs” que o algoritmo “backpropagation” padrão, porém, é computacionalmente mais complexo (HAYKIN, 1994).

CAPÍTULO 4

SISTEMA HÍBRIDO

4.1 – INTRODUÇÃO

Tarefas de reconhecimento de padrões na prática são raramente implementadas por uma única estrutura, seja na forma de uma simples RNA ou de um simples HMM, ou por qualquer outro componente homogêneo (MORGAN, 1995).

Existem vários estudos do relacionamento entre redes neurais e métodos estatísticos para reconhecimento de voz, dentre eles podemos citar os realizados por Morgan e Bourlard (MORGAN, 1995). Um importante aspecto desse relacionamento é que, quando devidamente treinada, a saída de uma rede neural fornece a probabilidade *a posteriori* (BOURLARD, 1990), (BOURLARD, 1991). Como será visto na Seção 5.6.5, na estimação da probabilidade *a posteriori* usando RNA's em um sistema híbrido, estaremos usando um critério de treinamento discriminativo e não realizando uma estimação de densidade de probabilidade (RENALS, 1994), como é feito no HMM.

A natureza estatística e seqüencial do sistema de produção da voz humana dificulta o reconhecimento automático da voz. Os modelos de Markov escondidos fornecem um boa representação dessas características da voz; no entanto, seu poder discriminatório é fraco quando são treinados com o critério da máxima verossimilhança (MLE) (BOURLARD, 1990), (BOURLARD, 1991). Além do mais, a incorporação da acústica ou informação contextual fonética requer um HMM complexo com larga capacidade de armazenamento e alta quantidade de dados de treinamento (BOURLARD, 1990), ou seja: o HMM apresenta

dificuldades para fazer o modelamento contextual da voz. Realmente, como mencionado na Seção 3.2.4, a suposição de observações independentes desconsidera completamente a informação contextual da fala (coarticulação).

Por outro lado, como já mencionado na Seção 3.3, as redes neurais, dentre elas as perceptrons multicamadas (MLP), e as redes com função de bases radiais (RBF) são uma ferramenta alternativa para a classificação de padrões e, devido à sua facilidade em incorporar informação contextual, podem ser integradas com HMM's suprindo a deficiência destes no modelamento contextual da voz.

> Como as MLP's são redes "feedforward", são geralmente usadas na classificação de entradas estáticas com nenhuma característica seqüencial. Mas, se acrescentarmos retardos ou retornos, podemos inserir um certo dinamismo e uma memória nestas redes (BOURLARD, 1990). Uma vantagem no uso de MLP's com retorno é a possibilidade de, durante o treinamento, fornecer, à camada de entrada realimentada, a informação correta obtida na saída da rede associada com a entrada anterior. Apesar disso as redes neurais ainda não são bem apropriadas para o modelamento seqüencial da voz.

A probabilidade dos vetores acústicos (observações) agrupados em cada estado será chamada de *contribuição local*, que se refere à probabilidade gerada em cada estado, não confundir com a probabilidade global que mede a similaridade entre a elocução completa e o modelo treinado. Será mostrado que essa mesma contribuição local pode ser gerada por uma MLP com ou sem retorno. Iremos mostrar, teoricamente e experimentalmente, que a saída de uma MLP aproxima-se da distribuição de probabilidade sobre classes condicionadas a entrada, ou seja, a máxima probabilidade *a posteriori* (MAP), também referida como probabilidade de Bayes (BOURLARD, 1990). O uso de informação contextual na entrada da MLP, melhora o desempenho de classificação de "*frames*" em relação ao correspondente desempenho para a estimativa da máxima verossimilhança (MLE) ou até mesmo a MAP sem o uso de contexto (BOURLARD, 1990).

Este capítulo analisa a integração de RNA´s, apropriadas para o modelamento contextual, com o sistema de reconhecimento de voz baseado em modelos de Markov escondidos (HMM), apropriado para modelamento temporal.

4.2 – DEDUÇÕES REFERENTES AO HMM

Iremos analisar nesta seção o HMM discreto já que, nesse caso, as deduções para as probabilidade das observações são mais simples. Na seção seguinte provaremos que uma MLP pode estimar essa mesma probabilidade.

Em um HMM discreto os vetores acústicos são quantizados por um extrator de características. Os vetores resultantes desse processo são trocados por um vetor protótipo, y_i , selecionado de um conjunto finito predeterminado, *Y,* com *I* elementos.

Seja Q o conjunto de K diferentes estados $q(k)$ com $k = 1,2,..., K$. Como já mencionado, os algoritmos de treinamento Baum-Welch ou Viterbi trabalham com o critério da maximização da verossimilhança (MLE), ou seja, maximização de $P(X|W)$ onde X é a seqüência de treinamento dos vetores acústicos quantizados $x_n \in Y$, com $n = 1,2,...,$ N e W é o modelo de Markov associado constituído de L estados $q_l \in Q$, com $l = 1,2,..., L$. A visita do estado q_l no tempo $n \in [1,N]$ será representada por q_l^n. Como os q_l^n são mutualmente excludentes, a probabilidade $P(X|W)$ pode ser escrita para qualquer n arbitrário (BOURLARD, 1990):

$$P(X \mid W) = \sum_{l=1}^{L} P(q_l^n, X \mid W). \qquad (4.1)$$

Nesta equação $P(q_l^n, X \mid W)$ refere-se à probabilidade de *X* ser produzido por *W* quando o estado corrente é q_l no tempo *n*. Esta pode ser fatorada da seguinte forma (BOURLARD, 1990):

$$P(q_l^n, X \mid W) = P(q_l^n, X_1^n \mid W) P(X_{n+1}^N \mid q_l^n, X_1^n, W), \tag{4.2}$$

onde, X_m^n representa uma seqüência de vetores acústicos $x_m, x_{m+1}, ..., x_n$. Com algumas manipulações podemos chegar a (BOURLARD, 1990):

$$P(q_l^n, X_1^n \mid W) = \sum_{k=1}^{L} P(q_k^{n-1}, X_1^{n-1} \mid W) p(q_l^n, x_n \mid q_k^{n-1}, X_1^{n-1}, W) \tag{4.3}$$

que é a recorrência forward do algoritmo de Baum-Welch (RABINER, 1993). A probabilidade condicional $p(q_l^n, x_n \mid q_k^{n-1}, X_1^{n-1}, W)$ em (4.3) é a *contribuição local* (probabilidade de transição e observação). Como no HMM supõe-se que a observação corrente seja independente das demais observações , X_1^{n-1}, $p(q_l^n, x_n \mid q_k^{n-1}, X_1^{n-1}, W)$ reduz-se a $p(q_l^n, x_n \mid q_k^{n-1}, W)$, que representa a probabilidade de fazer uma transição do estado q_k para o estado q_l quando x_n é observado. O conjunto de todas as subunidade do modelo de Markov é caracterizado por $I \times K^2$ parâmetros

$$p[q(l), y_i \mid q^{(-)}(k), W)], \tag{4.4}$$

para i = 1,2,..., I e k = 1,2,..., K. As notações $q^{(-)}(k)$ e $q(l)$ representam estados pertencentes a *Q*, observados em dois instantes consecutivos.

O algoritmo de Viterbi pode ser considerado uma simplificação do critério MLE já que em vez de levar em consideração todos os estados possíveis em um certo *W* capazes de produzir *X*, este só considera os estados mais prováveis, ou seja, aqueles que irão gerar a maior verossimilhança. Para tornar evidente todos os caminhos possíveis (4.1) pode ser escrita como (BOURLARD, 1990):

$$P(X \mid W) = \sum_{l_1=1}^{L} \cdots \sum_{l_N=1}^{L} p(q_{l_1}^1, \cdots, q_{l_N}^N, X \mid W). \tag{4.5}$$

Considerando-se apenas os caminhos que maximize P($X|W$), (4.1) é aproximada por (BOURLARD, 1990)

$$\bar{P}(X \mid W) = \max_{l_1,\cdots,l_N} P(q_{l_1}^1,\cdots,q_{l_N}^N, X \mid W)\,. \tag{4.6}$$

Aplicando o operador "max" também em (4.3) obtemos a equação para o modelamento dinâmico no tempo ("Dynamic Time Warping" – DTW) (BOURLARD, 1990):

$$\bar{P}(q_l^n, X_1^n \mid W) = \max_k [\bar{P}(q_k^{n-1}, X_1^{n-1} \mid W)\, p(q_l^n, x_n \mid q_k^{n-1}, X_1^{n-1}, W)]\,. \tag{4.7}$$

As probabilidades globais $\bar{P}(q_l^N, X \mid W)$ para todo l, estados percorridos, são computadas usando-se (4.7) recursivamente. Sendo n_{ikl} o número de vezes que cada vetor protótipo y_i foi associado com a transição $q(k) \rightarrow q(l)$ entre dois estados pertencentes a Q durante a seqüência de treinamento X, a estimativa da probabilidade $p[q(l), y_i \mid q^{(-)}(k), W)]$ é simplesmente dada por (BOURLARD, 1990)

$$\hat{p}[q(l), y_i \mid q^{(-)}(k), W] = \frac{n_{ikl}}{\sum_{j=1}^{I}\sum_{m=1}^{K} n_{jkm}}, \quad \forall\, i \in [1, I], \forall\, k, l \in [1, K]\,, \tag{4.8}$$

a probabilidade em (4.8) aproxima-se da unidade, ou seja:

$$\sum_{i=1}^{I}\sum_{l=1}^{K} \hat{p}[q(l), y_i \mid q^{(-)}(k), W)] = 1, \quad \forall\, k \in [1, k]\,. \tag{4.9}$$

A probabilidade em (4.8) é então dividida em duas: probabilidade de transição e probabilidade de emissão (probabilidade de observação associada com uma transição) que são estimadas respectivamente por (BOURLARD, 1990):

$$\hat{p}[q(l) \mid q^{(-)}(k), W)] = \frac{\sum_{j=1}^{I} n_{jkl}}{\sum_{j=1}^{I}\sum_{m=1}^{K} n_{jkm}} \tag{4.10}$$

e

$$\hat{p}[y_i \mid q(l), q^{(-)}(k), W)] = \frac{n_{jkl}}{\sum_{j=1}^{I} n_{jkl}} \tag{4.11}$$

Supondo que a emissão de probabilidade depende somente do estado corrente $q(l)$ a equação 4.11 pode ser escrita como:

$$\hat{p}[y_i \mid q(l), W)] = \frac{\sum_{m=1}^{K} n_{iml}}{\sum_{j=1}^{I}\sum_{m=1}^{K} n_{jml}} \tag{4.12}$$

Se os modelos são treinados com o algoritmo de Viterbi nenhuma discriminação é perceptível. É importante observar que a probabilidade (4.8) não é o melhor critério para achar o estado associado mais provável, dado o estado anterior. Baseado no critério da máxima probabilidade *a posteriori* (MAP), também referido com probabilidade de Bayes, que é um classificador ótimo, o estado mais provável pode ser definido como:

$$l_{opt} = \arg\max_{l} p[q(l) \mid y_i, q^{(-)}(k)], \tag{4.13}$$

e não como na equação 4.8

$$l_{opt} = \arg\max_{l} p[q(l), y_i \mid q^{(-)}(k)] \tag{4.14}$$

Assim como no HMM, estas probabilidade são relacionadas com a contribuição local e podem ser escritas como (BOURLARD, 1990)

$$p[q(l),\mid y_i, q^{(-)}(k)] = \frac{p[y_i, q(l) \mid q^{(-)}(k)]}{p[y_i \mid q^{(-)}(k)]} \tag{4.15}$$

Somando a equação 4.8 com l variando de 1 até K produz-se uma estimativa de $p[y_i \mid q^{(-)}(k)]$ (BOURLARD, 1990):

$$p[y_i \mid q^{(-)}(k)] = \frac{\sum_{l=1}^{K} n_{ikl}}{\sum_{j=1}^{I}\sum_{m=1}^{K} n_{jkm}} \tag{4.16}$$

Substituindo (4.8) e (4.16) em (4.15) obtemos uma estimativa da probabilidade local discriminante (BOURLARD, 1990):

$$\hat{p}[q(l) \mid y_i, q^{(-)}(k)] = \frac{p[y_i, q(l) \mid q^{(-)}(k)]}{p[y_i \mid q^{(-)}(k)]} = \frac{\dfrac{n_{ikl}}{\sum_{j=1}^{I}\sum_{m=1}^{K} n_{jkm}}}{\dfrac{\sum_{l=1}^{K} n_{ikl}}{\sum_{j=1}^{I}\sum_{m=1}^{K} n_{jkm}}} = \frac{n_{ikl}}{\sum_{m1}^{K} n_{jkm}} \tag{4.17}$$

a soma desta equação é no máximo igual a 1. Será mostrado na próxima seção que valores ótimos da saída de uma MLP pode ser uma estimativa da probabilidade local discriminante, definida pela equação 4.17.

4.3 – ESTIMAÇÃO DE PROBABILIDADES COM RNA'S

Nesta seção, será estabelecida a ligação entre o HMM e as RNA's, em particular as MLP's. Veremos também que a informação contextual é facilmente inserida na entrada das RNA's. Ainda nesta seção, será descrita uma MLP, com retorno na saída, capaz de fazer um certo modelamento do aspecto seqüencial da voz.

A quantização vetorial das características acústicas, também será usada nesta seção, já que foi usada para as deduções com o HMM. Porém, podemos usar entradas contínuas para MLP's e HMM (BOURLARD, 1990).

A Figura 4.1 mostra esquematicamente uma rede de 3 camadas, sendo a camada de entrada formada de 5 vetores de características (observações). Esta figura mostra o modelamento contextual realizado por uma MLP, onde a entrada contextual é obtida pela concatenação das observações; no caso da Figura 4.1 são 4 concatenações (2 à esquerda e 2 à direita), com o vetor corrente. Sendo $2p + 1$ a largura da entrada contextual, existem $2p + 1$ janelas na entrada da rede.

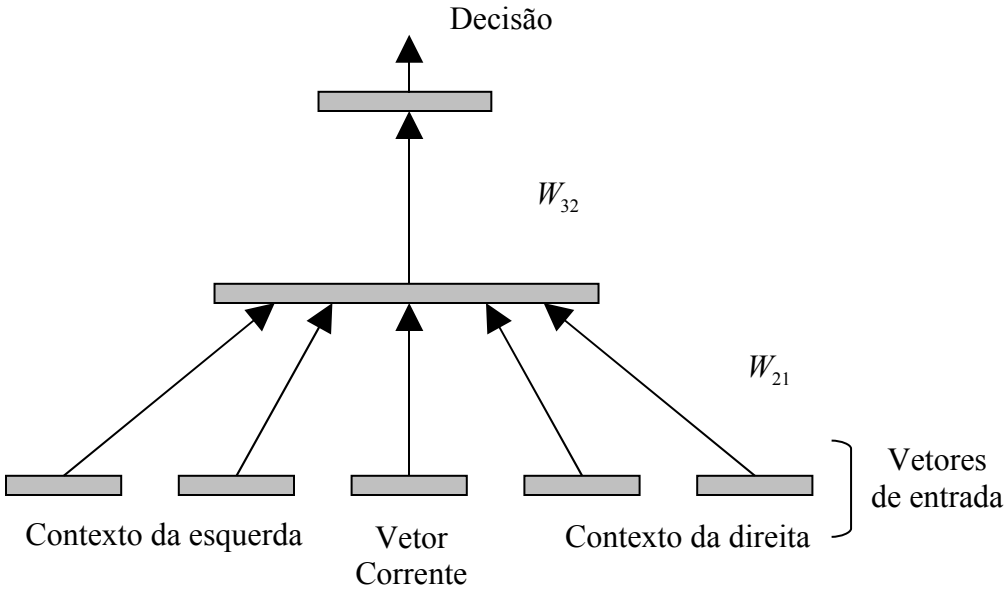


FIGURA 4.1. Rede com entrada contextual (BOURLARD, 1990).

Vamos considerar agora uma MLP sem entrada contextual ($p = 0$), consistindo o conjunto de treinamento em N entradas binárias $\{v_{i1},\cdots,v_{iN}\}$, onde cada i_n representa o índice do vetor protótipo no tempo n. A classificação seqüencial precisa lidar com decisões

anteriores, porém, o objetivo principal é a associação do vetor de entrada corrente com sua própria classe. Uma MLP que realiza essa tarefa, classificação seqüencial, irá gerar, para cada vetor corrente de entrada v_{in} e cada classe (estado do HMM) $q(l)$, l = 1,2,...,K, uma saída $g(i_n,k_n,l)$ dependendo da classe $q(k_n)$ na qual o vetor de entrada anterior v_{in-1} foi classificado. O supervisionamento é realizado pelo conhecimento *a priori* da classificação de cada v_n. O treinamento da MLP é usualmente baseado no critério do erro quadrático médio (MSE) que é igual a (BOURLARD, 1990), (HAYKIN, 1994):

$$E = \frac{1}{2}\sum_{n=1}^{N}\sum_{l=1}^{K}[g(i_n,k_n,l) - d(i_n,l)]^2, \tag{4.18}$$

onde, $d(i_n,l)$ representa os valores alvos da l-ésima saída associada com o vetor de entrada v_n. Desde que o objetivo é associar cada vetor de entrada com uma única classe, os valores alvos, para um vetor $v_i \in q(l)$ são (BOURLARD, 1990):

$$\begin{aligned} &d(i,l) = 1, \\ &d(i,m) = 0, \qquad \forall\ m \neq l, \end{aligned} \tag{4.19}$$

ou então, $d(i,m) = \delta_{ml}$. A diferença entre (4.18) e uma máquina sem memória é o termo adicional k_n que leva em consideração a decisão prévia. Juntado-se todos os termos dependentes do mesmo protótipo, a equação 4.18 pode ser reescrita como (BOURLARD, 1990):

$$E = \frac{1}{2}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{l=1}^{K}\sum_{m=1}^{K} n_{ikl}[g(i,k,m) - d(i,m)]^2, \tag{4.20}$$

onde, n_{ikl} representa o número de vezes que v_i foi classificado dentro da classe $q(l)$ quando o vetor anterior foi dado como pertencente a classe $q(k)$. O valor mínimo de E é obtido resolvendo-se (BOURLARD, 1990):

$$\frac{\partial E}{\partial w_{ij}} = \nabla_g^t E \frac{\partial g}{\partial w_{ij}} = 0, \qquad \forall\, i, j, \tag{4.21}$$

onde, t indica transposto e g denota o vetor de saída. Esta minimização é obtida pelo procedimento do gradiente (algoritmo "backpropagation"), que pode não convergir para o mínimo global. No entanto, se a rede tem um número suficiente de parâmetros e de padrões para treinamento o mínimo global pode ser obtido

$$\nabla_g E = 0 \tag{4.22}$$

ou

$$\frac{\partial E}{\partial g(i,k,m)} = \sum_{l=1}^{L} n_{ikl}[g(i,k,m) - d(i,m)] = 0. \tag{4.23}$$

Desde que E é uma função quadrática no espaço de saída, existe uma única solução e os correspondentes valores ótimos de saída são (BOURLARD, 1990):

$$g_{\text{opt}} = \frac{\sum_{l=1}^{K} n_{ikl} d(i,m)}{\sum_{l=1}^{K} n_{ikl}} = \frac{\sum_{l=1}^{K} n_{ikl} \delta_{lm}}{\sum_{l=1}^{K} n_{ikl}}, \tag{4.24}$$

e, finalmente:

$$g_{\text{opt}}(i,k,m) = \frac{n_{ikm}}{\sum_{l=1}^{K} n_{ikl}}. \tag{4.25}$$

Este mínimo global é relacionado somente com o critério do erro sendo independente da topologia da MLP (números de camadas escondidas e neurônios por camada). Comparando

as equações 4.25 e 4.17 conclui-se que os g(i,k,m)'s ótimos obtidos por meio das MLP's são realmente estimativas da probabilidade de Bayes (BOURLARD, 1990). A soma destes valores aproxima-se da unidade:

$$\sum_{m=1}^{K} g_{\text{opt}}(i,k,m) = 1 . \tag{4.26}$$

A função de transferência logarítmica sigmoidal da equação 3.33 garante que os valores ótimos em (4.25) estejam no intervalo [0,1]. Porém, quando a rede pára em um mínimo local não se pode garantir que (4.25) seja uma probabilidade. Isto pode ser resolvido substituindo-se a função sigmoidal aplicada na camada de saída pela função softmax (3.35):

$$g(i,k,l) = \frac{e^{x(i,k,l)}}{\sum_{m=1}^{K} e^{x(i,k,m)}}, \tag{4.27}$$

onde, $x(i,k,l)$ é o valor de saída do neurônio l antes da não linearidade para uma entrada y_i e a classe anterior $q(l)$.

Podemos acrescentar à entrada da MLP um vetor v_{in}^{+} que contém a saída desejada associada coma a observação $n - 1$. A informação da saída é então realimentada na entrada, fazendo com que esta MLP apresente uma topologia recorrente.

Para a classificação das observações, usaremos uma saída para cada classe, sendo que todos os alvos são zero exceto o alvo correspondente à classe correta, codificação "one-from-k". Então, caso existam parâmetros suficientes no sistema e se o treinamento não pára em um mínimo local, a saída de uma MLP irá aproximar a probabilidade *a posteriori* (probabilidade de Bayes), ou seja, a distribuição de probabilidade sobre classes de saída

condicionada à entrada. Esta conclusão também pode ser generalizada para entradas contínuas (BOURLARD, 1990).

Agora que está provado que uma MLP pode estimar a probabilidade local e desde que as observações de treinamento não são rotuladas (nenhuma segmentação explícita) o treinamento de uma MLP pode ser inserido no algoritmo de Viterbi usando-se os valores de saída $g(i,k,l)$ como probabilidade local, melhorando a segmentação inicial.

Os resultado apresentados até aqui são validos também para uma MLP com entrada contextual, Figura 4.2. É inserido nesta rede um retorno de saída, entrada extra, associada à saída da observação anterior. Desde que os dados de treinamento consistem de observações acústicas consecutivas, a seqüência correta dos estados de saída é conhecida, logo, é possível o treinamento com valores corretos de retorno. Este mesmo tipo de MLP recorrente pode ser usado para estimar probabilidade local de um modelo de Markov de alta ordem, onde as contribuições locais são não só dependentes do estado anterior como de vários outros estados anteriores. Basta para isso estender o retorno de saída para quantas classificações anteriores se desejar inserir na entrada. A saída da rede, então, será dependente do vetor de observação corrente, do contexto e dos valores de saída passados.

4.3.1 – Divisão pela Probabilidade de Classes *a Priori*

Os resultados obtidos na seção anterior dizem respeito à classificação de "frames" e não há garantia de obtermos um bom desempenho para reconhecimento de palavras em voz contínua não segmentadas**.** Por isso, antes de usarmos a saída de uma RNA como

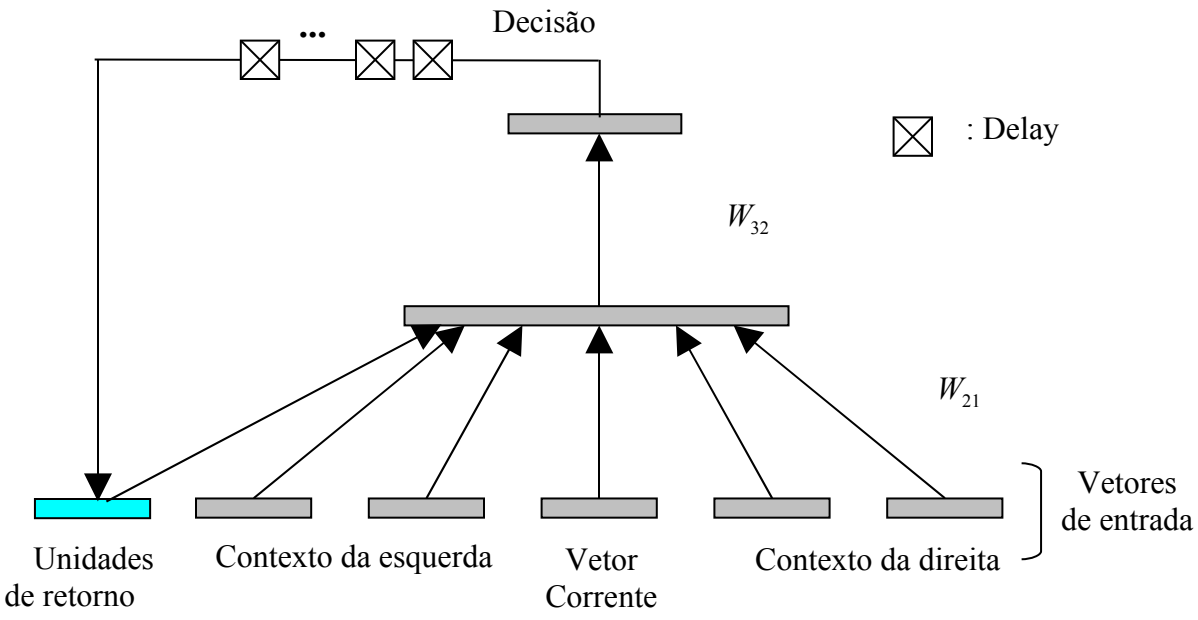


FIGURA 4.2: MLP com entrada contextual e retornos de saída (BOURLARD, 1990).

probabilidade para HMM's, é necessária dividi-la pela respectiva probabilidade de classe *a priori* $p(q_k)$ para obtermos verossimilhança. Esta probabilidade *a priori* são estimadas contando-se o número de vezes que cada classe $q_c, k = 1,2,...,K$, aparece no conjunto de treinamento (BOURLARD, 1993), a divisão é realizada da seguinte forma:

$$\frac{P(q_k \mid x_n, \theta)}{P(q_v\}} = \frac{p(x_n \mid q_k, \theta)}{p(x_n \mid \theta)} \tag{4.28}$$

onde θ representa os parâmetros da RNA. Essa divisão prejudica o desempenho quanto ao "frame", porém, melhora o desempenho quanto ao reconhecimento de palavra

(BOURLARD, 1991). Este procedimento serve para remover o efeito da probabilidade *a priori* no reconhecimento.

A necessidade desta divisão pode ser explicada pelo descasamento entre a freqüência relativa dos fonemas no conjunto de treinamento e no conjunto de teste.

4.3.2 – Vantagens do Uso de Redes Neurais para Estimação de Probabilidades

As principais vantagens em usarmos redes neurais para estimação de probabilidades são as seguintes (BOURLARD, 1997):

- Obtenção de um modelo mais preciso: a estimação de probabilidades por RNA's não requer suposições detalhadas sobre a forma da distribuição estatística a ser modelada, resultando em um modelo acústico mais preciso.
- Utilização de contexto: como discutido na Seção 4.1.
- Discriminação: pode-se facilmente implementar um treinamento discriminativo utilizando redes neurais.
- Uso racional dos parâmetros pois é mais econômico modelar contornos entre classes acústicas (*a posteriori*) do que superfícies de funções de densidade (verossiminhanças).
- Flexibilidade: O uso de redes neurais como estimadoras de probabilidades acústicas permite a combinação de diversas características.

CAPÍTULO 5

SISTEMAS IMPLEMENTADOS

5.1 – INTRODUÇÃO

Este capítulo descreve os procedimento utilizados desde a aquisição do banco de dados de voz até os testes realizados com as MLP's, HMM's e os sistemas híbridos para verificação automática do locutor

Todos os sistemas foram desenvolvidos em um computador Pentium MMX com frequência de operação de 233 Mhz.

5.2 – BASE DE DADOS

As elocuções utilizadas neste compêndio foram gravadas em uma placa "*Sound Blaster 16*", da Creative Labs, com taxa de amostragem de 11025 Hz e quantização de 16 bits.

Para treinamento e teste dos sistemas foram utilizados dois conjuntos de locutores: masculino (LM) e feminino (LF). O primeiro com 16 locutores para treinamento e teste e 10 somente para teste. O segundo com 8 locutores femininos para treinamento e teste e 7 somente para teste, totalizando 41 locutores, sendo que, deste total 15 gravações foram as mesmas usadas em (PARANAGUÁ, 1997).

As frases utilizadas foram "O prazo tá terminando" (E1) e "Amanhã ligo de novo" (E2) consideradas em (BEZERRA, 1994) as mais adequadas para reconhecimento de locutores para fins forenses.

Para cada locutor foram realizadas 60 gravações de cada frase: 30 para treinamento (MLP, HMM e Híbrido), 10 para validação cruzada (MLP e Híbrido 1) e 20 para teste. Foram feitas gravações a mais para 2 locutores, um masculino e um feminino, onde variou-se o tipo de microfone usado e o intervalo de tempo entre as gravações, buscando-se com isso avaliar o desempenho dos sistemas quanto a estes dois tipos de

variações. O total de gravações para treinamento dos 24 locutores (16 masculinos e 8 femininos), para cada frase, foi de 1440, e de gravações para teste dos outros 17 locutores, para cada frase, foi de 340 . O total geral para cada frase foi de 1780 elocuções, resultando, para as duas frases, um total de 3560 elocuções. A Tabela 5.1 mostra a relação de locutores com algumas informações importantes sobre as gravações, onde, *COD* é o código do locutor, *ID* a sua idade, *A* sua altura, *P* seu peso, *DATA* a data de gravação das elocuções e *UF* o estado de nascimento do locutor. As gravações números 12 à 16, 22 à 24, 31 à 34 e 37 à 39 foram realizadas em (PARANAGUÁ, 1997). As elocuções de locutores, masculinos e femininos, com número de identificação a partir de 100 foram usadas apenas para teste. Na gravação número 17 foi utilizado um microfone diferente do utilizado nas gravações número 1 e 25 para um mesmo locutor masculino, o mesmo foi feito com a gravação número 27 em relação as gravações números 40 e 41 para um locutor feminino. Com relação ao período de gravação as gravações 25 e 17 distam da gravação 1 por 7 meses e as gravações 40 e 41 distam da gravação 27 por 5 meses.

Utilizou-se somente a frase 2 ,"Amanhã ligo de novo", nos teste porque a mesma possui um poder de discriminação maior do que a frase 1, "O prazo tá terminando", já que apresenta um maior número de fonemas nasalados o que a torna mais robusta contra à mímica (SANTOS, 1989).

As elocuções foram salvas com o seguinte formato: para locutores masculinos R*X*E*Y*LM*Z*.WAV e para locutores femininos R*X*E*Y*LF*Z*.WAV, onde *X* é o número da repetição, *Y* o número da frase e *Z* o número do locutor.

TABELA 5.1: Relação dos locutores utilizados no compêndio.

(a): Locutores masculinos.

#	COD	LOCUTOR MASCULINO	*ID*	*A*	*P*	*UF*	*DATA*
1	1	Marcos Paulo B. Oliveira	27	1,88	98	MA	08/09/00
2	2	Charles Borges de Lima	23	1,76	61	RS	10/08/00
3	3	Michel Sousa Medeiros	23	1,77	65	PE	26/10/00
4	4	Adenilson R. S. Pontes					11/08/00
5	5	Fábio Miranda	36	1,78	100	RJ	14/08/00
6	6	Fausto Jr. M. Ferreira					15/08/00
7	7	Bruno de Pinho Silveira	21	1,86	83	MG	26/10/00
8	8	Álvaro de Jesus Netto					22/08/00
9	9	Guilherme M. Ottoni Silva	21	1,74	68	RJ	22/08/00
10	10	Vitor Cezar					24/10/00
11	11	Robson França de Moraes	22	1,85	86	RJ	26/10/00
12	12	LM1					07-09/96
13	13	LM2					07-09/96
14	14	LM3					08-09/96
15	15	LM4					07-09/96
16	16	LM5					08-09/96
17	100	Locutor 1	28	1,88	98	MA	21/04/01
18	101	Alexandre F. Nascimento	35	1,75	65	RJ	10/08/00
19	102	André Renato da S. Aguiar	19	1,70	71	RJ	06/12/00

20	103	Marco A. Rocca Andrade	30	1,75	78	RJ	14/12/00
21	104	Mário Jorge da S. Motta	26	1,68	58	MA	26/12/00
22	105	LM6					07-09/96
23	106	LM7					08-10/96
24	107	LM8					06-09/96
25	108	Locutor 1	28	1,88	98	MA	21/04/01
26	109	**Gilberto Gil**	33	1,67	67	MA	29/05/01

(b): Locutores femininos

#	COD	LOCUTOR FEMININO	*ID*	*A*	*P*	*UF*	*DATA*
27	1	Vanessa C. C. Rodrigues	19	1,66	57	MA	02/11/00
28	2	Joyce Sobrinho F. Dias	20	1,57	57	RJ	06/12/00
29	3	**Viviane Barros Oliveira**	24	1,69	61	MA	26/12/00
30	4	Adriana de F. Lima Maciel	24	1,67	62	MA	29/12/00
31	5	LF1					08-09/96
32	6	LF2					07-09/96
33	7	LF3					07-09/96
34	8	**LF4**					07-08/96
35	100	Daniela Barros Oliveira	26	1,70	77	MA	28/12/00
36	101	Elis Cristina Amaral	25	1,77	70	MA	27/12/00
37	102	LF5					07-08/96
38	103	LF6					07-09/96
39	104	LF7					07-09/96
40	105	Locutora 1	20	1,66	57	MA	21/04/01
41	106	Locutora 1	20	1,66	57	MA	21/04/01

5.3 – PRÉ-PROCESSAMENTO

Nesta fase extraem-se as características do sinal de voz que serão utilizadas nos sistemas de reconhecimento. O janelamento e a superposição no HMM e no sistema híbrido diferem da MLP's tendo em vista que nesta última o número de janelas tem que ser igual para todas as elocuções.

5.3.1 – Pontos Extremos ("Endpoints")

Nesta fase foram utilizados dois algoritmos para o cálculo dos "endpoints": o primeiro é o algoritmo de Rabiner e Sambur, (RABINER, 1975), (método B) com alterações introduzidas por (DINIZ, 1997) que melhoraram a taxa de acerto dos pontos terminais; o segundo é o algoritmo desenvolvido por Rocca (ANDRADE, 1999), (método C) mencionado na Seção 2.2.2. Para avaliarmos a eficiência dos dois algoritmos, compararam-se estes com os "endpoints" calculados manualmente (método A). A Tabela 5.2 mostras os resultados com 7 elocuções de cada frase para o locutor masculino 1. Considerou-se como acerto, para os métodos B e C, uma diferença, nos pontos terminais, de no máximo 3, a mais ou a menos, em relação aos pontos terminas obtidos com o método A. Esta tabela mostra que houve 4 erros no cálculo do ponto inicial para o método B e no método C houve 1 para o ponto inicial e 6 para o ponto final. Apesar do método B ter errado menos que o método C os erros neste último foram menores em número de janelas do que o primeiro, de qualquer forma, uma edição manual nos métodos B e C seria necessária, como temos um total de 3560 elocuções estas edições seriam muitos trabalhosas. Visando atenuar este problema, criou-se uma interface gráfica com o método B em Matlab que permite editar de forma mais rápida e agradável os erros no cálculo dos "endpoints".

TABELA 5.2: Resultados no cálculo dos "endpoints".

ELOCUÇÃO	*MÉTODO*					
	Método A		**Método B**		**Método C**	
	PI	**PF**	**PI**	**PF**	**PI**	**PF**
R1E1LM1	85	256	85	256	83	261
R10E1LM1	92	260	91	260	90	254
R20E1LM1	106	277	80	277	104	271

R30E1LM1	117	268	65	268	115	270
R40E1LM1	101	261	75	261	98	254
R50E1LM1	100	274	100	277	98	273
R60E1LM1	84	250	82	252	83	291
R1E2LM1	89	240	92	240	86	237
R10E2LM1	107	266	4	267	105	263
R20E2LM1	105	270	103	271	104	268
R30E2LM1	91	232	89	232	89	231
R40E2LM1	93	250	91	251	91	247
R50E2LM1	86	255	86	255	84	253
R60E2LM1	89	243	89	241	82	239

5.3.2 – Janelamento e Superposição

Foram utilizados nesta fase janelas de 220 amostras, o que a 11025 Hertz de taxa de amostragem resulta, aproximadamente, em janelas de 20 milisegundos. Para o HMM e o sistema híbrido aplicou-se uma superposição de 50% com número de janelas livre. Já as MLP's do tipo *"feedfoward"* requerem um número de janelas fixo, independente do tamanho da elocução. Fixou-se, então, o número de janelas em 170 variando-se a superposição entre janelas da seguinte forma (BEZERRA, 1994):

$$n = N / T$$

$$t_{\text{sup}} = T \cdot \frac{(k - n)}{(k - 1)}, \qquad (5.1)$$

onde, N é o número de amostras do sinal, T é duração (em amostras) da janela, k é o número de janelas com superposição (como dito anteriormente k foi fixado em 170 janelas), n é o número de janelas sem superposição e t_{sup} é a superposição (em amostras) entre janelas. Como T e k são fixos a superposição vai depender do número de amostras do sinal N (tamanho da elocução). A Tabela 5.3 mostra o intervalo de duração da menor e da maior elocução com o respectivo valor da superposição.

TABELA 5.3: Intervalo de duração das elocuções com a respectiva superposição com número fixo de janelas igual a 170.

FRASE	MENOR DURAÇÃO (s)	SUPERPOSIÇÃO	MAIOR DURAÇÃO (s)	SUPERPOSIÇÃO
2	0.71	78.41	2.66	20.33

5.3.3 – Considerações sobre a Extração das Características do Sinal de Voz

Foram utilizados 12 coeficientes mel cepestrais derivados do LPC, de ordem 14, (MCLPC) mais o log energia de tempo curto, em seguida calculou-se o delta de cada uma dessa características (DMCLPC) resultando em um total de 26 características. A Figura 5.1 mostra a disposição das características após a

extração dos atributos, onde, os coeficiente mel cepestrais são representados por MC*X,* sendo *X* o número do coeficiente, os coeficientes delta mel cepestrais por DMC*X*, LogENER representa o log energia e DLogENER o delta log energia.

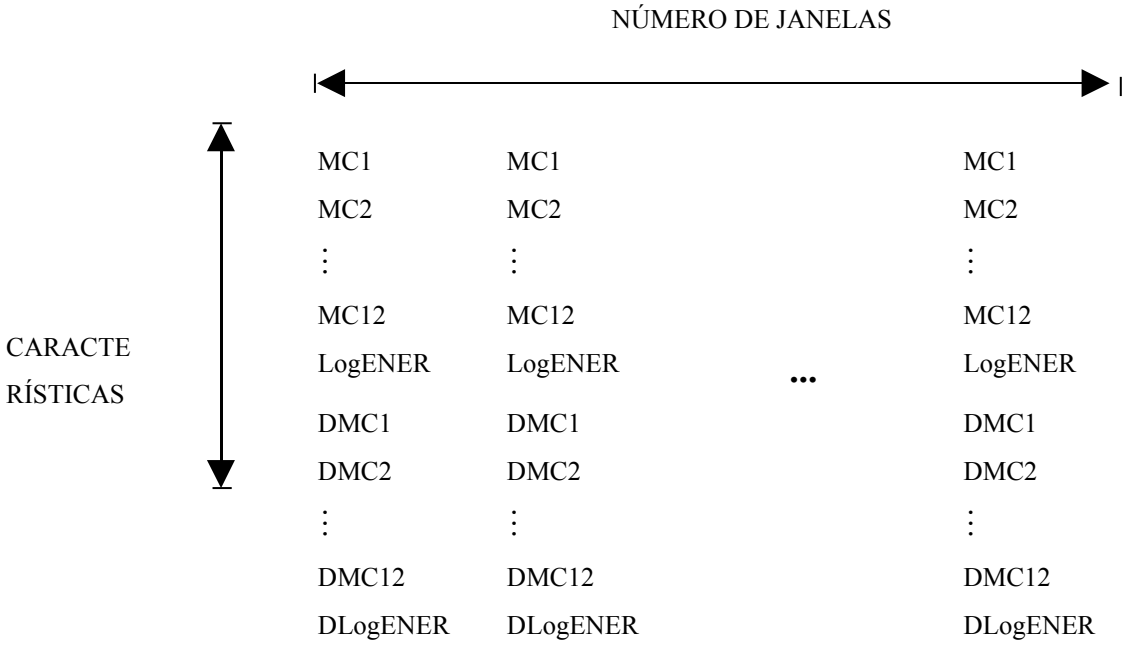


FIGURA 5.1: Disposição das características após a extração dos atributos.

5.3.4 – Considerações sobre a Seleção das Características mais Relevantes

Aplicou-se a equação (2.23) em 5 repetições dos 24 locutores de treinamento (16 masculinos e 8 femininos). A Tabela 5.4 mostra os resultados para as 26 "features" em ordem decrescente.

Foram seleciondas, para um teste inicial, as 5 features de maior razão *F* e seus respectivos deltas. Apesar dos deltas terem uma razão *F* menor, esses são úteis para atenuar a influência do canal de gravação (microfone, por exemplo) no sinal de voz (SILVA, 1997). Com isso, as características selecionadas, para um primeiro teste, foram as seguintes: MC3, MC4, MC8, MC2, MC5, DMC3, DMC4, DMC8, DMC2 e DMC5.

Com estas características as MLP tiveram dificuldade para convergir. Uma possível explicação é que embora os deltas sejam úteis para atenuar o efeito do canal, estes não têm um bom poder discriminatório entre locutores. Em virtude disso, utilizaram-se todas as "features" mencionadas na Seção 5.3.3, visando maior discriminação entre locutores.

TABELA 5.4: Razão *F* em ordem decrescente para as 26 “features”.

#	Feature	Razão F
1	MC3	1,7938
2	MC4	1,7716
3	MC8	1,6588
4	MC2	1,6545
5	MC5	1,6456
6	MC1	1,6309
7	MC10	1,6263
8	MC6	1,607
9	MC12	1,5539
10	MC7	1,515
11	MC11	1,4793
12	DMC12	1,4531
13	MC9	1,3693
14	DMC6	1,0044
15	DMC2	0,99055
16	LogENER	0,95211
17	DMC7	0,94915
18	DlogENER	0,93543
19	DMC1	0,90578
20	DMC10	0,90413
21	DMC3	0,89489
22	DMC4	0,88119
23	DMC5	0,86937
24	DMC9	0,85754
25	DMC11	0,82929
26	DMC8	0,77568

5.4 – CONSIDERAÇÕES SOBRE OS HMM'S UTILIZADOS

Descreveremos nesta seção os dois tipos de técnicas, adaptadas para reconhecimento do locutor, utilizadas para o modelamento das frases com HMM: uma foi a utilização de HMM's para reconhecimento de palavras isoladas e a outra foi a utilização de HMM's para reconhecimento de voz contínua (RVC).

5.4.1 – Utilização de um Único HMM para Modelar as Frases

Foi utilizado um único HMM para modelar a frase inteira, similar ao que é feito para reconhecimento de palavras isoladas, só que, nesse último, modelam-se palavras e não frases.

O HMM utilizado foi implementado por Dirceu (SILVA, 1997) em seu projeto de fim de curso, sendo um modelo com estrutura esquerda-direita e procedimento de treinamento "Segmental-Kmeans". Utilizou-se um modelo com 12 estados e 5 gaussianas, pois esta foi a configuração que obteve melhor resultado para RAL nos testes realizados em (PARANAGUÁ, 1997).

Maiores detalhes sobre a implementação do HMM podem ser encontradas em (RABINER, 1993), (PICONE, 1990), (RABINER, 1989).

Após o reconhecimento, as verossimilhanças foram divididas pelo número de janelas da respectiva elocução (PARANAGUÁ, 1997), este processo será descrito na Seção 5.6.4. Escolheu-se um limiar para cada locutor utilizando as elocuções de treinamento, com o seguinte critério: eleva-se o limiar até eliminarmos todas as falsas aceitações.

5.4.2 – Utilização de HMM's Concatenados para Modelar as Frases

Utilizou-se a mesma técnica empregada para reconhecimento de voz contínua desenvolvida em (SANTOS & ALCAIM, 2001), onde, os HMM's são concatenados para modelar as frases. As unidades fonética utilizadas foram as sílabas pois o português é uma língua silábica por natureza onde a sílaba é o núcleo com que se formam as palavras (SANTOS & ALCAIM, 2001).

Para um teste inicial foram segmentadas, manualmente, as frases em sílabas para o locutor masculino 1, resultando num total de 8 sílabas. Treinou-se um HMM para cada sílaba, onde cada modelo possui 3 estados e 5 gaussianas. Após o treinamento, foram concatenados os 8 HMM's, correspondentes às 8 sílabas, através de suas matrizes de transição de estados (SANTOS, 1997). A Figura 5.2 ilustra a estrutura dos modelos utilizados para gerar a frase 2 com essa técnica. Foi realizado, então, a verificação do locutor com o modelo resultante desta concatenação. Os resultados não foram melhores que os obtidos com o HMM para

reconhecimento de palavras isoladas, descrito na seção anterior.

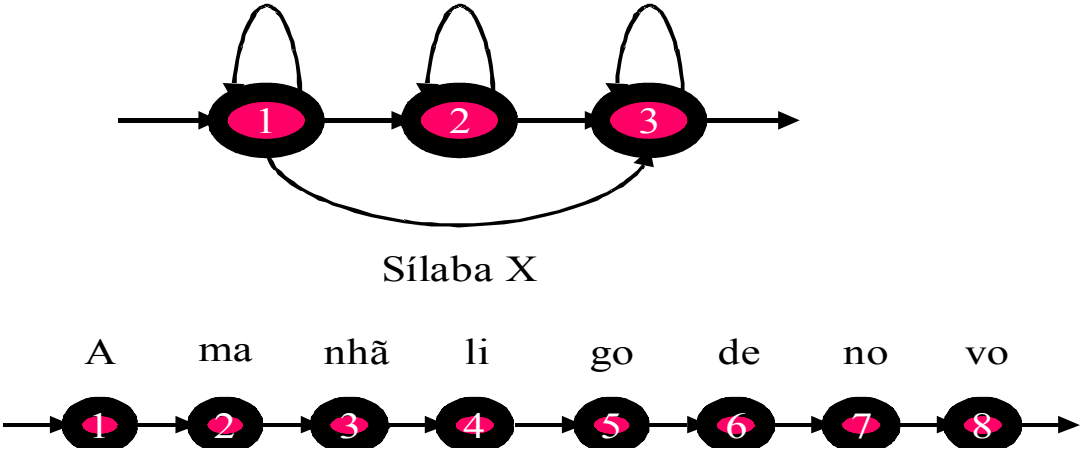


FIGURA 5.2: Estrutura dos modelos utilizadas para gerar a frase 2 (SANTOS, 1997)

Uma possível explicação para isso, é que, esse procedimento visa a formação dos vocabulários para um reconhecedor de voz que irá fornecer os maiores valores de verossimilhanças para esses vocabulários. Fato esse, que não ocorre para reconhecimento do locutor, dependente do texto, onde apenas uma frase é utilizada. Logo, escolheu-se o HMM descrito na Seção 5.4.1 visto que esse obteve resultados tão bons quantos os obtidos com o HMM descrito nessa seção, além do que, no primeiro, não há necessidade de nenhuma segmentação *a priori*.

5.5 – CONSIDERAÇÕES SOBRE AS MLP'S

Foram realizados testes com MLP's, para verificação do locutor, com o algoritmo "backpropagation" e método do gradiente conjugado, pois este obteve bons resultados quanto ao tempo de treinamento e ao desempenho. Todas as redes foram criadas utilizando o "toolbox" de redes neurais do MatLab ("The MathWorks, Inc"), versão 5.3.

5.5.1 – Considerações sobre Preparação dos Dados para Treinamento, Teste e Validação.

A matriz de treinamento das MLP's, para cada frase, é formada da seguinte maneira: cada coluna representa uma elocução. Para o locutor verdadeiro, usaram-se todas as 60 repetições e para os demais locutores, apenas 5 repetições. Buscou-se, com isso, modelar a classe que representa o mundo exterior dos locutores falsos. A matriz de dados resultante é mostrada na Figura 5.3 (elocução, nesta figura, representa os vetores de "features"). As elocuções de treinamento, teste e validação, para o locutor verdadeiro, foram separadas da seguinte forma:

- *Elocuções de treinamento*: todas as elocuções ímpares, totalizando 30 repetições.
- *Elocuções de teste*: foram selecionadas as seguintes repetições (20 no total):

2, 6, 8, 12, 14, 18, 20, 24, 26, 30, ... , 50, 54, 56, 60;

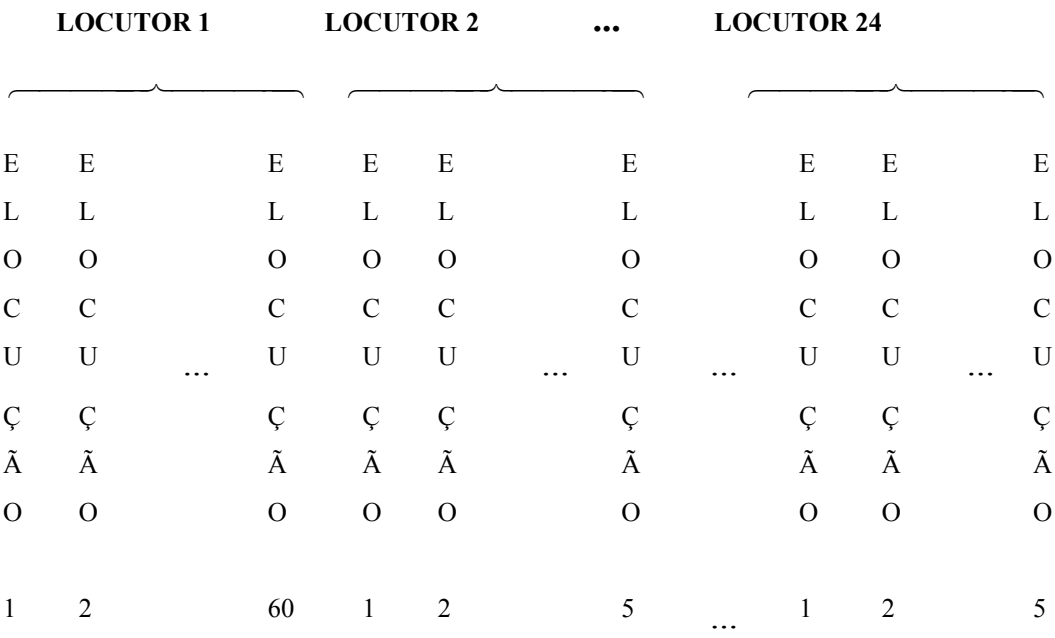


FIGURA 5.3: Matriz de treinamento para as MLP's.

- *Elocuções para validação cruzada:* foram selecionadas as seguintes elocuções (10 no total):

4, 10, 16, 22, 28, 34, 40, 46, 52, 58;

As características foram normalizadas entre –1 e 1, o que evita que o neurônio trabalhe na região de saturação quando este usa, por exemplo, a função de transferência logarítmica sigmoidal, *logsig*. Foram selecionadas, então, apenas as elocuções de treinamento para esta normalização, aplicou-se a função *premnmx,* do Matlab ("The MathWorks, Inc"), para obter os valores máximos e mínimos das elocuções de treinamento de cada locutor, e *tramnmx* para normalizar os dados.

Como a rede só possui um neurônio na camada de saída e sua função de transferência é a tangente sigmoidal, *tansig,* os alvos usados foram –1 e 1. Atribui-se, então, 1 para os alvos das elocuções verdadeiras (locuções que pertencem ao locutor a ser treinado) e -1 para as demais elocuções (elocuções falsas).

5.5.2 – Topologia das MLP's para Verificação Automática do Locutor

A MLP para verificação automática do locutor é formada por quatro camadas (1 de entrada, 2 escondidas e 1 de saída). As funções de transferência da primeira camada escondida foi *logsig*, da segunda camada escondida foi *purelin* e da camada de saída foi a *tansig*. O número de neurônios em cada uma das quatro camadas é o seguinte: 4420 (camada de entrada), 70 (primeira camada escondida), 20 (segunda camada escondida) e 1 (camada de saída), o número de neurônios na primeira camada (4420) se justifica pelo fato de termos 26 características e por termos um número fixo de 170 janelas, resultando, então, em um total de 4420 (26*170) características de entrada, o que representa uma elocução.

Escolheu-se esta estrutura, com uma camada *purelin* entre a *logsig* e a *tansig*, porque consegue-se "aprender" relações lineares e não lineares, entre entrada e saída, utilizando MLP´s com camadas não lineares (*logsig)* e lineares (*purelin*) (HAYKIN, 1994),(DEMUTH, 1997). É importante ressaltar que, para esta aplicação, MLP´s com apenas uma camada escondida poderiam ser suficientes para obtermos um desempenho satisfatório.

5.5.3 – Inicialização dos Pesos e Polaridades das MLP's

Antes de iniciarmos o treinamento da rede fez-se uma inicialização dos pesos e polaridades da rede. Na camada com função de transferência linear (*purelin*) os pesos e polaridades foram inicializados com valores aleatórios entre –1 e 1. Nas camadas com função de transferência sigmoidal (*logsig*) e tangente sigmoidal (*tansig*) a inicialização é baseado na técnica de Nguyen e Widrow, implementada em (DEMUTH, 1997), que gera valores de pesos e polaridades para a camada tal que a região de ativação dos neurônios desta camada tenham uma distribuição aproximadamente uniforme sobre o espaço de entrada. Este procedimento tem vantagem de proporcionar um treinamento mais rápido em relação a inicialização puramente aleatória dos pesos e polaridades (DEMUTH, 1997).

5.5.4 – Treinamento das MLP's para Verificação Automática do Locutor

Utilizou-se o algoritmo do gradiente conjugado "TRAINCGB" pois este obteve os melhores resultados, principalmente quanto ao tempo de treinamento. Este algoritmo reinicia o vetor direção, $\mathbf{p}(n) = -\mathbf{g}(n)$, quando existe pouca ortogonalidade entre o gradiente corrente e o gradiente anterior, isto é testado com a seguinte inequação:

$$|\mathbf{g}^T(n-1)\mathbf{g}(n)| \geq 0.2\|\mathbf{g}(n)\|^2 \qquad (5.2)$$

O parâmetro $\beta(n)$ da função "TRAINGCB" é o mesmo do procedimento Polak-Ribiére (FLETCHER, 1964), (HAGAN, 1996), ou seja:

$$\beta(n) = \frac{\mathbf{g}^T(n)[\mathbf{g}(n) - \mathbf{g}(n-1)]}{\mathbf{g}^T(n-1)\mathbf{g}(n-1)} \qquad (5.3)$$

A função de pesquisa de linha usada foi a "SRCHCHA", para detalhes sobre esta função consulte (DEMUTH, 1997). A rotina para se criar a rede com a topologia e os parâmetros mencionados acima é a seguinte:

```
net_mlp = newff(minmax(mat_treina),[70 20 1],...
      {'logsig' 'purelin' 'tansig'},'traincgb');
```

onde, *mat_treina* é a matriz de dados de treinamento, Figura 5.3, e *net_mlp* é a rede recém criada.

Estabeleceu-se um erro (MSE) de 10^{-5} como o objetivo a ser atingido pela função desempenho. O "overfitting" foi evitado utilizando-se o conjunto de validação cruzada no treinamento (o "overfitting" ocorre quando a rede apresenta bons resultados somente para os dados de treinamento), quando o erro do conjunto de validação cruzada aumenta para um número específico de iterações, o treinamento é parado, e os pesos e polaridades que proporcionaram o menor erro para o conjunto de validação são retornados. O desempenho em relação aos dados de testes também são monitorados na fase de treinamento. A rotina para iniciar o treinamento da rede é a seguinte:

```
[net_mlp,Tr] = train(net_mlp,mat_treina,dt,[],[],v,t);
```

onde, *dt* são os alvos da rede, *v* uma estrutura contendo os dados de validação cruzada e seus alvos e *t* também uma estrutura contendo os dados de teste com seus alvos, *Tr* uma outra estrutura onde é armazenado o MSE dos dados de treinamento, validação e teste e *net_mlp* a rede.

A Tabela 5.5 mostra alguns dados sobre o treinamento das MLP's para verificação do locutor utilizando-se a frase 2, onde *Epochs* é o numero de "epochs", *Tempo* é tempo total de treinamento para todas as reinicializações da rede, *MSE Trein* o erro quadrático médio dos dados de treinamento, *MSE* Valid o erro quadrático médio dos dados de validação cruzada e *MSE Teste* o erro quadrático médio dos dados de teste.

Podemos concluir que as MLP´s atingiram o erro desejado em poucas "epochs", como mencionado na Seção 3.3.7. Em relação ao tempo de treinamento, as redes tiveram uma variação muito grande, e uma possível razão para isso é que algumas redes param em mínimos locais na superfície de erro, precisando serem reinicializadas, o que aumenta o tempo de treinamento.

TABELA 5.5: Dados sobre treinamento das MLP's para verificação do locutor.

LOC	Epochs	Tempo (min)	MSE Trein	MSE Valid	MSE Teste
LM1	4	11.19	2.37e-10	1.56e-10	5.31e-07
LM2	4	106.15	4.24e-04	4.63e-02	2.57e-02
LM3	6	38.02	1.63e-04	1.72e-01	2.20e-01
LM4	8	20.44	4.20e-04	2.25e-04	1.43e-03
LM5	4	19.18	4.33e-05	1.26e-03	1.84e-04
LM6	7	14.52	3.32e-06	1.11e-06	8.79e-03
LM7	3	20.56	2.39e-04	6.24e-05	7.08e-05
LM8	3	20.21	8.70e-04	3.70e-02	5.04e-02
LM9	3	27.01	2.49e-03	2.45e-02	5.13e-02
LM10	3	39.57	5.15e-04	3.54e-03	5.51e-02
LM11	4	2.53	2.11e-04	2.16e-03	4.72e-03
LM12	2	4.06	7.74e-04	2.46e-03	4.13e-03
LM13	3	14.30	3.29e-04	6.89e-08	5.31e-06
LM14	4	7.60	2.35e-06	6.25e-04	1.10e-03
LM15	4	3.33	2.47e-06	2.12e-04	3.37e-03
LM16	4	4.48	1.26e-04	1.01e-06	1.84e-05
LF1	6	4.29	1.49e-06	6.19e-08	1.97e-04
LF2	3	3.35	4.68e-04	3.14e-06	2.65e-06
LF3	5	24.34	1.87e-05	5.96e-03	5.67e-05
LF4	3	13.36	6.57e-04	3.86e-03	2.21e-03
LF5	2	39.32	6.98e-05	1.80e-10	2.25e-02
LF6	6	110.09	9.09e-02	1.40e-01	2.93e-01
LF7	2	14.15	2.01e-05	2.27e-02	1.64e-02
LF8	7	22.24	2.10e-03	2.46e-02	2.06e-02

Assim como no HMM, escolheu-se um limiar para cada locutor utilizando as elocuções de treinamento elevando-o até eliminarmos todas as falsas aceitações.

A Figura 5.4 representa o treinamento de uma MLP para o locutor masculino 1, em azul temos o MSE dos dados de treinamento (função desempenho), em verde o MSE dos dados de validação cruzada e em vermelho o MSE dos dados de teste, a lista preta na horizontal indica o objetivo a ser atingindo pela função desempenho. Esta rede convergiu em 4 "epochs" e o tempo de treinamento foi de 11,19 minutos.

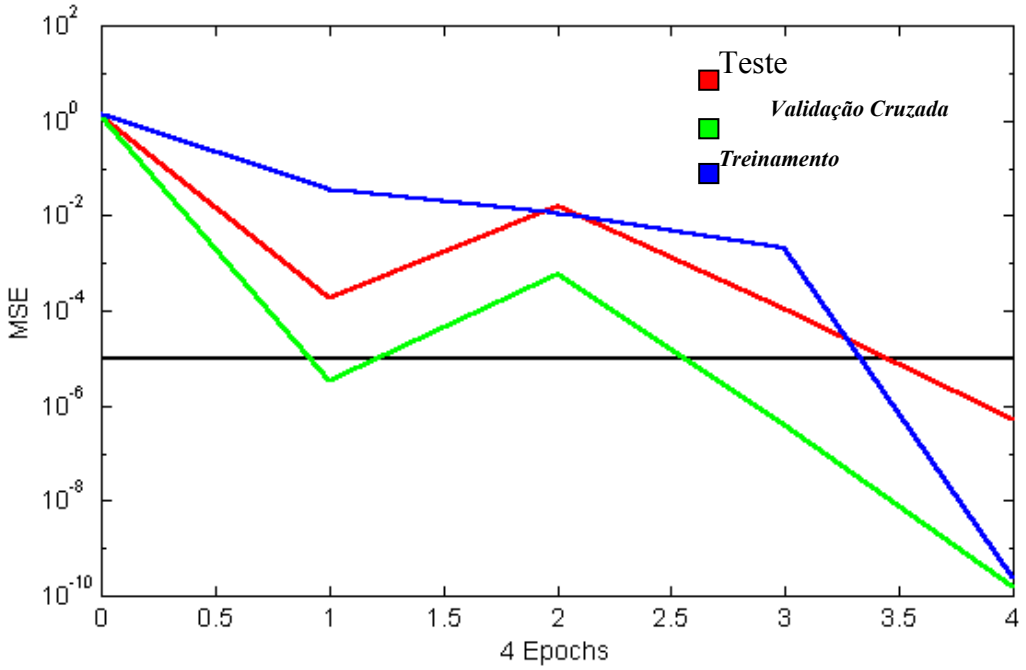


FIGURA 5.4: Treinamento de uma MLP para verificação automática do locutor.

5.6 – CONSIDERAÇÕES SOBRE O SISTEMA HÍBRIDO 1 – HIB1

Descreveremos, agora, os procedimentos utilizados para implementação do sistema híbrido 1 dos quais se destacam: a montagem dos dados, o treinamento das redes, as modificações realizadas no HMM e o algoritmo de treinamento.

5.6.1 – Montagem dos Dados e Treinamento das MLP's

O primeiro passo para a construção do modelo híbrido 1 foi a montagem da matriz de dados e alvos usados para treinar a rede neural de forma a levarmos o contexto em consideração. Escolheu-se um contexto igual a 9 ($2p + 1$, onde $p = 4$) (BOURLARD, 1991), ou seja, quatro observações (janelas) concatenadas à esquerda e quatro à direita. Os alvos para esta matriz são inicialmente gerados pela divisão do número total de observações pelo número de estados N ,12 no nosso caso; chamamos este processo de segmentação inicial, sendo o resto desta divisão, caso houver, somado ao último estado. Vamos supor, como exemplo, que temos uma elocução com 100 janelas após a segmentação inicial teríamos 8 observações nos 11 primeiros estados mais 12 (8 + 4) observações no último estado, já que 4 é o resto da divisão de 100 por 12. Como mencionado no Capítulo 4, o alvo é uma matriz 12xR, onde a observação que foi segmentada no estado 1, por exemplo, recebe o valor 1 na linha 1, correspondente ao primeiro estado, e 0 nos demais (codificação "one-from-k"). A Figura 5.5 mostra uma matriz de entrada, com contexto igual a 5 e número de janelas igual a 9, e seu alvo, gerado pela segmentação inicial, onde o número de estados é igual a 4.

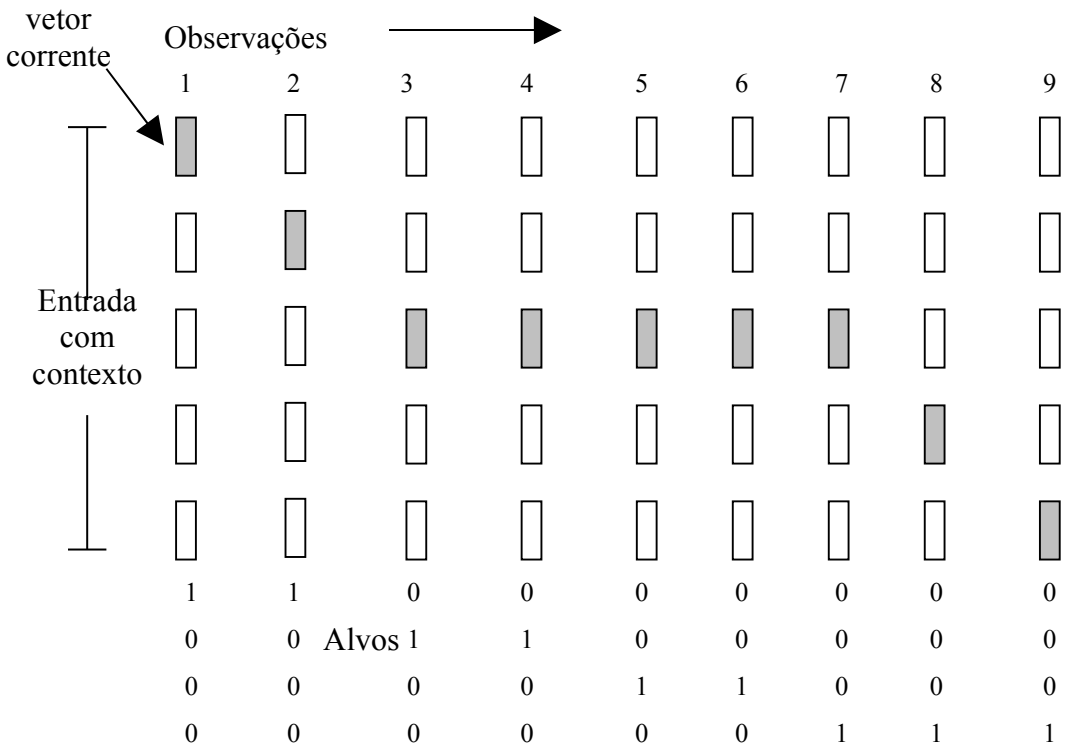


FIGURA 5.5: Matriz de entrada e alvos para o sistema híbrido 1.

5.6.2 – Topologia e Treinamento da MLP para o Sistema Híbrido

Após a montagem da matriz de entrada e dos alvos, por meio da segmentação inicial, é criada uma rede com 234 neurônios na entrada, 50 na camada escondida e 12 na camada de saída, as duas camadas com função de transferência logística sigmoidal (*logsig*). O número de neurônios na camada de entrada se justifica pelo fato de termos 26 características por 9 janelas contextuais na matriz de entrada e na camada de saída por termos 12 estados no HMM, pois será a rede que irá estimar as probabilidades de saída para os 12 estados do HMM, a Figura 5.6 mostra esquematicamente a rede utilizada no sistema híbrido, onde $P(q/x)$ é a probabilidade da classe q dado o vetor de entrada x.

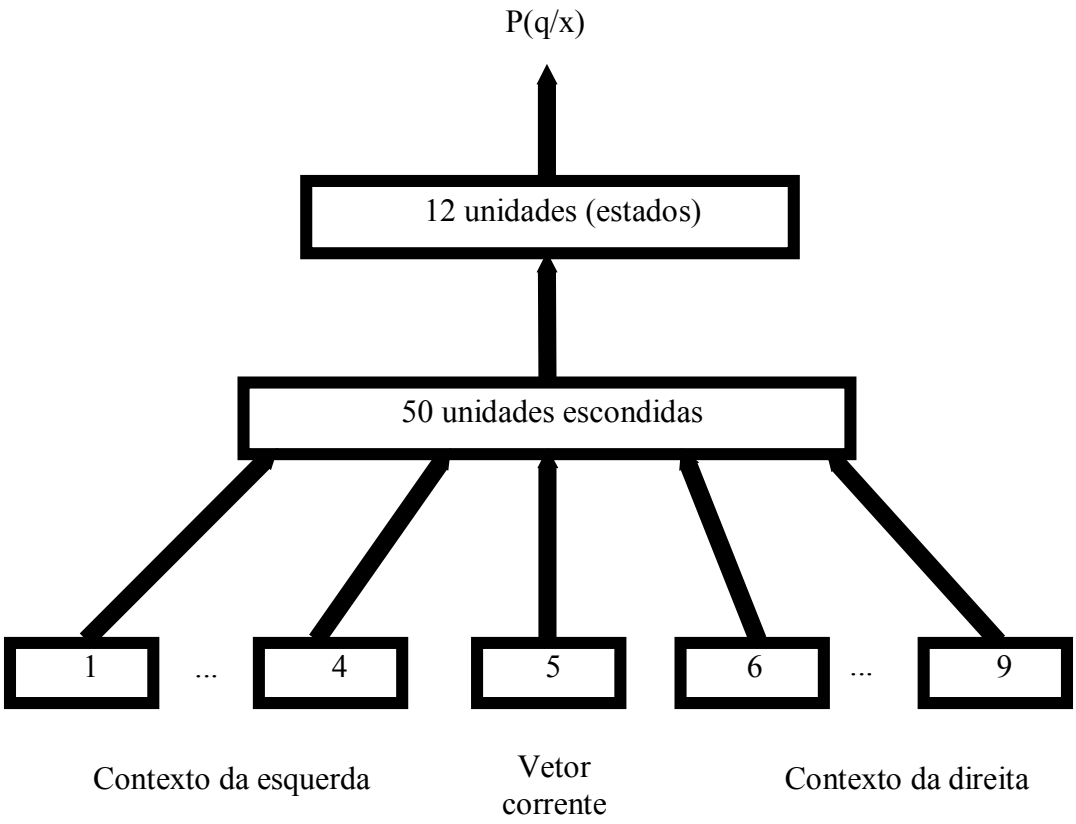


FIGURA 5.6: MLP utilizada no sistema híbrido 1.

Para treinamento desta rede utilizou-se o algoritmo "Resilient Backpropagation - (Rprop)" implementa pela função "TRAINRP" (DEMUTH, 1997), que obteve os melhores resultados quanto ao tempo de treinamento e convergência. O motivo para usarmos este algoritmo se deve pelo fato de que as funções sigmoidais têm uma inclinação muito próxima de zero quando os valores da função de ativação do neurônio se torna muito grande, e quando usamos o algoritmo do gradiente descendente para treinar uma MLP com funções sigmoidais o gradiente pode ter valores muitos pequenos causando, então, pequenas mudanças nos pesos e polaridades, mesmos estando estes longe de valores ótimos (DEMUTH, 1997). O algoritmo de treinamento "resilient backpropagation" elimina este problema, pois somente o sinal da derivada é usado para determinar a atualização dos pesos. Para maiores detalhes sobre este algoritmo consulte (DEMUTH, 1997), (RIEDMILLER, 1993).

A rotina para se criar uma rede com os parâmetros acima é a seguinte:

```
net_mlp = newff(minmax(mat_dad_trein),[50 nrestados],...
          {'logsig' 'logsig'},'trainrp');
```

onde, *net_mlp* é a rede criada e *mat_dad_trein* é a matriz de entrada da rede (Figura 5.4.). A rotina para treinamento é a seguinte:

```
[net_mlp,Tr]=train(net_mlp,mat_dad_trein,alvo_trein,[],[],v);
```

onde, *v* uma estrutura que contém os dados e alvos para validação cruzada, *mat_dad_trei*n e *alvo_trein* os dados e alvos de treinamento, respectivamente.

O erro desejado para os dados de treinamento foi ajustado para 10^{-3}, o treinamento terminava quando este erro era atingindo pelos dados de treinamento ou pela parada determinada pelo conjunto de validação cruzada (todas as redes treinadas, para o sistema híbrido, pararam sempre por este último critério). O treinamento de uma MLP para o locutor masculino 1 (LM1) é mostrado na Figura 5.7, onde, em azul temos o MSE para os dados de treinamento e em verde o MSE para os dados de validação cruzada.

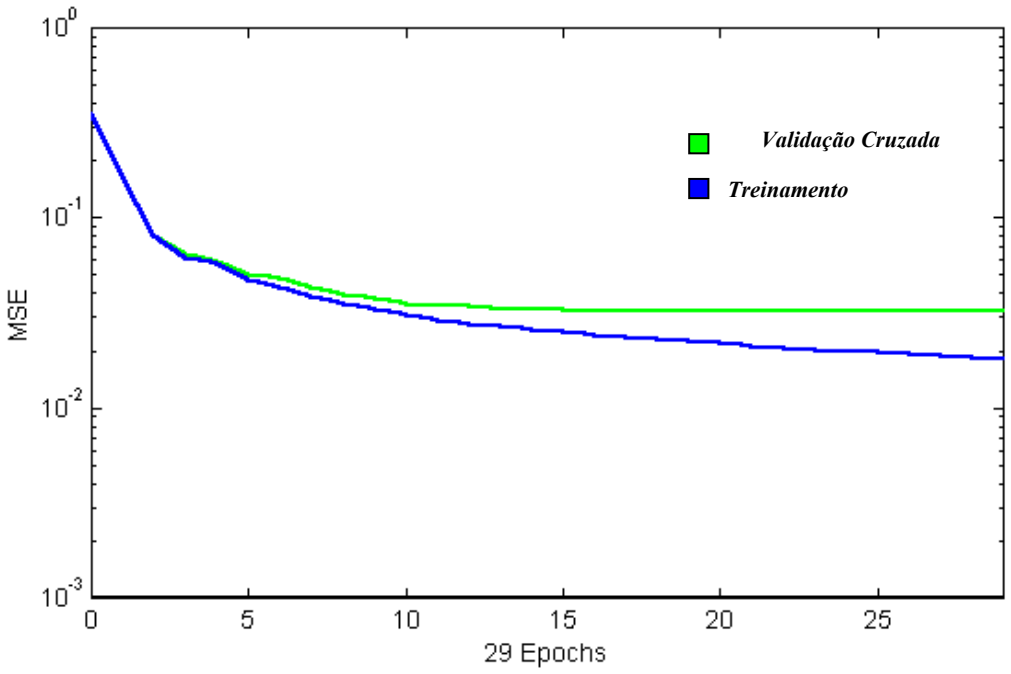


FIGURA 5.7: Treinamento de uma MLP para o sistema híbrido 1.

Podemos perceber que a rede não atinge um erro baixo o suficiente para produzir bons resultados (0.018 dados de treinamento e 0.032 dados de validação) e uma possível explicação para isso é que os alvos fornecidos não permitiam que a rede convergisse, visto que estes deviam causar uma mistura dos dados entre diferentes classes, ou seja, os alvos estão separando, em classes diferentes, observações que representavam um mesmo fenômeno acústico.

5.6.3 – Considerações sobre o HMM para Utilização no Sistema Híbrido

Utilizou-se um HMM com 12 estados, modelo esquerda-direita e procedimento de treinamento "Segmental-Kmeans", a diferença desse com o utilizado isoladamente para verificação do locutor é que o HMM para o sistema híbrido 1 não utiliza as gaussianas para estimar as distribuições de probabilidades dos vetores acústicos "features", as MLP´s é que serão encarregadas da estimação de probabilidades (MAP), quanto à estimação da matriz de transição de estados, modelamento temporal, não houve nenhuma alteração em relação ao algoritmo original descrito na Seção 3.2.

5.6.4 – Modificações da Saída da Rede Treinada

Uma vez que a rede foi treinada com a segmentação inicial, descrita na Seção 5.6.1, esta estará pronta para a estimação inicial das probabilidade de saída para o HMM.

Para garantir que as saídas das MLP´s sejam constituídas por probabilidades, troca-se a função de transferência, *logsig,* da camada de saída da rede por uma função de transferência *sofmax*, equação 3.35, antes de estimarmos as probabilidade de saída que serão enviadas para o HMM. No treinamento da rede a função de transferência *softmax* é substituída pela *logsig*.

Um procedimento também utilizado antes de enviarmos as probabilidades estimadas pelas redes ao HMM foi a divisão pela probabilidade *a priori*, Seção 4.3.1, calcula-se a probabilidade *a priori*, para cada estado, da seguinte forma: conta-se o número de observações segmentadas nos estados e divide-se o resultado pelo número total de observações.

5.6.5 – Interagindo MLP's com o HMM

Feitos os procedimentos descritos na seção anterior, as estimações das probabilidades *a posteriori* (MAP) feitas pela MLP já podem ser enviadas ao HMM que as armazena na matriz b_j, a partir daí, o algoritmo de Viterbi junto com o método de Baum-Welch utilizam estas probabilidades na estimação da matriz de transição a_{ij}, com o algoritmo "Segmental K-means". Após os dados serem segmentados pelo algoritmo de Viterbi, esta segmentação é transformada em alvo, como descrito na Seção 5.6.1, que são enviados para a MLP que será treinada novamente da mesma forma como foi descrito na Seção 5.6.2, após este novo treinamento a rede fornecerá novas estimações de probabilidade para o HMM.

Este processo se repete até que o crescimento do valor das verossimilhanças dos dados de treinamento seja inferior a um limiar. O treinamento deste sistema híbrido é mostrado esquematicamente na Figura 5.8.

No reconhecimento, os dados para teste são simulados pela rede treinada. Às saídas são aplicadas os procedimentos realizados na Seção 5.6.4. Feito isto, estas saídas, que são estimações da máxima probabilidade a *posteriori*, são enviadas para o HMM que através do algoritmo de Viterbi calcula o caminho de maior verossimilhança.

Por último, as verossimilhanças são divididas pelo número de janelas da respectiva elocução (PARANAGUÁ, 1997). Esta normalização cancela a influência do tamanho da elocução no valor da verossimilhança calculada pelo algoritmo de Viterbi, já que elocuções muito grandes terão verossimilhanças igualmente grandes.

A Figura 5.9 mostra o resultado da verificação do locutor masculino 1 utilizando este sistema. As verossimilhanças do locutor verdadeiro estão circuladas. Como pode ser visto nesta figura, apesar do sistema ter reconhecido as elocuções 1 e boa parte das elocuções 17, pertencentes a este locutor, a probabilidade de falsa aceitação é muito elevada. Na Figura 5.10, observamos que o resultado, para verificação deste mesmo locutor com o HMM, foi mais satisfatório que o do sistema híbrido pois, além de reconhecer todas as elocuções número 1 e 17, a probabilidade de falsas aceitações neste sistema é menor que no sistema híbrido 1.

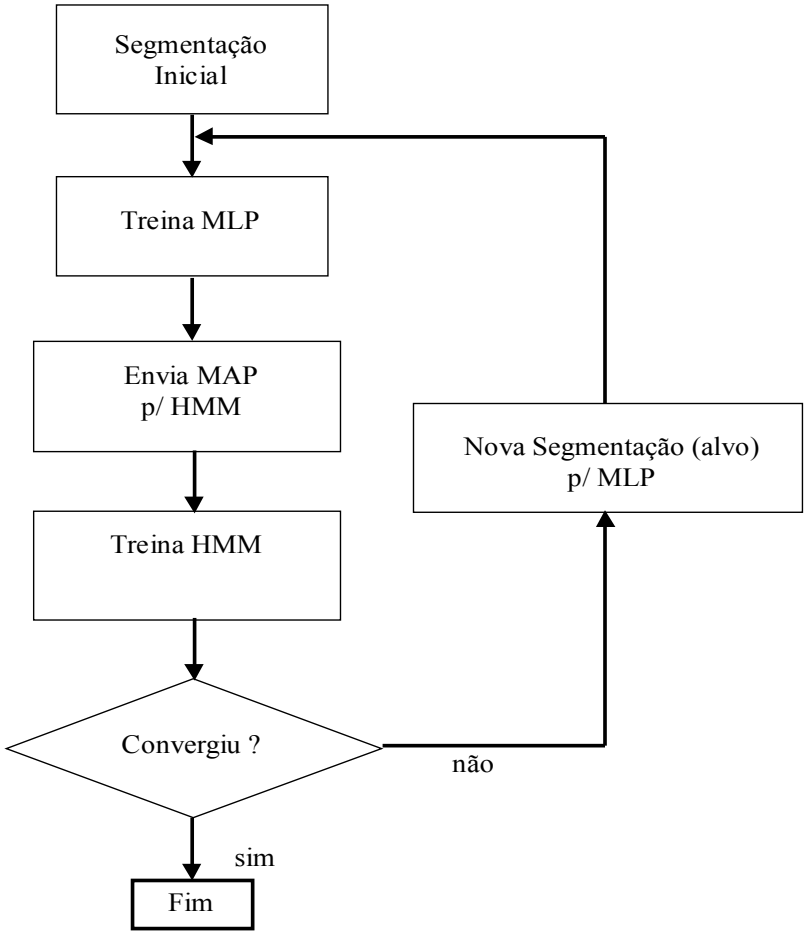


FIGURA 5.8: Algoritmo de treinamento do sistema híbrido 1.

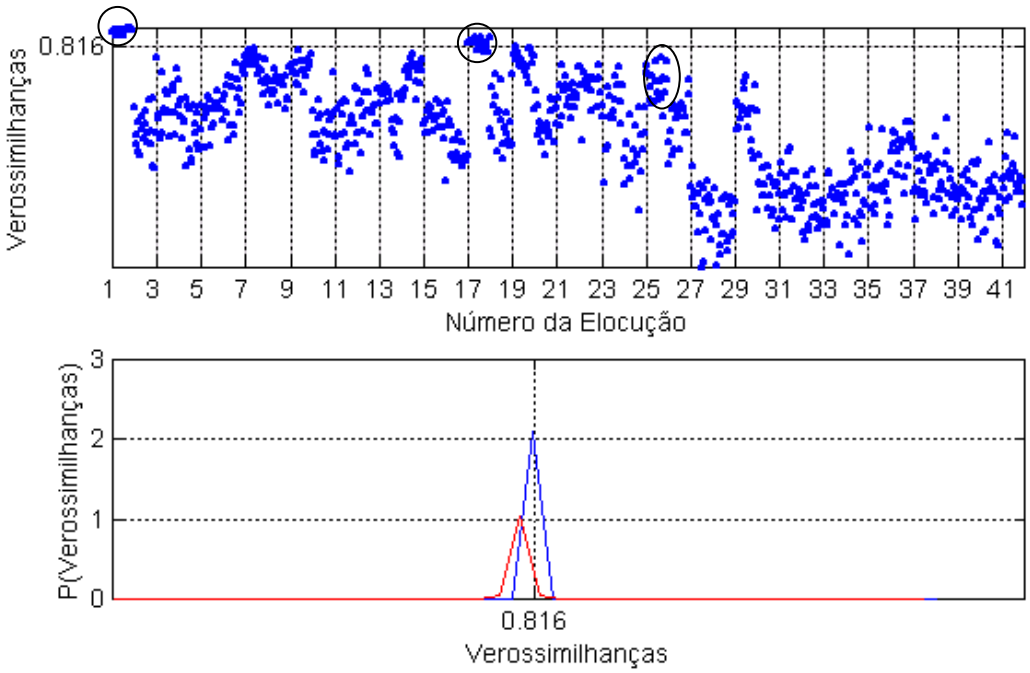


FIGURA 5.9: Resultado da verificação do locutor masculino 1 com o sistema híbrido 1. **(a)**: Verossimilhanças. **(b)**: Probabilidades das verossimilhanças.

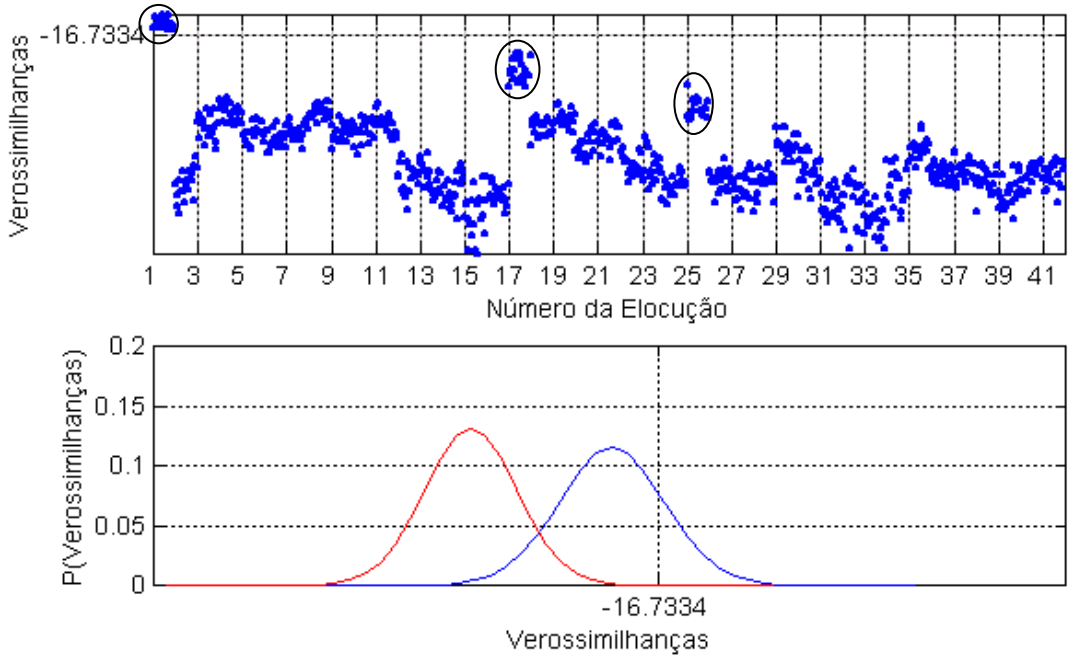


FIGURA 5.10: Resultado da verificação do locutor masculino 1 com HMM. **(a)**: Verossimilhanças. **(b)**: Probabilidades das verossimilhanças.

Podemos perceber, com os resultados acima, que o sistema híbrido 1 teve um desempenho inferior ao do HMM, concluímos, então, que não se consegue obter um treinamento discriminativo utilizando MLP's como estimadoras de probabilidades em um sistema de reconhecimento do locutor, dependente do texto,

baseado no critério da estimação da máxima verossimilhança. Um fato importante, para chegarmos a essa conclusão, é que as MLP's usam o critério da máxima probabilidade *a posteriori* que não fornece nenhuma informação da probabilidade de um dado vetor ser observado em uma determinada classe. A Figura 5.11 poderá esclarecer melhor a estimação de probabilidades *a posteriori* e de verossimilhanças. Considerando o ponto *A*, nesta figura, se usássemos estimadores de verossimilhanças gaussianas poderíamos afirmar que as probabilidades,

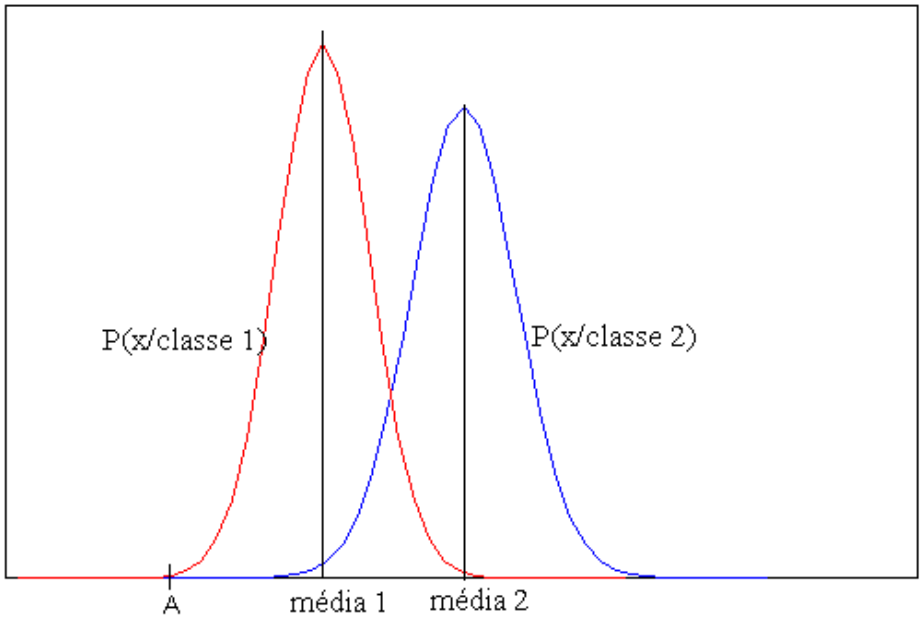


FIGURA 5.11: Estimação da probabilidade *a posteriori* e da verossimilhança (RENALS, 1992).

geradas por ambas as classes 1 e 2, seriam baixas para a observação *A*. No entanto, comparando estas pequenas probabilidades, poderíamos afirmar que é muito mais provável que *A* seja gerado pela classe 1 do que pela classe 2. Já a estimação da probabilidade *a posteriori*, através de MLP's, nos informaria que é mais provável que *A* pertença a classe 1 do que a 2, mas não nos diria qual a probabilidade que esta classe gera para a observação *A* (RENALS, 1992). Ora, como só há uma classe modelada, formada pelas elocuções do locutor a ser treinado, na fase de treinamento do sistema híbrido em questão, não havendo, então, uma segunda classe que representasse os locutores falsos, as redes neurais teriam que trabalhar como estimadores de verossimilhanças e, como acabamos de explicar, estas não fornecem este tipo de estimação.

Devido aos resultados insatisfatórios do sistema híbrido 1, decidiu-se, então, a implementação de um outro sistema híbrido – sistema híbrido 2.

5.7 – SISTEMA HÍBRIDO 2 – HIB2

HMM's modelam a duração de eventos a partir das probabilidades de transições, sendo que, a probabilidade de *d* observações consecutivas, $p(d_j)$, serem observadas no estado *j* de um HMM pode ser modelada por uma MLP.

Utilizou-se o método alternativo para introduzir a informação da duração de estado, que implica em medirmos diretamente as seqüências segmentadas das elocuções de treinamento através do algoritmo de Viterbi (RABINER, 1993). A nova verossimilhança, levando-se em conta a duração de estado, é calculada da seguinte forma:

$$\log \hat{P}(q,x \mid \lambda) = \log P(q,x \mid \lambda) + \alpha_d \sum_{j=1}^{N} \log[p_j(d_j)] \qquad (5.4)$$

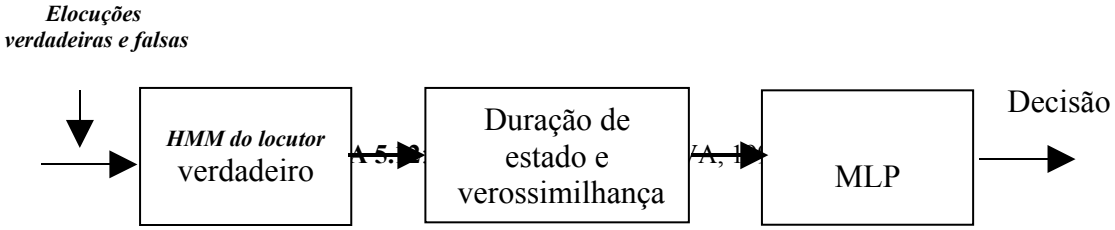
onde α_d é peso atribuído a duração d do estado, e d_j é a duração do estado j ao longo do caminho ótimo obtido pelo algoritmo de Viterbi. Utiliza-se uma MLP para estimar tanto α_d como $p(d_j)$, o logaritmo no segundo membro desta equação é calculado pelo HMM.

Em experimentos realizados em (SILVA, 1997) e (THOMÉ, 1999), Dirceu chegou à conclusão que a duração de estados é uma informação útil e discriminativa para reconhecimento de voz. Este fato sugeriu que utilizássemos a duração de estado para reconhecimento do locutor dependente do texto.

Implementou-se, então, o sistema híbrido 2, com algumas modificações significativas em relação aquele implementado em (SILVA, 1997). Neste último, utilizou-se uma única rede neural que recebia como entrada as verossimilhanças, sem normalização, e a duração de estados, também sem normalização, das elocuções utilizadas para treinar as palavras. Os alvos eram formados por um código de bits ortogonais, atribuídos à cada elocução usada no treinamento de uma determinada palavra. Treinava-se a rede neural até atingir-se um erro desejado.

No caso do sistema híbrido 2, utiliza-se uma rede para cada locutor, como é feito para verificação do locutor, consegue-se, com isso, um treinamento mais discriminativo, já que a rede neural irá se especializar em apenas um locutor (locutor verdadeiro) sendo que todos os outros locutores são incluídos em uma única classe, a dos locutores falsos, que modela o mundo exterior, a Figura 5.12 mostra o diagrama de blocos deste modelo.

Outras modificações importantes feita no sistema híbrido 2, foram a utilização de verossimilhanças normalizadas junto com a duração de estados, também normalizada. Na normalização da duração de estados o número de segmentações obtidas em cada estado são

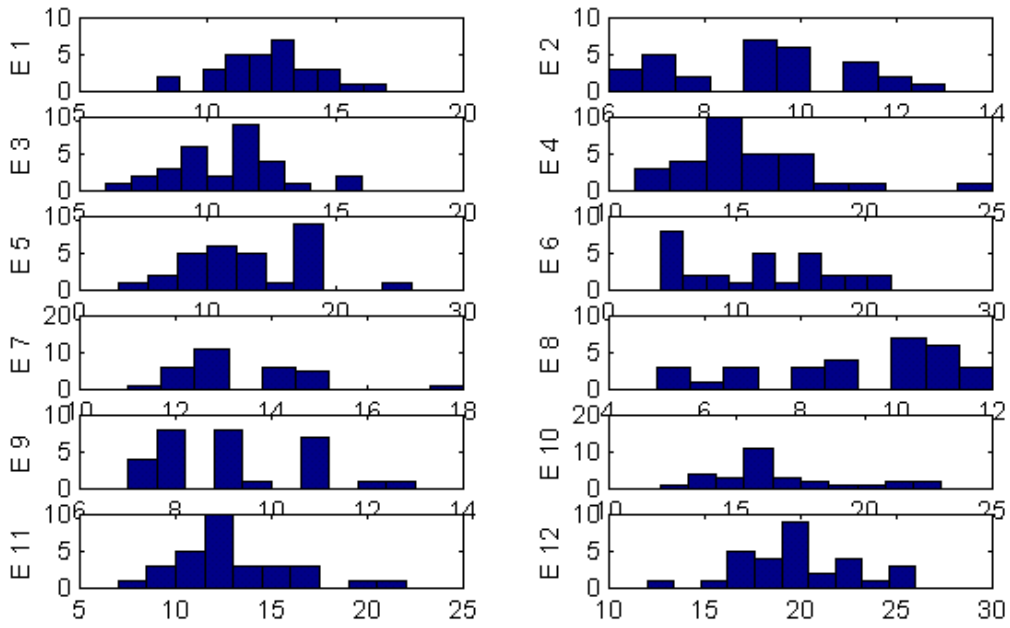


divididas pelo número total de janelas da elocução. Este procedimento cancela o efeito de elocuções com variações de tamanho elevados em relação as elocuções de treinamento. Comprovou-se, em testes realizados, que estas normalizações melhoraram o desempenho do sistema. Feito isto, realizaram-se duas normalizações, Seção 5.5.1, uma para a duração de estado e uma para a verossimilhança.

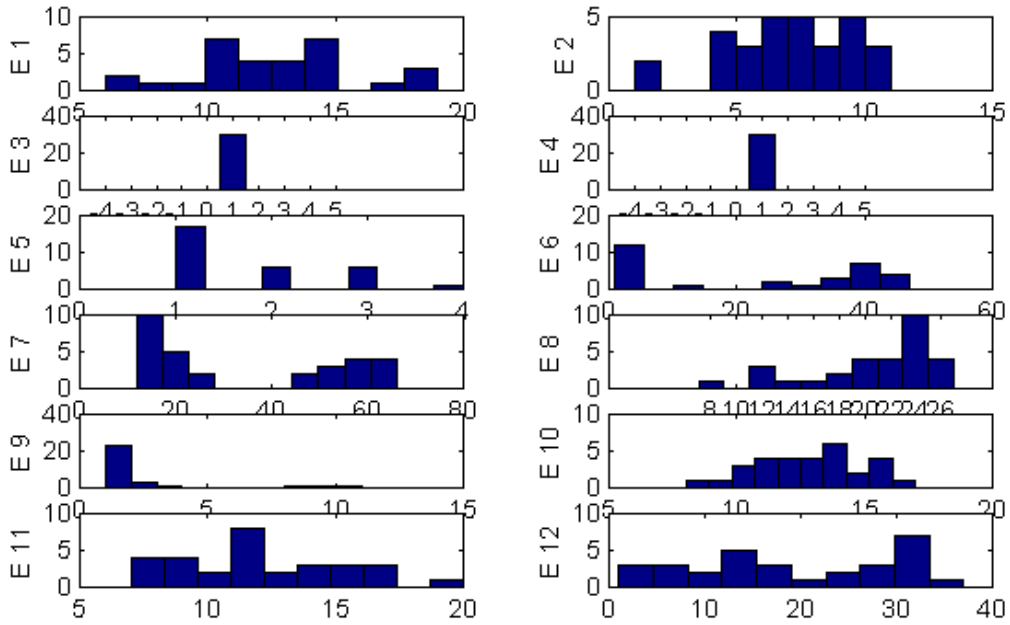
A Figura 5.13 (a) e (b) mostra o histograma de $p(d_j)$ para as 30 elocuções usadas no treinamento dos locutores masculinos 1 e 4, obtido com o HMM do locutor masculino 1. Podemos observar, nesta figura, que os locutores possuem duração de estados bem diferentes, o que possibilita, então, um treinamento discriminativo entre locutores.

Para a montagem da matriz de entrada de dados para a MLP utilizaram-se as 60 repetições das elocuções do locutor verdadeiro. Notar que selecionaram-se somente 30 delas para treinamento (Seção 5.5.1), e 30 elocuções, de treinamento, dos demais locutores. Atribui-se o valor 1 aos alvos correspondentes às elocuções verdadeiras e 0 as demais, como mostra a Figura 5.14.

Para esse sistema, escolheu-se um limiar igual a 0,8, pois, com esse limiar, não houve falsas rejeições e nem falsas aceitações para as elocuções de treinamento.



(a)



(b)

FIGURA 5.13: Histogramas de $p(d_j)$. **(a)** Locutor masculino 1. **(b)** Locutor masculino 4.

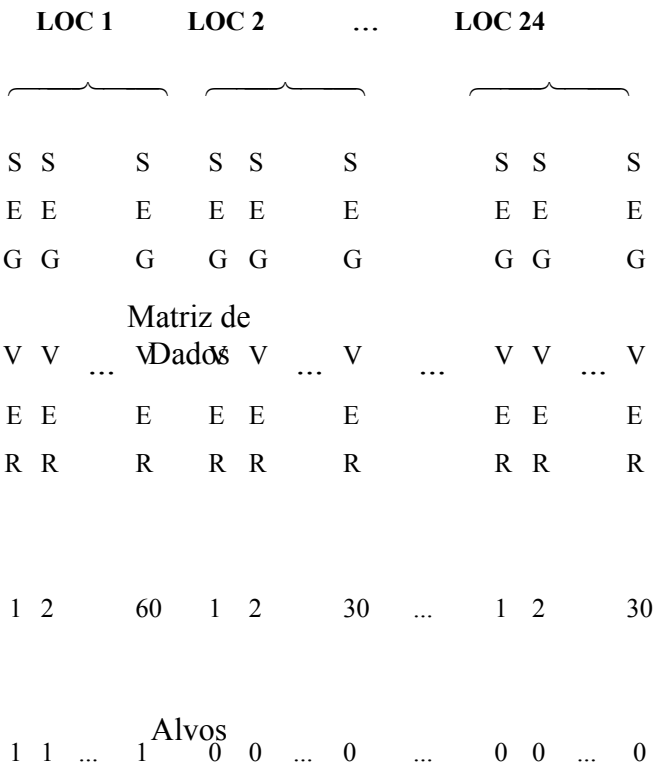


FIGURA 5.14: Matriz de dados e alvos para o sistema híbrido 2.

A Figura 5.15 mostra o treinamento de uma MLP para a verificação do locutor masculino 1 utilizando o sistema híbrido 2.

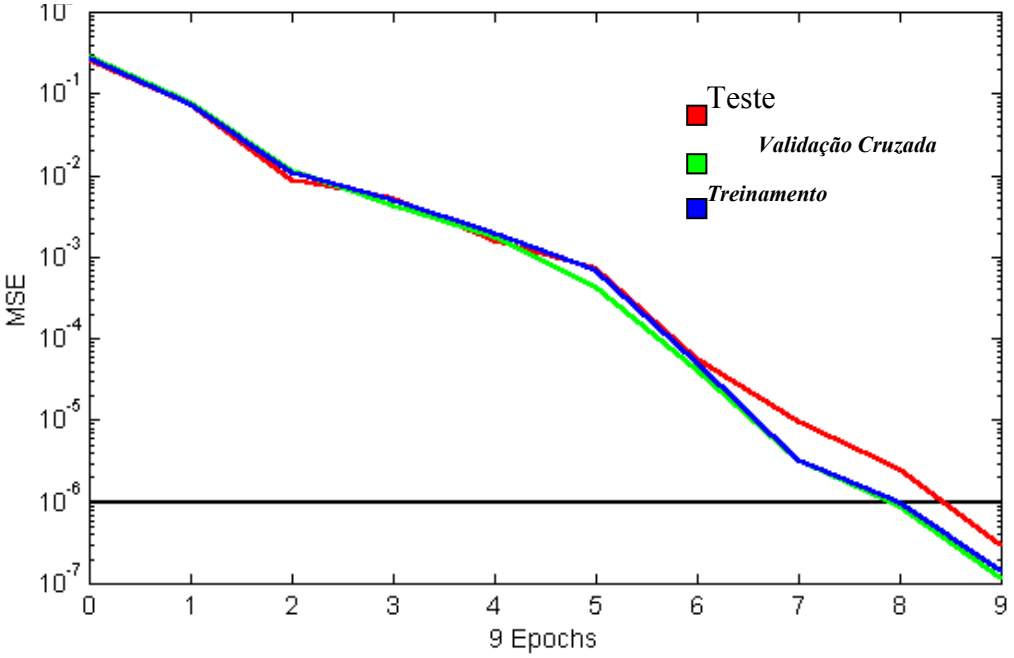


FIGURA 5.15: Treinamento de uma MLP para o sistema híbrido 2.

A Tabela 5.6 mostra o número de "epochs", MSE e o tempo de treinamento das redes utilizadas para implementação deste sistema.

TABELA 5.6: Dados de treinamento das MLP para o sistema híbrido 2.

LOC	Epochs	Tempo (min)	MSE Trein	MSE Valid	MSE Teste
LM1	10	0.12	1.41e-07	1.16e-07	2.96e-07
LM2	07	0.13	1.40e-06	1.82e-05	4.93e-05
LM3	09	0.06	3.87e-08	4.34e-08	1.77e-08
LM4	09	0.06	2.06e-07	9.54e-08	1.01e-07
LM5	09	0.16	4.83e-06	4.33e-04	3.22e-05
LM6	09	0.08	1.08e-06	1.65e-05	4.08e-06
LM7	05	0.11	4.22e-06	3.87e-06	3.08e-05
LM8	08	0.09	2.29e-06	6.36e-06	1.46e-05
LM9	04	0.05	2.57e-06	4.29e-05	3.50e-04
LM10	09	0.13	6.40e-06	9.14e-06	4.04e-05
LM11	08	0.10	9.46e-06	1.22e-05	1.31e-05
LM12	06	0.07	6.29e-06	1.01e-05	6.68e-04
LM13	06	0.07	4.67e-06	1.64e-05	5.37e-06
LM14	05	0.07	3.95e-06	5.91e-06	9.54e-06
LM15	08	0.10	4.74e-06	5.52e-06	4.87e-06
LM16	08	0.09	7.54e-06	1.53e-05	6.80e-05
LF1	07	0.05	2.71e-07	4.06e-05	8.49e-06
LF2	12	5.11	9.60e-08	7.92e-07	7.07e-06
LF3	09	0.10	7.33e-06	3.71e-06	1.16e-03
LF4	05	0.24	6.50e-06	2.74e-05	1.99e-04
LF5	09	0.33	4.43e-06	4.23e-05	6.79e-04
LF6	07	0.41	3.91e-06	3.78e-03	9.80e-05
LF7	10	0.43	3.86e-06	5.60e-03	7.08e-03
LF8	07	0.17	1.34e-06	1.88e-06	1.07e-04

5.8 – PROGRAMAS DESENVOLVIDOS

Os programas utilizados neste trabalho de mestrado foram desenvolvidos no aplicativo MatLab 5.3 (“The MathWorks, Inc”) e pela ferramenta de software C++ Builder 3 (“Borland International, Inc”). Estes programas, disponibilizados em cd, podem ser encontrados no laboratório de processamentos de sinais de voz do IME.

CAPÍTULO 6

RESULTADOS OBTIDOS E AVALIAÇÃO DOS SISTEMAS

6.1 – INTRODUÇÃO

Descreveremos neste capítulo os resultados obtidos com os HMM's as MLP's e o sistema híbrido 2 (HIB 2), assim com a medida de desempenho utilizada para avaliarmos e compararmos estes três sistemas.

Utilizaram-se dois grupos de elocuções: no primeiro encontram-se as elocuções de teste dos locutores treinados; no segundo estão as elocuções de locutores não treinados (locutores desconhecidos). Juntaram-se estes dois grupos em um só para o teste dos sistemas. Como nenhuma destas elocuções participaram do treinamento dos modelos, obteve-se uma avaliação mais rigorosa dos sistemas. Nestes testes não foi verificado o desempenho dos sistemas quanto à verificação de locutores mímicos. Ao todo foram usadas 820 elocuções para teste, 20 para cada um dos 41 locutores.

6.2 – MEDIDA DE DESEMPENHO UTILIZADA

Avaliaram-se os desempenhos dos sistemas para verificação automática do locutor utilizando a taxa de igual erro ("equal error rate – EER"). Obtém-se o EER através do erro *E*, que é calculado a partir da probabilidade de erro de falsa rejeição e da probabilidade de erro de falsa aceitação (VUUREN, 1999) da seguinte forma:

$$\begin{aligned} E &= E_{fr} + E_{fa} \\ &= C_{fr}\, p(fr/H_0) P(H_0) + C_{fa}\, p(fa/H_1) P(H_1) \end{aligned} \qquad (6.1)$$

onde C_{fr} e C_{fa} são os custos associados aos erros de falsa rejeição e falsa aceitação, respectivamente, H_0 a hipótese da elocução pertencer ao locutor verdadeiro, H_1 a hipótese da elocução pertencer a um locutor falso, $p(fr/H_0)$ a probabilidade de falsa rejeição e $p(fa/H_1)$ a probabilidade de falsa aceitação. O EER pondera a probabilidade de erro de falsa rejeição e a probabilidade de erro de falsa aceitação igualmente e é definido como o erro mínimo para o qual $C_{fr} = C_{fa} = 1$ e $P(H_0) = P(H_1) = 0{,}5$.

Nesta dissertação, o EER foi calculado elevando-se o limar até que todas as falsas aceitações fossem eliminadas, pois em aplicações do mundo real uma falsa aceitação, um impostor dado como verdadeiro em uma transação bancária, por exemplo, ocasionaria uma perda muito maior do que uma falsa rejeição. Este procedimento equivale a darmos uma ponderação maior para o custo de falsa aceitação C_{fa}.

6.3 – RESULTADOS OBTIDOS COM O HMM

Os resultados obtidos com o HMM constam na Tabela 6.1 onde *FR* é a probabilidade de erro de falsa rejeição, em porcentagem e *FA* a probabilidade de erro de falsa aceitação. Podemos observar que os erros mais significativos se encontram no locutor masculino 1 e feminino 1, o que era de se esperar, pois estes

locutores realizaram 2 gravações a mais que os demais, com duas variáveis importantes em relação às elocuções de treinamento que

TABELA 6.1: Resultado da verificação com HMM.

CÓDIGO LOCUTOR	VERIFICAÇÃO		
	FR (%)	FA (%)	EER (%)
LM1	66,66	0	33,33
LM2	15	0	7,5
LM3	10	0	5
LM4	5	0	2,5
LM5	0	0	0
LM6	15	0	7,5
LM7	5	0	2,5
LM8	0	0	0
LM9	0	0	0
LM10	25	0	12,5
LM11	5	0	2,5
LM12	15	0	7,5
LM13	15	0	7,5
LM14	5	0	2,5
LM15	10	0	5
LM16	0	0	0
LF1	66,66	0	33,33
LF2	10	0	5
LF3	10	0	5
LF4	25	0	12,5
LF5	15	0	7,5
LF6	5	0	2,5
LF7	5	0	2,5
LF8	10	0	5

tendem a prejudicar o desempenho dos sistemas de reconhecimento: microfone e período de gravação (Seção 5.2). A Figura 6.1 mostra o gráfico das verossimilhanças para alguns locutores masculinos e femininos, no eixo vertical temos o números das gravações (Tabela 5.1), no eixo horizontal os valores das verossimilhanças e a linha tracejada na horizontal representa o limiar. Os treinamentos duraram em média 6 minutos.

Podemos perceber que a probabilidade de ocorrer falsas aceitações, neste sistema, é elevada já que várias elocuções dos locutores falsos se aproximaram das do locutor verdadeiro, e isso em uma base de dados resumida com 41 locutores, utilizada nesta dissertação.

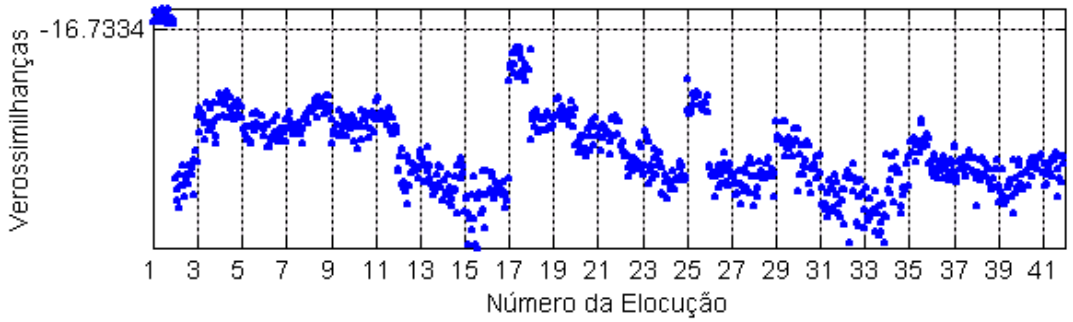


FIGURA 6.1: **(a)** Locutor masculino 1.

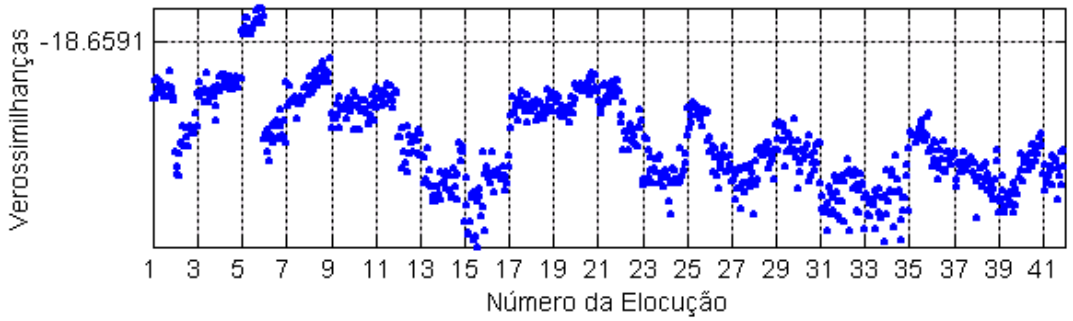


FIGURA 6.1: **(b)**Locutor masculino 5.

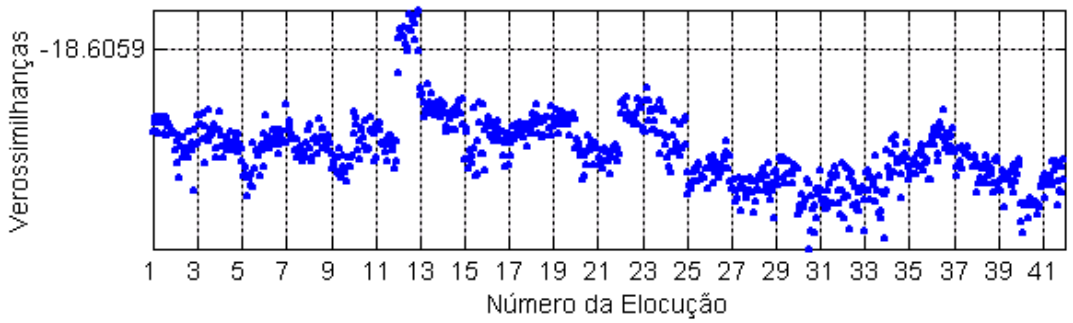


FIGURA 6.1: **(c)** Locutor masculino 12.

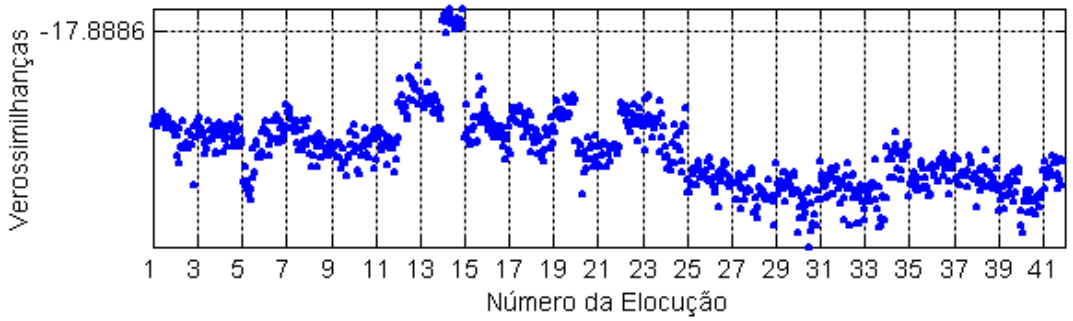


FIGURA 6.1: **(d)** Locutor masculino 14.

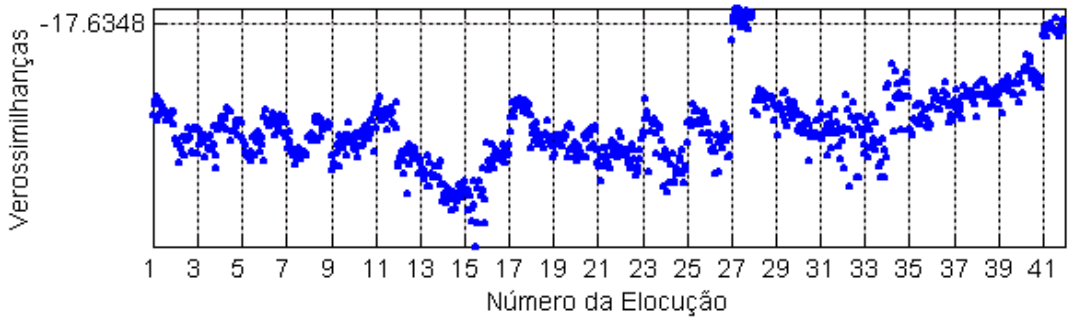


FIGURA 6.1: **(e)** Locutor feminino 1.

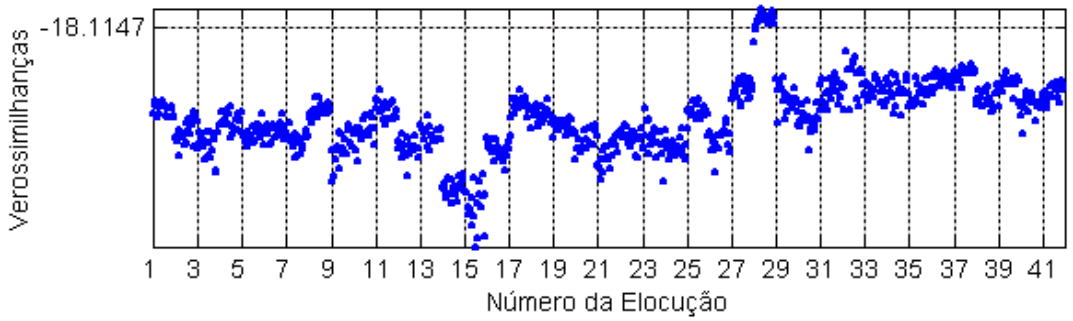


FIGURA 6.1: **(f)** Locutor feminino 2

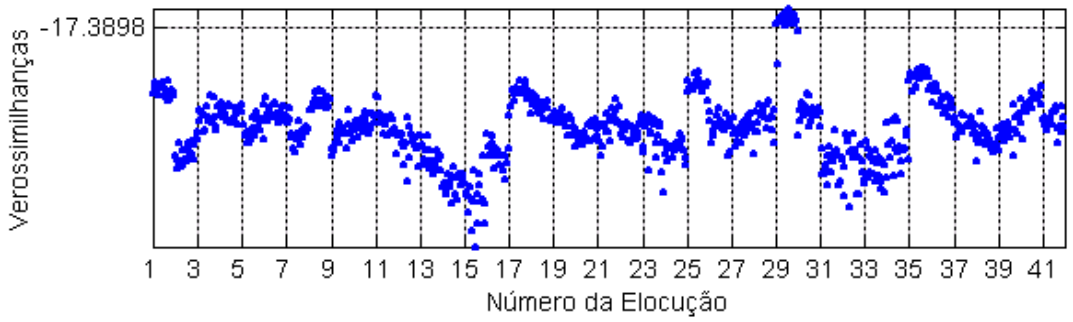


FIGURA 6.1: **(g)** locutor feminino 3

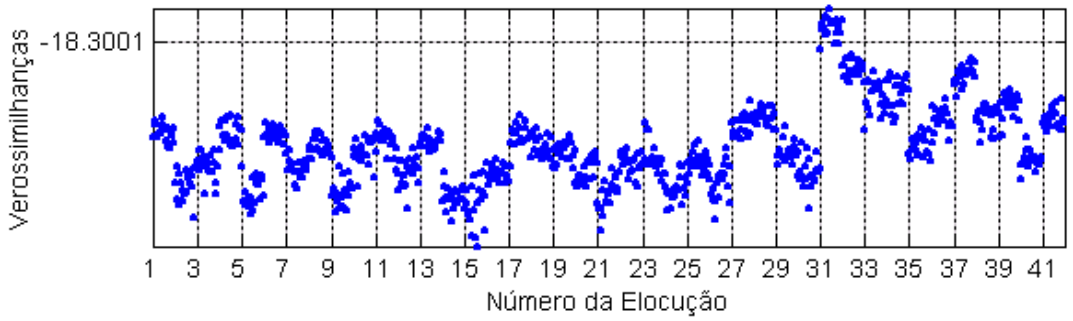


FIGURA 6.1: **(h)** Locutor feminino 5.

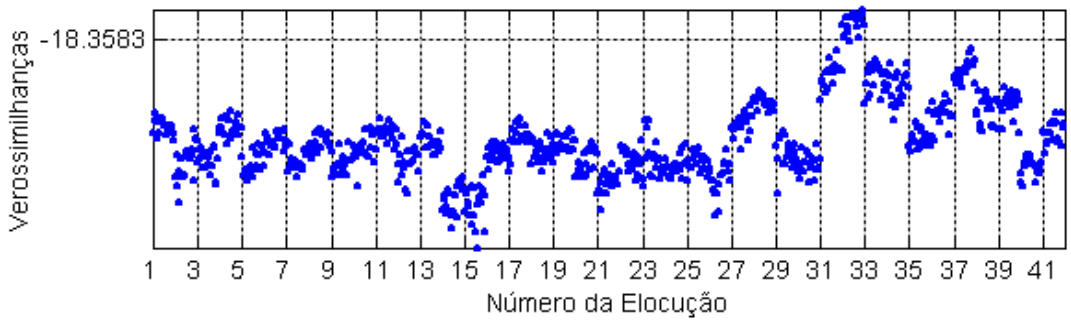


FIGURA 6.1: **(i)** Locutor feminino 6.

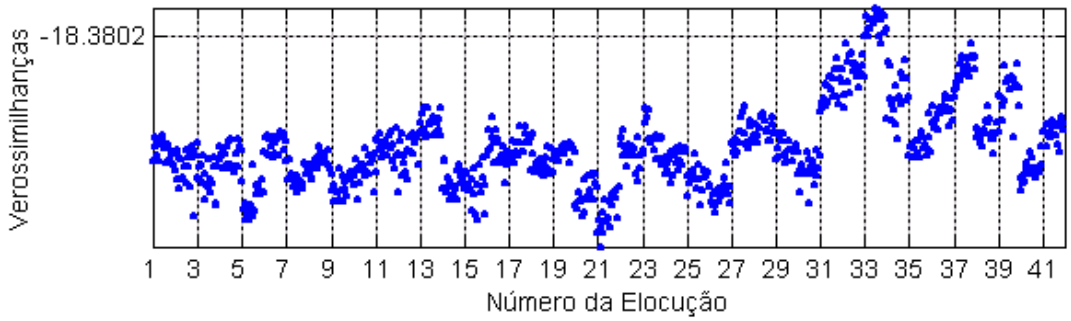


FIGURA 6.1: **(j)** Locutor feminino 7.

FIGURA 6.1: Gráfico das verossimilhanças para verificação com HMM. **(a) - (j)** Formados pelos locutores masculinos 1,5,12,14 e locutores femininos 1,2,3,5,6,7, nesta ordem.

6.4 – RESULTADOS OBTIDOS COM A REDE NEURAL

Os resultados obtidos com as MLP's são apresentados na Tabela 6.2. Nos testes realizados houveram falsas rejeições na grande maioria dos locutores treinados e várias falsas rejeições, visto que algumas elocuções dos locutores falsos, principalmente locutores desconhecidos, obtiveram na saída da rede valores iguais ou maiores do que várias elocuções verdadeiras. Uma explicação para isto é que a quantidade de dados usados para o modelamento da classe dos locutores falsos, que representa o mundo exterior, é insuficiente, o que gera, então, muitas regiões de indecisão.

TABELA 6.2: Resultado da verificação com as MLP's.

CÓDIGO LOCUTOR	VERIFICAÇÃO		
	FR (%)	FA (%)	EER (%)
LM1	40	0	20
LM2	25	0	12,5
LM3	20	0,25	11,25
LM4	5	0	2,5

LM5	10	0,5	5,25
LM6	15	0	7,5
LM7	5	0	2,5
LM8	40	0	20
LM9	20	0,13	10.06
LM10	25	0	12,5
LM11	20	0	10
LM12	5	0,25	2,63
LM13	0	0,13	0,06
LM14	5	0	2,5
LM15	5	0	2,5
LM16	0	0,5	0,25
LF1	10	0	5
LF2	0	0,13	0,06
LF3	45	0,25	22,63
LF4	25	0	12,5
LF5	30	0,25	15,13
LF6	50	0	25
LF7	30	0	15
LF8	15	0,5	7,75

A Figura 6.2 mostra o gráfico das verossimilhanças para os locutores masculino 1 e 5 e femininos 1, 5 e 7.

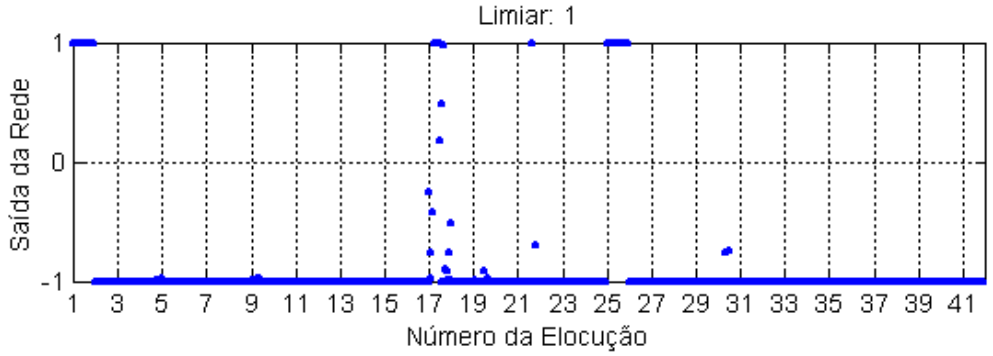


FIGURA 6.2: **(a)** Locutor masculino 1.

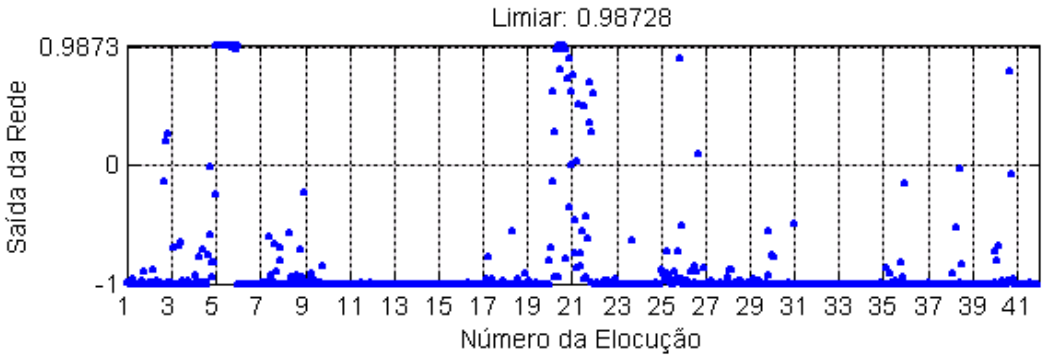


FIGURA 6.2: **(b)** Locutor masculino 5.

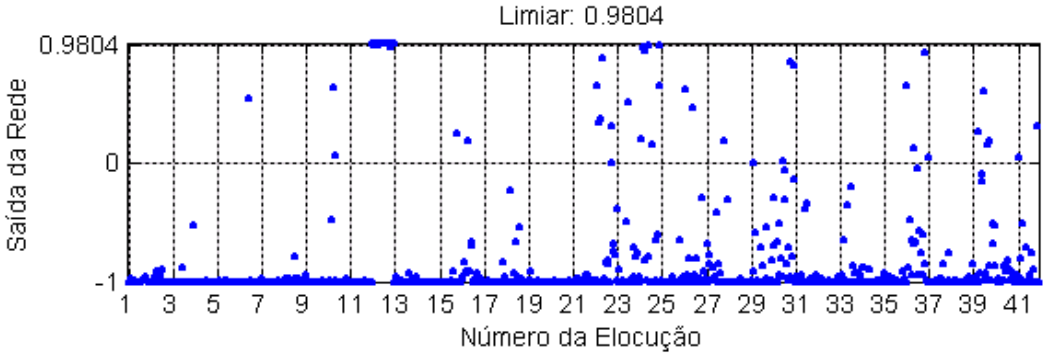


FIGURA 6.2: **(c)** Locutor masculino 12.

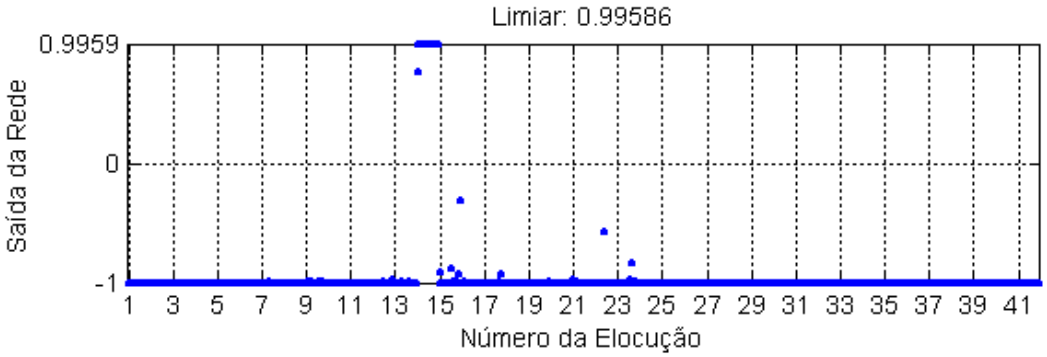


FIGURA 6.2: **(d)** Locutor masculino 14.

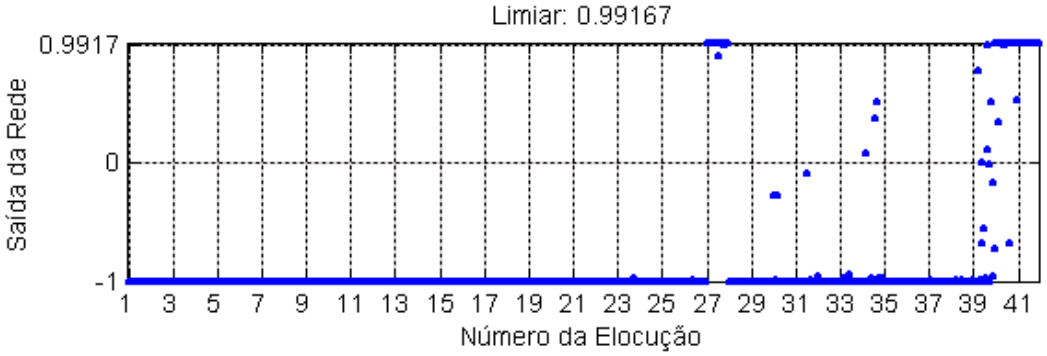


FIGURA 6.2: **(e)** Locutor feminino 1.

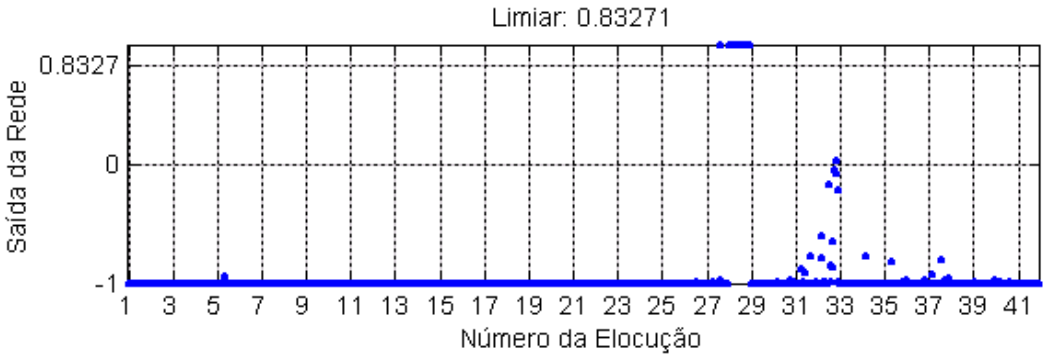


FIGURA 6.2: **(f)** Locutor feminino 2

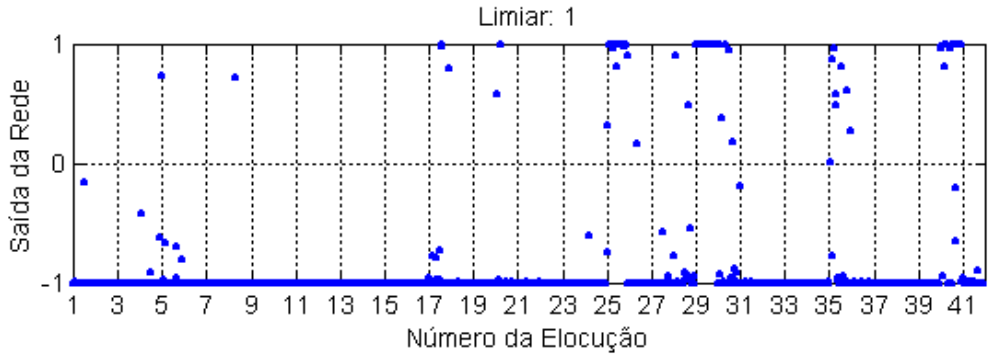


FIGURA 6.2: **(g)** Locutor feminino 3.

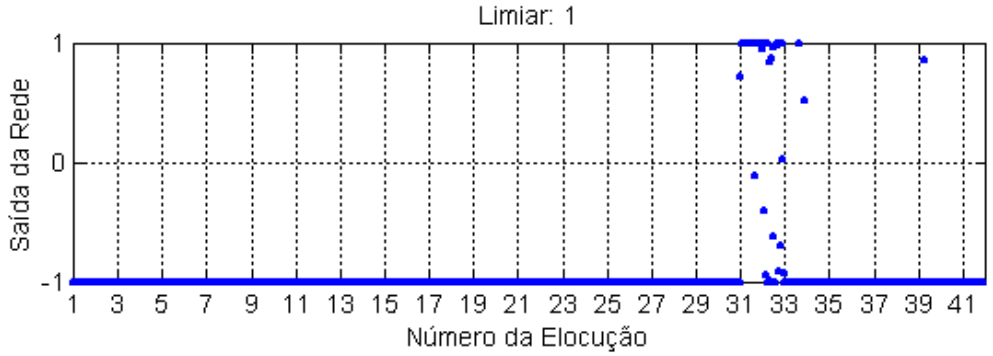


FIGURA 6.2: **(h)** Locutor feminino 5

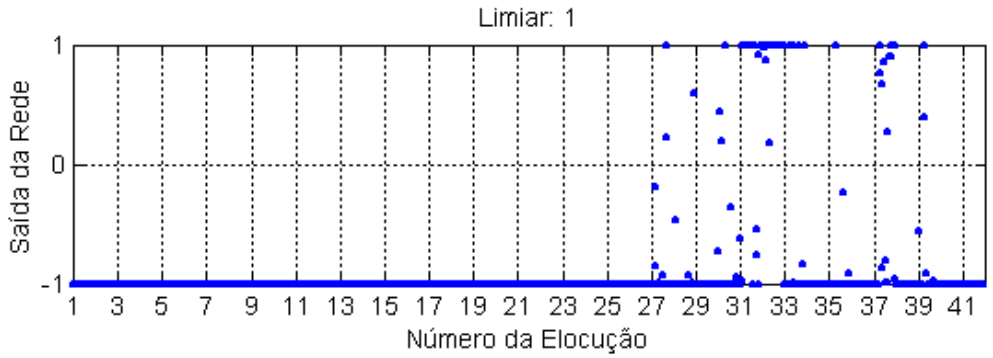


FIGURA 6.2: **(i)** Locutor feminino 6.

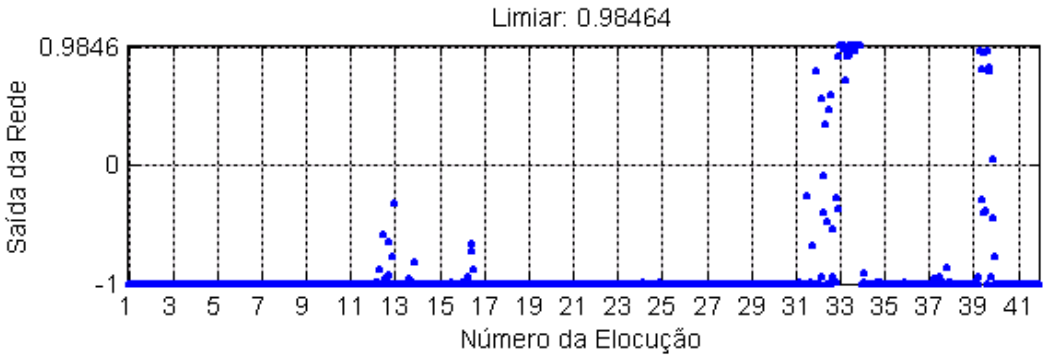


FIGURA 6.2: **(j)** Locutor feminino 7.

FIGURA 6.2: Gráfico das verossimilhanças para verificação com MLP. **(a)** – **(j)** Formados pelos locutores masculinos 1,5,12,14 e locutores femininos 1,2,3,5,6,7, nesta ordem.

6.5 – RESULTADOS OBTIDOS COM O SISTEMA HÍBRIDO 2

A Tabela 6.3 mostra os resultados obtidos com o sistema híbrido 2. A Figura 6.3 mostra a saída da rede para os locutores treinados. Como podemos observar, este sistema obteve os menores erros para todos os locutores, além do que, este sistema rejeitou mais fortemente as elocuções falsas, ou seja a probabilidade de falsas aceitações é bem menor, principalmente para os locutores masculinos. Obteve-se um desempenho um pouco inferior para os locutores femininos, porém, os outros dois sistemas apresentaram esta mesma deficiência. Mesmo assim, para os locutores femininos, o sistema híbrido apresentou os melhores resultados.

TABELA 6.3: Resultado da verificação com o sistema híbrido 2.

CÓDIGO LOCUTOR	VERIFICAÇÃO		
	FR (%)	FA (%)	EER (%)
LM1	60	0	30
LM2	0	0	0
LM3	5	0	2,5
LM4	0	0	0
LM5	0	0	0
LM6	0	0	0
LM7	0	0	0
LM8	0	0	0
LM9	0	0	0
LM10	0	0	0

LM11	0	0	0
LM12	5	0	2,5
LM13	0	0	0
LM14	0	0	0
LM15	0	0	0
LM16	0	0	0
LF1	33,33	0	16,66
LF2	0	0	0
LF3	5	0	2,5
LF4	0	0	0
LF5	5	0	2,5
LF6	0	0	0
LF7	0	0	0
LF8	0	0	0

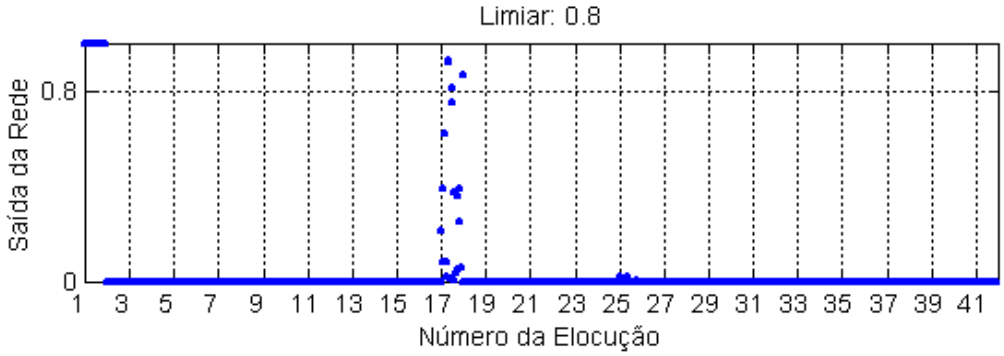


FIGURA 6.3: **(a)** Locutor masculino 1.

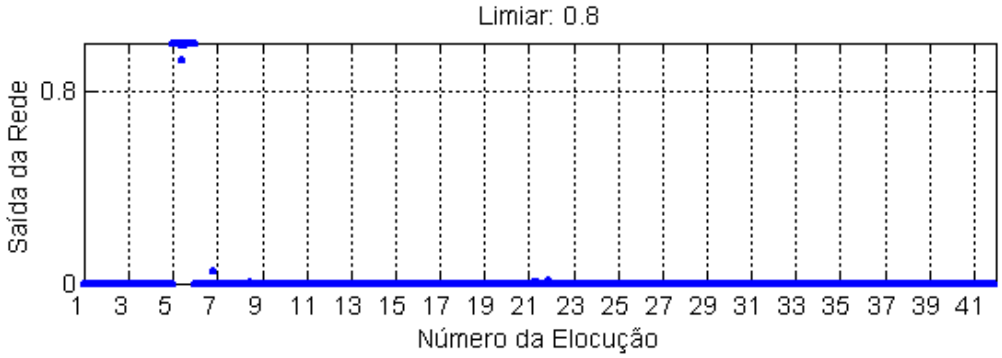


FIGURA 6.3: **(b)** Locutor masculino 5.

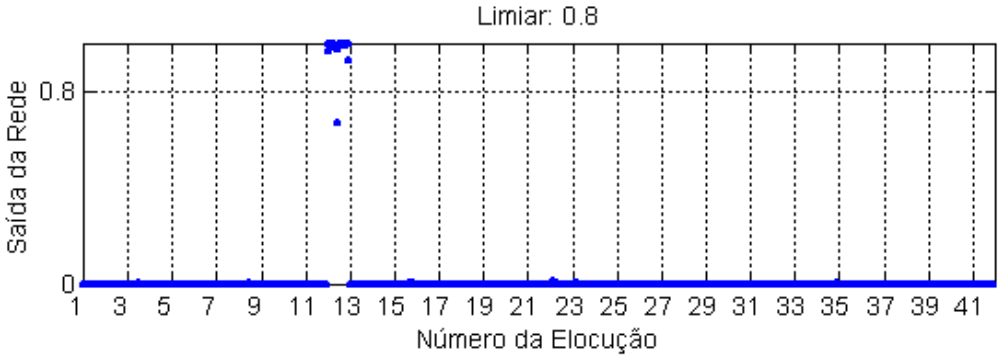


FIGURA 6.3: **(c)** Locutor masculino 12.

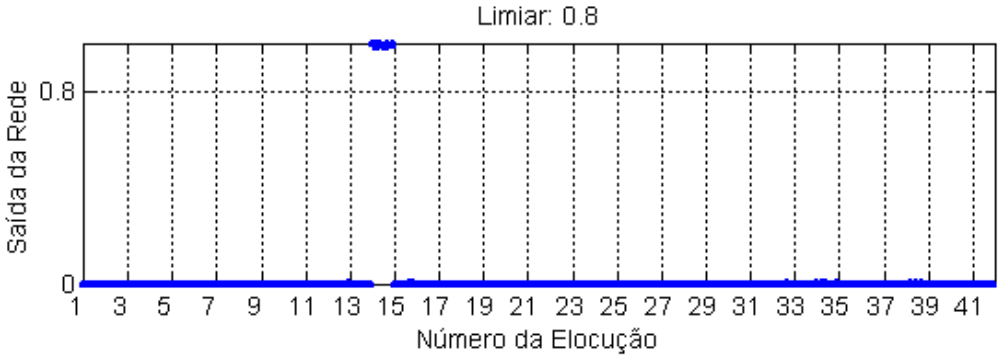


FIGURA 6.3: **(d)** Locutor masculino 14.

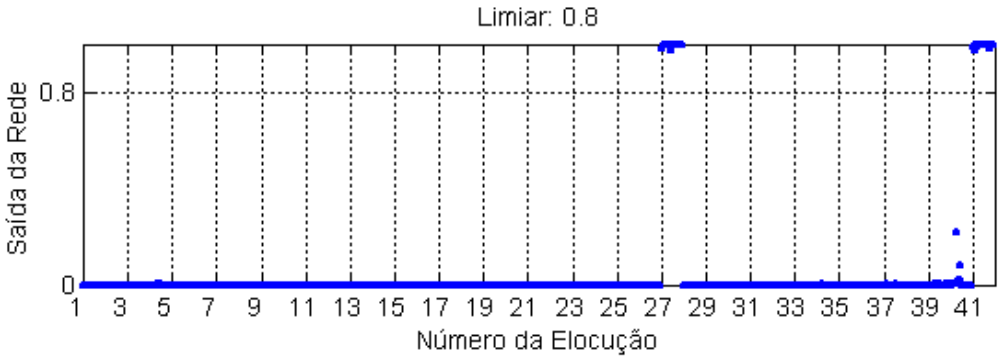


FIGURA 6.3: **(e)** Locutor feminino 1.

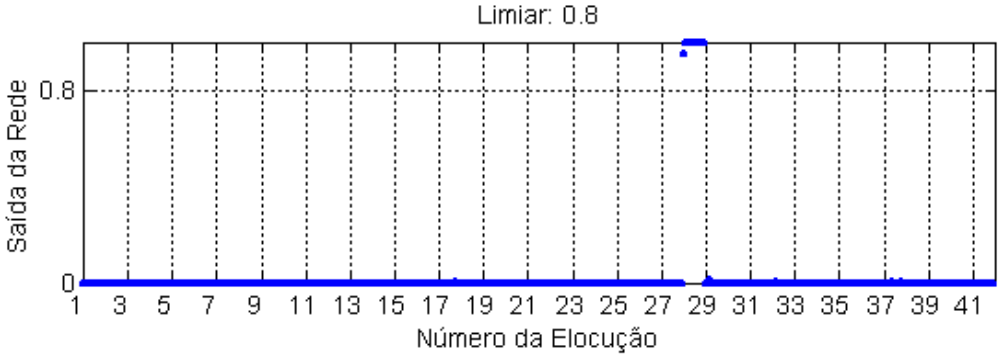


FIGURA 6.3: **(f)** Locutor feminino 2

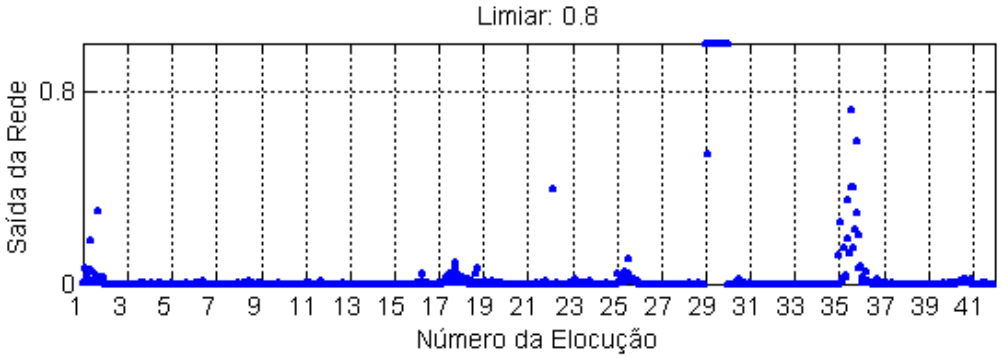


FIGURA 6.3: **(g)** Locutor feminino 3.

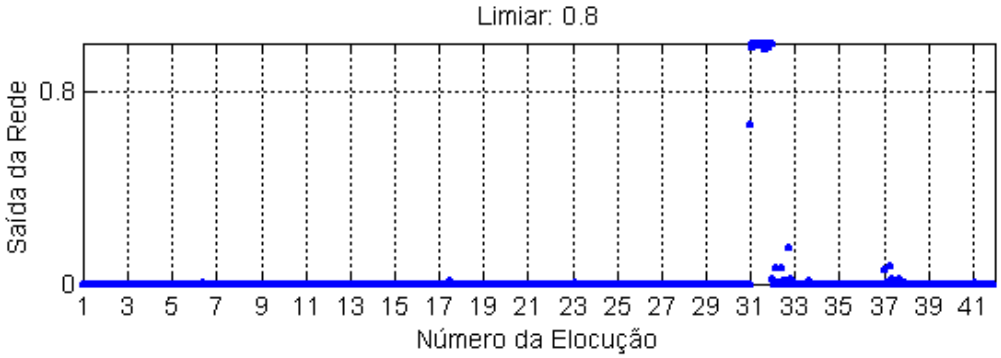


FIGURA 6.3: **(h)** Locutor feminino 5

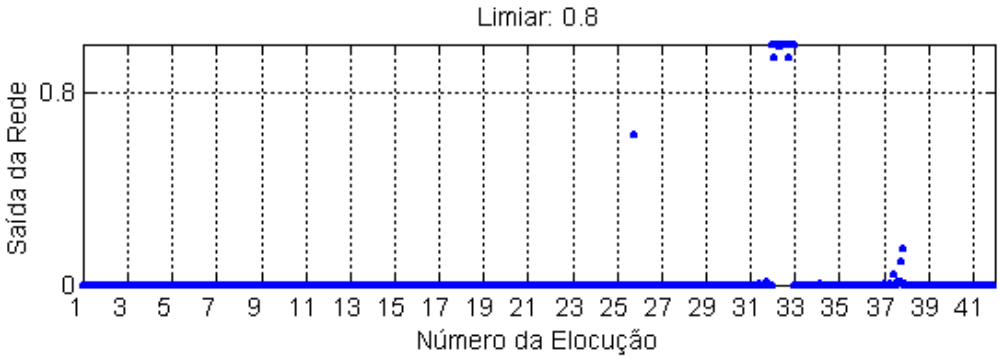


FIGURA 6.3: **(i)** Locutor feminino 6.

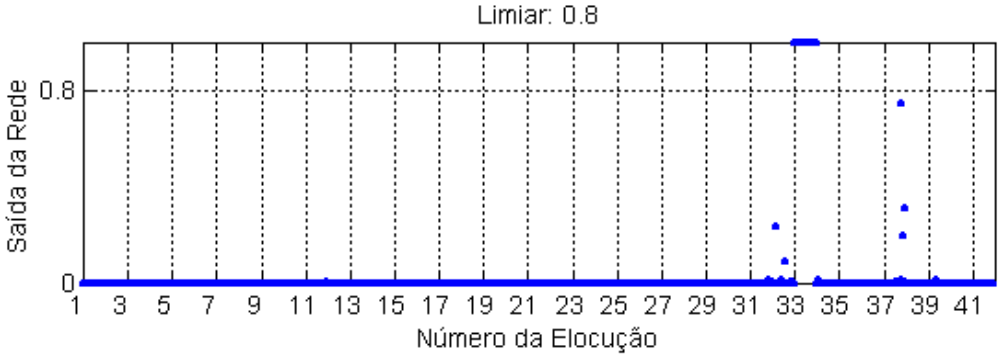


FIGURA 6.3: **(j)** Locutor feminino 7.

FIGURA 6.3: Gráfico das verossimilhanças para verificação com o sistema híbrido 2. **(a)** – **(j)** Formados pelos locutores masculinos 1,5,12,14 e locutores femininos 1,2,3,5,6,7, nesta ordem.

6.6 – RESULTADO COMPARATIVO ENTRE OS TRÊS SISTEMAS

A Tabela 6.4 mostra o resultado da verificação para os locutores masculino 1 e feminino 1 utilizando o HMM e o sistema híbrido 2. Nestes locutores verificou-se o maior número de falsas rejeições devido às variações, descritas na Seção 5.2, introduzidas nas elocuções dos referidos locutores. Nesta tabela G1, G17 e G25 referem-se às elocuções dos grupos 1, 17 e 25 do locutor masculino 1, G27, G40 e G41 referem-se às elocuções dos grupos 27, 40 e 41 do locutor feminino 1, S1 e S2 referem-se ao HMM e ao sistema híbrido 2, respectivamente, # indica o número da elocução de teste e "+" indica que houve erro de falsa rejeição naquela elocução.

TABELA 6.4: Resultado da verificação para os locutores masculino 1 e feminino 1.

#	G1		G17		G25		G27		G40		G41	
	S1	S2	S1	S2	S1	S2	S1	S2	S1	S2	S1	S2
2			+	+	+	+	+		+	+	+	
6			+	+	+	+			+	+	+	
8			+	+	+	+			+	+	+	
12			+	+	+	+			+	+	+	
14			+	+	+	+			+	+	+	
18			+	+	+	+			+	+	+	
20			+		+	+			+	+	+	
24			+		+	+			+	+		
26			+	+	+	+			+	+	+	
30			+	+	+	+	+		+	+	+	
32			+		+	+			+	+	+	
36			+	+	+	+			+	+	+	
38			+	+	+	+			+	+		
42			+	+	+	+			+	+	+	
44			+	+	+	+			+	+	+	
48			+	+	+	+	+		+	+	+	
50			+	+	+	+			+	+	+	
54			+	+	+	+			+	+	+	
56			+	+	+	+			+	+	+	
60			+		+	+			+	+		

Podemos concluir, observando esta tabela, que todas as elocuções dos grupos 25 e 40, gravadas com o microfone diferente daquele utilizada nas elocuções de treinamento, foram rejeitadas mostrando a sensibilidade dos dois sistemas quanto a mudanças no canal de gravação. Já nas elocuções dos grupos 1 e 27, gravadas nas mesmas condições que as de treinamento, obtiveram-se os melhores resultados. Nas elocuções dos grupos 25 e 41, gravadas em épocas diferentes das de treinamento, obteve-se um resultado satisfatório apenas para as elocuções do grupo 41 com o sistema híbrido 2. Isto demostra, também, que os dois sistemas têm seus desempenhos afetados pelo período da gravação.

Para compararmos o desempenho global dos três sistemas, calculou-se a média ponderada da probabilidade de falsa rejeição, falsa aceitação e do EER. A Tabela 6.5 mostra esses resultados.

TABELA 6.5: Resultado comparativo entre os três sistemas.

	VERIFICAÇÃO		
	FR (%)	**FA (%)**	**EER (%)**
HMM	14,10	0	7,05
MLP	18,54	0,12	8,96
HIB 2	4,72	0	2,36

Conclui-se, observando a tabela acima, que o sistema híbrido 2 teve o melhor desempenho em relação aos demais. Devemos, no entanto, ressaltar que as elocuções verdadeiras que foram rejeitadas (falsas rejeições), assim como as falsas, foram rejeitadas fortemente nesse sistema. Esse fato, dependendo da aplicação, poderá dificultar a melhoria do sistema híbrido 2, caso se deseje minimizar as falsas rejeições.

CAPÍTULO 7

CONCLUSÕES E SUGESTÕES

Podemos concluir, com os resultados obtidos, que a utilização de modelos híbridos podem proporcionar um desempenho melhor do que os modelos isolados, empregados na tarefa de verificação automática do locutor dependente do texto. É importante deixar bem claro que, devido ao pequeno número de locutores utilizados, nenhuma comprovação definitiva pode ser tirada dos resultados. Mais testes, então, poderiam ser realizados utilizando-se uma base de dados maior. Neste caso, poderíamos verificar realmente se houve aumento no desempenho do sistema híbrido 2.

Concluímos também que não se pode obter um treinamento discriminativo utilizando redes neurais "Multilayer Perceptrons" como estimadoras de probabilidades de classes de saída condicionadas a entrada (MAP) em um sistema de reconhecimento do locutor, dependente do texto, baseado no critério da estimação da máxima verossimilhança. Desta conclusão, surge a contribuição mais importante desta dissertação: a de que, para obtermos um desempenho satisfatório no sistema híbrido 1, descrito no Capítulo 4, devemos inserir nesse as elocuções do locutores falsos.

As MLP's não obtêm um bom resultado no reconhecimento de locutores desconhecidos, cujas locuções não foram apresentadas na fase de treinamento da rede, pois a quantidade de dados utilizados no treinamento, para modelar a classe dos locutores falsos, é insuficiente para obtermos um modelamento preciso dessa classe.

Obtém-se um resultado razoável com HMM, porém, nesse sistema, a probabilidade de erros de falsa aceitação é elevada.

A duração de estado e a verossimilhança, após normalizados pelo número de janelas, são informações úteis que proporcionam um treinamento discriminativo em um sistema híbrido.

Como sugestão para pesquisas futuras sugerimos:

1 – Modificar o sistema híbrido 1, inserindo nesse, elocuções de locutores falsos, possibilitando, desta forma, um treinamento discriminativo.

2 – Utilizar redes recorrentes no treinamento do sistema híbrido 2, onde, a matriz de entrada dessa rede seria a duração de estados fornecida pelo HMM.

3 – Utilizar rede neurais para análise discriminante não linear ("Nonlinear Discriminant Analysis - NDA") que realizam uma transformação não linear dos dados de entrada, de tal forma que possam ser facilmente discriminados.

4 – Utilizar uma base de dados maior e reconhecida internacionalmente.

REFERÊNCIAS BIBLIOGRÁFICAS

ANDRADE, M. A. R., **Reconhecimento Automático de Comandos Conectados,** Tese de Mestrado, IME, 1999.

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

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

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

BOURLARD, H & WELLEKENS C.J., **Links between Markov Models and Multilayer Perceptrons**, IEEE Trans. on Pattern Analysis and Machine Intelligence. 1990.

BOURLARD H. & MORGAN N., **Continuous Speech Recognition by Connectionist Statistical Methods**, IEEE Transactions on Neural Networks, Vol 4, nº 6, 1993.

BOURLARD, H. & MORGAN, N, **Hybrid HMM/ANN Systems for Speech Recognition: Overview and New Research Directions**, http://citeseer.nj.nec.com/cs, 1997.

BOURLARD, H. & MORGAN, N, **Speaker Verification: A Quick Overview**, IDIAP Research Report, August 1998.

DAVIS, S. B. & P. MERMELSTEIN, **Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences**, Trans. IEEE ASSP, Vol 28, pp. 357-366, 1980.

DEMUTH, H. & BEALE, M., **Neural Networks Toolbox - For Use with MATLAB**, MathWorks, Inc., User's Guide, Version 3.0, 1997.

DINIZ, S. S., **Uso de Técnicas Neurais para o Reconhecimento de Comandos à Voz**, Tese de Mestrado, IME, 1997.

FLETCHER, R & C. M. REEVES, **Function Minimization by Conjugate Gradientes**, Computer Journal, vol. 7, pp. 149-154, 1964.

FRANCO, H., COHEN, M., MORGAN, N., RUMELHART, D. & ABRASH, V., **Context-Dependent Connectionist Probability Estimation in a Hybrid**

Hidden Markov Model-Neural Net Speech Recognition System, Computer Speech and Language, vol 8, pp 211-232, 1994.

GENOUD, D., ELLIS, D. & MORGAN, N., **Simultaneous Speech and Speaker Recognition Using Hybrid Architecture**, International Computer Science Institute – ISI, July 1999.

GORIN, A. L. & MAMMONE, R. J. **Introduction to the Special Issue on Neural Networks for Speech Processing**. IEEE Transactions on Speech and Audio Processing, vol. 2, n° 1, 1994, pp 113-114.

HAGAN, M. T., H. B. DEMUTH & M. H. BEALE, **Neural Network Design**, Boston, MA: PWS Publishing, 1996.

HAYKIN. S., **Neural Networks - A Comprehensive Foundation.** Prentice Hall. 1994.

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

JAIN, A. K. & MAO, J. **Guest Editorial Special Issue on Artificial Neural Networks and Statistical Pattern Recongnition**. IEEE Transactions on Neural Networks, vol. 8, n° 1, 1997, pp 1-3.

MAKHOUL, J., **Linear Prediction: A Tutorial Review**, Proceedings of the IEEE. Vol. 63, nº 4, April 1975.

MAMMONE, R. J., ZHANG, X. & RAMACHANDRAN, R. P., **Robust Speaker Recognition**, IEEE Signal Processing Magazine, vol 13, nº 5, pp 58-71.

MARKEl, J. & GRAY, A. H. Jr., **Linear Prediction of Speech**, New York: Springer-Verlag, 1980.

MORGAN N. & BOURLARD H., **Neural Networks for Statistical Recognition of Continuous Speech**. Proceedings of the IEEE. v 83, no 5, 1995.

OPPENHEIM, A. V. & SCHAFER, R.W. **Discrete-Time Signal Processing**, Prentice-Hall, Inc, Englewood Clifs, New Jersey, 1989.

PARANAGUÁ, E. D. S**., Reconhecimento de Locutores Utilizando Modelos de Markov Escondidos Contínuo**, Tese de Mestrado, IME, 1997.

PEEBLES, P. Z. **Probability, Random Variables, and Random Signal Principles.** MacGraw-Hill, 3rd ed., 1993.

PICONE, J. **Continuous Speech Recognition Using Hidden Markov Models**, IEEE Acoustics, Speech and Signal Processing Magazine, Julho 1990, vol. 07, pp.26-41.

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

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

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

RABINER, L. R. **A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition**, Proc. IEEE, Fev. 1989, Vol. 77 (2):257-286.

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

RABINER, L. R.; SAMBUR, M. R. **An Algorithm for Determining the Endpoints of Isolated Utterances**, The Bell System Technical Journal, vol 54, nº 02, Fev 1975, pp 297-315.

RENALS, S., MORGAN N., BOURLARD H., COHEN, M. & FRANCO, H. **Connectionist Probability Estimators in HMM Speech Recognition**, IEEE Transactions on Speech and Audio Processing, January 1994, vol 2 nº 1, pp 161-174.

RENALS, S. & MORGAN N, **Connectionist Probability Estimators in HMM Speech Recognition**, International Computer Science Institute – ISI, December 1992.

RIEDMILLER, M. & H. BRAUN, **A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm**, Procedding of the IEEE International Conference on Neural Networks, 1993.

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

SANTOS, J. L. G., **Reconhecimento Automático da Voz Utilizando Sistemas Híbridos**, Tese de Mestrado, IME, 1999.

SANTOS, S. C. B. **Poder Discriminatório dos Fonemas Nasalados na Verificação Automática de Locutores**, Tese de Mestrado, IME, Dez. 1989.

SANTOS, S. C. B. , **Reconhecimento de Voz Contínua para o Português Utilizando Modelos de Markov Escondidos**, Tese de Doutorado, PUC, 1997.

SANTOS, S. C. B. & ALCAIM, A., **Sílabas como Unidades Fonéticas para o Reconhecimento Automático de Voz Contínua em Português**, SBA Controle & Automação, vol. 12, pp 64-70, 2001.

SCHAFER, R. W. & RABINER, L. R. **Digital Representations of Speech Signals,** Proceedings of IEEE, April 1975, vol. 63, nº 4, pp 662-667.

SILVA, D. G., **Comparação Entre os Modelos de Markov Escondidos Contínuos e as Redes Neurais Artificiais no Reconhecimento da Voz**. Projeto de Fim de Curso, IME, 1997.

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

STRUM, R. D. & KIRK D. E., **First Principles of Discrete Systems and Digital Signal Processing**, Addison-Wesley Publishing Company, Inc., 1988.

THOMÉ, A., G., SANTOS, S. C. B, DINIZ, S. S & SILVA, D. G, **Automatic Speech Recognition: A Comparative Evaluation between Neural Networks and Hidden Markov Models**, Relatório Técnico, Universidade Federal do Rio de Janeiro, NCE – 15/1999.

TOU, J.T. & GONZALEZ, R.C. **Pattern Recognition Principles**. Addison -Wesley Publishing Company, 1974.

VUUREN, S., **Speaker Verification in a Time-Feature Space**, Doctorate Thesis, Oregon Graduate Institute of Sciense and Technology, 1999.

WILPON, J. G. & RABINER, L. R. **A Modified K-Means Clustering Algorithm for Use in Isolated Word Recognition**, IEEE Transaction on Acoustics, Speech and Signal Processing, 1985, vol. ASSP-33, nº 3, pp. 587-594.

WOLF, J. J., **Efficient Acoustic Parameters for Speaker Recognition**, J. Acoust. Soc. Amer., vol 51, pt. 2, pp. 2044-2055, 1972.

YOUNG, S., **A Review of Large-Vocabulary Continuous-Speech Recognition**, IEEE Signal Processing Magazine, vol 13, nº 5, pp 45-57, 1996.

ZUE, Victor W., **The Use of Speech Knowledge in Automatic Speech Recognition**, Proceedings of the IEEE, vol.73, nº11, Novembro 1985.