Efeitos da Segmentação das Locuções de Treinamento em Modelos Híbridos ANN+HMM

José Antonio Moreira de Rezende & Carlos Alberto Ynoguti

***Resumo*— Este artigo apresenta uma investigação feita em torno da variação do desempenho de um sistema de reconhecimento de fala contínua baseado em modelos híbridos ANN+HMM, quando são deslocadas as marcas da segmentação fonética das locuções de treinamento. Com estes resultados seria possível determinar o erro máximo que um segmentador automático poderia cometer sem deteriorar de forma significativa o desempenho do sistema. Observou-se que erros de até aproximadamente 30ms nas marcas de segmentação não produziram aumentos notáveis na taxa de erros de palavra; valores de erro maiores do que este limiar produziram uma queda bastante acentuada no desempenho. Os testes foram conduzidos em um sistema de reconhecimento de fala contínua, dependente de locutor, operando sobre um vocabulário de aproximadamente 200 palavras.**

***Palavras-Chave*— Modelos híbridos ANN+HMM, reconhecimento de fala contínua, segmentação automática de fala.**

***Abstract*— This article presents an investigation about the performance of a hybrid ANN+HMM based continuous speech recognition system, when the phonetic segmentation marks of the training are moved from their right position. With these results, it would be possible to verify the maximum error allowed for an automatic segmenter to have. It was observed that segmentation errors less or equal 30 ms do not lead to noticeable errors, while greater differences dramatically drops the system's performance. The tests were performed on a continuous, speaker dependent speech recognition system, operating on a 200 word vocabulary.**

***Keywords*— Hybrid ANN+HMM models, continuous speech recognition, automatic speech segmentation.**

I. INTRODUÇÃO

Os modelos híbridos ANN+HMM foram propostos por Bourlard et al. [1] [2] [3] na década de 1990 como alternativa aos HMMs, tecnologia dominante na área de reconhecimento de fala até então. A motivação desta proposta era a de aproveitar o bom modelamento temporal fornecido pelos HMMs, e o alto poder discriminativo proporcionado pelas redes neurais para modelar as variações acústicas do sinal de fala, aproveitando assim os pontos fortes de cada uma destas técnicas.

Entretanto, tal abordagem apresenta uma séria desvantagem em relação aos sistemas baseados puramente em HMMs: enquanto os sistemas baseados em HMMs necessitam apenas de uma base de dados com a respectiva transcrição fonética, os sistemas baseados em modelos híbridos ANN+HMM necessitam de bases de dados de treinamento segmentadas.

A transcrição fonética de uma base de dados extensa pode ser realizada com certa precisão baseada apenas no texto que foi gravado e nas características da variante regional a que pertence cada locutor. Com isso, tem-se possíveis erros de transcrição de alguns fones, que podem ser compensados pela extensão do material gravado.

O processo de segmentação automática introduz uma nova fonte de variabilidades: além de conhecer a sequência de fones pronunciadas, faz-se necessário conhecer também a posição temporal de ocorrência de cada um deles. Obviamente este processo não pode ser realizado de forma manual para bases de dados muito extensas, e deve ser automatizado de alguma forma. Esta é uma das razões pelas quais muitas pesquisas vêm sendo feitas no sentido de produzir segmentadores automáticos cada vez mais precisos e confiáveis [5].

Este trabalho tem por objetivo elucidar algumas questões que surgem desta análise: em que proporção os erros de segmentação das locuções de treinamento influem no desempenho final do sistema? Qual é o erro de segmentação máximo "tolerável" para sistemas deste tipo?

Para alcançar estes objetivos foram feitos testes simulando vários níveis de erros de segmentação. Observou-se que erros de até aproximadamente 30ms nas marcas de segmentação não produziram aumentos notáveis na taxa de erros de palavra, sendo que para valores maiores do que este limiar produziram uma queda bastante acentuada no desempenho.

Os testes foram realizados sobre um sistema híbrido ANN+HMM de reconhecimento de fala contínua, dependente de locutor, e com um vocabulário de aproximadamente 200 palavras.

O artigo está organizado da seguinte forma: na seção seguinte será mostrado de forma resumida o algoritmo REMAP (*Recursive Estimation and Maximization of A Posteriori Probabilities*) [4] [6], para mostrar porque os modelos híbridos necessitam de bases de dados segmentadas. A seguir será apresentado o procedimento proposto para modificação das marcas de segmentação. Posteriormente serão descritos os testes realizados e seus respectivos resultados, juntamente com uma análise dos mesmos. Finalizando o trabalho, temos as conclusões e sugestões para trabalhos futuros.

II. REMAP

Devido ao efeito de coarticulação do aparelho fonador, as fronteiras entre os fones não são bem definidas e, portanto, seria interessante que o algoritmo de treinamento pudesse modelar tanto as regiões em que é possível afirmar a presença de um determinado fonema, assim como a presença de uma região de transição fonética. O algoritmo REMAP cumpre este objetivo através da maximização da probabilidade *a posteriori*

José Antonio Moreira de Rezende & Carlos Alberto Ynoguti, Instituto Nacional de Telecomunicações, Santa Rita do Sapucaí, Brasil, E-mails: joseamrz@inatel.br, ynoguti@inatel.br.

de um determinado modelo através da estimativa dos alvos da rede neural que são derivadas de suas próprias saídas [4]. Os alvos da rede são definidos por:

$$\gamma_n(k) = P(q_k^n|\mathbf{X}, M, \Theta) = \frac{P(q_k^n, \mathbf{X}|M, \Theta)}{p(\mathbf{X}|M, \Theta)} \quad (1)$$

Portanto,$\gamma_n(k)$ é uma estimativa da probabilidade *a posteriori* global, ou seja, a probabilidade do estado q_k ser visitado no instante n, dada a sequência de vetores acústicos $\mathbf{X}$, a cadeia de Markov M e o conjunto Θ de parâmetros fornecidos pela rede neural e pelo HMM. De (1), pode-se mostrar que $\gamma_n(k)$ resulta em:

$$\gamma_n(k) = \frac{\alpha_n(k)\beta_n(k)}{\sum_{l=1}^{L} \alpha_n(l)\beta_n(l)} \quad (2)$$

onde α e β são as probabilidades *forward* e *backward*, respectivamente e L é o número de estados do modelo. Pode-se mostrar também que, para α e β normalizados [8], $\gamma_n(k)$ fica:

$$\gamma_n(k) = \frac{\hat{\alpha}_n(k)\hat{\beta}_n(k)}{\sum_{l=1}^{L} \hat{\alpha}_n(l)\hat{\beta}_n(l)} \quad (3)$$

Para que as saídas da rede neural $y_i(n)$ tenham significado estatístico é necessário que a sua soma resulte na unidade. Neste caso aplicou-se uma normalização para garantir esta exigência:

$$y_i(n) = \frac{1}{1 + \exp[-x_i(n)]} \quad (4)$$

$$\bar{y}_i(n) = \frac{y_i(n)}{\sum_{i=1}^{K} y_i(n)} \quad (5)$$

onde K é o número de saídas da rede neural.

Pode-se interpretar $\bar{y}_i(n)$ como sendo a probabilidade *a posteriori* da classe (do fonema ou estado q_k), dada a entrada x_n. Assim:

$$\bar{y}_i(n) = P(q_k|x_n) \quad (6)$$

Usando a regra de Bayes:

$$p(x_n|q_k) = \frac{P(q_k|x_n)p(x_n)}{P(q_k)} \quad (7)$$

Com isto surge o termo $P(q_k)$ que é a probabilidade *a priori* das classes, que é dada por:

$$P(q_k) \approx \frac{1}{N_e} \sum_{n=1}^{N_e} P(q_k|x_n) \quad (8)$$

sendo N_e o número total de exemplos de treinamento. Esta aproximação, baseada no método de Monte Carlo, se justifica pela lei forte dos grandes números, assumindo que todas as observações x_n são realizações independentes da variável aleatória X_n, distribuída segundo $p(x_n)$.

É comum utilizar a verossimilhança de emissão de símbolos normalizada:

$$\frac{p(x_n|q_k)}{p(x_n)} = \frac{P(q_k|x_n)}{\frac{1}{N_e}\sum_{n=1}^{N_e} P(q_k|x_n)} \quad (9)$$

A. O algoritmo

Para o cálculo dos alvos da rede e reestimação dos pesos, probabilidade de classe e probabilidade de transição, o algoritmo segue os seguintes passos:

1) Inicializar a rede neural com exemplos centrados entre as marcas de segmentação. Estimar $P(q_k)$ conforme (8) e as probabilidades de permanência de estado da seguinte maneira:
$$a_{jj} = \frac{D_j - 10}{D_j} \quad (10)$$
sendo D_j a duração média (em milissegundos) do fonema associado a q_j e 10 é a sobreposição (também em milissegundos) entre as janelas.
2) Calcular os alvos suaves $\gamma_n(k)$ de cada locução, conforme (3). Após o cálculo de todos os alvos, reestimar as probabilidades de transição a_{jk}.
3) Treinar a rede com os alvos suaves calculados no item anterior. A cada quadro de análise x_n apresentado, a saída da rede neural é comparada com o conjunto de valores de alvos suaves correspondentes ao instante n analisado, com o intuito de atualizar as matrizes de pesos sinápticos, de acordo com o algoritmo *Error Back-Propagation* [9].
4) Reestimar $P(q_k)$, segundo (8).
5) Se o sistema não convergiu, voltar para o passo 2.

III. MODIFICANDO AS MARCAS DE SEGMENTAÇÃO

Seja uma locução de duração t_N composta de N fones, cada qual com duração $d_{f_n} = t_n - t_{n-1}$, conforme mostrado na figura abaixo:

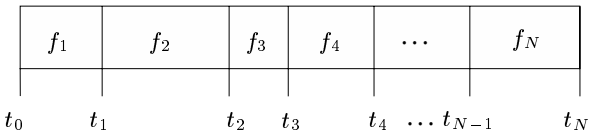

Fig. 1. *Locução contendo N fones cujas marcas de segmentação foram extraídas manualmente.*

O objetivo é simular erros de segmentação de forma controlada. Desta forma adotou-se o seguinte procedimento:

1) Inicialmente define-se o erro máximo que o segmentador poderá cometer (T milissegundos);
2) São gerados $(N-1)$ números aleatórios com distribuição uniforme no intervalo $[-T, +T]$;
3) Estes números correspondem aos deslocamentos que devem ser aplicados às marcas de segmentação manual das locuções de treinamento, simulando assim os erros de segmentação.

Foram realizados vários testes para diferentes valores de T com o intuito de verificar qual seria o erro máximo tolerado pelo sistema.

Este procedimento pode levar a algumas situações anômalas, como no exemplo a seguir: sejam $t'_1, t'_2, \ldots, t'_{N-1}$ as novas marcas de segmentação após a execução do procedimento acima, mostradas na Figura 2.

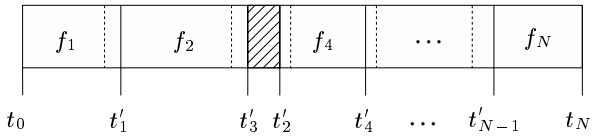

Fig. 2. Locução com as marcas de segmentação originais deslocadas.

Nota-se que houve uma sobreposição (área hachurada) dos fones f_2 e f_3, ou seja, o fone f_3 iniciou antes do término do fone f_2. Para esta situação, considera-se que a área seja um fonema (neste caso, f_3) com marcas inicial e final em t'_3 e t'_2, respectivamente.

Apesar de contornado o problema da sobreposição, corre-se o risco de que a duração arbitrada pela abordagem acima seja muito pequena ou nula. Com o intuito de evitar esta situação, foi adotado o seguinte procedimento:

1) foi feito um levantamento estatístico da média μ_n e do desvio padrão σ_n da duração de cada fone;
2) se a duração d_n arbitrada pelo algoritmo ao fone f_n for menor que o seu correspondente desvio padrão, a duração do mesmo foi alterada para $d'_n = \sigma_n$.

Na verdade, este segundo procedimento nada mais é do que o estabelecimento de um limiar mínimo de duração para cada fone. Com isso, as marcas de segmentação finais passam a ser aquelas mostradas na Figura 3:

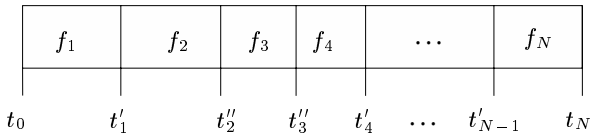

Fig. 3. Locução com as marcas de segmentação originais (em cima), e alteradas depois do primeiro e segundo procedimentos (embaixo).

IV. Sistema Implementado

A. Base de dados

Para este trabalho foi utilizada uma base de dados monolocutor, com 100 locuções, gravadas em ambiente de estúdio, por um locutor profissional [6]. A sentença mais curta é formada de apenas uma palavra, e a sentença mais longa, de 47 palavras. Todo o material foi gravado a partir de um texto escrito, de forma contínua, isto é, sem pausas entre as palavras. Estas 100 sentenças correspondem a um vocabulário de aproximadamente 200 palavras.

B. Parâmetros acústicos

Foram usados parâmetros mel-cepstrais de ordem 12 como vetores acústicos (o coeficiente C_0, correspondente à energia do quadro não foi considerado). Estes foram calculados a partir de trechos de 20ms do sinal de voz, com sobreposição de 50%. Antes da parametrização, o sinal foi submetido a um filtro de pré-ênfase $H(z) = 1 - 0,95z^{-1}$, e janelado através de uma janela de Hamming.

Para evitar a saturação dos neurônios da rede neural, foi feita uma normalização de amplitude destes coeficientes através da expressão:

$$X_n = \frac{X - \mu}{\sigma} \tag{11}$$

onde X é o vetor de parâmetros original, X_n é o vetor acústico normalizado, μ é o vetor média da locução, e σ é o desvio padrão de todas as componentes de todos os vetores acústicos.

Com isto garante-se que aproximadamente 95% destes coeficientes se concentram dentro do intervalo entre -1 e +1. O histograma levantado mostrou que os coeficientes assim normalizados seguem uma distribuição de média zero e desvio padrão 0,49.

C. Sub-unidades acústicas

Pelo tamanho do vocabulário optou-se por utilizar modelos de subunidades acústicas em lugar de modelos de palavras. Por causa do tamanho da base de dados, optou-se pelos fones independentes de contexto que, apesar de serem pouco discriminativos, são mais facilmente treináveis. Para o português brasileiro são identificados aproximadamente 40 fones [7], mas alguns destes foram unificados neste trabalho [11], e com isso chegou-se a 36 fones independentes de contexto.

D. Classificador

Como dito anteriormente, foi utilizado um sistema híbrido ANN+HMM como classificador. Neste, usa-se uma rede neural para a estimação da probabilidade de símbolo, e um HMM para o modelamento temporal. A seguir são apresentados os detalhes destes dois blocos:

1) Rede Neural: a rede implementada é do tipo *Multilayer Perceptron – MLP*, com apenas uma camada escondida, 36 neurônios na camada de saída (correspondentes aos 36 fones independentes de contexto), 100 neurônios na camada escondida e 108 entradas.

O número de entradas da rede neural foi definido com base no seguinte questionamento: quanto do sinal de entrada deve ser analisado de cada vez para uma boa definição de qual fone está sendo analisado?

Os HMMs trabalham com um quadro por vez, mas usam os parâmetros delta e delta-delta para fornecerem informação contextual ao classificador. O aumento do desempenho nestes casos indica que a análise de vários quadros ao invés de apenas um fornece informações preciosas ao sistema de reconhecimento.

O próximo passo é decidir quantos quadros devem ser considerados à esquerda e à direita do quadro sob análise. Para isto, foi feito inicialmente um levantamento das durações médias de cada fone para a base de dados utilizada. Este levantamento é mostrado na Tabela I:

TABELA I

LEVANTAMENTO ESTATÍSTICO DA DURAÇÃO DOS FONES.

Fone	Exemplo	# ocorrências	Duração média
a	c**a**sa	333	90 ms
ə	bol**a**	160	68 ms
ã	maç**ã**	95	87 ms
e	**e**levador	263	75 ms
ɛ	p**e**le	59	129 ms
ẽ	**en**saio	115	117 ms
i	**i**rmão	431	57 ms
ĩ	**ín**dio	89	105 ms
o	**o**velha	113	85 ms
ɔ	p**o**da	20	158 ms
õ	s**om**bra	79	102 ms
u	**lu**a	508	57 ms
ũ	m**un**do	22	99 ms
b	**b**ela	52	64 ms
d	**d**ente	180	50 ms
ʤ	**d**ia	60	56 ms
f	**f**aca	53	98 ms
g	**g**ueto	32	53 ms
ʒ	**j**ibóia	11	73 ms
k	**c**asa	187	84 ms
l	te**l**a	58	45 ms
ʎ	**lh**ama	6	58 ms
m	**m**esa	116	63 ms
n	**n**atal	120	45 ms
ŋ	ne**nh**um	2	92 ms
p	**p**ato	108	80 ms
ɾ	ca**r**a	207	36 ms
h	fo**rr**ó	19	65 ms
ɦ	ca**r**ta	51	58 ms
s	**s**oma	338	113 ms
t	**t**elha	225	77 ms
tʃ	**t**ia	80	96 ms
v	**v**elho	71	62 ms
ʃ	**ch**efe	19	104 ms
z	**z**ebra	155	62 ms

Fazendo uma média ponderada da duração dos fones, chegou-se a 74ms. Desta forma, uma janela de análise de aproximadamente 90 ms levaria em consideração um fone inteiro, mais um quadro à direita e outro à esquerda. Vários testes foram realizados variando-se o número de janelas, e realmente este intervalo foi o que apresentou o melhor desempenho. Considerando que a cada quadro tem-se um vetor mel-cepstral de ordem 12, o número de entradas da rede neural deve ser $12 \times 9 = 108$. Com isto, tem-se uma informação contextual de $d = 4$ quadros à direita e à esquerda.

A função de ativação escolhida foi a função logística, com as suas saídas normalizadas conforme (5). A rede implementada é mostrada de forma esquemática na Figura 4.

2) Algoritmo de Busca: foi utilizado o algoritmo *Level Building* com 50 níveis de busca (dado que a maior frase consta de 47 palavras, sem contar as pausas), com critério de parada automática proposto em [11], modelo de duração de palavras e gramática de pares de palavras, sendo esta uma simplificação da gramática *Bigram*.

V. RESULTADOS EXPERIMENTAIS

Inicialmente o sistema foi treinado usando a segmentação manual, assumida como sendo a correta, para estabelecer um desempenho padrão. Depois variou-se o limiar T descrito na Seção III, simulando vários níveis de erros de segmentação.

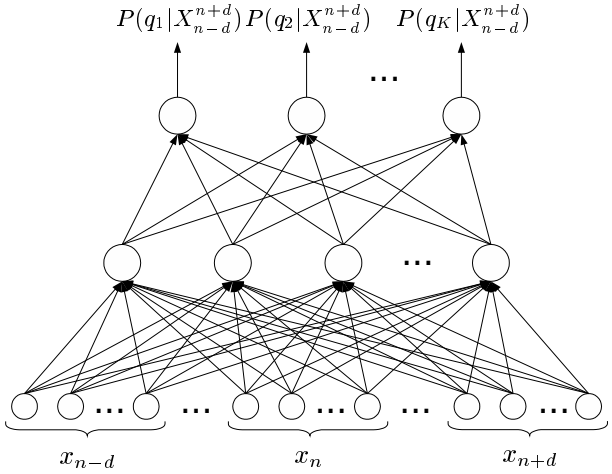

Fig. 4. *Rede neural com uso de informação contextual de d quadros à esquerda e à direita.*

O objetivo destes testes é verificar o desempenho do sistema em diferentes cenários de erros de segmentação.

Os resultados destes testes podem ser vistos nas Tabelas II (sem o uso da gramática) e III (com gramática). Nestas, *WER* é a taxa de erro de palavras (*Word Error Rate*).

TABELA II

DESEMPENHO DO SISTEMA HÍBRIDO ANN+HMM USANDO MODELO DE DURAÇÃO DE PALAVRAS.

Erros de Segmentação	S (%)	D (%)	I (%)	WER (%)
manual	9.9	4.2	11.8	25.9
10 ms	10.9	3.2	5.8	19.9
20 ms	11.3	4.9	11.6	22.8
30 ms	10.2	4.1	12.2	26.5
40 ms	18.8	4.9	28.3	52.0
50 ms	17.4	5.1	28.9	51.4
60 ms	21.5	5.5	25.8	52.8

TABELA III

DESEMPENHO DO SISTEMA HÍBRIDO ANN+HMM USANDO MODELO DE DURAÇÃO DE PALAVRAS E GRAMÁTICA PARES DE PALAVRAS.

Erros de Segmentação	S (%)	D (%)	I (%)	WER (%)
manual	3.3	2.3	2.7	8.3
10 ms	3.4	2.7	4.3	10.3
20 ms	3.8	2.1	3.8	9.7
30 ms	3.6	2.0	5.3	10.9
40 ms	4.7	2.6	4.4	11.7
50 ms	6.5	2.4	5.2	14.1
60 ms	8.0	1.9	5.8	15.7

A taxa de erros WER foi calculada a partir de

$$WER = \left(\frac{I + S + D}{W} \right) \times 100 \qquad (12)$$

onde I é o número de erros de inserção, S é o número de erros de substituição, D é o número de erros de deleção e W é o número de palavras presentes no conjunto de referência.

Para o cálculo de (12), foi utilizada a ferramenta SCLITE (*Score-Lite*), incluída no pacote NIST SCTK (*Speech Recognition Scoring Toolkit*) [10].

VI. Discussão

A partir da análise dos resultados apresentados nas Tabelas II e III, verifica-se que o desempenho do sistema permanece relativamente inalterado para erros de segmentação de até 30 milissegundos. Isto indica que os modelos híbridos ANN+HMM são bastante sensíveis à segmentação das locuções de treinamento.

Este fato é corroborado por outro resultado destes testes: na Tabela II, verifica-se que o sistema apresentou um desempenho melhor com erros de segmentação pequenos (10 ms e 20 ms) do que com a segmentação manual. Uma possível explicação para este fato é que as definições das novas marcas fizeram com que fossem "corrigidas" algumas imperfeições da segmentação manual.

VII. Conclusões e Trabalhos Futuros

Foi apresentado neste trabalho um estudo sobre o efeito da segmentação das locuções de treinamento em sistemas híbridos ANN+HMM de reconhecimento de fala contínua, operando em modo dependente de locutor sobre uma base de dados de aproximadamente 200 palavras.

Nos testes realizados observou-se que, em geral, o melhor desempenho foi obtido para os dados submetidos à segmentação manual, havendo uma piora à medida em que os erros de segmentação ficavam mais severos. Para erros de até 30 milissegundos (3 vetores acústicos), esta piora não é muito significativa, mas para erros de segmentação maiores, o desempenho cai acentuadamente. Estes resultados indicam que as redes neurais são razoavelmente sensíveis aos erros de segmentação das locuções de treinamento. Desta forma, se esta tarefa for confiada a um segmentador automático, deve-se garantir que o mesmo tenha este desempenho.

Um resultado aparentemente inesperado foi a melhora do desempenho com erros de 10 ms e 20 ms. Talvez isto tenha acontecido devido a eventuais erros na segmentação automática, que estes erros trataram de "corrigir". Entretanto, como a base de dados utilizada é bastante pequena, isto pode ter ocorrido devido a um mero acaso.

É importante ressaltar que os testes não foram realizados com segmentadores reais, mas sim com simulações de erros de segmentação. Desta forma, a continuação natural do trabalhos é a avaliação destes fatos com um segmentador automático real. A utilização de uma base de dados maior, independente de locutor também é interessante pois faz com que os resultados tenham uma significância estatística maior.

Agradecimentos

À Capes pelo financiamento parcial desta pesquisa.

Referências

[1] MORGAN, N. and BOURLARD, H., Continuous Speech Recognition: An Introduction to the Hybrid HMM/Connectionist Approach, IEEE Signal Processing Magazine, Invited Paper, vol. 12, no. 3, pp. 25–42, May 1995.

[2] BOURLARD, H. and WELLEKENS, C. J., Links Between Markov Models and Multilayer Perceptrons, IEEE Transactions on Patter Analysis and Machine Intelligence, vol. 12, no.12, pp. 1167–1178, December 1990.

[3] BOURLARD, H. and MORGAN, N. Connectionist Speech Recognition - A Hybrid Approach. Kluwer Academic Publishers, 1994.

[4] KONIG, Y., REMAP: Recursive Estimation and Maximization of A Posteriori Probabilities Application to Transition-Based Connectionist Speech Recognition, PhD Thesis, University of California at Berkeley, 1996.

[5] A. LJOLJE, J. HIRSCHBERG, and J. P.H. van Santen. Automatic speech segmentation for concatenative inventory selection. In J. P.H. van Santen, editor, Progress In Speech Synthesis, chapter 24, pages 304-311. Springer-Verlag New York, 1997.

[6] MORAIS, E. S., Reconhecimento Automático de Fala Contínua Empregando Modelos ANN+HMM, Tese de Mestrado, Universidade Estadual de Campinas, 1997.

[7] SILVA, T. C., Fonética e Fonologia do Português - roteiro de estudos e guia de exercícios. Editora Contexto. São Paulo, 2002.

[8] RABINER, L. R. and JUANG, B. H., Fundamentals of Speech Recognition, Prentice Hall Signal Processing Series, 1993.

[9] HAYKIN, S., Neural Networks: A Comprehensive Foundation, Prentice Hall, New Jersey, 1994.

[10] SCTK-1.3 - Speech Recognition Scoring Toolkit SCTK Version 1.3 (Includes the SCLITE Scoring program) ftp://jaguar.ncsl.nist.gov/pub/sctk-1.3.tgz (11/10/2003).

[11] YNOGUTI, C. A., Reconhecimento de Fala Contínua Usando Modelos Ocultos de Markov, Tese de Doutorado, Universidade Estadual de Campinas,1999.