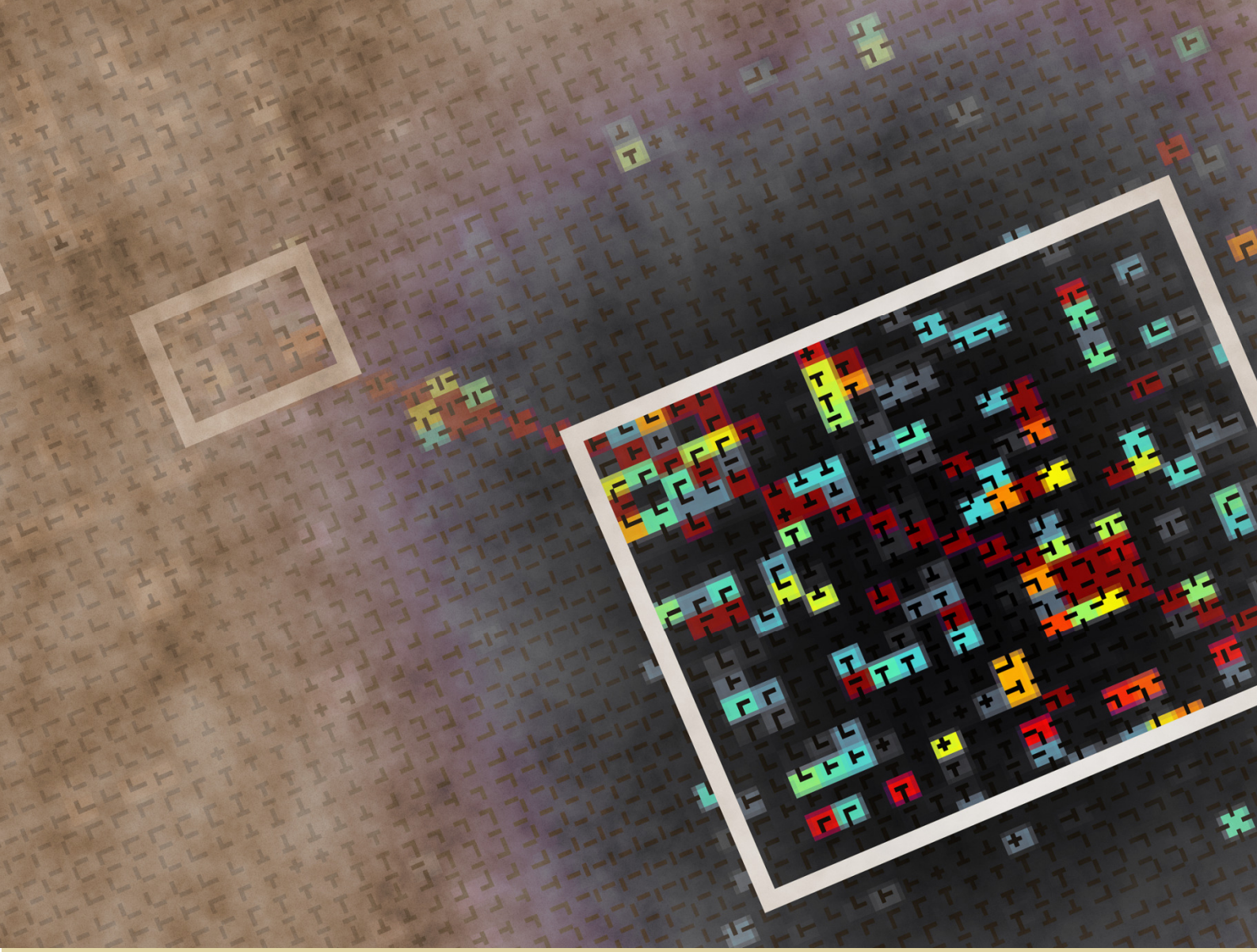


Carla Alexandra Calado Lopes

CLASSES FONÉTICAS ALARGADAS NO RECONHECIMENTO AUTOMÁTICO DE FONES

Dissertação de Doutoramento na área científica de Engenharia Eletrotécnica e Computadores, orientada pelo Senhor Doutor Fernando Manuel dos Perdigão, e apresentada ao Departamento de Engenharia Eletrotécnica e Computadores da Faculdade de Ciências e Tecnologia da Universidade de Coimbra.

Setembro 2011

UNIVERSIDADE DE COIMBRA
FACULDADE DE CIÊNCIAS E TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA E COMPUTADORES

CLASSES FONÉTICAS ALARGADAS NO RECONHECIMENTO AUTOMÁTICO DE FONES

Carla Alexandra Calado Lopes

Dissertação submetida à Universidade de Coimbra para obtenção do grau de Doutor em Engenharia Eletrotécnica e Computadores, especialidade de Telecomunicações e Eletrónica

COIMBRA
setembro de 2011

Tese realizada sob orientação do

Doutor Fernando Manuel dos Santos Perdigão

Professor Auxiliar do Departamento de Engenharia Eletrotécnica e Computadores
da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

AGRADECIMENTOS

Na meta desta etapa particularmente importante da minha vida, cabe-me agradecer a todos os que me acompanharam: aos que correram a meu lado, aos que me incentivaram, a quem me orientou, a que me deu suporte... nada na vida conquistamos sozinhos.

O meu primeiro agradecimento vai para o meu orientador Doutor Fernando Perdigão, pela competência científica com que me orientou e pelo tempo que generosamente me dedicou transmitindo-me os melhores e mais úteis ensinamentos, com paciência, lucidez, confiança e amizade. E, se me sinto hoje uma pessoa melhor formada a nível científico sinto-me também muito enriquecida a nível pessoal por ter tido o privilégio de privar com alguém com tão nobre modo de estar na vida.

Queria também agradecer a todos os colegas do laboratório de Processamento de Sinal do IT-Coimbra que me acompanharam durante este percurso. Sara, Arlindo, Cláudio, Zé David, Zé Nunes, Marco, Gabriel… obrigada por todos pela partilha de bons momentos, pelo bom ambiente do laboratório, por serem tão prestáveis mas principalmente pela vossa amizade.

Não posso deixar de me referir à minha família de trabalho - os meus colegas da ESTG-Leiria. Uma Grande família, ativa, positiva e especialmente amiga. Destaco as palavras de incentivo e de apoio dos colegas: Nuno Rodrigues, Telmo Fernandes, Sérgio Faria, Luís Távora e Catarina Silva. Uma palavra especial de agradecimento ao Pedro Assunção pelas suas críticas, correções, sugestões e pela forma como me ensinou com prazer e dedicação tanto do que sei.

Ao Pedro, meu porto de abrigo, agradeço por ser um companheiro, em tudo o que essa palavra melhor representa. A sua paciência e compreensão foram determinantes não só na chegada à meta mas ao longo destes anos (que já são 20!). Agradeço aos meus filhos Tiago e Camila, por terem uma mãe que os ama profundamente, mas que se vê sempre às voltas com seus trabalhos, condicionando muitas vezes o convívio familiar.

Aos meus pais, irmã, sogros e cunhados agradeço todas as oportunidades que me proporcionaram e que me permitem estar hoje aqui. Agradeço-lhes também pelo inestimável apoio familiar que preencheu as diversas falhas que fui tendo por força das circunstâncias.

Por último não posso deixar de agradecer às instituições que durante o decorrer deste trabalho me apoiaram incondicionalmente: o Instituto de Telecomunicações, a Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria e a Fundação para a Ciência e Tecnologia através da bolsa SFRH/BD/27966/2006.

Mais uma vez, a todos os meus sinceros agradecimentos.

RESUMO

Esta tese aborda o problema do reconhecimento automático de fones, a partir de um sinal de fala, usando o conceito de classes fonéticas alargadas. A definição automática de classes fonéticas alargadas com base na confusão entre fones é um assunto estudado neste trabalho. Contrariamente ao que é habitualmente feito, em que a definição de grupos de fones semelhantes é estabelecida por um especialista em fonética (recorrendo a conhecimentos de produção articulatória, de regras de fonologia ou de perceção auditiva), neste trabalho propõe-se um método de agrupamento de fones baseado nos resultados de um reconhecedor automático de fones. A informação de confusões entre fones é usada para definir uma métrica da distância entre eles e com ela estabelecer grupos de fones com alta probabilidade de confusão mútua.

Neste trabalho apresenta-se também um estudo comparativo envolvendo vários sistemas de deteção de classes fonéticas alargadas (HMM e sistemas híbridos SVM/HMM, SVM/NMD e ANN/HMM), que permitiu tirar conclusões sobre a eficácia das várias arquiteturas nesta tarefa. De forma a otimizar o desempenho do modelo híbrido entre redes neuronais artificiais (ANN) e modelos de Markov não observáveis (HMM), foi desenvolvido um método de treino global deste sistema.

Face à inexistência de um método de avaliação especificamente adequado à deteção de eventos, é proposto neste trabalho um novo método de avaliação com alinhamento temporal (AAT), o qual toma em consideração não só a sequência de etiquetas, mas também as respetivas marcas temporais.

No âmbito desta tese é ainda implementado um sistema global que integra reconhecimento de classes fonéticas alargadas e de fones, melhorando as taxas de acerto finais ao nível do fone. A integração consta da atribuição de um valor de confiança (através de um peso) à informação de classes fonéticas alargadas. O conjunto de pesos ótimo é encontrado por via de um processo iterativo desenvolvido com base no paradigma do treino discriminativo e que maximiza a taxa de precisão da sequência de fones no sistema de reconhecimento. As experiências realizadas com a base de dados TIMIT confirmam que um sistema de reconhecimento de fones beneficia de representações intermédias entre o sinal de fala e os fones.

ABSTRACT

This thesis addresses the problem of automatic phone recognition using the concept of broad phonetic classes.

A key issue addressed is the automatic definition of broad phonetic classes based on confusions among phones. Broad phonetic classes are usually defined by an expert in phonetics using articulatory information, rules of phonology or auditory perception. This work proposes a phone clustering method based on the output of a phone recognition system. Phone confusions are used to define a metric for phone distance that is used to establish clusters of phones with a high probability of mutual confusion.

This thesis also presents a comparative study involving several systems for the detection of broad phonetic classes (HMM and the hybrid systems SVM/HMM, SVM/NMD and ANN/HMM), making it possible to rank the performance of each system in this task. In order to optimize the hybrid system based on artificial neural network (ANN) and hidden Markov models (HMM), a global discriminative training method is proposed for this system. Because there is no well-established measure for evaluating event detection, a new evaluation method with time alignment is proposed in which both labels and their time boundaries are important.

A hierarchical classification structure is also implemented, integrating the classification of broad phonetic classes and phones, to improve phone recognition. This is done by weighting the broad phonetic classes' predictions. Optimal weights are obtained by an iterative discriminative training method. Experiments show improvements in phone recognition on the TIMIT database compared with a baseline system when intermediate representations between the speech signal and phones are used.

ÍNDICE

LISTA DE ACRÓNIMOS

AAT	Avaliação com Alinhamento Temporal
ANN	Artificial Neural Network
ASR	Automatic Speech Recognition
ASAT	Automatic Speech Attribute Transcription
CRF	Conditional Random Field
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
FER	*Frame* Error Rate
EER	Event Error Rate
HMM	Hidden Markov Model
HTK	Hidden Markov Model Toolkit
GMM	Gaussian Mixtures Model
LPC	Linear Predictive Coding
MAP	Maximum *a posteriori*
MCE	Minimum Classification Error
MEC	Mínima Entropia Cruzada
MFCC	Mel Frequency Cepstral Coefficients
ML	Maximum Likelihood
MLP	Multi-Layer Perceptron
MMI	Maximum Mutual Information
MPE	Minimum Phone Error
MTGT	Método de Treino Discriminativo Global
NMD	Non negative Matrix factor Deconvolution
NMF	Non negative Matrix Factorization
PER	Phone Error Rate
pdf	Probability Density Function
PLP	Perceptual Linear Prediction Coefficients
RBM	Restricted Bolzmann Machine
RNN	Recurrent Neural Network
SVM	Support Vector Machine
TDNN	Time Delay Neural Network
WSJ	Wall Street Journal

LISTA DE FIGURAS

LISTA DE TABELAS

CAPÍTULO 1

INTRODUÇÃO

Na era da informação, as aplicações informáticas tornaram-se parte integrante da vida moderna, e com elas cresceu a expectativa de facilidade de interação com elas. A fala, sendo o meio de comunicação por excelência, levou ao sucesso de diversas aplicações que incluem reconhecimento automático de fala (ASR - Automatic Speech Recognition): comando e controlo, ditação, sistemas de diálogo, sistemas de informação, tradução, etc. Mas o desafio atual vai além de usar a fala como um meio de controlo ou de acesso à informação. Pretende-se usar a fala como fonte de informação fazendo concorrência ao texto. São exemplos disso a anotação de conversas de fala para posterior tratamento, a ordenação e estabelecimento de prioridades em mensagens de voz, a pesquisa de assuntos em gravações de fala de reuniões ou em dados de programas de rádio, as transcrições de aulas e apresentações, etc. Estando a tecnologia de suporte a estas aplicações fortemente dependente do desempenho do reconhecimento automático de fala, a investigação nesta área mantém-se um tópico ativo. Prova disso é o extenso leque de direções de investigação apresentadas na *IEEE Signal Processing Magazine*, [8][9].

O reconhecimento automático de fala – a extração da informação contida no sinal de fala em termos de uma transcrição ortográfica, [57] – tem sido alvo de forte pesquisa por mais de quatro décadas, atingindo resultados notáveis. No entanto, só será de esperar que os avanços do reconhecimento de fala tornem a linguagem falada tão conveniente e acessível como o texto *online* quando os reconhecedores apresentarem taxas de erro muito próximas das dos humanos. E, se o reconhecimento de sequências de dígitos alcançou já uma taxa de acerto de 99,61%[1], [80] o mesmo não se pode dizer em relação ao reconhecimento de fones, cujas melhores taxas estão ainda abaixo dos 80%[2],[102], [143]. Sendo o fone a unidade de

[1] na base de dados TIDIGITS, [78]

[2] na base de dados TIMIT, [36]

reconhecimento usada nos reconhecedores de grandes vocabulários, não será de estranhar que tenha havido um interesse crescente por parte das equipas de investigação no sentido de desenvolver sistemas de reconhecimento de fones com desempenhos tão altos quanto possível. O presente trabalho insere-se neste domínio. Pretende dar um contributo na melhoria do desempenho da tarefa de reconhecimento automático de fones.

O desafio na construção de modelos acústicos de um sistema ASR passa pela aplicação de um bom algoritmo de treino a um conjunto adequado de dados. A base de dados define as unidades que podem ser usadas no treino e o sucesso do algoritmo está intimamente ligado à qualidade e ao detalhe da anotação dessas unidades. Muitas bases de dados estão insuficientemente etiquetadas e muito poucas contêm etiquetas usando a unidade mais pequena do som: o fone. No presente trabalho, os testes experimentais foram feitos recorrendo à base de dados TIMIT, [36]. Trata-se de uma base de dados integralmente anotada ao nível do fone e comummente usada pela comunidade científica. Em bases de dados onde não exista anotação manual ao nível do fone, é possível, recorrendo ao alinhamento forçado[3], gerar automaticamente a anotação ao nível do fone a partir de uma anotação a um nível superior (palavras ou frases). Neste caso, a anotação final fica claramente condicionada pela qualidade das marcas encontradas. Um outro fator que esteve na origem da escolha da base de dados TIMIT, é o facto de ser uma base de dados de referência sobre a qual existem inúmeros trabalhos publicados, permitindo assim, a comparação de resultados e a avaliação da qualidade de propostas alternativas.

1.1. FONEMAS

A fala humana pode ser segmentada em unidades acústicas fundamentais, os fones, mas que se podem associar a classes mais abstratas, os fonemas. Diferentes concatenações de fonemas expressam mensagens diferentes. Um sinal de fala pode ser visto como sendo composto por vários segmentos que correspondem à realização dos fonemas – os fones – e cuja presença e ordem, determinam a mensagem.

Os fonemas são as unidades básicas de uma Língua. São os sons que são distinguíveis pelo seu contraste dentro de uma palavra. Estes sons são unidades abstratas que formam a base de uma língua falada. Estas unidades têm a propriedade de mudar uma palavra quando uma unidade é substituída por outra.

[3] Partindo da informação da sequência de etiquetas, o sistema ASR gera o alinhamento dessas mesmas etiquetas

Diferentes línguas possuem diferentes fonemas, em qualidade e em número, uma vez que têm características fonéticas distintas. No entanto, todas as línguas são descritas por um conjunto relativamente reduzido de fonemas. A partir deste conjunto, e à custa de concatenações de fonemas, conseguem-se construir dicionários de milhares de palavras. Daí que na construção de sistemas de ASR de grandes vocabulários, a unidade de reconhecimento seja o fonema (ou a sua realização acústica, o fone), uma vez que a dimensão do vocabulário torna impraticável o uso de modelos de palavras completas ou mesmo o uso de modelos de sílabas. Na Figura 1. 1 apresenta-se uma estrutura de representação de um segmento de fala da TIMIT ao nível da frase, da palavra e do fone.

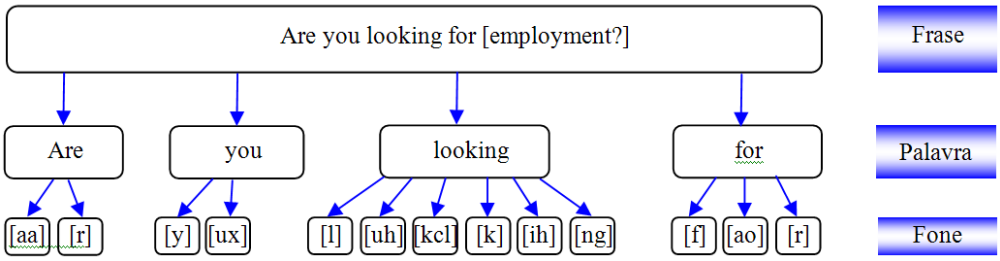


Figura 1. 1: Representação da estrutura hierárquica de uma frase da base de dados TIMIT seguindo o alfabeto fonético TIMITBET, [59].

O universo de aplicações em que o fonema é usado como unidade base de reconhecimento é bastante vasto. Além dos sistemas típicos de reconhecimento de fala de grandes vocabulários, [104][132][136], existem aplicações relacionadas com deteção de palavras-chave, [136], identificação de língua, [101][136], identificação de locutor, [34], bem como aplicações relacionadas com a identificação e transcrição de partes vocais de trechos de música, [33][39].

Enquanto um sistema de reconhecimento automático de fala procura encontrar a sequência de palavras correspondente à locução de entrada, um sistema de reconhecimento automático de fonemas visa identificar a sequência de fones presente num dado sinal de fala e atribuir a cada um dos segmentos o fonema correspondente. O desempenho global do reconhecedor depende da atribuição correta destas classes, uma vez que um fone mal classificado pode para dar origem ao reconhecimento de um fonema e palavra distinta. Veja-se o exemplo das palavras fala (/falɐ/)[4] e falha (/faʎɐ/) onde os fonemas /l/ e /ʎ/ ocorrem em par mínimo (um só fonema distingue as duas palavras).

[4] Escrita segundo o alfabeto fonético internacional - IPA

Num sistema automático de reconhecimento de fonemas baseado em modelos estatísticos, a atribuição de classes (decisão sobre qual o fonema reconhecido) depende essencialmente de dois modelos: o modelo de linguagem e o modelo acústico. O modelo de linguagem especifica as probabilidades de ocorrência de sequências de fonemas. O modelo acústico representa as estatísticas do sinal de fala para as realizações acústicas de cada fonema (fones). Estes modelos têm uma dupla abrangência: a nível acústico representam fones; mas representam fonemas a um nível mais abstrato. São estimados a partir de um conjunto vasto de dados de fala transcrita (base de dados) usando algoritmos adequados de treino.

Os modelos de Markov não observáveis (HMM) são a técnica mais usada nos sistemas de reconhecimento de fala. Depois de várias décadas de intensa investigação, tudo aponta para que o desempenho dos sistemas ASR baseados em HMMs tenha atingido um limite estável. No final da década de oitenta e início da de noventa (re) surgiram, em alternativa aos HMMs, as redes neuronais artificiais (ANN - Artificial Neural Network). Métodos híbridos HMM/ANN emergiram, com resultados comparáveis e por vezes superiores aos alcançados pelos HMM. Por outro lado, durante a última década, surgiram duas novas técnicas no campo da aprendizagem automática ("*machine learning*") com bons desempenhos em tarefas de classificação: as Máquinas de Vetores de Suporte (SVM - Support Vector Machines) e mais recentemente os Conditional Random Fields (CRFs).

Contudo, os melhores resultados na TIMIT são obtidos usando modelos híbridos: ANN/HMM, [127][132][143], e CRF/HMM, [104] onde os modelos acústicos são baseados em ANN no primeiro caso e em CRF no segundo, e onde a modelação temporal é feita com HMMs. Em qualquer uma das técnicas de reconhecimento referidas, os parâmetros dos modelos são aprendidos diretamente a partir dos dados, usando modelos simples da fala para mapear as observações acústicas em fones e palavras. No entanto, o seu desempenho é significativamente inferior ao desempenho humano [28][84]. Uma das razões apontadas para esta diferença é a abordagem "*top-down*" seguida no reconhecimento, onde apenas restrições sintáticas e linguísticas são impostas às sequências possíveis de palavras a reconhecer. Estudos perceptuais apontam que o reconhecimento é, de facto, influenciado por estas restrições em níveis neuronais superiores mas que existe uma interação entre vários níveis de representação, nomeadamente acústica, fonémica, fonológica, sintática e semântica [84].

Trabalhos recentes dão ênfase a uma abordagem "*bottom-up*" onde eventos acústico-fonéticos são primeiramente detetados [30][49][54][71][77][81][85][107][117] sendo posteriormente integrados no processo de descodificação. A ideia subjacente a esta

abordagem é que é possível identificar a partir do sinal de fala certos eventos ou marcos temporais que evidenciam a presença de entidades fonéticas específicas. É o caso da deteção de "*bursts*" de consoantes plosivas [49][108], "*voice-on-time*", [6] , "*vowel-onset-point*" [54][117], fecho glotal ("*glottal-closures*") [117], nasalação, fricação, etc., [6][54][81], ou simplesmente a deteção de eventos estruturais [30][85].

O presente trabalho enquadra-se nesta abordagem "bottom-up" visando uma deteção robusta de eventos acústico-fonéticos no sinal de fala. O objetivo é fornecer informação detalhada a nível acústico e fonético a um sistema de reconhecimento de fala de forma a melhorar o seu desempenho. O reconhecimento fonético é feito por via de um sistema híbrido ANN/HMM enquanto na deteção de eventos foram testadas várias abordagens híbridas ANN/HMM, [96] SVM/HMM, [90] e SVM/NMD (Non Negative Matrix Deconvolution), [94].

Uma ressalva final deve ser feita quanto à terminologia adotada neste trabalho. Uma vez que a base de dados usada está anotada finamente ao nível do fone e, sendo a saída do reconhecedor uma sequência de etiquetas definidas pela base de dados, esta corresponde a uma descrição fonética do sinal acústico de entrada e não a uma descrição fonológica. Assim, uma vez que não se passa a um nível de abstração superior (ao nível da palavra), estamos perante um reconhecedor de fones e não de fonemas. Por este motivo adotámos o termo "reconhecimento de fones" em todo este trabalho.

1.2. EVENTOS LINGUÍSTICOS

Desde há centenas de anos que filósofos e psicólogos defendem que a perceção humana é um processo que nos permite produzir uma representação mental do que nos rodeia a partir do extenso leque de informação que nos chega através dos sentidos. Isto significa que possuímos um mecanismo de formar essas representações e depois usá-las no planeamento de ações. Por outras palavras, o ser humano tem a capacidade de conceptualmente passar a informação sensorial de entrada para uma representação descrita por essa entrada, [157]. Infelizmente este processo não é simples de modelar e para passar da entrada sensorial para a representação perceptual é necessário descobrir como se descodifica o sinal da fala e se lhe atribuí um significado. Assim, o problema é precisamente como extrair eficientemente informação útil da entrada. A representação global é constituída por pequenas representações. No domínio visual, estas podem corresponder a objetos físicos; na audição podem ser sons, características acústicas, características fonéticas, marcos temporais, etc. A

questão principal é a de saber qual o tipo de segmentação que fazemos na aquisição sensorial auditiva. Será uma representação sequencial ou não sequencial? Extrairá em tempo real todos os traços distintivos[5] ao nível da produção da fala? É sabido que os traços distintivos constituem a unidade mais pequena com significado fonológico. No entanto, serão essas as características que extraímos a partir do sinal acústico? De facto, as características distintivas fornecem-nos informação de extrema importância sobre a dinâmica do sinal de fala. O vozeamento, que descreve a vibração glotal durante a articulação, permite muitas vezes a distinção entre vogais e consoantes. Outros parâmetros como o modo de articulação e o local de articulação fornecem também informação útil. O modo de articulação caracteriza, em geral, o tipo de fecho articulatório e o grau de obstrução do fluxo de ar. Informação da posição da língua durante o vozeamento, do arredondamento dos lábios, da tensão vocálica e a dinâmica espetral contribuem efetivamente para a distinção entre fones. Apesar dos traços distintivos se basearem em pares mínimos de fones, estão diretamente relacionados com movimentos articulatórios. Estando estes movimentos claramente correlacionados com o contexto fonético, os traços distintivos tornam-se difíceis ou mesmo impossíveis de determinar nos sinais de fala, especialmente em fala espontânea. Há mesmo casos em que determinadas combinações de características não se verificam de todo. Por outro lado, e apesar do seu significado fonológico, os traços distintivos não incluem qualquer noção de temporalidade - são binários: ou estão presentes ou estão ausentes. No entanto, a modulação temporal é claramente um requisito em questões de reconhecimento automático de fala.

Apesar da natureza contínua da fala, esta é vista sempre como uma sequência de unidades discretas, os fones. Como a fala resulta de alterações quer da fonte de excitação quer do trato vocal, pode ser descrita como uma sequência de eventos. Estes eventos podem estar relacionados com a acústica do sinal, com a produção da fala, com o locutor, etc., uma vez que qualquer alteração pode, por si só, representar ou ser tratada como um evento. É ainda interessante realçar que muitos destes eventos são comuns à maioria das línguas o que permite a sua utilização em aplicações multilingue.

Na literatura, os sistemas baseados em eventos, surgem em vários contextos, nomeadamente:

[5] Os traços distintivos são propriedades mínimas, de caráter acústico e articulatório, como nasalidade, sonoridade, labialidade, coronalidade, etc. que, de forma coocorrente caracterizam os sons da língua.

- na segmentação do sinal de entrada em classes fonéticas alargadas de acordo com a presença ou ausência de características específicas na estrutura acústica do sinal, [54][56][81];
- na deteção de marcos temporais onde ocorrem alterações como quedas de sílabas, fecho da glote ou ponto de início de vogal, [46][117];
- na identificação de eventos estruturais como limites de locuções, pausas, marcas de discurso e disfluências, [85];
- na deteção de limites de palavras e de atividade vocal, [30];
- aplicados ao reconhecimento de locutor, [133];
- na identificação de gestos, [42];
- na representação de eventos (cenas) auditórios, [51], etc..

Apesar deste conceito algo difuso de "eventos de fala" todos os sistemas de reconhecimento de eventos têm o mesmo objetivo: detetar a ocorrência de elementos importantes no sinal de fala assim como os instantes ou períodos em que ocorrem.
Seguindo a ideia defendida por Allen, [7], que o sistema de reconhecimento de fala humano usa informação parcial, propõe-se neste trabalho combinar informação de eventos com informação fonética de forma a contribuir para uma melhoria do desempenho dos sistemas de ASR.

1.3. CONTRIBUIÇÕES DESTE TRABALHO

No presente trabalho investiga-se a combinação de diferentes níveis de detalhe fonético de forma a melhorar a taxa de reconhecimento de fones. O estudo centrou-se primeiramente numa deteção robusta de classes fonéticas alargadas (eventos). Foram estudados e comparados vários sistemas de deteção de eventos, todos eles baseados em sistemas híbridos.
O estudo de um sistema de deteção de eventos pressupõe a avaliação das suas capacidades no reconhecimento da sequência de eventos detetada, bem como os instantes em que esses eventos ocorrem. Como não foi até à data estabelecida nenhuma medida de avaliação adequada a esta tarefa, foi proposta em [91], no âmbito deste trabalho, uma medida de avaliação de desempenho adequada a sistemas de reconhecimento de eventos.
O primeiro sistema de deteção eventos a ser implementado e testado foi um híbrido HMM/SVM, [90] que mostrou ter um desempenho superior a sistemas individuais HMM e

SVM. Atendendo ao facto de a aplicação de SVMs à deteção de classes alargadas ter sido bem-sucedida, foi estudada uma forma alternativa aos HMMs de fazer a modelação temporal. A escolha recaiu no método de factorização matricial *Non-negative Matrix Deconvolution* (NMD). O algoritmo procura padrões num conjunto de dados e tem um desempenho notável quando os padrões que se repetem (bases) são objetos estáticos. Contudo, esse não é o caso dos sinais de fala e, no âmbito deste trabalho, propusemo-nos investigar uma extensão do algoritmo que incluísse variabilidade dos dados, nomeadamente o deslocamento em frequência e compressão/expansão temporal dos sinais de fala [92]. O desempenho do método foi superior ao desempenho de um sistema HMM, mas a modulação temporal imposta pelo NMD ficou aquém da alcançada pela dos HMMs, [94][96].

Em [96] é feito um estudo que avalia e compara o desempenho de sistemas híbridos de deteção de eventos usando HMMs, SVMs e ANNs onde se enfatizam as abordagens que conduzem a taxas de erro inferiores. À semelhança do que acontece na tarefa de reconhecimento de fones na TIMIT, também no reconhecimento de eventos a combinação ANN/HMM obteve o melhor desempenho.

Os sistemas híbridos têm uma estrutura bietápica. Constam de uma classificação em fones, *frame* a *frame*, seguida por um sistema de alinhamento usando HMMs, cujo treino é feito em separado. A simplicidade destes sistemas é contrabalançada pela ausência de um esquema de treino do sistema completo. Assim, uma das contribuições deste trabalho passa pelo desenvolvimento de um método de treino discriminativo de sistemas híbridos MLP/HMM, [95]. Um MLP ("Multi-Layer Perceptron", rede ANN "feed-forward") é uma estrutura já por si discriminativa, no entanto os pesos da rede são atualizados de acordo com as classes alvo (*targets*[6]) apresentados na camada de saída, quando o melhor seria serem atualizados de acordo com a melhor sequência de estados dos HMMs. De forma a ultrapassar este problema propõe-se um método de treino baseado numa função de custo que minimiza o erro de classificação do sistema híbrido global, [87].

Pretende-se neste trabalho fornecer informação detalhada a nível acústico e fonético a um sistema de reconhecimento de fones de forma a melhorar o seu desempenho. Neste paradigma é proposta uma estrutura hierárquica, que parte de informação fonética grossa (classes alargadas) e atinge um detalhe fino (ao nível do fone) na última camada, [88]. As probabilidades de pertença a uma determinada classe alargada são pesadas e combinadas

[6] A rede utiliza o par (entrada, *target*) para aprender. O *target* é a resposta esperada da rede a cada um dos diferentes vetores de entrada. É um vetor de tamanho igual ao número de classes a classificar tomando o valor 1 na célula correspondente à classe a que pertence a *frame* de entrada e 0 nas restantes.

de forma a obter uma predição mais robusta ao nível do fone. Depois de estudados os melhores parâmetros a usar no treino do MLP, [93], bem como o contexto temporal ideal, desenvolveu-se um método de treino discriminativo para obtenção do melhor conjunto de pesos, [89].

As classes alargadas são, normalmente, formuladas por um especialista em fonética ou fonologia, o que pode imprimir alguma subjetividade na sua definição. No seguimento deste facto propõe-se, em alternativa, um método automático de divisão dos fones em classes alargadas. O método proposto parte de uma matriz de confusão, com origem num reconhecedor de fones, para encontrar grupos de fones que se confundem facilmente. O agrupamento é feito com base numa medida de distância entre fones, que contrariamente ao comum, cumpre os requisitos de uma métrica. O método proposto, [89] [97], para a formação de classes alargadas provou ser uma alternativa atrativa ao conhecimento de especialistas, alcançando resultados semelhantes ao nível da *frame* e superiorizando-se na contribuição para a melhoria de desempenho do sistema de reconhecimento de fones.

CAPÍTULO 2

RECONHECIMENTO AUTOMÁTICO DE FALA:

PARÂMETROS, CLASSIFICADORES E MEDIDAS DE DESEMPENHO

Este capítulo fornece uma introdução à parametrização do sinal de fala e aos algoritmos de classificação que formam a base do trabalho apresentado nesta tese.

Serão descritas as duas representações de parâmetros usadas: os bem conhecidos parâmetros MFCC (*Mel-Frequency Cepstrum Coefficients*) e um segundo conjunto de parâmetros que mostrou ser útil em tarefas quer de reconhecimento de fones, [93] quer de deteção de eventos, [94]. Os parâmetros MFCC descrevem de uma forma eficaz o sinal de fala e são suficientes na maioria das tarefas de classificação. No entanto, conforme indicado em vários trabalhos, nomeadamente [82], há situações onde parâmetros temporais são mais discriminativos, o que serviu de motivação para explorar o contributo que outro tipo de parâmetros temporais e espetrais poderiam dar na deteção de classes de eventos.

Neste capítulo revê-se também alguns dos principais métodos de reconhecimento de padrões com aplicação na deteção de eventos, nomeadamente: modelos de Markov não observáveis, redes neuronais, máquinas de vetores de suporte, métodos híbridos ANN/HMM, SVM/HMM e uma proposta que conjuga SVMs com uma técnica de factorização de matrizes de valores não negativos.

A avaliação do desempenho dos classificadores será também focada neste capítulo. Além de apresentadas as medidas clássicas de avaliação de desempenho, será proposta uma nova medida de avaliação de desempenho adequada à tarefa de deteção de eventos.

2.1. EXTRAÇÃO DE CARACTERÍSTICAS

A parametrização do sinal de fala é uma das componentes principais do processo de reconhecimento de fala, tendo a responsabilidade de fornecer ao sistema um conjunto de parâmetros que caracterizem eficientemente o sinal de entrada. O conjunto de parâmetros usado deve, por um lado, ser capaz de reter informação importante que permita a discriminação entre eventos distintos e, por outro lado, deve ignorar informação irrelevante, como é o caso da prosódia, do locutor, do ruído de fundo ou de canal, etc.
Seguidamente abordam-se os parâmetros usados no presente trabalho.

2.1.1. PARÂMETROS MFCC

O tipo de parametrização mais utilizado em reconhecimento automático da fala consiste no *cepstrum* derivado de uma análise espetral de bancos de filtros. São conhecidos como parâmetros MFCC.
Depois de dividido o sinal em *frames* por aplicação de uma janela de *Hamming*, é calculada a sua DFT (*Discrete Fourier Transform*). Seguidamente, obtém-se uma representação suavizada do espetro por aplicação de um banco de filtros. Consiste em passar o espetro de potência de cada *frame* do sinal de fala por um banco de filtros, cujas sub-bandas são relevantes no domínio da frequência. Segundo a modelação MFCC, os filtros têm uma forma triangular e as suas frequências centrais e larguras de banda são projetadas de forma a cobrir toda a gama frequências de interesse. No presente trabalho esta gama situa-se entre os 150Hz e os 7500Hz. As frequências centrais dos filtros estão de acordo com uma escala definida em unidades *Mel: Mel*$(f) = 2595 \times \log_{10}\left(1 + f/700\right)$.

Esta escala modela um aspeto importante do sistema auditório humano que é a relação não linear entre a perceção subjetiva de tom (musical) e a frequência. Esta relação é aproximadamente linear até 700Hz e aproximadamente logarítmica acima deste valor. Outro aspeto modulado neste banco de filtros é que as suas larguras de banda são crescentes com a frequência de acordo com o conceito bandas críticas, e de acordo com a diminuição da resolução de frequência do ouvido humano. As larguras de banda do banco de filtros são constantes na escala *Mel* e é nesta escala que a forma dos filtros é triangular. A frequência superior de um filtro triangular coincide com a frequência central do filtro seguinte. A energia do sinal à saída do banco de filtros corresponde a uma descrição espetral suavizada do sinal, com um número de parâmetros igual ao número de filtros.

O estágio seguinte consiste em obter uma representação *cepstral* das energias à saída do banco de filtros. Na representação MFCC é usada a Transformada Discreta do Cosseno (DCT - *Discrete Cosine Transform*) através da expressão: $c_k = \sqrt{\frac{2}{N}} \sum_{j=1}^{N} \log(m_j) \times \cos\left(k \frac{\pi}{2N} (2j-1) \right)$, $k = 1,...,N-1$, onde c_k representa os coeficientes da DCT, N o número de filtros do banco de filtros (no presente trabalho 32) e os coeficientes m_j a energia do sinal à saída do banco de filtros.

Os coeficientes de maior ordem do cepstrum são numericamente muito pequenos (o cepstrum c_k, decai à razão de pelo menos $1/k$) o que resulta numa grande variância entre os coeficientes de maior e menor ordem. Para uma taxa de amostragem de 16kHz, geralmente os 12 primeiros coeficientes são suficientes para capturar a maioria da informação pertinente necessária para o reconhecimento de fala.

A energia contida num segmento de fala é outro parâmetro que se mostra útil na discriminação de características fonéticas da locução. Por exemplo, as vogais têm uma energia muito mais alta que as fricativas e plosivas. Daí que se introduza também a energia no vetor de características, mais concretamente o logaritmo da energia, $E = \log\left(\sum_{n=0}^{N-1} s^2[n] \right)$, onde $s[n]$ representa uma *frame* de sinal de N amostras. Por vezes é ainda adicionado o coeficiente c_0 (ou em substituição da energia), dado por $c_0 = \frac{1}{\sqrt{N}} \sum_{j=1}^{N} \log(m_j)$.

Simultaneamente, as dependências entre *frames* sucessivas podem ser capturadas aumentando estes 13 componentes (12 MFCC + *E*) com os seus coeficientes dinâmicos (*delta* e *delta-delta*). Estes parâmetros correspondem a parâmetros de regressão linear de uma sequência de vetores cepstrais e acrescentam, de alguma forma, contexto ou informação dinâmica. Estes coeficientes são calculados pela equação do declive da reta de regressão linear $\Delta c_k[t] = \frac{\sum_{j=-n}^{n} j \times c_k[j+t]}{\sum_{j=-n}^{n} j^2}$, relativos às sequências $c_k[t]$, onde t representa o índice temporal da *frame*. No presente trabalho usou-se n=2, ou seja foram consideradas 5 *frames* para calcular a variação dinâmica espetral.

A concatenação de todos estes parâmetros resulta num vetor de dimensão 39. Este vetor de características foi usado em grande parte das experiências realizadas neste trabalho.

2.1.2. PARÂMETROS ESPECIALIZADOS

A secção anterior abordou a representação de tempo curto da envolvente espetral de um sinal de fala mais usada em sistemas ASR. No entanto, existem outros parâmetros que indicam também características importantes do sinal. No presente trabalho, e em algumas experiências em que é necessário um compromisso entre a quantidade de material usado e o número de parâmetros (ex: o treino de SVMs implica um grande número de vetores de características simultaneamente em memória, o que facilmente esgota os recursos computacionais disponíveis), foram considerados outros parâmetros. Uma breve descrição destes parâmetros é feita de seguida.

Centróide Espetral

O centróide espetral é uma medida que indica onde está concentrada a energia de um segmento de fala, dando neste caso indicação do *Centro de Gravidade* espetral:

$$Cent_t = \frac{\sum_{k=1}^{N} kY_t[k]}{\sum_{k=1}^{N} Y_t[k]} \quad \textbf{(2.1)}$$

Nesta expressão os valores $Y_t[k]$ são as energias à saída de um banco de filtros em escala *Mel* para a *frame t* e *N* o número de filtros. Um exemplo é mostrado na Figura 2.1.

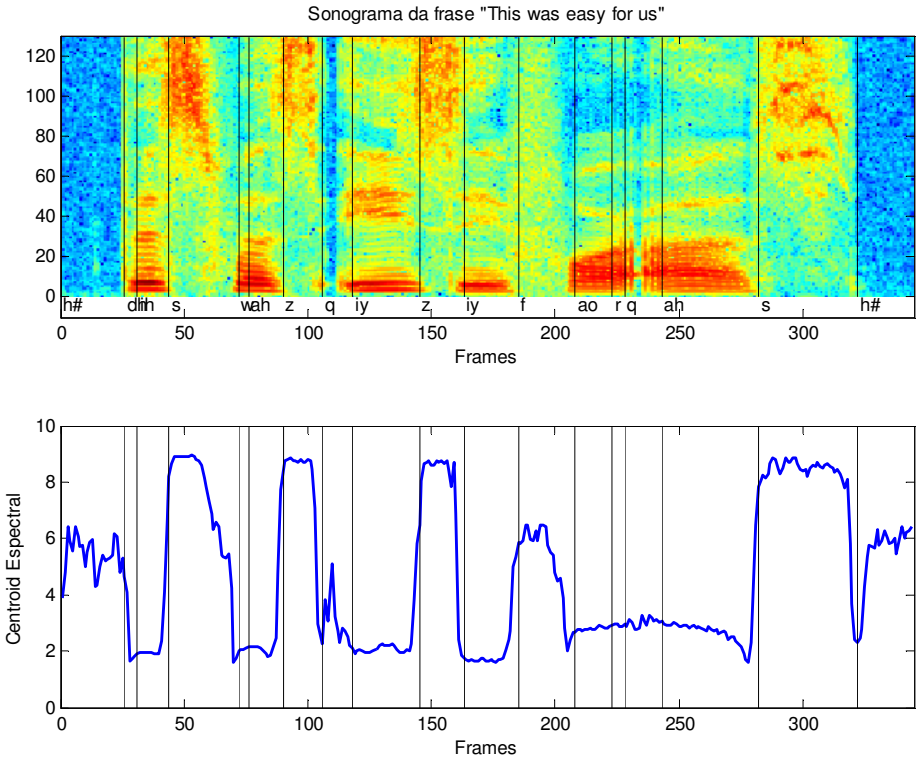


Figura 2.1: Exemplo da medida dos valores do centróide espetral na frase “This was easy for us”.

Pela figura verifica-se que fricativas como o [s] ou o [z] cuja energia está concentrada nas altas-frequências concentram o pico espetral nos filtros de ordem mais elevada enquanto as vogais apresentam valores de centróide espetral nos filtros de baixa ordem.

Espalhamento Espetral (SFM)

Uma forma de medir o espalhamento espetral de um segmento de sinal consiste em avaliar a medida conhecida como medida de planura espectral (SFM - *Spectral Flatness Measure*). A SFM é dada pela razão entre a média geométrica (μ_g) e a média aritmética (μ_a) do espetro do sinal. A média geométrica é sempre inferior ou igual à média aritmética. Estas duas médias apenas são iguais no caso de se tratar de um espetro plano (ruído branco), tomando a sua razão o valor de 1. O espetro é caracterizado pelo módulo da DFT do segmento, $X(k)$, e a SFM, em dB, é dada pela expressão

$$SFM_{dB} = 10\log\frac{\mu_g}{\mu_a} = 10\log\frac{\sqrt[N]{\prod_k |X(k)|}}{\frac{1}{N}\sum_k |X(k)|}. \qquad \textbf{(2.2)}$$

Quando o objetivo é averiguar a proximidade de um som a um tom ou a ruído, a SFM surge com a denominação de *Coeficiente de Tonalidade*.

Na Figura 2.2 apresenta-se um exemplo da informação fornecida pela SFM.

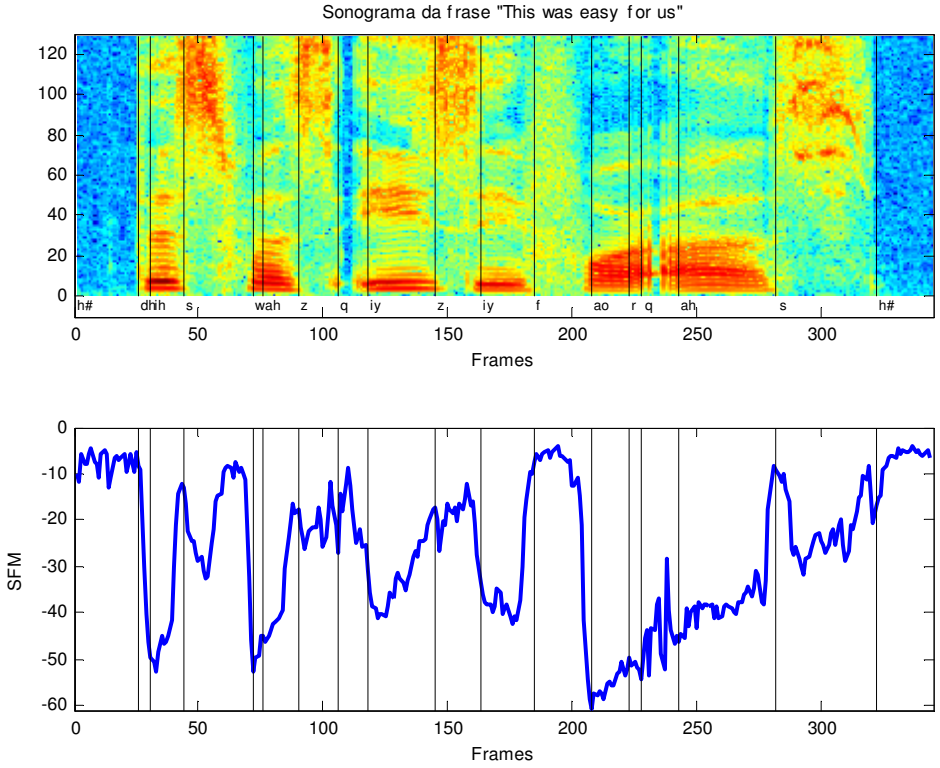


Figura 2.2: Exemplo da medida *Spectral Flatness Measure* na frase "This was easy for us".

Fricativas como o [f] cuja energia se espalha praticamente de forma uniforme na frequência têm valores de SFM em dB quase nulos enquanto vogais como [ao], com energia concentrada nas baixas frequências têm uma SFM muito baixa.

Peakiness (Medida de pico de ressonância)

A medida *Peakiness* é relação entre o valor RMS do espetro e a média espetral,

$$Peakiness = \frac{\sqrt{\frac{1}{N}\sum_{k}\left|X(k)\right|^2}}{\frac{1}{N}\sum_{i=1}^{N}\left|X(k)\right|}. \qquad \textbf{(2.3)}$$

Um sinal vozeado apresenta uma regularidade que contrasta com a de ruído ou com a de sons não vozeados. A energia de um sinal vozeado está normalmente concentrada à volta do pulso glotal resultando por isso em picos espetrais elevados e elevado valor de *Peakiness*. Pelo contrário sinais não vozeados têm a sua energia espalhada, resultando em valores baixos de *Peakiness*, que tendem para 1 no caso de um espetro plano (com baixa variância espetral). Esta medida é usada nos codificadores CELP (*Code-Excited Linear Prediction*) e MELP (*Mixed Excitation Linear Prediction*) na distinção entre sons vozeados e não vozeados. O exemplo da Figura 2.3 ilustra a informação fornecida por esta medida.

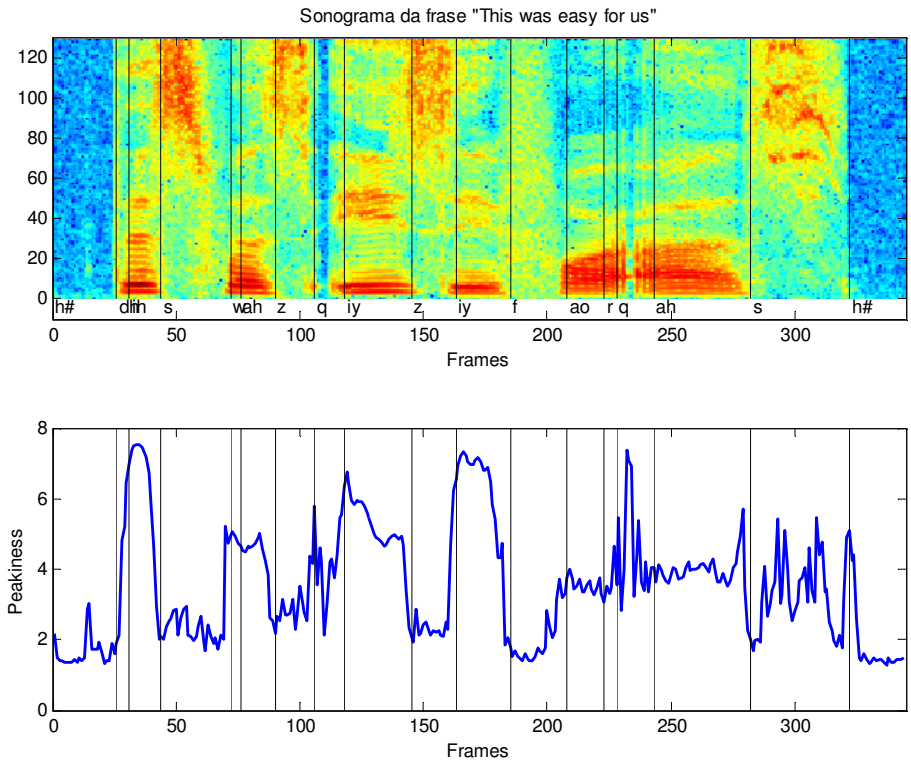


Figura 2.3: Exemplo da medida *Peakiness* na frase “This was easy for us”.

Evidência de Tom

Uma característica diferenciadora entre sinais vozeados e não vozeados é a existência de tom (F_0). A evidência de tom é muitas vezes tomada, como o valor máximo do coeficiente de correlação entre dois segmentos de sinal, $x[n]$ e $x[n+k]$,

$$\rho[k] = \frac{R_n[k]}{\sqrt{R_n[0] \cdot R_n[k]}} = \frac{\sum_n x[n] \cdot x[n+k]}{\sqrt{\sum_n x^2[n] \cdot \sum_n x^2[n+k]}} \qquad \textbf{(2.4)}$$

No caso presente, a evidência de tom é integrada no algoritmo de estimação de tom e é tomada de forma ligeiramente diferente, considerando uma média harmónica: no denominador da expressão (2.4) é tomada a média aritmética da energia dos segmentos em vez da sua média geométrica,

$$\hat{\rho}[k] = \frac{2\sum_n x[n] \cdot x[n+k]}{\sum_n x^2[n] + \sum_n x^2[n+k]} \qquad \textbf{(2.5)}$$

Na Figura 2.4 dá-se um exemplo da medida de Evidência de Tom.

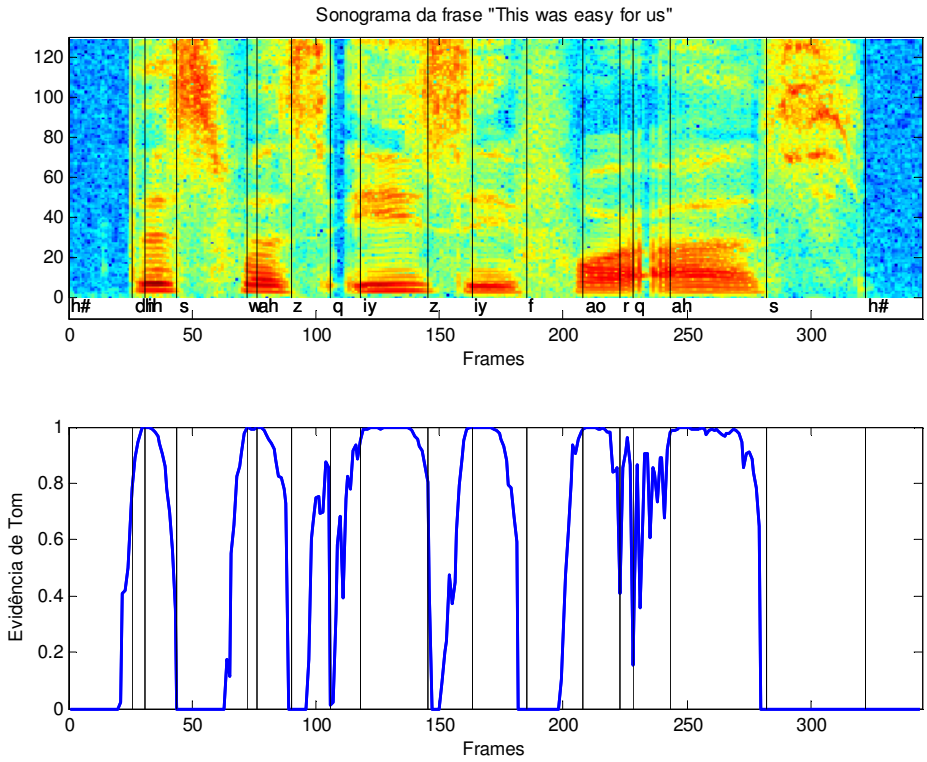


Figura 2.4: Exemplo da medida de Evidência de Tom na frase “This was easy for us”.

Relação entre energia de baixa e alta-frequência

Os sinais vozeados têm normalmente a energia concentrada nas baixas frequências quando comparados com os sinais não vozeados. Assim, foi também usada como característica a relação logarítmica entre a energia acima de 3500Hz e abaixo de 2500Hz para caracterizar a distribuição de energia no espetro do sinal.

Outras características tais como:

- energia numa janela de 35ms;
- amplitude máxima da *frame*;
- mediana das energias num banco de filtros;
- energia abaixo de 500Hz e
- energia entre 500Hz e 1500Hz,

foram também usadas em algumas experiências, nomeadamente no treino de SVMs.

2.2. CLASSIFICADORES

O problema de classificar um determinado segmento de fala ou o de descodificar uma sequência de fonemas em termos dos respetivos fones é resolvido por vários métodos seguindo abordagens e filosofias distintas. Nesta secção será feita uma breve abordagem ao problema do reconhecimento de fala seguida de uma descrição sumária das várias técnicas usadas no âmbito deste trabalho, nomeadamente HMM, ANN, SVM, fatorização de matrizes de valores não negativos e sistemas híbridos.

2.2.1. O PROBLEMA DO RECONHECIMENTO DE FALA

O reconhecimento de fala consiste em obter a sequência de palavras que melhor corresponda ao sinal acústico de fala observado. O sinal de fala começa por ser convertido numa sequência de vetores de características $X = \left[x_1, x_2, x_3, ..., x_T \right]$, que são designadas por observações. Pretende-se encontrar, de uma forma ótima, uma sequência de palavras, W^*, dadas as observações acústicas. Na abordagem estatística, essa sequência ótima é obtida de acordo com

$$W^* = \arg\max_{w} \mathrm{P}\left(W \mid X\right) \qquad \textbf{(2.6)}$$

onde *W* corresponde ao conjunto de todas das sequências possíveis de palavras.

Usando o critério de Bayes, esta expressão é equivalente à seguinte maximização:

$$W^* = \arg\max_{w} \frac{\mathrm{P}(X \mid W) \cdot \mathrm{P}(W)}{\mathrm{P}(X)} \qquad \textbf{(2.7)}$$

onde $\mathrm{P}(W)$ representa a probabilidade da sequência de palavras (modelo de linguagem), $\mathrm{P}(X \mid W)$ é a probabilidade de observar *X*, quando é pronunciada a sequência de palavras *W* (modelo acústico) e $\mathrm{P}(X)$ é a probabilidade das observações acústicas. Como as observações não variam, $\mathrm{P}(X)$ é fixo na expressão anterior pelo que (2.7) resulta em

$$W^* = \arg\max_{w} \mathrm{P}(X \mid W) \cdot \mathrm{P}(W). \qquad \textbf{(2.8)}$$

O problema é usualmente decomposto em termos de unidades de reconhecimento mais simples que palavras, tipicamente fones. Neste caso as palavras são decompostas em termos de uma sequência de fones e as observações são segmentadas assumindo uma sequência alofónica correspondente a estes fones. O modelo acústico é assim definido em termos de funções de distribuição das observações para cada fone ou para cada segmento (estado) de um fone, assumindo algumas simplificações, como é o caso da independência entre observações nos modelos de Markov não observáveis.

O problema do reconhecimento de fones pode ser definido exatamente segundo esta mesma formulação, onde se pretende encontrar a sequência ótima de fones (em vez de palavras) dado o sinal acústico.

Os modelos de reconhecimento de fala podem ser divididos em 2 tipos: discriminativos e generativos, dependendo se o treino é feito segundo (2.6) ou (2.8) respetivamente. Usando (2.8) o modelo é denominado de generativo uma vez que $\mathrm{P}(X \mid W)$ "gera" observações aleatórias segundo a sequência *W*. O treino consiste em estimar os parâmetros do modelo acústico à custa de pares observações/fones, por maximização da verosimilhança (ML - *Maximum Likelihood*) das observações. Este treino não é discriminativo no sentido em que apenas os pares "corretos" são tomados em conta na estimação do modelo.

Pelo contrário, em (2.6), $\mathrm{P}(W \mid X)$ discrimina diretamente a sequência *W* para cada sequência de observações *X*. O treino consiste em estimar as probabilidades *a posteriori* dos fones a partir do sinal acústico. Todas as observações são usadas para estimar os parâmetros do modelo, dizendo-se então que o treino é discriminativo (além dos pares observações/fones "corretos", podemos ter vários pares observações/fones "incorretos").

Os modelos probabilísticos generativos com ML foram desde sempre os mais usados em ASR. No entanto tem sido crescente o interesse nos modelos discriminativos, uma vez que melhoram a exatidão da sequência de reconhecimento (especialmente se a unidade de reconhecimento for o fone). Os modelos discriminativos procuram estruturas que maximizem o reconhecimento da classe em causa, ao mesmo tempo que procuram diferenciá-lo das restantes classes. Os métodos de treino discriminativo têm uma função objetivo que passa pela minimização de um erro observado no conjunto de treino. Diferentes critérios têm sido adotados com sucesso: "*Mutual Information*" [26][156], "*Minimum Classification Error*" [21], "*Minimum Phone/Word Error*" [115], e métodos baseados no "*Principle of Large Margin*", [165]. Estes métodos, quando aplicados no treino de HMMs, contribuíram para uma melhoria significativa do desempenho dos sistemas de reconhecimento de fala e continuam atualmente a ser um tópico ativo de pesquisa. Comparações analíticas e experimentais entre alguns destes métodos podem ser encontradas em [99][134].

Nas classes mais importantes de treino discriminativo para problemas de classificação encontram-se as ANNs, as SVMs e os CRFs. Como estas técnicas têm limitações no alinhamento temporal, uma vez que são treinadas de forma a classificar ao nível da *frame* e não do segmento, são vulgarmente combinadas com HMMs, dando origem aos sistemas híbridos. No entanto, o treino destes sistemas não é, usualmente, feito como um todo: a classificação e o alinhamento tomam treinos separados. Raramente é feito um treino integrado entre o classificador baseado em *frames* (ANNs, SVMs ou CRFs) e o modulador temporal (HMMs). Daí que, no âmbito deste trabalho, se tenha considerado o desafio do desenvolvimento de um método de treino de sistemas híbridos que se apresenta na secção 4.5.

2.2.2. MODELOS DE *MARKOV* NÃO OBSERVÁVEIS

Os modelos de *Markov* não observáveis (HMM) são modelos estatísticos poderosos, com aplicações bem-sucedidas em diversos domínios, nomeadamente no reconhecimento automático de fala. A teoria associada a estes modelos é bem conhecida e extensivamente documentada. Desta forma, serão apresentados apenas alguns conceitos básicos e notações importantes para a compreensão das secções posteriores. Informação mais detalhada pode ser encontrada em [118][119][162]. Interessante também é o artigo de Young, [163], onde

se encontra uma revisão das arquiteturas de reconhecimento de fala contínua baseadas em HMMs.

Um sinal da fala apresenta variações na sua composição espetral ao longo do tempo. Um HMM é um modelo estocástico que modela estas variações à custa de dois processos embebidos:

→ uma cadeia de *Markov* com *N* estados, cujas transições entre estados sinalizam uma mudança das propriedades estatísticas do sinal modelado, por isso uma variação temporal;

→ um processo (contínuo ou discreto) que descreve estatisticamente o sinal associado a cada estado da cadeia de *Markov*.

O HMM consiste então numa cadeia de *Markov* que determina uma sequência temporal de estados (que se assumem corresponder a estados do processo de produção de fala) e por um conjunto de funções densidade de probabilidade associadas a cada estado da cadeia.

→ A probabilidade de transição entre estados é definida pela matriz de transições com elementos a_{ij} - a probabilidade de transitar do estado *i* para o estado *j*.

→ A cada estado é associada a probabilidade condicional de observar o vetor de características do sinal de fala do instante *t*, $\mathbf{x}_t$, no estado *j*, $b_j(\mathbf{x}_t)$. Em sistemas modelados por observações contínuas (*Continuous Density Hidden Markov Model*) são definidas funções densidade de distribuição para modelar as observações. A função densidade de probabilidade contínua é, neste caso, composta por uma mistura pesada de componentes Gaussianas (mistura de Gaussianas).

Na modelação com HMMs com distribuições contínuas, assume-se que a função densidade de probabilidade associada a cada estado é definida como a soma de *M* funções Gaussianas a *Q* dimensões (mistura de *M* componentes), onde *Q* representa a dimensão do vetor de observação, **x**. A função densidade associada ao estado *j* caracteriza-se pela equação (2.9) onde μ_{jm} é um vetor de médias, $\sum_{jm}$ uma matriz de covariância; c_{jm} os pesos da mistura; $\mathcal{N}\left(\mathbf{x};\mu_{jm},\boldsymbol{\Sigma}_{jm}\right)$ a função densidade Gaussiana a *Q* dimensões dada pela equação (2.10), onde ^{T} representa transposição; $\sum_{jm}^{-1}$ representa a matriz de covariância inversa e $\left|\sum_{jm}\right|$ o seu determinante.

$$b_j(\mathbf{x}) = \sum_{m=1}^{M} c_{jm} \mathcal{N}\left(\mathbf{x}; \mu_{jm}, \Sigma_{jm}\right), \qquad \sum_{m=1}^{M} c_{jm} = 1 \tag{2.9}$$

$$\mathcal{N}\left(\mathbf{x}; \mu_{jm}, \Sigma_{jm}\right) = \frac{1}{\sqrt{(2\pi)^{Q}\left|\Sigma_{jm}\right|}} \exp\left(-\tfrac{1}{2}\left(\mathbf{x} - \mu_{jm}\right)^{T} \Sigma_{jm}^{-1}\left(\mathbf{x} - \mu_{jm}\right)\right) \tag{2.10}$$

Na Figura 2.5 apresenta-se o esquema de uma cadeia de *Markov* para modelar um fone (numa sequência). A topologia usada para a cadeia é uma topologia esquerda-direita, onde são só permitidas transições para o mesmo estado ou para estados seguintes. É ilustrada a modelação temporal em termos de uma sequência de estados (a cada estado encontra-se associada uma sequência de vetores de características que é a melhor modelada pela respetiva pdf do estado).

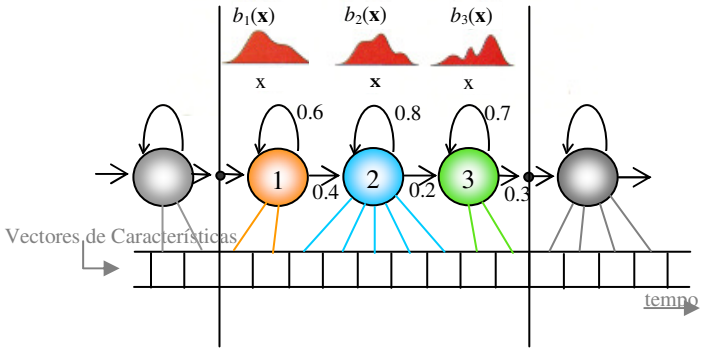


Figura 2.5: Exemplo de um modelo HMM em termos de uma cadeia de *Markov* com topologia esquerda direita e com funções densidade de probabilidade, associadas a cada estado, compostas por uma soma pesada de Gaussianas (neste caso representadas a 1D).

Na abordagem estatística de reconhecimento de fala utilizando HMMs, assume-se que a produção da sequência de observações **x** resulta de um HMM. Em cada instante *t*, o modelo "emite" um vetor de observações num estado *j* cujas propriedades seguem a função $b_j(\mathbf{x}_t)$. No entanto, apenas podemos observar a sequência de observações e não a sequência de estados intrínseca do modelo. Daí a designação de "não observável" dos HMMs. Usando um grande número de sequências de observação, podemos estimar iterativamente os parâmetros do modelo. De forma idêntica podemos determinar qual a sequência de estados mais provável, dada uma sequência de observações, bem como a probabilidade de um dado modelo ter gerado essa sequência. De seguida apresentam-se os três problemas clássicos da formulação dos HMMs, [118],[162]:

- O **problema da avaliação**: Dado um modelo e uma sequência de observações, qual é a probabilidade de o modelo ter gerado tais observações? Esta solução pode ser obtida usando o algoritmo *Forward-Backward*.
- O **problema da estimação (treino)**: Dado um modelo e uma sequência de observações, quais deverão ser os parâmetros do modelo de forma a que tenha a probabilidade máxima de gerar tais observações? A solução para este problema é conseguida através do algoritmo de Baum-Welch que usa o critério de maximização da verosimilhança ML.
- O **problema da descodificação (teste)**: Dado um modelo e uma sequência de observações, qual é a sequência de estados mais provável? A solução para este problema pode ser obtida usando o algoritmo de Viterbi.

2.2.3. REDES NEURONAIS ARTIFICIAIS

Uma rede neuronal artificial pode ser vista como uma máquina adaptativa capaz de armazenar conhecimento proveniente de experiência adquirida e torná-lo disponível para uso. Obedecem a técnicas computacionais que apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes. Uma ANN pode ter milhares de unidades de processamento; já o cérebro de um mamífero pode ter muitos milhões de neurónios. Para atingir bom desempenho, as redes neuronais aplicam uma forte interconexão entre células computacionais designadas por “neurónios”, "percetrões". “nodos” ou simplesmente "unidades". As semelhanças com o cérebro humano dizem respeito ao conhecimento que é adquirido pela rede através de um processo de aprendizagem, e às forças de ligação inter-neurónios, conhecidas como pesos sinápticos, que são usados para armazenar conhecimento. A aprendizagem a nível computacional consiste em escolher os pesos sinápticos de forma que a rede possa executar determinada função. Esta aprendizagem poderá ser feita de uma forma supervisionada, isto é, indicando qual deve ser o valor desejado da saída para cada entrada, ou não-supervisionada em que a aprendizagem é feita sem o conhecimento *a priori* dos valores desejados nas saídas, sendo a única informação disponível a correlação dos dados de entrada.

Em determinadas condições, as saídas de uma rede neuronal podem ser interpretadas como estimadores da probabilidade condicional das várias classes de saída. Desta forma a rede neuronal é usada na classificação de dados, como é o caso do presente trabalho.

Matematicamente podemos descrever o neurónio *j* com *p* entradas através de:

$$y_k = \varphi(u_k); \qquad u_k = \sum_{i=1}^{p} w_{ik} \cdot x_i - \theta_k \tag{2.11}$$

onde θ_k é a polarização (*bias*) associada à atividade u_k, a combinação linear dos parâmetros de entrada (os valores $x_1, x_2, ..., x_p$ são os sinais de entrada e $w_{1k}, w_{2k}, ..., w_{pk}$ os pesos sinápticos do neurónio *k*). A polarização θ_k pode ser vista como um peso adicional, associado a uma entrada extra x_{p+1}= -1.

A função de ativação, denotada por $\varphi(\cdot)$, define a saída do neurónio em termos do nível de atividade da sua entrada. Funções de ativação típicas são a função degrau, a função linear e a função sigmóide. Esta última é a mais utilizada por se poder aproximar da função degrau e ser diferenciável.

Existe ainda uma outra função de ativação, a função *softmax*, que, ao contrário das funções de ativação anteriores, normaliza as saídas, garantindo que a sua soma iguale a unidade, podendo as saídas representar probabilidades. A função é dada por (2.12) onde *j* representa o índice da saída da rede e é usada na camada de saída em todas as experiências com redes neuronais efetuadas neste trabalho.

$$y_k = \varphi_k(\mathbf{u}) = \frac{\exp(u_k)}{\sum_{k=1}^{p} \exp(u_k)}. \tag{2.12}$$

O poder computacional de uma rede neuronal advém da sua capacidade de generalizar após aprendizagem. A aprendizagem supervisionada, usada no treino de uma rede neuronal, consiste na modificação dos pesos sinápticos da rede neuronal através dos pares {*entrada*, *erro de saída*}.

Percetrão Multicamada

Uma classe de redes neuronais denominadas por percetrão multicamada (MLP - *Multi Layer Perceptron*) representa uma generalização do neurónio. Neste caso tem-se uma rede estruturada em camadas, em que as saídas de uma camada são as entradas de uma camada seguinte e assim sucessivamente. São referidas como redes "*feedforward*" uma vez que não existe realimentação das saídas para entradas anteriores na sequência das camadas. As saídas da última camada constituem a resposta da rede ao sinal de entrada. Sendo este último propagado na rede camada a camada. Um algoritmo de treino supervisionado muito popular, é o algoritmo de retropropagação do erro (*back-propagation*), que é baseado na aprendizagem por gradiente descendente.

Considerando d_k a resposta desejada para o neurónio *k*, e y_k o valor da resposta atual desse mesmo neurónio, podemos definir o sinal de erro como a diferença entre a resposta desejada, d_k, e a resposta obtida, y_k,

$$e_k = d_k - y_k . \quad \textbf{(2.13)}$$

O objetivo de uma aprendizagem por correção de erro é minimizar uma função de custo baseada nos erros e_k, de forma que a saída de cada neurónio da rede se aproxime da resposta desejada para esse neurónio. Um exemplo de uma função de custo muito utilizada é o *Erro Quadrático*, onde se pretende minimizar a soma do quadrado dos erros. Consideremos a função de custo como sendo o valor instantâneo da soma dos erros quadráticos das saídas, definida por:

$$\varepsilon = \frac{1}{2}\sum_{k=1}^{p} e_k^2 = \frac{1}{2}\sum_{k=1}^{p}\left(d_k - y_k\right)^2 . \quad \textbf{(2.14)}$$

Quanto mais pequeno for o valor desta função, melhores serão os valores dos pesos w_{ik}. Não existe uma solução analítica para este problema, pelo que se usam aproximações iterativas baseadas na diminuição do gradiente do erro. ε é positivo e tende para zero à medida que nos aproximamos de uma solução satisfatória.

Dada uma determinada medida de erro ε, podemos melhorar o conjunto dos pesos w_{ik} através de um "*deslizamento descendente*" na superfície definida pelo espaço dos pesos. De facto, o algoritmo do gradiente descendente sugere uma modificação de cada w_{ik} por uma quantidade Δw_{ik}, no sentido contrário do gradiente de ε na localização atual, através de uma constante de aprendizagem η, isto é,

$$\Delta w_{ik} = -\eta \frac{\partial \varepsilon}{\partial w_{ik}} . \quad \textbf{(2.15)}$$

Fazendo estas alterações individualmente para cada amostra de entrada x_i, tem-se:

$$\frac{\partial \varepsilon}{\partial w_{ik}} = -\sum_{m=1}^{p} e_m \frac{\partial y_m}{\partial w_{ik}} = -\delta_k x_i \,;\; \Delta w_{ik} = \eta \delta_k x_i , \quad \textbf{(2.16)}$$

Onde δ_k é o erro de retropropagação após passagem pela não-linearidade da função de ativação da saída (se a função de ativação for sigmoide, na última camada temos que $\delta_k = y_k(1-y_k)e_k$).

O resultado apresentado em (2.16) é conhecido por *"Regra delta"*, ou por *"Regra LMS"* (*least mean square*). O algoritmo do gradiente descendente faz descer o erro, desde que η seja suficientemente pequeno. Assim, num determinado número de iterações chega-se a um ponto arbitrariamente próximo de uma solução ótima (local).

Esta abordagem permite uma generalização para mais camadas de uma forma simples, apesar de obrigar à existência de funções de ativação diferenciáveis.

Mínima Entropia Cruzada (MEC)

Quando se pretende, como é o caso do presente trabalho, que as saídas da rede correspondam a estimativas das probabilidades de ocorrência de fones, toma-se como função de ativação nas saídas da última camada a função *softmax*. Neste caso, é conveniente considerar como função objetivo, não o critério do erro quadrático mínimo, mas o critério da entropia cruzada mínima (*Minimum Cross Entropy*). A justificação deste critério está muito bem apresentada em [27]. O erro passa a ser definido (para um único exemplar) através da expressão

$$\varepsilon = \sum_{k=1}^{p} d_k \log\left(\frac{d_k}{y_k}\right), \qquad \textbf{(2.17)}$$

onde, tal como antes, d_k é a saída desejada e y_k a saída da rede de índice k. Neste caso o algoritmo de "*back-propagation*" apresenta, ligeiras diferenças na camada de saída. Apresentam-se aqui os resultados para o critério MEC e função de ativação *softmax*, uma vez que não é habitual aparecer em textos sobre o assunto. O gradiente desta função objetivo em ordem a um peso w_{ik} da última camada passa a ser:

$$\frac{\partial \varepsilon}{\partial w_{ik}} = -\sum_{m=1}^{p} \frac{d_m}{y_m} \frac{\partial y_m}{\partial w_{ik}}. \qquad \textbf{(2.18)}$$

Assim, se definirmos o erro na última camada a retropropagar como $e_m = d_m / y_m$, o algoritmo de *back-propagation* mantém-se inalterado, isto é, mantém-se a equação (2.16). Para o caso da função *softmax*, na última camada tem-se [7]:

$$\delta_k = y_k \left(e_k - \sum_{m=1}^{p} e_m y_m \right). \qquad \textbf{(2.19)}$$

[7] Neste caso $\frac{\partial y_j}{\partial w_{ik}} = x_i y_j \left(\delta_{kj} - y_k \right)$ onde δ_{kj}=1 sse k=j.

Para as camadas interiores o algoritmo de *back-propagation* mantém-se inalterado uma vez que as funções de ativação usadas são sigmoides. As equações anteriores dizem respeito a uma única amostra; para um conjunto de treino os erros são acumulados com a apresentação de cada amostra.

Treino com retropropagação resiliente do erro – Rprop

Os percetrões multicamada usam tipicamente funções de transferência sigmoidais nas camadas escondidas. Estas funções comprimem uma gama possivelmente infinita de entradas numa gama finita de saídas. As funções sigmoidais são caracterizadas pelo facto do seu declive tender para zero quando a entrada se aproxima de +/-∞. Esta situação causa problemas quando se usa um método de gradiente descendente para treinar uma rede neuronal que possui estas funções nas suas unidades, já que o gradiente apresenta variações muito pequenas. Pequenos valores do gradiente originam pequenas atualizações nos pesos, mesmo que ainda estejam longe dos valores ótimos. O treino com retropropagação resiliente do erro, [121], elimina estes efeitos ineficientes da pequena magnitude das derivadas parciais uma vez que a atualização é feita tendo em conta apenas o sinal do gradiente. Além disso é usado um coeficiente de aprendizagem diferente para cada peso sináptico. O passo de atualização é determinado por outro valor independente. O valor de atualização para cada peso é adaptativo, sendo incrementado sempre que o sinal do gradiente (relativo a cada peso) não varia em duas épocas sucessivas de treino. Da mesma forma, o passo de atualização é decrementado sempre que o sinal do gradiente varia em duas épocas consecutivas. Enquanto os pesos oscilarem o valor da atualização vai sendo reduzido, caso contrário é incrementado. Existe no entanto um valor máximo e um mínimo para este valor de atualização.

Este algoritmo de treino apresenta, geralmente, maior rapidez de convergência e foi por isso usado neste trabalho. Também no âmbito deste trabalho, e usando a base de dados TIMIT, a atualização dos pesos foi feita em modo "*Batch*" (os pesos são atualizados após a apresentação de todas as amostras de treino).

2.2.4. MÁQUINAS DE VETORES DE SUPORTE

As máquinas de vetores de suporte enquadram-se na área da aprendizagem automática e baseiam-se na teoria da aprendizagem estatística. São uma classe muito específica de algoritmos, caracterizada pelo uso de *kernels*, ausência de mínimos locais e solução esparsa. Quando aplicada ao reconhecimento de padrões, ao contrário de outras técnicas como HMMs ou ANNs que minimizam o erro de treino, a técnica das SVMs baseia-se na minimização do risco estrutural, ou seja, procura minimizar o limite superior do erro de generalização. A atratividade das SVMs recai na sua capacidade de implicitamente transformar dados num espaço multidimensional e construir classificadores binários nesse mesmo espaço. Como isto é feito implicitamente, sem recurso a cálculos num espaço de grande dimensão, nem a dimensionalidade dos dados nem a dispersidade dos mesmos é um problema para as SVMs. A técnica foi bem-sucedida em diversas áreas. No domínio do reconhecimento de fala as aplicações são diversas, sendo [35][54][69] alguns exemplos disso. O princípio de treino em que assentam serviu ainda de inspiração para um treino discriminativo alternativo de GMMs em [138].

Considere-se o exemplo da Figura 2.6, onde se mostra um espaço de entrada bidimensional, com duas classes. Observando a figura, depreende-se que não existe nenhuma reta capaz de separar as duas classes. No entanto, se mapearmos cada vetor de entrada para um novo espaço a três dimensões, poderá ser possível encontrar um plano de separação. É este o princípio de funcionamento de uma SVM: mapeia os dados de entrada num espaço de dimensão suficientemente alta, no qual seja possível fazer separação linear dos dados com margem máxima.

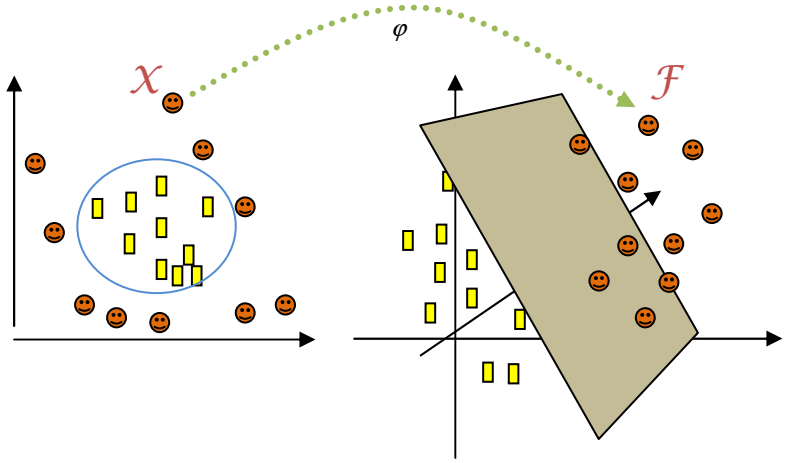


Figura 2.6: Princípio de funcionamento das SVMs: mapeamento dos dados de entrada num espaço de dimensão superior, no qual seja possível fazer separação ótima dos dados.

A teoria da aprendizagem estatística, desenvolvida por Vapnik, [152], define os conceitos base das SVMs.

A técnica das SVM assenta na classificação binária. Partindo de um conjunto de treino $\{\mathbf{x}_i, y_i\}_{i=1}^{N}$ com vetores de entrada $\mathbf{x}_i \in \Re^n$ e etiquetas correspondentes y_i, cria um hiperplano de separação ótimo entre exemplos positivos e exemplos negativos, dado pela equação:

$$\mathbf{w}^T\mathbf{x} + b = 0\,, \qquad \textbf{(2.20)}$$

onde $\mathbf{w}$ é um vector de pesos (normal do hiperplano de separação) e *b* é uma variável que permite que a margem do hiperplano seja maximizada.

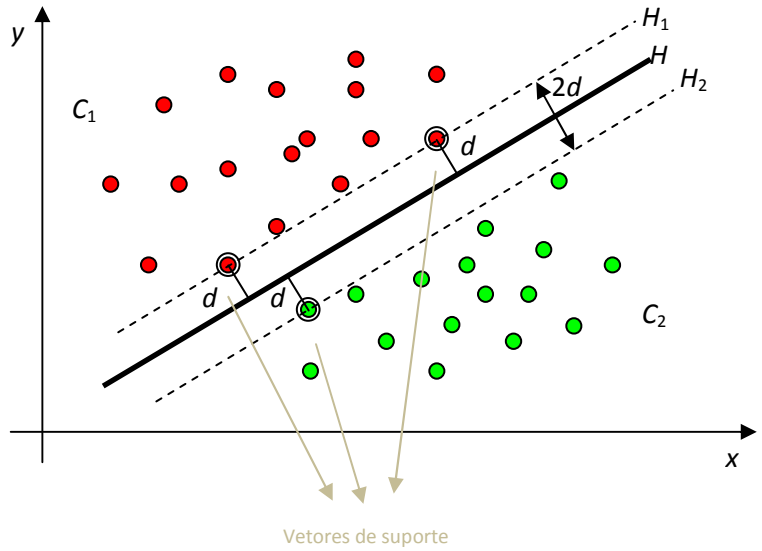


Figura 2.7: Ilustração de uma superfície ótima de separação entre duas classes C_1 e C_2.

Apesar de haver infinitos planos separadores (no caso de classes separáveis), pretende-se obter aquele que permita definir a maior margem de separação possível entre as classes. Recai-se nos hiperplanos paralelos ao hiperplano ótimo que passam pelos vetores mais próximos de cada classe, a uma distância *d*. Os vetores que definem a margem máxima, Δ_b, (vetores de suporte) são todos os vetores que estão situados sobre os dois hiperplanos H_1 e H_2 (Figura 2.7), que são paralelos a *H*, verificando-se:

H_1: $\mathbf{w}^T\mathbf{x}_i + b = \Delta_b$ ($\mathbf{x}_i$ são vetores de suporte da classe C_1)

H_2: $\mathbf{w}^T\mathbf{x}_i + b = -\Delta_b$ ($\mathbf{x}_i$ são vetores de suporte da classe C_2)

A distância dos vetores de suporte ao hiperplano *H* é $d = \frac{\left|\mathbf{w}^T\mathbf{x}_i + b\right|}{\|\mathbf{w}\|} = \frac{\left|\pm\Delta_b\right|}{\|\mathbf{w}\|} = \frac{\Delta_b}{\|\mathbf{w}\|}$.

Dada a ambiguidade entre **w** e *b*, podemos considerar, sem perda de generalidade, que Δ_b=1. Equivale a escalonar **w** e *b* com $1/\Delta_b$. Com esta escolha, a margem do classificador passa a depender apenas de $\|\mathbf{w}\|$ e vale $2d = \frac{2}{\|\mathbf{w}\|}$ (ver Figura 2.7). Portanto, maximizar a margem do classificador é equivalente a minimizar $\|\mathbf{w}\|$ ou $\|\mathbf{w}\|^2$. Para simplificar, é usual minimizar $\|\mathbf{w}\|^2/2$. As restrições são que não podem existir vetores entre os dois planos. As classes C_1 e C_2 são definidas de acordo com:

$$\mathbf{w}^T\mathbf{x}_i + b \geq 1 \qquad \text{se } y_i = +1 \qquad (C_1)$$

$$\mathbf{w}^T\mathbf{x}_i + b \leq -1 \qquad \text{se } y_i = -1 \qquad (C_2)$$

Uma vez que se define a pertença com y_i=±1, as duas condições anteriores podem ser combinadas numa única: $y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) - 1 \geq 0$, $\forall i$. Assim, o problema a resolver é $\min_{\mathbf{w}}\left\{\frac{1}{2}\|\mathbf{w}\|^2\right\}$ sujeito às restrições $y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) - 1 \geq 0$. Uma vez encontrado **w**, o valor de *b* pode ser obtido com os vetores de suporte, obrigando a que $y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) - 1 = 0$.

Na fase de teste, dado um vetor $\mathbf{x}_i$, a SVM classifica-o como pertencente à classe C_1 ou C_2 de acordo com o lado do hiperplano a que pertence:

$$\mathbf{w}^T\mathbf{x}_i + b > 0 \quad \Rightarrow y_i = +1 \qquad (C_1)$$

$$\mathbf{w}^T\mathbf{x}_i + b < 0 \quad \Rightarrow y_i = -1 \qquad (C_2)$$

ou seja, a classe de pertença será $y_i = \text{sgn}(\mathbf{w}^T\mathbf{x}_i + b)$, onde sgn() é a função sinal.

Em problemas reais as classes quase nunca são perfeitamente separáveis. Para possibilitar uma correta classificação destes exemplos surgem as variáveis de folga ("*slack*"), $\boldsymbol{\xi} = [\xi_1, \ldots, \xi_N]^T$ e o problema (primal) de uma SVM passa a ser a minimização da função:

$$J_p(\mathbf{w}, \boldsymbol{\xi}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i \quad \text{sujeita a} \quad \begin{cases} y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) \geq 1 - \xi_i \\ \xi_i \geq 0, \quad i = 1, \ldots, N \end{cases}. \qquad \textbf{(2.21)}$$

Este problema de otimização pode apresentar dificuldade na obtenção de solução, devido principalmente à natureza das restrições. Por este motivo, é comum recorrer-se a uma formulação dual para o problema, mais simples de resolver. É então aplicado o Lagrangeano

a (2.21) resultando no problema dual, que deve verificar as condições KKT (Karush-Kuhn-Tucker),[31].

As funções de *kernel*, usualmente separáveis na forma $K(\mathbf{x}_i,\mathbf{x}_j)=\varphi(\mathbf{x}_i)^T\varphi(\mathbf{x}_j)$, como as da Tabela 2.1, realizam um produto escalar no próprio espaço de entrada, e não no espaço de características de dimensão mais elevada. Graças às funções de *kernel*, problemas não linearmente separáveis podem ser resolvidos pelas SVMs, uma vez que no espaço de dimensão mais elevada pode ser mais fácil encontrar uma superfície de separação linear.

Kernel	Expressão
Linear	$\mathbf{x}_i^T\cdot\mathbf{x}_j$
RBF	$e^{-\left\|\mathbf{x}_i-\mathbf{x}_j\right\|^2/2\sigma^2}$
Polinomial	$\left(\mathbf{x}_i^T\cdot\mathbf{x}_j+a\right)^b$
Sigmóide	$\tanh\left(\beta_0\mathbf{x}_i^T\cdot\mathbf{x}_j+\beta_1\right)$

Tabela 2.1: Exemplos de funções de *kernel*.

Após o processo de treino e resolução do problema dual (por via de um algoritmo de programação quadrática) não é necessário fazer o mapeamento $\varphi(\mathbf{x}_i)$. Basta usar as funções de *kernel*, juntamente com os exemplares de treino, os multiplicadores de Lagrange, α_i, e um parâmetro *C*, definido empiricamente. O valor de predição de pertença à classe para um vetor $\mathbf{x}_k$ qualquer é dado por:

$$f(\mathbf{x}_k)=\sum_{i=1}^{N}\alpha_i^* y_k K(\mathbf{x}_i,\mathbf{x}_k)+b^*\ , \qquad \textbf{(2.22)}$$

enquanto a classificação binária corresponde simplesmente a aplicar a função sgn() a este valor. Uma característica interessante é que muitos valores α_i^* só são não nulos para os vetores de suporte que estão nos hiperplanos de separação, como exemplificado na Figura 2.7 (no caso separável).

Uma boa e extensa base teórica sobre SVMs pode ser encontrada no tutorial de Burges [18] e também em [150][151].

Tal como referido, a técnica das SVM assenta num princípio de classificação binária. No entanto em muitas aplicações a tarefa envolve a classificação de múltiplas classes. Uma solução passa por combinar vários classificadores binários normalmente seguindo uma de duas estratégias:

um contra um - um classificador binário é treinado para cada uma das possíveis combinações de duas classes. É um método de fácil implementação mas onde o número de classificadores binários cresce com o aumento do número de classes. Os resultados individuais de cada classificador são tipicamente combinados usando *Directed Acyclic Graph*, [114]e a árvore binária, [131].

um contra todos - treina um classificador binário por cada classe. O resultado da classificação de cada classe é comparado com o resultado da classificação do conjunto formado por todas as restantes classes. Neste método, o número de classificadores é relativamente pequeno, mas na fase de treino a quantidade de memória necessária é elevada uma vez que cada classificador é treinado com todos os dados de treino. A combinação de classificadores pode ser feita usando o *Error-correcting output code*, [25] que combina a informação dos vários classificadores calculando uma distância a um *codebook* ou simplesmente classificando como vencedora a classe com maior valor de predição, como foi o caso do presente trabalho.

2.2.5. FACTORIZAÇÃO DE MATRIZES DE VALORES NÃO NEGATIVOS

A fatorização de matrizes de valores não negativos (NMF - *Non-negative Matrix Factorization*), é uma técnica que visa decompor um grande conjunto de dados numa combinação de um número relativamente pequeno de fatores (bases), reduzindo assim a dimensão dos dados em análise. É uma técnica semelhante a *Principal Component Analysis*, *Singular Value Decomposition*, *Factor Analysis* e a outros métodos clássicos de fatorização matricial, cujo objetivo é o de reduzir o número de variáveis e encontrar uma estrutura comum (bases) entre os dados envolvidos. A diferença está em que estes métodos não garantem manter a não negatividade da factorização, enquanto a NMF garante. Baseia-se na ideia que em muitas tarefas de processamento de dados, os valores negativos envolvidos não têm um significado físico. A procura de bases representativas dos dados de entrada é então confinada a valores não negativos, [148]. Formalmente esta ideia pode ser interpretada como a decomposição uma matriz **V**, de valores não negativos, em duas matrizes **W** e **H** igualmente constituídas por valores não negativos, usando-se como critério a minimização da norma *Frobenius* do erro de aproximação:

$$\min_{\mathbf{W},\mathbf{H}\geq 0} \left\| \mathbf{V} - \mathbf{WH} \right\|_F^2 \qquad \textbf{(2.23)}$$

Este paradigma foi intitulado *Non-Negative Matrix Factorization*.

A ideia da NMF tem o seu início em 1994 num trabalho de Paatero e Tapper [111] sob o nome de *Positive Matrix Factorization*, englobado num programa de monitorização de fontes poluentes ambientais. O algoritmo de fatorização inicial baseia-se nos mínimos quadrados para resolver o problema, mas em trabalhos seguintes, [112] desenvolveram outros tipos de otimização, mas com uma complexidade elevadíssima. Contudo, os seus trabalhos não faziam prova da convergência do método, o que fez com que este não suscitasse grande interesse junto da comunidade científica. Paralelamente, Lee e Seung introduziram o conceito de NMF num trabalho de 1997 sobre treino não supervisionado, [72]. O objetivo é encontrar o melhor conjunto de vetores característicos que representem os dados de entrada. Isto conduziu-os ao problema (2.23), onde as colunas de **V** contêm exemplos de treino e **W** tem uma característica (*rank*) menor que **V**. Num trabalho posterior, [73] Lee e Seung desenvolveram um algoritmo iterativo muito simples para resolver o problema (2.23). O algoritmo generaliza-se facilmente a um outro critério: minimização da medida de divergência modificada *Kullback-Liebler*. Em ambos os casos as regras de atualização são multiplicativas, o que é perfeitamente adequado à descida do gradiente. O procedimento é aplicado várias vezes com condições iniciais diferentes. Lee e Seung demonstram que as regras multiplicativas de atualização convergem para um mínimo local das respetivas funções objetivo.

O algoritmo NMF proposto por Lee e Seung [73][74] surgiu na tentativa de identificar partes de objetos complexos de forma não supervisionada. A grande vantagem do método é a garantia de convergência sob uma única restrição, que obriga os dados a serem não negativos. O problema NMF descrito por Lee e Seung [73][74], fatoriza uma base de dados de imagens de faces em duas matrizes (não negativas), produzindo uma representação das imagens como uma combinação aditiva de partes independentes (nariz, olhos, boca, etc.). Por esta razão, a restrição de não negatividade é compatível com a noção intuitiva de combinar partes para formar um todo. Se aplicado a um sonograma de um segmento musical, cada base poderá representar as notas de cada instrumento. Se aplicado ao reconhecimento de imagens de dígitos manuscritos, as bases podem representar as formas manuscritas dos dígitos, [72]. O algoritmo surge então aplicado ao domínio do reconhecimento de imagens: faces, como em [50][155], ou sequências de imagens médicas como em [75] e [130], mas tem provado ser uma técnica com aplicações em muitos domínios. Novak e Mammone [109][110] utilizam o NMF no domínio da adaptação do modelo de linguagem em sistemas de reconhecimento de fala. Pretendem fazer transcrição automática de sessões de uma conferência. Na área do processamento de texto são também

inúmeras as aplicações. Em [140] e [159] o NMF aparece aplicado à classificação de documentos onde as bases representam as palavras de classificação. Já em [68] o NMF é usado com o objetivo de melhorar o desempenho de sistemas de reconhecimento óticos de carateres e de indexação de documentos, servindo para a identificação da fonte do texto em documentos digitalizados. No domínio do processamento de áudio/vídeo surgem trabalhos como os de Cooper e Foot [22] e os de Kim e Sikora aplicados à norma MPEG-7, [63][64]. Também na genética o algoritmo dá o seu contributo. Em [17] através da fatorização milhares de genes consegue-se descrever um pequeno conjunto de meta genes. O NMF surge também aplicado ao domínio do processamento de sinais de áudio. Smaragdis e Brown, [144] usam uma modificação do NMF para modelar música polifónica, e em [145] Smaragdis modela o espetro de duas palavras. Sha e Saul, [139] recorrem ao algoritmo para identificar a presença de uma ou mais vozes humanas. As bases modulam timbres particulares em sinais de fala vozeados que variam em função do contexto fonético e em função do locutor. Behnke, no seu trabalho [10], procura hierarquicamente características importantes no sinal de entrada de forma a que possam ser úteis em tarefas de compressão ou classificação.

O algoritmo de Lee-Seung

A popularidade do algoritmo de Lee-Seung na resolução do problema de NMF (2.23), pode ser atribuída quer à sua simplicidade quer ao seu desempenho prático. O algoritmo pode ser visto como um algoritmo de gradiente descendente. Requer um baixo custo computacional por iteração e já provou fornecer soluções de elevada qualidade. O paradigma da factorização recai então na solução do seguinte problema: dada uma matriz $\mathbf{V} \in \Re^{m \times n}$ e um inteiro positivo $r < \min\{m,n\}$, encontrar duas matrizes não negativas $\mathbf{W} \in \Re^{m \times r}$ e $\mathbf{H} \in \Re^{r \times n}$ de modo que:

$$\underset{m \times n}{\mathbf{V}} \approx \underset{m \times r}{\mathbf{W}} \, \underset{r \times n}{\mathbf{H}} \tag{2.24}$$

Estas matrizes, segundo Lee e Seung, [73], podem ser obtidas através das regras multiplicativas de atualização dadas em (2.25) ou (2.26) consoante a função de custo escolhida seja a distância Euclidiana, $\left\|\mathbf{V}-\mathbf{WH}\right\|$, ou a medida de divergência de Kullback-Leibler, $D\left(\mathbf{V}\middle\|\mathbf{WH}\right)$, respetivamente.

$$\mathbf{H}_{ab} \leftarrow \mathbf{H}_{ab} \frac{\left(\mathbf{W}^T\mathbf{V}\right)_{ab}}{\left(\mathbf{W}^T\mathbf{WH}\right)_{ab}} \qquad \mathbf{W}_{ab} \leftarrow \mathbf{W}_{ab} \frac{\left(\mathbf{VH}^T\right)_{ab}}{\left(\mathbf{WHH}^T\right)_{ab}} \tag{2.25}$$

$$\mathbf{H}_{ab} \leftarrow \mathbf{H}_{ab} \frac{\sum_i \mathbf{W}_{ia} \frac{\mathbf{V}_{ib}}{(\mathbf{WH})_{ib}}}{\sum_i \mathbf{W}_{ia}} \qquad \mathbf{W}_{ab} \leftarrow \mathbf{W}_{ab} \frac{\sum_j \mathbf{H}_{bj} \frac{\mathbf{V}_{aj}}{(\mathbf{WH})_{aj}}}{\sum_j \mathbf{H}_{bj}} \tag{2.26}$$

Os valores iniciais das matrizes **W** e **H** são gerados aleatoriamente e depois atualizados em várias iterações. Estas atualizações podem ser vistas de forma análoga ao método de gradiente descendente onde a constante de aprendizagem é obtida através da fatorização atual.

Para ilustrar o desempenho do NMF aplicou-se o método a um sonograma artificial constituído por apenas dois segmentos distintos. Um dos segmentos corresponde a uma locução de /Et/ (do dígito sete (s/Et/@)) e o segundo a /S/ (em três (tre/S). Os modelos ocorrem várias vezes e entre eles existe silêncio. Ambos os segmentos são constituídos por 10 *frames*. A matriz de entrada (**V**) corresponde ao espetro LPC (*Linear Predictive Coding*), em dB, com 129 pontos. As matrizes **W** e **H** foram inicializadas aleatoriamente e foram feitas 20 iterações do algoritmo. Procuraram-se 2 bases (*r=2*) de forma a encontrar os dois padrões espetrais do sinal. Os resultados são apresentados na Figura 2.8.

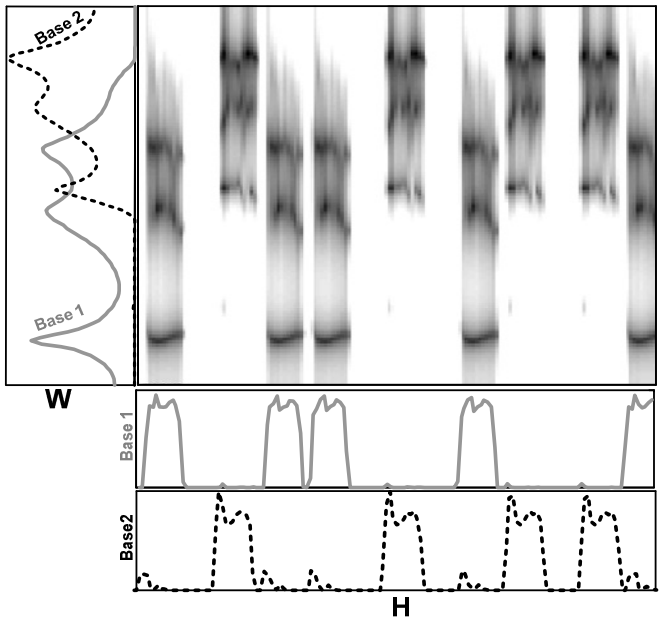


Figura 2.8: Resultados do NMF. A imagem central representa os dados de entrada (espetrograma com valores não negativos). As duas colunas de **W**, correspondentes às bases espetrais, são mostradas à esquerda enquanto no gráfico de baixo se apresentam as linhas da matriz **H** que representam o peso de cada base em função do tempo.

De forma a interpretar os resultados, examinemos a decomposição obtida. A entrada é uma matriz $\underset{m\times n}{\mathbf{V}}$ com *m* pontos de frequência e *n frames*. Esta matriz é decomposta em duas matrizes $\underset{m\times r}{\mathbf{W}}$ e $\underset{r\times n}{\mathbf{H}}$. A matriz **W** contém as bases espetrais nas suas duas colunas, evidenciando as frequências dominantes presentes no sonograma, enquanto a matriz **H** descreve a estrutura temporal nas duas linhas, i.e. a forma como as bases aparecem ao longo do tempo. Por análise da figura conclui-se que esta decomposição revelou a estrutura dos dados de entrada, descrevendo os seus elementos quer em frequência quer temporalmente. Mostra-se assim que o NMF descobre e extrai facilmente objetos estáticos[8] num espetro. Paris Smaragdis, [145] mostra que mesmo que os objetos se sobreponham, o NMF deteta eficientemente a sua estrutura e localização. Para lidar com o problema de um objeto possuir variações temporais, Smaragdis em [144][145] propõe uma extensão ao algoritmo NMF exposto, denominada *Non negative Matrix factor Deconvolution* (NMD) que passamos a descrever.

Non-negative Matrix Factor Deconvolution

No algoritmo NMF os objetos são descritos pelo seu espetro e sua correspondente energia ao longo do tempo. Nos seus trabalhos [144][145] Smaragdis propõe uma extensão ao NMF que considera não só um padrão por objeto, mas uma sequência de padrões sucessivos e sua correspondente energia ao longo do tempo.

Detalhes sobre as diferenças entre o NMF e NMD podem ser encontrados em, [92]. Para ilustrar o desempenho do método e as diferenças em relação ao NMF considere-se a Figura 2.9. Neste exemplo foram calculadas 3 bases usando *T*=17 e o método foi iterado 200 vezes. As bases são mostradas no gráfico mais à esquerda onde a evolução espetral dos objetos está encapsulada nas colunas de todas as matrizes $\mathbf{W}^{(t)}$. As linhas de **H** (presentes no gráfico de baixo) apontam o local onde a base tem início.

[8] Entenda-se por objeto estático aquele que do ponto de vista temporal e frequencial mantém ou repete as suas características.

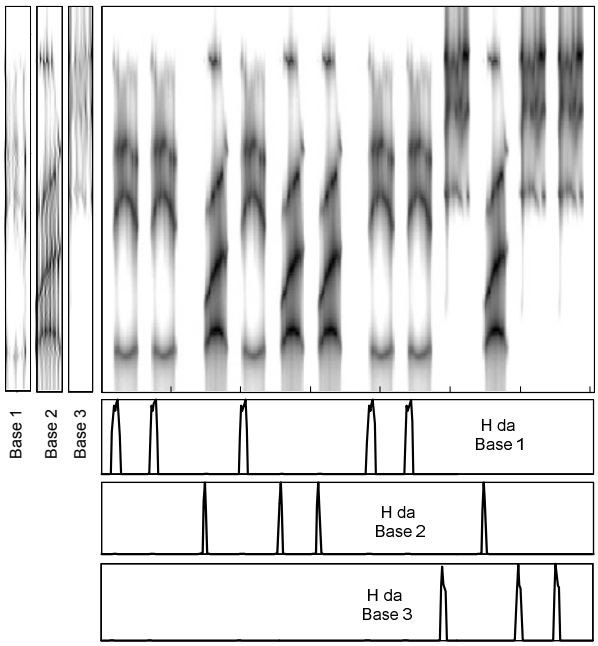


Figura 2.9: Decomposição por NMD**.** A imagem central representa os dados de entrada, o gráfico da esquerda mostra as bases espetrais e no gráfico de baixo as linhas de **H** apontam o local onde a base tem início.

Neste exemplo todos os objetos foram corretamente detetados. Comparando os resultados com a decomposição por NMF e por NMD conclui-se que apesar de o NMF detetar as frequências dominantes do sinal, não evidencia a forma como elas evoluem ao longo do tempo. Pelo contrário o NMD representa claramente padrões tempo-frequência (do mesmo tamanho). Uma vantagem adicional deste tipo de decomposição é o facto de podermos reconstruir o espetrograma original, a partir dos objetos (que se retiram das colunas de $\mathbf{W}^{(t)}$) e da sua localização (contida nas linhas de **H**). No caso do NMF se quisermos extrair um objeto *k* faríamos:

$$\mathbf{V}_k \approx \mathbf{W}_{(:,k)}\mathbf{H}_{(k,:)} \tag{2.27}$$

onde $\mathbf{W}_{(:,k)}$ é a coluna *k* de **W** e $\mathbf{H}_{(k,:)}$ é a linha *k* de **H**. Já no caso do NMD a extração do objeto *k* é feita pela convolução:

$$\mathbf{V}_k \approx \sum_{t=0}^{T-1} \mathbf{W}_{(:,k)}^{(t)} \overset{t\rightarrow}{\mathbf{H}}_{(k,:)} \tag{2.28}$$

onde $\overset{t\rightarrow}{\mathbf{H}}$ representa o deslocamento das colunas de **H** *t* vezes para a direita colocando colunas a zero à esquerda.

Apesar do bom desempenho quer do NMF quer do NMD os métodos têm um desempenho limitado quando os objetos (apesar de poderem manter a sua estrutura global) sofrem uma translação em frequência ou exibem uma expansão ou compressão temporal. A Figura 2.10 ilustra estas situações. A figura da esquerda representa o mesmo espetrograma que a da Figura 2.9, mas com objetos deslocados na frequência. Examinando as linhas de **H** e comprando-as com as da Figura 2.9, conclui-se que o NMD não lida bem com esta situação: alguns objetos não são bem detetados e outros nem sequer são detetados. Outra limitação do NMD é a incapacidade de lidar com expansões ou compressões temporais dos objetos. A Figura 2.10b) exemplifica esta limitação. Olhando para as linhas de **H** verificamos que a expansão teve um grande efeito na deteção de todas as bases (especialmente na primeira e segunda). Pior ainda: não afetou só a deteção dos objetos modificados, mas também a dos restantes. Isto é particularmente evidente na deteção da base 1.

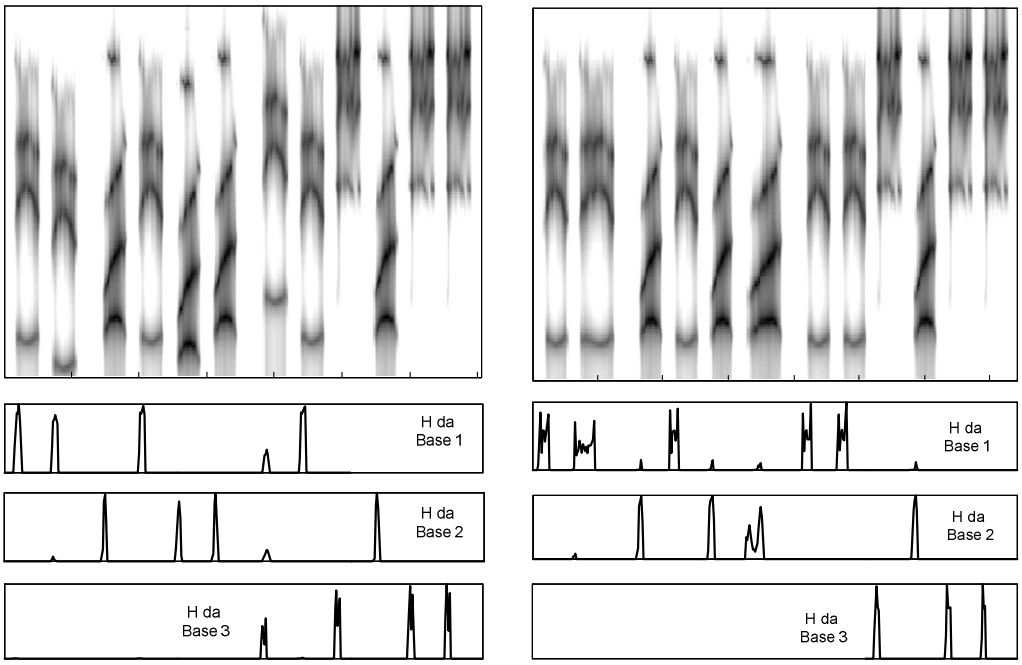


Figura 2.10: Resultados do NMD: a) aplicados a um espetrograma com objetos deslocados na frequência; b) aplicados a um espetrograma com objetos temporalmente expandidos.

As limitações apresentadas condicionam a aplicação do método NMD diretamente à classificação de dados reais, nomeadamente ao reconhecimento de fones onde este tipo de variações é uma constante. No seguimento do trabalho apresentado em [92] tentou-se, sem sucesso, fazer uma reformulação ao problema da NMD que contornasse a questão das variações tempo-frequência dos padrões de fala. O método provou, contudo, ser eficiente na deteção de eventos de fala, como mais à frente se mostrará.

2.2.6. SISTEMAS HÍBRIDOS

Os modelos HMM são amplamente usados em reconhecimento de fala. O seu sucesso advém da sua capacidade de modelar características quer temporais quer acústicas do sinal de fala. Expressam o sinal estatisticamente ao mesmo tempo que modelam a sua evolução temporal. Por outro lado, os classificadores como ANNs ou SVMs têm a vantagem de serem naturalmente discriminativos. No entanto, são classificadores estáticos pois não modelam inerentemente a evolução temporal dos dados (fazem classificação ao nível da *frame*). Os sistemas híbridos surgem como uma possível solução para este problema: combinar HMMs com classificadores discriminativos num sistema único de forma a beneficiar das melhores características de ambos. A forma como é conjugada a informação dos dois sistemas assume geralmente um de dois cenários:

- As saídas dos classificadores são usadas como observações nos HMM. Esta configuração é vulgarmente chamada de *Tandem* – a saída de um sistema alimenta a entrada do seguinte.
- As saídas dos classificadores substituem as pdf's associadas aos estados dos HMMs, tradicionalmente definidas por misturas Gaussianas, mantendo os HMMs a restante topologia inalterada. Seja $Q=\{c_j\}$ um conjunto de *J* classes de padrões. A probabilidade *a posteriori* da classe c_j, representada por $P(c_j|\mathbf{X})$, é a probabilidade de um padrão pertencer à classe c_j dada a ocorrência do vetor de observações **X**. Se o classificador tiver *J* valores de saída normalizados[9], estes são vistos como estimativas da distribuição de probabilidade das classes condicionada à entrada, $y_j(\mathbf{X})=P(c_j|\mathbf{X})$. A probabilidade associada a um estado de um HMM, que se pode associar ao evento ou classe c_j, pode ser estimada a partir da saída correspondente do classificador. De facto, aplicando-se a regra de Bayes, tem-se a seguinte relação:

$$\frac{p(\mathbf{X}|c_j)}{p(\mathbf{X})}=\frac{P(c_j|\mathbf{X})}{P(c_j)} \qquad \textbf{(2.29)}$$

Isto é, as pdf´s associadas aos estados podem ser substituídas pelo quociente entre as saídas do classificador e a probabilidade de ocorrência da classe (ou estado) c_j. A partir daqui, o reconhecimento é feito usando os algoritmos de descodificação tradicionais (algoritmo de Viterbi no presente caso).

[9] No caso de se tratar de um MLP será uma rede com *J* unidades de saída. Se for um sistema baseado em SVMs numa configuração *um-contra-todos*, deverá conter *J* SVMs a fim de fornecer *J* saídas.

O processo de descodificação clássico num modelo híbrido pode ser interpretado como uma procura da melhor segmentação de estados, na medida em que os estados correspondem, segundo esta abordagem, diretamente a fones. Assim, qualquer sequência de estados corresponde naturalmente a uma segmentação fonética e vice-versa. Contudo, uma vez que esta abordagem condiciona a modelação temporal dos modelos, surgem alternativas, [32][154], onde cada fone é modelado por mais do que um estado. No caso de serem usados os usuais 3 estados, o primeiro estado tem a seu cargo a modelação do início do fone, o estado do meio a parte central e o terceiro estado a parte final do fone. Em [154] é sugerida uma abordagem algo semelhante à usada no treino clássico com HMMs em que os estados iniciais e finais modelam transições, no caso apresentado, difones. De realçar o facto de as abordagens multi-estado obrigarem a que os classificadores não sejam treinados só em função do conjunto de fones pretendido mas sim em função dos estados usados no modelo híbrido. Em [1], Abad e Neto treinam um MLP onde adicionam ao número de saídas das classes que pretendem reconhecer, um conjunto adicional de saídas que representam as transições mais frequentes presentes nos dados de treino. Em [147] é feita uma revisão de um número significativo e sistemas híbridos HMM/ANN).

No presente trabalho foram usados sistemas híbridos combinando o resultado de classificação dado pelos métodos descritos em 2.2.3, 2.2.4 e 2.2.5 com HMMs. Os classificadores foram treinados em função dos fones e classes pretendidas, fornecendo assim, uma única saída por cada elemento a classificar. De forma a impor um modelo de duração, os modelos HMM são compostos por 3 estados, mas os estados partilham entre si a mesma saída do classificador.

2.3. METODOLOGIA DE ANÁLISE DE DESEMPENHO

Na avaliação de um sistema de reconhecimento devem ser usadas medidas que não só permitam uma análise cuidadosa e uma interpretação clara do desempenho do sistema, mas que também estejam adequadas à tarefa em questão. E, se por exemplo, é suficiente comparar a sequência de palavras reconhecidas com as proferidas numa tarefa de ditado o mesmo não se poderá dizer se, por exemplo, pretendermos identificar apenas determinadas palavras num sinal de fala. Neste último caso interessa avaliar não só a sequência de palavras detetadas bem como os instantes em foram reconhecidas.

O presente trabalho centrou-se nas tarefas de reconhecimento fonético e na deteção de eventos. Obrigando as tarefas a uma metodologia de avaliação diferente, foram adotadas

medidas também diferentes. No caso da deteção de eventos foi proposta uma nova metodologia de avaliação, [91]. As subsecções seguintes descrevem separadamente as medidas usadas.

2.3.1. AVALIAÇÃO DE SEQUÊNCIAS FONÉTICAS

Nos sistemas de reconhecimento de fala, a medida de avaliação de desempenho vulgarmente usada é a taxa de erro de palavras (WER - *Word Error Rate*)[10] ou a taxa de exatidão correspondente, Word *Accuracy* Rate.

O cálculo da exatidão (*Accuracy*), para um dado conjunto de resultados de reconhecimento, requer a existência de transcrições de referência para as locuções de teste. A comparação entre a sequência de etiquetas[11] reconhecida (REC) e a sequência de etiquetas de referência (REF) é feita através de um algoritmo de programação dinâmica que calcula uma medida da diferença entre duas sequências, normalmente conhecida como "*edit distance*" ou distância de *Levenshtein*. De forma a otimizar a correspondência entre as sequências REC e REF, o algoritmo considera três operações de edição: substituição (S), apagamento (D) e inserção (I). O WER é então a distância de *Levenshtein* entre as sequências REC e REF, normalizada pelo tamanho da sequência de referência. Esta normalização é aplicada de forma a permitir a comparação entre sistemas diferentes em tarefas diferentes, uma vez que a magnitude da distância depende do tamanho da sequência.

Assim, define-se *Accuracy* como:

$$Accuracy(\%) = 1 - WER = \left(1 - \frac{S + D + I}{N}\right) \times 100 \qquad \textbf{(2.30)}$$

onde:

N é o número total de etiquetas do conjunto de teste;

S é o número total de etiquetas substituídas por outras do vocabulário;

D é o número total de erros de apagamentos e

I é o número total de erros de inserções (não aparecem na sequência de referência).

Se definirmos *H* como o número de etiquetas corretamente reconhecidas, (2.30) toma a forma,

[10] No caso da unidade de reconhecimento ser o fone a medida mantém-se mas toma a designação de Phone Error Rate (PER).

[11] Etiqueta refere-se à unidade em que se está a medir o desempenho (palavras, fones, ou outra).

$$Accuracy(\%) = \frac{H - I}{N} \times 100 \quad \textbf{(2.31)}$$

.

Outra medida similar é a *Correctness*, (2.32) cuja única diferença em relação à anterior é o facto de não considerar os erros de inserção. Mede simplesmente a percentagem de acertos face ao número total de etiquetas.

$$Correctness(\%) = \frac{H}{N} \times 100 \quad \textbf{(2.32)}$$

Pacotes de *software* de reconhecimento de fala, tal como o HTK, [162] incluem ferramentas para calcular a *Accuracy* e as medidas relacionadas, tendo como base os dados transcritos e as saídas do ASR, usando um algoritmo de programação dinâmica.

Tal como referido, a distância de *Levenshtein* procura a melhor correspondência possível entre as etiquetas das sequências REC e REF. Não se trata de alinhamento temporal, uma vez que o algoritmo não considera a informação dos instantes de tempo em que cada palavra/fone foi reconhecido. Assim, as medidas apresentadas são adequadas a tarefas onde o interesse recaia só na sequência de etiquetas reconhecidas, mas são inadequadas se as marcas temporais onde ocorrem essas mesmas palavras/fones forem importantes.

2.3.2. AVALIAÇÃO DE DETEÇÃO DE EVENTOS

O objetivo de um sistema de deteção de eventos passa pela determinação da ocorrência de elementos importantes no sinal de fala. Num sistema de reconhecimento de fala, os eventos podem ser combinados para detetar fones, palavras ou frases, ou para identificar marcos temporais aos quais um classificador ou um descodificador possa ser sincronizado. Os limites temporais dos eventos são, então, tão importantes como a identificação do próprio evento. Assim, na avaliação de um sistema de deteção de eventos, deve ser tomada em consideração não só a sequência de eventos detetada, mas também os instantes em que foram detetados.

Tipicamente, os sistemas de deteção de eventos são avaliados usando as medidas de reconhecimento de fala, [46][54][56][81][133] e segmentação de fala, [49],[85]. Os sistemas de segmentação são em geral avaliados comparando as marcas de alinhamento automáticas com as marcas manuais de referência, [49],[62],[66]. Trata-se de uma avaliação de natureza quantitativa e conduz-nos ao paradigma da segmentação: *Que percentagem das marcas encontradas pelo segmentador automático é que estão corretas?*

A medida de avaliação usada conduz a uma medida da *concordância* entre marcas. Diz-se *concordância* e não número de *marcas corretas* uma vez que se admite uma tolerância, [49]. Assim, o desempenho do sistema é avaliado definindo-se um intervalo de tolerância em que se considera a fronteira correta. Este intervalo é normalmente de ± 20ms. Ou seja, se a marca alinhada estiver a menos de 20ms da marca de referência considera-se que a marca alinhada está em concordância com a manual.

Define-se então taxa de concordância (*Agreement*) como:

$$Agree(\%) = \frac{N_C}{N_T} \times 100\,, \tag{2.33}$$

sendo N_C o número de fronteiras em concordância com o alinhamento manual e N_T o número total de fronteiras existentes.

Quando o sistema de segmentação não faz alinhamento forçado (tem disponível somente o sinal acústico), deixa de estar disponível o número de etiquetas da locução em causa e por isso o número de limites encontrados pelo método dificilmente coincide com o da locução de referência (a do alinhamento manual), dando origem a inserções (I) e apagamentos (D) de limites[12]. Nesta situação a taxa de *concordância* torna-se insuficiente para avaliar a qualidade da segmentação. Surge assim um outro indicador que considera estes erros - a *taxa de exatidão*.

$$Exact(\%) = \frac{N_T - D - I}{N_T} \times 100\,, \tag{2.34}$$

sendo *D* o número de apagamentos e *I* de inserções.

De notar que este indicador pode tomar valores negativos, nas situações em que existem muitas inserções e muitos apagamentos.

Outras duas medidas que derivam da teoria da recuperação de informação (*Information Retrieval*) bem como dos testes estatísticos na teoria da deteção, são designadas por *Precision* e *Recall*. Estas medidas aplicadas à avaliação da segmentação (como em [46][66]) são dadas por:

$$Precision = \frac{N_C}{N_C + I} \qquad Recall = \frac{N_C}{N_C + D} \tag{2.35}$$

[12] Considera-se uma inserção quando o sistema automático deteta uma marca que não existe no alinhamento manual e considera-se um apagamento na situação contrária, i.e. quando o alinhamento automático não deteta uma marca existente na referência.

Fazendo uma média harmónica (pesada) entre a *Precision* e a *Recall* obtém-se a medida F_β, onde β é o peso dado à *Precision* e à *Recall* [123].

$$F_\beta = \frac{\left(1+\beta^2\right)\times Precision\times Recall}{\beta^2\times Recall+Precision} \quad \textbf{(2.36)}$$

Se β=1, $F=F_1$ corresponde à média harmónica entre *Precision* e *Recall*.

Tal como referido, as medidas *Accuracy* e *Correctness* são úteis quando a tarefa é de reconhecimento, mas frágeis se pretendermos reconhecer eventos. Veja-se o exemplo da Figura 2.11.

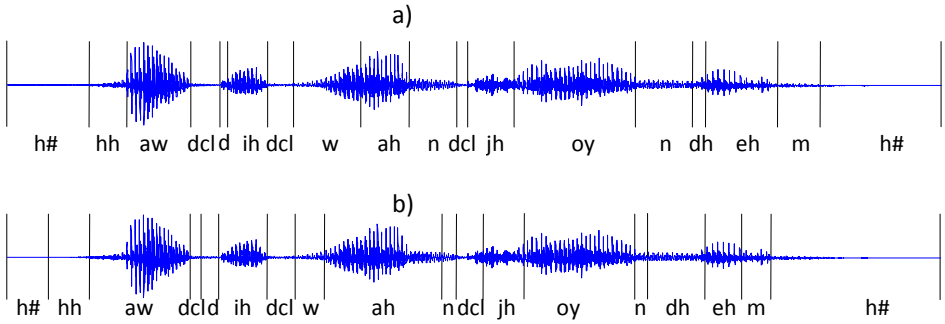


Figura 2.11: a) Locução da base de dados TIMIT ("*how did one join them*") alinhada manualmente, b) Exemplo de saída de um sistema ASR.

O sinal a) em cima corresponde a uma locução da base de dados TIMIT, [36], alinhada manualmente, enquanto o sinal b) em baixo é um exemplo de saída de um sistema de reconhecimento de fones. Como a sequência de etiquetas na Figura 2.11 a) e Figura 2.11b) é a mesma, o alinhamento entre elas é perfeito, o que corresponderá a um reconhecimento ótimo (*Accuracy*=*Correctness*=100%). Contudo, para a deteção de eventos este resultado é incorreto; alguns segmentos (etiquetas) nem sequer se sobrepõem no tempo. Então, se o alinhamento não for feito também em função de uma correspondência temporal as medidas *Accuracy* e *Correctness* pouco dirão acerca do desempenho do sistema.

2.3.3. AVALIAÇÃO COM ALINHAMENTO TEMPORAL

Nenhuma medida de avaliação parece adequada a sistemas de reconhecimento de eventos. Uma solução possível consiste em definir uma função de alinhamento entre duas sequências (referência e teste) na qual é atribuída uma penalização aos pares de etiquetas, em função do desalinhamento entre as marcas temporais. O procedimento comum é alinhar as

sequências de etiquetas de acordo com o algoritmo de *Levensthein*, [41]. Neste algoritmo, se as etiquetas coincidirem o custo é nulo. Na proposta que se apresenta é incluída uma penalização que é proporcional à média dos desalinhamentos entre as marcas de início e fim de cada etiqueta.

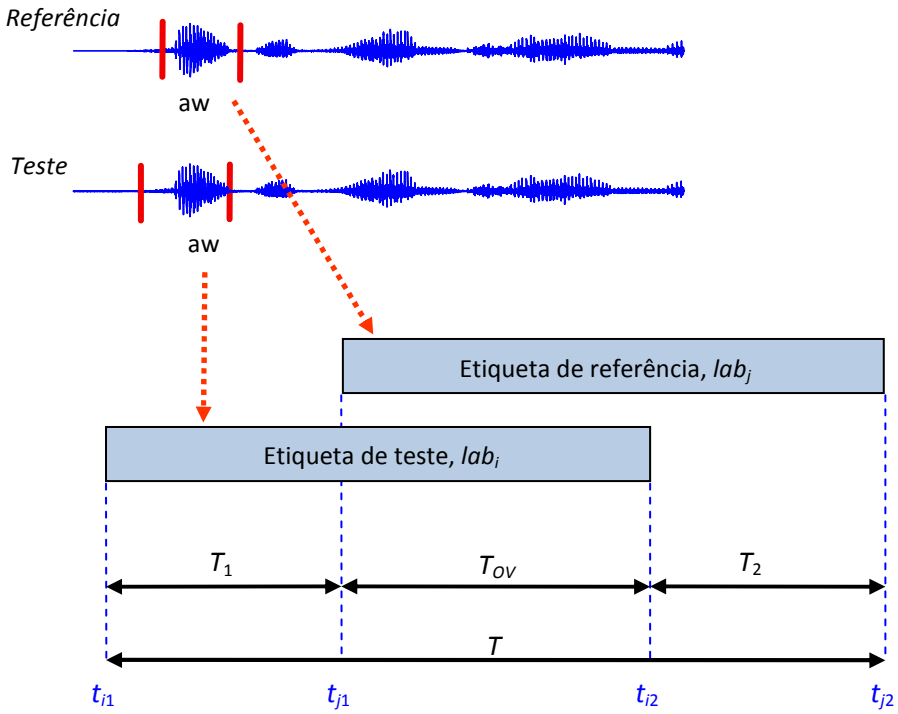


Figura 2.12: Medidas do desalinhamento das marcas temporais entre duas etiquetas.

Se as etiquetas não se sobrepuserem ($T_{OV} \leq 0$ na Figura 2.12), é atribuída uma penalização máxima (p_{max}), de forma que seja preferível incluir erros de inserção ou de apagamento a permitir uma associação de etiquetas que não se intersetam em qualquer instante de tempo.

Considerando t_{i1}, t_{i2} e t_{j1}, t_{j2} como os limites das etiquetas de teste e de referência, respetivamente, como indicado na Figura 2.12, então

$$T = \max(t_{i2}, t_{j2}) - \min(t_{i1}, t_{j1}) = T_1 + T_2 + T_{OV}, \quad \textbf{(2.37)}$$

o tempo de sobreposição é:

$$T_{OV} = \min(t_{i2}, t_{j2}) - \max(t_{i1}, t_{j1}) \quad \textbf{(2.38)}$$

e os desalinhamentos à esquerda e à direita são, respetivamente, $T_1 = \left| t_{j1} - t_{i1} \right|$ e $T_2 = \left| t_{j2} - t_{i2} \right|$. Se as etiquetas lab_i e lab_j coincidirem, mas não estiverem perfeitamente

alinhadas é atribuída uma penalização – penalização de associação $p_A(i,j)$ – inversamente proporcional ao tempo de coincidência, de acordo com a seguinte expressão:

$$p_A(i,j) = \frac{(T_1 + T_2)/2}{T_{OV}} = \frac{1}{2}\left(\frac{T}{T_{OV}} - 1\right) \quad \textbf{(2.39)}$$

Se as etiquetas se sobrepuserem mais de 50%, p_A é menor que 0.5. À medida que a sobreposição diminui, esta distância aumenta tendo como teto p_{max}=15, o que corresponde a uma sobreposição de 3.2%.

De acordo com o algoritmo de *Levensthein* há quatro tipos de associação (acerto, substituição, inserção e apagamento), cada um com uma penalização respetiva. Na Tabela 2.2 apresentam-se as penalizações (custo de fazer um erro) usadas na ferramenta de avaliação de resultados do HTK (`HResults`) e as usadas pelo método que se propõe.

Tipo de alinhamento	Penalizações HTK	Penalizações propostas
Hit	$p_{HIT} = 0$	$p = p_A$
Substitution	$p_{SUB} = 10$	$p = p_A + p_{SUB}$; $p_{SUB} = 7$
Insertion	$p_{INS} = 7$	$p_{INS} = 4$
Deletion	$p_{DEL} = 7$	$p_{DEL} = 4$

Tabela 2.2: Tipos de alinhamento do algoritmo de *Levensthein* e penalizações correspondentes.

O algoritmo de programação dinâmica é então definido de acordo com a equação:

$$D(i,j) = \min\begin{cases} D(i-1,j) + p_{INS} \\ D(i,j-1) + p_{DEL} \\ D(i-1,j-1) + p(i,j) \end{cases} \quad \textbf{(2.40)}$$

onde

$$p(i,j) = p_A(i,j) + \begin{cases} p_{SUB} & , lab_i \neq lab_j \\ 0 & , lab_i = lab_j \end{cases} \quad \textbf{(2.41)}$$

e onde $D(i,j)$ é a distância acumulada até ao nodo (i, j) no espaço de alinhamento. O alinhamento ótimo obtém-se fazendo o percurso contrário desde $D(m,n)$ até à origem (*backtracking*), onde m e n se referem, respetivamente, ao tamanho da sequência de teste e de referência. De realçar que neste caso as substituições são mais penalizadas que na situação comum. Se duas etiquetas não estiverem consideravelmente sobrepostas, então é

melhor optar por um apagamento ou por uma inserção do que aceitar uma substituição. De forma a demonstrar o desempenho do método proposto apresentam-se dois exemplos. A Figura 2.13 ilustra uma saída de um detetor de fricativas. A avaliação passa pela comparação (alinhamento) entre as sequências de etiquetas: a de teste e a de referência (mostradas na Figura 2.14). Os segmentos que não correspondem a eventos são etiquetados como "não-eventos".

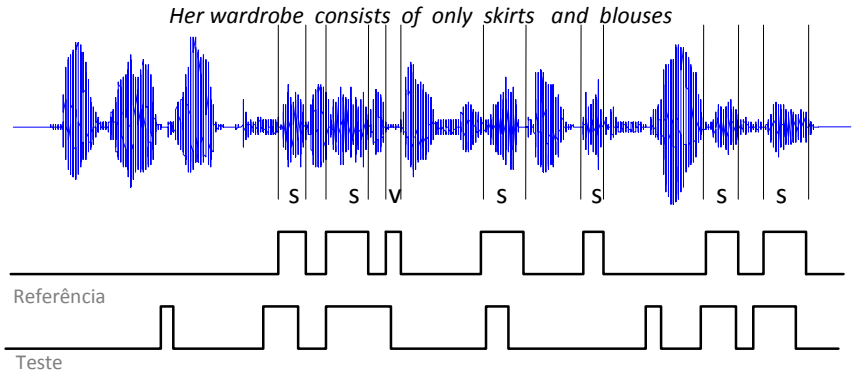


Figura 2.13: Exemplo da saída dada por um detetor de fricativas.

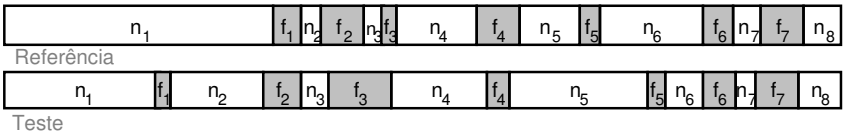


Figura 2.14: Sequências de etiquetas de teste e de referência a ser alinhadas.

Se calcularmos a *Accuracy* usando o `HResults` do HTK [162] obtêm-se um desempenho ótimo (100%) uma vez que as sequências de etiquetas de referência e de teste são perfeitamente iguais. Olhando para as marcas temporais verifica-se que é uma medida de avaliação de desempenho desadequada à aplicação em causa. Há alinhamentos que não são possíveis de todo:

- o evento de teste f_1 pode ser alinhado com a referência n_1 ou pode ser considerado uma inserção, mas nunca deve ser alinhado com a referência f_1;
- o evento de teste f_3 pode ser reconhecido como a referência f_2, n_3 ou f_3, mas nunca com eventos com os quais temporalmente não tem nada em comum.

Neste exemplo os eventos de teste f_1, n_2, f_5 devem resultar em inserções e os eventos de referência n_3, f_3 e f_5 em apagamentos.

Considere-se um segundo exemplo, ilustrado na Figura 2.15, onde se pretende separar períodos de fala de períodos de silêncio. Comparando a sequência de referência com a sequência reconhecida verifica-se que o desempenho do sistema está longe do ideal.

No entanto segundo o alinhamento típico (ferramenta `HResults` do software HTK) ilustrado na Figura 2.16 a) o sistema gera somente um erro - uma inserção no final da locução. O custo do alinhamento é 7. Valor que corresponde à penalização associada a um erro de inserção (Tabela 2.2).

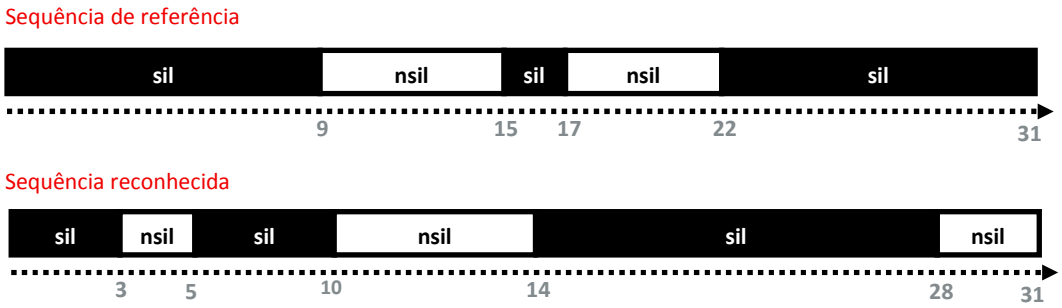


Figura 2.15: Sequências de etiquetas de teste e de referência a ser alinhadas.

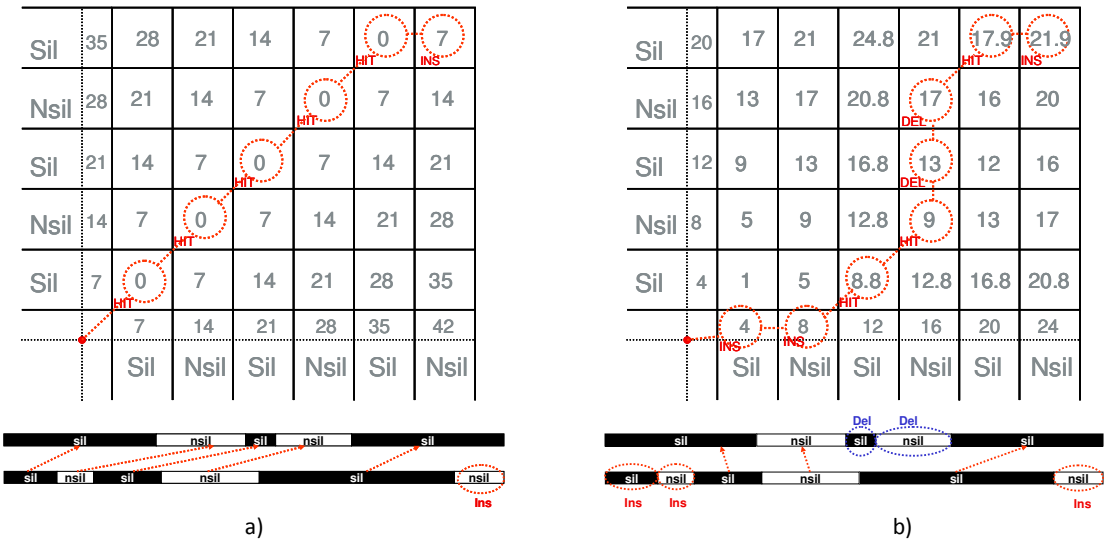


Figura 2.16: Para a mesma tarefa de alinhamento exemplo das diferenças entre:

a)Alinhamento típico HTK (`HResults`); b) Alinhamento proposto.

Na Figura 2.16 b) apresenta-se o mesmo problema de alinhamento, mas considerando o novo conjunto de penalizações. A grelha é preenchida de acordo com (2.40). O custo deste alinhamento é agora substancialmente superior (21.9), correspondendo a 5 erros: três inserções e dois apagamentos.

O método proposto é significativamente mais rigoroso dado que analisa não só a sequência de etiquetas, mas também as respetivas marcas temporais.

De forma a comparar os resultados do método de avaliação proposto com os do método que usa o alinhamento do HTK, foi avaliado o desempenho de um reconhecedor de fricativas usando o conjunto de teste completo da base de dados TIMIT, [36]. Os 61 fones da TIMIT foram reduzidos a duas classes (‘*fri*’, ‘*nfri*’). Aos fones {’f’, ’th’, ‘z’, ’s’, ’zh’, ’sh’, ’jh’, ’ch’} foi atribuída a etiqueta ‘*fri*’, e a todos os restantes a etiqueta ‘*nfri*’. Mais detalhes podem ser encontrados em [91].

Para avaliar a qualidade da segmentação, foram calculadas as taxas *Correctness* e *Accuracy* de acordo com (2.31) e (2.32) e também a taxa de concordância – *Agreement* - (dos acertos) com as marcas temporais manuais em intervalos de 10, 20 e 30ms, de acordo com (2.33). A Tabela 2.3 apresenta os valores obtidos, quer com o método de avaliação proposto, quer com o HTK. Este método será referido como "avaliação com alinhamento temporal" (AAT).

(%)	Corr.	Acc.	Agree. (10ms)	Agree. (20ms)	Agree. (30ms)
`HResults` (HTK)	94.44	**85.52**	50.27	**59.17**	61.85
Avaliação com Alinhamento Temporal	92.26	**81.21**	72.54	**85.2**	88.98

Tabela 2.3: Comparação entre os resultados obtidos usando o método de avaliação proposto (AAT) e a ferramenta `HRsults` do HTK em termos das taxas *Correctness*, *Accuracy* e *Agreement*, [91].

Comparando ambos verifica-se que a *Accuracy* baixa se o alinhamento for feito considerando a sobreposição entre etiquetas, note-se contudo que a qualidade das marcas temporais dos acertos é significativamente superior. O método proposto tem um desempenho de *Agreement* 26% acima do HTK, (margem de $\pm$20 ms) sacrificando, contudo, 4.3% de *Accuracy*. Esta queda deve-se ao facto de o método proposto gerar muito mais inserções (23.8%) e apagamentos (38.2%) e à existência também de erros de substituição, ao contrário da ferramenta do HTK onde este último tipo de erro nunca ocorre (o custo de substituir é superior ao custo quer de apagar quer de inserir).

Apesar de o método conduzir a taxas de *Accuracy* inferiores, avalia com muito mais rigor o alinhamento o que é de extrema importância na avaliação de detetores de eventos de fala. Em ambos os métodos indicados na Tabela 2.3 foram calculados os desalinhamentos à esquerda (T_1) e à direita (T_2), como definidos na Figura 2.12. O histograma das distâncias entre as marcas de teste e de referência para os dois métodos em consideração é apresentado na Figura 2.17. Se não for usada informação temporal no alinhamento, as distâncias resultantes serão altas, porque o método de alinhamento pode associar etiquetas

que temporalmente podem não ter nada em comum. Neste caso 36% das distâncias das marcas distam mais de 50ms das de referência. É interessante constatar que com o método proposto mais de 50% das distâncias estão a menos de 5ms das de referência e se se permitir uma margem de 20ms então 85% estão em conformidade com o alinhamento manual. Estes resultados indicam que o método proposto é apropriado para avaliar um sistema baseado em eventos.

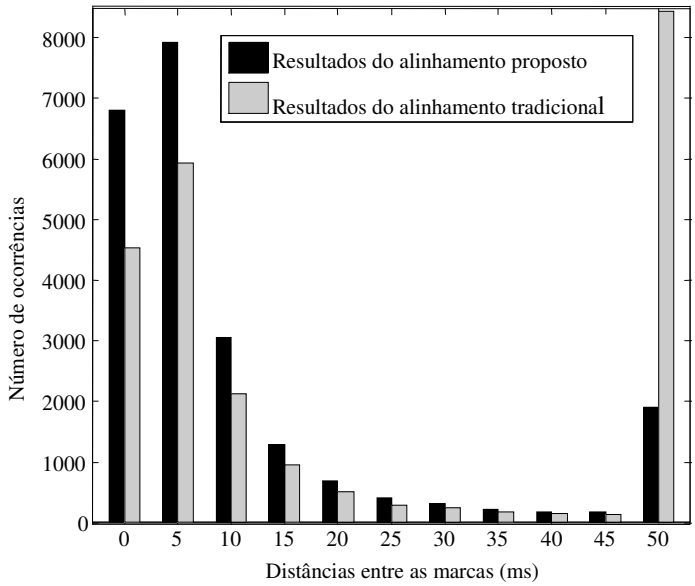


Figura 2.17: Histograma das distâncias entre as marcas dos eventos bem reconhecidos e as marcas dos eventos de referência.

CAPÍTULO 3

RECONHECIMENTO DE CLASSES FONÉTICAS ALARGADAS

A unidade de reconhecimento mais usada nos sistemas de reconhecimento automático de fala é, como se disse, o fone. No entanto há fones com características acústico-fonéticas semelhantes e que se confundem facilmente. Experiências em classificação de fones usando a base de dados TIMIT, [36] mostram que 80% dos fones incorretamente classificados correspondem a fones semelhantes, [45][132]. Assim, tem sido sugerido [45] [104][132], que representações intermédias, entre o sinal de fala e as unidades fonéticas correspondentes, possam ser usadas em auxílio à classificação. Estas representações intermédias tomam, na literatura, diversas denominações: classes fonéticas alargadas (CFA), grupos fonéticos alargados, eventos, etc., mas em todos os casos, referem-se a conjuntos de fones com características acústico-fonéticas semelhantes. As CFA têm sido objeto de estudo em vários domínios nomeadamente na identificação automática da língua, [61]; na estimação do débito de fala, [166], em sistemas multilingue, [160][167] e, especialmente no reconhecimento de fones, [104][132][143]. O sucesso do seu uso prende-se com o facto de representarem informação adicional que comprovadamente contribui para a melhoria das taxas de reconhecimento, especialmente em condições de ruído, [129]. Na tarefa de reconhecimento de fones, a informação de CFA pode ser usada como um conjunto adicional de parâmetros de entrada ou pode ser integrada nas predições ao nível do fone. Um exemplo bem-sucedido é apresentado por Siniscalchi *et al* em [143] um dos melhores resultados reportados na tarefa de reconhecimento de fones na TIMIT, onde 15 classes articulatórias alargadas são usadas na predição de probabilidades *a posteriori* dos fones e para repontuar *lattices*[13]. Outro exemplo interessante é dado por Morris e Fosler--Lussier [104], onde as saídas de 8 classificadores de CFA são usadas como parâmetros de entrada num CRF.

[13] *lattices* são grafos de palavras, fones, etc. provenientes do reconhecimento de fala, contendo as diferentes alternativas de transcrição.

3.1. CLASSES FONÉTICAS ALARGADAS

A definição de classes fonéticas alargadas refere-se ao agrupamento de fones em categorias ou classes. Cada fone é classificado como pertencente a uma determinada classe. Esta classificação pode ser feita manualmente, por um perito em fonética (por via do saber/conhecimento), ou automaticamente.

No caso de o agrupamento ser feito por um perito, as categorias são selecionadas de acordo com as propriedades acústico-fonéticas que advêm do conhecimento da produção articulatória, de regras de fonologia ou da perceção auditiva (classes doravante denominadas por *knowledge-based*).

No caso de o agrupamento ser feito automaticamente (classes doravante denominadas por *data-driven*), as classes geradas têm normalmente origem em algoritmos de agrupamento que usam alguma medida de semelhança ou distância entre fones. Abordam-se, de seguida e separadamente, cada uma das vertentes de classificação mencionadas.

3.1.1. DEFINIÇÃO DE CLASSES FONÉTICAS ALARGADAS BASEADA EM FONÉTICA

A definição das CFA para a língua inglesa é um assunto amplamente estudado pela comunidade científica. Está geralmente relacionada com o modo e local de articulação dos fones de forma que todas as classes apresentem uma forte concordância em aspetos fonéticos, articulatórios e/ou acústicos. Mas, a construção de conjuntos de fones não é uma tarefa trivial, o que se confirma pela falta de consenso existente entre várias propostas. Por exemplo, Juneja em [55] classifica o fone [hv] como uma consoante soante enquanto em [132] e [45] é considerado uma fricativa. Este tipo de discordâncias de classificação ocorre em quase todas as propostas analisadas. Enquanto em [55] o fone [dx] é classificado como soante, Ali em [6] atribui ao mesmo fone a classificação de plosiva e Halberstadt em [45] a classificação de *nasal/flap*. Se compararmos a classe "fricativas" proposta em [2] e [132] verificamos que a primeira inclui o fone [hv] enquanto a segunda não[14].

Os exemplos mencionados pretendem evidenciar a subjetividade associada às abordagens baseadas em conhecimento humano, mesmo que realizadas por especialistas e mesmo quando nos referimos a línguas muito estudadas como é o caso do inglês americano. Esta subjetividade afeta não só o número de classes como os fones associados a cada classe.

[14] No Anexo I encontra-se uma lista com os símbolos fonémicos e fonéticos usados na TIMIT, bem como exemplos de palavras e transcrições fonéticas possíveis.

3.1.2. DEFINIÇÃO AUTOMÁTICA DE CLASSES FONÉTICAS ALARGADAS

Em alternativa à definição de classes fonéticas alargadas do tipo *knowledge-based*, existem os métodos automáticos de agrupamento de fones que extraem informação dos dados para definir as classes fonéticas. Estas classes podem ser definidas de acordo com os dados acústicos ou de acordo com o resultado de um sistema automático de reconhecimento de fones.

Uma questão central em todos os algoritmos de agrupamento de fones é a escolha de uma medida de proximidade ou de distância entre fones. Esta medida pode ser obtida a partir dos modelos acústicos [15][160] ou a partir da matriz de confusão [52][141][167]. Os métodos baseados em modelos (*model-driven*) e os baseados em confusões entre fones (*confusion-driven*[15]) representam então as duas categorias principais associadas aos algoritmos de agrupamento do tipo *data-driven*.

Nos métodos *model-driven*, a semelhança acústica entre um par de fones pode ser obtida a partir de uma medida de distância entre os modelos acústicos desses mesmos fones. Esta distância pode ser, por exemplo, a distância de *Bhattacharyya* entre dois modelos de misturas Gaussianas, [15][160], ou uma medida de distância relativa baseada em entropia (divergência de *Kullback-Leibler*) entre duas misturas Laplacianas, [65], etc. Em [19] é proposta a geração de classes fonéticas *data-driven*, onde a informação mútua é aplicada ao cálculo da semelhança entre modelos, enquanto Mareuil em [100] propõe uma medida de semelhança baseada na verosimilhança entre *frames* acústicas e modelos HMM.

Nos métodos baseados nas confusões entre fones, [141][167], a medida de semelhança surge por análise da matriz de confusão (MC). Esta matriz é obtida a partir da saída de um reconhecedor de fones alinhando a sequência reconhecida com a sequência de fones de referência (algoritmo de *Levenshtein*). Esta matriz inclui *acertos*, *confusões*, *inserções* (INS) e *apagamentos* (DEL). Um exemplo de parte de uma matriz de confusão de um reconhecedor de fones (da base de dados TIMIT) é apresentado na Figura 3.1. Os elementos da diagonal indicam o número de vezes em que o fone foi corretamente classificado (acertos) e os fora da diagonal representam os erros por confusão com outros fones.

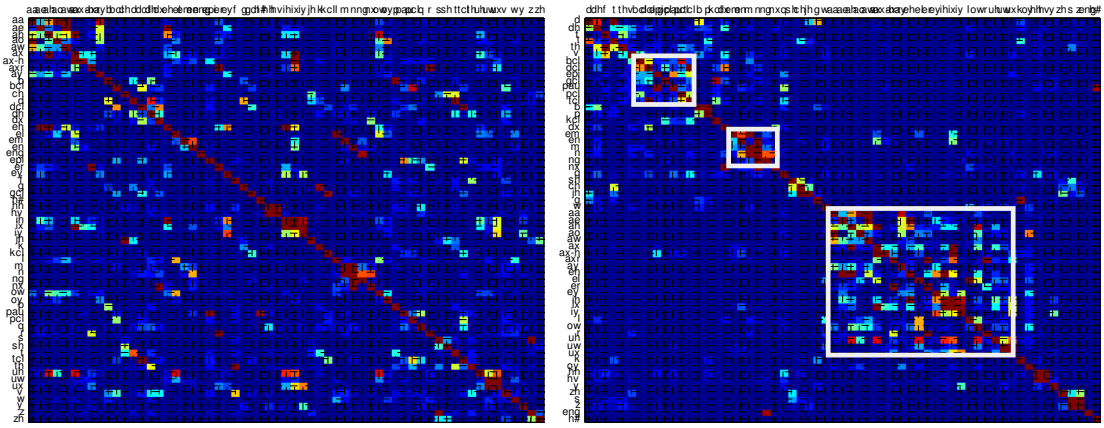
[15] A designação confusion-driven foi tomada uma vez que o agrupamento de fones é função de uma matriz de confusão.

		RECONHECIDO						
		aa	ae	ah	ao	aw	ax	DEL
REFERÊNCIA	aa	456	8	52	87	16	3	125
	ae	12	448	23	2	10	5	88
	ah	45	31	369	16	9	71	111
	ao	67	2	21	441	8	7	113
	aw	16	14	6	6	121	0	14
	ax	5	2	64	17	8	592	217
	INS	21	21	19	17	14	30	

Figura 3.1: Secção de uma matriz de confusão: saída de um sistema de reconhecimento de fones usando a TIMIT.

A ideia subjacente aos métodos *confusion-driven* é que fones semelhantes tendem a confundir-se mais e por isso devem pertencer à mesma classe fonética.

A Figura 3.2a) apresenta em pseudocores uma matriz de confusão usando os 61 fones da TIMIT[16]. A escala de cores vai desde azul (o valor mais baixo) até ao vermelho escuro (o valor mais elevado). Se se juntarem fones que se confundem significativamente, a mesma matriz resulta na Figura 3.2b). Observando esta matriz torna-se claro que há grupos (*clusters*) com uma concentração de erros entre si muito maior do que com outros grupos ou fones. É no sentido de encontrar este conjunto de *clusters* que se explora, neste trabalho, um método de classificação automática onde as CFA são definidas de acordo com o resultado de um sistema de reconhecimento automático de fones. Os fones são agrupados de acordo com a sua semelhança, que é calculada a partir da análise da matriz de confusão do reconhecedor.



a) b)

Figura 3.2: a) Matriz de confusão de um reconhecedor usando a TIMIT;

b) Matriz de a) mas juntando fones que se confundem com frequência.

[16] Nesta matriz de confusão as linhas correspondem a fones (classes de referência) e as colunas às classificações dos fones reconhecidos.

3.2. AGRUPAMENTO DE FONES COM BASE NA MATRIZ DE CONFUSÕES

Nesta secção define-se um método de geração automática de CFA baseado na matriz de confusão de um reconhecedor de fones. Genericamente as técnicas de *clustering* envolvem 3 conceitos:

1. Modelo de representação dos dados;
2. Critério de proximidade (semelhança, distância, etc.);
3. Algoritmo de *clustering* que gera os *clusters* (as classes) usando o modelo dos dados e a medida de semelhança.

No presente caso, os dados são representados pela matriz de confusão originária de um sistema de reconhecimento de fones da TIMIT. O critério de proximidade usado bem como o algoritmo de agrupamento são descritos nas próximas subsecções.

3.2.1. PROPOSTA DE MEDIDA DE DISTÂNCIA ENTRE FONES

Na literatura encontram-se várias propostas de medidas de semelhança. Zgank, [167], define uma medida de semelhança entre fones usando o número de confusões entre fones. Os fones são agrupados definindo um limiar que depende de um peso que é escolhido empiricamente de acordo com resultados experimentais. Apesar da dificuldade na obtenção deste peso, o método proposto por Zgank para a geração de classes fonéticas, melhorou os resultados de reconhecimento quando comparado com o método baseado em conhecimento (*knowledge-driven*).

Em [52][141] a matriz de confusão é convertida numa matriz simétrica de semelhança usando a medida de *Houtgast*. Esta mede a semelhança entre as classes *i* e *j* usando o número de vezes que estes são reconhecidos como um fone diferente *k*. Se *N* for o número total de classes (fones), a semelhança de *Houtgast* entre os fones *i* e *j*, s_{ij} é dada por (3.1).

$$s_{ij} = s_{ji} = \sum_{k=1}^{N} \min\left(f_{ik}, f_{jk} \right) \qquad \textbf{(3.1)}$$

com $1 \le i \le N$ e $1 \le j \le N$ onde f_{ij} é o número de confusões entre o fone *i* e o fone *j* (ou o número de acertos se $i = j$). De acordo com esta medida, dois fones *i* e *j* são semelhantes se ambos forem confundidos com os mesmos fones. A semelhança entre dois fones *i* e *j* será nula se os fones com quem *i* se confunde forem todos diferentes dos fones com quem *j* se confunde. A Figura 3.3 mostra um exemplo simples desta medida.

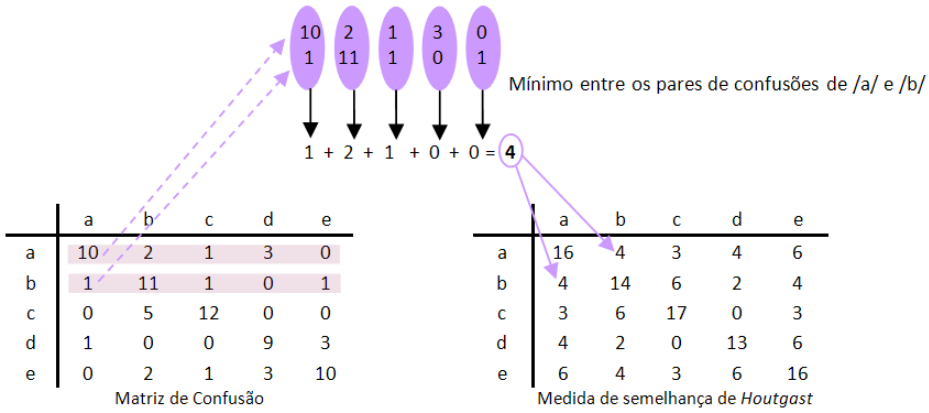


Figura 3.3: Exemplo de uma matriz de semelhanças de *Houtgast*.

A matriz de confusão de um sistema de reconhecimento de fones de qualidade elevada, aproxima-se de uma matriz diagonal. Os valores fora da diagonal representam confusões entre os fones. Como o número de ocorrências de cada fone é diferente (a base de dados não contém material foneticamente balanceado), pode normalizar-se a matriz de confusão, dividindo a frequência de ocorrência f_{ij} pelo número total de exemplos do fone de referência *i* na base de dados:

$$p_{ij} = \frac{f_{ij}}{\sum_{n=1}^{N} f_{in}} = P\left(\hat{c}_j \mid c_i\right) \qquad \textbf{(3.2)}$$

Desta forma ficamos com uma estimativa de $P(\hat{c}_j \mid c_i)$, a probabilidade de classificar um fone em termos do fone $\hat{c}_j$ quando a sua classe de referência é c_i. De acordo com esta definição, a medida de semelhança de *Houtgast* passa a ser:

$$s'_{ij} = \sum_{n=1}^{N} \min\left(p_{in}, p_{jn}\right) = \sum_{n=1}^{N} \min\left(P(\hat{c}_i \mid c_n), P(\hat{c}_j \mid c_n)\right). \qquad \textbf{(3.3)}$$

Esta nova medida goza das seguintes propriedades:

$$s'_{ij} \leq 1 \ ; \qquad \textbf{(3.4)a}$$

$$s'_{ii} = 1 \ . \qquad \textbf{(3.4)b}$$

Uma vez que $\min(a,b) = \frac{1}{2}\left(a + b - |a - b|\right)$ e $\sum_{n=1}^{N} p_{in} = 1$, pode-se definir uma medida de distância entre fones de acordo com (3.5):

$$d_1(c_i, \hat{c}_j) = 1 - s'_{ij} = \tfrac{1}{2} \sum_{n=1}^{N} \left| p_{in} - p_{jn} \right| \qquad \textbf{(3.5)}$$

Esta distância forma uma métrica porque se baseia na norma L_1 aplicada a diferenças entre linhas da matriz **P** com elementos p_{ij} definidos em (3.2).

Na literatura encontram-se várias propostas de medidas de semelhança, [52][153][167], que como referido em [154], não cumprem as propriedades de uma métrica. Na medida proposta em (3.5), d_1 cumpre as 3 propriedades de uma métrica:

é positiva,
é simétrica e
respeita a desigualdade triangular

Outras distâncias podem ser definidas com base no mesmo princípio, nomeadamente a distância Euclidiana entre linhas,

$$d_2(c_i, \hat{c}_j) = \sqrt{\sum_{n=1}^{N} \left(p_{in} - p_{jn}\right)^2} \qquad \textbf{(3.6)}$$

Apesar de, numa tarefa de reconhecimento, os erros incluírem não só confusões, mas também erros de inserção e apagamento, estes não foram incluídos na medida de distância por se considerar que se relacionam com a qualidade dos modelos acústicos e não com a semelhança entre esses mesmos modelos.

3.2.2. ALGORITMO DE AGRUPAMENTO

O método de agrupamento que se propõe, junta os fones segundo uma hierarquia multinível, onde *clusters* de um nível são agrupados para formar um novo *cluster* no nível seguinte, num formato semelhante ao apresentado na Figura 3.4.

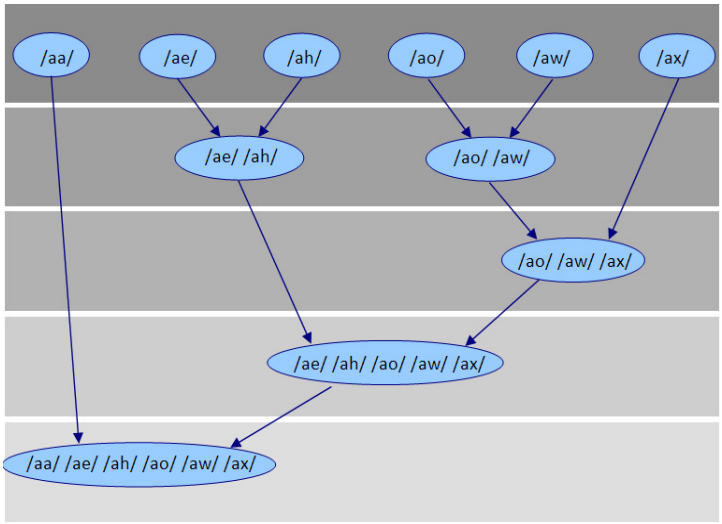


Figura 3.4: Agrupamento hierárquico em níveis.

O agrupamento de fones pode ser feito seguindo os passos descritos no quadro seguinte, considerando que, inicialmente, cada fone constitui um *cluster* distinto.

Passo 1→Calcular a matriz **P** usando (3.2).

Passo 2→Encontrar todas as distâncias entre pares de fones usando (3.5)

Passo 3→Calcular a distância entre todos os *clusters*. A distância entre o *cluster r* e *s* pode ser calculada seguindo vários critérios, sendo o mais simples o *nearest neighbour*:

$d(r,s) = \min\left(d\left(c_{i|r}, c_{j|s}\right)\right), i = 1...n_r; j = 1..n_s$ onde n_k é o número de fones no *cluster k* e $c_{i|k}$ é o fone de ordem *i* no *cluster k*.

Passo 4→Criar um novo *cluster* agrupando os dois *clusters* mais próximos.

Os Passos 3 e 4 serão repetidos até se obter o número de *clusters* desejado ou até que um dado limiar seja atingido. Outra possibilidade é agrupar *clusters* iterativamente até que todos os fones pertençam ao mesmo *cluster*. A partir daqui pode construir-se um dendrograma, permitindo a decisão do nível ou da escala de agrupamento que é mais adequada para a aplicação.

3.2.3. RESULTADOS

A análise e teste da proposta de *clustering*, parte de uma matriz de confusão entre fones. Esta matriz pode conter resultados ao nível da *frame* (caso se usem classificadores baseados em ANN, SVM ou outros) ou ao nível do segmento (caso se use reconhecimento de fones com HMMs ou sistemas híbridos).

Inicialmente o estudo centrou-se em matrizes de confusão calculadas com resultados ao nível da *frame*. Foi usada uma rede neuronal com 39 parâmetros de entrada (sinal de fala analisado a cada 10ms usando uma janela de Hamming de 25ms), representando 12 MFCC, energia e suas 1ª e 2ª derivadas. A janela de contexto usada foi de 170ms, usando somente 9 *frames* (uma sim uma não), seguindo o método descrito na secção 4.3.2. A rede apresenta 61 saídas sendo a camada intermédia constituída por 1000 unidades. No total a rede é definida com 413k parâmetros.

A matriz de confusão gerada a partir dos resultados de saída desta rede neuronal é uma matriz que fornece informação de FER (*frame error rate*). Como as saídas da rede são estimativas da probabilidade das classes para a *frame* em questão, toma-se para esta *frame* a classe correspondente ao maior valor das saídas da rede (fone mais provável). A partir daqui é possível calcular quantas vezes é que uma *frame* da classe *i* foi classificada como

classe *j*. Os resultados envolvem a classificação de mais de 10^6 *frames*, parte delas relacionadas com a classe da TIMIT h# (etiqueta atribuída sempre no início e fim de cada locução; o início tem muitas vezes ruído de aspiração). A confusão desta classe com as restantes é enorme mascarando as restantes confusões.

Foi ainda estudada outra abordagem. Em vez de tomar uma decisão sobre qual a classe vencedora (situação em que a célula correspondente à classe vencedora na matriz de confusão é incrementada de uma unidade), foram colocados na matriz de confusão os próprios valores à saída da rede neuronal (probabilidades de cada fone). No entanto, o que se verifica é que, em situações de erro, não há correlação entre o valor da saída vencedora e o fone de referência, estando o valor da saída vencedora longe da unidade (vence com pouca certeza, ou seja, todos os fones têm baixa probabilidade). Usando todos os pares de valores de saída da rede (*target*, saída vencedora) nos casos em que os índices destes não coincidem (caso de erro) verificou-se que o valor das saídas vencedoras pode ser representado por uma Gaussiana centrada em 0.5 com desvio padrão perto da unidade. Significa que em caso de erro a saída vencedora tem valor perto de 0.5, mas a do *target* pode assumir qualquer valor abaixo.

Face às dificuldades impostas pelo recurso a matrizes de confusão compostas por resultados ao nível da *frame*, recorreu-se aos resultados de reconhecimento de um sistema híbrido MLP/HMM. Nesta situação os resultados são ao nível do segmento, o que no conjunto de treino da TIMIT envolve 143k segmentos. Assim, usaram-se modelos HMM treinados usando ferramentas do HTK3.4, [162] de forma a estimar as probabilidades de transição entre estados. Cada fone foi modelado por um modelo de 3 estados esquerda-direita com uma Gaussiana. No sistema híbrido as probabilidades *a priori* associadas aos estados são substituídas por probabilidades *a posteriori* dadas pelas predições de saída da rede neuronal. Os três estados dos HMM partilham a mesma saída do MLP. De forma a substituir as tradicionais Gaussianas pelas saídas da rede neuronal, foi usado o HTK com algumas alterações. Para obter a matriz de confusão usou-se o conjunto completo de treino da TIMIT e a medida de avaliação com alinhamento temporal, AAT, descrita na secção 2.3.3, mas atribuindo grande penalização às inserções (p_{INS}=12) e apagamentos (p_{DEL}=12), potenciando assim as confusões.

Seguindo os passos descritos na secção 3.2.2 chegamos a uma estrutura de *clustering*, mostrada na Figura 3.5, representada como uma árvore binária (dendrograma).

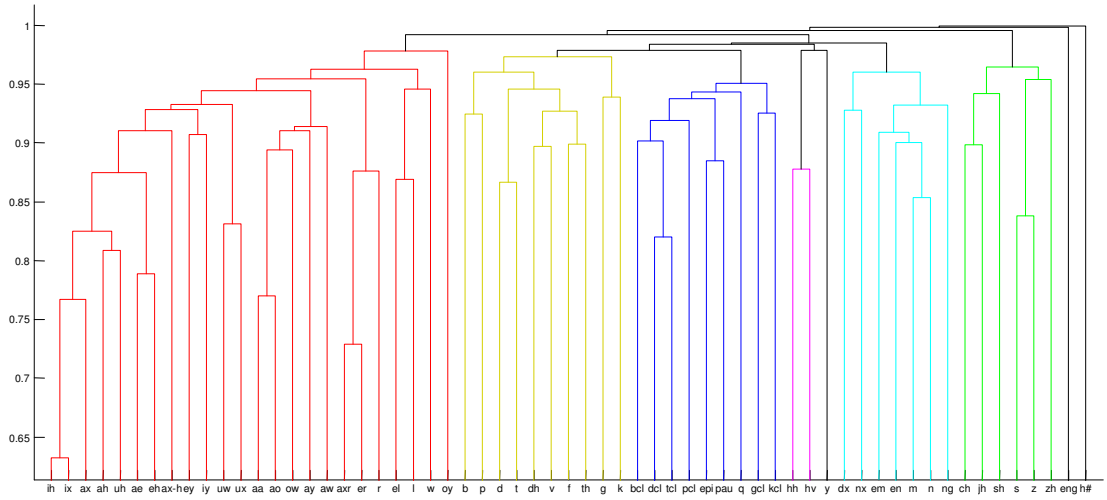


Figura 3.5: Dendrograma de agrupamento hierárquico de fones para a TIMIT usando a distância *average* entre *clusters*. Coeficiente de correlação cofenética =0.873.

As etiquetas do eixo horizontal representam os fones originais e o eixo vertical refere-se à distância entre os fones. Esta distância é calculada a partir da medida de semelhança usando $d_1(c_i, c_j)$ dada por (3.5). O número de *clusters* depende do limiar de corte da árvore. No exemplo dado na Figura 3.5 os 61 fones da TIMIT são representados pelos 9 *clusters* apresentados na Tabela 3.1.

É notória a consistência dos *clusters* (CFA), nomeadamente os que envolvem vogais. Se compararmos o *cluster* das vogais (CFA 6) com a divisão *knowledge-based* da Tabela 3.2 apresentada em baixo, diferem somente no fone [y]. Este fone isolou-se, não porque seja acusticamente diferente mas devido às poucas confusões que têm com os restantes fones. O *cluster* forte que corresponde às nasais (CFA 5) é também semelhante à divisão *knowledge-based*. O fone [eng] isolou-se devido à sua baixa ocorrência e consequente baixo número de confusões com outros fones. O método automático de *clustering* dá indicação que as africadas[17] apresentam semelhanças com as fricativas [s], [sh], [z], [zh]. As divisões *knowledge-based* por vezes colocam as africadas junto com as plosivas e outras vezes com as fricativas. Em relação às fricativas e plosivas o método origina mais que 2 *clusters* (CFA 2 e CFA 7). O número de confusões entre algumas fricativas e plosivas sugere que acusticamente apresentam semelhanças.

O coeficiente de correlação cofenética, referido na legenda da Figura 3.5, é uma medida da distorção produzida pela redução dimensional obtida no agrupamento. Este coeficiente é uma correlação matricial entre a matriz de similaridade original e a matriz de similaridade

[17] Fricativas com oclusão inicial.

derivada do dendrograma, [146]. Quanto mais perto este valor estiver da unidade mais precisa é a solução apresentada. No caso do dendrograma da na Figura 3.5 o coeficiente de correlação cofenética obtido foi 0.873, o que demonstra consistência dos *clusters* encontrados.

Classes fonéticas alargadas	**Fones da TIMIT**
CFA 1	bcl dcl epi gcl kcl pau pcl q tcl
CFA 2	b d dh f g k p t th v
CFA 3	y
CFA 4	hh hv
CFA 5	dx em en m n ng nx
CFA 6	aa ae ah ao aw ax ax-h axr ay eh el er ey ih ix iy l ow oy r uh uw ux w
CFA 7	ch jh s sh z zh
CFA 8	eng
CFA 9	h#

Tabela 3.1 Resultado da divisão dos 61 fones da TIMIT em termos de 9 *clusters* usando o método *confusion-driven*.

Classes fonéticas alargadas	**Fones da TIMIT**
Soantes	aa ae ah ao ax ax-h axr ay aw eh el er ey ih ix iy l ow oy r uh uw ux w y
Plosivas	p t k b d g jh ch
Fricativas	s sh z zh f th v dh hh hv
Nasais	m em n nx ng eng en
Silêncios	h# epi pau bcl dcl gcl pcl tcl kcl q dx

Tabela 3.2 Resultado da divisão dos 61 fones da TIMIT em termos de 5 classes alargadas usando o método *knowledge-driven*, proposta por Scanlon em [132].

Apesar das classes geradas pelo método automático parecerem consistentes com os princípios subjacentes à divisão feita com base no conhecimento humano, interessa-nos essencialmente averiguar se estas classes podem ser usadas em sistemas de reconhecimento automático de fala, alcançando o mesmo desempenho que as propostas por peritos. Nesse sentido, foram treinadas, separadamente, duas redes MLP, com o mesmo número de parâmetros e com mesmo número de camadas. As redes obedecem a uma estrutura hierárquica onde as primeiras camadas são treinadas com um conjunto menor de CFA. À medida que o número de camadas aumenta, aumenta também o número de classes alargadas alcançando o detalhe do fone na última camada. As predições dos primeiros classificadores são usadas nos classificadores seguintes. Um exemplo de uma rede semelhante é apresentado na Figura 4.7 (página 102). A rede hierárquica treinada com classes alargadas geradas pelo método *confusion-driven* usa as classes apresentadas na Tabela 3.3, enquanto a rede hierárquica treinada com CFA formadas a partir do conhecimento fonético usa as classes apresentadas na Tabela 3.4.

Apesar das camadas intermédias terem um número diferente de saídas (na rede *confusion-driven* 9, 16, 40 e na *knowledge-based* 5, 12, 34), em ambas, a camada de saída é treinada em função dos 61 fones originais da TIMIT. Na Figura 3.6 apresentam-se os resultados da comparação entre os dois sistemas em termos de FER. O número de *frames* erradas, do conjunto completo de teste, é muito semelhante em ambos os sistemas, o que é um bom indicador para que o método de agrupamento proposto seja usado em substituição de técnicos especializados na divisão fonética.

9 classes alargadas	16 classes alargadas	40 classes alargadas
l, el, w, r, er, axr, ey, aw, ay, iy, ih, eh, ae, ah, ax, uh, ix, uw, ux, ax-h, aa, ao, ow, oy	l, el, w	l, el
		w
	r, er, axr	r, er, axr
	ey, aw, ay, iy, ih, eh, ae, ah, ax, uh, ix, uw, ux, ax-h, aa, ao, ow	ey
		aw
		ay
		iy
		ih, eh, ae, ah, ax, uh, ix
		uw, ux
		ax-h
		aa, ao, ow
	oy	oy
p, b, d, t, f, th, v, dh, k, g	p, b	p
		b
	d, t, f, th, v, dh	d, t
		f , th
		v, dh
	k, g	k
		g
jh, ch, sh, z, s, zh	jh, ch, sh	jh, ch
		sh
	z, s, zh	z, s
		zh
hh, hv	hh, hv	hh, hv
m, n, em, ng, en, nx, dx	m, n, em, ng, en	m , n
		em
		ng
		en
	nx, dx	nx
		dx
pcl, tcl, dcl, kcl, q bcl, gcl, pau, epi	pcl, tcl, dcl, kcl, q bcl, gcl, pau, epi	pcl
		tcl, dcl
		kcl
		q
		bcl
		gcl
		pau, epi
y	y	y
eng	eng	eng
h#	h#	h#

Tabela 3.3: Descrição das classes alargadas resultantes da aplicação do método confusion-driven.

5 classes alargadas	12 classes alargadas	34 classes alargadas	Fones da TIMIT
Vogais	Vogais	v1	iy
		v2	uh uw ux
		v3	ax ax-h ah
		v4	ix ih
		v5	aa ao
		v6	eh
		v7	ae
	Ditongos	d1	ey
		d2	aw
		d3	ay
		d4	oy
		d5	ow
	Semi-Vogais	sv1	r w y
		sv2	l el
		sv3	er axr
Plosivas	Plosivas -V	stV	b d g
	Plosivas –uV	stuV	p t k
	Africativas	afr	jh ch
Fricativas	Fricativas-V	fV1	z
		fV2	zh
		fV3	v dh
	Fricativas-uV	fuV1	s
		fuV2	sh
		fuV3	f th
	Sussurros	w	hh hv
Nasais	Nasais	n1	en n nx
		n2	m em
		n3	ng eng
Silêncios	Silêncios	sil1	h#
		sil2	pau epi
	Oclusivas	vcl	bcl dcl gcl
		uvcl	pcl tcl kcl
		cl1	dx
		cl2	q

Tabela 3.4 Descrição das classes alargadas resultantes da aplicação do método knowledge-driven.

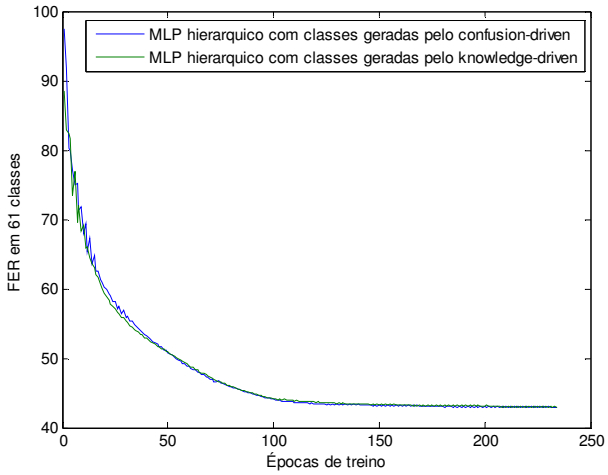


Figura 3.6 Comparação, em termos de FER de duas redes MLPs semelhantes. Uma treinada usando as classes alargadas obtidas pelo método de agrupamento proposto e outra usando as divisões feitas por peritos humanos.

3.3. EXPERIÊNCIAS EM DETEÇÃO DE CLASSES FONÉTICAS ALARGADAS

Tal como referido, vários autores já se dedicaram à tarefa de detetar classes fonéticas alargadas (eventos) usando HMMs [81][85], SVMs [55], ANNs [81] e também arquiteturas híbridas [14]. Contudo, não é do nosso conhecimento que haja algum estudo que os avalie e compare. Assim, o objetivo desta secção é comparar o desempenho de sistemas de deteção de eventos usando HMMs, SVMs e ANNs e enfatizar as abordagens que conduzem a taxas de erro inferiores.

À semelhança dos testes anteriores, todas as experiências foram feitas usando a base de dados TIMIT, [36]. O desempenho dos sistemas é avaliado em termos de *Correctness* (*Corr*), *Accuracy* (*Acc*), e taxas de erro de eventos (EER- *Event Error Rate*) usando respetivamente (2.31), (2.32) e *EER*=1-*Acc*. Os resultados foram calculados usando a ferramenta `HResults` do HTK, [162] e o método AAT descrito na secção 2.3.3. Neste último avalia-se também a qualidade das marcas temporais através da taxa de concordância – *Agreement* – dada por (2.33).

Serão descritos e comparados quatro sistemas de detecção de eventos. O primeiro é baseado em HMMs tradicionais e os restantes são baseados em arquiteturas híbridas: SVM/HMM, SVM/NMD e ANN/HMM. Os sistemas fazem uma segmentação das locuções em termos de uma sequência de eventos (CFA) contínua e sem sobreposição. Nesse sentido, serão detetadas 4 classes de eventos: silêncios, fricação, plosivas e soantes, de acordo com a divisão apresentada na Tabela 3.5. Estas classes foram definidas seguindo uma abordagem *knowledge-driven* e seguem a divisão proposta por Juneja em [55].

Eventos (CFA)	Fones da TIMIT
Soantes	aa, ae, ah, ao, ax, ax-h, axr, ay, aw, eh, el, er, ey, ih, ix, iy, l, ow, oy, r, uh, uw, ux, w, y, dx, em, en, eng, hh, hv, m, n, ng, nx,
Plosivas	p, t, k, b, d, g, jh, ch,
Fricativas	s, sh, z, zh, f, th, v, dh
Silêncios	h#, epi, pau, bcl, dcl, gcl, pcl, tcl, kcl, q

Tabela 3.5: Divisão dos 61 fones da TIMIT em 4 classes de eventos.

3.3.1. DETETOR DE EVENTOS BASEADO EM HMMS

De forma a construir um classificador de eventos baseado em HMMs, foram treinados modelos para cada classe: soantes, plosivas, fricativas e silêncios, usando o HTK, [162]. Cada classe foi modelada por um HMM com 3 estados e topologia esquerda-direita. Os parâmetros de entrada são 12 MFCC e energia e suas 1ª e 2ª derivadas e o critério de treino usado foi ML. O reconhecimento foi feito sem recurso a qualquer modelo de linguagem. Eventos adjacentes da mesma classe são unidos. Os resultados, usando o `HResults` são apresentados na Tabela 3.6. Usando só uma Gaussiana, a taxa *Correctness* alcança os 88.47% e a taxa *Accuracy* os 73.14%. Com 8 misturas os resultados sobem para 89.36%, 77.57% para *Correctness* e *Accuracy*, respetivamente, contudo o número de parâmetros de treino também aumenta, naturalmente. Apesar de algumas diferenças entre as condições de treino e teste, estes resultados estão muito acima dos resultados apresentados por Juneja em [55], que para um classificador de 5 classes de eventos obteve 69.6% para *Correctness* e 64,9% para *Accuracy* usando modelos com 8 misturas por estado.

A tabela inclui igualmente os resultados obtidos pelo AAT em termos de *Correctness*, *Accuracy* e *Agreement* num intervalo de 20ms. Em termos de alinhamento, e considerando uma mistura Gaussiana, 84.37% das marcas foram corretamente detetadas com uma distância inferior a 20ms das marcas manuais da TIMIT. Os resultados de *Correctness* e *Accuracy* são naturalmente inferiores aos obtidos pelo `HResults` uma vez que esse método só permite considerar um acerto quando existe uma coincidência temporal entre as sequências.

Classificador HMM	*Correctness*	*Accuracy*	EER	*Agreement* (20ms)	Número de parâmetros de treino
1 mistura Gaussiana	88.47%	73.14%	26.86%	--	979
	86.69%	68.79%	31.21%	84.37%	
8 misturas Gaussianas	89.36%	77.57%	22.43%	--	7615
	88.08%	73.87%	26.13%	75.37%	

Tabela 3.6: Resultados do detetor de eventos baseado em HMMs avaliado usando:

- `HResults` do HTK;
- Avaliação com Alinhamento Temporal (AAT; secção 2.3.3).

3.3.2. DETETOR DE EVENTOS BASEADO EM SVMS

Na solução do problema da deteção de um conjunto de eventos usando SVMs, recorreu-se ao treino de um classificador binário para cada classe a detetar (4 classes), seguindo uma abordagem um-contra-todos ("*one-versus-all*"). Cada classificador concorre uma classe contra as restantes três, num modelo semelhante ao mostrado na Figura 3.7.

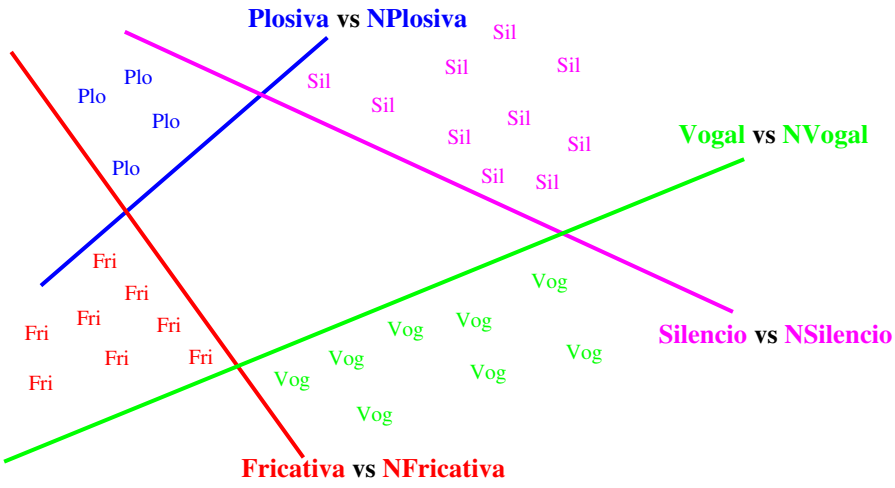


Figura 3.7 : Exemplo da configuração um-contra-todos.

Cada classificador é treinado individualmente usando a *package* de *software* SVM^{light}, [53]. Os *targets* são definidos recorrendo à segmentação manual da TIMIT. Se a *frame* pertence à classe a treinar, tem *target* y_i=1, caso contrário tem *target* y_i= -1.

Na Tabela 3.7 mostram-se os parâmetros de entrada usados no treino de cada classificador SVM. Ao contrário da abordagem seguida nos HMMs, onde é feita uma modulação temporal das características de entrada, as SVMs operam em regime estático: só são usados como parâmetros de entrada parâmetros que descrevam a *frame* atual. Todos os classificadores SVM usam somente 4 parâmetros acústicos estáticos, exceto o classificador de plosivas que usa adicionalmente 8 características dinâmicas (1ª e 2ª derivadas das características estáticas), alargando o contexto a 9 *frames*. Para evitar um desequilíbrio entre parâmetros no cálculo das distâncias, devido a diferentes gamas dinâmicas, o vetor de parâmetros foi normalizado por média e variância através de $x_{norm} = \frac{x - mean(x)}{\sqrt{\text{var}(x)}}$.

	Fricativas	Silêncios	Plosivas	Soantes
5 ms log-energia	×	×	×	
Amplitude máxima		×		
Spectral Flatness Measure	×	×	×	
Centróide Espectral	×			
Ratio de log energia altas/baixas frequências	×	×	×	
Mediana da energia de um banco de filtros			×	
Energia <500Hz				×
500<Energia <1500Hz				×
Evidência de Tom				×
Peakiness				×

Tabela 3.7: Parâmetros de entrada usados no treino das SVMs por classe de eventos.

Estes pequenos conjuntos de parâmetros parecem caracterizar bem cada classe e conduzem a um número de vetores de suporte reduzido e consequentemente a um tempo de classificação muito baixo.

Para cada *frame*, é fornecido um par (x_i, y_i) para treino, sendo *x* os parâmetros e *y* o *target*. É comum usar-se nas SVMs *kernels* não lineares, [56][135]. No entanto, e apesar das comprovadas capacidades de classificação, a carga computacional associada a este tipo de *kernel* é bastante elevada, especialmente na fase de classificação quando o número de vetores de suporte é elevado, e por isso optou-se por usar um kernel linear. Como referido, *C* é a penalização associada a pontos de treino erróneos e que não podem ser classificados corretamente. Como os parâmetros estão normalizados, a escolha do *C* ótimo foi feita partindo de *C*=1 e fazendo uma procura em grelha à volta deste valor usando subconjuntos do treino e do teste (*frames* de 100 locuções).

Como estamos perante conjuntos desequilibrados das duas classes (a classe negativa tem muitos mais exemplares que a positiva) introduz-se um fator de custo para ajustar o número de falsos positivos e falsos negativos. Este fator, *j* determina a relação entre o número de exemplos de treino da classe positiva em relação à classe negativa. A importância de definir este valor está relacionada com o facto de a menor classe poder ser sobreamostrada em relação à maior. Para resolver este problema surgem fatores de custo *C+* e *C-*, que ajustam o número de falsos positivos (amostras -1 classificadas como +1) e falsos negativos, passando a função objetivo a minimizar a ser $J = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C_+ \sum_{i=y_{i=1}} \xi_i + C_- \sum_{i=y_{i=-1}} \xi_i$.

Passando para o problema dual é preciso calcular a relação entre *C+* e *C-* e não as quantidades em separado. No nosso caso, e para encontrar estes valores, foi seguido o modelo de custo [83] onde:

$$\begin{cases} L(+1) = c^{-}\pi_s^{-}\pi^{+} \\ L(-1) = c^{+}\pi_s^{+}\pi^{-} \end{cases} \text{ sendo } \begin{cases} c^{-} \rightarrow \text{custo dos falsos negativos} \\ c^{+} \rightarrow \text{custo dos falsos positivos} \\ \pi^{+} \text{ e } \pi^{-} \rightarrow \text{proporção populacional} \\ \pi_s^{+} \text{ e } \pi_s^{-} \rightarrow \text{proporção nos conjuntos de treino} \end{cases}$$

Veja-se, a título de exemplo, o cálculo para o evento fricativas:

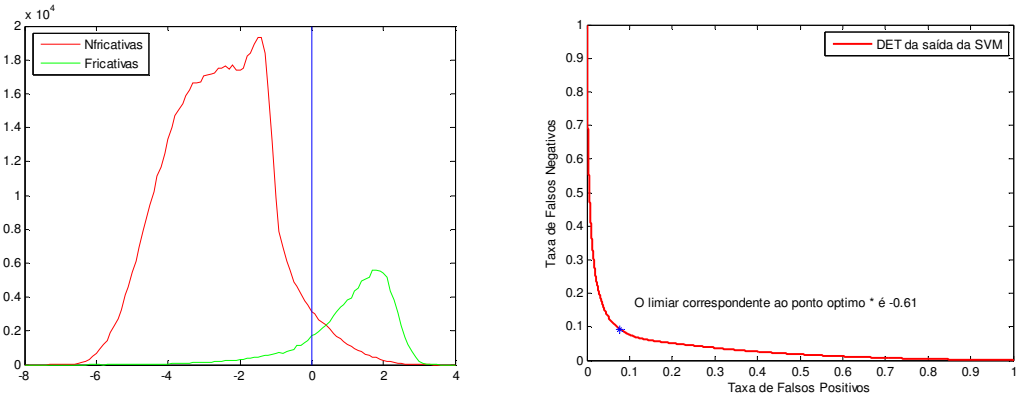
$$\begin{cases} c^{-} = c^{+} \\ \pi^{+} = 11\% \\ \pi^{-} = 89\% \\ \pi_s^{+} = 15\% \\ \pi_s^{-} = 85\% \end{cases} \Rightarrow \begin{cases} L(+1) = c^{-} \times 0.85 \times 0.11 = 0.0935 \times c \\ L(-1) = c^{+} \times 0.15 \times 0.89 = 0.1685 \times c \end{cases}$$

No SVM$^{\text{light}}$ o fator de custo é escolhido de forma a que o potencial custo dos falsos positivos iguale o dos falsos negativos. O que significa que $\frac{C^{+}}{C_{-}} = \frac{L(-1)}{L(1)} = 1.8$. Por outras palavras: os erros gerados pelos exemplos de treino negativos dominariam o erro total, uma vez que são em maior número. Assim, admite-se um erro para a classe negativa 1.8 vezes maior que para a classe negativa. Os valores destes parâmetros de ajuste por classe constam da Tabela 3.8 bem como as estatísticas de treino e teste (em %).

	C	***j***	**Treino**	**Resultados por *frame***	
			XiAlpha-error	**Acc**	**Prec / Recall**
Fricativas	1	1.8	≤ 22.35	92.75	78.01 / 78.21
Silêncios	0.01	1.4	≤ 15.69	94.09	86.6 / 88.72
Plosivas	1	3	≤ 20.25	94.84	52.40 / 30.28
Soantes	0.01	-	≤ 9.51	92.21	91.56 / 94.13

Tabela 3.8: Resultados de treino dos classificadores SVM e parâmetros usados.

Os resultados apresentados são por *frame* e referentes à deteção de cada uma das classes em separado. São resultados animadores uma vez que se conseguiu encontrar um bom compromisso entre as falsas aceitações e rejeições. Novamente, no caso do detetor de fricativas, veja-se esta relação bem como a DET (*Detection Error Tradeoff*), Figura 3.8 b).



a) b)

Figura 3.8: Relação entre falsas aceitações e falsas rejeições bem como a curva DET para o classificador de fricativas. a) Histogramas em função do valor de predição da SVM; b) curva DET indicando o ponto ótimo (menor distância à origem).

De forma a permitir uma comparação dos resultados com os do sistema baseado em HMMs, foram combinados os valores de predição[18] destes quatro classificadores permitindo, assim, obter um sinal segmentado em termos de eventos. O sistema é ilustrado na Figura 3.9.

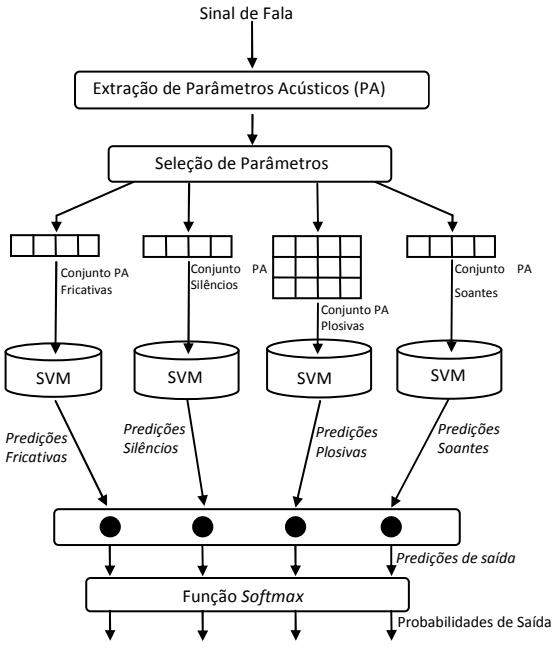


Figura 3.9: Detetor de eventos baseado em SVMs.

Depois de fazer a extração dos parâmetros acústicos, estes são organizados em 4 conjuntos, em função da classe em questão (Tabela 3.7). Cada conjunto de parâmetros alimenta um

[18] Os valores de predição são os valores de saída de uma SVM. Não são valores probabilísticos. Correspondem a uma medida da distância de uma amostra ao hiperplano separador no espaço das características.

classificador SVM, cujas saídas fornecem, para cada *frame*, predições para as quatro classes de eventos (cada classificador SVM é responsável pela deteção de um dado evento). Uma vez que as SVM não fornecem valores de saída interpretáveis como probabilidades *a posteriori*, as predições foram normalizadas por uma função *softmax*, garantindo assim, que a cada classe está associado um número entre 0 e 1 e que a soma de todas as saídas perfaz 1. A decisão de qual a classe correspondente a uma dada *frame* é tomada com base no maior valor de predição, i.e., seguindo a estratégia "*winner-takes-all*".

Sendo os resultados *frame* a *frame*, há muitas flutuações (ex: aparece uma *frame* isolada identificada como vogal no meio de um conjunto de *frames* identificadas como silêncio). Para reduzir estas flutuações, as predições de cada classificador são suavizadas ao longo do tempo. Esta suavização é feita através de uma difusão dinâmica usando a difusão anisotrópica de Perona-Malik, [113]. Um segmento é então classificado como pertencente a uma classe *x* quando nesse período a saída do classificador *x* se mantiver superior à dos restantes. Os resultados desta abordagem encontram-se na Tabela 3.9, e não foram muito animadores. Estes resultados estão bastante abaixo dos alcançados pelo sistema HMM e motivaram-nos para a construção de modelos híbridos. Foram testados dois sistemas híbridos: SVM/NMD e SVM/HMM que se descrevem de seguida.

Híbrido SVM/NMD

De forma a transformar as saídas das SVMs, que operam ao nível da *frame*, num sinal segmentado em termos de eventos é necessário um "conversor" *frame*–segmento. Nesta secção estuda-se a aplicação do método *Non-Negative Matrix Deconvolution* (NMD) a este problema numa filosofia elucidada através da Figura 3.10.

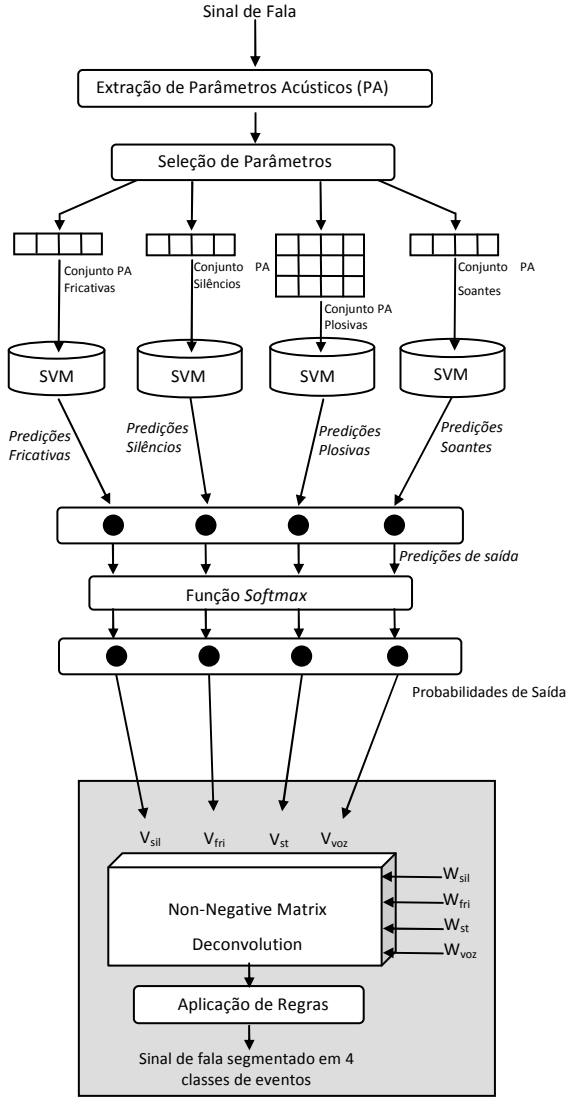


Figura 3.10: Detetor de eventos híbrido SVM/NMD.

Como descrito na secção 2.2.5, o NMD expõe a estrutura temporal dos elementos de entrada. A coluna *i* de $\mathbf{W}^{(t)}$ descreve o espetro do elemento *i, t* tempos depois de este ter tido início. Pretendemos que o NMD detete quatro classes de eventos a partir das saídas normalizadas das SVMs. O NMD funciona em blocos, e por isso foram testados vários tamanhos para as bases e obtiveram-se bons resultados recorrendo a bases em blocos de 4 *frames* (*T*=3). A Figura 3.11 apresenta um pequeno exemplo. A imagem de cima representa o espetrograma de um sinal de fala composto por vários eventos que se repetem ao longo do tempo, enquanto a imagem central representa as saídas do classificador com SVMs. As bases são mostradas à esquerda, seguindo a ordem: silêncios, fricativas, plosivas e soantes. As linhas de **H** (apresentadas no gráfico de baixo) indicam temporalmente o local onde as bases ocorrem.

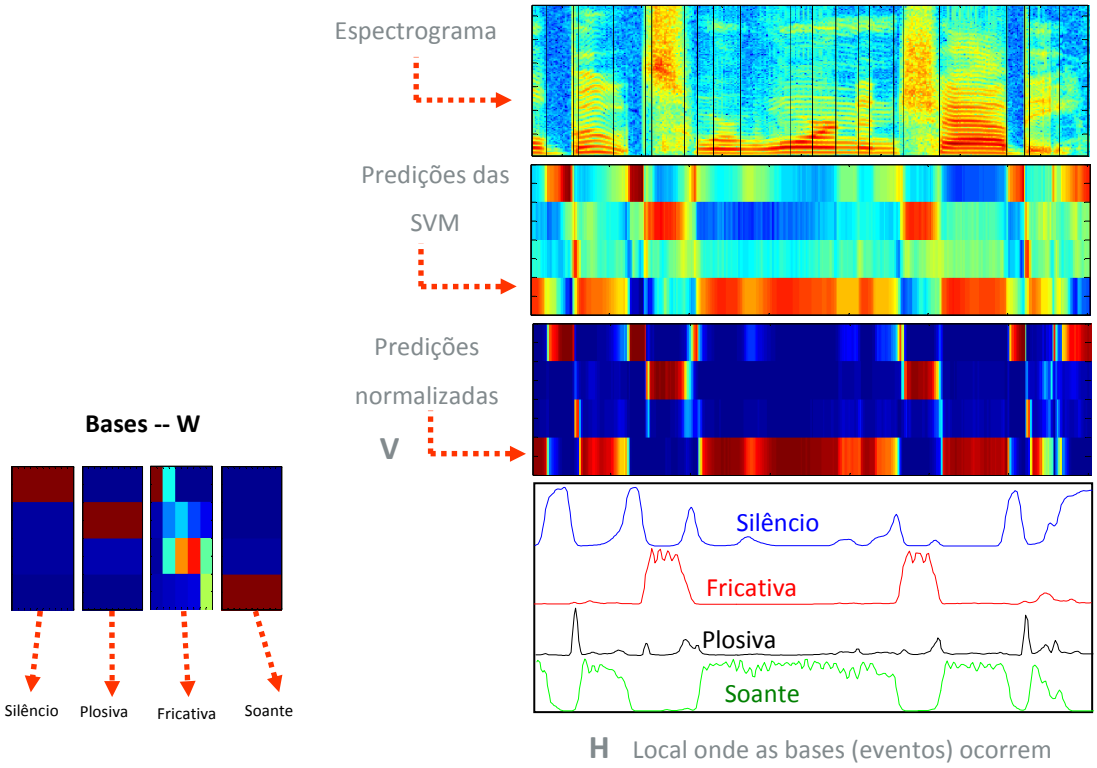


Figura 3.11: Exemplo da deteção de eventos pelo detetor híbrido SVM/NMD.

Neste exemplo todos os objetos foram corretamente detetados. Contudo, há erros de inserção que persistem. Estes erros foram posteriormente reduzidos aplicando regras simples inferidas a partir da análise da sequência de eventos e da duração de alguns eventos. Regras como apagar eventos da classe soantes muito pequenos ou analisar situações improváveis (ocorrência da sequência silêncio→fricativa→silêncio ou sonante→plosiva→silêncio) melhoraram o desempenho do detetor de eventos.

Os resultados desta abordagem são apresentados na Tabela 3.9. Usando o `HResults` a taxa *Correctness* alcançou 88.86% e *Accuracy* 74% resultados estes muito acima dos do classificador SVM e também acima dos do HMM. Comparando com este último a melhoria relativa em termos de EER foi de 3.24%. No entanto, se a avaliação for feita com o AAT, o método apresenta taxas inferiores às do classificador HMM, o que é facilmente justificado pelo facto de os eventos plosivas serem de muito curta duração e muitas vezes terem pouca sobreposição com a referência. A qualidade dos limites temporais é, como esperado, muito boa e bastante acima de qualquer um dos restantes métodos em comparação.

	Correctness	*Accuracy*	EER	*Agreement* (20ms)
Classificador HMM	88.47	73.14	26.86	--
	86.69	68.79	31.21	84.37
Classificador SVM	85.22	70.88	29.12	--
	82.86	64.39	35.61	84.34
Classificador híbrido SVM / NMD	88.86	74.01	25.99	--
	85.60	66.55	33.45	86.90

Tabela 3.9: Resultados do detetor de eventos baseado em SVM/NMD avaliado usando:

- ☐ HResults do HTK;
- ▩ Avaliação com Alinhamento Temporal (AAT; secção 2.3.3).

Híbrido SVM/HMM

O sistema híbrido SVM/HMM combina a estrutura típica dos HMMs com as predições ao nível da classe dos eventos fornecidas pelos classificadores SVMs. O sistema segmenta uma locução de entrada em termos de uma sequência das quatro classes de eventos em questão. Usa modelos de Markov para modelar temporalmente a sequência de eventos predita pelas SVM, como probabilidades associadas aos estados dos HMMs. De acordo com (2.29), as predições das SVMs normalizadas são usadas em substituição das misturas Gaussianas usadas nos HMMs. A matriz de transição usada corresponde a modelos com uma mistura obtida a partir do detetor descrito na secção 3.3.1. O teste foi feito recorrendo à ferramenta `HVite` do HTK, [162] com algumas alterações de forma a permitir a substituição dos modelos de misturas Gaussianas pelas probabilidades *a posteriori* (tal como no caso dos modelos híbridos ANN/HMM). Os resultados são apresentados na Tabela 3.10 e mostram que, com o sistema híbrido, a taxa *Correctness* cai comparando quer com o sistema HMM quer com o SVM puro. Contudo, a melhoria de *Accuracy* é expressiva, e sendo esta medida mais precisa, uma vez que considera todos os erros (*Correctness* não considera os erros de inserção), leva-nos a considerar o método como bem-sucedido na tarefa. Merece-nos também atenção a melhoria alcançada no que se refere à qualidade das marcas de segmentação encontradas pelo híbrido, cuja melhoria relativa ronda os 2.8% comparando com a segmentação do HMM.

	Correctness	*Accuracy*	EER	*Agreement* (20ms)
Classificador HMM	88.47	73.14	26.86	--
	86.69	68.79	31.21	84.37
Classificador SVM	85.22	70.88	29.12	--
	82.86	64.39	35.61	84.34
Classificador híbrido SVM / NMD	88.86	74.01	25.99	--
	85.60	66.55	33.45	86.90
Classificador híbrido SVM / HMM	81.30	74.77	25.23	--
	79.30	69.94	30.06	86.70

Tabela 3.10 – Resultados, do detetor de eventos baseado em SVM/HMM avaliado usando:

- ☐ HResults do HTK;
- ▒ Avaliação com Alinhamento Temporal (AAT; secção 2.3.3).

Os resultados apresentados mostram que o desempenho de um sistema de deteção de eventos de fala pode ser melhorado usando um sistema híbrido SVM/HMM. Os resultados são interessantes especialmente face ao baixo número de parâmetros usado no treino das SVMs se compararmos com o vetor de dimensão 39 usado nos HMMs.

3.3.3. Detetor de Eventos Baseado em ANNs

O sistema proposto funciona de forma análoga ao descrito na secção 3.3.2: faz uma segmentação das locuções em termos de um conjunto de quatro eventos usando, numa primeira etapa, um MLP como classificador de *frames* e, numa segunda etapa, um HMM como modelador temporal e segmentador tal como mostrado na Figura 3.12. O sistema consiste numa rede MLP com 4 conjuntos de entrada que ligam a 4 camadas intermédias distintas e uma camada de saída treinada para classificação de *frames* de 4 classes de eventos. Como se pretende comparar o desempenho deste sistema com o híbrido SVM/HMM, foram usados os mesmos parâmetros de entrada deste último e que constam da Tabela 3.7. Cada camada intermédia é composta por 20 nodos escondidos, exceto a camada de plosivas que tem 24 nodos. A camada de saída tem 4 nodos, correspondendo às quatro classes a detetar. De forma a ser comparável com o híbrido SVM/HMM, o treino da rede foi feito sem incluir contexto. Para que os valores de saída da última camada possam ser interpretados como probabilidades *a posteriori* de cada classe, usou-se a função *softmax* como função de ativação na camada de saída. Todos os pesos e *bias* da rede foram ajustados

usando treino *batch* e o algoritmo *resilient back-propagation* [121] de forma a minimizar o erro da *minimum-cross-entropy* entre as saídas da rede e os *targets* correspondentes. A rede tem 952 parâmetros de ajuste e o sistema híbrido adiciona-lhe mais 24 (6 parâmetros a_{ij} para cada um dos 4 modelos HMM cada um com 3 estados).

O treino global da rede foi feito com base em ferramentas desenvolvidas especificamente para o efeito, não recorrendo a nenhum pacote de *software* disponível. O investimento no desenvolvimento destas ferramentas traduziu-se numa mais-valia dando-nos liberdade de propor arquiteturas de rede e treino alternativas. Exemplo disso é o uso de *targets* em camadas escondidas, que se veio a revelar como uma boa contribuição na melhoria do reconhecedor de fones e que se descreve no capítulo seguinte.

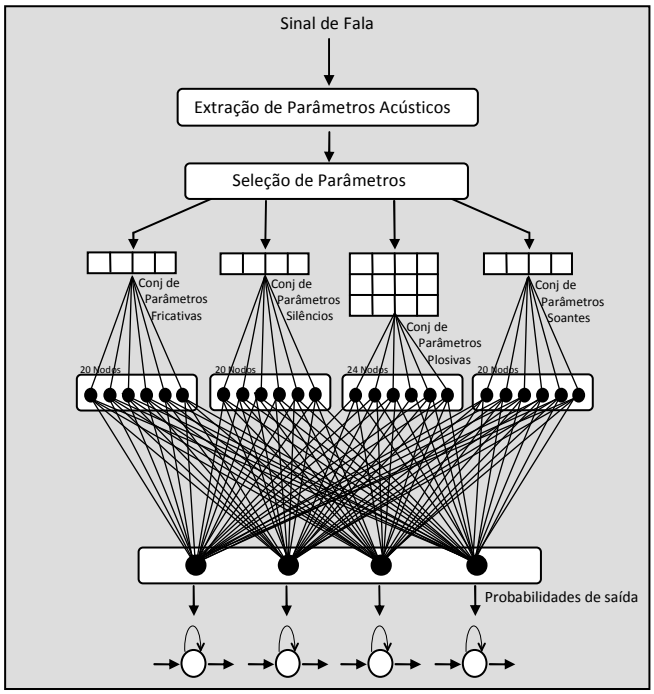


Figura 3.12. Híbrido MLP/HMM.

Os resultados experimentais são apresentados na Tabela 3.11. O classificador ANN/HMM superou largamente os resultados do classificador SVM/HMM. A *Correctness* chegou aos 81.93% e a *Accuracy* aos 76.16% usando o `HResults` do HTK. Com o AAT atingiu 80.75% *Correctness*, 72.88% de *Accuracy* e 87.55% de *Agreement*.

	Correctness	*Accuracy*	EER	*Agreement* (20ms)
Classificador híbrido SVM / HMM	81.30	74.77	25.23	--
	79.30	69.94	30.06	86.70
Classificador híbrido MLP/ HMM	81.93	76.16	23.84	--
	80.75	72.88	27.12	87.55

Tabela 3.11: Resultados do detetor de eventos baseado em MLP/HMM avaliado usando:

- HResults do HTK;
- Avaliação com Alinhamento Temporal (AAT; secção 2.3.3).

3.3.4. COMPARAÇÃO

Nas secções anteriores descreveram-se quatro sistemas de deteção de eventos. A arquitetura, filosofia e tecnologia subjacente é distinta mas a tarefa é a mesma: fornecer à saída um sinal segmentado em termos de 4 classes de eventos. Eleger um sistema como o melhor pode simplesmente passar pelo que obtém o melhor desempenho ou pode estar condicionada aos recursos envolvidos, uma vez que o número de parâmetros de treino de cada sistema é distinto.

Pretende-se com a Tabela 3.12 expor de uma forma comparativa os vários sistemas testados. Os valores apresentados foram obtidos usando o `HResults` do HTK. O sistema híbrido ANN/HMM obteve o melhor resultado em termos de *Accuracy*. Comparando com o detetor baseado em HMM obtiveram-se melhorias relativas de 11.2% de EER. Comparando com o detetor híbrido SVM/HMM as melhorias obtidas foram de 8.3%, apesar de este último sistema híbrido ter um número de parâmetros substancialmente mais baixo.

Adicionalmente treinou-se um MLP com os mesmos parâmetros de entrada usados pelo detetor HMM (12 MFCC mais energia e suas derivadas de ordem 1 e 2), usando, no treino, um contexto de 9 *frames*. Das duas últimas linhas da Tabela 3.12 consta uma comparação entre um sistema híbrido que usa o MLP referido e um sistema HMM com 8 misturas Gaussianas. O número de parâmetros de treino é semelhante em ambos os sistemas. Os resultados obtidos pelo híbrido ANN/HMM são novamente promissores. A *Correctness* chegou aos 87.12% e a *Accuracy* atingiu a melhor taxa apresentada, 80.17%. Este valor representa uma melhoria relativa de EER de 11.6% comparando com o sistema HMM com 8 misturas Gaussianas.

O bom desempenho deste híbrido levou-nos a selecionar esta arquitetura híbrida na implementação de um sistema que integra a deteção de eventos no reconhecimento fonético, descrito na secção 4.4.

	%Corr	%Acc	%EER	Número de Parâmetros de treino	Melhorias relativas de EER (%)		
					HMM	SVM/NMD	SVM/HMM
HMM (1 mistura)	88.47	73.14	26.86	979			
Híbrido SVM/NMD	88.86	74.01	25.99	<100	3.2		
Híbrido SVM/HMM	81.30	74.77	25.23	<100	6.1	2.9	
Híbrido ANN/HMM	81.93	76.16	23.84	976	11.2	8.3	5.5
HMM (8 misturas)	89.36	77.57	22.43	7615			
Híbrido ANN/HMM	87.12	**80.17**	19.83	7513	11.6		

Tabela 3.12: Resultados globais de deteção de eventos usando o HResults do HTK.

CAPÍTULO 4

RECONHECIMENTO DE FONES NA TIMIT

O reconhecimento de fones na TIMIT é uma tarefa com mais de duas décadas e cujo desempenho tem, naturalmente, vindo a crescer com o tempo. Existe uma panóplia de sistemas que avaliam a TIMIT centrando-se em três grandes domínios: o da segmentação, o da classificação e o do reconhecimento de fones. Enquanto o primeiro atinge taxas de 93%[19], [48], o segundo alcança 83%, [58] e o terceiro fica-se pelos 79%, [102][143].

A segmentação fonética é o processo de encontrar, numa dada locução, os limites de uma sequência de fones. Encontrar esses limites, ou marcas, ao nível do fone é uma tarefa complexa atendendo aos efeitos de coarticulação, onde fones adjacentes se influenciam mutuamente. Se a sequência de fones for conhecida, a tarefa de segmentação denomina-se por alinhamento uma vez que se conhecem à partida o número de limites a encontrar.

A classificação fonética é um problema artificial, mas instrutivo em ASR, [138]. Parte-se de um sinal de fala corretamente segmentado em unidades fonéticas, mas cuja sequência de fones é desconhecida. O problema consiste em identificar o fone correto para cada segmento. Os modelos dos fones competem entre si na tentativa de atribuir a sua etiqueta a esse segmento. A etiqueta do modelo vencedor (a mais parecida na abordagem *a posteriori*) é comparada com a etiqueta respetiva da TIMIT e é calculada a taxa de acerto. Esta medida, permite uma boa avaliação da qualidade da modelação acústica, uma vez que calcula o desempenho à margem da utilização de qualquer tipo de gramática para modelar a sequência de fones [120].

O reconhecimento de fones obedece a um critério mais complexo e exigente. Ao reconhecedor é fornecido apenas o sinal acústico e o sistema terá de encontrar a melhor sequência de fones para locução em questão. Neste caso, pode ser usado qualquer modelo

[19] Considerando um intervalo de tolerância de 20ms

de gramática e qualquer modelo acústico. A melhor sequência de fones encontrada é comparada com a de referência tomando em consideração os fones corretos, as substituições, as inserções e os apagamentos.

Será dado, neste trabalho, destaque ao reconhecimento de fones, uma vez que é o seu tema central. Na secção seguinte far-se-á uma descrição da base de dados TIMIT prosseguindo o capítulo fazendo referência a alguns trabalhos, que se considera terem mais destaque na literatura, centrados na tarefa de reconhecimento de fones na TIMIT. Além de uma breve descrição da abordagem e técnica inerente a cada sistema, serão apresentadas as taxas de desempenho de cada um, apesar de este critério não ser o único envolvido numa comparação justa. Há inúmeros fatores importantes na avaliação de um sistema, que condicionam determinantemente o seu desempenho. Estes fatores relacionam-se com a rapidez de convergência do treino, com o número e tipo de parâmetros de treino, com o grau de afinação do sistema, com a independência à base de dados de treino, etc. Ainda assim será feita uma análise comparativa dos marcos em reconhecimento de fones na TIMIT.

4.1. A BASE DE DADOS TIMIT

A base de dados DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus (TIMIT - Texas Instruments (TI) e Massachusetts Institute of Technology (MIT)), [36], é composta por gravações de frases lidas em Inglês foneticamente balanceadas.

O sinal de fala foi gravado usando um microfone Sennheiser, a uma taxa de amostragem de 16 kHz com uma resolução de 16 bits. Contém um total de 6300 locuções (5.4 horas), provenientes de 630 locutores (10 frases cada) oriundos das 8 maiores regiões dos Estados Unidos. Todas as locuções estão manualmente transcritas ao nível do fone.

As locuções da TIMIT estão divididas em 3 grupos: dialeto (`SA`), foneticamente compactas (`SX`) e foneticamente diversas (`SI`). A escolha das duas frases[20] `SA` foi feita de forma a revelar os sotaques das diversas regiões, e foram lidas por todos os 630 locutores. As frases `SX` foram escolhidas manualmente de forma a serem foneticamente compreensíveis e ao mesmo tempo compactas no sentido de serem breves. As frases `SI` foram escolhidas de fontes de texto já existente. A Tabela 4.1 fornece informação detalhada sobre os tipos e número de frases em cada grupo e número de locutores envolvidos.

[20] \She had your dark suit in greasy wash water all year\ e \Don't ask me to carry an oily rag like that\

Tipo de locução	# locuções	#locutores/frases	TOTAL	#frases/locutor
Dialeto (SA)	2	630	1260	2
Compactas (SX)	450	7	3150	5
Diversas (SI)	1890	1	1890	3
TOTAL	2342	-	6300	10

Tabela 4.1: Descrição do tipo de frases da TIMIT.

A documentação do Corpus sugere conjuntos quer de treino (≈ 70%) quer de teste, como descritos na Tabela 4.2. Para o teste são propostos 2 conjuntos: o *core* e o completo. O conjunto "*test core*" inclui 2 locutores masculinos e 1 feminino de cada uma das 8 regiões, envolvendo no total 24 locutores. Cada locutor profere 8 frases resultando num conjunto de 192 locuções. O conjunto de teste completo foi formado de modo análogo ao anterior mas envolvendo 168 locutores (112 masculinos e 56 femininos) resultando em 1344 locuções. O conjunto de treino consiste em 4620 locuções, mas normalmente não são usadas as locuções SA. O conjunto das frases SI e SX perfaz 3696 locuções (462 locutores). À exceção das frases SA (dialetos) que, classicamente, se excluem nos testes, não há sobreposição entre os conjuntos de treino e teste. A base de dados é descrita em detalhe em [169].

Conjunto	# locutores	#locuções	#horas
Treino	462	3696	3.14
Test Core	24	192	0.16
Teste Completo	168	1344	0.81

Tabela 4.2: Descrição dos conjuntos de treino e teste propostos na TIMIT.

As transcrições da TIMIT, cujo dicionário fonético é apresentado no Anexo I, são baseadas em 61 fones. O alfabeto usado - TIMITBET - deriva do ARPABET[21], mas com algumas modificações [59]. Detalhes sobre a transcrição e alinhamento manual podem ser encontrados em [168], e em [60] é feita uma análise fonética da base de dados.

Os 61 fones e símbolos originais da TIMIT são por vezes considerados demasiado detalhados para aplicações práticas, de forma que, no treino, alguns autores agrupam fones, reduzindo o conjunto de treino para 48 fones. Na avaliação é comum reduzir-se a um conjunto de 39

[21] O conjunto mais importante de símbolos para transcrição fonética é o *International Phonetic Alphabet* (IPA). No entanto este alfabeto não é conveniente para algumas aplicações (usa símbolos não existentes nos teclados normais) surgindo assim mapeamentos de símbolos IPA para símbolos ASCII. É disso exemplo o ARPABET usado pela ARPA - *Advanced Research Projects Agency*.

fones, tal como proposto por Lee e Hon, [76], cuja fusão se ilustra na Tabela 4.3. Os fones da coluna da direita são agrupados adotando a etiqueta correspondente na coluna da esquerda. São assim removidos 23 fones do conjunto original e é adicionado a etiqueta *sil* ao conjunto final (para pausas, oclusões e início e fim de locução). Os restantes fones do conjunto inicial de 61 mantêm-se inalterados (não agrupam com nenhum outro fone). Esta categorização de fones foi seguida por muitos autores e tornou-se numa norma efectiva de avaliação com o Corpus TIMIT, quer em classificação quer em reconhecimento. Neste trabalho o treino foi feito usando o conjunto dos 61 fones originais e o teste com o conjunto de 39 fones. Há no entanto uma diferença em relação à proposta de Lee e Hon. A proposta destes últimos autores ignora o fone [q] e no presente trabalho este fone faz parte da classe *sil*.

Fone resultante	**Fones Agrupados**
aa	aa, ao
ah	ah, ax, ax-h
er	er, axr
hh	hh, hv
ih	ih, ix
l	l, el
m	m, em
n	n, en, nx
ng	ng, eng
sh	sh, zh
uw	uw, ux
sil	pcl, tcl, kcl, bcl, dcl, gcl, h#, pau, epi
-	q

Tabela 4.3: Correspondência entre os 61 fones originais da TIMIT e os 39 propostos por Lee e Hon, [76]. Os fones na coluna da direita são agrupados ficando com a designação da coluna da esquerda. O fone [q] é ignorado. Todos os restantes fones mantêm-se.

A base de dados TIMIT tem sido uma base de dados usada durante várias décadas e é ainda hoje amplamente usada em testes, quer relacionados com reconhecimento e classificação de fala quer com reconhecimento de locutor. Isto deve-se não só ao facto de cada locução estar etiquetada manualmente ao nível do fone e conter indicação do locutor, seu género e região de origem, mas também por ser considerada uma base de dados suficientemente pequena para garantir um treino rápido e suficientemente grande para demonstrar as capacidades do sistema em análise.

4.2. ESTADO DA ARTE EM RECONHECIMENTO FONÉTICO NA TIMIT

Apesar dos avanços das últimas décadas, a tarefa de reconhecimento de fones na TIMIT continua a ser uma tarefa difícil e desafiante. No extenso role de abordagens seguidas no sentido de melhorar o desempenho dos reconhecedores de fones, inclui-se o uso de múltiplos conjuntos de características, o refinamento das características de entrada, dos modelos estatísticos, dos critérios de treino, da modelação acústica, o tratamento de ruído, o recurso a modelos de linguagem, etc. Apesar de ser muito vasto o tipo de abordagens e técnicas aplicadas à tarefa de reconhecimento de fones na TIMIT, nesta secção far-se-á referência a vários trabalhos de investigação desenvolvidos nas últimas décadas considerados de interesse ou pela abordagem seguida ou pelos resultados alcançados.

Um dos primeiros trabalhos que envolveu o reconhecimento de fones na TIMIT foi apresentado por Lee e Hon, [76] em 1989, imediatamente após o lançamento da base de dados TIMIT em Dezembro de 1988. O sistema é baseado em HMMs discretos. Os melhores resultados foram obtidos modelando os fones com contexto à direita (1450 difones) usando uma bigrama como modelo de linguagem. Os parâmetros de entrada constam de 3 codebooks de 256 vetores de coeficientes cepstrais derivados de LPC. Atingem os valores de 73.80% para *Correctness* e 66.08% para *Accuracy* usando 160 locuções de um conjunto de teste (o TID7). Os autores propõem que os seus resultados sejam considerados um marco de referência. Na verdade, o artigo viria a ser realmente um marco, não tanto pelos resultados, mas mais pela fusão de fones que os autores propõem fazer e que se encontra descrita na Tabela 4.3.

Também em 1989, Steve Young apresenta a primeira versão do HTK (*Hidden Markov Model Toolkit*), [162]. O *software*, desenvolvido na Universidade de Cambridge, e que permite a construção e manipulação de modelos de Markov, viria a dar um incremento notável na área do reconhecimento de fala. Em, [161] o autor apresenta o conceito de partilha de estados no treino de trifones. O objetivo é produzir um conjunto compacto de HMMs dependentes do contexto, mostrando que a partilha de estados reduz significativamente o número de parâmetros a treinar. Os trifones gerados partem de um conjunto de 48 fones. As condições experimentais são semelhantes às estabelecidas por Lee e Hon, [76], exceto nos parâmetros que são MFCC com energia e coeficientes de regressão de 1ª ordem (deltas). Os melhores

resultados apresentados são 73.7% para *Correctness* e 59.9% para *Accuracy*, usando o conjunto de 39 fones e 160 locuções geradas aleatoriamente a partir do conjunto de teste.

Em 1991, Robinson e Fallside, desenvolvem um sistema de reconhecimento de fones, baseado numa rede neuronal recorrente, [124], que produziu os resultados de: 76.4 % para *Correctness* e 68,9% para *Accuracy* usando o mesmo conjunto de teste que Lee e Hon, [76]. Estes resultados sobem para 76.5 % para *Correctness* e 69,8% para *Accuracy* considerando o conjunto completo de teste. Os autores dão indicação de taxas ainda superiores (71.2% para *Accuracy*), mas o conjunto de fones deixa de ser os tradicionais 39 fones para ser um conjunto de 50 fones. Em 1993, os mesmos autores [126] integraram a rede recorrente com um descodificador HMM, onde as redes são usadas no cálculo de probabilidades *a posteriori* dos estados. O sistema foi testado com a base de dados WSJ (*Wall Street Journal*). Já com a TIMIT surgem resultados num híbrido RNN/HMM em 1994, [125]. À rede é aplicado um contexto longo à esquerda. No treino é usada uma função de ativação de saída *softmax* e o critério de erro *cross-entropy*. As saídas da rede são treinadas em função dos 61 fones originais da TIMIT. Os resultados alcançados, em relação ao conjunto tradicional de 39 fones, foram 78.6 % para *Correctness* e 75% para *Accuracy*. Este resultado está ainda hoje acima de publicações recentes. O artigo apresenta ainda uma comparação interessante de vários trabalhos de reconhecimento de fones.

Em 1993, Lamel e Gauvain [67] apresentaram um trabalho de investigação em reconhecimento de fones usando HMM contínuos (CDHMM - *continuous density HMMs*) na estimação de modelos dependentes do contexto treinados usando as técnicas de estimação ML e MAP (*maximum a posteriori*). Os parâmetros constam de coeficientes cepstrais derivados de LPC com coeficientes de regressão de primeira e segunda ordem. Usando o conjunto completo de teste alcançaram os resultados de 77.5% e 72.9% para *Correctness* e *Accuracy*, respetivamente.

Halberstadt e Glass em 1998, [44], e no seguimento das pesquisas do trabalho de doutoramento do primeiro, [43], propõem um sistema que combina vários classificadores. O treino é feito de forma a otimizar a modelação acústica por via de várias medidas acústicas heterogéneas. Cada classificador é treinado individualmente e é responsável pela identificação de um subconjunto dos fones originais da TIMIT. São usados 6 classificadores para treinar 60 fones (não consideram a plosiva glotal [q]). Adicionalmente, são usados mais

3 classificadores para combinar a informação resultante dos 6 primeiros. A Tabela 4.4 mostra o conjunto de fones que é treinado em cada classificador.

Classes de fones	# etiquetas da TIMIT	etiquetas da TIMIT
Vogais e Semivogais	25	aa ae ah ao aw ax axh axr ay eh er ey ih ix iy ow oy uh uw ux el l r w y
Nasais/Flaps	8	em en eng m n ng nx dx
Fricativas fortes	6	s z sh zh ch jh
Fricativas fracas	6	v f dh th hh hv
Plosivas	6	b d g p t k
Oclusivas	9	bcl dcl gcl pcl tcl kcl epi pau h#
Soantes	33	Vogais e Semivogais + Nasais/Flap
Obstrusivas	18	Fricativas fortes e fracas + Plosivas
Silêncios	9	bcl dcl gcl pcl tcl kcl epi pau h#

Tabela 4.4: Classes fonéticas alargadas usadas no sistema proposto por Halberstadt, [43].

Na classificação é usado o reconhecedor baseado em segmentos SUMMIT[22], [37]. São usados modelos de misturas Gaussianas e conjuntos de fones diferentes usam parâmetros também diferentes: MFCC, PLP e ainda uma representação de MFCC denominada pelos autores como "*discrete cosine transforms coefficients*". Testam ainda janelas de tamanho diferente, parâmetros temporais, etc. O reconhecimento de fones é feito por via de duas filosofias distintas: uma hierárquica e outra paralela. Os resultados, à época, superaram todos os sistemas existentes. A medida *Accuracy* no conjunto de *test core* atingiu os 75.6%. A imensa lista de testes que realizaram permitiu aos autores concluir que se obtêm melhores resultados usando combinações de classificadores treinados em separado do que um simples classificador treinado para distinguir entre várias classes. O melhor resultado é obtido usando uma combinação de 5 dos 8 classificadores disponíveis e usando também conjuntos de parâmetros diferentes nos vários classificadores.

Em 2003, Reynolds e Antoniou, [120] propõem-se treinar uma rede MLP modular. Treinam num primeiro nível os mesmos fones (39), mas usando conjuntos de parâmetros diferentes (MFCC, PLP, LPC e combinações entre estes). Obtêm assim várias predições para o mesmo fone que são posteriormente combinadas num outro MLP.
Os melhores resultados são obtidos otimizando o número de nodos escondidos e usando também informação de 7 classes fonéticas alargadas, cuja constituição consta da Tabela 4.5.

[22] SUMMIT é um sistema de reconhecimento de fala desenvolvido no MIT. Baseia-se no recurso a marcos do sinal para modelar eventos acústico-fonéticos e usa um *finite-state transducer* (FST) para representar os aspetos da hierarquia do sinal: regras fonológicas, léxico e modelo probabilístico de linguagem.

Classes de fones	# etiquetas TIMIT	Etiquetas da TIMIT
Plosivas	8	b d g p t k jh ch
Fricativas	8	s sh z f th v dh h
Nasais	3	m n ng
Semi-vogais	5	l r er w y
Vogais	8	iy ih eh ae aa ah uh uw
Ditongos	5	ey aw ay oy ow
Oclusivas	2	sil dx

Tabela 4.5: Classes fonéticas alargadas usadas no sistema proposto por Reynolds e Antoniou, [120].

Na deteção destas classes foram testados vários contextos tendo os melhores resultados sido obtidos com um contexto de 35 *frames* (350ms). Num conjunto de teste um pouco diferente do habitual (retiram ao conjunto de teste completo as locuções do conjunto de teste *core*) obtêm uma *Accuracy* de 75.8%, nos 39 fones da TIMIT. O artigo apresenta resultados quer de reconhecimento quer de classificação de fones e apresenta uma tabela comparativa com outros trabalhos na mesma tarefa.

Sha e Saul apresentam em [138] um trabalho que, apesar de não apresentar um desempenho muito competitivo, parte de uma ideia interessante. Treinaram discriminativamente GMMs, usando o princípio base das SVMs: maximizar a margem entre as classes. Com 39 parâmetros MFCC *standard* e com 16 misturas Gaussianas, obtiveram uma *Accuracy* de 69.9%.

Na continuidade do estudo sobre TRAPs (*TempoRAl Patterns*) desenvolvido na Universidade de Brno, surge um trabalho sobre estruturas hierárquicas de redes neuronais para reconhecimento de fones, [137]. O objetivo do trabalho é averiguar a contribuição que o contexto temporal do sinal poderá dar no reconhecimento de fones. O sistema assenta em duas linhas principais:

TRAPs – um conjunto de MLPs onde cada rede neuronal recebe como entrada parâmetros de uma dada banda crítica de frequência. O vetor de entrada TRAP descreve a evolução temporal da densidade espetral de uma dada banda crítica. As redes são treinadas de forma a classificar os padrões de entrada em probabilidades de fones. As probabilidades de todos estes classificadores (um por cada banda crítica) são entregues a outro MLP – um combinador de probabilidades – que fornece nova probabilidade *a posteriori* final para cada fone.

Divisão do contexto temporal – também baseada num conjunto de MLPs, considera que se podem processar em separado duas partes de cada fone: uma considerando contexto à

esquerda e outra à direita que depois são combinadas em nova rede que fornece probabilidades *a posteriori* finais dos fones.

A combinação entre TRAPs e divisão do contexto temporal pode ser vista como uma divisão horizontal (TRAPs divide na frequência) seguida de uma vertical (divisão do contexto temporal divide no tempo – metade para a esquerda e metade para a direita).

Os autores testaram vários conjuntos de parâmetros de entrada, redes com saídas correspondentes a fones e a estados HMMs dos fones e otimizaram também a divisão em bandas críticas encontrando o melhor número de bandas a analisar tendo atingido 75.16% de *Accuracy*. Afinando o número de nodos escondidos dos MLP; usando uma bigrama na descodificação de Viterbi e usando 5 blocos de contexto (em vez de usarem só esquerda e direita) alcançaram uma melhoria interessante (4.5% de melhoria relativa) atingindo os 78.52% de *Accuracy*.

Há, no entanto, uma questão pouco clara no que se refere ao conjunto de fones usado. O reconhecedor de Brno baseia-se em 39 fones, mas que não coincidem na íntegra com o conjunto de 39 fones usuais da TIMIT. Nas plosivas, fundem o período de oclusão com a respetiva plusão (ex: bcl b → b) contrariamente a serem considerados silêncios. No entanto, a nossa análise à base de dados TIMIT diz-nos que se é verdade que em 87% das ocorrências do fone [bcl] este é seguido por [b] nos restantes 13% de ocorrências existem sequências como: bcl–t , bcl–el, bcl–ix, etc., para as quais não é indicado o procedimento seguido. O mesmo se passa em relação às restantes oclusões. Na documentação publicada não é claro se a fusão se mantêm nestas situações. Alguns autores, [102] põem em questão os resultados afirmando que seriam provavelmente inferiores se fossem usados os fones normais. Nos testes que realizámos fazendo as fusões indicadas, os nossos reconhecedores (HMM e MLP/HMM) apresentaram um desempenho mais baixo.

Resultados também interessantes são reportados por um grupo da Microsoft que se dedica ao estudo dos *Hidden Trajectory Models* (HTM). Os HTM são um tipo de modelo probabilístico generativo que procura modelar a dinâmica do sinal de fala e incluir capacidades contextuais alargadas que não existem nos HMMs, [24]. O artigo [164] faz uma descrição detalhada dos HTMs. O HTM usa um filtro bidirecional para estimar trajetórias dos dados a partir de uma sequência de fones. Esta estimativa é posteriormente usada para calcular a verosimilhança do modelo para os dados em observação. A melhor sequência de fones (maior verosimilhança) é obtida através de uma procura numa *lattice* e de um algoritmo de repontuação (*rescoring*) especialmente desenvolvido para os HTM. Os

resultados destes autores atingem os 78.40% de *Correctness* e 75.17% de *Accuracy*, [23]. Os parâmetros usados no modelo HTM são ceptrum estático e respetivas derivadas.

Rose e Momayyez, [128], usam os resultados de oito detetores de características fonológicas para produzir conjuntos de parâmetros a fornecer a um reconhecedor HMM. Os detetores são redes TDNN (*Time Delay Neural Network*) cujas entradas são os tradicionais MFCC, com primeiras e segundas derivadas. Os HMMs definidos com este conjunto de parâmetros fonológicos são combinados com HMMs definidos com os tradicionais 39 parâmetros MFCC por via de um procedimento de repontuação de uma *lattice*. Para o conjunto de teste completo, alcançam uma *Accuracy* ao nível do fone (não referem o número de fones usado) de 72.2%.

Sabendo que as confusões ao nível do fone ocorrem em fones semelhantes [45], Scanlon, Ellis e Reilly [132] propõem um sistema onde combinam a informação oriunda de um sistema base (um híbrido MLP/HMM alimentado com parâmetros PLP e suas 1ª e 2ª derivadas, 9 *frames* de contexto, 1000 nodos escondidos e treinado para discriminar os 61 fones da TIMIT), com informação de um grupo de redes especialistas em classes fonéticas alargadas. Apesar do artigo referir a divisão exposta na Tabela 3.2, treinam somente 4 redes especialistas: vogais (com 25 fones), plosivas (com 8 fones), fricativas (com 10 fones) e nasais (com 7 fones).
Uma vez que as características dos fones de cada classe são substancialmente diferentes, usaram, em cada rede especialista, parâmetros diferentes escolhidos através de um critério baseado em *Mutual Information*. As redes são MLPs, com um número de saídas igual ao número de fones que procuram distinguir. Combinam a informação de um detetor de classes alargadas (também um MLP que, para cada *frame*, atribui uma probabilidade de esta pertencer a uma dada classe) com a informação do sistema base e em caso de coerência na classe identificada, juntam (substituem) as predições da rede especialista às predições do sistema base. Esta fusão de informação é então passada a um descodificador HMM. Obtiveram 74.2% de *Accuracy*, usando o conjunto de teste completo da TIMIT e os 39 fones usuais.

Em 2004, surge um projeto de investigação com aplicações em ASR de nome ASAT (*Automatic Speech Attribute Transcription*), [70], e no âmbito deste projeto, são publicados vários trabalhos em reconhecimento fonético com a TIMIT, [16][104][105][106]. Trata-se de

um projeto financiado pela ITR (*Industry Training Register*) e que envolve 4 institutos: *Georgia Institute of Technology*, *Ohio State University*, *University of California at Berkeley* e *Rutgers University*. O maior objetivo do ASAT é promover o desenvolvimento de abordagens de ASR baseadas no conceito de deteção de atributos e integração de conhecimento.

Em 2007 surge um artigo conjunto, [16] onde apresentam vários métodos de deteção de atributos do sinal de fala. O sistema global é composto por um *front-end* cuja saída fornece predições dos atributos detetados de uma forma probabilística, seguido de um módulo de combinação, cuja função é fundir a informação dos vários atributos que passada a um descodificador faz o reconhecimento de fones. Os detetores de atributos acústico-fonéticos usam tecnologias diversas: MLPs, SVMs, HMMs, TDNNs. Dependendo do classificador, foram usados parâmetros de entrada ligeiramente diferentes: MFCC ou PLP com 1ª e 2ª derivada e contexto. O conjunto de atributos detetados também é diferente em cada classificador, quer em número quer na característica que deteta. A Tabela 4.6 mostra os atributos e parâmetros usados por detetor.

Métodos de deteção	Parâmetros de entrada	Atributos detetados
MLP (*Sound Pattern of English*)	13 MFCC 10ms *frames*	vocalic, consonantal, high, back, low, anterior, coronal, round, tense, voice, continuant, nasal, strident, silence. (14 attributes)
SVM	13 MFCC 9 context *frames* 10ms *frames*	coronal, dental, fricative, glottal, high, labial, low, mid, nasal, round minus, round plus, silence, stop, velar, voiced minus, voiced plus, vowel. (17 attributes)
HMM	13 MFCC+Δ+ΔΔ 10ms *frames*	
Multi-class MLPs	13 PLP+Δ+ΔΔ 9 context *frames* 10ms *frames*	Sonority: obstruent, silence, sonorant, syllabic, vowel; Voicing: voiced, voiceless, NA; Manner: approximant, flap, fricative, nasal, flap, stop-closure, stop, NA; Place: alveolar, dental, glottal, labial, lateral, palatal, rhotic, velar, NA; Height: high, low-high, low, mid-high, mid, NA; Backness: back, back-front, central, front, NA; Roundness: nonround, nonround-round, round-nonround, round, NA; Tenseness: lax, tense, NA. (44 atributes)

Tabela 4.6: Atributos e parâmetros em função do classificador usados no artigo do grupo ASAT, [16].

De forma a dotar um sistema de reconhecimento com informação compactada, combinaram a informação ao nível dos atributos. A fusão de atributos foi feita usando CRFs, que, combinada com uma repontuação da *lattice*, apresenta resultados interessantes, [106]: 73.39/69.52 (*Correctness/Accuracy*). Os autores afirmam que os atributos detetados podem

ser combinados de forma a formarem um conjunto de informação de alto nível, útil nas tarefas de reconhecimento de fala. No entanto, os melhores resultados foram obtidos usando um híbrido MLP/HMM em que o MLP fornece predições de 44 atributos.

Morris e Fosler-Lussier em [106] usaram 8 MLPs para extrair atributos fonéticos do sinal de fala que constam da Tabela 4.7. Às saídas lineares dos MLPs foi aplicada uma transformação KL (*Karhunen-Loeve*) de forma a descorrelacioná-las. Os 44 atributos fonéticos são então modelados por HMM convencionais com Gaussianas e por CRFs. Os melhores resultados apresentados provêm de uma arquitetura TANDEM (atributos fonéticos são usados como parâmetros de entrada nos HMMs) com trifones modelados com 4 misturas Gaussianas: 72.52/66.69% (*Correctness/ Accuracy*). Com a arquitetura CRF o desempenho do sistema é relativamente baixo 66.74/65.23% (*Correctness/Accuracy*), mas ainda assim, superior ao TANDEM HMM com monofones modelados com uma Gaussiana simples.

Os mesmo autores publicam em 2008 um outro trabalho [104] onde a arquitetura TANDEM, com trifones treinados até 16 misturas Gaussianas, sobe o desempenho até 68.53% de *Accuracy*. O melhor resultado apontado por estes autores, usando o conjunto *test core* é de 70.74% de *Accuracy* e usando um conjunto de 118 locutores (juntaram ao conjunto *core* os locutores do conjunto de teste completo retirando os locutores de um conjunto de desenvolvimento) de 71.49% de *Accuracy*. Estes resultados foram obtidos aplicando aos CRFs 105 parâmetros de entrada, 61 dos quais correspondem a probabilidades *a posteriori* dos fones da TIMIT e os restantes a 44 são atributos fonéticos provenientes dos 8 classificadores MLP, descritos na Tabela 4.7. Os MLPs foram todos treinados tendo como parâmetros de entrada PLP, suas derivadas e segundas derivadas e 9 *frames* de contexto.

Atributo	**Saídas**
SONORITY	vowel, obstruent, sonorant, syllabic, silence
VOICE	voiced, unvoiced, n/a
MANNER	fric., stop, closure, flap, nasal, approx., nasal flap, n/a
PLACE	lab., dent., alveolar, pal., vel., glot., lat., rhotic, n/a
HEIGHT	high, mid, low, lowhigh, midhigh, n/a
FRONT	front, back, central, backfront, n/a
ROUND	round, nonround, round-nonround, nonround-round, n/a
TENSE	tense, lax, n/a

Tabela 4.7: Atributos fonéticos extraídos no trabalho de Morris e Fosler-Lussier em [106].

Unindo o conhecimento da *Brno University of Technology* com o do *Georgia Institute of Technology* surgem um dos melhores resultados reportados na tarefa de reconhecimento de fones na TIMIT. São apresentados por Siniscalchi, Schwarz e Lee, [143] e alcançam os 79% de

Accuracy no conjunto de teste completo da TIMIT. O sistema base está apresentado na Figura 4.1, é análogo ao descrito por Schwarz, Matejka, e Cernocky em [137], e pode ser visto como arquitetura TANDEM de MLPs, que termina num descodificador HMM. O contexto à esquerda e direita são processados em separado e aplicados a diferentes redes neuronais. As saídas destas duas redes são dadas como entrada a uma terceira rede cujas saídas são processadas por um descodificador HMM, fornecendo à saída um sinal segmentado em termos de fones. A alteração proposta em, [143] é incluir na descodificação um módulo de repontuação de *lattices*. Genericamente, a repontuação de *lattices* é feita em duas fases. Na primeira, o descodificador gera uma coleção de hipóteses. Segue-se um algoritmo de repontuação que reordena estas hipóteses incluindo no processo de descodificação informação adicional. Esta informação adicional consta de informação articulatória (modo e lugar de articulação) dada por uma combinação entre um banco de detetores e uma ANN. O banco de detetores recorre a HMMs para mapear um segmento de fala numa de 15 classes (fricativa, vogal, plosiva, nasal, semivogal, baixa, media, alta, labial, coronal, dental, velar, glotal, retroflexa e silêncio). A partir de uma razão de verosimilhança, associada a cada detetor, é treinada uma ANN que fornece na saída probabilidades *a posteriori* para o conjunto de fones em avaliação. São estes valores que são usados na repontuação das *lattices* alterando o valor associado aos arcos como uma soma pesada entre os valores originais e os vindos do detetor de eventos. De referir que o conjunto de fones usado foi o mesmo que em, [137] que como já referimos, não nos é muito claro.

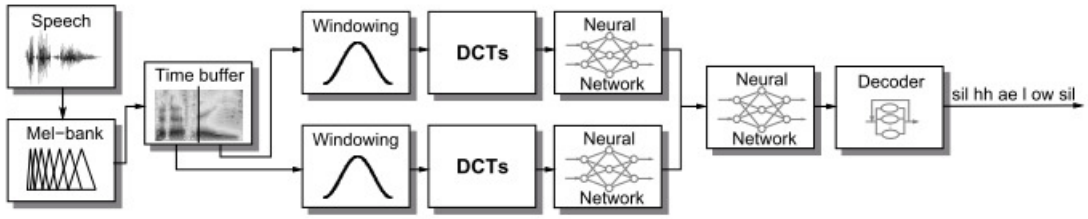


Figura 4.1: Reconhecedor de fones usado em [143].

No início de 2009, Hifny e Renals [47] apresentam um sistema de reconhecimento fonético onde a modelação acústica é feita através de *Augmented Conditional Random Fields*. Os resultados na TIMIT são de 73.4% de *Accuracy* usando o conjunto core e 77% num outro conjunto que inclui o conjunto de teste completo e as frases SA.

Recentemente foi apresentada [103] uma nova técnica de aprendizagem automática para aplicações de reconhecimento de fala. Os autores, Mohamed e Hinton, aplicam *Restricted Bolzmann Machines* (RBMs) ao reconhecimento fonético. As RBMs têm a vantagem de abordar a questão da assunção de independência condicional feita pelos HMMs. Mais especificamente, usam um estado oculto que permite, de uma forma conjunta, que vários parâmetros diferentes gerem uma saída para uma dada *frame*. Em reconhecimento fonético com a TIMIT os autores referem que as RBMs superam um sistema convencional baseado em HMMs, em 0.6% de PER. O resultado em termos de *Accuracy* e usando o *test core* é de 77.3%. Uma publicação recente destes autores, [102] refere-se à aplicação de redes neuronais em modelação acústica: múltiplas camadas são pré-treinadas generativamente. Surgem, os que são do nosso conhecimento, os melhores resultados reportados na tarefa de reconhecimento de fones na TIMIT considerando o conjunto *test core*: 79.3% de *Accuracy*.

Apesar de não ser possível uma comparação inteiramente justa, sumaria-se na Tabela 4.8., [86] alguns dos que consideramos serem os sistemas mais importantes das últimas 2 décadas, e que consideramos serem marcos na tarefa de reconhecimento de fones na TIMIT. Os sistemas diferem consideravelmente em termos de parâmetros usados, material de teste, conjunto de fones, modelação acústica, etc., o que torna a comparação ainda mais difícil. Acresce ainda o facto de a comparação de trabalhos atuais com trabalhos de há duas décadas não ser inteiramente justa, uma vez que as limitações do ponto de vista de capacidade computacional a que os primeiros trabalhos estiveram expostos não se colocam atualmente. A tabela está ordenada por ordem cronológica e inclui, além da tecnologia de reconhecimento envolvida, as taxas de reconhecimento alcançadas assim como o conjunto de teste usado.

Pela tabela conclui-se que nos últimos 20 anos o desempenho da tarefa de reconhecimento de fones na TIMIT melhorou aproximadamente 13%, essencialmente nos 5 primeiros anos de investigação. Em [86] abrimos a discussão se haverá espaço para melhorias nesta tarefa afirmando mesmo que se terá atingido um teto de desempenho. Contudo, independentemente das taxas de reconhecimento da TIMIT virem ou não a sofrer acréscimos, esta continuará a ser uma base de dados de referência no teste de novas abordagens.

Ano	Sistema	Método	%Corr	%Acc	Conjunto de Teste
1989	Lee e Hon, [76]	HMM	73.80	66.08	160 frases (TID7)
1991	Robinson e Fallside, [124]	Recurrent Error Propagation Network	76.4 76.5	68.9 69.8	160 frases (TID7) Conjunto completo
1992	Young, [161]	HMM	72.5	61.07	160 frases escolhidas aleatoriamente
1993	Lamel e Gauvain, [67]	HMMs, trifones	77.5	72.9	Conjunto completo
1994	Robinson, [125]	RNN	78.6 77.5	75.0 73.9	Conjunto completo Core
1998	Halberstadt, [43]	Parâmetros de entrada heterogéneos. SUMMIT. classes alargadas	-	75.6	Core
2003	Reynolds e Antoniou, [120]	MLP, classes alargadas	-	75.8	1152 frases
2006	Sha e Saul [138]	GMMs treinados segundo o princípio das SVMs	-	69.9	Conjunto completo
2006	Schwarz, Matejka e Cernocky,[137]	TRAPs e Divisão do contexto temporal	-	78.52	Conjunto completo
2007	Deng, Yu e Acero [24]	Hidden Trajectory Models	78.40	75.17	Core
2007	Rose e Momayyez [128]	TDNN, phonological features HMM		72.2	Conjunto completo
2007	Scanlon, Ellis e Reilly, [132]	MLP/HMM	-	74.2	Conjunto completo
2007	ASAT, [16]	MLP/HMM	73.39	69.52	-
2007	Siniscalchi, Schwarz e Lee, [143]	TRAPs e Divisão do contexto temporal + repontuação de lattice	-	79.04	Conjunto completo
2008	Morris e Fosler-Lussier [106]	MLP/CRF	- 74.76	70.74 71.49	Core 944 frases
2009	Hifny e Renals, [47]	Augmented CRFs	-	77.0	Conjunto completo+SA
2010	Mohamed, Hinton, [103]	Boltzmann Machines	-	77.3	Core
2011	Mohamed, et al., [102]	Monophone Deep Belief Networks		79.3	Core

Tabela 4.8: Comparação de vários trabalhos usando técnicas distintas no reconhecimento dos 39 fones TIMIT.

Outro facto interessante a reter por análise da Tabela 4.8. refere-se com o método ou tecnologia de reconhecimento envolvida. Os melhores resultados alcançados, [102][137][143] advêm de sistemas baseados em redes neuronais: sejam elas MLPs ou a

nova geração de redes neuronais, as RBMs. O intenso estudo sobre o assunto aproxima o processo de reconhecimento de fala humano (até hoje pouco conhecido) a um processo sequencial de processamento em camadas (em que a informação processada numa camada advém de informação processada em camadas anteriores) ou seja – a uma rede neuronal.

Atendendo ao exposto, a opção tomada foi a de abordar o problema por via de um híbrido MLP/HMM que se passa a descrever.

4.3. O RECONHECEDOR DE FONES HÍBRIDO MLP/HMM

Dois dos aspetos mais relevantes no desenvolvimento de um sistema de reconhecimento de fala são a extração da informação útil do sinal e a escolha e otimização da arquitetura mais adequada à tarefa. O nosso sistema inicial de reconhecimento de fones – um híbrido MLP/HMM – foi desenvolvido tomando especial atenção a estes dois aspetos.

4.3.1. PARÂMETROS DE ENTRADA DA REDE

Os parâmetros MFCC, PLP, e variantes, são os parâmetros mais usados nos sistemas ASR. Estas parametrizações são conhecidas por reterem a informação acústica mais importante para um reconhecimento de fala eficiente. Contudo, parametrizações alternativas obtidas igualmente ao nível do *front-end* acústico conduziram também a bons resultados de reconhecimento, [5][20]. Investiga-se, nesta secção, a contribuição do uso de parâmetros específicos no reconhecimento de fones quando combinados com os tradicionais MFCC. Combinar as capacidades dos parâmetros MFCC com um conjunto de parâmetros com significado de alguma forma físico tais como vozeamento, espalhamento espetral, etc., é uma forma de explicitamente incorporar informação de detalhes da produção de fala humana no processo de reconhecimento.

Parametrizações diferentes do sinal de fala de entrada podem potencialmente extrair informação adicional útil na melhoria da discriminação entre classes. Como Li refere em [82], os parâmetros MFCC resultam muito bem na classificação de alguns atributos, mas falham noutras situações onde parâmetros temporais podem ser mais discriminativos. Esta informação esteve na origem do presente estudo: explorar a contribuição que outros tipos de parâmetros, quer temporais quer espetrais, podem dar na discriminação ao nível do fone. As redes neuronais admitem todo o tipo de parâmetros de entrada e ajustam-se de forma a encontrar a melhor combinação destes parâmetros na classificação dos padrões de entrada.

Assim, foi testada a combinação de parâmetros de entrada com origem em dois algoritmos de parametrização diferentes: MFCC e um conjunto adicional. O conjunto adicional é composto por 10 parâmetros. Estes parâmetros são os apresentados na Tabela 3.7 (com a única diferença que o logaritmo da energia usado corresponde neste caso a 35ms e não a 5ms como indicado na Tabela 3.7), que já provaram ser úteis na identificação de classes fonéticas alargadas, [94].

O sistema usado para testar a combinação de parâmetros referida consiste num MLP de uma só camada escondida, que treina os 61 fones originais da TIMIT. O sinal de fala foi analisado a cada 10ms usando uma janela de Hamming de 25ms. No treino usou-se uma janela de contexto de 9 *frames*. O desempenho do MLP é avaliado através da taxa de erro de *frames*. No cálculo do FER não são consideradas as *frames* confinantes entre dois modelos adjacentes. Os sistemas em comparação têm sensivelmente o mesmo número de parâmetros livres (124k). Como o número de parâmetros de entrada é diferente: num são usados 39 parâmetros de entrada, representando 12 MFCC, energia e suas 1ª e 2ª derivadas e no outro adicionam-se a estes os 10 parâmetros referidos; o número de nodos na camada escondida é diferente (o MLP dos 39 parâmetros tem 300 nodos e o dos 49 parâmetros tem apenas 250). A evolução da taxa de erros por *frame* em função do número de iterações de treino é apresentada na Figura 4.2.

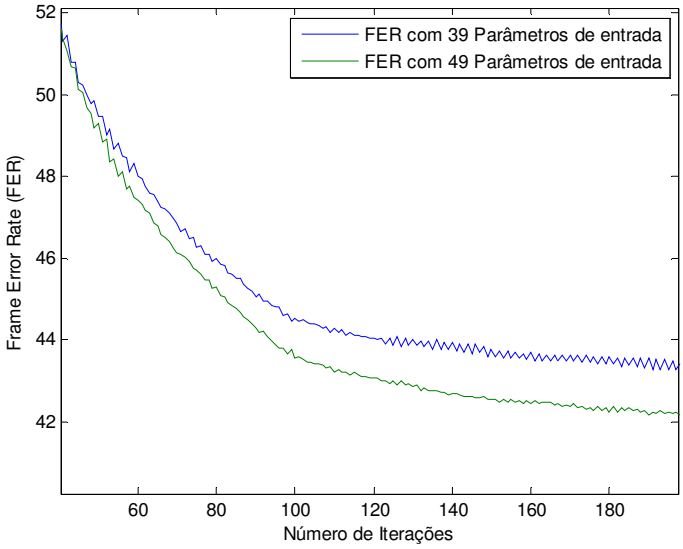


Figura 4.2: Comparação de FER entre dois sistemas: um treinado com 39 parâmetros de entrada (12 MFCC, energia e suas 1ª e 2ª derivadas) e outro treinado com 49 parâmetros de entrada (juntam-se aos anteriores os 10 apresentados na Tabela 3.7).

Em todas as iterações de treino o MLP treinado com 49 parâmetros obteve o melhor desempenho. As melhorias são de 1.3% (3% de melhoria relativa) comparativamente à situação onde se usam os tradicionais 39 parâmetros MFCC, o que indica que o conjunto proposto de parâmetros contribui efetivamente para a discriminação entre classes. Este efeito é mais evidente nas vogais, mas também os silêncios, as plosivas e as nasais viram os seus desempenhos aumentar. Aliás, a maior melhoria foi alcançada por um fone nasal ([nx]) com um valor de 8.6%. A Figura 4.3 apresenta um gráfico onde se compara, por fone, o desempenho dos dois sistemas em análise. Por uma questão de facilidade de análise são só apresentadas as diferenças mais significativas.

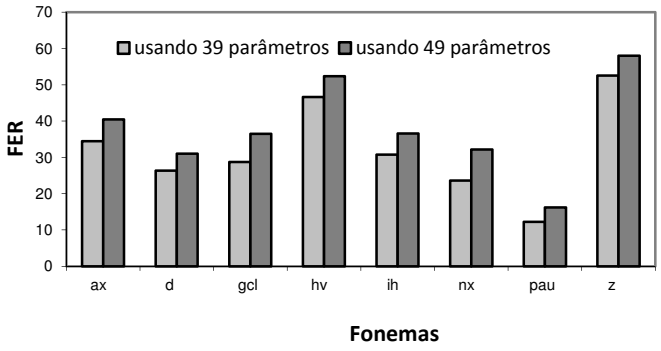


Figura 4.3: Maiores melhorias verificadas na adição dos 10 parâmetros específicos aos tradicionais MFCC.

Conclui-se assim que o conjunto formado pelos 10 parâmetros não inclui a mesma informação que os parâmetros MFCC, constituindo assim uma fonte de informação suplementar capaz de gerar uma melhoria da discriminação ao nível da *frame* de um conjunto de fones.

4.3.2. ALARGAMENTO DO CONTEXTO ACÚSTICO

As representações de tempo curto são amplamente usadas nos sistemas atuais de reconhecimento de fala. Contudo, há indicação de que é possível alcançar melhores resultados de reconhecimento quando é incorporada no sistema informação de termo longo, [20][38][158].

Nas redes neuronais, o uso de uma janela de contexto que engloba várias *frames* de entrada, permite ao sistema aprender, dentro de certos limites, os padrões temporais das unidades de fala. Esta janela de contexto, tipicamente, 50ms de ambos os lados de cada

frame. É normalmente definida com base num compromisso entre o número de parâmetros e o desempenho da rede. A janela associa tipicamente à *frame* atual as 4 *frames* anteriores e 4 *frames* posteriores, incluindo assim informação passada e futura (um total de 9 *frames* com um *frame-rate* de 100 *frames* por segundo). Na Figura 4.4 b) mostra-se graficamente como é constituída a janela de contexto em termos das *frames* usadas. De forma a avaliar o contributo que uma janela temporal maior pode dar, foi também treinado um MLP duplicando o tamanho da janela de contexto. No entanto não se usa o dobro das *frames* como seria de esperar, mas sim as mesmas 9. O intuito é alargar o contexto acústico sem recorrer ao uso de parâmetros adicionais. Por isso usam-se *frames* alternadamente numa estrutura como mostrado na Figura 4.4 c).

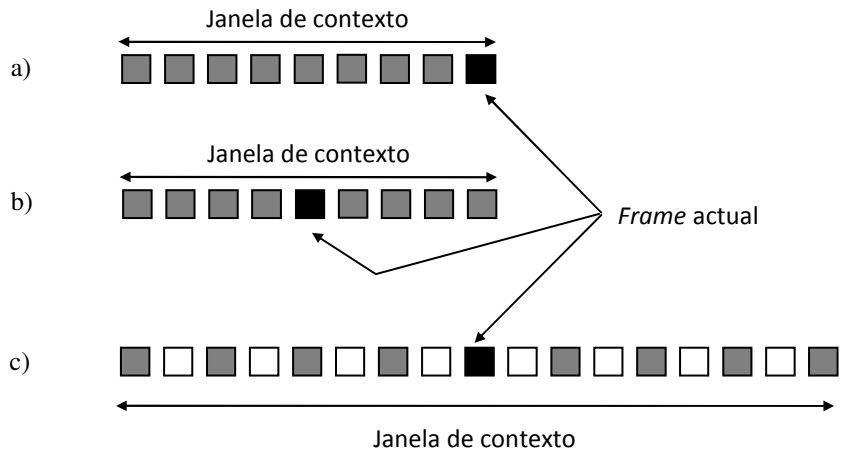


Figura 4.4: Composição de janelas de contexto a usar no MLP compostas por 9 *frames*. a) Janela de contexto causal onde o contexto se compõe à base e informação do passado; b) Janela de contexto onde a *frame* atual é a central: inclui informação do passado e do futuro; c) Janela de contexto proposta: duplica temporalmente o contexto usando o mesmo número de *frames*.

Neste caso o contexto é de 170ms (com um *frame-rate* de 100 *frames* por segundo ou avanço da janela de análise de 10ms). As *frames* não usadas serão usadas como contexto no calculo dos parâmetros da *frame* seguinte. Na Figura 4.4 os quadrados brancos representam as *frames* ignoradas e os cinzentos as consideradas. Desta forma, o número de parâmetros de treino é mantido e abraça-se um contexto temporal muito superior.

Os resultados em termos de FER são apresentados na Figura 4.5. Em todos os testes usaram-se os 49 parâmetros descritos na secção anterior. Comparando os dois MLP que usam informação passada e futura, verifica-se que nas primeiras iterações se alcança um FER mais

baixo usando *frames* contínuas (90ms passado-futuro). No entanto, à medida que o número de iterações cresce a situação inverte-se sendo o desempenho da arquitetura proposta muito superior (ex: na iteração 180, $FER_{170ms\ (passado\text{-}futuro)}$=42.3% enquanto $FER_{90ms\ (passado\text{-}futuro)}$=46.5%).

Comparando o MLP treinado com a arquitetura de contexto alargado a 170ms e o MLP treinado com contexto tradicional (90ms passado) verifica-se que o primeiro apresenta um FER inferior em todas as iterações. As melhorias relativas rondam os 2.8%, o que significa que há de facto vantagem em alargar o contexto acústico, mesmo que a informação usada não seja completa.

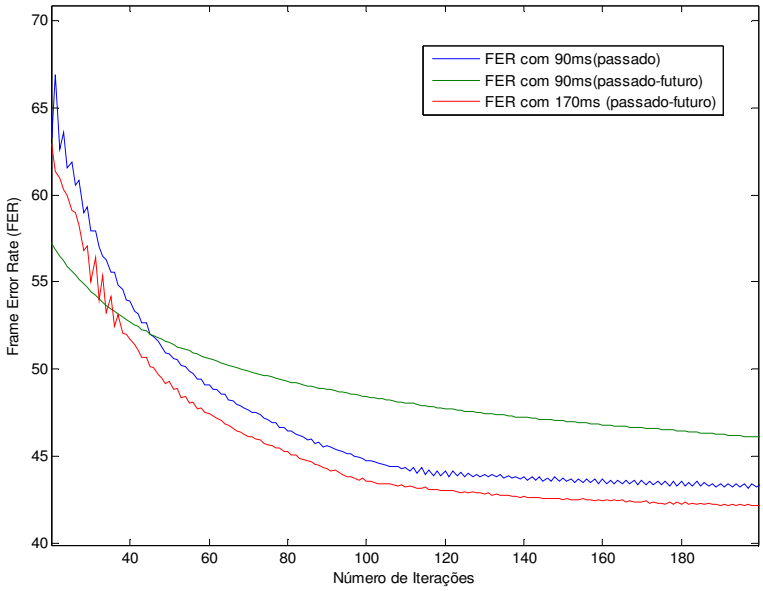


Figura 4.5: Comparação em termos de FER do desempenho de 3 MLPs treinados usando as janelas de contexto apresentadas na Figura 4.4.

De forma a testar a contribuição conjunta do uso do conjunto de 49 parâmetros na (secção 4.3.1) e do alargamento do contexto a 170ms, usando 9 *frames*, (Secção 4.3.2) treinou-se um MLP com uma camada escondida com 250 nodos e uma camada de saída correspondendo aos 61 fones da TIMIT. Após o treino as 61 saídas da rede foram convertidas nos 39 fones propostos por Lee e Hon em [76]. Nas 39 classes obteve-se um FER global de 25.62%. O desempenho deste MLP foi comparado com o trabalho proposto por Scanlon, Ellis e Reilly, [132], apesar da arquitectura usada no trabalho destes autores ser diferente da atual. Apresentam um sistema modular onde cada módulo fornece, para cada *frame*, uma probabilidade de pertença a um dado grupo fonético. Estas probabilidades são embebidas num classificador clássico. Apresentam resultados de FER, mas em termos de 4

grupos fonéticos (vogais, plosivas, fricativas e nasais). A última linha da Tabela 4.9 contém estes resultados para redes com 3000 nodos escondidos. Usando o material da TIMIT e a descrição das classes que eles fornecem na Tabela 1 do seu artigo, foram também calculados resultados, do sistema proposto neste trabalho, em termos de 5 classes alargadas (vogais, plosivas, fricativas, nasais e silêncios). As 61 saídas da rede forma convertidas em 5 classes, somando as saídas que pertencem à mesma classe. Este procedimento difere do proposto em [132], uma vez que neste existem 4 redes operando em paralelo (*frame* pertence/não pertence à classe) enquanto no presente caso existe uma única rede. À parte desta diferença, foram obtidos resultados promissores. Estes resultados encontram-se na Tabela 4.9. Exceto relativamente à classe plosivas, todos os resultados encontrados superam os da outra proposta. No que concerne à classe vogais, resultado do sistema proposto neste trabalho apresenta uma melhoria relativa de 90%, apesar de este valor causar alguma estranheza. Tipicamente a classe de vogais apresenta uma taxa de erro baixa, o que não acontece em [132].

	***Frame* Error Rate (%)**				
	39 Fones	Vogais	Plosivas	Fricativas	Nasais
Proposta do presente trabalho	**25.62**	**4.2**	23.5	**14.3**	**21.1**
Proposta de Scanlon, Ellis e Reilly, [132]	—	40.2	**16.9**	18.6	24.1

Tabela 4.9: Comparação de resultados entre o trabalho apresentado por Scanlon, Ellis e Reilly, [132], e um MLP onde são usados 49 parâmetros na entrada da rede e um alargamento do contexto a 170ms.

Concluiu-se assim que a arquitetura testada tem um bom desempenho no reconhecimento fones. A estratégia de alargar o contexto fonético e combinar dois conjuntos de parâmetros de entrada é benéfica, resultando numa melhoria das taxas de acerto de fones por *frame*.

4.3.3. Erro e Ajuste dos Pesos e Bias

Todos os pesos e *bias* da rede que compreende o reconhecedor híbrido proposto são ajustados usando gradiente descendente e retropropagação resiliente do erro (RProp [121]) de forma a minimizar o erro entre a saída da rede e os *targets* respetivos. A escolha do método teve a ver com a rapidez de convergência à solução ótima. Testou-se o ajuste por via do próprio valor do gradiente (BP), no entanto como as funções de ativação das camadas

internas são sigmoidais[23] fazendo com que o gradiente apresente valores muito pequenos, o método convergia muito lentamente. A Figura 4.6 compara o comportamento de ambos os algoritmos para uma rede que discrimina os 61 fones originais da TIMIT.

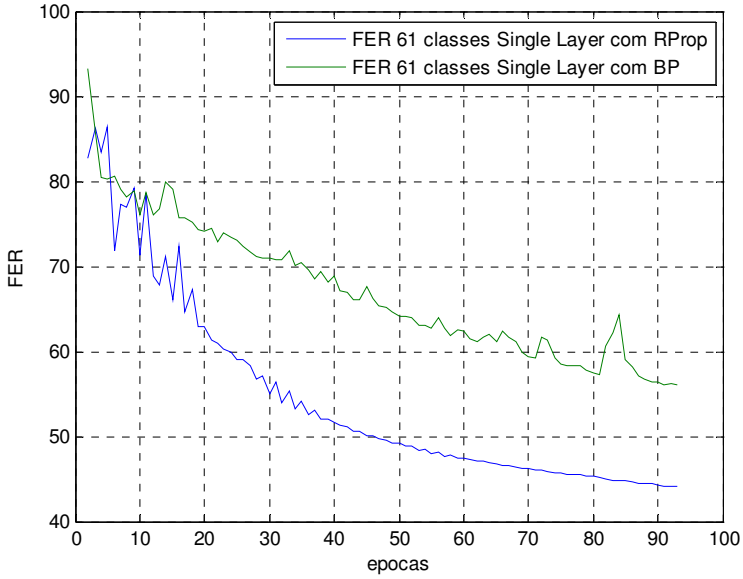


Figura 4.6: Comparação em termos de FER do desempenho de uma rede treinada com backpropagation (ajuste pelo valor do gradiente) e com RProp (ajuste pelo sinal do gradiente).

O erro retropropagado advém da medida de *cross-entropy*, em alternativa ao erro médio quadrático. A escolha da função de erro segue uma sugestão do Bishop, [13] que foi mais tarde clarificada por Dunne, [27] e que refere que a função de ativação *softmax* deve estar associada à função de penalização de entropia cruzada.

4.4. CLASSIFICAÇÃO HIERÁRQUICA DE CLASSES ALARGADAS E DE FONES

Com o intuito de estimular a discussão entre a comunidade científica sobre direções de investigação futuras e contribuições para a perceção do que há ainda a ser feito para alcançar o desempenho humano, Lippmann, [84] comparou o desempenho de humanos e de sistemas de ASR em reconhecimento de fala. Os resultados apresentados mostram que o ouvinte humano tem um desempenho relativo muito superior ao de qualquer sistema automático de então, num ambiente sem ruído e em situações onde não se recorre a gramáticas passíveis de ajudar na discriminação de sons ambíguos. Por exemplo, em

[23] Na camada de saída é usada a função *softmax* como função de ativação de forma a que os valores de saída possam ser interpretados como probabilidades *a posteriori*.

reconhecimento de dígitos e letras o reconhecimento humano chega a estar uma ordem de grandeza acima do sistema ASR. Esta diferença de desempenho (10% de WER em reconhecimento de fala continua na WSJ em várias condições acústicas) deve-se, naturalmente, a um conjunto de aspetos acústico-fonéticos. O estudo aponta o facto de existir no reconhecimento humano uma interação entre vários níveis de representação (acústica, fonémica, fonológica, sintática e semântica), que a ser considerada pelos sistemas ASR poderia contribuir para a melhoria de desempenho dos mesmos. Indo ao encontro desta ideia, propõe-se neste trabalho um sistema onde se reconhecem classes com níveis diferentes de informação fonética, alcançando o detalhe ao nível do fone no último nível. Acredita-se que representações intermédias entre o sinal de fala e a correspondente unidade fonética possam ajudar o reconhecimento final de fones. Uma vez que os sons de fala partilham propriedades acústicas, articulatórias, fonológicas, ou outras, entre si, é natural que se consiga introduzir num sistema ASR informação heterogénea usando diferentes categorias de informação. Esta ideia não é totalmente nova: com perspetivas diferentes, já foi usada informação fonológica, [104][127], informação de classes alargadas, [132], informação articulatória, [79], etc. em sistemas ASR.

Em linha com o trabalho de Scanlon em [132], propõe-se a deteção de várias CFA. No entanto, contrariamente a este trabalho, onde as classes são detetadas por sistemas independentes numa arquitetura paralela, propõe-se incluir informação de vários conjuntos de CFA, com granularidades diferentes, num sistema único onde num nível final se integra esta informação com informação ao nível do fone.

O reconhecimento de classes pode ser feito de forma paralela ou hierárquica (sequencial). Uma estrutura paralela assume que as características de cada classe são independentes, quando na realidade não o são. No entanto, esta consideração evita o problema da propagação de erro que existe quando se usa uma estrutura hierárquica. A representação hierárquica é muito mais eficiente na representação de todos os sons da linguagem, mas propaga erros dos níveis mais altos para os níveis mais baixos.

Na presente arquitetura é usada uma rede MLP, com uma estrutura hierárquica partindo de um detalhe grosso e chegando a um detalhe fino. As predições dos primeiros classificadores são usadas nos classificadores seguintes de forma a melhorar a discriminação entre classes.

O sistema proposto consiste num conjunto de 10 camadas de acordo com a estrutura mostrada na Figura 4.7.

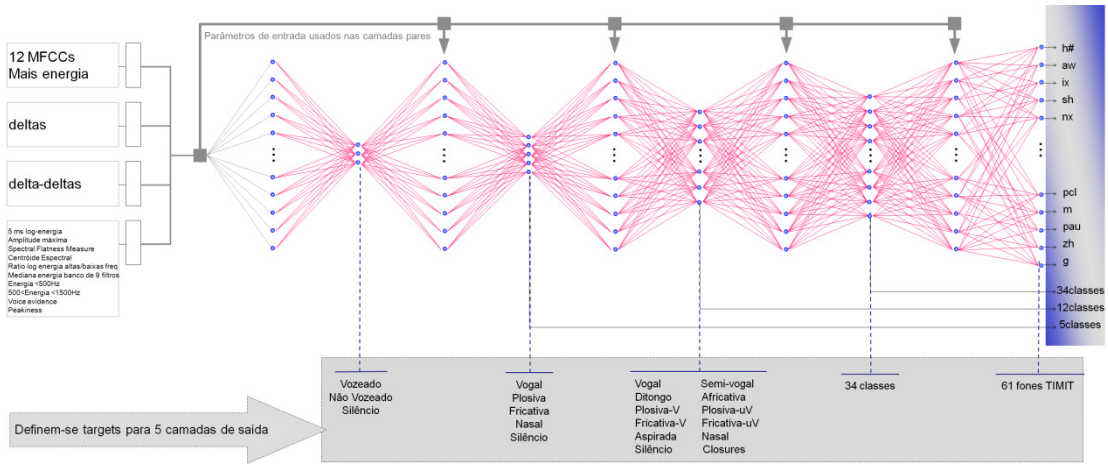


Figura 4.7: Rede MLP hierárquica.

Todas as camadas são treinadas em simultâneo e são fornecidos parâmetros de entrada a todas as camadas ímpares. A rede foi treinada quer em função dos 61 fones originais da TIMIT, quer em função de quatro conjuntos adicionais de CFA com 3, 5, 12 e 34 elementos (são fornecidos *targets* a todas as camadas pares: camadas 2, 4, 6, 8 e 10). O primeiro conjunto classifica o sinal segundo 3 hipóteses: *frame* vozeada, não-vozeada ou silêncio, de acordo com a divisão proposta em [3]. Os outros conjuntos são agrupados de acordo com a divisão apresentada na Tabela 3.4.

A rede foi treinada e aplicada a um modelo híbrido MLP/HMM. A última camada da rede faz uma classificação de 1-a-61 de acordo com o conjunto completo de fones da TIMIT. O desempenho do modelo híbrido, considerando somente os valores da última camada, ficou aquém do desejável, alcançando valores inferiores ao sistema base composto por uma única camada escondida, o mesmo conjunto de parâmetros de entrada e com um número de parâmetros livres aproximadamente igual. Os resultados nos 39 fones da TIMIT foram: 67.0% para *Correctness* e 65.6% para *Accuracy*, aproximadamente 1% inferiores aos do sistema base cujos resultados são de 68.30% para *Correctness* e 66.7% para *Accuracy*. Estes valores encontram-se representados nas duas primeiras linhas da Tabela 4.11 (pag 113). Uma das justificações possíveis para este desempenho assenta na possibilidade de esta subdivisão das CFA não ser a ótima.

Contudo, se forem usadas não só as predições ao nível do fone, mas também as predições ao nível das várias CFA definidas, obtém-se um reconhecedor com melhor desempenho. Esta combinação de informação *classes vs fones* foi feita combinando e pesando as predições das classes de várias formas.

4.4.1. COMBINAÇÃO DE SAÍDAS DA REDE

Embora os métodos de classificação se possam dividir entre série ou paralelo,uma divisão possível de ser feita é entre métodos de combinação treináveis e não treináveis. A Tabela 4.10 inclui um conjunto de métodos de classificação clássicos categorizados pelas suas características.

Método de Combinação	Tipo	Treinável?
Métodos baseados em votação	Paralelo	Não
Boosting	Paralelo	Sim
Bagging	Paralelo	Não
Métodos Algébricos (média, soma, produto, máximo, mínimo, mediana, etc)	Paralelo	Não
Classificação hierárquica (árvores de decisão)	Série	Sim

Tabela 4.10: Categorização de alguns métodos de combinação de classificadores, [4].

Uma descrição mais detalhada de cada um destes métodos (assim como outros) pode ser encontrada em [4][12] e [149].

No âmbito deste trabalho, e de forma a combinar o resultado da camada de saída do MLP (predições dos 61 fones) com os resultados das camadas de saída intermédios (predições das camadas que classificam classes alargadas), foi usado o método algébrico do produto pesado. Resulta na predição final dos fones dada por:

$$\hat{P}(p_k \mid \mathbf{x}) = \frac{1}{Z}\exp\left(\sum_{l=1}^{N_L} \alpha_{c_k}^{(l)} \log(y_{c_k}^{(l)})\right) \quad \textbf{(4.1)}$$

onde $Z = \sum_{j=1}^{K}\exp\left(\sum_{l=1}^{N_L} \alpha_{c_j}^{(l)} \log(y_{c_j}^{(l)})\right)$ é um fator de normalização do tipo *softmax* que faz com que a soma de todas as predições globais dos K=61 fones em causa, seja 1. $\hat{P}(p_k \mid \mathbf{x})$ é a predição da probabilidade do fone k dado o vetor de observação **x** e as predições das classes alargadas nas 4 camadas de saída. N_L=4 é o número dessas camadas (as camadas 4, 6, 8 e 10 na figura anterior, por esta ordem) e $y_{c_k}^{(l)}$ é a saída da rede da camada l para a classe alargada c_k à qual o fone k pertence. Cada fone é predito multiplicando as saídas das 4 classes associadas ao fone k (que são diferentes para cada fone em cada camada) pelo respetivo peso $\alpha_{c_k}^{(l)}$. Por exemplo, a predição global do fone [ay] vai incluir pesos relacionados com a classe *Vogais*, da 1ª camada, com a classe *Ditongos* da 2ª camada e com a classe d3 da 3ª

camada (Tabela 3.4, página 63). Se os pesos forem nulos, exceto o da última camada com valor 1, resulta na situação inicial sem combinação de informação. Se todos os pesos forem 1, a medida proposta em (4.1) corresponde ao produto das 4 saídas:

$$\hat{P}(p_k \mid \mathbf{x}) = \frac{1}{Z}\exp\left(\sum_{l=1}^{N_L}\log(y_{c_k}^{(l)})\right) = \frac{1}{Z}\prod_{l=1}^{N_L} y_{c_k}^{(l)} \qquad \textbf{(4.2)}$$

Usando o produto (e também a soma) os testes mostram que nem todos os fones beneficiam da incorporação das predições de CFA. Este facto esteve na origem da definição de um esquema de pesagem das predições de cada CFA. A pesagem relaciona-se com a qualidade de reconhecimento da classe, entendendo-se qualidade como o valor de uma medida de confiança.

Para uma compreensão mais aprofundada deste problema, foi experimentado um método empírico, antes de se desenvolver um método otimizado de treino discriminativo dos pesos.

Por análise dos resultados, verificou-se que, nas *frames* classificadas como corretas, o valor de saída mais elevado sobressai claramente dos restantes, i.e. a diferença entre o valor mais elevado e o segundo maior aproxima-se da unidade. Pelo contrário, nas situações em que a *frame* é classificada incorretamente, esta diferença é baixa, dando a rede indicação que não é muito evidente qual o fone a que corresponde a *frame*. Assim, aparentemente, a medida da diferença entre os dois valores mais altos da saída do MLP pode ser vista como uma medida da discriminação entre os fones em competição. Quanto mais baixo é o valor do 2º candidato, mais alta é a medida de confiança. Como os valores de saída do MLP derivam de uma função *softmax*, esta medida tem uma gama de variação limitada ao intervalo [0,1]. De acordo com esta heurística, os pesos (em cada *frame*) são dados pela diferença entre os dois maiores valores em cada uma das 4 camadas de saída, aos quais se aplica a função *softmax* de forma a que a soma dos pesos seja unitária.

Para ilustrar este método, apresenta-se um caso real. Na *frame* 44 da locução da TIMIT TEST\DR1\FELC0\SI1386.WAV a saída da última camada com predição mais alta corresponde ao fone [ng], enquanto o fone de referência é o [n], a que corresponde o 2º maior valor de saída. Os dois maiores valores da camada de saída de 61 fones são $y_{[n]}^{(4)} = 0.4039$; $y_{[ng]}^{(4)} = 0.4627$ e a sua diferença 0.0588. As diferenças entre os dois maiores valores em cada camada são

$$\begin{bmatrix} d^{(1)} \\ d^{(2)} \\ d^{(3)} \\ d^{(4)} \end{bmatrix} = \begin{bmatrix} 0.8390 \\ 0.9721 \\ 0.1829 \\ 0.0588 \end{bmatrix} \text{ ; depois de softmax: } \begin{bmatrix} \alpha^{(1)} \\ \alpha^{(2)} \\ \alpha^{(3)} \\ \alpha^{(4)} \end{bmatrix} = \begin{bmatrix} 0.3206 \\ 0.3662 \\ 0.1663 \\ 0.1469 \end{bmatrix}$$

Quer no conjunto de 5 classes quer no de 12, ambos os fones pertencem à mesma classe "nasais" (cf. Tabela 3.4). No conjunto de 34 classes (3ª camada de saída) estão em classes distintas: [n] está na classe "n1" e [ng] na classe "n3". Os respetivos valores de saída são:

Saída [n]	Saída [ng]
$y^{(1)}_{[nasal]} = 0.9032$	$y^{(1)}_{[nasal]} = 0.9032$
$y^{(2)}_{[nasal]} = 0.9831$	$y^{(2)}_{[nasal]} = 0.9831$
$y^{(3)}_{[n1]} = 0.4878$	$y^{(3)}_{[n3]} = 0.3050$
$y^{(4)}_{[n]} = 0.4039$	$y^{(4)}_{[ng]} = 0.4627$

As predições para os dois fones de acordo com o esquema de pesagem são:

$$\hat{P}([n]|\mathbf{x}) \cdot Z = 0.9032^{0.3206} \times 0.9831^{0.3662} \times 0.4878^{0.1663} \times 0.4039^{0.1469} = \mathbf{0.7472}$$
$$\hat{P}([ng]|\mathbf{x}) \cdot Z = 0.9032^{0.3206} \times 0.9831^{0.3662} \times 0.3050^{0.1663} \times 0.4627^{0.1469} = \mathbf{0.7050}$$

o que significa que a aplicação do conjunto de pesos de acordo com (4.1) faz com que o fone vencedor seja agora o fone [n], corrigindo assim o erro que existia na última camada.

Analisando as predições de todas as *frames* cuja classificação não é a correta, conclui-se que em 48 % destas, o 2º candidato é o correto. Naturalmente este valor varia de fone para fone, mas em quase todos atinge um valor expressivo. Na Figura 4.8 apresentam-se estes valores por fone.

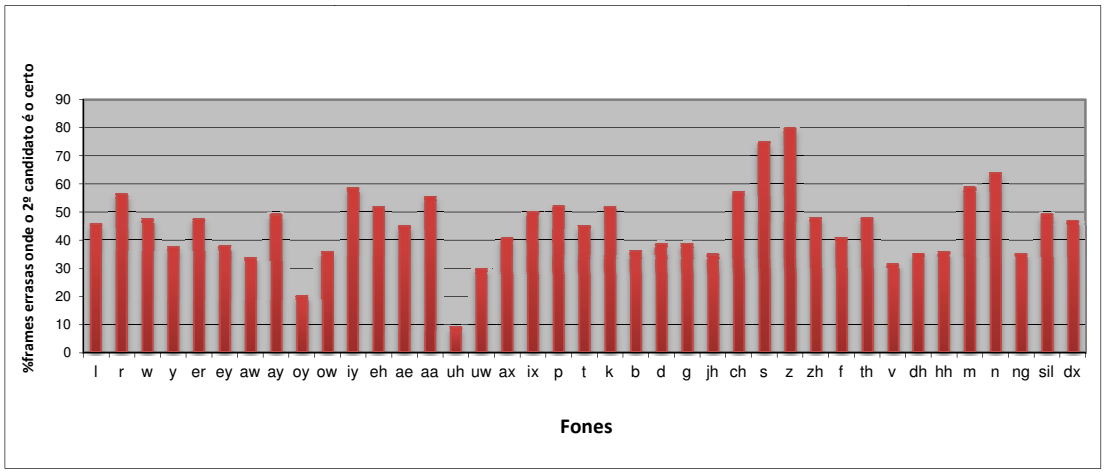


Figura 4.8: Percentagem das *frames* erradas onde o 2º candidato aponta o fone certo.

Apesar de estes valores abrirem espaço para uma possível correção, para tal é necessário que as predições das CFA estejam corretas. Na Figura 4.9 mostra-se a percentagem de vezes em que a predição da camada de saída falha e a predição das CFA acerta. Naturalmente esta percentagem tende a ser mais baixa à medida que a granularidade das classes se torna mais fina. É natural que, na maioria dos fones, mesmo que a saída falhe, o preditor a 5 classes acerte. Já se nos referirmos ao preditor a 34 classes a situação altera-se apresentando valores bastante baixos. Interessante também é a análise apresentada na Figura 4.10 que mostra a situação em que todas as classes alargadas fornecem a informação certa sobre o fone em questão e só a camada de saída falha.

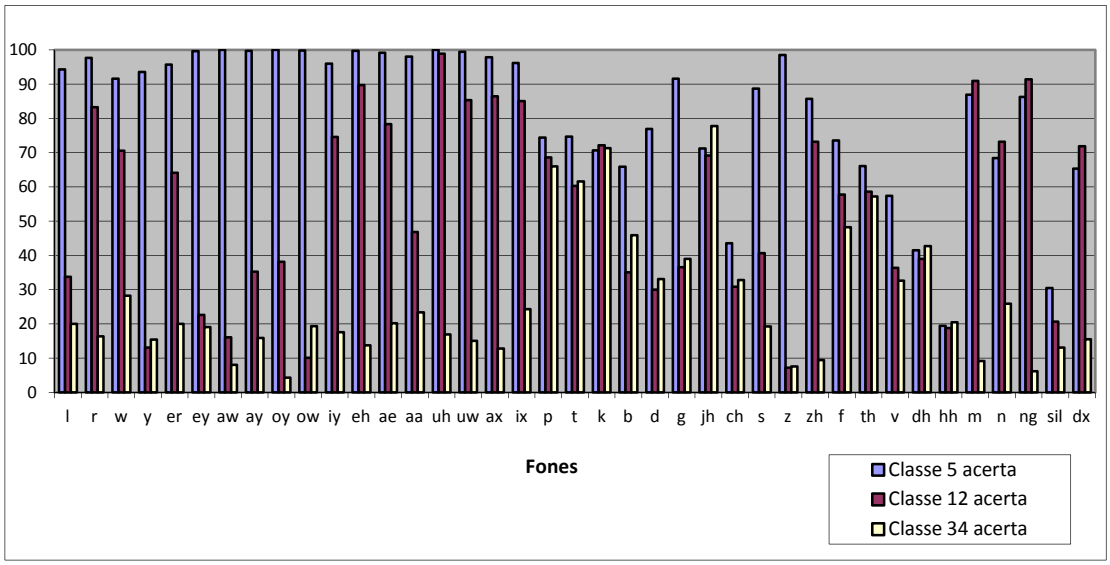


Figura 4.9: Taxas de acerto por classe alargada quando a saída falha.

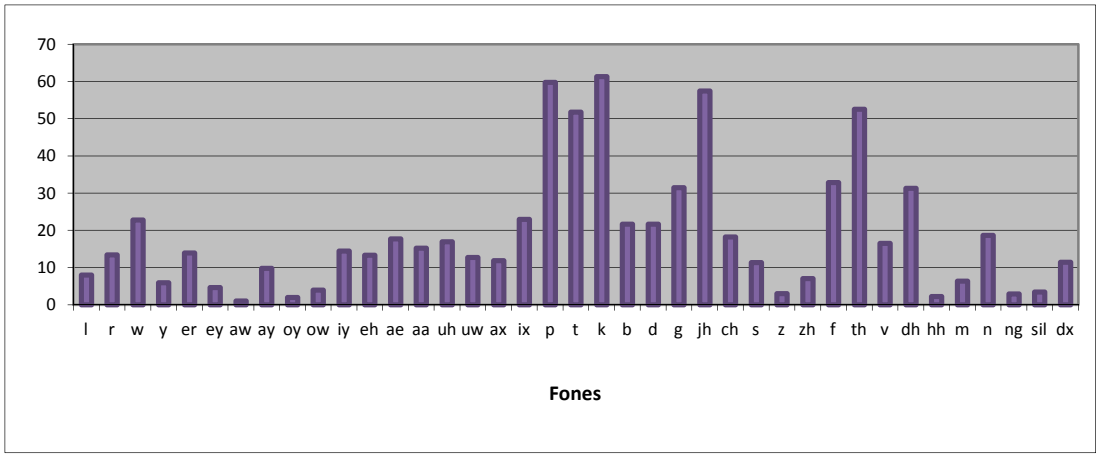


Figura 4.10: Percentagem de *frames* em que a predição de saída falha,
mas todas as classes alargadas dão a predição certa.

De forma a avaliar se a pesagem diferenciada por camada, contribuiria para a melhoria dos resultados, foi calculada a diferença média entre os dois maiores valores em cada uma das 4 camadas de saída, dando origem aos pesos $\alpha^{(1)}$, $\alpha^{(2)}$, $\alpha^{(3)}$ e $\alpha^{(4)}$. As predições finais são dadas por $\hat{P}(p_k \mid \mathbf{x}) = \frac{1}{Z}\exp\left(\sum_{l=1}^{N_L} \alpha^{(l)} \log(y_{c_k}^{(l)})\right)$. À semelhança do exemplo apresentado (fones [n] e [ng]) verificou-se que esta pesagem levou à correção dos valores de predição. No entanto, a pesagem é feita através de um peso único para cada camada de saída da rede. Intuitivamente tudo aponta para que uma pesagem individual de cada saída de cada nível da rede possa resultar ainda em melhores resultados na discriminação entre fones. Assim, propõe-se definir um método treinável de obtenção de pesos individuais por elemento de cada CFA. Para o MLP de múltiplas saídas apresentado na Figura 4.7 é definida uma regra de combinação pesada onde os pesos são treinados em função da *Accuracy* de cada fone no reconhecedor híbrido global. O método, denominado *Treino Discriminativo de Pesos*, é apresentado na secção seguinte.

4.4.2. TREINO DISCRIMINATIVO DOS PESOS DA COMBINAÇÃO DE SAÍDAS

Pretende-se nesta abordagem, encontrar os pesos da combinação (4.1) que maximiza a taxa *Accuracy* do sistema de reconhecimento híbrido. Deverá obedecer a um processo de treino iterativo baseado no paradigma do treino discriminativo, dada a imensurabilidade de possibilidades de combinações de valores de pesos. Sendo o objetivo a maximização da taxa *Accuracy*, consideram-se todos os tipos de erros envolvidos nesta: substituições, inserções e apagamentos. Uma vez que estes erros resultam da aplicação da distância de *Levenshtein*, a função objetivo visa a minimização desta função. No entanto, a distância de *Levenshtein* não é diretamente diferenciável e a função objetivo terá de passar por minimizar o número de erros dados por esta distância. Em primeiro lugar é necessário determinar uma expressão para a função objetivo com base nas saídas da rede neuronal.

Verosimilhança de uma locução

Consideremos $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T\}$ a sequência de vetores de observação que representa uma unidade acústica. Consideremos também um conjunto de *K* HMMs correspondendo aos *K* fones $\{p_1, p_2, ..., p_K\}$. Cada HMM tem 3 estados $\{s_1, s_2, s_3\}$ com matriz de probabilidades de

transição $\mathbf{A}=\left[a_{ij}\right]$. Os estados são caracterizados com $b(p_k\,|\,\mathbf{x})=\left\{b_j\left(p_k\,|\,\mathbf{x}\right)\right\}$, a probabilidade *a posteriori* associada aos estados,

$$b\left(p_k\,|\,\mathbf{x}\right)=\frac{\hat{P}\left(p_k\,|\,\mathbf{x}\right)}{\hat{P}(p_k)} \tag{4.3}$$

de acordo com (2.29). Neste caso, este valor é igual para todos os estados de cada modelo.

Através do algoritmo de Viterbi pode-se obter a melhor sequência de estados $q^*=\{q_1,q_1,...,q_T\}$. Interessa a melhor sequência de modelos/fones, $W=\left\{w_1,w_2,\cdots,w_{N_W}\right\}$, uma vez que dentro de cada modelo a sequência de estados é indiferente. A fim de obter uma expressão da pontuação obtida pelo algoritmo de Viterbi, considere-se o grafo da figura seguinte com arestas l_k que correspondem à verosimilhança de cada fone e com vértices $w_{n_k}(t_k)$ que identificam o fone e a respetiva *frame* final.

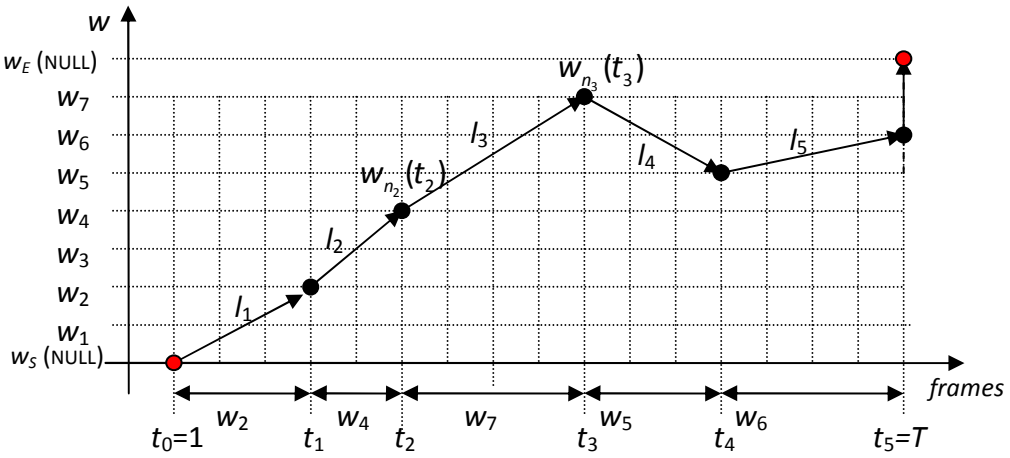


Figura 4.11: Exemplo de descodificação de Viterbi por modelo.

Neste grafo w_S e w_E são os vértices não emissores de começo (Start) e de fim (End). A verosimilhança total da locução é dada por

$$L(W)=\prod_{k=1}^{N_W}P\left(\mathbf{X}_{t_{k-1}}^{t_k}\,|\,w_{n_k}\right)P(w_{n_k}\,|\,w_{n_{k-1}}) \tag{4.4}$$

que corresponde ao produto das verosimilhanças das observações consumidas por cada modelo, pesadas pelo modelo de bigrama, $P(w_{n_k}\,|\,w_{n_{k-1}})$.

Tomando o logaritmo, obtemos

$$g(W)=\sum_{k=1}^{N_W}\left[\log\left(P\left(\mathbf{X}_{t_{k-1}}^{t_k}\mid w_{n_k}\right)\right)+\log\left(P(w_{n_k}\mid w_{n_{k-1}})\right)\right]$$
$$=\sum_{k=1}^{N_W}\left[l_k\left(\mathbf{X}_{t_{k-1}}^{t_k}\right)+\log\left(P(w_{n_k}\mid w_{n_{k-1}})\right)\right] \quad \textbf{(4.5)}$$

onde $l_k\left(\mathbf{X}_{t_{k-1}}^{t_k}\right)$ corresponde à verosimilhança logarítmica associada às arestas do grafo da figura.

Dentro de cada modelo HMM, a verosimilhança logarítmica associada a uma aresta de T_i *frames* pode ser calculada atendendo à Figura 4.12. Uma vez que os 3 estados partilham a mesma *pdf*, podemos assumir, sem perda de generalidade, que só existem duas probabilidades de transição: $a_{ii}=a$ e $a_{ij}=1-a$ para $i\neq j$. Assumimos também a formulação do HTK, na qual existem dois estados não emissores (o estado 1 e o estado 5) para ligação a outros modelos HMM. Desta forma teremos sempre $a_{12}=1$. Assim, o custo de atravessar um só modelo HMM desde $t=t_s$ até $t=t_e$ é:

$$l_k\left(\mathbf{X}_{t_s}^{t_e}\right)=3\log(1-a)+(T_i-3)\log(a)+\sum_{t=1}^{T_i}\log\left(b\left(p_k\mid\mathbf{x}_{t_s+t-1}\right)\right)$$
$$=l_{trans}(a,T_i)+\sum_{t=1}^{T_i}\log\left(b\left(p_k\mid\mathbf{x}_{t_s+t-1}\right)\right) \quad \textbf{(4.6)}$$

Nesta expressão $l_{trans}(a,T_i)$ é o custo das transições; o outro termo é o custo das observações.

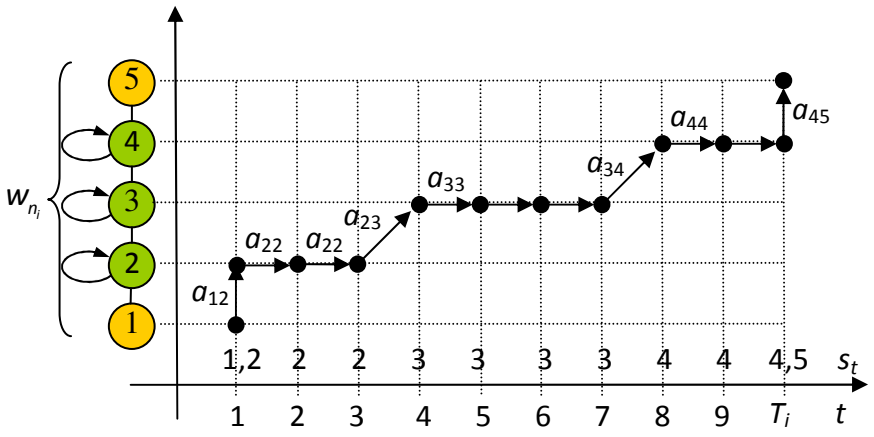


Figura 4.12: Exemplo de descodificação de Viterbi por estado do HMM.

Atendendo a (4.1) e (4.3) teremos

$$\log\left(b\left(p_k\mid\mathbf{x}\right)\right)=\sum_{l=1}^{N_L}\alpha_{c_k}^{(l)}\log y_{c_k}^{(l)}(\mathbf{x})-\log\left(Z\cdot\hat{P}(p_k)\right) \quad \textbf{(4.7)}$$

e

$$
\begin{aligned}
l_k\left(X_{t_s}^{t_e}\right) &= l_{trans}(a,T_i)+\sum_{t=1}^{T_i}\log\left(b\left(p_k \mid \mathbf{x}_{t_{k-1}+t-1}\right)\right) \\
&= \sum_{t=1}^{T_i}\sum_{l=1}^{N_L}\alpha_{c_k}^{(l)}\log y_{c_k}^{(l)}(\mathbf{x}_{t_{k-1}+t-1}) - T_i\log\left(Z\cdot\hat{P}(p_k)\right)+l_{trans}(a,T_i)
\end{aligned}
\tag{4.8}
$$

Daqui resulta a expressão requerida para a pontuação do algoritmo de Viterbi para uma locução segundo a sequência de modelos e instantes do caminho ótimo:

$$
\begin{aligned}
g(W) &= \sum_{k=1}^{N_W}\left[l_k\left(\mathbf{X}_{t_{k-1}}^{t_k}\right)+\log\left(P(w_{n_k}\mid w_{n_{k-1}})\right)\right] \\
&= \sum_{k=1}^{N_W}\sum_{t=1}^{T_k}\sum_{l=1}^{N_L}\alpha_{c_k}^{(l)}\log y_{c_k}^{(l)}(\mathbf{x}_{t_{k-1}+t-1}) \\
&\quad -\sum_{k=1}^{N_W}\left[T_k\log\left(Z\cdot\hat{P}(p_k)\right)+l_{trans}(a^{(k)},T_k)\right] \\
&\quad +\sum_{k=1}^{N_W}\log\left(P(w_{n_k}\mid w_{n_{k-1}})\right)
\end{aligned}
\tag{4.9}
$$

Cada fone w_{n_k} gasta $T_k = t_k - t_{k-1} + 1$ *frames*, tendo o respetivo modelo apenas definida uma probabilidade de transição, $a^{(k)}$.

Função de custo

A distância de *Levenshtein* alinha duas sequências de etiquetas. Uma é a sequência correta, W_{lab}, e a outra é a melhor hipótese de descodificação dada pelo reconhecedor, W_{rec}. Assim, define-se uma função de erro como a diferença de pontuações segundo estes dois caminhos:

$$
d\left(W_{rec},W_{lab}\right) = g(W_{rec}) - g(W_{lab}) \tag{4.10}
$$

$g(W_{lab})$ e $g(W_{rec})$ representam, respetivamente, a pontuação de referência e a pontuação ótima do algoritmo de *Viterbi*. Esta diferença é sempre maior que zero anulando-se unicamente quando as duas transcrições são exatamente iguais (etiquetas e marcas temporais coincidentes).

Se N_{BD} for o número total de locuções de treino, o custo total é dado por:

$$
E = \sum_{n=1}^{N_{BD}} d\left(W_{rec}^{(n)},W_{lab}^{(n)}\right) = \sum_{n=1}^{N_{BD}} e^{(n)} \tag{4.11}
$$

Uma vez definida esta função de custo, podemos escolher o melhor conjunto de pesos, $\alpha_{c_k}^{(l)}$, de forma iterativa, através do método de gradiente descendente:

$$\alpha_{c_k}^{(l)} \leftarrow \alpha_{c_k}^{(l)} - \eta \nabla \alpha_{c_k}^{(l)} . \tag{4.12}$$

Nesta expressão η é uma constante de aprendizagem.

O gradiente $\nabla \alpha_{c_k}^{(l)}$ é dado por:

$$\begin{aligned} \nabla \alpha_{c_k}^{(l)} &= \frac{\partial E}{\partial \alpha_{c_k}^{(l)}} = \sum_{n=1}^{N_{BD}} \left(\frac{\partial g\left(W_{rec}^{(n)}\right)}{\partial \alpha_{c_k}^{(l)}} - \frac{\partial g\left(W_{lab}^{(n)}\right)}{\partial \alpha_{c_k}^{(l)}} \right) \\ &= \sum_{n=1}^{N_{BD}} \frac{\partial g\left(W_{rec}^{(n)}\right)}{\partial \alpha_{c_k}^{(l)}} - \sum_{n=1}^{N_{BD}} \frac{\partial g\left(W_{lab}^{(n)}\right)}{\partial \alpha_{c_k}^{(l)}} \end{aligned} \tag{4.13}$$

Por outro lado,

$$\frac{\partial g(W)}{\partial \alpha_j^{(i)}} = \sum_{k=1}^{N_W} \delta[c_k^{(i)} - j] \sum_{t=1}^{T_k} \log y_j^{(i)}(\mathbf{x}_{t_{k-1}+t-1}) . \tag{4.14}$$

Nesta expressão o termo com o impulso discreto $\delta[c_k^{(i)} - j]$ advém do facto de a saída j poder ser referenciada mais do que uma vez na sequência de fones da locução.

Na Figura 4.13 apresenta-se o método de treino proposto. Os resultados obtidos com este método são apresentados nas secções 4.4.3 e 4.4.4.

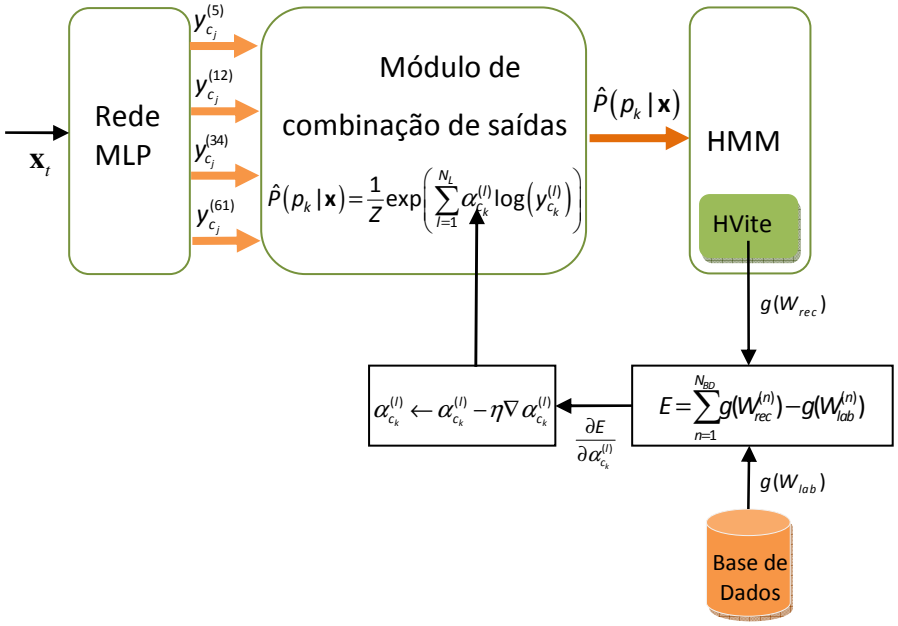


Figura 4.13: Esquema do método de treino dos pesos proposto.

4.4.3. Resultados de Reconhecimento de Fones usando Classificação Hierárquica de Classes obtidas pelo Método Knowledge-Driven

Tal como referido, o sistema consta de um híbrido MLP/HMM no qual a rede é composta por vários níveis de classificação de conjuntos de classes alargadas. Estes conjuntos encontram-se descritos na Tabela 3.4. As predições dos primeiros classificadores são usadas nos classificadores seguintes de forma a melhorar a discriminação entre classes. Na Figura 4.7 mostra-se a estrutura da rede.

O MLP foi treinado quer para classificação de fones quer para classificação de CFA. Foram usados 4 conjuntos de parâmetros. Os três primeiros correspondem aos tradicionais 39 parâmetros MFCC e o quarto ao conjunto definido na Tabela 3.7, perfazendo um total de 49 parâmetros. Usou-se uma janela de contexto de 9 *frames* com a arquitetura definida na Figura 4.4c). Em todos as camadas que fornecem saídas foi usada a função de ativação *softmax* de forma a permitir uma interpretação destes valores em termos probabilísticos. As restantes as camadas usam uma sigmoide como função de ativação. São usados quatros conjuntos de parâmetros de entrada, que em conjunto representam os 49 parâmetros descritos na secção 4.3.1 e que conduziram noutros testes a bons resultados. Todos os pesos e bias são ajustados usando o algoritmo RProp de acordo com o descrito na secção 4.3.3. A rede neuronal tem aproximadamente 85k parâmetros e o número de nodos em cada camada é (por ordem numérica): 50-3-50-5-50-12-50-34-50-61.

A descodificação e a avaliação é feita em termos dos 39 fones propostos por Lee e Hon, [76]. O mapeamento dos 61 fones da TIMIT nos 39 é feito somando as predições dos fones que agrupam, de acordo com a Tabela 4.3. Assim, foram treinados modelos HMM para cada um dos 39 fones usando o HTK 3.4, [162]. A descodificação é igualmente feita usando ferramentas do HTK, mas modificadas de forma a substituir as misturas Gaussianas pelas predições que provêm das saídas do MLP. Cada fone foi modelado por três estados seguindo uma topologia esquerda-direita onde os estados partilham todos o mesmo valor.

Os resultados são apresentados na Tabela 4.11, onde:

- a primeira linha, *Baseline*, se refere a um sistema base constituído por uma rede com uma única camada escondida com aproximadamente o mesmo número de parâmetros da rede da Figura 4.7;
- a segunda linha corresponde a um sistema híbrido onde é usada somente a saída da última camada da rede da Figura 4.7, que fornece probabilidades *a posteriori* para os 61 fones da TIMIT;

- a terceira linha corresponde ao produto das saídas de classes alargadas e probabilidades *a posteriori* de fones, o que pode ser visto como uma probabilidade conjunta destas predições (equação (4.1)). Os resultados mostram que a simples inclusão da classificação de classes alargadas resulta numa melhoria relativa de 9.6% na Correctness e 4.1% na Accuracy, o que abre margem para explorar formas de combinação mais efetivas destes valores;
- a última linha apresenta os resultados obtidos usando os pesos $\alpha_{c_k}^{(l)}$ resultantes da aplicação do método de treino discriminativo de pesos descrito na secção 4.4.2. A taxa Correctness atingiu os 72.4% e a Accuracy 68.9%. O que representa melhorias relativas de 8.1% e 5.1%, respetivamente. Tal como esperado, a melhoria das taxas de reconhecimento foi acompanhada por um decréscimo do erro. À medida que os pesos são atualizados, o alinhamento de Viterbi converge para o alinhamento de referência, o que significa que a função discriminante $g(W_{rec})$ se aproxima de $g(W_{lab})$. Os resultados, apesar de não terem superado os melhores resultados em reconhecimento fonético na TIMIT, mostram que a aplicação de um conjunto de pesos adequado às predições das classes alargadas e sua integração com as predições ao nível do fone podem efetivamente contribuir para uma melhor discriminação fonética.

Pesos				% Corr	% Acc	% Melhoria	
Rede 5 classes	Rede 12 classes	Rede 34 classes	Rede 61 classes			Corr	Acc
Baseline				68.3	66.7		
0	**0**	**0**	**1**	67.0	65.6	-	-
1	**1**	**1**	**1**	73.5	68.3	9.6	4.1
Treino Discriminativo				72.4	68.9	8.1	5.1

Tabela 4.11: Resultados de reconhecimento de fones usando uma rede hierárquica cujas classes intermédias são as obtidas pelo método knowledge-driven.

A rede hierárquica é treinada de forma a fornecer nas suas camadas intermédias saídas de classes alargadas. Haverá vantagem em manter esta estrutura hierárquica ou é preferível treinar redes especializadas no treino destas classes? A resposta passa forçosamente pela comparação em termos de desempenho das duas abordagens. Assim, foram treinadas 4 redes com saídas iguais às das camadas 4, 6, 8 e 10 correspondendo ao treino de 5, 12, 34 e 61 classes. Têm uma só camada escondida (com 50 nodos), e foram treinadas de forma análoga à da rede hierárquica incluindo os mesmos parâmetros de entrada. As redes foram

treinadas várias épocas tendo parado o treino quando em 3 épocas sucessivas o FER se mantinha igual. A soma dos parâmetros destas 4 redes é semelhante ao número de parâmetros da rede hierárquica: ≈85k.

Na Figura 4.14 compara-se a evolução da taxa FER entre as duas tipologias de rede. Em todas as classes se verifica um decréscimo mais rápido do FER usando a rede hierárquica, não atingindo nenhuma das redes de 1 camada os valores de FER da rede hierárquica.

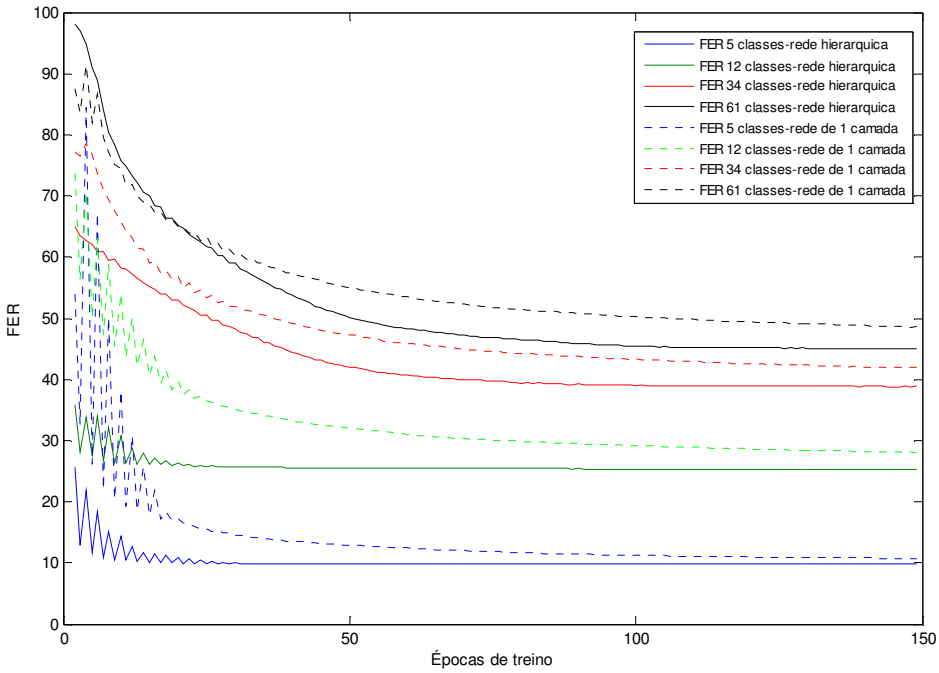


Figura 4.14: Comparação da evolução do FER entre a rede hierárquica e as redes de 1 camada.

A combinação de classificadores apresentada na Tabela 4.11 pode ser igualmente feita com as redes de 1 camada. Os resultados são apresentados na Tabela 4.12 e em todas as situações ficaram aquém dos resultados alcançados pela rede hierárquica, o que vem realçar a vantagem deste tipo de estrutura face ao treino em separado de vários classificadores e posterior combinação de resultados.

Pesos				% Corr	% Acc
Rede 5 classes	Rede 12 classes	Rede 34 classes	Rede 61 classes		
0	**0**	**0**	**1**	59.59	57.86
1	**1**	**1**	**1**	68.55	63.25
Treino Discriminativo				67.08	63.52

Tabela 4.12: Resultados de reconhecimento de fones com redes de 1 camada treinadas em separado para os vários conjuntos de classes alargadas.

4.4.4. Resultados de Reconhecimento de Fones usando Classificação Hierárquica de Classes obtidas pelo Método Confusion-Driven

De forma análoga ao descrito na secção anterior, foi testada uma rede hierárquica com aproximadamente o mesmo número de parâmetros, mas usando a divisão por classes apresentada na Tabela 3.3 que deriva do método automático proposto de agrupamento de fones. Os resultados constam da Tabela 4.13.

Pesos				% Corr	% Acc	% Melhoria	
Rede 9 classes	Rede 16 classes	Rede 40 classes	Rede 61 classes			Corr	Acc
0	**0**	**0**	**1**	71.8	68.6	-	-
Treino Discriminativo				73.8	70.0	2.8	2.0

Tabela 4.13: Resultados de reconhecimento de fones usando uma rede hierárquica cujas classes intermédias são as obtidas pelo método confusion-driven.

Apesar dos resultados não atingem os melhores resultados apresentados na tarefa de reconhecimento de fones na TIMIT (cf Tabela 4.8) mostram que o método contribui para uma melhoria das taxas de reconhecimento. Comparando os resultados de usar só as probabilidades à posteriori dos 61 fones (1ª linha) com os resultados da aplicação do método de treino de pesos (2ª linha) verifica-se que a Correctness sobe de 71.8% para 73.8%, o que representa uma acréscimo relativo de 2.8%. Em relação à Accuracy, a melhoria relativa foi de 2% alcançando os 70%. Estes resultados comparam-se favoravelmente com os apresentados pelo grupo ASAT em [16] e por Morris e Fosler-Lussier em [104]. Estes trabalhos têm em comum com o presente, o facto de recorrem também a informação de CFA e apresentarem resultados nas mesmas condições de avaliação (mesmo material de fala e mesmas taxas de reconhecimento). O grupo ASAT usa medidas de atributos de CFA provenientes de um MLP, de um HMM e de uma SVM num CRF para o reconhecimento de fones. Morris e Fosler-Lussier usam parâmetros fonológicos (vindos de uma ANN) em conjunto com probabilidades *a posteriori* de 61 classes (dadas por outra ANN) também como entrada de um CRF.

4.5. MÉTODO DE TREINO DISCRIMINATIVO GLOBAL DE RECONHECEDORES HÍBRIDOS

Os reconhecedores híbridos incluem normalmente uma etapa de classificação seguida de uma etapa de alinhamento. Contudo, treinar um sistema híbrido deste tipo não é simples, o que justifica que habitualmente se faça o treino da classificação e do alinhamento separadamente. A simplicidade desta abordagem contrasta com a carência de um esquema global de otimização para o sistema completo, no qual o desempenho do sistema seja otimizado. Assim, um desafio passa pelo desenvolvimento de um método de treino discriminativo que treine sistemas híbridos como um todo. Segue-se uma breve descrição do estado da arte neste domínio.

Bengio *et al* [11], já abordaram o assunto relativamente a um híbrido ANN/HMM. A ANN é treinada de forma a representar um "compactador" de características para um CDHMM. As saídas da rede neuronal são então usadas como entradas do HMM (e não como probabilidades associadas aos estados do HMM, como no nosso caso). Primeiro treinam só a ANN com os valores desejados (*targets*) na camada de saída. Depois treinam o HMM com o algoritmo Baum-Welch usando as saídas da ANN como parâmetros de entrada. A otimização global é feita num passo final onde os pesos da rede são atualizados através do algoritmo do gradiente descendente tomando as derivadas parciais da verosimilhança das observações acústicas dado o modelo. Uma vez que os estados dos HMMs são modulados por GMMs o treino global centra-se essencialmente na atualização das médias, variâncias e matrizes de transição dos HMMs. O método de otimização global foi testado com sucesso na caracterização de plosivas.

Droppo e Acero propuseram em [26] um método global de treino de um sistema que, não sendo um híbrido, é também composto por 2 etapas: uma etapa de extração de parâmetros de entrada e uma segunda etapa de modelação acústica com HMMs. Numa tentativa de não perder informação na etapa de extração de parâmetros, propõem uma otimização global do sistema de forma que a modelação acústica não esteja restrita ao uso de parâmetros pré fixados. Na aprendizagem global é usado o método Rprop,[121] originalmente desenvolvido para o treino de ANNs, para encontrar os valores dos parâmetros de entrada que otimizam uma função objetivo baseada em MMI. O HMM é inicialmente treinado com ML e só depois é que o método global é aplicado. No entanto, só as médias são atualizadas (pesos das misturas, variâncias e matrizes de transição são mantidas constantes).

Em [122], Riis e Krogh apresentam um híbrido ANN/HMM, que denominam HNN (*Hidden Neural Network*) no qual todos os parâmetros são treinados em simultâneo de acordo com o critério CML (Conditional Maximum Likelihood). As ANNs são usadas na estimação de parâmetros dos HMM. Para contornar a exigência computacional do treino de misturas gaussianas, treinam as probabilidades associadas aos estados dos HMMs através de redes ANN de uma só saída (cada estado dos HMMs tem uma rede própria). A actualização dos pesos baseia-se no algoritmo do gradiente descendente e os erros são calculados através do algoritmo forward backward de forma análoga à usada por Bengio *et al* [11]. O sistema foi testado na tarefa de reconhecimento de fones na TIMIT tendo alcançado 69% de accuracy.

No caso presente, propõe-se um método de treino discriminativo aplicado a um reconhecedor de fones híbrido ANN/HMM. A ANN consiste numa rede MLP cujas saídas representam probabilidades *a posteriori* de ocorrência de fones e são usadas como probabilidades de ocupação dos estados dos HMMs. É definido um esquema de aprendizagem global por retropropagação considerando uma total integração entre o HMM e a ANN. A minimização do erro é baseada no algoritmo do gradiente descendente e o resultado é uma maximização da *Accuracy* ao nível do fone tal como descrito na secção anterior. A ideia é semelhante à do treino por *Minimum Phone Error* (MPE) e *Minimum Word Error*. O MPE foi introduzido por Povey em [115][116] e centra-se, não no erro de classificação mas, nas taxas de acerto finais (de fones e de palavras). O critério de treino passa pela minimização dos erros de descodificação ao nível do fone. A função objetivo é a soma de aproximações da Accuracy dos fones de todas as hipóteses possíveis, dada a referência, pesada pela verosimilhança de cada hipótese em função do modelo, [115].
O critério MPE permite assim incluir estimativas dos erros de descodificação diretamente no processo de treino, no entanto, o MPE é aplicado ao treino de HMMs e não a um modelo híbrido.

4.5.1. MÉTODO DE TREINO DISCRIMINATIVO GLOBAL

O método de treino discriminativo global (MTDG) que se propõe treina, como referido, os parâmetros de um modelo híbrido MLP/HMM, não em duas etapas como usual, mas como um todo. O MLP é por si uma estrutura de natureza discriminativa; contudo os pesos da rede são habitualmente atualizados de acordo com os valores desejados (*targets*) apresentados na camada de saída, *frame*-a-*frame*, quando deveriam ser atualizados de acordo com a

melhor sequência de estados do HMM. Neste sentido propõe-se um método de treino baseado numa função de custo que minimiza o erro de classificação operando ao nível do reconhecimento. Trata-se de uma abordagem em tudo semelhante à da secção anterior, mas agora usada para treinar os parâmetros do modelo híbrido, nomeadamente os pesos da rede neuronal. Os parâmetros livres do sistema são atualizados de acordo com as classificações erradas entre a sequência otima de saída (otimização 1-best) e a sequência de etiquetas de referência. A Figura 4.15 ilustra o método proposto: as diferenças entre a sequência de saída do reconhecedor e a sequência de referência são usadas para atualizar as matrizes de transição dos HMMs e os pesos da rede neuronal.

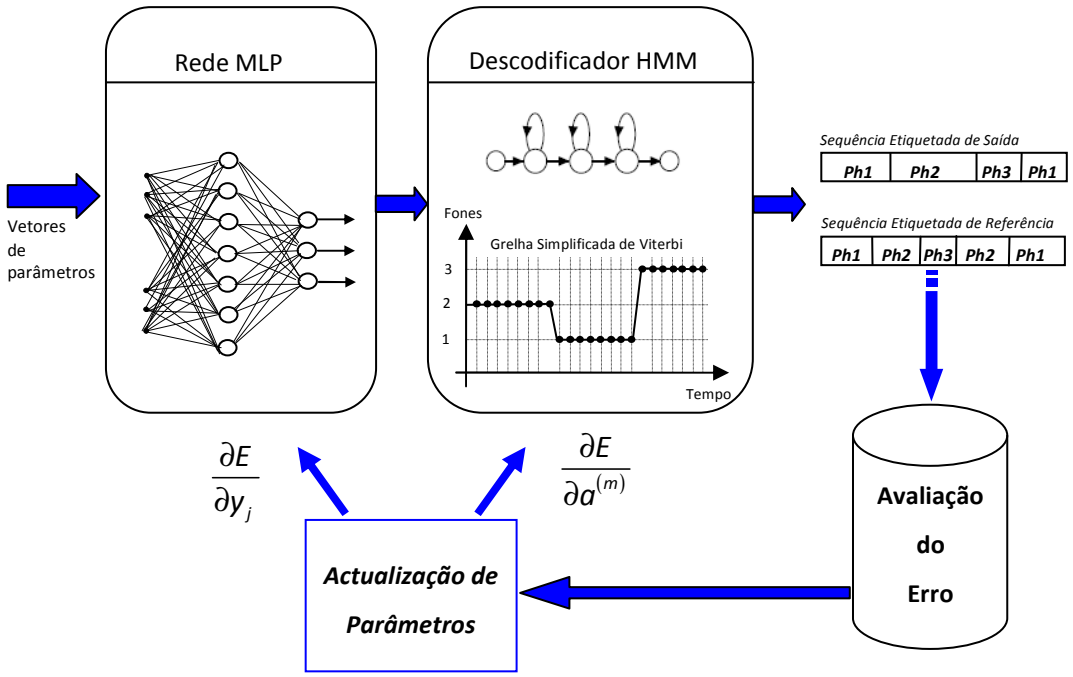


Figura 4.15: Esquema do método de treino discriminativo global proposto.

A originalidade introduzida pelo método advém do cálculo do gradiente da função de custo em ordem às saídas do MLP e a posterior propagação deste gradiente através de toda a estrutura até à primeira camada da rede. O alinhamento de saída fornecido pela grelha de Viterbi é decisivo no processo de treino uma vez que o gradiente da função de custo em ordem às saídas do MLP é calculado com base neste alinhamento.

A expressão da verosimilhança logarítmica para uma sequência *W* é idêntica a (4.4), com a alteração da predição da probabilidade *a posteriori* dos fones que é agora simplesmente:

$$\hat{P}(p_k \mid \mathbf{x}) = y_k \qquad \textbf{(4.15)}$$

a saída *k* da última camada da rede, calculada segundo (2.12). Resulta assim a seguinte expressão:

$$\begin{aligned} g(W) = & \sum_{k=1}^{N_W} \sum_{t=1}^{T_k} \log y_k(\mathbf{x}_{t_{k-1}+t-1}) \\ & - \sum_{k=1}^{N_W} \left[T_k \log\left(Z \cdot \hat{P}(p_k) \right) + l_{trans}(a^{(k)}, T_k) \right] \\ & + \sum_{k=1}^{N_W} \log\left(P(w_{n_k} \mid w_{n_{k-1}}) \right) \end{aligned} \qquad \textbf{(4.16)}$$

A expressão do custo global é a mesma que (4.10) e (4.11). As probabilidades *a priori* dos fones $P(w_k)$ são estimadas previamente a partir dos dados de treino. No modelo híbrido os parâmetros livres são os pesos e *bias* da rede neuronal e as probabilidades de transição nas cadeias de Markov dos HMM.

4.5.2. GRADIENTE EM ORDEM ÀS SAÍDAS DO MLP

A atualização dos pesos da rede é feita pelo método do gradiente descendente. Neste caso o gradiente do erro para a saída y_j do MLP (fone/modelo w_j) é:

$$\frac{\partial E}{\partial y_j} = \sum_{n=1}^{N_{BD}} \frac{\partial e^{(n)}}{\partial y_j}, \qquad \textbf{(4.17)}$$

sendo

$$\frac{\partial e^{(n)}}{\partial y_j} = \left(\sum_{t=1}^{T_n} \delta\left[j - w_t^{(rec)} \right] \frac{1}{y_j(\mathbf{x}_t)} \right) - \left(\sum_{t=1}^{T_n} \delta\left[j - w_t^{(lab)} \right] \frac{1}{y_j(\mathbf{x}_t)} \right) \qquad \textbf{(4.18)}$$

onde $w_t^{(rec)}$ e $w_t^{(lab)}$ são o índice do fone observado na *frame t* na sequência de fones do alinhamento de Viterbi e na de referência, respetivamente, e onde a locução *n* da base de dados tem T_n observações. O impulso discreto $\delta[i-j]$ vale 1 sse *i=j* e zero caso contrário, o que equivale ao símbolo delta de Kronecker, δ_{ij}. De notar que sempre que ambas as sequências coincidam no modelo ocupado numa dada *frame*, não existe acumulação de erro. Quando existe uma *frame* mal classificada (um fone $w_t^{(rec)}$ diferente de $w_t^{(lab)}$), o gradiente compõe-se de contribuições de duas saídas, com sinais opostos. A saída correspondente à referência (lab), contribuirá com um valor negativo e a saída errada (rec) com um valor positivo. Isto serve para indicar à rede que deve aumentar ou diminuir,

respetivamente, os seus valores de saída, de acordo com o algoritmo do gradiente descendente. A Figura 4.16 pretende ilustrar o mecanismo descrito considerando o reconhecimento de quatro fones. REF corresponde à locução de referência e REC ao resultado de reconhecimento. Se ocorre um erro (classificação errada ou desalinhamento temporal) será dada uma indicação ao MLP para subir ou descer duas saídas.

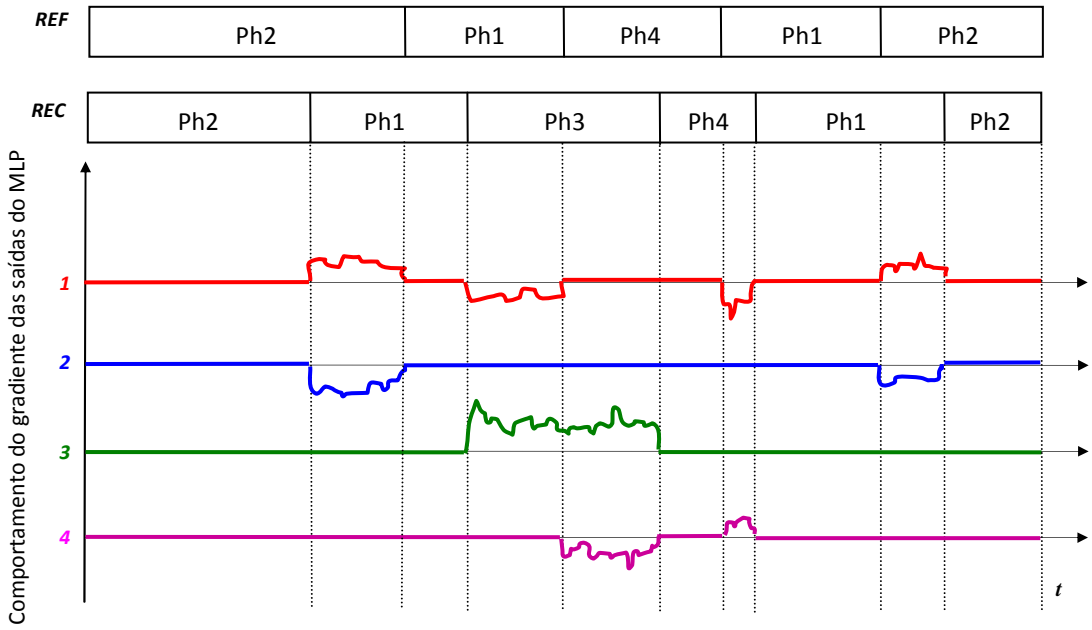


Figura 4.16: Exemplo do gradiente do erro para cada saída do MLP, na presença de classificações erradas ou com desalinhamentos temporais.

Fazendo um contraponto com o paradigma *Minimum Classification Error* (MCE) o que se verifica é que este último atualiza a saída correspondente à referência (*target*≠0) num valor correspondente a $1/y_j$. O que corresponde a uma só parcela da expressão (4.18) e o que torna a convergência deste método mais lenta que a do método proposto.

Outro ponto interessante é que a função de custo baseada em entropia cruzada, usada no treino comum dos MLP com *targets frame*-a-*frame* também tem gradientes inversamente proporcionais às saídas (cf. com (2.18)). Contudo, é calculado um termo de gradiente para cada *frame*, usando a todas as saídas com valor desejado 1, o que contrasta com a presente função de custo global que só "culpa" as saídas do MLP quando ocorre um desalinhamento entre as sequências de fones da referência e do alinhamento de Viterbi.

Depois de calcular o gradiente da função de custo em ordem às saídas do MLP, estes gradientes são simplesmente propagados através da estrutura da rede, até atingirem a

primeira camada. Tal como referido em secções anteriores, de forma a acelerar a convergência a uma solução, foi usado o algoritmo RProp.

4.5.3. GRADIENTE EM ORDEM AOS PARÂMETROS DO HMM

Nos HMMs dos fones do modelo híbrido, os únicos parâmetros atualizáveis são as transições entre estados, $a_{ij}^{(m)}$, da cadeia de Markov do modelo *m*. A expressão para o gradiente do erro é

$$\frac{\partial E}{\partial a_{ij}^{(m)}} = \sum_{n=1}^{N_{BD}} \frac{\partial e^{(n)}}{\partial a_{ij}^{(m)}} . \tag{4.19}$$

sendo,

$$\begin{aligned} \frac{\partial e^{(n)}}{\partial a_{ij}^{(m)}} = &+ \sum_{k=1}^{N_W^{(rec)}} \delta[k-m] \left(\sum_{t=1}^{T_n} \delta\left[i - s_{t-1}^{(rec)} \right] \delta\left[j - s_t^{(rec)} \right] \frac{1}{a_{ij}^{(m)}} \right) \\ &- \sum_{k=1}^{N_W^{(lab)}} \delta[k-m] \left(\sum_{t=1}^{T_n} \delta\left[i - s_{t-1}^{(lab)} \right] \delta\left[j - s_t^{(lab)} \right] \frac{1}{a_{ij}^{(m)}} \right) \end{aligned} \tag{4.20}$$

Esta expressão conta todas as ocorrências do modelo *m* na sequência e, quando acontece, conta os desalinhamentos entre estados. Considerando os valores a_{ij} iguais, $a_{ij}^{(m)} = a^{(m)}$, obtemos, usando $l_{trans}(a^{(m)}, T_k)$ em (4.6):

$$\frac{\partial E}{\partial a^m} = \sum_{k=1}^{N_W^{(rec)}} \delta[k-m] \left[\frac{T_k - 3}{a^m} - \frac{3}{1-a^m} \right] - \sum_{k=1}^{N_W^{(lab)}} \delta[k-m] \left[\frac{T_k - 3}{a^m} - \frac{3}{1-a^m} \right] \tag{4.21}$$

O método proposto foi testado com sucesso usando dois conjuntos de fala: fala Inglesa da base de dados TIMIT, [36], e fala em Português Europeu da base de dados Tecnovoz, [98]. Os resultados relativos ao primeiro conjunto são apresentados na secção 4.5.4. e os resultados em Português Europeu encontram-se em [87].

4.5.4. RESULTADOS DO MÉTODO DE TREINO GLOBAL DE UM HÍBRIDO

A análise de desempenho do MTDG foi feita com base num reconhecedor de fones híbrido MLP/HMM treinado usando a base de dados TIMIT. A rede é semelhante às descritas em

secções anteriores no que concerne à análise do sinal, ao tipo de parâmetros usados, contexto usado, função de ativação intermédia e de saída e método de treino. Os conjuntos de treino e teste são os mesmos que nos restantes testes descritos no presente trabalho. A camada escondida tem 1000 nodos e a camada de saída tem tantas saídas quantas os fones que pretende classificar. Foram realizados testes com 39 fones e com 61 na camada de saída que se designam: Baseline61 e Baseline39 (resultados nas duas primeiras linhas da Tabela 4.14). Entenda-se por Baseline um sistema híbrido MLP/HMM onde o treino da rede neuronal e do HMM são feitos em separado. Os sistemas atingiram resultados semelhantes. A Baseline39 atingiu uma *Correctness* de 72,79% e uma *Accuracy* de 69,52% enquanto a Baseline 61 atingiu 72,46% e 69,60% nas taxas correspondentes.

	%Corr	%Acc	Melhorias relativas	
			%Corr	%Acc
Baseline39	72.79	69.52		
Baseline61	72.46	69.60	-	-
MTDG-MLP39/HMM	73.94	70.30	1.6	1.1
MTDG-MLP61/HMM	73.83	70.27	1.9	1.0

Tabela 4.14: Reconhecimento de fones na TIMIT usando o método de treino discriminativo global (MTDG).

De forma a avaliar a capacidade de treino do MTDG proposto e também uma convergência mais rápida para uma solução, adotou-se como ponto de partida os sistemas MLP e HMM previamente treinados nas *baselines*, de forma análoga ao realizado em [11].

Aos híbridos treinados com o algoritmo de treino discriminativo global atribuímos a designação de MTDG-MLP39/HMM e MTDG-MLP61/HMM. Os resultados constam das duas últimas linhas da Tabela 4.14, e indicam melhorias quer de *Correctness* quer de *Accuracy* comparando com as Baselines correspondentes. Usando 39 fones a *Correctness* subiu aos 73.94%, enquanto com 61 fones atingiu 73.84% o que equivale a uma melhoria de 1,37% (1,9% de melhoria relativa). Em relação à taxa *Accuracy* as melhorias não são tão expressivas (rondam 1% de melhoria relativa) em ambas as situações.

A abordagem típica, e seguida neste trabalho, para a minimização das funções de custo, baseia-se em métodos de gradiente descendente. Estes algoritmos são fáceis de implementar e produzem resultados efetivos, no entanto, são lentos e apresentam dificuldade na estimação do parâmetro de aprendizagem. Na sequência deste estudo

investigámos um novo método de treino, *o Fast Minimum Error Training* (FMET), [142], que apresenta uma redução muito significativa no tempo de treino e permite uma melhoria significativa na performance do reconhecedor quando comparado com o RProp. Trata-se de um método de treino discriminativo de modelos para reconhecimento de fala baseado em MCE envolvendo não só a sequência otima de saída (1-best) mas N sequências (N-best).

CAPÍTULO 5

CONCLUSÕES E DIREÇÕES FUTURAS

O estudo apresentado permitiu tirar conclusões de acordo com o objetivo a que se propunha: avaliar a contribuição que níveis diferentes de informação fonética, podem dar no reconhecimento ao nível do fone. As conclusões obtidas acentuam a ideia apresentada por outros autores [79][104][127][132], que o recurso a informação fonológica, informação de classes alargadas, informação articulatória, etc. é uma mais-valia para o processo de reconhecimento. Neste trabalho seguiu-se uma abordagem um pouco diferente na medida em que esta informação está embebida no próprio processo de descodificação e não é usada como parâmetros adicionais como tinha sido feito até então. Além disso, a informação usada não advém de uma análise articulatória ou fonológica. É o próprio reconhecedor a criar e a identificar as classes semelhantes entre si.

Mostrou-se que, conjugando os resultados de reconhecimento de várias classes fonéticas alargadas (eventos) com os resultados de reconhecimento de fones, se conseguem corrigir erros que ocorrem ao nível do reconhecimento de fones. A conjugação foi feita pesando informação de vários conjuntos de classes fonéticas alargadas (conjuntos de classes com uma granularidade decrescente partindo de classes amplas até ao detalhe do fone). Dada a imensurabilidade de possibilidades de combinações de valores de pesos, propôs-se encontrar o conjunto de pesos ótimo através de um processo de treino iterativo baseado no paradigma do treino discriminativo que maximizasse a taxa *Accuracy* do sistema de reconhecimento. Os testes validaram a proposta e mostraram que um sistema de reconhecimento de fones beneficia desta estratégia. O método foi implementado e testado usando um sistema híbrido MLP/HMM, mas é facilmente adaptável a outro tipo de sistemas híbridos. A opção da escolha deste sistema híbrido surgiu no seguimento de dois fatores. Por um lado, os melhores resultados encontrados na literatura no que se refere ao

reconhecimento de fones na TIMIT (conjunto completo de teste) , [143], têm origem em sistemas que combinam HMMs com redes neuronais. Por outro lado, o estudo sobre a eficácia da deteção de eventos efetuada no âmbito deste trabalho conduziu também a esta abordagem: comparados os vários sistemas híbridos, SVM/HMM, SVM/NMD e MLP/HMM, [96], concluiu-se que os melhores resultados são alcançados com este último. Daí que também a estrutura do sistema global, onde se combina informação de classes alargadas com informação ao nível do fone, passe por um híbrido que combina a capacidade de modelação temporal dos HMMs com a capacidade discriminativa típica das redes neuronais. A otimização desta estrutura foi, no decorrer deste trabalho, conseguida por via de uma estratégia de treino global de um híbrido. Face à inexistência de métodos específicos de treino de sistemas híbridos MLP/HMM, foi proposto e testado, com sucesso, um método de treino global. O método compara os resultados de reconhecimento (em termos de sequência de fones) com as sequências de referência. As diferenças (erros) são usadas para atualizar quer os modelos HMM quer os pesos da rede neuronal. A função de custo definida, que minimiza o erro global de classificação, mostrou contribuir para uma melhor discriminação entre fones permitindo melhorar as taxas de reconhecimento do reconhecedor de fones.

Tal como referido, a questão central deste trabalho, recai na contribuição que informação da presença de classes fonéticas alargadas pode dar na melhoria do reconhecimento de fones. Tradicionalmente, estas classes são definidas *a priori*, por linguistas ou peritos em análise acústica conjugando diversos tipos de informação. Neste trabalho desenvolveu-se um método automático de agrupamento de fones baseado numa métrica de semelhança entre fones. A semelhança é calculada a partir de uma matriz de confusão obtida à saída de um reconhecedor automático de fones. No que respeita às classes alargadas obtidas automaticamente, os resultados foram interessantes, mostrando que se consegue alcançar o mesmo desempenho de reconhecimento (ao nível da *frame*) no reconhecimento das classes obtidas automaticamente que se obtém no reconhecimento das classes formadas por peritos humanos. A grande potencialidade do método refere-se à sua versatilidade. Além de não estar sujeito a subjetividade, não obrigar à existência de recursos humanos especializados nem requerer tempo de análise, o método permite gerar classes de outras unidades que não o fone o que manualmente seria muito complicado. Outra vantagem concerne com a possibilidade de generalização e aplicação a qualquer tipo de língua e a sistemas multilingue. Há que referir, no entanto, que o método depende do desempenho do

reconhecedor inicial, uma vez que a métrica de semelhança entre unidades é calculada a partir destes resultados de reconhecimento. Assim, devem ser usados modelos acústicos de boa qualidade no reconhecedor inicial para que a matriz de confusão reflita confusões representativas.

Apesar de, ao nível da *frame*, os resultados usando classes obtidas manualmente e automaticamente serem semelhantes, as classes obtidas automaticamente viriam a demonstrar ser mais úteis quando incluídas no processo de reconhecimento global. Entende-se por processo global, uma estrutura hierárquica que treina em simultâneo fones e classes fonéticas alargadas. Tipicamente as classes alargadas e os fones são treinados por sistemas separados. No decurso do trabalho que se apresenta, foram testadas e comparadas ambas as soluções (treino de todos os classificadores em separado e treino hierárquico incluindo o treino simultâneo de todas as classes) tendo o sistema hierárquico evidenciado superioridade nas taxas de reconhecimento ao nível da *frame*. Em consequência, foi projetada uma estrutura hierárquica onde os níveis intermédios são treinados segundo as classes geradas pelo procedimento automático de agrupamento e onde o último nível corresponde a fones. A integração da informação de cada elemento de cada CFA (em todos os níveis) com as probabilidades dos fones fornecidas pelo último nível da estrutura, conduziu a uma melhoria das taxas de reconhecimento dos fones. As taxas *Correctness* e *Accuracy* apresentam melhorias de 2.8% e 2%, respetivamente, comparando com um sistema base que usa só as probabilidades dos fones fornecidas na camada de saída da estrutura hierárquica. Concluiu-se também que se atinge o melhor resultado usando as classes obtidas através do método automático de *clustering*. Se forem usadas as classes obtidas manualmente alcançamos uma taxa final de reconhecimento de fones de 72.4% para *Correctness* e 68.9% para *Accuracy*. Se forem usadas as classes geradas automaticamente obtemos 73.8% para *correctness* e 70.0% para *Accuracy*.

Apesar de poder ser dispendioso em termos de tempo, cada resultado final pode conduzir ainda a melhorias. A partir de cada nova matriz de confusão pode-se aplicar o método de agrupamento automático e gerar novos *clusters* aperfeiçoando-se a divisão e melhorando a discriminação entre fones outra vez.

O conhecimento adquirido no estudo relativo à deteção de eventos abriu a possibilidade de desenvolvimento de um sistema de auxílio à terapia da fala que já se encontra atualmente em curso. O processo de desenvolvimento da linguagem é um processo iterativo. Todas as crianças começam por ter problemas na dicção de algumas palavras, mas à medida que crescem, melhoram a capacidade articulatória e a pronunciação também melhora. Quando

este processo de aprendizagem foge dos limites temporais típicos (por vários motivos: patológicos ou não) surgem as dificuldades de comunicação verbais. Estas ocorrem normalmente em idade pré-escolar e podem ser manifestadas das mais diversas formas. O problema agrava-se, com a entrada no 1º ciclo, não só devido à dificuldade patente de comunicação mas também pelos problemas sociais que podem a surgir. A terapia da fala é a especialidade associada à análise e intervenção direta na recuperação destes problemas. O programa de recuperação consta de sessões interativas entre a criança e o terapeuta. As sessões são normalmente individuais e de curta duração, o que torna impraticável incluir-se um programa alargado de terapia da fala à população estudantil em geral. No entanto, o sucesso de um programa de terapia da fala é limitado se se restringir às sessões com o terapeuta (habitualmente semanais). A família e os educadores têm um papel fundamental de estimulação, correção e treino.

Recorrendo ao conhecimento adquirido na deteção e reconhecimento de eventos de fala, o objetivo a que nos propomos, consiste em criar ferramentas que permitam futuramente a sua aplicação em jogos e desafios interativos simples para que a criança melhore a sua dicção. Trata-se de uma via complementar às sessões de terapia da fala que, de uma forma lúdica, estimula a criança a aperfeiçoar a sua dicção. Os objetivos de cada jogo são diferentes e vão desde a melhoria da produção da fala ao aperfeiçoamento do posicionamento articulatório, entre outros. Atuam no controlo da frequência de vibração das cordas vocais, no controlo do fluxo expiratório, no controlo da intensidade assim como na correção do posicionamento dos articuladores ou na coordenação fonoarticulatória.

A tarefa impõe alguma complexidade, na medida em que envolve fala de crianças e não existe nenhuma base de dados de fala de crianças em Português Europeu. O custo de usar modelos de adultos para reconhecimento de fala de crianças é demasiado elevado (as taxas de reconhecimento caem de 97% para 61%, [29]) e foi necessário recolher, validar e etiquetar uma base de dados com fala de crianças. Recolheram-se 3745 locuções resultando um total de 14099 etiquetas provenientes de 111 locutores com idades compreendidas entre os 5 e os 6 anos. Por outro lado os métodos conhecidos de extração de características do sinal de fala estão adaptados a fala de adultos cuja frequência fundamental e formantes estão abaixo dos das crianças. Apesar do grau de maturação da tecnologia existente no que se refere à deteção de eventos, o tema em questão tem muitas especificidades o que obriga a um estudo cuidadoso em cada etapa.

BIBLIOGRAFIA

[1] Abad, A., Neto, J., "Incorporating acoustical modelling of phone transitions in an hybrid ANN/HMM speech recognizer", in Proceedings of Interspeech 2008, Brisbane, Australia, 2008.

[2] Abdelatty, A.M., Spiegel, A. *et al*, "An acoustic-phonetic feature-based system for the automatic recognition of fricative consonants", In Proc. ICASSP-98, pp. 961-964, 1998.

[3] Abu-Amer, T. e Carson-Berndsen, J., "Multi-linear HMM based system for articulatory feature extraction". In Proc. ICASSP'03, volume 2, 21–24, Hong Kong, 2003.

[4] Aksela, M., "Adaptive Combinations of Classifiers with Application to On-line Handwritten Character Recognition", Ph.D. Thesis, Department of Computer Science and Engineering, Helsinki University of Technology, Helsinki, 2007.

[5] Ali, A., Spiegel, J. der, Mueller, P., Haentjens, G. e Berman, J., "An Acoustic-Phonetic Feature-Based System For Automatic Phoneme Recognition In Continuous Speech", in Proc. ISCAS'99-Vol.3, Florida, pp.118-121, June 1999.

[6] Ali, A., Spiegel, J., Mueller, P., "Acoustic-Phonetic Features For The Automatic Classification Of Stop Consonants", IEEE Transactions On Speech And Audio Processing, Vol. 9, No. 8, November 2001.

[7] Allen, J., "How Do Humans Process and Recognize Speech", IEEE Transactions on Speech and Audio Processing, October 1994.

[8] Baker, J. M., Deng, L., Glass, J., Khudanpur, S., Lee, C., Morgan, N. e O'Shaugnessy, D., "Research Developments and Directions in Speech Recognition and Understanding, Part 1," IEEE Signal Processing Magazine, vol. 26, no. 3, pp. 75-80, May 2009.

[9] Baker, J. M., Deng, L., Glass, J., Khudanpur, S., Lee, C., Morgan, N. e O'Shaugnessy, D., "Updated MINDS Report on Speech Recognition and Understanding, Part 2," IEEE Signal Processing Magazine, vol. 26, no. 4, pp. 78-85, July 2009.

[10] Behnke, S., "Discovering hierarchical speech features using convolutional non-negative matrix factorization", In Proceedings of International Joint Conference on Neural Networks (IJCNN'03), vol. 4, pp. 2758-2763 – Portland, OR, 2003.

[11] Bengio, Y., Mori, R., Flammia, G. e Kompe, R., "Global Optimization of a Neural Network-Hidden Markov Model Hybrid", IEEE Transactions on Neural Networks, Vol. 3, No. 2, pp 252-259, March 1992.

[12] Bilmes, J. e Kirchhoff, K. "Generalized rules for combination and joint training of classifiers", Pattern Analysis and Applications, 6(3) pp. 201-211, 2003.

[13] Bishop, C., Neural Networks for Pattern Recognition. Oxford University Press, 1995.

[14] Bourlard, H. e Morgan, N., "Hybrid HMM/ANN Systems for Speech Recognition: Overview and New. Research Directions", Springer Verlag, 1997.

[15] Brian, M., Etienne, B.: Phone *Clustering* Using the Bhattacharyya Distance, In Proc. ICSLP, 2005-2008, (1996).

[16] Bromberg, I., *et al*., "Detection-based ASR in the automatic speech attribute transcription project," in Proc. of Interspeech2007, pp. 1829-1832, August, 2007.

[17] Brunet, J.-P., Tamayo, P., Golub, T. e Mesirov, J., "Metagenes and molecular pattern discovery using matrix factorization" Proceedings of the National Academy of Sciences, volume 101, pp 4161-4169, 2004.

[18] Burges, C., "A tutorial on support vector machines for pattern recognition", Data Mining and Knowledge Discovery, vol 2, nº2, pp 121-167, 1998.

[19] Chelba, C., Morton, R., “Mutual information phone *clustering* for decision tree induction”. In: Proc. ICSLP 2002, Denver, 2002.

[20] Chen, B., Zhu, Q. e Morgan, N., “Learning long-term temporal features in LVCSR using neural networks," in Proc. ICSLP 2004, Jeju Island, KR, October 2004.

[21] Chou, W., Lee, C.-H., Juang, B.-H. e Soong, F.-K. “A minimum speech error rate pattern recognition approach to speech recognition,” Int. J. Pattern Recognition Artificial Intelligence, Special Issue on Speech Recognition for Different Language, vol. 8, no. 1, pp. 5–31, 1994.

[22] Cooper, M. e Foote, J., “Summarizing Video using Non-Negative Similarity Matrix Factorization”, IEEE Multimedia Signal Processing Workshop, December 11, 2002.

[23] Deng, L. e Yu, D., “Use of differential cepstra as acoustic features in hidden trajectory modelling for phonetic recognition,” in Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP), Honolulu, Hawaii, USA, Apr 2007.

[24] Deng, L., Yu, D. e Acero, A., “A generative modeling *frame*work for structured hidden speech dynamics”. In Proceedings of NIPS Workshop on Advances in Structured Learning for Text and Speech Processing, Whistler, BC, Canada, December 2005.

[25] Dietterich, T. e Bakiri, G. "Error-correcting output codes: A general method for improving multiclass inductive learning programs.", In Proceedings of the Ninth National Conference on Artificial Intelligence, pp.572-577, 1991.

[26] Droppo, J. e Acero, A., “Joint Discriminative Front End and Back End Training for Improved Speech Recognition Accuracy”, in Proc. of the Int. Conf. on Acoustics, Speech, and Signal Processing. Toulouse, May, 2006.

[27] Dunne R. e Campbell, N., “On the pairing of the *softmax* activation and cross-entropy penalty functions and the derivation of the *softmax* activation function”, in Proc Eighth Australasian Conf. on Neural Networks, pp. 181-185, 1997.

[28] Dusan, S., Rabiner, L. R., “Can automatic speech recognition learn more from human speech perception?”, Trends in Speech Technology, C. Burileanu (Ed.), Romanian Academic Publisher, 2005.

[29] Elenius, D and Blomberg, M, “Comparing speech recognition for adults and children”, Department of Speech, Music and Hearing, KTH, Stockholm, 2004.

[30] Evangelopoulos, G., e Maragos, P., "Speech Event Detection using Multiband Modulation Energy," Interspeech2005, Lisbon, 2005.

[31] Fletcher, R., “Practical Methods of Optimization”. John Wiley and Sons, Inc., 2nd edition, 1987.

[32] Franco, H., Cohen, M., Morgan, N., Rumelhart, D. e Abrash, V., “Context-dependent Connectionist Probability Estimation in a Hybrid Hidden Markov Model-Neural Net Speech Recognition System”, Computer Speech and Language, 8(3):211–222, 1994.

[33] Fujihara, H. e Goto, M. “Three Techniques for Improving Automatic Synchronization between Music and Lyrics: Fricative Detection, Filler Model, and Novel Feature Vectors for Vocal Activity Detection”, Proceedings of the 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2008), pp.69-72, April 2008.

[34] Furui, S., “50 Years of Progress in Speech and Speaker Recognition Research Identification”, In ECTI Transformations on Computer and Information Technology, vol. 1, no. 2, 2003.

[35] Ganapathiraju, A., Hamaker, J., Picone, J., “Hybrid SVM/HMM architectures for speech recognition, ICSLP, 2000.

[36] Garofolo, J., Lamel, L., Fisher, W., Fiscus, J., Pallett, D., e Dahlgren, n., DARPA, TIMIT Acoustic-Phonetic Continuous Speech Corpus CD-ROM. National Institute of Standards and Technology, 1990.

[37] Glass, J., Chang, J. e McCandless, M., "A probabilistic *frame*work for feature based speech recognition". In Proc. Int. Conf. Spoken Language Processing, pages 2277-2280, Philadelphia, October 1996.

[38] Grézl, F., Cernocký, J., "TRAP-based Techniques for Recognition of Noisy Speech", Lecture Notes in Computer Science, c. 9, DE, s. 270-277, ISBN 978-3-540-74627-0, ISSN 0302-9743, 2007.

[39] Gruhne, M., Schmidt, K., Dittmar, C., "Phoneme recognition in popular music," Proc. 8th International Conference on Music Information Retrieval, pp. 369–370, 2007.

[40] Gunawardana, A., Mahajan, M., Acero, A., e Platt. J., "Hidden conditional random fields for phone classification," in Proc. Interspeech, 2005, pp. 1117–1120, 2005.

[41] Gusfield, D. Algorithms on strings, trees, and sequences: computer science and computational biology. Cambridge University Press, UK, 1997.

[42] Gutkin, A. e King, S., "Detection of Symbolic Gestural Events in Articulatory Data for use in Structural Representations of Continuous Speech," ICASSP'05, 2005.

[43] Halberstadt, A. K., "Heterogeneous Acoustic Measurements and Multiple Classifiers for Speech Recognition", Ph.D. thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 1998.

[44] Halberstadt, A. e Glass, J. "Heterogeneous measurements and multiple classifiers for speech recognition". In Proc. ICSLP, volume 3, pages 995–998, Sydney, Australia, 1998.

[45] Halberstadt, A. e Glass, J., "Heterogeneous Acoustic Measurements for Phonetic Classification," in Eurospeech, 1997.

[46] Hasegawa-Johnson, M., Baker, J., Borys, S., Chen, K., Coogan, E., Greenberg, S., Juneja, A., Kirchhoff, K., Livescu, K., Mohan, S., Muller, J., Sonmez, K., and Wang, T. Landmark-based speech recognition: Report of the 2004 Johns Hopkins summer workshop. In Proc. IEEE ICASSP, volume 1, pages 213– 216, Philadelphia, 2005.

[47] Hifny Y. e Renals S., "Speech recognition using augmented conditional random fields," IEEE Transactions on Audio, Speech & Language Processing, vol. 17, no. 2, pp. 354–365, 2009.

[48] Hosom, J.-P., "Speaker-independent phoneme alignment using transition-dependent states," Speech Commun., vol. 51, no. 4, pp. 352–368, 2009.

[49] Hosom, J-P., Cole, R. "Burst Detection Based on Measurements of Intensity Discrimination", ICASSP'00, 2000.

[50] Hoyer, P. O., "Non-negative Matrix Factorization with sparseness constraints "Journal of Machine earning Research 5:1457-1469, 2004.

[51] Hu, G., e Wang, D.L., "Auditory segmentation based on event detection," ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio Processing, 2004.

[52] Imperl, B., Kacic, Z., Horvat, B. e Zgank, A., "*Clustering* of triphones using phoneme similarity estimation for the definition of a multilingual set of triphones", Speech Communication 39(3-4): 353-366, 2003.

[53] Joachims, T., Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press, 1999.

[54] Juneja, A. "Speech Recognition Based On Phonetic Features and Acoustic Landmarks," Ph.D. Thesis: University of Maryland, 2004.

[55] Juneja, A. e Espy-Wilson, C., "Segmentation of continuous speech using acoustic-phonetic parameters and statistical learning," in Proc. ICONIP, Singapore, 2002.

[56] Juneja, A. e Espy-Wilson, C., "Speech segmentation using probabilistic phonetic feature hierarchy and support vector machines," International Joint Conference on Neural Networks, Portland, 2003.

[57] Junqua, Jean-Claude e Jean- Paul Haton. “Robustness In Automatic Speech Recognition: Fundamentals and Applications”. Boston: Kluwer Academic Publishers, 1996.

[58] Karsmakers, P., Pelckmans, K., Suykens, J., Van hamme, H., “Fixedsize kernel logistic regression for phoneme classification”. In: Proc. InterSpeech Antwerp, Belgium, pp. 78–81, Aug. 2007.

[59] Keating, P.A., “Word-level phonetic variation in large speech corpora", paper presented at “The Word as a Phonetic Unit” in Berlin, October 1997.

[60] Keating, P.A., D. Byrd, E. Flemming, e Y. Todaka Phonetic analyses of word and segment variation using the TIMIT corpus of American English, Speech Communication. 14, 131-142, 1994.

[61] Kempton, T. e Moore, R. K., “Language identification: insights from the classification of hand annotated phone transcripts”, Proc. Odyssey Workshop on Speaker and Language Recognition, Stellenbosch, South Africa, 2008.

[62] Keshet, J., *et al*, "Phoneme Alignment Based on Discriminative Learning," Interspeech2005, Lisbon, 2005.

[63] Kim, H.-G. e Sikora, T., “Audio Spectrum Projection Based on Several Basis Decomposition Algorithms applied to General Sound Recognition and Audio Segmentation”, EUSIPCO 2004, Vienna, Austria, September 6-10, 2004.

[64] Kim, H.-G., Burred, J., e Sikora, T., “How efficient is MPEG-7 for General Sound Recognition? “ 25th International AES Conference Metadata for Audio, London, UK, June 17-19, 2004.

[65] Köhler, J., “Multi-lingual phoneme recognition exploiting acoustic-phonetic similarities of sounds”, ICSLP’96, pp. 2195-2198, Philadelphia, PA, USA, October 1996.

[66] Korhonen, P., e Laine, U., "Unsupervised Segmentation of Continuous Speech Using Vector Autoregressive Time-Frequency Modeling Errors," Interspeech2005, Lisbon, 2005.

[67] Lamel L. e Gauvain J. L., “High Performance Speaker Independent Phone Recognition using CDHMM”, Proc. Eurospeech, pp. 121-124, 1993.

[68] Lee, C. W., Kang, H., Jung, K. e Kim, H. J., "Font Classification Using NMF", Lecture Notes in Computer Science (LNCS 2756) , pp. 470-477,August, 2003.

[69] Lee, C.-H e Rabiner, L., “On the use of support vector machines for phonetic classification”, Speech and Signal Processing, vol II, pp 585-588, 2000.

[70] Lee, C.-H., Clements, M. A., Dusan, S., Fosler-Lussier, E., Johnson, K.,. Juang, B.-H e Rabiner, L. R., "An Overview on Automatic Speech Attribute Transcription (ASAT)," Proc. Interspeech, Antwerp, Belgium, 2007.

[71] Lee, C.-H., e Juang, B.-H., “A new detection paradigm for collaborative automatic speech recognition and understanding,” SWIM 2004, January 2004.

[72] Lee, D. D. e Seung, H. S., "Unsupervised learning by convex and conic coding," in Advances in Neural Information Processing Systems, M. C. Mozer, M. I. Jordan, and T. Petsche, Eds., vol. 9. The MIT Press, pp. 515-521, 1997.

[73] Lee, D. D. e Seung, H. S., “Algorithms for nonnegative matrix factorization” Adv. Neural Info. Proc. Syst. 13, pp. 556—562, 2000.

[74] Lee, D. D. e Seung, H. S., “Learning the parts of objects by non-negative matrix factorization” In Nature, no. 401, pp. 788—791, 1999.

[75] Lee, J. S., Lee, D. D., Choi, S. e Lee, D. S., "Application of nonnegative matrix factorization to dynamic positron emission tomography," in Proc. Int. Conf. Independent Component Anal. Signal Separation, T.-W. Lee, T.-P Jung, S. Makeig, and T. J. Sejnowski, Eds., San Diego, CA, Dec. 9-13, pp. 629-632, 2001.

[76] Lee, K. e Hon, H. "Speaker-independent phone recognition using hidden Markov models", IEEE Transactions on Acoustics, Speech, and Signal Processing, vol.37 (11), pp. 1642-1648, November 1989.

[77] Lee. C.-H., "From knowledge-ignorant to knowledge-rich modeling: A new speech research paradigm for next-generation automatic speech recognition, Proc. ICSLP 2004, pp. 109-111, 2004.

[78] Leonard, R. G., "A database for speaker-independent digit recognition," Proc. ICASSP, pp.328-331, 1984.

[79] Leung, K.Y. e Siu, M., "Speech Recognition Using Combined Acoustic and Articulatory Information with Re-training of Acoustic Model Parameters,'' in Proc. of ICSLP2002, volume 3, pages 2117-2120, Sep, 2002.

[80] Li, J., "Soft Margin Estimation For Automatic Speech Recognition," PhD thesis, Georgia Institute of Technology, School of Electrical and Computer Engineering, 2008.

[81] Li, J., e Lee, C-H., "On Designing and Evaluating Speech Event Detectors", Interspeech 2005.

[82] Li, J., Tsao Y. e Lee, C.-H., "A study on knowledge source integration for candidate rescoring in automatic speech recognition," in Proc. ICASSP05, Philadelphia, 2005.

[83] Lin, Y., Lee, Y. and Wabba, G., "Support vector machines for classification in nonstandard situations". Technical Report 1016. Department of Statistics, University of Wisconsin, 2000.

[84] Lippmann, R. "Speech recognition by machines and humans", Speech Communication, 22, pp.1-15, 1997.

[85] Liu, Y, "Structural Event Detection for Rich Transcription of Speech", Ph.D. Thesis, Purdue University, December 2004.

[86] Lopes, C. and Perdigão, F." Phone Recognition on the TIMIT Database", Speech Technologies / Book 1", Intech, pp 285-302, June 2011.

[87] Lopes, C., Perdigão, F. , "Global Discriminative Training of a Hybrid Speech Recognizer", proceedings of the I Iberian SLTech - I Joint SIG-IL/Microsoft Workshop on Speech and Language Technologies for Iberian Languages, pages 57-60, September 2009.

[88] Lopes, C., Perdigão, F., "A Hierarchical Broad-Class Classification to Enhance Phoneme Recognition", 17th European Signal Processing Conference (EUSIPCO-2009), Glasgow, Scotland, August 2009.

[89] Lopes, C., Perdigão, F., "Confusion-Driven Phoneme *Clustering* to Enhance Phoneme Recognition", EURASIP Journal on Advances in Signal Processing (2ª fase de revisão).

[90] Lopes, C., Perdigão, F., "Hybrid HMM/SVM Speech Event Detector ", 6th Conference on Telecommunications, Conftele 2007, Peniche, Portugal, v. 1. pp 601-604, May 2007.

[91] Lopes, C., Perdigão, F., "Improved Performance Evaluation of Speech Event Detectors", International Conference on Spoken Language Processing (Interspeech2006), Pittsburgh, September, 2006.

[92] Lopes, C., Perdigão, F., "Non-negative matrix factorization: dealing with scaled objects", 3rd International Workshop on Mathematical Techniques and Problems in Telecommunications", Leiria, Portugal, pp. 40-44, September, 2006.

[93] Lopes, C., Perdigão, F., "Phonetic Recognition Improvements through Input Feature Set Combination and Acoustic Context Window Widening", 7th Conference on Telecommunications, Conftele, Sta Maria da Feira, Portugal, v. 1. pp 449-452, May 2009.

[94] Lopes, C., Perdigão, F., "Speech Event Detection By Non Negative Matrix Deconvolution", 15th European Signal Processing Conference (EUSIPCO-2007), Poznan, Poland, v. 1. pp 1280-1284, September 2007.

[95] Lopes, C., Perdigão, F., "A Discriminative Training Method Applied to a Hybrid ANN/HMM Phoneme Recognizer", IEEE ICSP'08, 9th International Conference on Signal Processing, Beijing, China, October 2008.

[96] Lopes, C., Perdigão, F., "Event Detection by HMM, SVM and ANN: A Comparative Study". In A. Teixeira, V. Lima, L. Oliveira, and P. Quaresma (Eds.), Computational Processing of the Portuguese Language, Lecture Notes in Computer Science, 2008, Volume 5190/2008, 1-10: Springer-Verlag.

[97] Lopes, C., Veiga, A., Perdigão, F. , "Automatic Phone *Clustering* based on Confusion Matrices". In T. Pardo, A. Branco, A., R. Vieira and V. de Lima (Eds.), Computational Processing of the Portuguese Language, Lecture Notes in Computer Science, 2010, Volume 6001/2010, 124-127: Springer-Verlag.

[98] Lopes, J., Neves, C., Veiga, A., Maciel, A., Lopes, C., Perdigão, F., Sá, L., "Development of a Speech Recognizer with the Tecnovoz Database", Propor 2008, International Conference on Computational Processing of Portuguese, Aveiro, Portugal, 2008.

[99] Macherey, W., Haferkamp, L., Schlüter, R., Ney, H., "Investigations on error minimizing training criteria for discriminative training in automatic speech recognition". INTERSPEECH 2005: 2133-2136, 2005.

[100] Mareuil, P., Corredor Ardoy, C., Adda-Decker, M. "Multi-lingual automatic phoneme *clustering*", 14th Int. Conf. on Phonetic Science, ICPhS-99, August 1999.

[101] Matejka, P., "Phonotactic And Acoustic Language Recognition", PhD thesis, Brno University of Technology, Faculty of Electrical Engineering and Communication, 2009.

[102] Mohamed, A.; Dahl, G.; Hinton, G., "Acoustic Modeling using Deep Belief Networks", IEEE Transactions on Audio, Speech, and Language Processing, 2011.

[103] Mohamed, A.-R., Hinton, G., "Phone recognition using Restricted Boltzmann Machines", IEEE ICASSP 2010.

[104] Morris, J. e Fosler-Lussier, E., "Conditional Random Fields for Integrating Local Discriminative Classifiers," IEEE Transactions on Acoustics, Speech, and Language Processing, 16:3, pp 617-628, March 2008.

[105] Morris, J. e Fosler-Lussier, E., "Further experiments with detector-based conditional random fields in phonetic recognition," International Conference on Acoustic, Speech, and Signal Processing (ICASSP-2007), Hawaii, 2007.

[106] Morris, J., e Fosler-Lussier, E., "Combining phonetic attributes using conditional random fields," in Proc. Interspeech, Pittsburg, USA, pp. 597 – 600, September 2006.

[107] Nearey, T., "Speech perception as pattern recognition". JASA, 101(6), pp. 3241-3254, 1997.

[108] Niyogi, P., Burges, C., Ramesh, P., "Distinctive Feature Detection Using Support Vector Machines", ICASSP'99, 1999.

[109] Novak, M. e Mammone, R., "Use of nonnegative matrix factorization for language model adaptation in a lecture transcription task," in Proc. 2001 IEEE Int. Conf. Acoust., Speech, Signal Processing, vol. 1, Salt Lake City, UT, May 7-11, pp. 541-544, 2001.

[110] Novak, M. e Mammone, R., "Improvement of non-negative matrix factorization based language model using exponential models". Automatic Speech Recognition and Understanding, 2001, on page(s): 190- 193, 2001.

[111] Paatero, P. e Tapper, U., "Positive Matrix Factorization: A non-negative factor model with optimal utilization of error estimates of data values," Environmetrics, vol. 5, pp. 111-126, 1994.

[112] Paatero, P., "The Multilinear Engine—a table-driven least squares program for solving multilinear problems, including the n-way parallel factor analysis model," J. Comput. and Graph. Stat., vol. 8, no. 4, pp. 854-888, 1999.

[113] Perona, P. e Malik, J., "Scale-space and edge detection using anisotropic diffusion", PAMI 12(7), 629-639, 1990.

[114] Platt, J., Cristianini, N. e Shawe-Taylor, J., "Large margin dags for multi-class classification", In S. A. Solla, T. K. Leen, and K.-R. Muller, editors, Advances in Neural Information Processing Systems., 2000.

[115] Povey, D. and Woodland, P., "Minimum phone error and i-smoothing for improved discriminative training," in IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, vol. 1, Orlando, FL, pp. 105–108, May 2002.

[116] Povey, D., "Discriminative Training for Large Vocabulary Speech Recognition". PhD thesis, University of Cambridge, UK, July 2004.

[117] Prasanna, S., "Event based analysis of speech," in Dept. of Computer Science and Engineering, Ph.D. Thesis: Indian Institute of Technology Madras, India, 2004.

[118] Rabiner, L. R., "A tutorial on hidden Markov models and selected applications in speech recognition". Proceedings of the IEEE, 77(2):257-286. February, 1989.

[119] Rabiner, L., Juang, B., "An Introduction to Hidden Markov Models", IEEE ASSP Magazine 3(1): 4-16, January, 1986.

[120] Reynolds, T.J. e Antoniou, C.A., "Experiments in speech recognition using a modular MLP architecture for acoustic modeling", Information Sciences, Volume 156, Issue 1-2, pp 39 – 54, 2003.

[121] Riedmiller M. and Braun H., "A direct adaptive method for faster backpropagation learning: The RPROP algorithm," in Proc. ICNN, San Francisco, CA, pp. 586–591, 1993.

[122] Riis, S., Krogh, A. "Hidden Neural Networks: A *Frame*work for HMM/NN Hybrids", in Proc of ICASSP97 April, 1997.

[123] Rijsbergen, C. J., Information Retrieval: Department of Computing Science - University of Glasgow, 1979.

[124] Robinson T. e Fallside F., "A Recurrent Error Propagation Network Speech Recognition System," Computer Speech & Language, 5:3, pp. 259-274, 1991.

[125] Robinson, T., "An application of recurrent nets to phone probability estimation", IEEE Transactions on Neural Networks, vol. 5, no. 3, 1994.

[126] Robinson, T., Almeida, L., Boite, J. M., Bourlard, H., Fallside, F., Hochberg, M., Kershaw, D., Kohn, P., Konig, Y., Morgan, N., Neto, J. P., Renals, S., Saerens, M., e Wooters, C., "A neural network based, speaker independent, large vocabulary, continuous speech recognition system: The wernicke project". In Eurospeech93, Berlin, Germany, 1993.

[127] Rose, R. e Momayyez, P. "Integration of Multiple Feature Sets for Reducing Ambiguity in ASR". IEEE International Conference on Acoustics, Speech and Signal Processing, 2007 (ICASSP), Volume 4, Pages: 325-328, 2007.

[128] Rose, R., Momayyez, P., "Integration Of Multiple Feature Sets For Reducing Ambiguity In ASR". IEEE International Conference on Acoustics, Speech and Signal Processing, 2007 (ICASSP), Volume 4, Pages: 325-328, 2007.

[129] Sainath, T. N. e Zue, V. "A Comparison of Broad Phonetic and Acoustic Units for Noise Robust Segment-Based Speech Recognition," Proc. Interspeech, Brisbane, Australia, September 2008.

[130] Sajda, P., Du, S., Brown, T.R., Stoyanova, R., Shungu, D.C., Mao, X. e Parra, L.C., "on-negative matrix factorization for rapid recovery of constituent spectra in magnetic resonance chemical shift imaging of the brain" IEEE Transactions on Medical Imaging, 23 (12) 1453-1465, 2004.

[131] Salomon, J., "Support vector machines for phoneme classification", MsC thesis, University of Edinburgh, 2001.

[132] Scanlon, P., Ellis, D. e Reilly, R., "Using Broad Phonetic Group Experts for Improved Speech Recognition", IEEE Transactions on Audio, Speech and Language Processing, vol.15 (3) , pp 803-812, March 2007.

[133] Scheffer, N., e Bonastre, J.-F., "Speaker Detection using Acoustic Event Sequences," Interspeech2005, Lisbon, 2005.

[134] Schluter, R., Macherey, W., Muller, B., Ney, H., "Comparison of discriminative training criteria and optimization methods for speech recognition". Speech Communication 34, 287–310, 2001.

[135] Schutte K. e Glass, J., "Robust Detection of Sonorant Landmarks," Interspeech2005, Portugal, Sept, 2005.

[136] Schwarz, P., "Phoneme recognition based on long temporal context", PhD thesis, Brno University of Technology, Faculty of Information Technology, 2008.

[137] Schwarz, P., Matejka, P., e Cernocky, J., "Hierarchical structures of neural networks for phoneme recognition," in Proc. IEEE ICASSP, Toulouse, France, pp. 325–328, 2006.

[138] Sha, F. e Saul, L., "Large margin gaussian mixture modelling for phonetic classification and recognition," in Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP), Toulouse, France, pp. 265 – 268, May 2006.

[139] Sha, F. e Saul, L., "Real-Time Pitch Determination of One or More Voices by Nonnegative Matrix Factorization". To appear in L. K. Saul, Y. Weiss, and L. Bottou (eds.), Advances in Neural Information Processing Systems 17. MIT Press: Cambridge, MA, 2005.

[140] Shahnaz, F., Berry, M., Pauca, P. e Plemmons, R., "Document *Clustering* using Nonnegative Matrix Factorization", Journal on Information Processing & Management, August 2004.

[141] Shih, P.-Y., Wang, J.-F. Lee, H. P., Kai, H.-J., Kao, H.-T., Lin, Y.-N., "Acoustic and Phoneme Modeling Based on Confusion Matrix for Ubiquitous Mixed-Language Speech Recognition". SUTC 2008: 500-506, 2008.

[142] Silva, B., Mendes, H., Lopes, C., Veiga, A., Perdigão, F., "A Fast Discriminative Training Algorithm For Minimum Classification Error", proceedings of the I Iberian SLTech - I Joint SIG-IL/Microsoft Workshop on Speech and Language Technologies for Iberian Languages, pages 53-56, September 2009.

[143] Siniscalchi, S. M., Schwarz, P. e Lee, C.-H., "High-accuracy phone recognition by combining high-performance lattice generation and knowledge based rescoring," in Proc. ICASSP, 2007, pp. IV-869–IV-872.

[144] Smaragdis, P. e Brown, J. C., "Non-negative matrix factorization for polyphonic music transcription". In Proc. of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pages 177–180, 2003.

[145] Smaragdis, P., "Discovering Auditory Objects through Non-Negativity Constraints," SAPA 2004, Jeju, Korea, October 2004.

[146] Sokal, R. R. e F. J. Rohlf, "The comparison of dendrograms by objective methods". Taxon, 11:33-40, 1962.

[147] Trentin, E., Gori, M., "A survey of hybrid ANN/HMM models for automatic speech recognition". Neurocomputing. , vol. 37, pp. 91-126, March 2001.

[148] Tropp, J., Literature survey: Nonnegative matrix factorization, University of Texas at Asutin, preprint, 2003.

[149] Tulyakov S., *et al*., "Review of Classifier Combination Methods", Studies in Computational Intelligence (SCI) 90, 361–386 (2008).

[150] Vapnik, V., "An overview of statistical learning theory", IEEE Transactions on Neural Networks, vol 10, nº 5, pp 988-1000, September 1999.

[151] Vapnik, V., "Statistical Learning Theory", Wiley, Interscience, 1998.

[152] Vapnik, V., "The Nature of Statistical learning Theory", Springer, New York, 1995.

[153] Vihola, M., Harju, M., Salmela, P., Suontausta, J. e Savela. J. “Two Dissimilarity Measures for HMMs and Their Application in Phoneme Model *Clustering*,” Proc. ICASSP 2002, Vol. I, pp. 933–936, May 13–17, Orlando, Florida, 2002.

[154] Vihola, M., “Dissimilarity Measures for Hidden Markov Models and Their Application in Multilingual Speech Recognition,” MSc thesis, Tampere University of Technology, May 2002.

[155] Wang, Y., Jiar, Y., Hu, C., Turk, M., "Fisher non-negative matrix factorization for learning local features," Asian Conference on Computer Vision, Korea, January 27-30, 2004.

[156] Woodland, P.C. e Povey. D. “Large scale discriminative training of hidden Markov models for speech recognition”. Computer Speech and Language, 16:25–47, 2002.

[157] Wrigley, S., “A Theory and Computational Model of Auditory Selective Attention”, Ph.D. thesis, University of Sheffield, 2002.

[158] Wu, S.L., Kingsbury, B., Morgan, N. e Greenberg, S., "Incorporating Information from Syllable-length Time Scales into Automatic Speech Recognition", in Proc. ICASSP98, 1998.

[159] Xu, W., Liu, X. e Gong, Y., “Document *clustering* based on non-negative matrix factorization”, SIGIR '03: Proceedings of the 26th International ACM SIGIR conference on Research and development in information retrieval, 2003.

[160] Yang, L., Zhang, J., e Yan. Y., “Acoustic Units Selection in Chinese-English Bilingual Speech Recognition”. in NOLISP. 2007.

[161] Young, S. J., “The general use of tying in phoneme-based hmm speech recognizers,” in Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP), San Francisco, USA, Mar 1992.

[162] Young, S. J., *et al*, “The HTK book”. Revised for HTK version 3.4, Cambridge University Engineering Department, Cambridge, December 2006.

[163] Young, S., "HMMs and Related Speech Recognition Technologies." Springer Handbook of Speech Processing. J. Benesty, M. Sondhi and Y. Huang, Springer, 2007.

[164] Yu, D., Deng, L. e Acero, A., “A lattice search technique for a long-contextual-span hidden trajectory model of speech”. Speech Communication, 48:1214-1226, 2006.

[165] Yu, D., Deng, L., “Large-Margin Discriminative Training of Hidden Markov Models for Speech Recognition”, in Proc of the International Conference on Semantic Computing, 17-19 Sept. 2007 Page(s):429 – 438.

[166] Yuan, J., Liberman, M., “Robust speaking rate estimation using broad phonetic class recognition,” Proceedings of ICASSP 2010, pp. 4222-4225.

[167] Zgank, A., Horvat, B. e Kacic, Z., “Data-driven generation of phonetic broad classes, based on phoneme confusion matrix similarity”. Speech Commun. 47(3): 379-393, 2005.

[168] Zue, V. e Seneff, S., “Transcription and alignment of the TIMIT database”. In Hiroya Fujisaki (Ed.), Recent research toward advanced man-machine interface through spoken language. Amsterdam: Elsevier, pp 464-447, 1996.

[169] Zue, V., Seneff, S. e Glass J., "Speech database development at MIT: TIMIT and beyond", Speech Communication, Vol. 9, No. 4, pp. 351-356, 1990.

ANEXO I

Phonemic and phonetic symbols used in the TIMIT.

	SYMBOL	EXAMPLE WORD	POSSIBLE PHONETIC TRANSCRIPTION
Stops:	b	bee	BCL B iy
	d	day	DCL D ey
	g	gay	GCL G ey
	p	pea	PCL P iy
	t	tea	TCL T iy
	k	key	KCL K iy
	dx	muddy, dirty	m ah DX iy, dcl d er DX iy
	q	bat	bcl b ae Q
Affricates:			
	jh	joke	DCL JH ow kcl k
	ch	choke	TCL CH ow kcl k
Fricatives:			
	s	sea	S iy
	sh	she	SH iy
	z	zone	Z ow n
	zh	azure	ae ZH er
	f	fin	F ih n
	th	thin	TH ih n
	v	van	V ae n
	dh	then	DH e n
Nasals:			
	m	mom	M aa M
	n	noon	N uw N
	ng	sing	s ih NG
	em	bottom	b aa tcl t EM
	en	button	b ah q EN
	eng	washington	w aa sh ENG tcl t ax n
	nx	winner	w ih NX axr
Semivowels and Glides:	l	lay	L ey
	r	ray	R ey
	w	way	W ey
	y	yacht	Y aa tcl t
	hh	hay	HH ey
	hv	ahead	ax HV eh dcl d
	el	bottle	bcl b aa tcl t EL
Vowels:			
	iy	beet	bcl b IY tcl t
	ih	bit	bcl b IH tcl t
	eh	bet	bcl b EH tcl t
	ey	bait	bcl b EY tcl t
	ae	bat	bcl b AE tcl t
	aa	bott	bcl b AA tcl t
	aw	bout	bcl b AW tcl t
	ay	bite	bcl b AY tcl t
	ah	but	bcl b AH tcl t
	ao	bought	bcl b AO tcl t
	oy	boy	bcl b OY
	ow	boat	bcl b OW tcl t
	uh	book	bcl b UH kcl k
	uw	boot	bcl b UW tcl t
	ux	toot	tcl t UX tcl t
	er	bird	bcl b ER dcl d
	ax	about	AX bcl b aw tcl t
	ix	debit	dcl d eh bcl b IX tcl t
	axr	butter	bcl b ah dx AXR
	ax-h	suspect	s AX-H s pcl p eh kcl k tcl t
Others:			
	pau	pause	
	epi	epenthetic silence	
	h#	begin/end marker (non-speech events)	