Luis Miguel Bagagem Castela

Pesquisa de Fala

Setembro de 2015

Universidade de Coimbra

Departamento de Engenharia Electrotécnica e de Computadores
Faculdade de Ciências e Tecnologia
Universidade de Coimbra

Dissertação para a Obtenção de Grau de Mestre em
Engenharia Electrotécnica e de Computadores

Pesquisa de Fala

Luis Miguel Bagagem Castela

Desenvolvido com a Supervisão de
Prof. Doutor Fernando Santos Perdigão

Júri

Presidente:	Prof. Doutor António Paulo Mendes Breda Dias Coimbra
Orientador:	Prof. Doutor Fernando Santos Perdigão
Vogal:	Prof. Doutor Rui Pedro Pinto de Carvalho e Paiva
Vogal:	Prof. Adjunta Carla Alexandra Calado Lopes

Setembro de 2015

Agradecimentos

Agradeço à minha família, em especial, aos meus pais e irmã pela forma como me apoiaram ao longo de todo o meu percurso académico. Foram anos recheados de altos e baixos e, tal como estiveram presentes para me congratular pelos sucessos alcançados, também pude contar sempre com eles nos momentos de maiores dificuldades. Por isso e muito mais, tenho a certeza que sem eles nada disto teria sido possível.

Gostava também de deixar uma palavra de gratidão aos meus amigos dentro e fora do seio académico, por terem ajudado a superar os momentos de maior aperto.

Agradeço profundamente ao meu orientador, o Professor Doutor Fernando Perdigão pelo apoio prestado e conhecimento transmitido durante a realização desta dissertação, bem como pelo tempo dispendido em prol deste projecto.

Agradeço a todos os meus colegas de laboratório pela preciosa ajuda que me deram neste trabalho e pela sua disponibilidade, em especial ao Jorge Proença pela ajuda em várias etapas do trabalho. Contribuiu largamente para a evolução do trabalho, bem como na fase de testes. Sem ele, a conclusão desta dissertação seria muito mais complicada.

A todos, o meu mais sincero Obrigado,

Luis Castela

Resumo

Nos últimos anos, a deteção de fala em ficheiros de áudio tem recebido um aumento de interesse por parte das comunidades de pesquisa e desenvolvimento desta área. Como tal, foram criados inúmeros sistemas que implementam as mais diversas técnicas desenvolvidas para o reconhecimento de fala.

Neste trabalho é explorada uma abordagem de deteção de fala no âmbito do desafio *Query by Example Search on Speech Task* (QUESST) 2015 que pressupõe a procura de fala de acordo com certos tipos de exemplos de *queries* de fala, independentemente da língua em questão.

Para tal, é considerado um sistema de reconhecimento de fonemas que utiliza uma análise de termo curto dos sinais de áudio para a extração das caraterísticas da fala sob a forma de coeficientes cepstrais em escala MEL (MFCC), que são utilizados por um subsistema com uma arquitetura híbrida entre uma rede neuronal artifical (ANN) e modelos de *Markov* escondidos (HMM) para a descodificação dos respetivos fonemas sob a forma de ficheiros posteriorgramas (valores de probabilidades *a posteriori* dos fonemas). Foram treinados dois sistemas adicionais para o reconhecimento de fonemas, um em Português Europeu e outro em Inglês.

Posteriormente, são aplicadas várias variantes desenvolvidas de uma técnica de alinhamento temporal dinâmico (DTW) aos posteriorgramas obtidos, para realizar a localização da *query* de exemplo nos vários áudios de procura.

Em adição a este sistema e para combater as condições impostas nos ficheiros de áudio pela organização do desafio, foi desenvolvido um algoritmo de subtração espectral de ruído baseado em níveis de energia para o melhoramento da qualidade dos mesmos.

As abordagens desenvolvidas neste trabalho permitiram obter os segundos melhores resultados deste desafio.

Palavras-Chave: Deteção de Fala, Caraterísticas da Fala, Reconhecimento de Fonemas, QUESST 2015, Rede Neuronal Artificial, Alinhamento Temporal Dinâmico, Subtração Espectral de Ruído.

Abstract

In the recent years, speech detection in audio data has received increased attention in the research and development communities. Therefore, it were created several systems that implement various techniques developed for speech recognition.

This work explored a speech detection approach for the challenge Query by Example Search on Speech Task (QUESST) 2015, which is based on speech browsing for certain types of speech queries regardless of the language.

To this end, it was considered a phoneme recognition system that uses a short-term analysis of the audio signals to extract the speech characteristics in the form of Mel Frequency Cepstral Coefficients (MFCC), which are used by a subsystem with a hybrid architecture between an Artificial Neural Network (ANN) and Hidden Markov Models (HMM) for decoding the respective phonemes to the form of posteriorgrams (values of posterior probabilities of phonemes). Two additional systems were trained for phonemes recognition, one for European Portuguese and the other for English.

Subsequently, several developed variants of the Dynamic Time Warping (DTW) technique were applied to the posteriorgrams obtained to perform the match between the example query and all the search audio.

In addition to this system and to fight the challenging conditions imposed by the organization of this challenge in the audio data, it was developed an algorithm of spectral noise subtraction based on energy levels for the improvement of the quality of this data.

The approaches developed in this work allowed to obtain the second best results in this challenge.

Keywords: Speech Detection, Speech Characteristics, Phoneme Recognition, QUESST 2015, Artifical Neural Network, Dynamic Time Warping, Spectral Noise Subtraction.

Índice

Lista de Acrónimos **iii**

Lista de Figuras **vi**

Lista de Tabelas **viii**

1 Introdução **1**

1.1 Definição do Problema 1

1.2 Processamento de termo curto dos sinais de áudio 3

1.3 Reconhecimento de fonemas com Redes Neuronais 5

1.4 Sistemas de alinhamento temporal dinâmico 8

1.5 MediaEval 9

1.6 Melhoramento da SNR das locuções 12

2 Treino da Rede Neuronal Artificial **13**

2.1 Arquitetura do Sistema 13

2.2 Sistemas Disponibilizados 17

2.3 Treino de Novos Sistemas 18

2.3.1 Sistema para a Língua Portuguesa 20

2.3.2 Sistema para a Língua Inglesa 22

3 Alinhamento Temporal Dinâmico **25**

3.1 Cálculo da Matriz de Distâncias 26
3.2 Estratégias com alinhamento temporal dinâmico 27
3.3 Estratégias com alinhamento temporal dinâmico modificado 28

4 Teste do Sistema Inicial **33**

5 Tratamento da base de dados de desenvolvimento do MediaEval **37**

5.1 Reverberação 37
5.2 Algoritmo de Subtração Espectral 38

6 Teste do Sistema Final **45**

7 Conclusão **53**

Bibliografia **58**

A Apêndice A **59**

B Apêndice B **61**

C Apêndice C **63**

D Apêndice D **65**

Lista de Acrónimos

ANN: Artificial Neural Network.

ATWV: Actual Term Weighted Value.

BUT: Brnu University of Technology.

CNXE: Normalized Cross Entropy Cost.

DCT: Discret Cosine Transform.

DET: Detection Error Tradeoff.

DFT: Discret Fourier Transform.

DNN: Deep Neural Network.

DTW: Dynamic Time Warping.

ERR: Error Recognition Rate.

HMM: Hidden Markov Models.

HTK: Hidden Markov Model Toolkit.

IPA: International Phonetic Alphabet.

MFCC: Mel Frequency Cepstral Coefficients.

MLF: Master Label File.

MLP: Multilayer Perceptron.

NIST: National Institute of Standards and Technology.

PAV: Pool-Adjacent Violators.

QUESST: Query by Example Search on Speech Task.

SAMPA: Speech Assessment Methods Phonetic Alphabet.

SNR: Signal-to-Noise Ratio.

TWV: Term Weighted Value.

Lista de Figuras

1.1 Exemplo de criação dos vetores de caraterísticas da fala. Editado de [31]. 4

1.2 Exemplo de um modelo de um Neurónio. Editado de [6]. 6

1.3 Exemplo da Arquitetura de uma *Multilayer Perceptron*. Editado de [30]. 6

1.4 Exemplo de um modelo um *Perceptron*. Editado de [6]. 7

2.1 Exemplo da Arquitetura do Sistema de Reconhecimento de Fonemas com duas redes neuronais com contexto à esquerda e à direita e uma rede de fusão. Editado de [31]. 14

2.2 Exemplo da Criação de Vetores de Contexto Temporal. 15

2.3 Exemplo da estrutura da rede neuronal MLP considerada para a língua portuguesa. 16

2.4 Resultados das Iterações de Treino do Sistema de Língua Portuguesa I. 21

2.5 Resultados das Iterações de Treino do Sistema de Língua Portuguesa II. 22

2.6 Resultados das Iterações de Treino do Sistema de Língua Inglesa I. 23

2.7 Resultados das Iterações de Treino do Sistema de Língua Inglesa II. 24

3.1 Exemplo de um posteriorgrama. 25

3.2 Exemplo de um esquemático de caminhos com peso unitário considerados para a criação da DTW. Retirado de [25]. 27

3.3 Exemplo de uma correspondência para queries do Tipo 1. 28

3.4 Exemplo de uma correspondência para queries do Tipo 2 com variação lexical no fim. 29

3.5 Exemplo de uma correspondência para queries do Tipo 2 com variação lexical no ínicio. 30

3.6 Exemplo de uma correspondência para queries do Tipo 2 com um salto horizontal no caminho ótimo. 31

3.7 Exemplo de uma correspondência para queries do Tipo 2 com uma reordenação de palavras. 32

3.8 Exemplo de uma correspondência para queries do Tipo 3 com um salto vertical no caminho ótimo. 32

5.1 Exemplo de Reverberação de um Sinal Acústico. 38

5.2 Exemplo do Filtro Passa-Alto do tipo Butterworth considerado. 39

5.3 Exemplo das Tramas de Energia de um Sinal consideradas para Subtração Espectral. 40

5.4 Exemplo dos Quantis de Energia e das Medianas dos Quantis de Energia de um sinal considerados para a decisão da realização de Subtração Espectral. 41

5.5 Exemplo do Limiar de Ruído e de Segmentos de Ruído considerados de um sinal de exemplo. 42

6.1 Curvas DET para os sistemas Inical, Primário com Informação Paralela e Secundário com Informação Paralela. 49

6.2 Curvas DET para os sistemas de 2014, Primário 2015 e Secundário 2015 com Informação Paralela. 51

Lista de Tabelas

2.1 Sistemas de Reconhecimento de Fonemas disponibilizados por BUT Speech@FIT [33]. 18

2.2 Número de Fonemas e Taxas de Erro de Reconhecimento de Fonemas (ERR) dos sistemas disponibilizados por BUT Speech@FIT [33]. 18

6.1 Resultados obtidos para a métrica principal Cnxe das diferentes estratégias DTW para o conjunto de desenvolvimento. 49

6.2 Comparação de resultados de sistemas QUESST de 2014 e de 2015. 51

A.1 Tabela de fonemas considerados para vogais da língua portuguesa 59

A.2 Tabela de fonemas considerados para consoantes da língua portuguesa. 60

A.3 Tabela de fonemas considerados para silêncios/ruídos da língua portuguesa. 60

B.1 Mapeamento de fonemas de TIMIT considerado para a língua inglesa. Editado de [31]. 61

B.2 Mapeamento de fonemas de Resource Management considerado para a língua inglesa. 62

C.1 Resultados obtidos para a métrica principal Cnxe para o conjunto de desenvolvimento. 63

C.2 Resultados obtidos para a métrica secundária ATWV para o conjunto de desenvolvimento. 63

C.3 Resultados obtidos para a métrica principal Cnxe para o conjunto de avaliação. . 64

C.4 Resultados obtidos para a métrica secundária ATWV para o conjunto de avaliação. 64

D.1 Resultados oficiais do desafio QUESST 2015. 65

Capítulo 1

Introdução

A fala humana é um dos meios mais importantes que permite a comunicação entre pessoas por todo o Mundo. É baseada num conjunto de regras que ditam a criação de palavras e frases a partir de grandes vocabulários de modo a permitirem a construção de frases bem conjugadas numa linguagem, onde cada palavra é composta por um conjunto de fonemas. Um fonema é uma unidade básica da fonologia de uma linguagem que transporta a informação linguística da fala. Mais concretamente, uma palavra é criada a partir da combinação fonética de um conjunto limitado de fonemas que dependem da respetiva linguagem. É a combinação dos vocabulários, dos seus conjuntos de regras e dos seus sons fonéticos que permitem a existência de milhares de tipos de linguagens humanas que são entre si ininteligíveis. [39]

1.1 Definição do Problema

Com a evolução tecnológica, a maior parte da fala é agora convertida em ficheiros de áudio pelas mais variadas razões: para o seu armazenamento, para a sua manipulação, para as comunicações de voz entre pessoas, etc. Tem grande interesse a existência de um sistema automático que seja usado em tempo real e que rapidamente localize palavras ou frases nestes ficheiros de áudio, isto é, um sistema rápido de pesquisa em material de áudio. Pode-se constatar que um sistema deste género está perante um problema de deteção, onde se pretende encontrar um *query* num ficheiro de áudio. Usualmente aquilo que se procura (*query*) está sob a forma textual mas pode também ser uma expressão ditada pelo utilizador do sistema. Neste caso temos um *query* de áudio, que é o caso abordado nesta dissertação. Resumidamente, é possível afirmar que se pretende detetar

áudio em áudio.

Um exemplo para o qual este sistema teria uma grande utilidade seria para as empresas de radiodifusão, uma vez que são obrigadas por lei a gravar e a armazenar todos os seus programas. Imagine-se que se pretendia encontrar um ficheiro de áudio de um programa de Rádio que continha uma palavra-chave e que se sabia que este ocorreu com certeza nos últimos 10 anos. O tempo que um humano levaria a analisar toda a base de dados seria o tempo da duração total da mesma contra o tempo que levaria um sistema descrito anteriormente, uma questão de minutos ou horas.

Com o intuito de desenvolver o sistema, a primeira abordagem inicialmente pensada foi transcrever o texto de uma *query* para uma sequência de fonemas e posteriormente fazer a sua procura em todo o áudio. Para essa procura ser possível, era necessário que todo o áudio fosse pré-processado passando do domínio acústico para o domínio fonético e para tal seria necessário um reconhecedor de fonemas. Esse reconhecedor de fonemas iria criar um mapa do áudio onde iria ser assinalada a existência dos vários fonemas, uma espécie de assinatura de cada áudio. Estas assinaturas seriam as probabilidades de ocorrência de um dado fonema num dado instante do ficheiro de procura. Assim, em vez de o ficheiro a procurar ser o áudio acústico, seria um ficheiro com as melhores probabilidades de sequências de fonemas do respetivo áudio.

Uma alternativa semelhante a esta abordagem seria a de utilizar uma *query* de áudio acústico e fazer a sua conversão para o domínio fonético bem como fazer toda a conversão de todo o áudio que se pretende analisar, obtendo assim as melhores sequências de fonemas de ambos. Por fim seria necessário realizar a comparação da sequência de fonemas da *query* contra todas as sequências de fonemas dos áudios.

Por outro lado, também seria possível trabalhar apenas no domínio acústico, onde se iria fazer a comparação das assinaturas acústicas da *query* e do áudio. Esta comparação poderia ser realizada através da técnica de Dynamic Time Warping (DTW) [18] que permite fazer a expansão/compressão da *query* ou do áudio de modo a que seja possível realizar a correspondência entre ambos.

Conclui-se que a melhor abordagem seria realizar a conversão do domínio acústico para o domínio fonético, isto é, obter a melhor transcrição possível em termos de sequências de fonemas de todas as *queries* e de todos os áudios. Estas sequências seriam obtidas através de um sistema de reconhecimento de fonemas e apenas seria necessário realizar uma vez todo este processo. O

resultado deste passo seria um mapeamento do áudio onde para cada ficheiro se iria obter uma matriz que caracteriza as probabilidades de ocorrência dos fonemas de acordo com a variação temporal do ficheiro de áudio em questão, i. e., para cada trama temporal iria existir a probabilidade de ocorrência de cada um dos fonemas de uma dada língua. Posteriormente, usando uma técnica baseada nas sequências das medidas probabilísticas de distâncias de fonemas, é possível utilizar um critério que calcula distâncias que vão especificar o quanto distante se encontra o fonema da *query* em relação ao fonema do áudio num determinado instante de tempo. Estas distâncias vão corresponder a um valor baixo quando existir uma igualdade de fonemas e a um valor alto para o caso contrário. Para tal foi considerada uma matriz de distâncias em que no eixo das abcissas se encontra o número de tramas do respetivo áudio e no eixo das ordenadas o número de tramas da respetiva *query*. Esta matriz vai permitir verificar se existe a ocorrência da *query* no áudio, e existindo vai haver um rasto de pequenas distâncias ao longo do tempo, uma vez que tanto a *query* como o áudio existe a mesma sequência de fonemas. A técnica utilizada vai permitir que este rasto seja comprimido/expandido em conformidade com as durações dos fonemas da *query* e do áudio.

1.2 Processamento de termo curto dos sinais de áudio

A fala é produzida através de um sistema de trato vocal humano com uma excitação variante no tempo. Como resultado, um sinal de fala é por sua natureza não-estacionário. De forma a ser possível extrair a informação linguística que está codificada no sinal acústico da fala é necessário utilizar uma técnica de análise espectral de termo curto do sinal [11]. Nesta técnica admite-se que num curto espaço de tempo o espectro do sinal vai ser estacionário. Uma vez que o espectro do sinal só pode variar tão depressa o quanto os órgãos humanos conseguem produzir a fala e tendo em conta que o trato vocal varia lentamente, por convenção considera-se que o espectro vai ser constante quando observado em tramas de 10-30ms, aproximadamente.

Com base na metodologia implementa em [31], considerou-se a utilização de tramas com sobreposição, de comprimento de 25ms e deslocamento de 10ms, assumindo-se que a fala é estacionária nestas tramas. O resultado é um vetor com as características da fala a cada 10ms do sinal. Estes vetores de características podem ser vistos como pontos num espaço de características de N-dimensões, onde N representa a dimensão dos vetores. Estes vão representar todas as características presentes no sinal de fala: a fala, a influência de todo o canal de transmissão (ar,

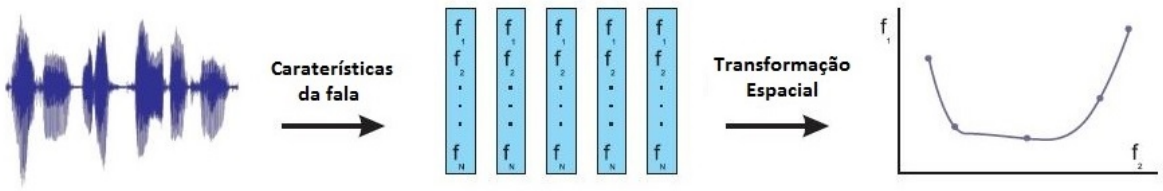

Figura 1.1: Exemplo de criação dos vetores de caraterísticas da fala. Editado de [31].

microfone, canal de comunicação, etc), o estado dos nossos órgãos de articulação, etc. Uma vez que o movimento dos órgãos de articulação é lento, os pontos que representam as características de vetores vizinhos neste espaço de N-dimensões vão ser também próximos no espaço. Uma pequena distância entre dois pontos vizinhos vai indicar que se está perante duas tramas semelhantes, indicando possivelmente o mesmo fonema, e uma grande distância vai indicar precisamente o contrário, que existe uma possível transição entre fonemas. Um conjunto destes pontos vai formar uma trajetória, e esta pode ser vista como o resultado do processo de criação da fala. Este processo carateriza-se como um ponto que se move com uma velocidade variável neste espaço de características de N-dimensões, onde a velocidade é maior em partes não-estacionárias da fala e menor em partes estacionárias da fala. Esta velocidade revela-se importante uma vez que transporta informações cruciais acerca das durações dos fonemas.

Posteriormente à extração das tramas de áudio por aplicação de uma janela de *Hamming* passa-se para o domínio da frequência realizando o cálculo da Discret Fourier Transform (DFT) de cada trama do sinal.

De forma a simular as restrições de resolução em frequência do nosso sistema auditivo, pode consisderar-se uma escala de resolução de tom (escala mel) [38]. Este passo deriva do facto de que a sensibilidade do ouvido humano não ser igual para todas as frequências - ela diminui com a frequência. Utiliza-se então um sistema baseado em filtros triangulares numa escala de melodias onde uma melhor resolução do espectro é preservada para as baixas frequências em relação às altas frequências. Um vetor deste banco de filtros de energias obtido para uma trama pode ser visto como uma versão perceptual do espectro original. E, de acordo com a perceção humana da sonoridade do som é aplicada uma escala logarítmica às energias espectrais. O vetor de caraterísticas fica por fim descorrelacionado e a sua dimensão é reduzida através da aplicação de Discret Cosine Transform (DCT). Obtêm-se assim os coeficientes que vão definir o vetor de espaço de N-dimensões. Estes coeficientes são conhecidos como Mel Frequency Cepstral Coefficients

(MFCC) [7]. Os MFCC são amplamente usados hoje em dia em reconhecimento automático da fala e têm um papel importante no sistema de reconhecimento de fonemas utilizado neste projeto. As redes neuronais vão trabalhar de acordo com a entrada destas tramas codificadas já no domínio *cepstral*.

1.3 Reconhecimento de fonemas com Redes Neuronais

Uma rede neuronal articifial (Artificial Neural Network, ANN) pode ser vista como um processador com uma enorme capacidade de computação paralela e com uma propensão natural para o armazenamento de conhecimento experimental, de modo a permitir a sua utilização [13]. Consiste num conjunto de unidades de processamento (neurónios) com ligações sinápticas a outras unidades neuronais. Ela assemelha-se ao cérebro humano em dois aspetos:

- O conhecimento experimental é adquirido pela rede através de um processo de aprendizagem.

- As forças das conexões entre neurónios são conhecidas como pesos sinápticos e estes são usados para armazenar o conhecimento experimental.

O procedimento utilizado para o processo de aprendizagem designa-se por algoritmo de aprendizagem, que tem como função modificar os pesos sinápticos da rede neuronal de acordo com a informação recebida com o objetivo de esta se adaptar a esta informação.

Um neurónio é considerado uma unidade de processamento de informação fundamental para as operações de uma rede neuronal. A forma de como os neurónios são dispostos na estrutura da rede está relacionada com o algoritmo de aprendizagem utilizado para o treino da rede. É geralmente composto por três elementos básicos: um conjunto de ligações sinápticas onde cada ligação é caracterizada pelo seu próprio peso, um somatório que permite a soma dos sinais após a aplicação do respetivo peso sináptico e uma função de ativação que permite limitar os níveis de amplitude da saída do neurónio. Um exemplo para o modelo de um neurónio pode ser observado na figura 1.2. A forma da função de ativação mais utilizada para a construção de redes neuronais artificiais é do tipo *sigmoid*. Esta é caraterizada por ser uma função crescente, suave e com propriedades assimptóticas da função de degrau unitário.

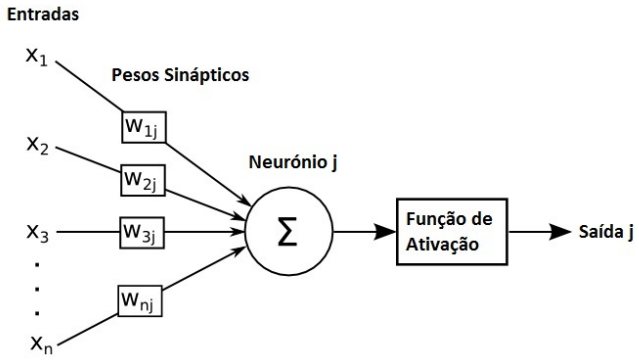

Figura 1.2: Exemplo de um modelo de neurónio: O neurónio j com os pesos que caraterizam as ligações sinápticas de cada entrada e a sua posterior soma, seguida da aplicação de uma função de ativação para a normalização de amplitudes da sua saída. Editado de [6].

A arquitetura da rede neuronal que foi aceite de uma forma geral e que melhor se enquadra com o reconhecimento de fala é denominada por Multilayer Perceptron (MLP) [2]. É caraterizada por ser uma rede sem realimentação (*feedforward*) com várias camadas e tipicamente é constituída por um conjunto de unidades sensoriais (nodos de entrada) que formam a camada de entrada, de uma ou mais camadas escondidas de nodos de computação e de uma camada de saída de nodos que contém informação acerca da resposta da rede. O sinal de entrada propaga-se pela rede de camada em camada na direção convencional, i. e., da entrada para a saída. Um exemplo de uma arquitetura de uma MLP pode ser observado na figura 1.3.

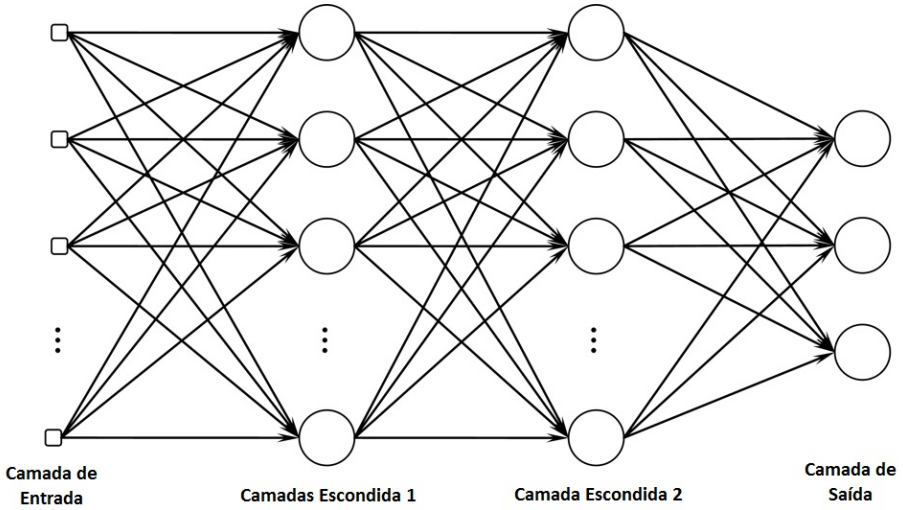

Figura 1.3: Exemplo da arquitetura de uma MLP: A camada de entrada contém os nodos que vão receber a informação, que por sua vez vai ser fornecida aos nodos de computação das camadas escondidas e manipulada de acordo com os pesos sinápticos das ligações e as funções de ativação, até atingirem os nodos da camada de saída com a resposta da rede neuronal. Editado de [30].

Os neurónios utilizados nestas redes são conhecidos como *perceptrons*, e são a forma mais simples da rede neuronal usada para a classificação de padrões binários, e consiste tipicamente num neurónio com pesos sinápticos adaptáveis e com um limiar de decisão. Um exemplo de um modelo de um *perceptron* pode ser observado na figura 1.4.

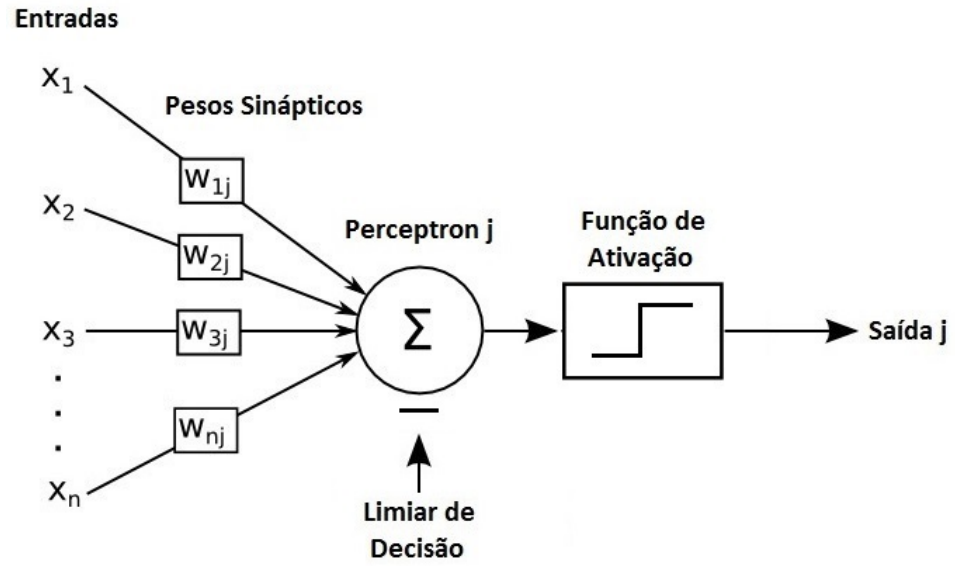

Figura 1.4: Exemplo de um modelo de um *perceptron*: O *perceptron* j com os pesos que caraterizam as ligações sinápticas de cada entrada, com uma função de ativação para binarizar a resposta da saída j em função da diferença da soma dos pesos sinápticos e de um limiar de decisão (BIAS). Editado de [6].

As MLP apresentam bons resultados em relação a diversos problemas complexos quando treinadas com o algoritmo de aprendizagem denominado por algoritmo da retropropagação do erro (*error back-propagation algorithm*). Este processo consiste em duas propagações pelas diferentes camadas da rede: uma propagação no sentido convencional que permite o cálculo do erro nas saídas (*forward pass*), e uma propagação no sentido inverso desse erro (*backward pass*). Na propagação convencional, a informação é fornecida aos nodos sensoriais da camada de entrada e o seu efeito é propagado através da rede, de camada em camada, até aos nodos da camada de saída onde é obtida uma resposta da rede neuronal. Durante esta propagação todos os pesos sinápticos da rede não sofrem quaisquer alterações. Idealmente, a resposta da rede neuronal seria um mapeamento perfeito da informação de entrada, o que na prática é muito complicado de se verificar. A diferença entre o mapeamento perfeito e o mapeamento obtido pela rede neuronal permite a realização do cálculo de um sinal de erro. Este sinal vai então percorrer as ligações sinápticas no sentido inverso, calculando o gradiente do erro em relação a cada peso da rede. Este gradiente serve depois para aplicar uma correção aos pesos sinápticos que vai permitir uma apro-

ximação da resposta da rede neuronal em relação à resposta ideal. Este processo repete-se por um determinado número de vezes até que o algoritmo atinja a sua condição ótima de paragem.

Para que os gradientes possam ser calculados, a função de ativação deve ser continuamente derivável e em vez de uma função em degrau usa-se uma versão suave desta não linearidade: a função sigmoide (função logística):

$$y_j = \frac{1}{1 + \exp(-v_j)} \quad , \tag{1.1}$$

onde v_j é o nível de atividade interna do *perceptron* j na rede e y_j é a saída do *perceptron*. Uma atividade elevada conduz a uma saída próxima de 1 e uma atividade muito negativa a uma saída próxima de zero. Uma das outras caraterísticas é a de a rede conter uma ou mais camadas escondidas que permitem que a rede adquira conhecimentos acerca de tarefas complexas ao extrair progressivamente a informação das entradas da rede.

O processo de reconhecimento de fonemas fica mais claro com o conhecimento da forma de funcionamento de uma rede MLP. Após a análise de termo curto de sinais de áudio descrita em 1.2, é possível realizar um treino de uma rede neuronal controlando a informação fornecida aos nodos da camada de entrada. Os *perceptrons* das camadas escondidas adaptam-se a essa informação de modo a que a resposta da rede neuronal se aproxime da resposta ideal. Neste caso, a resposta ideal é um mapeamento perfeito entre os coeficientes MFCC de uma trama de um sinal acústico fornecida aos nodos de entrada para o fonema dessa trama. Assim, cada saída da rede representa um fonema. Existem tantas saídas quantos os fonemas da língua representada. Além disso, a função de ativação vai ser escolhida de forma a que a soma das saídas da rede seja unitária. A função que garante a soma unitária de N saídas é a função *softmax*, definida como:

$$z_j = \frac{\exp(y_j)}{\sum_{i=1}^{N} \exp(y_i)} \quad , \tag{1.2}$$

onde z_j é o valor da saída j normalizado entre 0 e 1, e y_j é valor da saída j. Desta forma, as saídas podem ser interpretadas como probabilidades *a posteriori* dos fonemas.

1.4 Sistemas de alinhamento temporal dinâmico

Alinhamento temporal dinâmico (DTW) é uma técnica que consiste em encontrar um alinhamento ótimo entre duas sequências, temporalmente dependentes, de acordo com certas restrições

[18]. Estas sequências sofrem uma transformação de uma forma não-linear que permite a comparação entre ambas.

Esta técnica tem sido usado de forma consistente para a comparação de diferentes padrões da fala no reconhecimento automático de fala [29]. Em campos como a recuperação de informação e a extração de informação, esta técnica foi implementada com sucesso para cooperar automaticamente com deformações temporais e com diferentes velocidades associadas a informação temporalmente dependente.

No âmbito desta dissertação, e uma vez que as informações obtidas através das redes neuronais artificiais são temporalmente dependentes (vetores de probabilidades ao longo do tempo), utiliza-se esta técnica para realizar a comparação destes vetores. Esta encontra-se explicada com mais detalhe no capítulo 3.

1.5 MediaEval

Esta dissertação foi desenvolvida no âmbito de um desafio que está integrado no *MediaEval* 2015 [20]. O *MediaEval* é uma iniciativa dedicada à avaliação do desempenho de sistemas aliados de novas técnicas e algoritmos para a manipulação de multimédia. Aborda diferentes tipos de conceitos como reconhecimento da fala, análise de conteúdo de multimédia, análise de áudio e música, redes sociais, etc.

O desafio em que se insere denomina-se por Query by Example Search on Speech Task (QUESST) [21], e propõe a procura por áudio em conteúdo de áudio usando uma *query* com conteúdo de áudio. Pretende-se então desenvolver um sistema que determine com que certeza uma *query* se encontra num dado ficheiro de áudio, sendo apenas necessário verificar que a *query* se encontra em qualquer parte do ficheiro de áudio em questão, independentemente da língua.

Para tal foi fornecido um conjunto de ficheiros de áudio derivados de cerca de 8 línguas, sendo a maior parte delas europeias. Estes áudios apresentam algumas limitações acerca dos materiais utilizados para a sua gravação, diversas condições acústicas (nomeadamente ruído e reverberação) e os mais variados tipos de pronunciação. Este conjunto de áudios é utilizado para realizar a procura das *queries* e, nesta edição, é composto por 11662 ficheiros com uma duração média de 6 segundos.

Em adição a estes ficheiros, foram disponibilizados também mais dois conjuntos de *queries*

de fala. Denominam-se por conjuntos de desenvolvimento e de avaliação. O conjunto de desenvolvimento é composto por 445 ficheiros de *query* com uma duração média de 1.4 segundos com indicação dos ficheiros de áudio onde estas queries estão presentes. Este conjunto serve para testes durante o desenvolvimento que permitem melhorar o sistema em construção. Finalmente, o conjunto de avaliação serve para obter os resultados finais utilizando o melhor sistema desenvolvido e que posteriormente serão enviados para a sua avaliação por parte da organização do *MediaEval*. Este conjunto é composto por 447 ficheiros de áudio com uma duração média de 1.3 segundos.

Estes dois conjuntos contêm 3 diferentes tipos de *query* propostos para o desafio em questão. Para tal foram propostos 3 tipos de pesquisa que refletem a maneira de como a *query* foi criada:

- **Pesquisa do Tipo 1: Exatamente igual.**
  Indica que a ocorrência de uma ou várias palavras do áudio devem igualar exatamente a representação lexical da *query*. Um exemplo para este tipo de pesquisa é considerar que a *query* continha as seguintes palavras 'white horse', e que esta deveria ser encontrada na seguinte locução de áudio que continha as palavras *"My white horse is beautiful"*, mas não ser encontrada na locução que continha as palavras *"The whiter horse is fast"*.

- **Pesquisa do Tipo 2: Pequenas variações lexicais e reordenação de palavras.**
  Entende-se por pequenas variações lexicais que a ocorrência de uma ou várias palavras possam diferir da forma lexical da *query*, quer no início ou no final da mesma. Um exemplo para este tipo de pesquisa seria a *query* com a palavra *"researcher"* ser encontrada num áudio que continha a palavra *"research"*, sendo o inverso deste caso também um exemplo possível para este caso.

  Por reordenação de palavras considera-se que numa locução de áudio com várias palavras pode existir uma mudança na ordem das mesmas em relação à ordem das palavras da *query*. Um exemplo para este caso seria que ao procurar a *query* com as palavras *"white horse"*, iria existir uma igualdade para o caso da locução do áudio conter as palavras *"horse white"*. Para este caso é também considerado que as *queries* não contêm nenhum silêncio entre palavras, mas que o áudio pode conter pequenos conteúdos entre as mesmas e que também pode existir uma pequena variação lexical em relação às palavras da *query*. Então ao procurar pelas palavras *"white horse*, estas deveriam também ser encontradas nas frases *"My horse is white"* e *"I have two white and beatiful horses"*.

- **Pesquisa do Tipo 3: Contexto de conversa.**

  Este tipo de pesquisa vai considerar cenários mais realísticos. As *queries* deste tipo vão ser compostas tanto por fala com conteúdo relevante como irrelevante. Neste caso as *queries* vão ser apenas uma parte da locução do áudio e este pode conter pequenos conteúdos entre as palavras das *queries* como silêncio, ruídos ou mesmo até palavras irrelevantes. Um exemplo complexo para este tipo seria *"OK Google, let me find some red [uh] white [pause] horse to ride this weekend"*. Neste caso é muito complicado fazer a distinção entre as palavras da *query*, *"white [pause] horse"*, e os conteúdos irrelevantes, *"OK Google, let me find some red [uh]"* e *"to ride this weekend"*.

Para a afinação do sistema são fornecidos vários ficheiros de *Ground truth* para os testes do sistema com as *queries* de desenvolvimento. Estes ficheiros contêm informação sobre a existência de cada *query* em relação a cada áudio. No total existem 4 ficheiros de *Ground truth*: um geral com todo o tipo de *queries* e três para cada um dos tipos de *queries* existentes, respetivamente.

Os resultados obtidos são avaliados de acordo com duas métricas: a métrica primária Normalized Cross Entropy Cost (CNXE) e a métrica secundária Actual Term Weighted Value (ATWV).

Term Weighted Value (TWV) é uma métrica muito conhecida definida pelo National Institute of Standards and Technology (NIST) [22] e é usada para a avaliação do desempenho de sistemas de deteção de *queries* de fala. O ATWV é calculado de acordo com uma decisão por *query* de Sim/Não atribuída a cada deteção do sistema. Podem existir dois erros: aceitar um *query* e ele não existir (falso alarme) ou não detetar a presença de um *query* (falsa rejeição ou *"miss"*). Mostra-se que o ATWV com um valor unitário representa um sistema com um desempenho ideal (sem falhas e sem falsos alarmes). Valores mais baixos que a unidade representam sistemas com um desempenho mais fraco (com algumas decisões erradas).

Em contraste com a métrica TWV que avalia as decisões do sistema, a métrica CNXE é calculada diretamente a partir dos resultados do sistema. Esta métrica calibra estes resultados, em relação a um ficheiro de *Ground truth* que não é fornecido pelos resultados dos sistema, assumindo que estes podem ser interpretados como relações quantitativas de logaritmos de verosimilhanças. Um sistema ideal teria um valor nulo nesta métrica, enquanto que um valor unitário indica um sistema pouco informativo e um valor maior que a unidade indica uma má calibração do sistema das relações quantitativas de logaritmos de verosimilhanças.

1.6 Melhoramento da SNR das locuções

Como referido em 1.5, com o objetivo de criar ambientes acústicos mais desafiantes, a organização do *MediaEval* contaminou consideravelmente os ficheiros de áudio com ruído e reverberação. Com o objetivo de contornar esta situação, desenvolveu-se uma técnica de melhoramento de Signal-to-Noise Ratio (SNR) baseado em deteção de ruído e de subtração espectral do mesmo. Esta técnica encontra-se descrita no capítulo 5.

Capítulo 2

Treino da Rede Neuronal Artificial

Este capítulo aborda todo o procedimento necessário e conhecimento adquirido para o processo de treino de uma rede neuronal. Para tal, considerou-se a utilização de uma ferramenta desenvolvida pelo grupo de Processamento de Sinal da Faculdade de Tecnologia da Informação, da Universidade de Tecnologia de Brno [33]. Este grupo especializa-se em diversas áreas tais como as de reconhecimento da fala, identificação de línguas e *keyword spotting* [36], [34]. É reconhecido por ter desenvolvido um dos melhores sistemas de reconhecimento de fonemas do mundo [32], sendo este muito popular entre sistemas para os desafios do *MediaEval*, como por exemplo o sistema desenvolvido pelo grupo *SPL-IT* do pólo de Coimbra em 2014 [26].

2.1 Arquitetura do Sistema

Este sistema de reconhecimento de fonemas utiliza uma arquitetura híbrida entre Hidden Markov Models (HMM) [28] e redes neuronais artificiais do tipo MLP, descritas em 1.3. As redes neuronais são treinadas para realizarem o mapeamento de parâmetros de entrada para probabilidades *a posteriori* de fonemas em função de uma etiquetagem rigorosa dos mesmos, e utiliza modelos HMM para uma descodificação da sequência ótima de fonemas dado o sinal acústico. A figura 2.1 contém um diagrama de blocos que mostra de forma simples a arquitetura do sistema. O sistema tem como parâmetros de entrada coeficientes MFCC derivados de energias de bancos de filtro do tipo MEL. São obtidas as energias de 15 bancos de filtro através de uma análise de termo curto do sinal de fala, como descrita em 1.2. Através destas energias são extraídos vetores temporais longos com durações de $310ms$ (31 tramas), que se traduzem em 31 valores que caraterizam a

evolução das energias das bandas criticas ao longo do tempo.

Caraterísticas destes vetores que se encontrem convolvidas no domínio temporal, encontram-se multiplicadas no domínio da frequência, que por sua vez se encontram somadas no domínio *cepstral* (logaritmo da frequência). A normalização da média e da variância ao longo destes vetores temporais permite a remoção de caraterísticas com pouco relevo para a análise, i. e., remove caraterísticas cujas respostas em frequência não variem temporalmente (respostas em frequência de microfones, canais de comunicação, etc.).

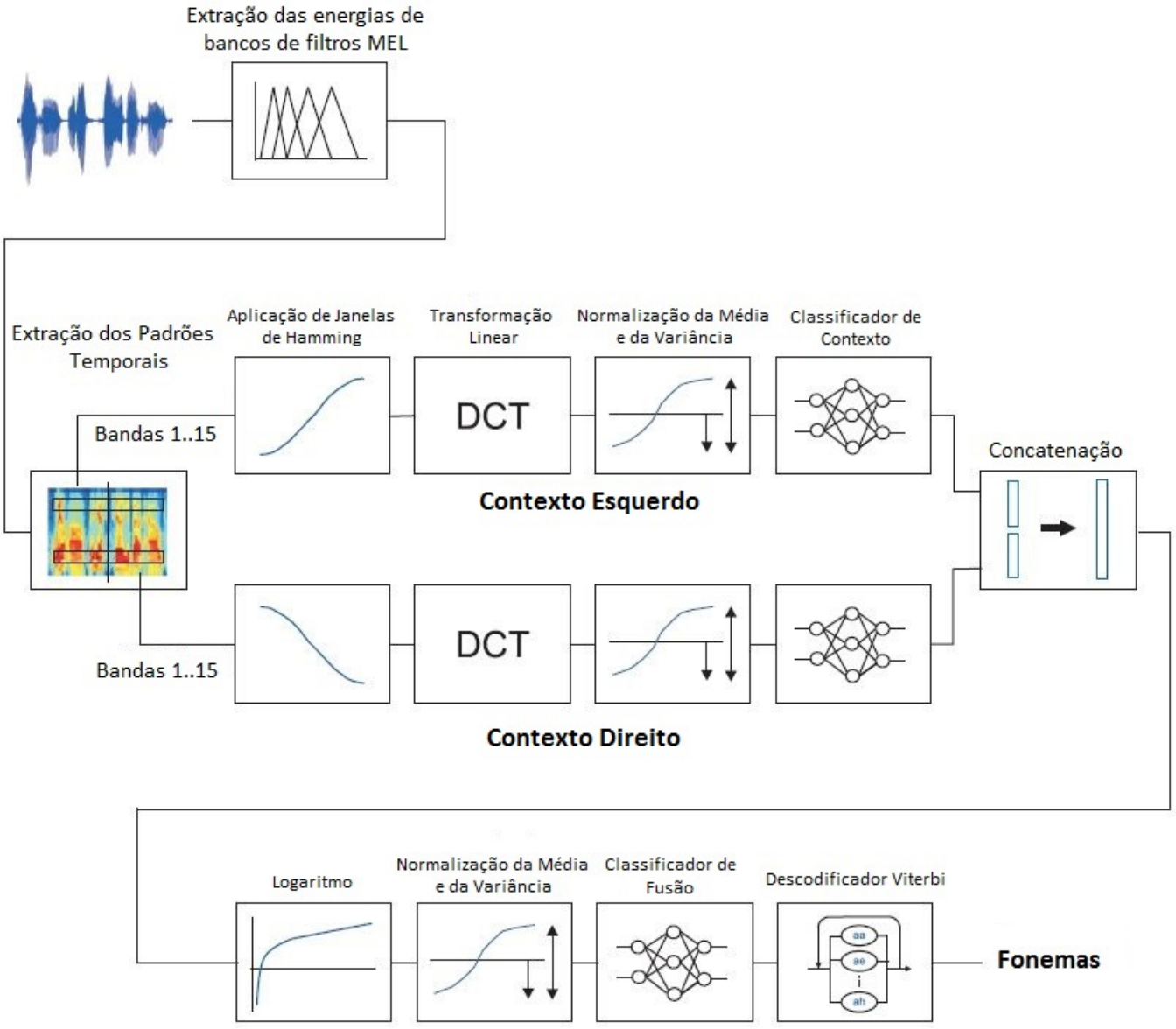

Figura 2.1: Exemplo da Arquitetura do Sistema de Reconhecimento de Fonemas com duas redes neuronais com contexto à esquerda e à direita e uma rede de fusão. Editado de [31].

O contexto temporal de cada vetor é então dividido em duas partes. Uma parte vai conter a informação acerca do seu passado temporal (15 tramas anteriores mais a trama atual) e a outra parte acerca do seu futuro temporal (15 tramas seguintes mais a trama atual). Denominam-se por contexto esquerdo e contexto direito, respetivamente. A figura 2.2 ilustra este processo de criação de contexto.

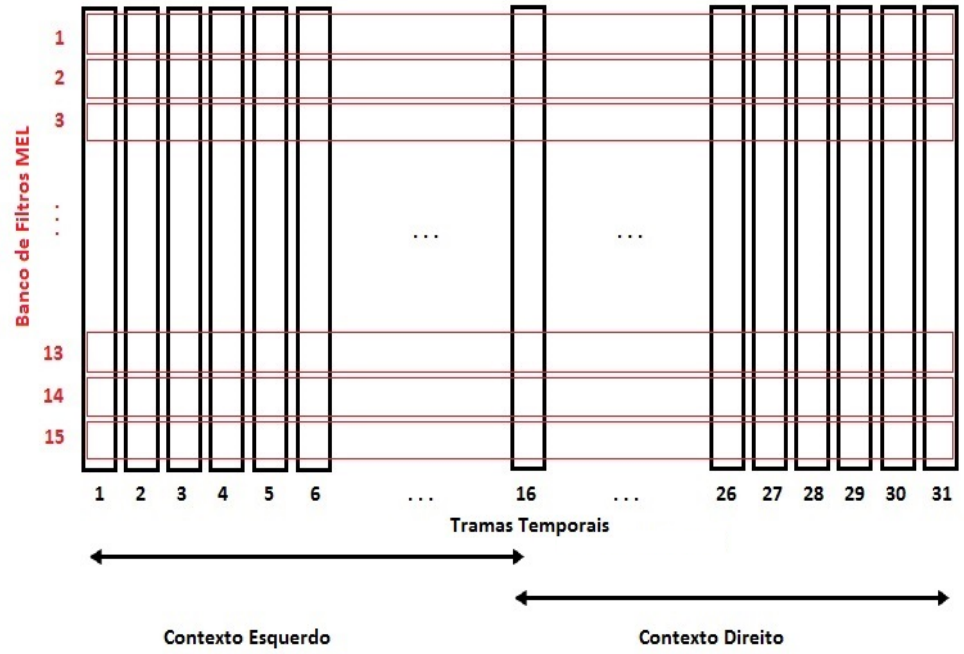

Figura 2.2: Exemplo da Criação de Vetores de Contexto Temporal: divisão de 31 tramas em 16 tramas à esquerda e 16 tramas à direita, sendo a trama atual a 16ª trama.

A cada contexto é aplicada a metade correspondente da janela de *Hamming*, onde se considera que a informação mais relevante é aquela que se encontra mais perto do presente, isto é, as janelas apresentam uma simetria em relação ao eixo vertical. A dimensão destes vetores é então reduzida de 16 para 11 coeficientes através da aplicação de uma transformação linear (DCT). Uma normalização da média e da variância do conjunto destes vetores garante que todos os coeficientes se encontrem na mesma gama dinâmica, o que se traduz num aumento da eficiência do treino. Estes vetores pré-processados são concatenados de acordo com o seu respetivo contexto, e são fornecidos respetivamente aos nodos de entrada de duas redes neuronais do tipo MLP.

As redes neuronais são treinadas para realizarem o mapeamento destes vetores de coeficientes MFCC para probabilidades *a posteriori* de fonemas em função de uma etiquetagem rigorosa dos mesmos. Estas redes têm uma estrutura idêntica, a qual se encontra exemplificada na figura 2.3.

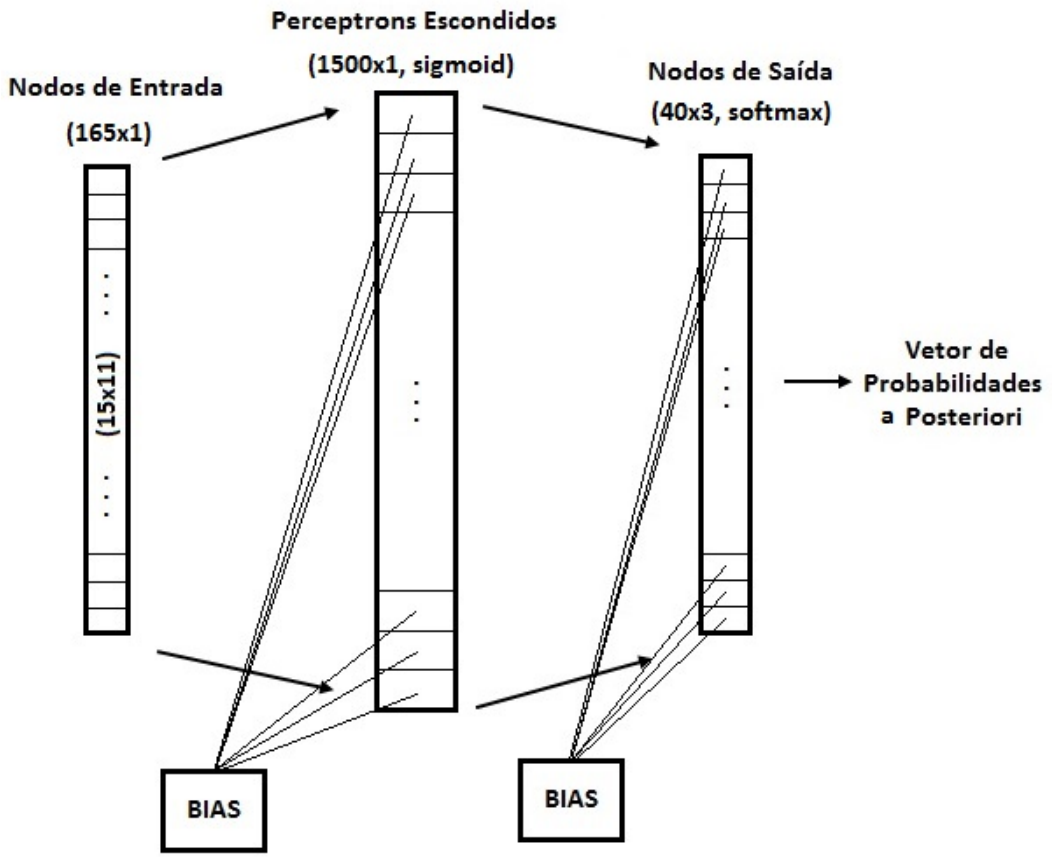

Figura 2.3: Exemplo da estrutura da rede neuronal MLP considerada para a língua portuguesa: É composta por 165 nodos de entrada, 1500 *perceptrons* na camada escondida aos quais é aplicado um limiar de decisão (BIAS) e uma função de ativação do tipo *sigmoid*. A camada de saída é composta por 120 nodos (39 fonemas da língua portuguesa + fonema 'oth', com 3 estados por cada fonema), aos quais também é aplicado um limiar de decisão (BIAS) e uma função de ativação do tipo softmax.

Estas redes neuronais são constituídas por 3 camadas: uma camada de entrada, uma camada escondida e uma camada de saída. A camada de entrada da rede neuronal é constituída por 165 nodos (15 *MelBanks* $\times$ 11 MFCC), que vão receber os vetores pré-processados. A camada escondida é composta por 1500 *perceptrons* que se vão "reagir"à informação de entrada, e aos quais é somada uma constante. No decorrer do treino, são estes que se vão alterando de modo a se adaptarem à informação fornecida.

O número de nodos da camada de saída vai depender do conjunto de fonemas definido para uma dada linguagem. A esse conjunto de fonemas é introduzido um fonema adicional, o fonema "*oth*". Teoricamente todos os ficheiros de fala considerados para o treino da rede neuronal estão etiquetados de forma correta mas, por vezes existem frações destes ficheiros que se encontram sem etiquetas. Este fonema serve para etiquetar essas frações de modo a garantir uma melhor distri-

buição de probabilidades *a posteriori* nas saídas da rede neuronal. Este fonema não é considerado aquando da realização da descodificação dos fonemas. Para os novos sistemas desenvolvidos nesta dissertação, descritos em 2.3, foram considerados dois conjuntos de 39 fonemas para as línguas portuguesa e inglesa, respetivamente, aos quais foi posteriormente adicionado o fonema 'oth'.

A descodificação é realizada com base numa das técnicas mais comuns para o reconhecimento de fala, os modelos de estados HMM [28]. Os estados introduzem a particularidade de se obter mais informação temporal relativa ao início, meio e fim de cada fonema. Considera-se assim a utilização de três estados por fonema que definem uma duração mínima de $30ms$ para os fonemas (3 tramas).

A camada de saída terá então um nodo por cada estado de cada fonema. A esta camada é aplicada uma não linearidade de *SoftMax*, descrita na equação 1.2, que vai garantir que a soma de todas as probabilidades seja unitária. Os vetores de probabilidades obtidos pelas redes neuronais de cada contexto são concatenados, transformados para o seu logaritmo e novamente normalizados na média e na variância de forma a garantir que se encontrem todos novamente na mesma gama dinâmica. Posteriormente, são enviados para a rede neuronal de fusão.

A estrutura da rede neuronal de fusão é semelhante às estruturas das redes neuronais para os contextos. A camada escondida contém o mesmo número de *perceptrons* e a camada de saída o mesmo número de nodos. A diferença está no número de nodos na camada de entrada desta rede neuronal, que vai ser o dobro do número de nodos das camadas de saída das redes neuronais de contexto, uma vez que vai receber como entrada os vetores de probabilidades concatenados. Esta rede tem como função fundir a informação probabilística obtida pelas redes neuronais dos contextos esquerdo e direito, i.e., passado e futuro, e apresentar uma resposta final novamente através de um vetor de probabilidades. A descodificação é por fim realizada através destes vetores com base num algoritmo de *Viterbi* [40].

2.2 Sistemas Disponibilizados

O grupo de Processamento de Sinal da Faculdade de Tecnologia da Informação, da Universidade de Tecnologia de Brno [33], disponibiliza quatro sistemas prontos a utilizar para o reconhecimento de fonemas nas seguintes línguas: Checo, Húngaro, Russo e Inglês. Estes sistemas apresentam a arquitetura descrita em 2.1, à exceção do sistema da língua inglesa.

Tabela 2.1: Sistemas de Reconhecimento de Fonemas disponibilizados por BUT Speech@FIT [33].

Sistema	Língua	Frequência de Amostragem	Número *Perceptrons*
PHN_CZ_SPDAT_LCRC_N1500	Checo	8 kHz	1500
PHN_HU_SPDAT_LCRC_N1500	Húngaro	8 kHz	1500
PHN_RU_SPDAT_LCRC_N1500	Russo	8 kHz	1500
PHN_EN_TIMIT_LCRC_N500	Inglês	16 kHz	500

Os sistemas foram treinados com as bases de dados: *Czech SpeechDat-E* [8], *Hungarian SpeechDat-E* [9], *Russian SpeechDat-E* [10], e *TIMIT* [5], respetivamente. A tabela 2.2 contém informação acerca do conjunto de fonemas utilizado para o treino e da taxa de erro de reconhecimento de fonemas para cada sistema.

Tabela 2.2: Número de Fonemas e Taxas de Erro de Reconhecimento de Fonemas (ERR) dos sistemas disponibilizados por BUT Speech@FIT [33].

Sistema	Fonemas	ERR (%)
PHN_CZ_SPDAT_LCRC_N1500	45	24.24
PHN_HU_SPDAT_LCRC_N1500	51	33.32
PHN_RU_SPDAT_LCRC_N1500	62	39.27
PHN_EN_TIMIT_LCRC_N500	39	24.24

2.3 Treino de Novos Sistemas

Com vista o novo desafio QUESST [21], treinaram-se dois sistemas adicionais seguindo a arquitetura descrita em 2.1. Estes sistemas foram treinados de acordo com um conjunto de scripts desenvolvidos e disponibilizados pelo grupo BUT Speech@FIT [33].

Estes scripts baseiam-se no excelente *software* disponibilizado por *QuickNet* [16] para o treino das redes neuronais, em diversas ferramentas do conjunto de ferramentas STK, como por exemplo

a ferramenta *SVite* que utiliza um algoritmo de *Viterbi* para a descodificação híbrida, e no conjunto de ferramentas Hidden Markov Model Toolkit (HTK) [15], como por exemplo a ferramenta *HResults* para a avaliação do desempenho dos sistemas.

O processo de treino destes novos sistemas foi um processo bastante penoso, uma vez que envolveu a realização de *debug* de um conjunto de scripts para se obter um conhecimento pormenorizado da arquitetura do sistema de reconhecimento de fonemas. Questões como a mudança de versões do sistema operativo, os diferentes conjuntos de fonemas para cada linguagem, as bases de dados a utilizar para o treino dos sistemas, entre outras coisas, revelaram um processo que não foi tão linear como era esperado.

O treino de uma rede neuronal requer uma divisão da base de dados em três conjuntos: os conjuntos de treino, de desenvolvimento e de teste. O conjunto de treino é composto por todos os ficheiros que vão ser fornecidos às entradas da rede neuronal para o processo de aprendizagem da mesma. O conjunto de desenvolvimento permite a avaliação de ficheiros aos quais a rede neuronal nunca se adaptou durante o treino, de modo determinar alguns parâmetros de afinação do sistema. Em relação ao conjunto de teste, este nunca esteve em contato direto com a rede neuronal durante o processo de treino. Serve para realizar uma avaliação da performance do sistema final.

Em adição às bases de dados, existe um ficheiro de anotação fonética do tipo Master Label File (MLF) que contém o alinhamento de todas as locuções de fala, ou seja, contém o tempo inicial e final de cada fonema presente na base de dados. Este ficheiro permite que a rede neuronal conheça a sua resposta ideal durante o seu treino para a possível utilização do algoritmo de aprendizagem descrito em 1.3.

A complexidade e duração do treino é definida por um número de iterações, que é deixado ao critério do desenvolvedor. A cada iteração é realizado o treino das três redes neuronais do sistema. As redes neuronais dos contextos esquerdo e direito são treinadas paralelamente em primeiro lugar, uma vez que são independentes uma da outra. Treinam-se por épocas, onde uma época consiste na propagação de todo o conjunto de treino no sentido convencional da rede e só depois na propagação inversa do erro, tal como descrito no algoritmo em 1.3. O número de épocas de treino de cada rede é definido quando este algoritmo de aprendizagem atinge a sua condição ótima de paragem, respetivamente. Os resultados destes conjuntos de épocas definem os pesos sinápticos das camadas escondidas destas redes.

Uma vez definidas as redes dos contextos esquerdo e direito, é realizado uma nova propagação da informação que permite obter as informações necessárias para a camada de entrada da rede neuronal de fusão. Com a fusão destas informações, o processo de obtenção dos pesos sinápticos para a camada escondida desta rede é realizado da mesma forma. Após atingida a condição ótima de paragem do algoritmo, armazenam-se os melhores pesos sinápticos obtidos para a nova rede, o que define o sistema final. Por último é realizada uma descodificação para os conjuntos de treino, teste e desenvolvimento, onde são obtidos os resultados que caraterizam o desempenho do sistema.

2.3.1 Sistema para a Língua Portuguesa

O sistema de reconhecimento de fonemas para língua portuguesa foi treinado com uma base de dados em Português Europeu, existente no Laboratório de Processamento de Sinal do Departamento de Engenharia Electrotécnica e de Computadores da Universidade de Coimbra. Esta base de dados é composta pela junção de três diferentes bases de dados utilizadas anteriormente no âmbito da realização de outros projetos:

- **Tecnovoz:** Esta base de dados foi revista no âmbito deste projeto e é composta por aproximadamente por 36.62% (78) locutores femininos e por 63.38% (135) locutores masculinos. Tem uma duração de aproximadamente 2.9 horas de fala.
- **Telejornal:** Esta base de dados contém locuções retiradas de telejornais previamente gravados. É composta por 71,818% (237) locutores femininos e 28.182% (93) locutores masculinos. Tem uma duração de aproximadamente 1.2 horas de fala.
- **Controlo:** Esta base de dados contém locuções de comandos de controlo de fala. É composta por 48.86% (3) locutores femininos e 57.14% (4) locutores masculinos. Tem uma duração de aproximadamente 1.9 horas de fala.

A junção destas bases de dados contém uma grande diversidade de locuções de fala limpa e de fala com ruídos. Na totalidade é composta por 57.82% (318) locutores femininos e 42.18% (232) locutores masculinos e tem uma duração de aproximadamente 6 horas de fala.

O conjunto de fonemas utilizado para o treino da rede neuronal para português europeu é derivado de Speech Assessment Methods Phonetic Alphabet (SAMPA) [27], que utiliza carateres do código ASCII que podem ser introduzidos por um teclado normal de computador, para

representar cada fonema. Esta solução torna-se prática pois é possível representar cada fonema apenas com um símbolo do teclado e tem por base o International Phonetic Alphabet (IPA) [37]. A lista de fonemas encontra-se no apêndice A. Foram considerados para este sistema um conjunto de 40 fonemas.

A base de dados é composta na totalidade por 3573 ficheiros de fala, dos quais 3073 foram aleatoriamente atribuídos ao conjunto de treino, 300 ao conjunto de desenvolvimento e 500 ao conjunto de teste. A escolha desta repartição foi feita empiricamente de acordo com o número de ficheiros disponíveis.

A evolução do erro (médio) aquando do treino deste sistema está representado na figura 2.4. Como esperado, o conjunto de treino vai apresentar sempre os melhores resultados, visto que é o conjunto ao qual o sistema tem uma melhor adaptação. Os conjuntos de desenvolvimento e teste apresentam resultados semelhantes, sendo o conjunto de teste quem vai definir o melhor resultado do sistema. De acordo com a figura 2.5, o melhor resultado foi obtido na iteração 30 com uma ERR de 31.31%.

De notar que este erro é aceitável num sistema de reconhecimento de fonemas dada a confusibilidade dos fonemas tomados de forma isolada, ou independente das palavras que formam. Este erro corresponde à análise das cadeias de fonemas depois da descodificação de *Viterbi* em comparação com a anotação de referência, considerando erros de substituição, apagamento e inserção de fonemas.

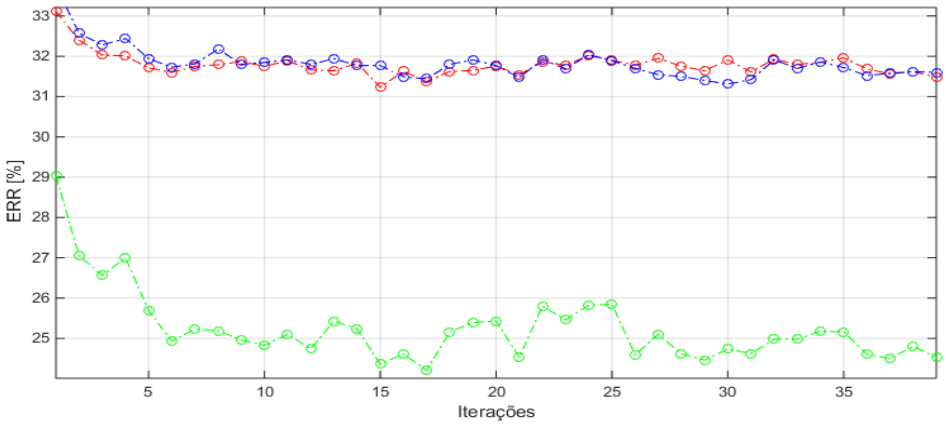

Figura 2.4: Evolução das ERR para os conjuntos: de Treino (verde), de Desenvolvimento (vermelho) e de Teste (azul), ao longo das Iterações do Treino do Sistema de Língua Portuguesa.

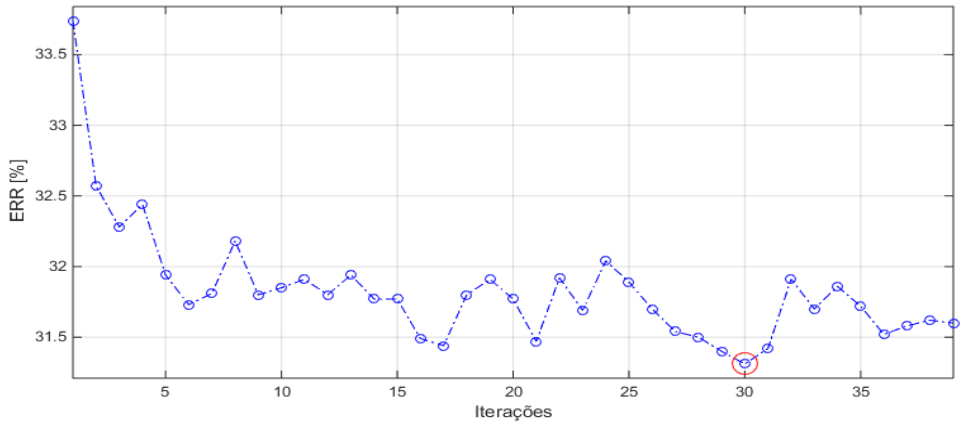

Figura 2.5: Evolução da ERR para o conjunto de Teste (azul), ao longo das Iterações do Treino do Sistema de Língua Portuguesa.

2.3.2 Sistema para a Língua Inglesa

Com o processo de treino de novos sistemas baseados em redes neuronais artificiais para o reconhecimento de fonemas dominado, e uma vez que o sistema para a língua inglesa em 2.2 não apresenta a mesma arquitetura dos sistemas pretendidos, procedeu-se ao treino de um novo sistema para esta língua.

A base de dados considerada para o treino deste sistema é composta pela junção de *TIMIT* [5] e *Resource Management* [4].

- ***TIMIT*:** Esta base de dados é composta por aproximadamente por 30% (192) locutores femininos e por 70% (438) locutores masculinos. Tem uma duração de aproximadamente 5.4 horas de fala.
- ***Resource Management*:** É composta por aproximadamente 32.2323% (53) locutores femininos e 67.8787% (112) locutores masculinos. Tem uma duração de aproximadamente 5 horas de fala.

A junção destas bases de dados contém uma grande diversidade de locuções de fala limpa. Na totalidade é composta por 30.8176% (245) locutores femininos e 69.1824% (550) locutores masculinos e tem uma duração de aproximadamente 10.5 horas de fala.

Tendo em conta que o desafio QUESST utiliza ficheiros de áudio com $8kHz$, realizou-se uma conversão da frequência de amostragem de ambas as bases de dados de $16kHz$ para $8kHz$. Devido também às condições acústicas desafiantes impostas pelo desafio, contaminaram-se as locuções

de fala limpa com um pouco de ruído branco, garantindo sempre uma SNR superior a $26dB$. Pretende-se com esta ação obter um sistema mais robusto em relação aos diversos tipos de ruído que terá de analisar.

O conjunto de fonemas considerado para o treino foi definido através conjunto de 39 fonemas do sistema da língua inglesa apresentado em 2.2. O mapeamento deste conjunto de fonemas baseia-se na conversão de fonemas de *TIMIT* considerada em [17] e [31]. O mapeamento do conjunto de fonemas de *Resource Management* foi também realizado com essa referência. A lista de fonemas encontra-se no apêndice B.

A base de dados é composta na totalidade por 10340 ficheiros de fala, dos quais 9810 foram aleatoriamente atribuídos ao conjunto de treino, 500 ao conjunto de desenvolvimento e 530 ao conjunto de teste. A escolha desta repartição foi feita empiricamente de acordo com o número de ficheiros disponíveis.

A evolução do treino deste sistema está representado na figura 2.6. Novamente, o conjunto de treino vai apresentar os melhores resultados uma vez que é o conjunto ao qual o sistema tem uma melhor adaptação. Os conjuntos de desenvolvimento e teste apresentam também resultados semelhantes, sendo o conjunto de teste quem vai definir o melhor resultado do sistema. De acordo com a figura 2.7, o melhor resultado foi obtido na iteração 7 com uma ERR de 25.93%. Por falta de tempo, não foi possível a realização de mais iterações de treino deste sistema.

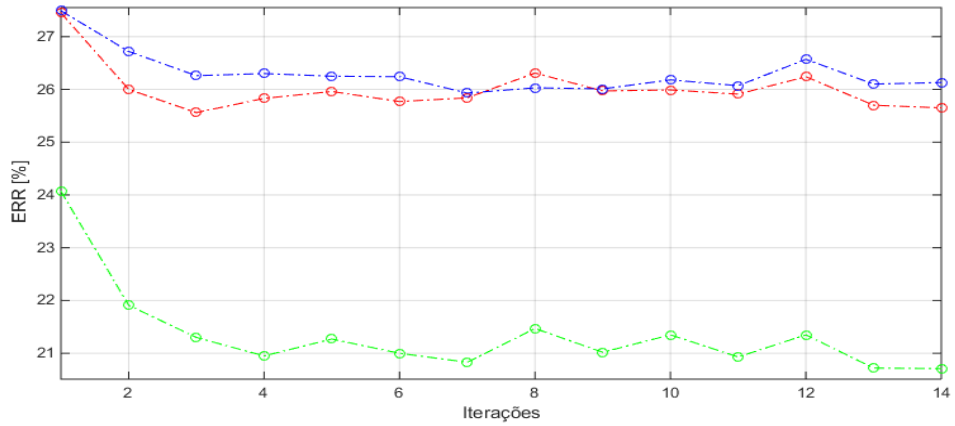

Figura 2.6: Evolução das ERR para os conjuntos: de Treino (verde), de Desenvolvimento (vermelho) e de Teste (azul), ao longo das Iterações do Treino do Sistema de Língua Inglesa.

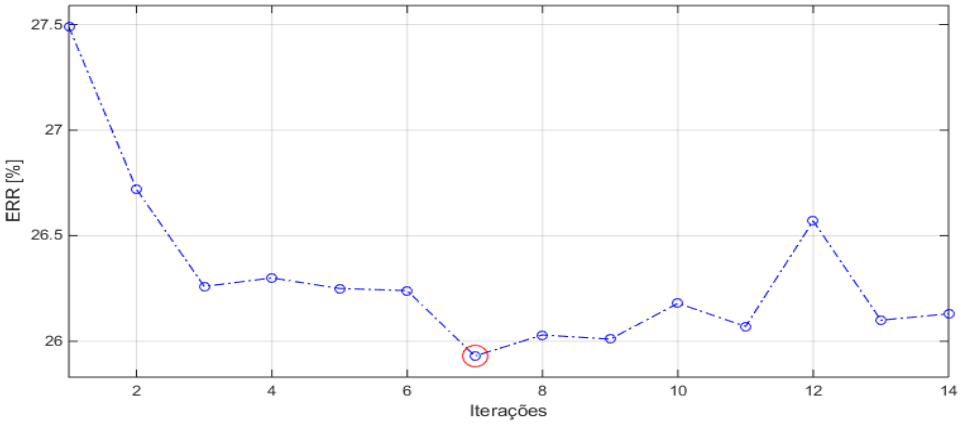

Figura 2.7: Evolução da ERR para o conjunto de Teste (azul), ao longo das Iterações do Treino do Sistema de Língua Inglesa.

Capítulo 3

Alinhamento Temporal Dinâmico

Através da aplicação do sistema de reconhecimento de fonemas aos ficheiros de áudio, obtém-se como resultado uma matriz que vai representar a probabilidade de ocorrência de todos os fonemas numa dada trama temporal. A esta representação fonética dá-se o nome de posteriorgrama [14]. Um exemplo pode ser observado na figura 3.1.

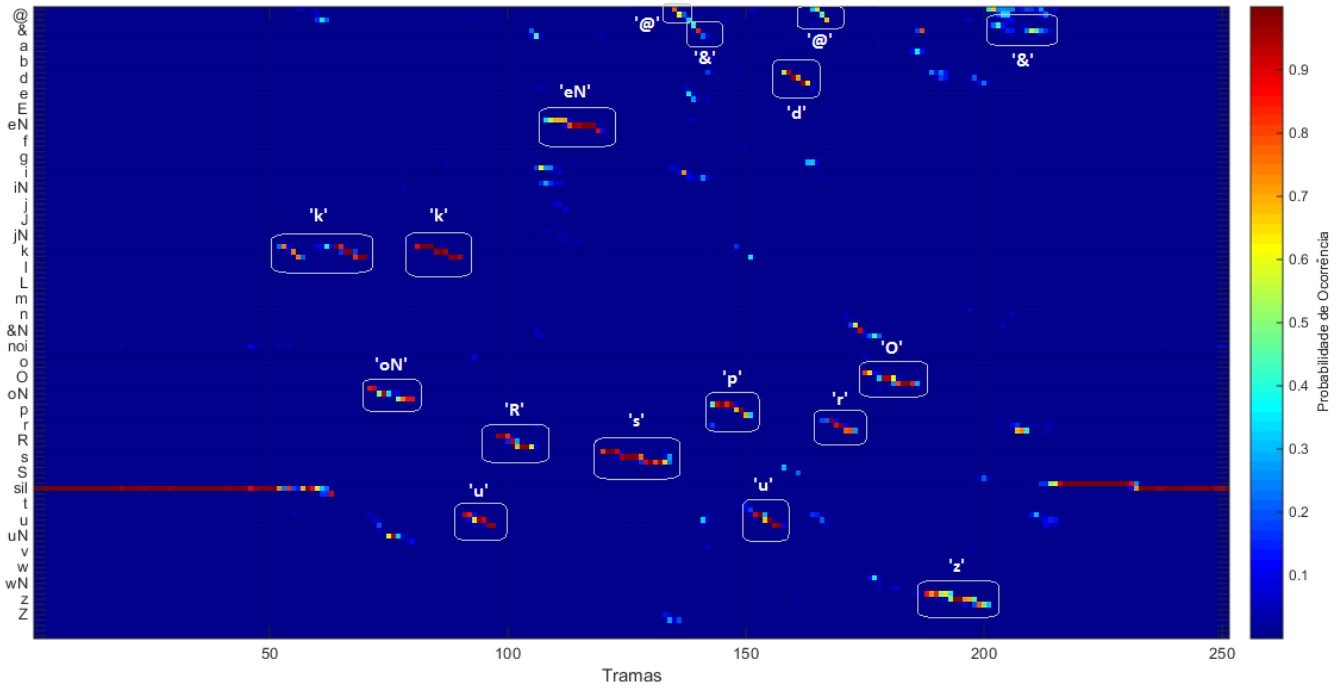

Figura 3.1: Exemplo de um posteriorgrama que contém as palavras *"concorrência poderosa"*, com respetiva transcrição fonética: "k-oN-k-u-R-eN-s-@-& p-u-d-@-r-O-z-&".

É um gráfico onde eixo horizontal vai representar as tramas temporais e o eixo vertical as probabilidades *a posteriori* dos 3 estados de cada fonema. Um tom de vermelho mais escuro na

figura representa uma probabilidade elevada de ocorrência do respetivo fonema, por enquanto que um tom de azul mais escuro representa o inverso, uma baixa probabilidade.

Esta representação fonética permite a criação de uma matriz de distâncias locais quando é realizada uma comparação baseadas nas tramas temporais entre o posteriorgrama da *query* e o posteriorgrama do áudio onde se vai realizar a procura. Esta matriz de distâncias locais permite determinar se existe alguma similaridade entre ambos os posteriorgramas, e também a aplicação do algoritmo DTW.

3.1 Cálculo da Matriz de Distâncias

Como descrito em [14], dadas duas distribuições de posteriorgramas, da *query* $\vec{q}$ e do áudio $\vec{x}$, a probabilidade de ambas resultarem no mesmo evento fonético é representada pelo seu produto escalar:

$$P(fonema\{\vec{q}\} = fonema\{\vec{x}\}) = \vec{q'} \cdot \vec{x} \tag{3.1}$$

Ao realizar a conversão desta probabilidade para o logaritmo, interpreta-se esta nova medida como sendo baseada em distâncias:

$$D(\vec{q}, \vec{x}) = -\log(\vec{q'} \cdot \vec{x}) \quad , \tag{3.2}$$

onde distâncias próximas de zero indicam um grande semelhança entre $\vec{q}$ e $\vec{x}$ enquanto que grandes distâncias representam o inverso. Na prática esta equação pode falhar quando os vetores de probabilidades $\vec{q}$ e $\vec{x}$ contém valores de zero, resultando num produto escalar $\vec{q} \cdot \vec{x} = 0$ que iria implicar uma distância local de $D(\vec{q}, \vec{x}) = \inf$. Para contornar esta situação, realiza-se uma suavização das distribuições dos posteriorgramas da seguinte maneira:

$$\vec{q'} = (1 - \lambda)\vec{q} + \lambda\vec{u} \tag{3.3}$$

Onde $\vec{u}$ representa uma distribuição de probabilidades uniforme e $\lambda = 10^{-4}$ garante uma probabilidade não nula para todos os fonemas em $\vec{q'}$. Para realizar a comparação de posteriorgramas de uma *query* e do respetivo áudio de procura, é calculada uma medida de semelhanças entre cada distribuição de posteriorgramas de todas as N tramas da *query* contra cada distribuição de posteriorgramas de todas as M tramas do áudio. O resultado é uma matriz de distâncias de $N \times M$ tramas, como se pode verificar na figura 3.3.

3.2 Estratégias com alinhamento temporal dinâmico

Idealmente, uma correspondência entre um segmento da *query* e um segmento do áudio de procura seria representado por uma diagonal, da esquerda para a direita e de cima para baixo, de uma sequência de pontos nesta matriz de distâncias. Foram consideradas como base as estratégias descritas em [25] e [26], sendo todas elas otimizadas neste projeto.

Primeira Estratégia

O principal objetivo desta estratégia é encontrar as *queries* do Tipo 1 descritas em 1.5, um exemplo pode ser observado na figura 3.3. Foi considerado que o caminho ótimo poderia seguir 3 diferentes movimentos diretamente para pontos adjacentes com a menor distância local na matriz de distâncias: horizontal, vertical e diagonal, como exemplificados na figura 3.2. Considerou-se que não haveria diferenciação na penalização dos movimentos, tendo todos eles um peso unitário.

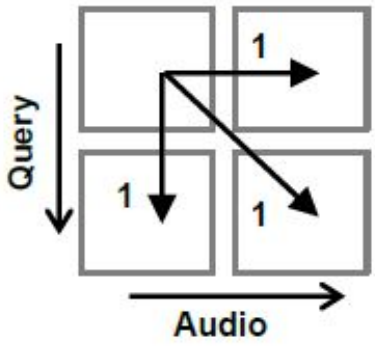

Figura 3.2: Exemplo de um esquemático de caminhos com peso unitário considerados para a criação da DTW. Retirado de [25].

A distância do caminho ótimo seria simplesmente a soma das distâncias ao longo do caminho, normalizadas pelo comprimento deste caminho. Para que a correspondência da *query* fosse possível em qualquer parte do áudio de procura, o início do alinhamento não foi restringido no áudio. Esta estratégia serve como base para as seguintes estratégias.

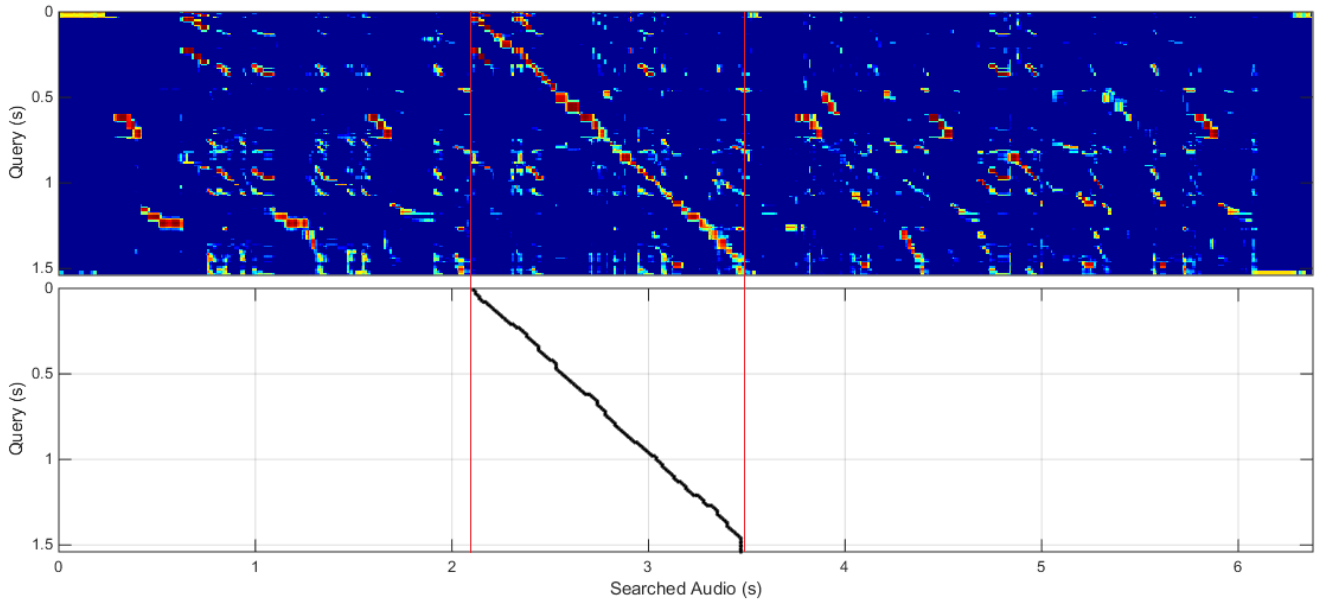

Figura 3.3: Exemplo de uma correspondência para *queries* do Tipo 1. Neste exemplo a *query* contém as palavras "*concorrência poderosa*", e o áudio de procura contém as palavras "*Só que os blogues são uma concorrência poderosa à centralização do poder da informação*". Na matriz de distâncias, está claramente identificada a diagonal que vai representar o caminho ótimo desta estratégia, bem como a parte do áudio onde se encontra a *query*.

3.3 Estratégias com alinhamento temporal dinâmico modificado

Tendo em conta as *queries* dos Tipos 2 e 3 descritas em 1.5, foram implementadas 4 estratégias de [25] para a pesquisa do Tipo 2 e uma nova estratégia para a pesquisa do Tipo 3, introduzida pela primeira vez na presente edição do desafio *MediaEval*. Estas estratégias são baseadas num alinhamento temporal dinâmico modificado que permite a obtenção de diferentes caminhos ótimos, de acordo com as necessidades de cada tipo de *query*. Para tal, são calculadas duas matrizes adicionais: uma matriz de distâncias acumuladas do caminho ótimo para cada ponto e uma matriz com informação para a reconstrução do caminho (*backtracking*). Estas matrizes vão permitir ter um maior controlo sobre o rastreamento do caminho ótimo de modo a encontrar *queries* de diferentes tipos. Todos os pré-requisitos presentes nas estratégias descritas nos próximos subcapítulos foram escolhidos com base em testes e nas especificações das bases de dados fornecidas pela organização do desafio *MediaEval*, que indicam que as palavras das *queries* são compostas por mais de 5 fonemas ($\approx 250ms$).

Segunda Estratégia

Esta estratégia aborda as variações lexicais no final das *queries*, um dos casos do Tipo 2 descrito em 1.5. Consideram-se cortes até $250ms$ no final da *query*, garantido sempre que o caminho ótimo tem uma duração acima de $500ms$. Um exemplo pode ser observado na figura 3.4.

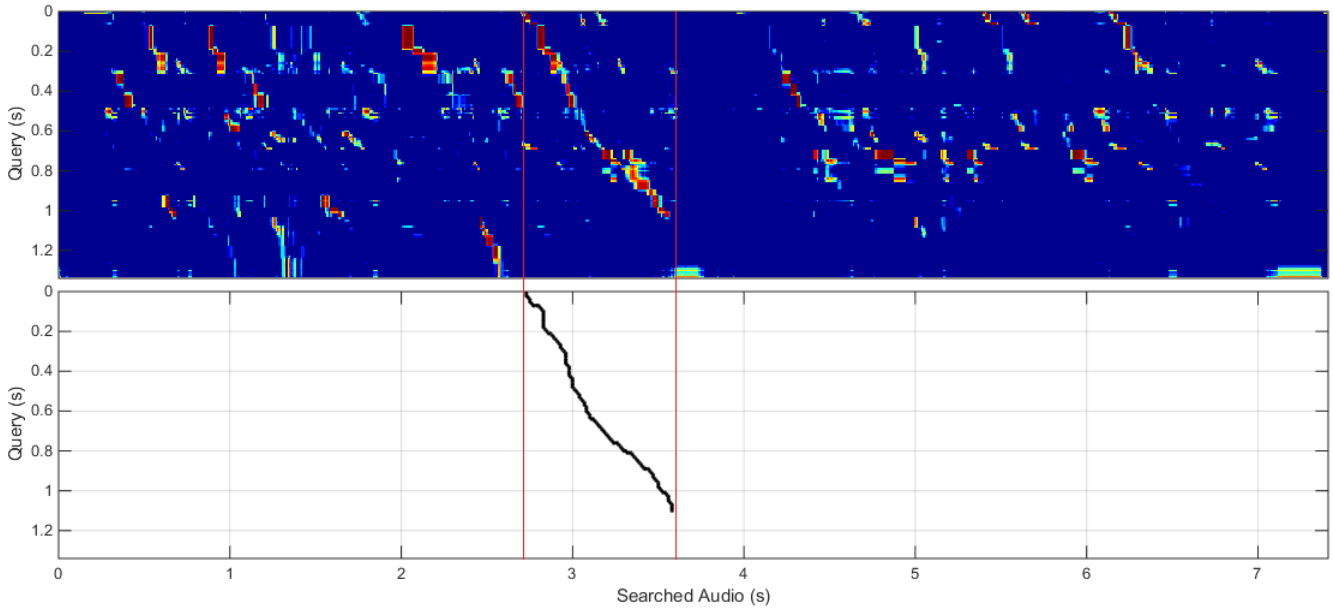

Figura 3.4: Exemplo de uma correspondência para *queries* do Tipo 2 com variação lexical no fim. Neste exemplo a *query* contém as palavras "*bares Labirintão*", e o áudio de procura contém as palavras "*Os quatro palcos da Rota do Jazz serão os bares Labirinto, Foz Clube e Aniki-Bóbó [silêncio] e o barco Endouro.*". Na matriz de distâncias, está claramente identificada a diagonal que vai representar o caminho ótimo parcial desta estratégia, bem como a parte do áudio onde se encontra a *query*.

Terceira Estratégia

Neste caso pretende-se o inverso da segunda estratégia, uma variação lexical no início das *queries*, também um dos casos do Tipo 2 descritos em 1.5. Consideram-se também cortes até $250ms$ mas agora no início da *query*, garantido sempre também que o caminho ótimo tem uma duração acima de $500ms$. Uma vez que neste caso a matriz de distâncias acumuladas não vai indicar diretamente os valores de novos caminhos possíveis, assume-se que os caminhos que já contêm a correspondência da *query* com o áudio, vão ser os caminhos com as menores distâncias. Para uma melhor eficiência computacional, apenas se realiza a reconstrução dos 5 melhores caminhos de modo a obter a melhor distância normalizada possível. Um exemplo pode ser observado na figura 3.5.

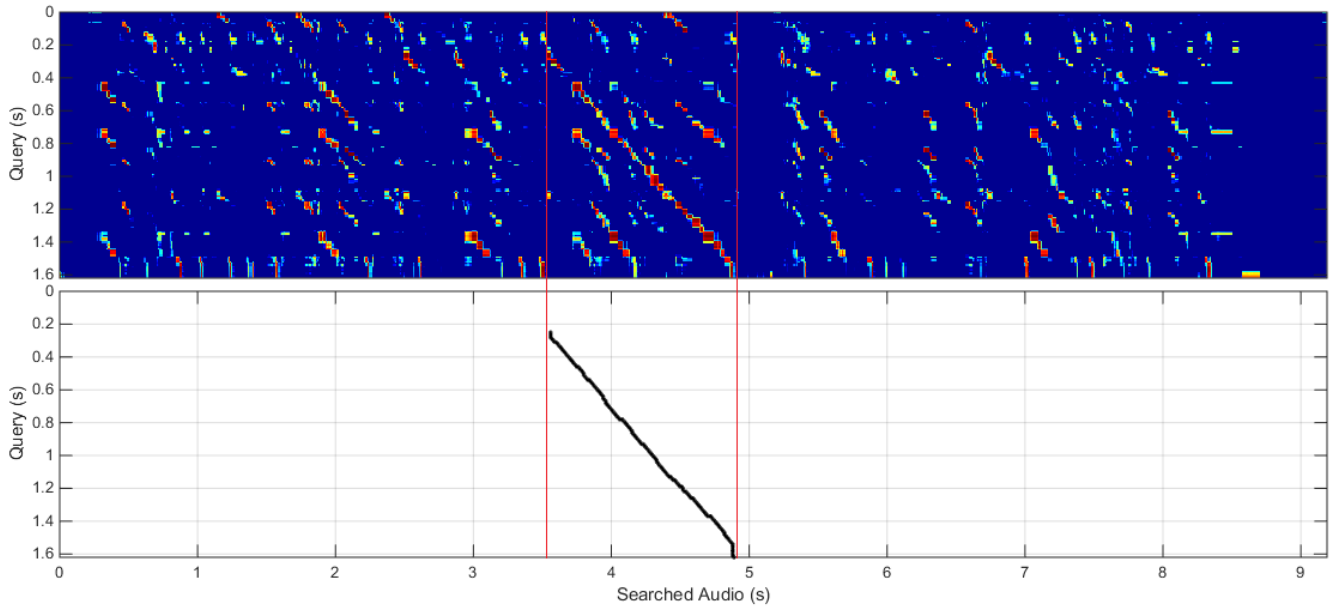

Figura 3.5: Exemplo de uma correspondência para *queries* do Tipo 2 com variação lexical no início. Neste exemplo a *query* contém as palavras "*pré-conferência de imprensa*", e o áudio de procura contém as palavras "*Foram estas as palavras proferidas por Cavaco Silva na conferência de imprensa,[respiração] realizada num auditório com uma centena de jornalistas.*". Na matriz de distâncias, está claramente identificada a diagonal que vai representar o caminho ótimo desta estratégia sem a parte inicial *"pré"*, bem como a parte do áudio onde se encontra a *query*.

Quarta Estratégia

Esta estratégia aborda a existência de pequenas palavras irrelevantes presentes no áudio de procura entre as palavras da *query*, outro dos casos do Tipo 2 descritos em 1.5. A solução encontrada para este caso foi a de permitir a realização de um salto horizontal no caminho ótimo, i.e., ao longo do áudio de procura. Os pré-requisitos necessários para a ocorrência deste salto são que: a *query* tem de ter pelo menos uma duração mínima de 800 *ms*, não pode ocorrer durante os primeiros e últimos 250 *ms* da *query* e que o tamanho do salto é no máximo metade do tamanho da *query*. Um exemplo pode ser observado na figura 3.6.

Quinta Estratégia

Esta estratégia tem em conta a reordenação de palavras da *query*, também do Tipo 2 descritas em 1.5. A abordagem considerada é semelhante à da quarta estratégia uma vez que também permite a existência de palavras irrelevantes entre os segmentos do áudio, com a diferença de que

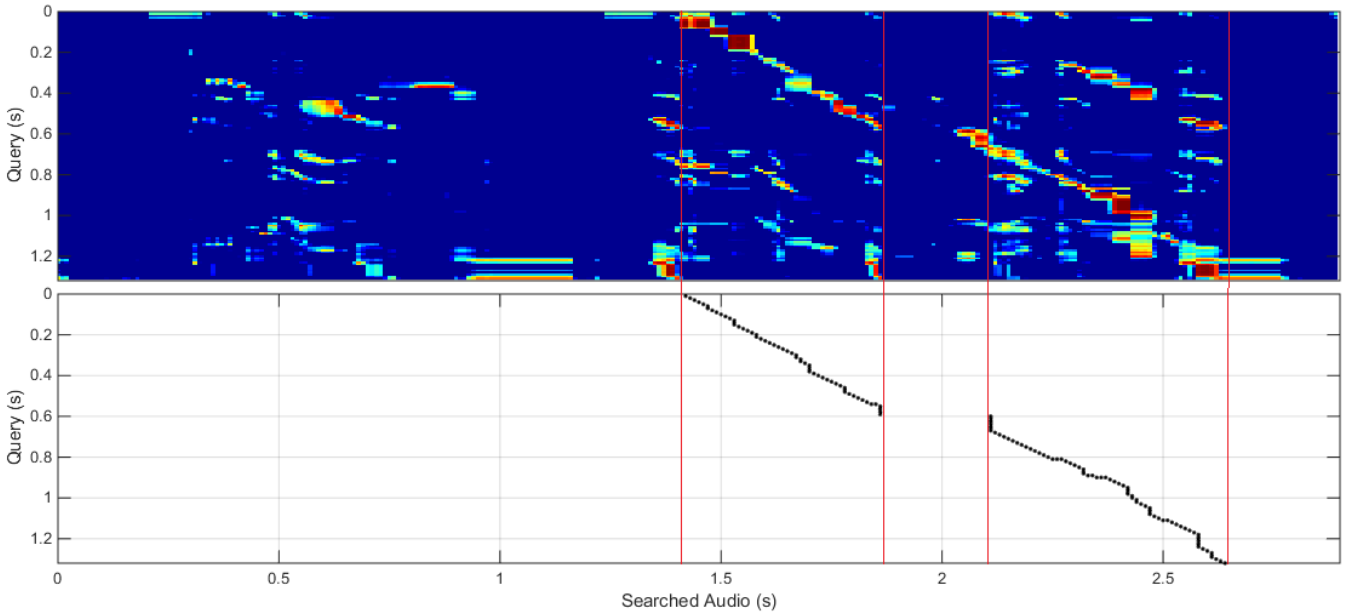

Figura 3.6: Exemplo de uma correspondência para *queries* do Tipo 2 com um salto horizontal no caminho ótimo. Neste exemplo a *query* contém as palavras "*estrela rebrilha*", e o áudio de procura contém as palavras "*Em Belém, a estrela não rebrilha.*". Na matriz de distâncias, estão claramente identificadas as diagonais que vão representar o caminho ótimo desta estratégia, bem como a parte do áudio onde se encontra a *query*.

agora existe uma troca desses segmentos, i.e., a primeira palavra da *query* deve ser encontrada no áudio depois da segunda palavra da *query*. Como na terceira estratégia, realiza-se apenas a reconstrução dos 5 melhores caminhos e estes vão permitir encontrar um segmento que contém a segunda palavra da *query*. Considera-se que o final deste segmento é um ponto de quebra e que a partir deste ponto se vai encontrar um caminho alternativo que melhor iguale o segmento da primeira palavra da *query*, garantindo-se que não existe a ocorrência de sobreposição entre os dois segmentos. Os pré-requisitos para a posição e comprimento do salto são iguais aos da quarta estratégia. Um exemplo pode ser observado na figura 3.7.

Sexta Estratégia

Esta estratégia foi desenvolvida para abordar as *queries* do Tipo 3 descritas em 1.5. Pode considerar-se de certa forma semelhante à quarta estratégia, uma vez que vai existir conteúdo irrelevante entre as palavras da *query*. A solução encontrada foi a de permitir a realização de um salto vertical no caminho ótimo, i.e., ao longo da *query*. Foram considerados os mesmo pré-requisitos para a realização salto que na quarta estraégia, com a adição de que é necessário que

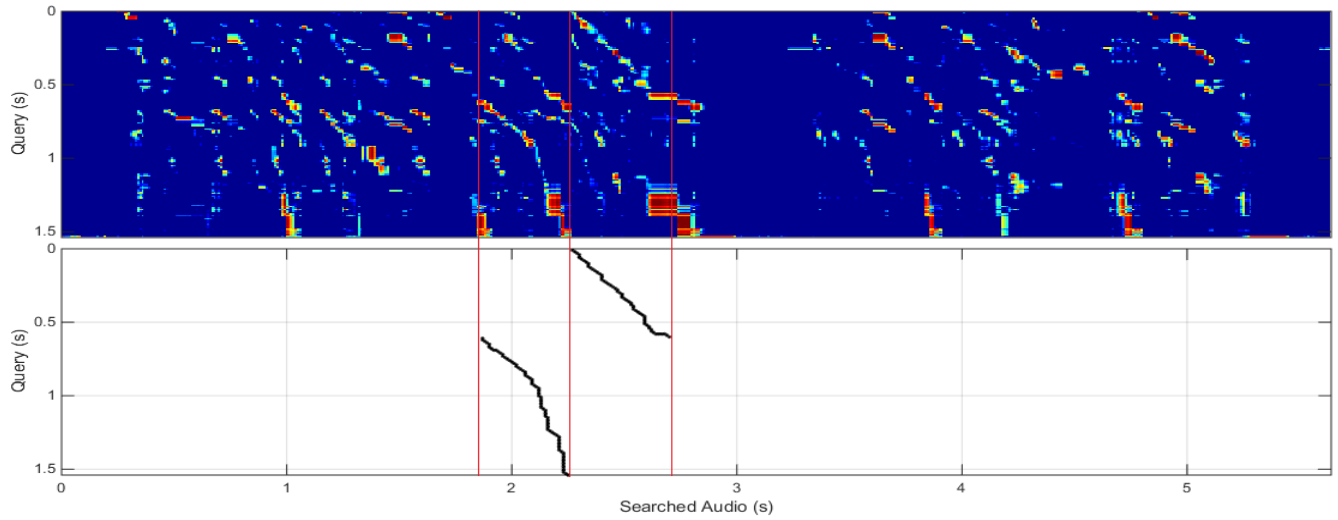

Figura 3.7: Exemplo de uma correspondência para *queries* do Tipo 2 com uma reordenação de palavras. Neste exemplo a *query* contém as palavras "*campeões actuais*", e o áudio de procura contém as palavras "*Conhece-se [silêncio] apenas uma derrota com os actuais campeões, [silêncio] por um a três, no pavilhão Borges Coutinho.*". Na matriz de distâncias, estão claramente identificadas as diagonais que vão representar o caminho ótimo desta estratégia, bem como a parte do áudio onde se encontra a *query*.

o comprimento máximo do salto seja no máximo 33% do tamanho da *query*. Um exemplo pode ser observado na figura 3.8

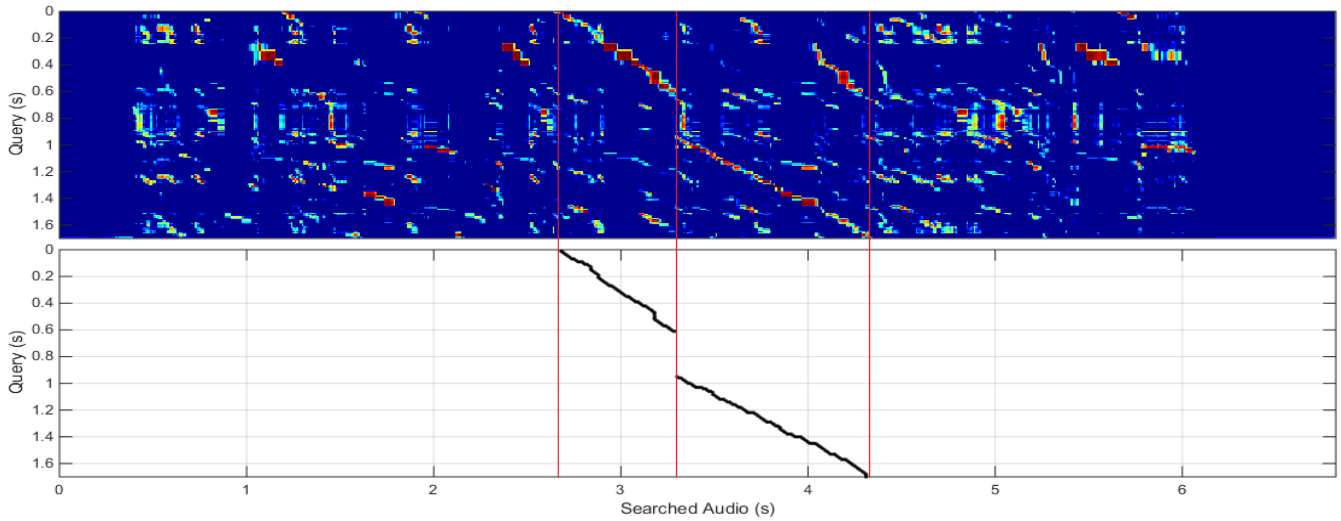

Figura 3.8: Exemplo de uma correspondência para *queries* do Tipo 3 com conteúdo irrelevante entre as palavras da *query*. Neste exemplo a *query* contém as palavras "*evolução [uh] desfavorável*", e o áudio de procura contém as palavras "*No comércio a retalho espera-se uma evolução desfavorável do volume de negócios.*". Na matriz de distâncias, estão claramente identificadas as diagonais que vão representar o caminho ótimo desta estratégia, bem como a parte do áudio onde se encontra a *query*.

Capítulo 4

Teste do Sistema Inicial

Como Sistema Inicial, recriou-se o sistema desenvolvido pelo Laboratório de Processamento de Sinal do Instituto de Telecomunicações do pólo de Coimbra (SPL-IT-UC) [26], no âmbito do desafio de QUESST para o *MediaEval* 2014 [19].

Posteriorgramas de fonemas com 3 estados

O primeiro passo consistiu em usar o reconhecedor de fonemas baseado em redes neuronais, desenvolvido por Brnu University of Technology (BUT) e descrito em 2, para a obtenção dos posteriorgramas das diferentes línguas para a base de dados de desenvolvimento e todos os áudios. Foram considerados os 3 sistemas disponíveis para áudio de 8 *kHz* das seguintes línguas: Checo, Húngaro e Russo, onde cada língua contém o seu respetivo conjunto de fonemas. Como resultado obtiveram-se os posteriorgramas de fonemas com 3 estados para os diferentes sistemas.

Corte de Silêncios e Ruídos dos *Posteriorgrams*

Posteriormente procedeu-se ao corte de todos os silêncios e ruídos dos posteriorgramas obtidos das *queries* da base de dados de desenvolvimento. Para tal, para cada trama considerou-se a soma das probabilidades dos 3 estados de cada fonema aos quais correspondem silêncio ou ruído. Após esta soma é calculada uma média aritmética de cada um destes fonemas considerando as 3 línguas. Caso esta média seja superior a 50%, procede-se á exclusão das respetiva trama do posteriorgrama em questão. Este processo permite o cálculo de uma matriz de distâncias mais "limpa" para a aplicação da DTW.

Aplicação da DTW

Após o corte de silêncios e ruídos dos posteriorgramas dos *queries*, procedeu-se ao cálculo da matriz de distâncias como descrito em 3.1. O passo seguinte foi a aplicação das 5 estratégias inicialmente consideradas, descritas em 3.2 e 3.3. Como resultado obtiveram-se as distâncias de cada uma das estratégias para cada par de *query*-áudio. Este passo revelou-se um processo consideravelmente longo.

Fusão e Calibração de Resultados

Para a fusão dos resultados e uma vez que é o processo com o qual se obteve melhor resultado de acordo com [25], é aplicada uma média harmónica às 5 distâncias obtidas pela aplicação das 5 estratégias da DTW, para cada par *query*-áudio. Esta fusão permite obter um valor único de distância para cada um destes pares. A média harmónica é descrita por:

$$\textit{Média Harmónica} = d_h = \frac{1}{\sum_{i=1}^{N} \frac{N}{d_i}} \quad , \tag{4.1}$$

onde d_i é a distância da estratégia i e N é o número das estratégias consideradas. Experimentalmente verifica-se que vai convergir para um valor próximo do mínimo destas 5 distâncias. Após esta fusão das distâncias obtidas, é realizada uma normalização por *query* através da subtração da média aritmética das distâncias e posterior divisão do desvio padrão das mesmas:

$$\textit{Normalização por Query} = \frac{d_h - \bar{d}_h}{\sigma_h} \tag{4.2}$$

O último passo antes da calibração dos resultados é a fusão das distâncias obtidas para as diferentes línguas. Para esta fusão é simplesmente considerada uma média aritmética das matrizes de distâncias já normalizadas das 3 línguas. A anterior normalização das distâncias permite obter as figuras de mérito necessárias para a calibração final dos resultados por simplesmente se considerarem os valores simétricos destas distâncias normalizadas.

A principal métrica do desafio, Cnxe, é calibrada através da aplicação de uma transformação linear ao conjunto de dados. Os parâmetros desta transformação linear são treinados no conjunto dos ficheiros de desenvolvimento através do conjunto de ferramentas *Bosaris* [3], tendo em conta o ficheiro *ground truth* do mesmo conjunto e constantes sugeridas pela organização do *MediaEval*

para esta calibração. Por motivos de comparação, é calculado um valor mínimo do Cnxe através de uma transformação Pool-Adjacent Violators (PAV). Esta transformação é uma abordagem mais rigorosa que a anterior, que não necessita de parâmetros e que conduz sempre a valores mais baixos desta métrica.

A decisão final para o caso de uma determinada *query* se encontrar num determinado áudio, é definida por um limiar que é calculado através do valor máximo da métrica secundária ATWV no conjunto dos ficheiros de desenvolvimento, usando custos de falsos alarmes e de falhas e de alvos anteriores.

Os resultados obtidos por este sistema inicial para o conjunto de ficheiros de desenvolvimento foram de 0.8368 para a métrica Cnxe e de 0.1712 para a métrica ATWV. O resultado ideal seria um valor aproximadamente nulo para a métrica Cnxe e um valor perto da unidade para a métrica ATWV. Constata-se que os resultados obtidos para este sistema estão longe de serem promissores. Numa tentativa de melhorar estes resultados, foi desenvolvida um algoritmo de subtração espectral que se encontra descrito no capítulo seguinte.

Capítulo 5

Tratamento da base de dados de desenvolvimento do MediaEval

5.1 Reverberação

Os sinais de fala que são recebidos por um recetor a uma distância da fonte de fala geralmente contém reverberação, ruído de ambiente e outras interferências. A reverberação é o processo de propagação de um sinal acústico através de vários caminhos (*multi-path*) desde a sua fonte até ao seu recetor [12]. Este sinal acústico recebido geralmente consiste do sinal acústico direto, de reflexões recebidas após um curto espaço de tempo, reverberação rápida, e de reflexões que são recebidas depois destas, reverberação tardia. A reverberação rápida contribui principalmente para a coloração do espectro do sinal, por enquanto que a reverberação tardia modifica temporalmente a forma de onda da envolvente do sinal acrescentando-lhe uma espécie de cauda com um *offset*. Este fenómeno denomina-se por T_60 e é o tempo que o sinal demora a atenuar 60 *dB*, o que corresponde ao mesmo que desaparecer. Em sinais onde existe muita reverberação, este tempo revela-se longo e isso é prejudicial.

A reverberação é um processo geralmente descrito usando modelos determinísticos que dependem de um grande número de parâmetros que são desconhecidos. É de extrema dificuldade realizar uma estimação cega destes parâmetros uma vez que dependem da posição espacial exata da sua fonte e do seu recetor, bem como das suas caraterísticas. As respostas em frequência dos ambientes em que se encontram e dos dispositivos de gravação são fulcrais para a sua deteção.

Uma vez que não existe informação acerca dos dispositivos utilizados para a gravação das

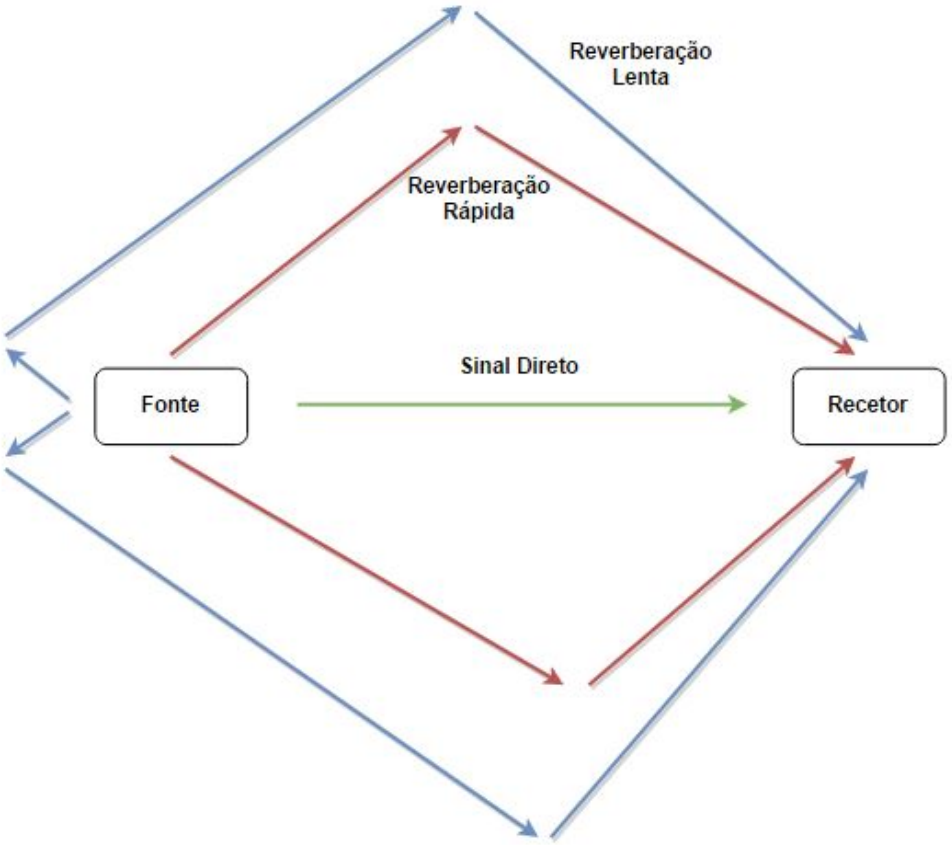

Figura 5.1: Exemplo de Reverberação de um Sinal Acústico.

bases de dados e dos ambientes onde se realizaram as gravações para este desafio, a remoção da reverberação dos ficheiros de fala torna-se num processo bastante complexo e foi deixada para um trabalho futuro.

5.2 Algoritmo de Subtração Espectral

O ruído quando é acusticamente adicionado à fala tende a baixar o desempenho do reconhecimento da fala. O pré-processamento do sinal com vista a do ruído antes da utilização deste tipo de aplicações permite contrariar o abaixamento de desempenho. Com esse objetivo foi desenvolvido uma técnica de subtração espectral com base na estimação de ruído [1].

Esta técnica consiste em estimar o espectro do sinal de fala limpo através da identificação do ruído com base nos níveis de energia do sinal e consequente subtração do espectro deste ruído ao espectro do sinal original. Uma versão resumida desta técnica pode ser observada no algoritmo 1.

Corte de Componentes com Baixas Frequências

Anteriormente à estimação de ruído de cada sinal, revelou-se uma mais valia realizar o corte de componentes com frequências abaixo dos 50 *Hz* através da aplicação de um filtro Passa-Alto de do tipo *Butterworth*. A resposta em frequência deste filtro, figura 5.2, é caraterizada por ser plana nas regiões de banda passante, i. e., por não conter *ripple* ou ondulações, e aproximadamente nula nas regiões de banda rejeitada.

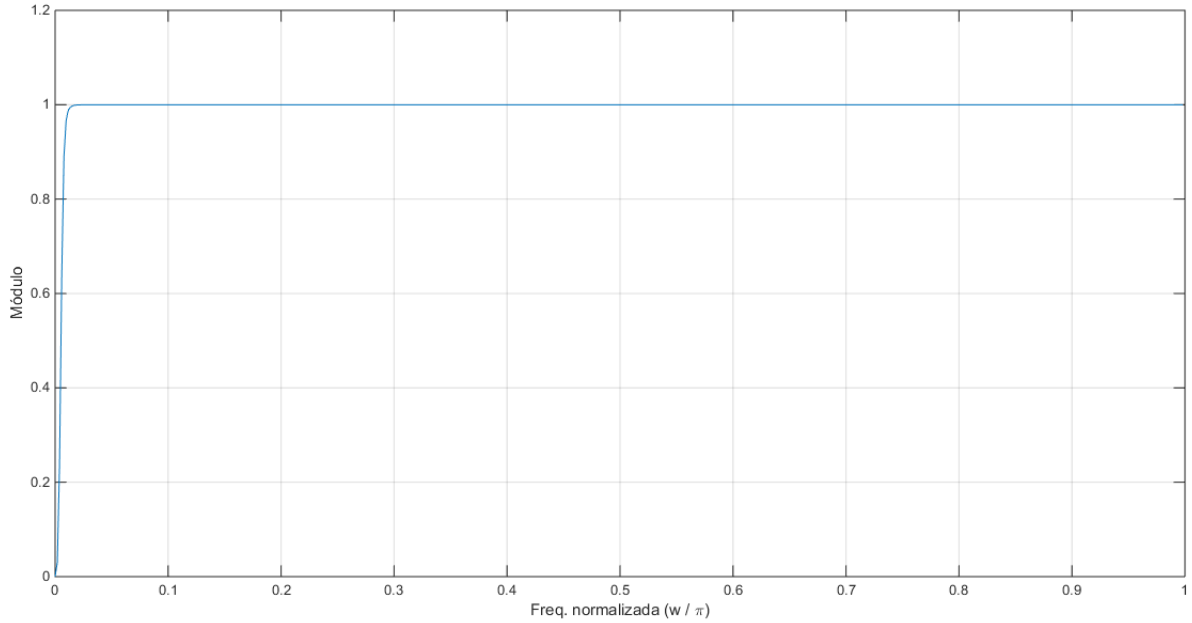

Figura 5.2: Resposta em Frequência do Filtro Passa-Alto do tipo *Butterworth* considerado com frequência de corte em 50 *Hz* e com frequência de amostragem de 8 *kHz*.

Cálculo dos Níveis de Energia dos Sinais

Posteriormente ao corte das componentes de baixas frequências, procedeu-se parcialmente ao processamento de termo curto dos sinais de áudio como descrito em 1.2. Como resultado foram obtidas as tramas com sobreposição, com comprimento de $25ms$ e deslocamento de $10ms$, que foram utilizadas como base para o cálculo dos níveis de energia do sinal.

Perante os obstáculos encontrados, nomeadamente as quantidades absurdas de ruído e reverberação presentes nos ficheiros de áudio, a melhor abordagem para o cálculo da SNR dos sinais foi através do cálculo de limiares de energia baseados nos quartis e medianas destes quartis da energia do sinal. Empiricamente verificou-se que o tempo de reverberação descrito em 5.1 iria originar valores de energia decrescentes até aproximadamente -120dB. Após a observação de al-

guns destes casos e com o intuito de contrariar este acontecimento, descartar todos os valores da energia abaixo dos -60dB revelou-se uma boa abordagem para este problema. A figura 5.3 apresenta um exemplo sobre este passo. Com esta suavização dos níveis de energia, procede-se o cálculo do 1º e 3º Quantis dos "novos" valores de energia. Por sua vez, o cálculo destes quantis vai permitir obter duas medianas: a mediana dos valores de energia menores que o valor do 1º quantil (M1) e a mediana dos valores de energia iguais ou maiores que o valor do 3º quantil (M3). Uma vez que nem todos os ficheiros de áudio vão necessitar da realização da subtração espectral e que os níveis de energia variam de sinal para sinal, estas medianas permitem a criação de um limiar de decisão dependendo do sinal em análise. Este limiar foi encontrado experimentalmente através da aplicação de uma simples equação linear às duas medianas obtidas, onde apenas terá lugar este acontecimento caso a diferença entre a mediana M3 e a mediana M1 seja menor que 25dB:

$$M3 - 25 < M1 \quad , \tag{5.1}$$

onde $(M3 - 25)$ é o Valor de Decisão e $M1$ é o Limiar de Decisão. Um exemplo deste procedimento pode ser observado na figura 5.4.

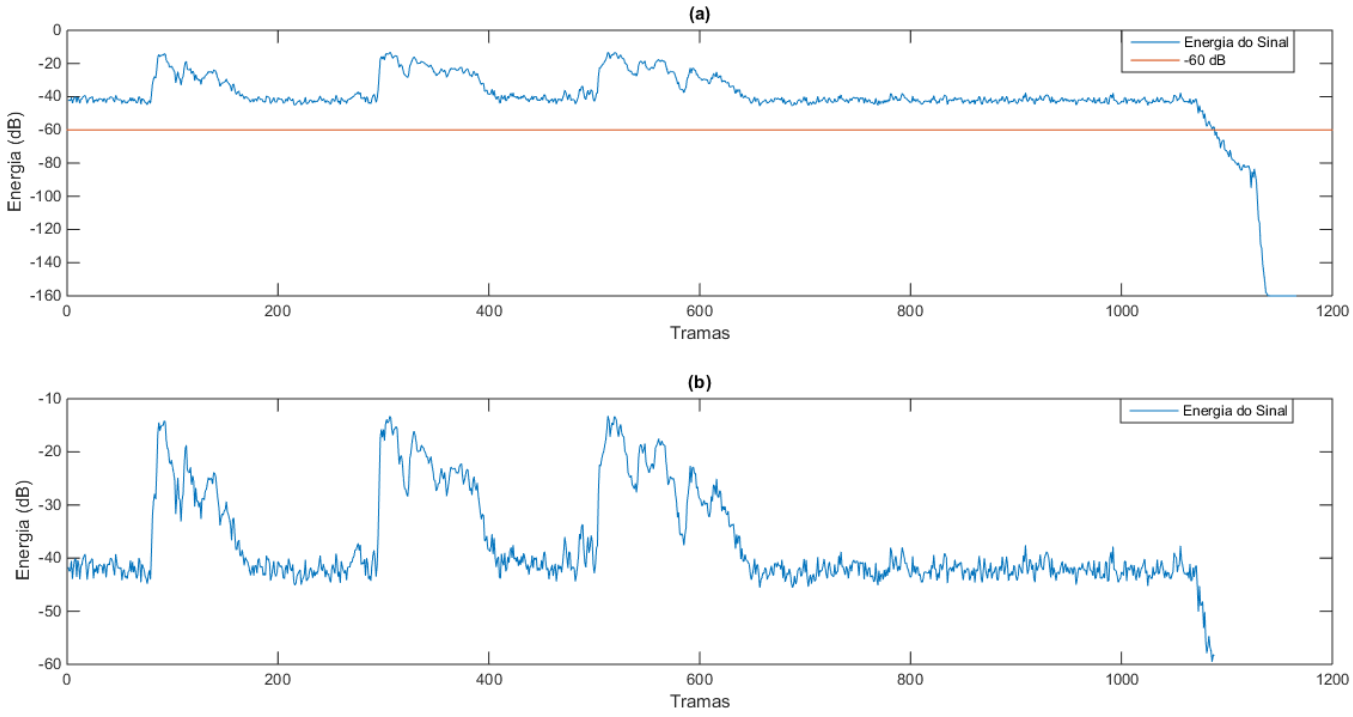

Figura 5.3: Exemplo das Tramas de Energia de um Sinal consideradas para Subtração Espectral. (a) Energia original de um sinal de exemplo. (b) Energia do sinal após o corte de tramas com energia abaixo de -60dB.

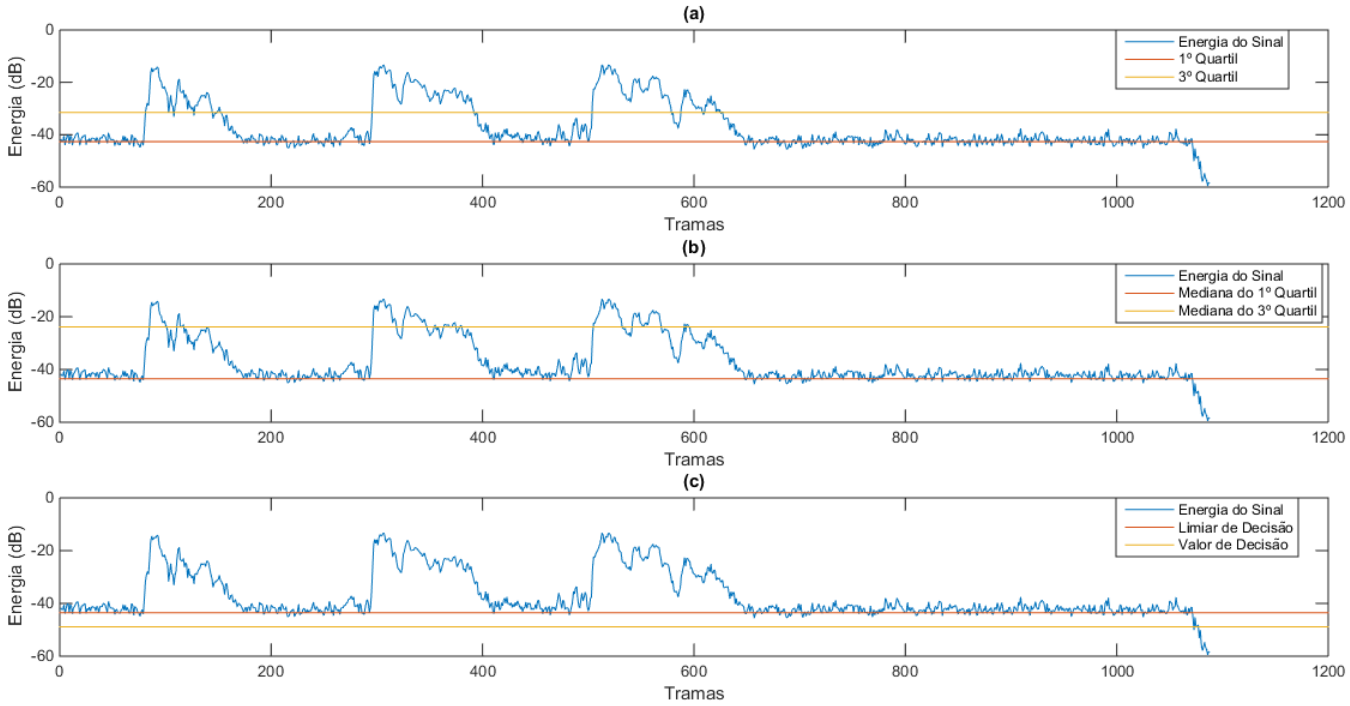

Figura 5.4: Exemplo dos Quantis de Energia e das Medianas dos Quantis de Energia de um sinal considerados para a decisão da realização de Subtração Espectral. (a) 1º e 3º Quantis da Energia do Sinal. (b) Medianas M1 e M3 da Energia do Sinal. (c) Limiar de Decisão (M1) e Valor de Decisão para a realização da Subtração Espectral.

Estimação das Tramas de Ruído

Tal como em [1], vamos considerar que o ruído de fundo quando é adicionado ao sinal de fala vai permanecer estacionário ao longo do espectro do sinal. É então possível encontrar segmentos ao longo do espectro do sinal onde não existe fala e onde se pode estimar este ruído de fundo. Uma vez que não existe qualquer informação acerca da localização da fala nos ficheiros de áudio, desenvolver um detetor de fala que se seja compatível com os mais diversos tipos de áudios presentes nas bases de dados vai permitir a estimação deste ruído.

As medianas calculadas em 5.2, vão ser utilizadas como referência para o cálculo de um limiar de ruído dos ficheiros de áudio. Experimentalmente verificou-se que uma solução genérica e rudimentar para este cálculo foi:

$$Limiar\ de\ Ruído = M1 + 0.2 \times (M3 - M1) \quad , \tag{5.2}$$

onde M1 é a mediana dos valores de energia menores que o valor do 1º quantil e M3 é a mediana dos valores de energia iguais ou maiores que o valor do 3º quantil. Assume-se que todas

as tramas com valores de energia abaixo deste limiar são consideradas como tramas de ruído, figura 5.5. Pode assumir-se que este limiar vai funcionar de forma semelhante a um detetor de fala, uma vez que de uma forma geral a energia do sinal num segmento de fala é maior do que a energia num semento de ruído. Para garantir que tramas durante a pronunciação de palavras não sejam consideradas como ruído, considerou-se que os segmentos têm de ter uma duração superior a $100ms$. Caso não seja possível a estimação das tramas de ruído, não se realiza a subtração espectral.

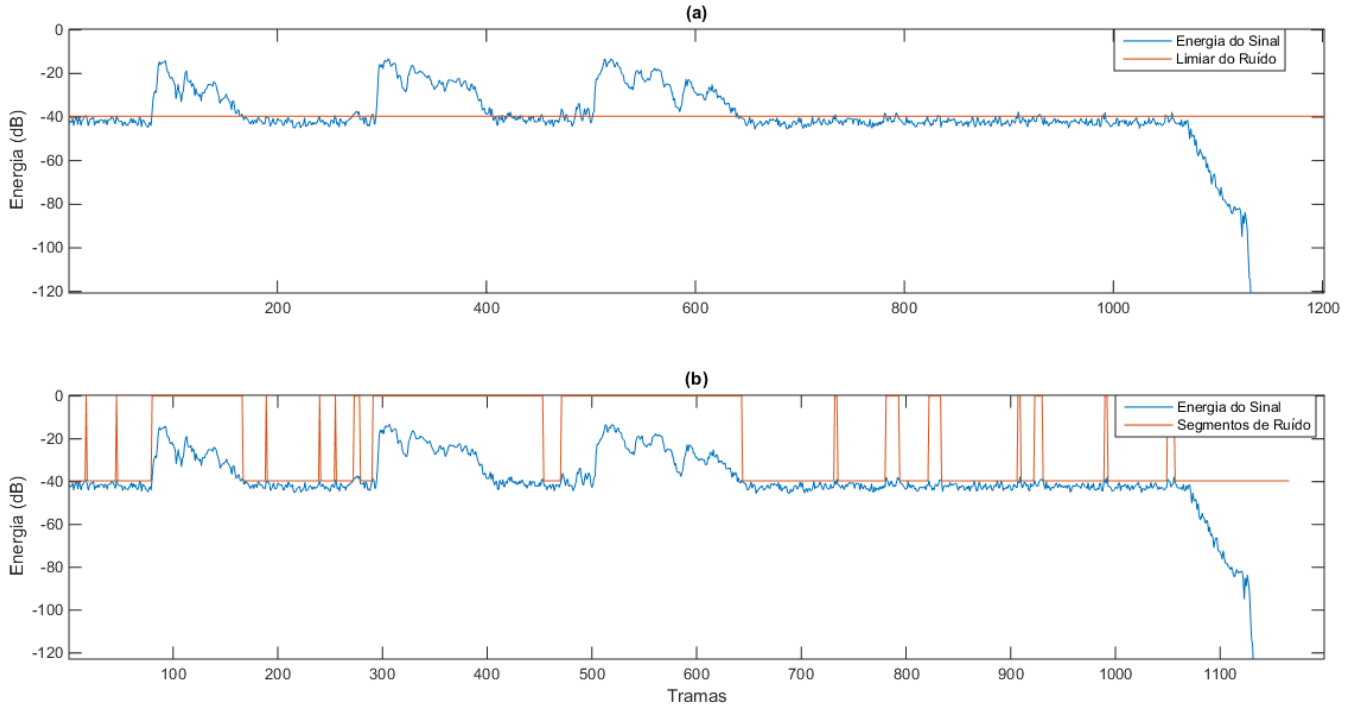

Figura 5.5: Exemplo do Limiar de Ruído e de Segmentos de Ruído considerados de um sinal de exemplo. (a) Limiar de Ruído considerado para o sinal de exemplo. (b) Segmentos de Ruído considerados de acordo com o limiar de ruído e a duração dos mesmos.

Subtração Espectral e Filtragem de *Wiener*

Considerando que se está perante um modelo com ruído aditivo, a passagem do domínio do tempo para o domínio da frequência através de uma DFT permite a simples realização de uma subtração deste ruído estimado em todo o sinal. Como não é possível estimar o ruído em cada trama do sinal, a potência do ruído estimado é calculada através de uma média de todas as

tramas que foram consideradas como ruído:

$$P_N(k) = \sqrt{\frac{\sum_{i=1}^{n} X_N^2(i,k)}{n}} \quad , \tag{5.3}$$

onde P_N é a potência média do ruído estimado, X_N é o módulo da DFT de uma trama do sinal que foi considerada como ruído, n é o número total de tramas que foram consideradas como ruído e k é o índice da DFT tomada. Após a subtração da potência média do ruído em cada trama do sinal, pode acontecer que se obtenham valores de potência negativos. Estes têm de ser pelo menos anulados. A técnica da retificação de meia-onda apresenta a desvantagem de que caso a potência média do ruído seja maior que a soma da potência do ruído e da fala numa certa trama do sinal, a informação da fala vai ser removida incorretamente numa dada frequência causando o chamado ruído musical ou até mesmo a sua ininteligibilidade. De forma a evitar esta situação, para estes casos considera-se que o valor da potência da trama em questão será uma percentagem da potência original da trama, i. e., pretende-se obter um valor máximo entre a subtração espectral e uma percentagem do sinal:

$$P_{S_{estimado}} = max\left[P_X - P_{N_{estimado}}, \alpha \times P_X\right] \quad , \tag{5.4}$$

onde $P_{S_{estimado}}$ é a potência de cada trama do sinal sem ruído estimado, P_X é a potência de cada trama do sinal original, $P_{N_{estimado}}$ é a potência média do ruído estimado e α é uma percentagem. Experimentalmente verificou-se que o valor mais compatível com as bases de dados era uma percentagem de 5% da potência original da respetiva trama do sinal. Através do sinal estimado e do ruído estimado calcula-se o filtro de *Wiener* da seguinte forma:

$$H = \frac{P_{S_{estimado}}}{P_{X_{estimado}}} = \frac{P_{S_{estimado}}}{P_{S_{estimado}} + P_{N_{estimado}}} \quad , \tag{5.5}$$

onde H é o filtro de *Wiener* e $P_{X_{estimado}}$ é a potência de cada trama do sinal estimado. Por último aplica-se este filtro ao sinal original de forma a se obter uma versão menos ruidosa do mesmo.

$$P_Y = H \times P_X \quad , \tag{5.6}$$

onde P_Y é a potência de cada trama do sinal filtrado e P_X a potência de cada trama do

sinal original. A partir deste ponto seria apenas necessário converter o sinal para uma escala de melodias e aplicar posteriormente uma DCT para se obter um espaço de caraterísticas da fala com uma dimensão reduzida. Porém, este passo não foi realizado uma vez que obter uma opinião humana acerca dos resultados da aplicação deste algoritmo nas bases de dados foi considerado uma mais-valia. Procede-se então a reconstrução do sinal, adicionando ao módulo a fase do sinal original e fazendo a síntese do sinal com o método de sobreposição e soma. Ao sinal de áudio obtido é posteriormente aplicado o sistema de reconhecimento de fonemas para a realização de testes com o sistema final.

Algoritmo 1 Subtração Espectral Baseado em Níveis de Energia

Entrada: Matriz de Áudio

Saída: Matriz de Áudio Reconstruído após Subtração Espectral

Por cada áudio:

1: Corte das componentes com frequências abaixo de $50Hz$

2: Cálculo dos Níveis de Energia do Sinal

- Descartar valores de energia do sinal abaixo dos -$60dB$
- Cálculo dos 1º e 3º quantis da energia do sinal
- Cálculo das medianas para valores abaixo do 1º quantil (M1) e para os valores iguais ou maiores que o 3º quantil (M3)
- Verificar elegibilidade do sinal de áudio para a subtração espectral (eq. 5.1)

3: Caso o sinal de áudio seja elegível

- Cálculo do limiar de ruído com base nas medianas (eq. 5.2)

4: Cálculo da potência média das tramas de ruído (eq. 5.3)

5: Por cada trama do sinal

- Subtração Espectral da trama de ruído estimado (eq. 5.4)
- Filtragem de Wiener(eq. 5.5)

6: Reconstrução do sinal de áudio

Capítulo 6

Teste do Sistema Final

Este capítulo descreve o sistema desenvolvido pelo Laboratório de Processamento de Sinal do Instituto de Telecomunicações do pólo de Coimbra (SPL-IT-UC) [24], no âmbito do desafio de QUESST para o *MediaEval* 2015 [21].

Melhoramento da SNR das locuções

Previamente à aplicação do sistema de reconhecimento de fonemas foi implementado o algoritmo de subtração espectral descrito em 5.2, com o objetivo de reduzir as grandes quantidades de ruído presentes em todos os conjuntos de ficheiros disponibilizados pela organização do desafio QUESST 2015.

Posteriorgramas de fonemas com 3 estados

Os posteriorgramas de fonemas com 3 estados para os conjuntos de ficheiros foram obtidos de forma semelhante ao sistema inicial, considerando agora a adição de dois novos sistemas desenvolvidos no âmbito deste desafio. Utilizaram-se então 5 sistemas disponíveis para áudio de $8kHz$ nas seguintes línguas: Checo, Húngaro, Russo, Português Europeu e Inglês, onde cada língua contém o seu respetivo conjunto de fonemas.

Corte de Silêncios e Ruídos dos Posteriorgramas

De forma semelhante, o próximo passo passa pelo corte de todos os silêncios e ruídos dos posteriorgramas obtidos para as *queries* dos conjuntos de desenvolvimento e de avaliação. Da mesma

forma, para cada trama considerou-se a soma das probabilidades dos 3 estados de cada fonema daqueles que correspondem silêncio ou ruído. Após esta soma é calculada uma média aritmética de cada um destes fonemas considerando agora as 5 línguas. Caso esta média seja superior a 50%, procede-se à exclusão das respetiva trama do posteriorgrama em questão. Este processo permite o cálculo de uma matriz de distâncias mais "limpa" para a aplicação da DTW.

Aplicação da DTW

Após o corte de silêncios e ruídos, procede-se o cálculo da matriz de distâncias para os posteriorgramas das 5 línguas, como descrito em 3.1. Em adição a estas, foi considerada uma nova "língua"denominada por Multi-Língua (ML) cuja matriz de distâncias consiste numa média aritmética das matrizes de distâncias obtidas para as 5 línguas. Esta nova abordagem originou uma melhoria significativa na fusão de resultados.

Considerou-se também previamente ao cálculo das matrizes de distâncias, remover por completo as probabilidades dos fonemas de silêncio e ruído dos posteriorgramas e realizar uma normalização das restantes probabilidades de fonemas de fala de modo a que a soma de todas as probabilidades seja unitária. Contudo, este processo não ajudou e esta ideia foi então colocada de parte.

O passo seguinte foi a aplicação das 6 estratégias de alinhamento temporal dinâmico descritas em 3.2 e 3.3. Como resultado obtêm-se os valores das distâncias para cada par *query*-áudio para as 6 estratégias de DTW das 6 línguas.

Fusão e Calibração de Resultados

As primeiras modificações para a fusão de resultados foram ao nível das distribuições de distâncias dos pares *query*-áudio por estratégia. Verificou-se que ao realizar a atribuição de um valor máximo de distância para os casos onde não seja possível aplicar nenhuma estratégia de DTW, isto é, para os casos que não satisfazem os pré-requisitos destas estratégias (como por exemplo o caso de a *query* em questão ser maior que o áudio de procura), uma truncagem considerável deste valor de distância apresentava melhores resultados. Aliás, verificou-se que uma truncagem de todos os valores de distâncias para um valor máximo perto da média das distribuições de distâncias apresentava os melhores resultados. Como referido em [24], pensa-se que isto seja devido ao facto de existirem muitos casos onde se obtiveram grandes valores de distâncias para o *match* do par

query-áudio onde se deveria ter obtido um baixo valor, isto é, casos de falsos negativos.

De forma semelhante ao sistema inicial descrito no capítulo 4, realiza-se uma normalização por *query* através da subtração da nova média aritmética das distâncias e posterior divisão do novo desvio padrão das mesmas.

Através do seu valor simétrico, são obtidas as suas figuras de mérito necessárias para a fusão dos resultados. Isto deve-se ao facto de que para a tomada de decisão do detetor ser utilizada uma função indicadora (discriminador ou figura de mérito) que, por convenção, indica que quanto maior for o valor da distância, mais certeza existe na decisão de o aceitar. Esta decisão é realizada através da comparação deste indicador com um limiar, onde é considerada aceite caso o indicador seja superior ao limiar. No presente caso, a utilização do valor simétrico das distâncias vai significar que para valores muito negativos a *query* não vai estar presente, e que para valores menos negativos e em torno de zero, esta vai lá estar presente com uma grande certeza.

Foram considerados dois sistemas para a fusão dos resultados das diferentes estratégias DTW e línguas. O sistema primário consiste na fusão de todas as estratégias de todas as línguas, ou seja, uma vez que a solução de cada estratégia é representada através de um vetor e que cada língua contém 6 estratégias, este sistema consiste na fusão de 36 vetores (6 estratégias $\times$ 6 línguas). O sistema secundário consiste na fusão da média harmónica das 6 estratégias de cada língua, ou seja, um vetor por cada língua. Este sistema foi considerado para apenas realizar a avaliação dos resultados de acordo com as línguas, que de certa forma contraria a hipótese de a fusão dos resultados do sistema primário ser demasiado ajustada ao conjunto de resultados considerados neste.

Para a fusão dos resultados utilizou-se novamente o conjunto de ferramentas *Bosaris* [3]. Este conjunto contém uma ferramenta que fornece a solução de uma regressão logística, através de uma fusão paramétrica e posterior calibração. Esta ferramenta tem a capacidade de treinar um conjunto de parâmetros de forma a fundir múltiplos subsistemas num só subsistema, o qual fornece uma solução de relações quantitativas de logaritmos de verosimilhanças. Permite também a adição de informação paralela aos subsistemas que pode ser revelante durante a sua fusão. Esta ferramenta requer uma base de dados onde seja possível realizar a calibração, que neste caso é o *ground truth* do conjunto de desenvolvimento.

Tomando como exemplo [35], verificou-se que a adição de 7 vetores de informação paralela

melhoravam os resultados da fusão. Foram consideradas os seguintes vetores de informações: tamanho da *query* em tramas, o logaritmo do tamanho da *query*, o valor original da SNR da *query*, o valor da SNR da *query* após a substração espectral, o valor original da SNR do áudio, o valor da SNR do áudio após a substração espectral e o valor médio das distâncias por *query* antes da aplicação da truncagem e da normalização descritas anteriormente.

No total foram submetidos 4 sistemas para avaliação para o desafio QUESST 2015. Foram considerados os dois sistemas descritos anteriormente com e sem a adição da informação paralela.

Os resultados obtidos encontram-se dispostos sob a forma de tabelas no apêndice C, onde o resultado ideal seria um valor aproximadamente nulo para a métrica Cnxe e um valor perto da unidade para a métrica ATWV. Analisando os resultados em relação ao conjunto de desenvolvimento, tabelas C.1 e C.2, como esperado o sistema primário com informação paralela obteve os melhores resultados com um Cnxe de 0.7782 e um ATWV de 0.2341. Quando realizada uma comparação entre sistemas inicial e final, obteve-se uma melhoria nos resultados de 0.8368 para 0.7782 na métrica Cnxe e de 0.1712 para 0.2341 na métrica ATWV. Perante as dificuldades encontradas neste desafio, considera-se que foi um resultado muito bom, aliás, o segundo melhor resultado do desafio, como se pode observar na tabela D.1. A figura 6.1 permite analisar as melhorias através de uma comparação do desempenho dos sistemas Inicial e Finais, em termos de curvas do tipo Detection Error Tradeoff (DET). Estas curvas representam as taxas de erro da classificação binária de sistemas através da comparação das taxas de casos de falsos positivos e de falsos negativos.

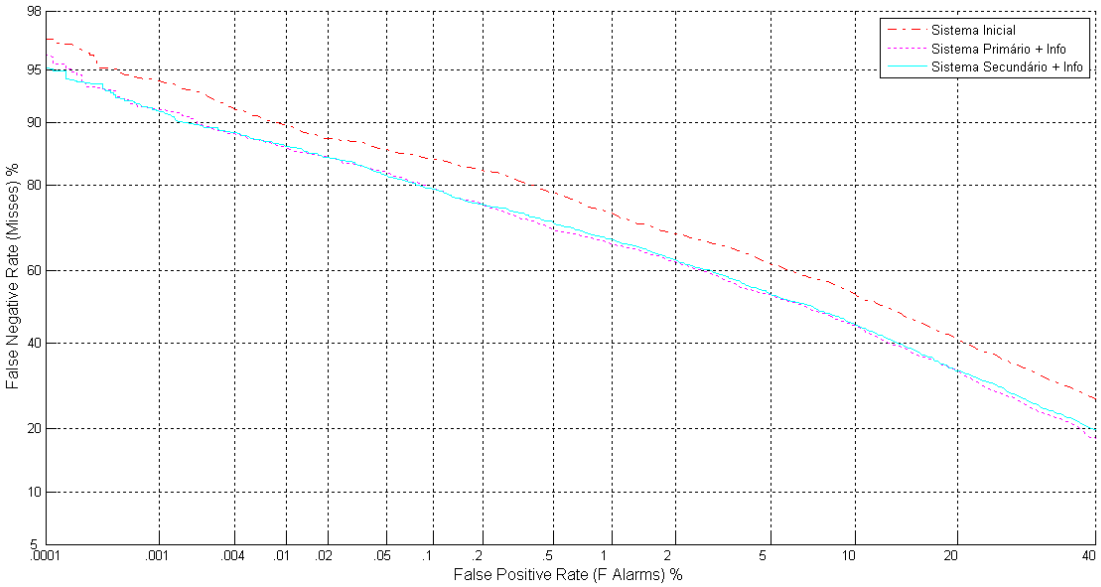

Figura 6.1: Curvas DET para os sistemas Inical (vermelho), Primário com Informação Paralela (magenta) e Secundário com Informação Paralela (azul ciano).

Em relação ao conjunto de avaliação, tabelas C.3 e C.4, o sistema com a fusão da média harmónica das estratégias e com adição da informação paralela revelou-se como o melhor resultado, com um Cnxe de 0.7842 e um ATWV de 0.2017. Embora que por pouco, este resultado confirmou as suspeitas da possibilidade de o sistema primário estar demasiado adaptado ao conjunto de desenvolvimento.

Por curiosidade, foi também realizada uma análise ao desempenho individual das estratégias desenvolvidas, descritas em 3.2 e 3.3, onde se analisou os resultados para a métrica principal Cnxe apenas no conjunto de desenvolvimento. A tabela 6.1 apresenta os resultados desta análise.

Tabela 6.1: Resultados obtidos para a métrica principal Cnxe das diferentes estratégias DTW para o conjunto de desenvolvimento.

Estratégia	1	2	3	4	5	6
Cnxe	0.8041	0.7978	0.8335	0.8137	0.8184	0.8460

Verifica-se que a estratégia que obteve um melhor desempenho global foi a estratégia 2, que permite a ocorrência de uma variação lexical no final da *query*. A nova estratégia apresentou um desempenho aquém das espetativas e necessita de uma revisão. Pensa-se que possa ser devido aos casos em que o conteúdo irrelevante entre as palavras da *query* é uma extensão de um fonema

existente, o que dá origem a um caminho sem saltos e pode induzir o algoritmo em erro.

De uma forma geral, estes sistemas permitiram obter os segundos melhores resultados do desafio de entre um grupo de 10 equipas.

Resultado para QUESST 2014

De modo a ter uma avaliação extra acerca do desempenho dos novos sistemas desenvolvidos, resolveu-se proceder à aplicação destas novas técnicas nas bases de dados facultadas pelo desafio QUESST 2014 [19].

Verificou-se que o algortimo de subtração espectral não produziu melhorias, pelo contrário, piorou os resultados uma vez que as bases de dados disponibilizados para este desafio continham locuções com elevadas SNR. Constatou-se também que não seria benéfico a utilização da nova estratégia desenvolvida para combater os casos onde a *query* poderia ter alguma informação irrelevante entre as palavras, uma vez que o desafio em questão não considerava as *queries* do tipo 3 descritas em 1.5.

Excluindo estes dois passos, o processo considerado foi semelhante. Passou pela criação dos posteriorgramas para todas as línguas e de seguida pelo corte dos silêncios e ruídos dos mesmos. Foi agora apenas considerada a aplicação das 5 estratégias iniciais a cada par *query*-áudio. A fusão e calibração dos resultados realizou-se de forma igual, excluíndo apenas nos casos onde se utiliza a informação paralela, as informações acerca das SNR das locuções.

Analisando os resultados da tabela 6.2, verifica-se uma melhoria muito significante quando realizada a comparação dos sistemas da equipa SPL-IC-UT de 2014 [25] e 2015. Aliás, o sistema primário supera os melhores resultados globais obtidos para o desafio QUESST 2014 [19] pela equipa BUT [35].

Estima-se que ainda fosse possível obter um melhor resultado através do sistema primário com informação paralela, mas não foi possível realizar a calibração dos resultados para este sistema.

A comparação do desempenho dos sistemas de 2014 e de 2015 em termos de curvas do tipo DET pode ser observada na figura 6.2.

Este teste comprova que realmente o sistema desenvolvida nesta dissertação tem um elevado desempenho. Também permite a avaliação das dificuldades encontradas no desafio QUESST 2015 [21] em relação ao desafio do ano anterior.

Tabela 6.2: Comparação de resultados de sistemas MediaEval de 2014 e de 2015.

Equipa	Sistema	Conjunto Desenvolvimento		Conjunto Avaliação	
		ATWV	CNXE	ATWV	CNXE
BUT 2014	Sem Informação Paralela	0.4976	0.4949	0.4966	0.4735
	Com Informação Paralela	0.4729	**0.4667**	0.4729	0.4732
SPL-IT-UC 2014	Sem Informação Paralela	0.4608	0.5615	0.4538	0.5153
SPL-IT-UC 2015	Primário	**0.5134**	0.5171	**0.5066**	**0.4646**
	Secundário	0.4989	0.5313	0.4964	0.4785
	Secundário + Informação Paralela	0.4817	0.4828	0.4801	0.4695

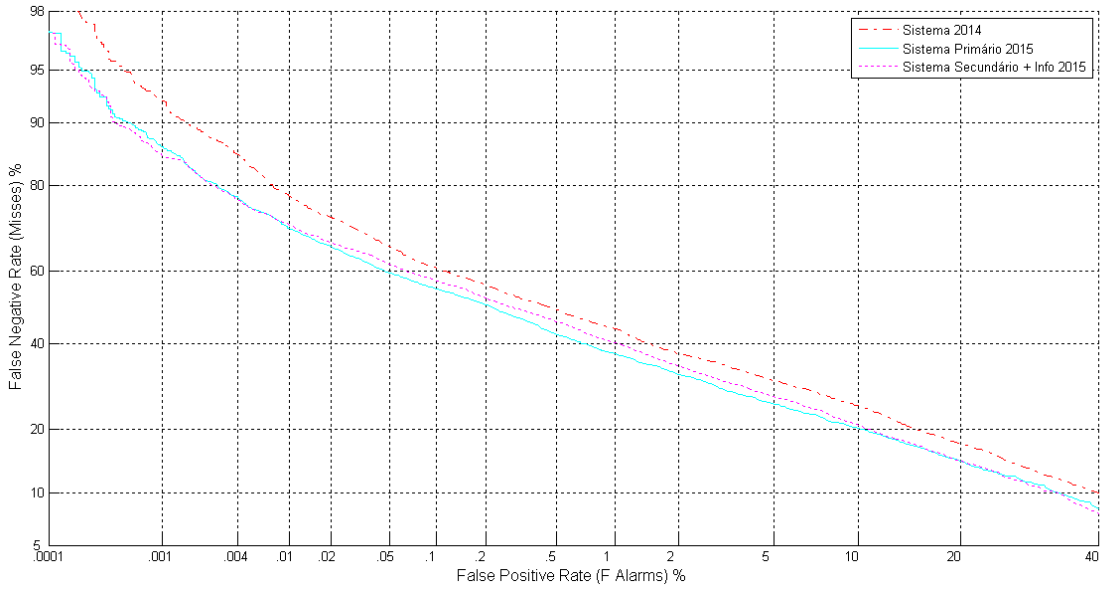

Figura 6.2: Curvas DET para os sistemas de 2014 (vermelho), Primário 2015 (magenta) e Secundário 2015 com Informação Paralela (azul ciano).

Capítulo 7

Conclusão

Um dos objetivos principais desta dissertação era desenvolver um sistema automático que fosse utilizado em tempo real para a deteção de áudio em áudio, independentemente do tipo de língua, e esse objetivo foi cumprido. O sistema desenvolvido garante um desempenho bastante satisfatório.

Outro dos principais objetivos consistia em realizar o treino de uma rede neuronal artificial Artificial Neural Network (ANN) para Português Europeu, de modo a ser utilizada pelo mesmo sistema para o reconhecimento de fonemas para a língua portuguesa. Este objetivo foi igualmente cumprido e em adição a este foi também treinada uma ANN para o reconhecimento de fonemas na língua inglesa. Estes processos de treino de redes neuronais artificiais revelaram-se muito úteis na elaboração de outros projetos no âmbito do laboratório, na área do reconhecimento de fala.

Como futuro trabalho propõe-se a utilização de uma Deep Neural Network (DNN), no lugar de uma típica ANN, para o reconhecimento de fonemas [23], no aumento da qualidade e do número de locuções das bases de dados utilizadas para o treino das ANN e na melhoria de algumas das abordagens consideradas para o desenvolvimento das estratégias baseadas em alinhamento temporal dinâmico.

Concluindo, o trabalho desenvolvido ao longo desta dissertação produziu resultados muito interessantes, onde todos os objetivos foram atingidos. Permitiu ainda a criação de um método de treino de uma ANN para realizar o reconhecimento de fonemas em qualquer tipo de língua, desde que esta seja treinada com a respetiva base de dados.

Em termos pessoais foi também um trabalho muito gratificante numa área tão interessante como o reconhecimento de fala.

Bibliografia

[1] Boll, S.: *Suppression of Acoustic Noise in Speech Using Spectral Subtraction.* IEEE Trans. Acoust. Speech, Signal Processing, páginas 113–120, 1979.

[2] Boulard, H. e N.Morgan: *Connectionist speech recognition.* Em *A Hybrid Approach.* Academic Publishers, Boston, USA, 1994.

[3] Brummer, N. e E. de Villiers: *The BOSARIS Toolkit User Guide: Theory, Algorithms and Code for Binary Clas-sifer Score Processing.* `https://sites.google.com/site/bosaristoolkit/`, 2011.

[4] Catalog.ldc.upenn.edu: *Resource Management RM1 2.0 - Linguistic Data Consortium.* [Online] `https://catalog.ldc.upenn.edu/LDC93S3B`.

[5] Catalog.ldc.upenn.edu: *TIMIT Acoustic-Phonetic Continuous Speech Corpus - Linguistic Data Consortium.* [Online] `https://catalog.ldc.upenn.edu/LDC93S1`, 1993.

[6] Dataminingtheworld.blogspot.pt: *Neural network classification of countries in the OECD.* [Online] `http://dataminingtheworld.blogspot.pt/`, 2015.

[7] Davis, S. B. e P. Mermelstein: *Comparison of parametric representation for monosyllabic word recognition in continuously spoken sentences.* IEEE Transactions on Acoustic, Speech and Signal Processing, 28(4):357–366, 1980.

[8] Fee.vutbr.cz: *Czech SpeechDat-E.* [Online] `http://www.fee.vutbr.cz/SPEECHDAT-E/sample/czech.html`, 2001.

[9] Fee.vutbr.cz: *Hungarian SpeechDat-E.* [Online] `http://www.fee.vutbr.cz/SPEECHDAT-E/sample/hungarian.html`, 2001.

[10] Fee.vutbr.cz: *Russian SpeechDat-E.* [Online] `http://www.fee.vutbr.cz/SPEECHDAT-E/sample/russian.html`, 2001.

[11] Guwahati website, Indian Institute of Tecnology of: *Short Term Time Domain Processing Speech.* [Online] `http://iitg.vlab.co.in/?sub=59&brch=164&sim=857&cnt=1`, 2011.

[12] Habets, E.: *Speech dereverberation using statistical reverberation models.* Springer, 2010.

[13] Haykin, S.: *Neural Networks: A Comprehensive Foundation.* Macmillan College Publishing Company, 1994.

[14] Hazen, T. J., W. Shen e C. White: *Query-By-Example Spoken Term Detection Using Phonetic Posteriorgram Templates.* IEEE Automatic Speech Recognition & Understanding, 2009.

[15] Htk.eng.cam.ac.uk: *Hidden Markov Model ToolKit (HTK).* [Online] `http://htk.eng.cam.ac.uk/`.

[16] Icsi.berkeley.edu: *QuickNet.* [Online] `http://www.icsi.berkeley.edu/Speech/qn.html`.

[17] Lee, K. e H. Hon: *Speaker-independent phone recognition using hidden markov models.* IEEE Transactions on Acoustic, Speech and Signal Processing, 37(11):1641–1648, Nov. 1989.

[18] Müller, M.: *Information Retrieval for Music and Motion.*, páginas 69–74. Springer, 2007.

[19] Multimediaeval.org: *Query by Example Search on Speech Task (QUESST) 2014.* [Online] `http://www.multimediaeval.org/mediaeval2014/quesst2014/`, 2014.

[20] Multimediaeval.org: *MediaEval.* [Online] `http://www.multimediaeval.org/about/`, 2015.

[21] Multimediaeval.org: *Query by Example Search on Speech Task (QUESST) 2015.* [Online] `http://www.multimediaeval.org/mediaeval2015/quesst2015/index.html`, 2015.

[22] NIST: *The Spoken Term Detection (STD) 2006 Evaluation Plan.* [Online] `http://www.itl.nist.gov/iad/mig/tests/std/2006/`, 2006.

[23] Pan, J., C. Liu, Zh. Wang, Y. Hu e H. Jiang: *Investigation of deep neural networks (DNN) for large vocabulary continuous speech recognition: Why DNN surpasses GMMs in acoustic modeling.* 8th International Symposium on Chinese Spoken Language Processing (ISCSLP), páginas 301–305, 2012.

[24] Proença, J., L. Castela e F. Perdigão: *The SPL-IT Query by Example Search on Speech system for MediaEval 2015.* Mediaeval 2015 Workshop, Wurzen, Germany, Setembro 2015.

[25] Proença, J., A. Veiga e F. Perdigão: *Query by Example Search with Segmented Dynamic Time Warping for Non-Exact Spoken Queries*. European Signal Processing Conf. - EUSIPCO, Nice, France, 2015.

[26] Proença, J., A. Veiga e F. Perdigão: *The SPL-IT Query by Example Search on Speech system for MediaEval 2014*. Mediaeval 2015 Workshop, Wurzen, Germany, 2015.

[27] Pt.wikipedia.org: *Sampa*. [Online] `https://pt.wikipedia.org/wiki/SAMPA`, 2015.

[28] Rabiner, L. e B.H. Juang: *An introduction to hidden Markov models*. ASSP Magazine, IEEE, páginas 4–16, 1986.

[29] Rabiner, L. R. e B. H. Juang: *Fundamentals of speech recognition*. Prentice Hall Signal Processing Series, 1993.

[30] Saracoglu, Ö. G. e H. Altural: *Color Regeneration from Reflective Color Sensor Using an Artificial Intelligent Technique*. Sensors (Basel), 10(9):8363–8374, 2010.

[31] Schwarz, P.: *Phoneme Recongnition Based on Long Temporal Context*. Tese de Doutoramento, Brno University of Technology, 2009.

[32] Speech.fit.vutbr.cz: *Phoneme recognizer based on long temporal context*. [Online] `http://speech.fit.vutbr.cz/software/phoneme-recognizer-based-long-temporal-context`.

[33] Speech.fit.vutbr.cz: *BUT Speech@FIT*. [Online] `http://speech.fit.vutbr.cz/`, 2015.

[34] Szöke, I., P. Schwarz, L. Burget, M. Fapso, M. Karafiat, J. Cernocky e P. Matejka: *Comparison of Keyword Spotting Approaches for Informal Continuous Speech*. Eurospeech2005, 2005.

[35] Szöke, I., M. Skácel e L. Burget: *BUT QUESST 2014 system description*. Mediaeval 2015 Workshop, Wurzen, Germany, Setembro 2014.

[36] Veiga, A., C. Lopes, L. Sá e F. Perdigão: *Acoustic Similarity Scores for Keyword Spotting*. 11th International Conference, PROPOR 2014, São Carlos/SP, Brazil, 8775:48–58, Outubro 2014.

[37] Wikipedia: *International Phonetic Alphabet*. [Online] `https://en.wikipedia.org/wiki/International_Phonetic_Alphabet`, 2015.

[38] Wikipedia: *Mel Scale*. [Online] https://en.wikipedia.org/wiki/Mel_scale, 2015.

[39] Wikipedia: *Speech*. [Online] https://en.wikipedia.org/wiki/Speech, 2015.

[40] Wikipedia: *Viterbi Algorithm*. [Online] http://en.wikipedia.org/wiki/Viterbi_algorithm, 2015.

Apêndice A

Tabela A.1: Tabela de fonemas considerados para vogais da língua portuguesa.

Vogais				
Tipo	Sampa	SPL-IT-UC	Exemplo	Transcrição Fonética
Vogais abertas	a	a	autor	a w t o r
	E	E	esta	E S t &
	i	i	radia	R & d i k &
	O	O	própria	p r O p r i &
	u	u	uma	u m &
Vogais Fechadas	6	&	lagoa	l & g o &
	e	e	evitar	e v i t a r
	@	@	pretende	p r @ t eN d @
	o	o	outro	o t r u
Semi Vogais	j	j	noite	n o j t @
	w	w	causa	k a w z &
Vogais Nasais	6~	&N	grandes	g r & N d @ S
	e~	eN	entrada	eN t r a d &
	i~	iN	interior	iN t @ r i o r
	o~	oN	contos	k oN t u S
	u~	uN	um	uN
	j~	jN	em	&N jN
	w~	wN	são	s &N wN

Tabela A.2: Tabela de fonemas considerados para consoantes da língua portuguesa.

Consoantes				
Tipo	Sampa	SPL-IT-UC	Exemplo	Transcrição Fonética
Plosivas Surdas	p	p	poeira	p u & j r &
	t	t	forte	f O r t @
	k	k	comitiva	k u m i t i v &
Plosivas Sonoras	b	b	ibérica	i b E r i k&
	d	d	proferidas	p r u f @ r i d & S
	g	g	algumas	a l g u m & S
Fricativas Surdas	f	f	semáforos	s @ m a f u r u S
	s	s	concelhia	k oN s @ L i &
	S	S	buracos	b u r a k u S
Fricativas Sonoras	v	v	viária	v i a r i &
	z	z	meses	m e z @ S
	Z	Z	laranjas	l & r &N Z & S
Nasais	n	n	centena	s eN t e n &
	m	m	américa	& m E r i k &
	J	J	espanha	@ S p a J &
Laterais/Líquidas	l	l	plano	p l & n u
	L	L	trabalho	t r & b a L u
	r	r	praça	p r a s &
	R	R	regional	R @ Z i u n a l

Tabela A.3: Tabela de fonemas considerados para silêncios/ruídos da língua portuguesa.

Silêncios e Ruídos				
Tipo	Sampa	SPL-IT-UC	Exemplo	Transcrição Fonética
Silêncio	sil	sil	silêncios	sil
	sp	sil	respirações	sil
Ruído	-	noi	ruídos	noi

Apêndice B

Tabela B.1: Mapeamento de fonemas de TIMIT considerado para a língua inglesa. Editado de [31].

Número de Fonemas				61	39	39	39
TIMIT	CMU/MIT	BUT	SPL-IT-UC	TIMIT	CMU/MIT	BUT	SPL-IT-UC
p	p	p	p	b	b	b	b
t	t	t	t	d	d	d	d
k	k	k	k	g	g	g	g
pcl	sil	p	p	bcl	sil	b	b
tcl	sil	r	r	dcl	sil	d	d
kcl	sil	k	k	gcl	sil	g	g
dx	dx	dx	dx	q	-	-	-
m	m	m	m	em	m	m	m
n	n	n	n	en	n	n	n
ng	ng	ng	ng	eng	ng	ng	ng
nx	n	n	n	-	-	-	-
s	s	s	s	sh	sh	sh	sh
z	z	z	z	zh	zh	sh	sh
ch	ch	ch	ch	jh	jh	jh	jh
th	th	th	th	dh	dh	dh	dh
f	f	f	f	v	v	v	v
l	l	l	l	el	l	l	l
r	r	r	r	w	w	w	w
y	y	y	y	h#	sil	pau	pau
pau	sil	pau	pau	epi	sil	pau	pau
hh	hh	hh	hh	hv	hh	hh	hh
eh	eh	eh	eh	ih	ih	ih	ih
ao	aa	aa	aa	ae	ae	ae	ae
aa	aa	aa	aa	ah	ah	ah	ah
uw	uw	uw	uw	uh	uh	uh	uh
er	er	er	er	ux	uw	uw	uw
ay	ay	ay	ay	oy	oy	oy	oy
ey	ey	ey	ey	iy	iy	iy	iy
aw	aw	aw	aw	ow	ow	ow	ow
ax	ah	ah	ah	axr	er	er	er
ix	ih	ih	ih	ax-h	ah	ah	ah

Tabela B.2: Mapeamento de fonemas de Resource Management considerado para a língua inglesa.

Número de Fonemas		48	39
RM	SPL-IT-UC	RM	SPL-IT-UC
p	p	b	b
t	t	d	d
k	k	g	g
pd	p	dd	d
td	t	kd	k
ts	t + s	dx	dx
m	m	en	n
n	n	ng	ng
s	s	sh	sh
z	z	jh	jh
ch	ch	dh	dh
f	f	th	th
v	v	-	-
l	l	el	l
r	r	w	w
y	y	sil	pau
hh	hh	pau	pau
eh	eh	ih	ih
ao	aa	ae	ae
aa	aa	ah	ah
uw	uw	uh	uh
er	er	oy	oy
ay	ay	iy	iy
ey	ey	ow	ow
aw	aw	ax	ah

Apêndice C

Tabela C.1: Resultados obtidos para a métrica principal Cnxe para o conjunto de desenvolvimento para *queries* de todos os tipos (T1+T2+T3), do tipo 1 (T1), do Tipo 2 (T2) e do Tipo 3 (T3).

Sistemas de Fusão	T1+T2+T3	T1	T2	T3
Sistema Primário + Informação Paralela	**0.7782**	**0.7101**	**0.7861**	**0.8123**
Sistema Secundário + Informação Paralela	0.7862	0.7163	0.7961	0.8198
Sistema Primário	0.7873	0.7207	0.7895	0.8298
Sistema Secundário	0.7957	0.7282	0.7992	0.8378

Tabela C.2: Resultados obtidos para a métrica secundária ATWV para o conjunto de desenvolvimento para *queries* de todos os tipos (T1+T2+T3), do tipo 1 (T1), do Tipo 2 (T2) e do Tipo 3 (T3).

Sistemas de Fusão	T1+T2+T3	T1	T2	T3
Sistema Primário + Informação Paralela	0.2341	**0.3488**	**0.1629**	0.1743
Sistema Secundário + Informação Paralela	0.2195	0.3366	0.1374	0.1665
Sistema Primário	**0.2343**	0.3461	0.1543	**0.1841**
Sistema Secundário	0.2276	0.3333	0.1483	0.1839

Tabela C.3: Resultados obtidos para a métrica principal Cnxe para o conjunto de avaliação para *queries* de todos os tipos (T1+T2+T3), do tipo 1 (T1), do Tipo 2 (T2) e do Tipo 3 (T3).

Sistemas de Fusão	T1+T2+T3	T1	T2	T3
Sistema Primário + Informação Paralela	0.7866	**0.7079**	0.8195	0.8151
Sistema Secundário + Informação Paralela	**0.7842**	0.7107	**0.8147**	**0.8115**
Sistema Primário	0.7930	0.7181	0.8219	0.8226
Sistema Secundário	0.7914	0.7223	0.8175	0.8200

Tabela C.4: Resultados obtidos para a métrica secundária ATWV para o conjunto de avaliação para *queries* de todos os tipos (T1+T2+T3), do tipo 1 (T1), do Tipo 2 (T2) e do Tipo 3 (T3).

Sistemas de Fusão	T1+T2+T3	T1	T2	T3
Sistema Primário + Informação Paralela	0.2064	0.3065	0.1714	0.1545
Sistema Secundário + Informação Paralela	0.2017	0.3150	0.1504	0.1513
Sistema Primário	**0.2157**	**0.3187**	**0.1846**	0.1585
Sistema Secundário	0.2098	0.3221	0.1595	**0.1600**

Apêndice D

Tabela D.1: Resultados oficais do desafio QUESST 2015 para o conjunto de avaliação de acordo com *queries* dos três tipos (All), do tipo 1 (T1), do Tipo 2 (T2) e do Tipo 3 (T3).

Equipa	Métrica Principal: Cnxe				Métrica Secundária: ATWV			
	All	T1	T2	T3	All	T1	T2	T3
NNI	0.7610	0.6093	0.8537	0.7828	0.2703	0.4356	0.1890	0.2030
SPL-IT-UC	**0.7842**	**0.7107**	**0.8147**	**0.8115**	**0.2017**	**0.3150**	**0.1504**	**0.1513**
BUT	0.8452	0.7859	0.8791	0.8587	0.1513	0.2539	0.0835	0.1217
GTM-UVIGO	0.9185	0.8640	0.9586	0.9128	0.0403	0.0692	-0.0050	0.0684
IIT-B	0.9536	0.9330	0.9852	0.9313	0.0254	0.0531	-0.0099	0.0525
TUKE	0.9714	0.9615	0.9757	0.9737	0.0029	0.0097	-0.0041	0.0076
CUNY	0.9989	0.9989	0.9978	1.0000	0.0006	-0.0009	0.0038	-0.0002
SPEED	1.0379	1.0383	1.0372	1.0385	-0.0762	-0.0764	-0.0799	-0.0749
ELIRF	1.0734	0.9167	1.1276	1.1381	0.1125	0.1978	0.0755	0.0801
NTU	2.0067	2.0070	2.0093	2.0029	-1.0828	-0.9959	-1.0705	-1.1273