Instituto Superior de Engenharia de Lisboa

Processamento de Fala

Engenharia Informática e Multimédia

Reconhecimento de palavras isoladas

Engº Carlos Meneses

MM2N

Arman Freitas nº45414

25 de junho de 2022

Conteúdo

Lista de Figuras

1 Introdução e objetivos

O presente projeto pretende o desenvolvimento de um reconhecedor de palavras isoladas. Para tal, foi utilizada a linguagem de programação *Python* com ajuda da biblioteca *Numpy*.

Procura-se um reconhecedor de palavras isoladas capaz de reconhecer dígitos de 0 a 10 em língua inglesa. Assim, tem-se um classificador multiclasse de 11 classes.

Serão abordados os seguintes tópicos no desenvolvimento:

- Algoritmo DTW (*Dynamic Time Warping*)
- Classificação.

Será utilizado o algoritmo DTW de forma a conseguir uma métrica para a classificação dos dígitos isolados. Este algoritmo utilizará os coeficientes LPC abordados no trabalho passado (elaboração de um *vocoder*) de forma a conseguir uma distância entre diferentes sinais de fala.

A classificação poderia ser realizada de várias formas, pelo que, será utilizado o classificador *K-Means*, como forma de predizer a classe presente num determinado sinal de fala.

O *K-Means* tem em conta os centroides das K classes mais próximas e, avalia novas instâncias consoante as suas distâncias a essas classes.

2 Desenvolvimento

O desenvolvimento contém os seguintes segmentos:

- Algoritmo DTW
- Fase de treino
- Fase de reconhecimento
- Palavras fora do léxico

O DTW encontrará a relação entre sequências. Já a fase de treino procura treinar o modelo com os dados de treino. A fase de reconhecimento deve-se a reconhecer os dados de teste apartir do treino. Por fim, são detetadas caso as palavras a detetar pertencem ao dicionário (dígitos de 0 a 9).

2.1 Algoritmo DTW (*Dynamic Time Warping*)

Devido a deformações no tempo, o DTW é bom para a classificação de sequências que têm diferentes frequências ou que estão fora de fase. Por exemplo, poderiam ser detetadas similaridades no andar de duas pessoas, mesmo que as suas velocidades variem uma da outra.

Este algoritmo foi proposto para reconhecimento de fala para palavras isoladas e léxicos de pequena dimensão, embora também hajam métodos mais robustos para reconhecimento.

Em geral, o algoritmo DTW encontra o alinhamento ótimo entre duas sequências com tempos diferentes, com algumas regras e restrições:

- Todos os índices da primeira sequência devem corresponder a um ou mais índices da outra sequência.
- O primeiro índice da primeira sequência deve corresponder ao primeiro índice da outra.
- O ultimo índice da primeira sequência deve corresponder ao ultimo índice da outra.

2.1.1 Corte do sinal

Para a comparação de duas sequências foram utilizados os LPCs de cada sinal. No entanto, antes de prosseguir o alinhamento de dois sinais de fala é necessário cortar o mesmo, removendo as zonas silenciadas.

Para tal, são utilizados dois limiares, k_1 e k_2, que ajudam no corte do sinal. Para poder cortar os coeficientes LPC nas zonas de silêncio, descobrem-se os pontos de corte na potência do sinal (energia por intervalo de tempo). A razão de ser utilizada a potência tem a ver com o facto da mesma estar diretamente relacionada ao volume do sinal de fala.

Nas seguintes figuras (fig. 1 e 2) é possível ver a potência do mesmo sinal completo e, cortado com $k_1 = 0.0001$ e $k_2 = 0.0003$:

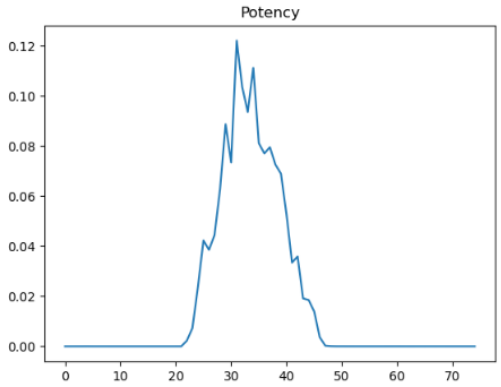

Figura 1: Potência do sinal

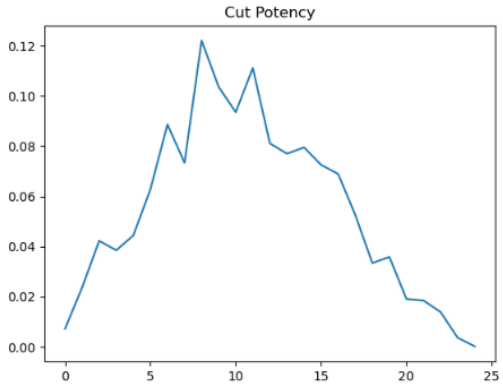

Figura 2: Potência do sinal cortada

Com as zonas de corte estabelecidas, procede-se a cortar os coeficientes LPC nas mesmas zonas.

2.1.2 Construção da matriz DTW

Com os sinais cortados, constrói-se a matriz DTW. Esta consiste em criar uma matriz 2D, onde o eixo do x corresponde ao sinal de teste e o do y ao sinal de treino.

Numa matriz DTW, para cada célula é calculada a distância euclidiana entre os vetores dos LPCs dos dois sinais. No entanto, apenas é tida em conta a seguinte região da matriz, contida pelas retas A, B, C e D:

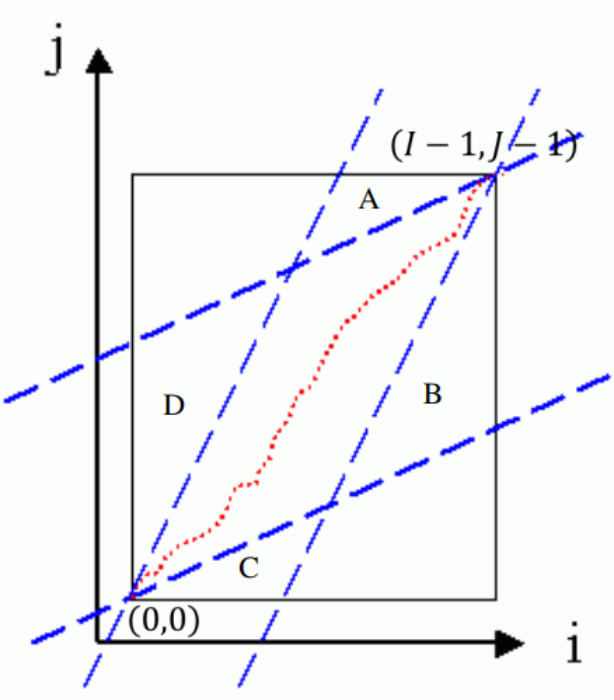

Figura 3: Região global da matriz DTW

De seguida, esta matriz é alinhada, percorrendo as suas células e, para cada uma somando a célula vizinha com o valor mínimo. Desta forma, a matriz DTW encontra-se construída. A figura 4 demonstra uma matriz DTW construída entre dois sinais diferentes. Através desta matriz, é possível obter o caminho mais pequeno (fig. 5) e, assim a distância.

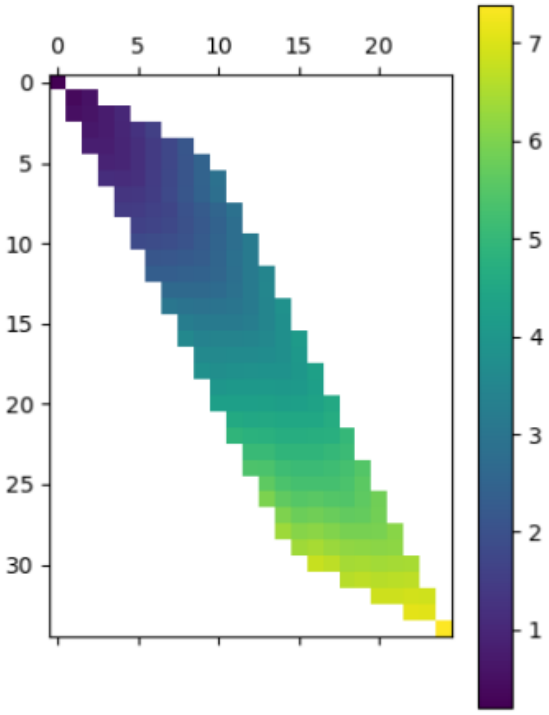

Figura 4: Matriz DTW

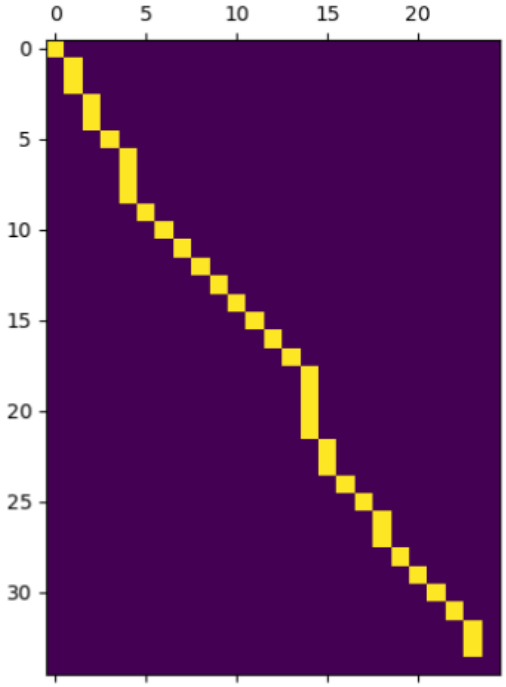

Figura 5: Caminho da matriz DTW

2.2 Fase de treino

Como se pôde verificar no algoritmo DTW, é possível calcular a distância entre dois sinais. É através desta distância que se classificar futuros sinais de fala.

O projeto dispõe de vários ficheiros de fala, uns para treino, outros para teste.

Como se determinou no início do projeto, será utilizado o algoritmo K-distâncias mais próximas. Este algoritmo procura aumentar a eficiência do treino e, posteriormente do reconhecimento. Assim, será determinada uma instância representante por cada classe, dos ficheiros de treino.

A instância representante por cada classe é o vetor LPC que contém o somatório da distância (calculada pelo DTW) a todas as outras instâncias da mesma classe menor.

A cada instância representante dá-se o nome de centroide.

2.3 Fase de reconhecimento

A fase de reconhecimento, tal como o nome implica, é a fase onde é possível reconhecer a classe presente em cada ficheiro de teste.

Nesta fase, são utilizados os dados de treino, incluindo os centroides previamente calculados, para reconhecer a que classe pertence cada instância de teste. Existem dois possíveis métodos para este reconhecimento:

Método 1: Assume-se a sequência a reconhecer como pertencente à classe que obtiver a menor distância global em relação ao seu centroide.

Método 2: Este método é um pouco mais complexo sendo constituído pelos seguintes passos:

1. Escolhe-se as k classes que obtiverem menor distância global em relação ao seu centroide.
2. Calcula-se a distância global em relação a todas as instâncias das k classes escolhidas.
3. Assume-se a sequência a reconhecer como pertencente à classe da instância que obtiver menor distância global.
4. No limite pode-se logo calcular a distância para todas as instâncias de todas as classes.

2.4 Palavras fora do Léxico

O presente projeto abrange um total de 11 classes. Este segmento visa compreender caso um sinal de fala que entre no algoritmo, realmente pertence a uma das classes ou, se trata de uma palavra fora do léxico.

Uma palavra fora do léxico, tal como o nome diz, é uma palavra que não está presente no vocabulário (neste caso nas classes presentes).

Para que palavras fora do léxico não sejam reconhecidas erradamente, foi utilizada a seguinte solução:

1. Na fase de treino são calculadas as distâncias máximas de cada classe. Em cada classe, a distância máxima, é a distância da instância mais distante do centroide da mesma classe. É de notar que tiveram de ser ignoradas distâncias infinitas.

2. Na fase de reconhecimento assume-se uma sequência como não pertencente ao léxico se a distância mínima calculada for maior que uma tolerância (neste caso 1,5) multiplicada pela distância máima da classe reconhecida.

2.5 Resultados

Por fim, após concluído o desenvolvimento do classificador, foram realizados vários testes com diferentes dados. Assim, foram tidos em conta os seguintes cenários:

1. Classificação aos 2 centroides mais próximos. Onde são escolhidos os dois centroides mais próximos e realizadas as distâncias a todos os elementos de treino destas duas classes, o elemento com a distância mínima é a classe a ser classificado (figuras 6 e 7). É de notar que aqui são utilizadas palavras fora do léxico.

2. Classificação de centroides sem aprofundação. Isto é, apenas são realizadas as distâncias aos centroides e, classificado no centroide mais perto (figuras 8 e 9).

3. Classificação sem o uso de centroides, tendo sido realizada a distância de cada instância de teste a todas as instâncias de treino (figuras 10 e 10).

4. Classificação tal que o género a ser testado seja contrário do género treinados (figuras 12 e 13).

Para uma compreensão mais fácil, seguem-se as matrizes de confusão para os cenários descritos acima:

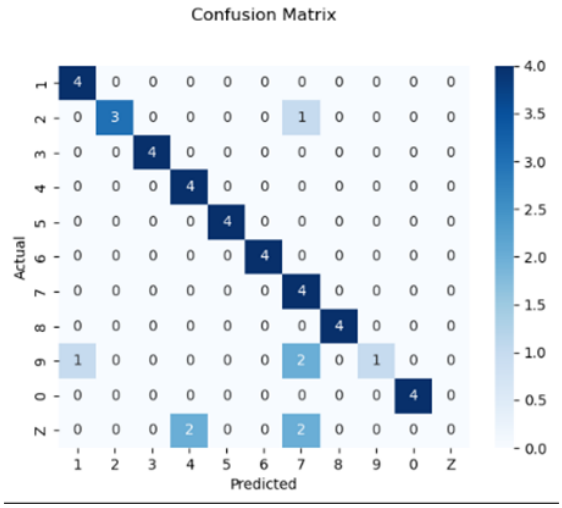

Figura 6: Matriz de confusão (total) - Dados femininos

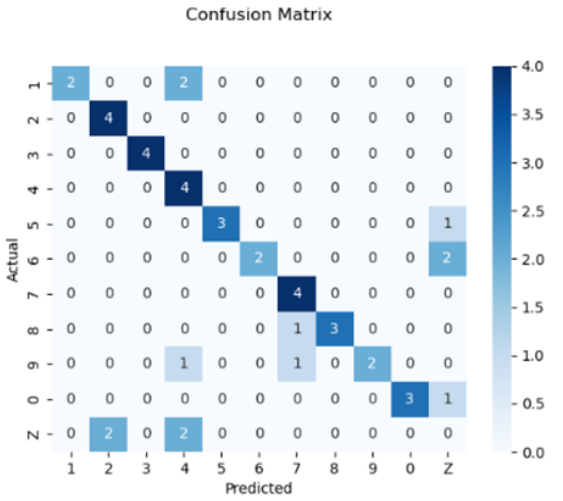

Figura 7: Matriz de confusão (total) - Dados masculinos

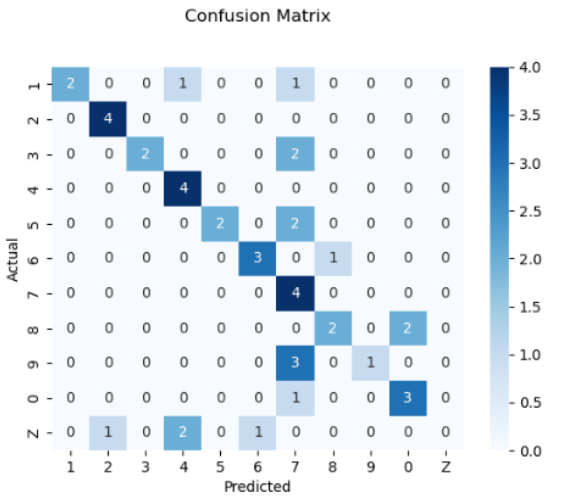

Figura 8: Matriz de confusão (apenas classificação por centroides) - Dados femininos

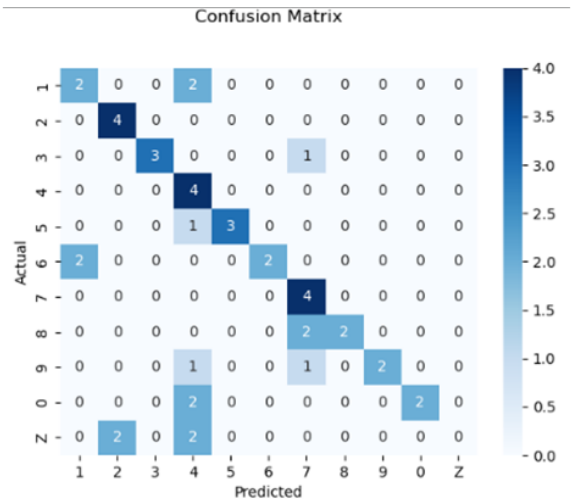

Figura 9: Matriz de confusão (apenas classificação por centroides) - Dados masculinos

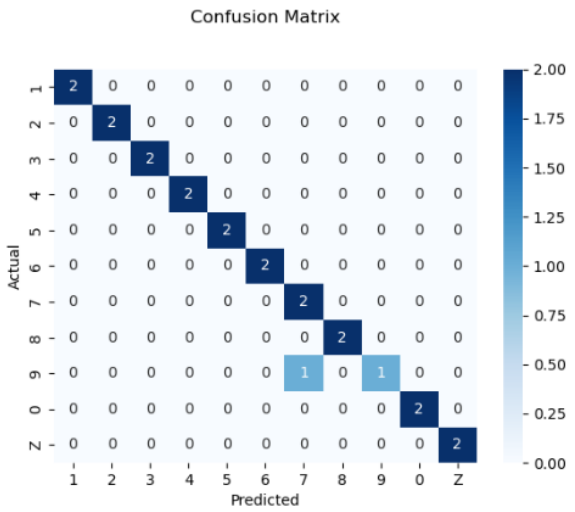

Figura 10: Matriz de confusão (classificação total) - Dados femininos

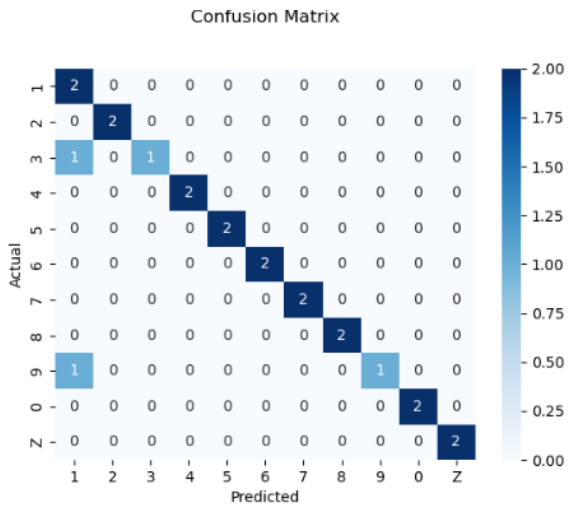

Figura 11: Matriz de confusão (classificação total) - Dados masculinos

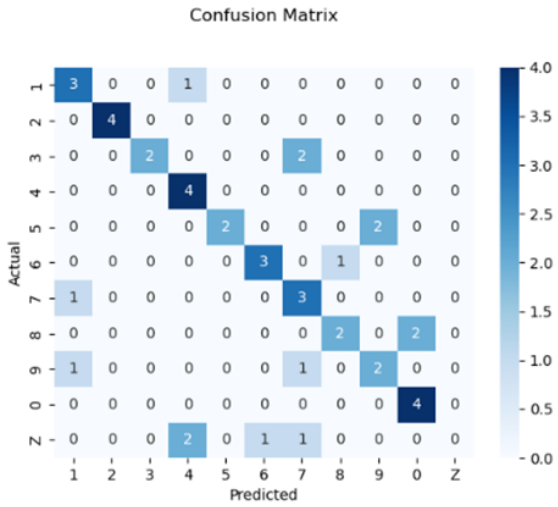

Figura 12: Matriz de confusão - Dados de teste femininos e Dados de treino masculinos

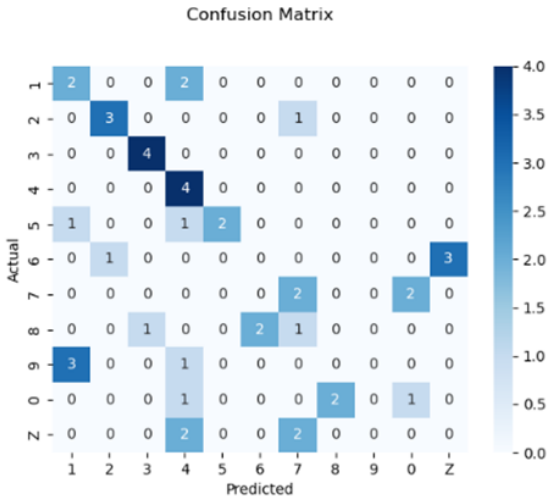

Figura 13: Matriz de confusão - Dados de teste masculinos e Dados de treino femininos

Assim como as matrizes de confusão, foram recolhidas as respetivas precisões para os respetivos cenários.

Género / Tipo de Classificação	Masculino	Feminino
1. Dois Centroides e Palavras fora do léxico	70.5%	81.8%
2. Centroides sem aprofundação	63.6%	61.4%
3. Treino e teste a todos os dados	90.9%	95.5%
4. Treino/Teste com sexos opostos	65.9%	40.9%

Após uma análise aos dados recolhidos, consegue-se concluir que, de todos os cenários, o treino e teste com todos os dados do conjunto de treino (3) geram os melhores resultados. É um resultado esperado pois o tempo de processamento é bastante maior relativamente a qualquer outro cenário.

Conclui-se também que a classificação apenas aos centroides sem qualquer aprofundação (2) retorna os piores resultados. Este cenário, abstrai de quase todas as instâncias de treino (mantendo apenas o centroide) e, apenas utiliza uma por classe. Naturalmente, apesar de um tempo de classificação rápido, obteve resultados abaixo dos 65%.

O uso dos dois centroides mais próximos e, das palavras fora do léxico (1) proporcionou resultados mais coerentes, sendo que os resultados são bons e, o tempo de classificação também.

Em termos de comparação de géneros, pode-se concluir pelos cenários 1 e 3, que vozes femininas são mais fáceis de classificar.

Também foram classificadas instâncias masculinas com conjuntos de treino femininos e vice-versa. Aqui, conseguiu-se verificar que, o uso de um conjunto de treino masculino para testes femininos obteve resultados muito baixos. Pelo contrário, o uso de um conjunto de treino feminino para testes

masculinos proporcionou uma precisão e cerca de 65.9%, o que é significativamente mais alto que o treino masculino.

3 Conclusões

Este projeto proporcionou uma melhor compreensão de como o reconhecimento de fala realmente funciona. Foi criado um maior entusiasmo pela forma de como os algoritmos avançados de hoje tal como a Siri, o Google Assistant e a Alexa, reconhecem não só palavras isoladas, mas frases inteiras através apenas da fala.

Apesar de ter sido utilizado para sequências de áudio, o algoritmo DTW pode também ser aplicado a outras diversas áreas tais como vídeo, gráficos e dados. Qualquer dado que se consiga tornar numa sequencia linear pode ser analisado através de DTW.

Ao longo do desenvolvimento do projeto houve alguns obstáculos que, certamente foram superados com a ajuda do docente e dos colegas presentes. Outros aspetos como a classificação de instâncias com a distância a LPCs pertencentes ao conjunto de treino inteiro, demoraram bastante a correr no computador, devido ao elevado número de instâncias de treino e teste presentes que, para um computador mais fraco pode levar a um tempo de processamento alto. Foi também interessante ver a diferença entre a classificação de voz feminina e masculina, havendo, na maior parte das vezes uma classificação melhor nas vozes femininas.

Considera-se que os objetivos propostos para o projeto foram todos alcançados, tendo sido realizado o reconhecedor de palavras isoladas no *python* e, posteriormente analisados vários cenários de forma a compreender a forma de classificação utilizada.

4 Referências

- *Slides* fornecidos pelo docente.
- *Wikipedia* - https://en.wikipedia.org/wiki/Dynamic_time_warping