

Universidade do Minho
Departamento de Informática

Engenharia de Telecomunicações e Informática

PROCESSAMENTO DIGITAL DE SINAL

Reconhecimento de Voz

Grupo 9

Fernando João Santos Mendes A101263

Junlin Lu A101270

Conteúdo

Resumo.....	3
1. Introdução	3
1.1 Objetivo do Trabalho	3
1.2 Funcionalidades/Possibilidade.....	4
1.3 Público-Alvo	4
2. Instalação de Software/Bibliotecas.....	4
2.1 Software/configuração necessário	4
2.2 Requisitos das bibliotecas.....	5
2.2 Configuração	5
3. Interface do Utilizador.....	6
4. Funcionalidades Principais.....	6
5. Gestão de dados	8
• Importação de dados	8
• Exportação de Dados	8
• Backup e Restauração.....	8
6. Resolução de problemas	8
7. Exemplos Práticos	8
8. Atualizações futuras	9
9. Conclusão	9
10. Links.....	10
Github do projeto	10
Dataset	10
Training script	10
Gravação com filtro.....	10
Diagrama(early version)	10

Resumo

Este projeto consistiu na criação de um programa que permitisse otimizar e automatizar a interação entre um utilizador e o seu próprio dispositivo através do reconhecimento de voz.

Através de comandos pré-definidos é possível aceder a aplicações que estejam no nosso dispositivo mediante a voz do utilizador.

Para que tal fosse possível procuramos criar um código que se sustentasse sobre nos Modelos Ocultos de Markov (HMMs) como fundamento principal do projeto e seguimos então com o treinamento deste modelo segundo o contexto a cima descrito. As palavras chave utilizadas foram “Excel”, “Yes”, “No”.

Assim sendo conseguimos de forma um pouco mais restrita simular uma situação exemplo onde colocamos em prática o reconhecimento de voz.

Além dos filtros aplicados no processamento do HMMs tentamos ainda implementar um filtro passa-baixo (utilizando uma biblioteca) para reduzir o ruído de forma a tornar as gravações mais perceptíveis e fundamentar os conceitos abordados em sala de aula.

1. Introdução

1.1 Objetivo do Trabalho

O objetivo deste trabalho foi pensado e criado tendo em conta as aprendizagens retiradas ao longo das aulas referentes a filtros, processamento da informação e também foi fomentado pela curiosidade sobre a área das IA e métodos relacionados.

Definimos que a nossa aplicação deveria responder a uma necessidade ou utilidade da qual serviria de interesse comum.

Como tal o nosso programa foca-se em otimizar a procura das aplicações no nosso computador e incentivar à tecnologia moderna de comunicação com os dispositivos.

1.2 Funcionalidades/Possibilidade

Algumas das funcionalidades do nosso projeto que se destacam são:

- A utilização da voz para ordenar a abertura de aplicações em tempo real;
- O uso de filtros que permitem suavizar as amostras para assim uma melhor comparação;
- Possibilidade de incrementar o número de aplicações executáveis;

1.3 Público-Alvo

O nosso projeto foca-se naqueles que tem por hábito a utilização do computador para as suas tarefas, lazer ou trabalho e que queiram tornar esta interação com a sua máquina algo mais dinâmico e iterativo de uma forma divertida e otimizada.

2. Instalação de Software/Bibliotecas

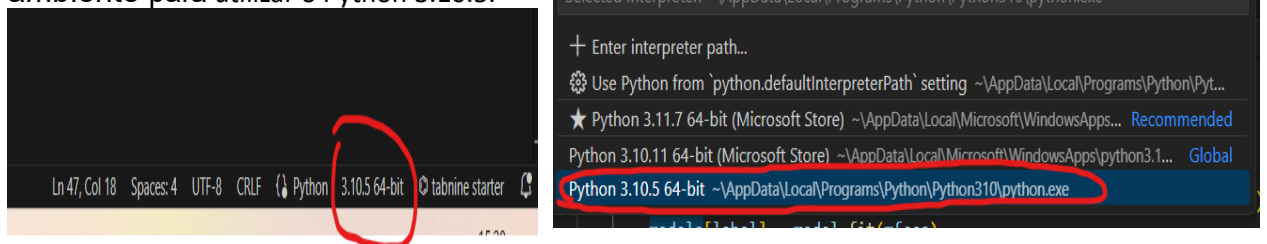
2.1 Software/configuração necessário

Para o desenvolvimento e funcionamento do nosso projeto, é imprescindível que você tenha instalado o IDE (recomenda VS code) e a versão 3.10.5 do Python. Estas são as ferramentas fundamentais para o trabalho com o nosso projeto.

VS code: <https://code.visualstudio.com/download>

Python 3.10.5: <https://www.python.org/downloads/release/python-3105/>

Após concluir todas as instalações, abra qualquer script Python e configure o ambiente para utilizar o Python 3.10.5.

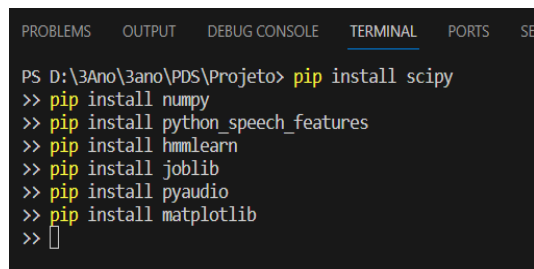


2.2 Requisitos das bibliotecas

- scipy
- python_speech_features
- time
- hmmlearn.hmm
- joblib
- os
- pyaudio

Instalação das bibliotecas é feita através seguintes comandos no terminal:

```
pip install scipy
pip install numpy
pip install python_speech_features
pip install hmmlearn
pip install joblib
pip install pyaudio
pip install matplotlib
```



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SE
PS D:\3Ano\3ano\PDS\Projeto> pip install scipy
>> pip install numpy
>> pip install python_speech_features
>> pip install hmmlearn
>> pip install joblib
>> pip install pyaudio
>> pip install matplotlib
>> 
```

2.2 Configuração

Por favor, **lembre-se de alterar o endereço e o nome do arquivo de saída no código**. O caminho padrão está configurado como:

X:\\xxx\\xxx\\PDS_G9_A101263_A101270\\Voice\\test\\test.wav

'X:\\xxx\\xxx\\PDS_G9_A101263_A101270\\Voice\\response\\response.wav

o que significa que o arquivo será salvo no diretório atual.

Se desejar salvar em um local diferente, ajuste a variável WAVE_OUTPUT_FILENAME com o caminho desejado. Além disso, verifique e ajuste o caminho do modelo wave que será carregado, para garantir que o programa funcione corretamente.

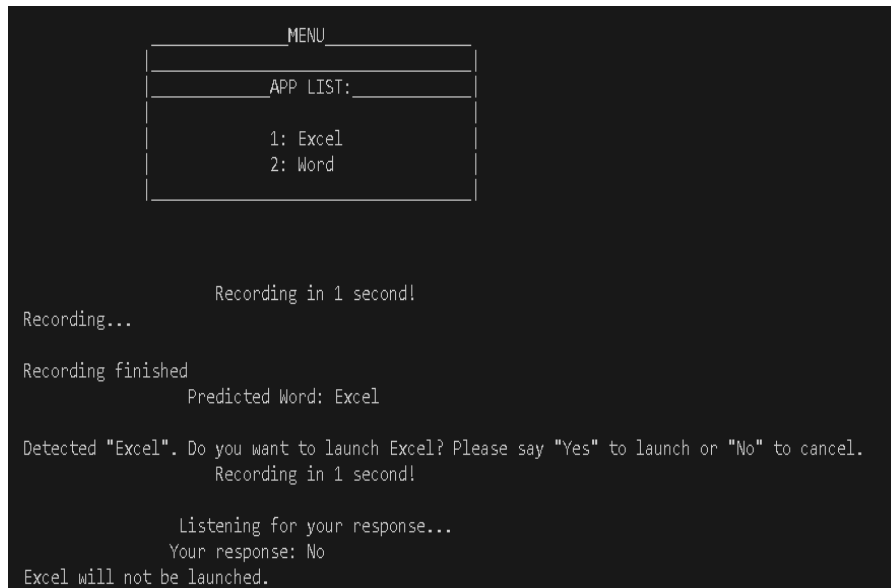
```
15 WAVE_OUTPUT_FILENAME = "D:\\3Ano\\3ano\\PDS\\PDS_G9_A101263_A101270\\Voice\\test\\test.wav"
```

```
RESPONSE_WAVE_OUTPUT_FILENAME = "D:\\3Ano\\3ano\\PDS\\PDS_G9_A101263_A101270\\Voice\\response\\response.wav"
```

```
models = joblib.load('wave.ckpt')
```

3. Interface do Utilizador

Para que a interação com utilizador seja mínima procuramos ter uma interface simples e objetiva.



4. Funcionalidades Principais

- Reconhecimento de voz:

Descrição:

A funcionalidade permite captar a voz do utilizador;

Passos para a utilização:

Inicie o programa e aguarde que lhe seja dada uma mensagem a sinalizar o início da captação do áudio;

Dicas / Boas práticas:

Evitar ambiente com altas perturbações sonoras;

Evitar diversos locutores ao mesmo tempo;

- Escolha da aplicação que deseja executar:

Descrição:

A funcionalidade permite captar qual a aplicação pronunciada pelo utilizador;

Passos para a utilização:

Após iniciar a gravação pronuncie a palavra que se refere à sua aplicação;

Dicas / Boas práticas:

Pronunciar os nomes e não os seus diminutivos ou abreviaturas;

- Filtragem do áudio:

Descrição:

A funcionalidade permite utilizar um filtro que diminua o ruído externo de forma a suavizar o som para melhor recolha das características do mesmo;

Passos para a utilização:

Utilizar um outro ficheiro onde colocamos o filtro em questão a funcionar para treinar o modelo;

Dicas / Boas práticas:

Pronunciar os nomes e não os seus diminutivos ou abreviaturas;

Evitar ambiente com altas perturbações sonoras;

Evitar diversos locutores ao mesmo tempo;

5. Gestão de dados

- **Importação de dados**

Os dados usados são importados de ficheiros Wav. Devidamente organizados nas suas pastas.

- **Exportação de Dados**

Caso seja utilizada a funcionalidade de treino do modelo o resultado da mesma será um ficheiro Wav.

- **Backup e Restauração**

Não implementamos nenhum método que permita fazer o Backup e a Restauração.

6. Resolução de problemas

Nas situações onde eventualmente o programa sofre algum erro como não identificar as palavras que fazem parte das palavras chave, ele envia uma mensagem a mostrar aquilo que foi percebido pelo mesmo e desta forma o utilizador poderá recomeçar a sua interação.

7. Exemplos Práticos

- **Casos de uso comuns:**

Navegar pelas aplicações do sistema através do uso da voz.

- **Demonstração de funcionalidades em Situações Reais**

Para comprovar o funcionamento do nosso projeto vamos enviar junto com o relatório um vídeo expositivo.

8. Atualizações futuras

Num futuro próximo pretendemos realizar uma interface que seja intuitiva e se possível criarmos um atalho no ambiente de trabalho de fácil e rápido acesso para se utilizar o programa.

Esperamos também aumentar o nosso DataSet treinando o modelo com mais exemplos para que seja maior a sua assertividade.

9. Conclusão

O desenvolvimento deste projeto proporcionou-nos o desenvolvimento de um equalizador do zero, uma ferramenta que já nos era familiar, mas que ganhou uma nova dimensão ao ser desenvolvida por nós. A experiência de desenvolver o equalizador, desde a captura do áudio até a implementação dos filtros digitais, tornou o resultado final mais gratificante.

Aplicamos na prática os conhecimentos que adquirimos na disciplina de Processamento Digital de Sinal, desde a parte da amostragem até aos filtros digitais, fomos capazes de entender cada etapa do processo. Termos tido o objetivo bem definido desde o início do projeto foi importante e o resultado final foi ao encontro do que tínhamos planeado

Este projeto permitiu-nos enfrentar desafios reais no processamento digital de sinal. A capacidade de criar um equalizador personalizado, onde é possível ajustar os parâmetros como frequência de corte, largura de banda e ganho, demonstrou-nos realmente qual a utilidade dos conceitos aprendidos ao longo do semestre.

10. Links

Github do projeto

<https://github.com/K0Kosuki/3ano/tree/main/PDS/Projeto>

Dataset

<https://github.com/K0Kosuki/3ano/tree/main/PDS/Projeto/Dataset>

Training script

<https://github.com/K0Kosuki/3ano/blob/main/PDS/Projeto/Train.py>

Gravação com filtro

https://github.com/K0Kosuki/3ano/blob/main/PDS/Projeto/Record_Filter.py

Reconhecimento de voz

<https://github.com/K0Kosuki/3ano/blob/main/PDS/Projeto/test.py>

Diagrama(early version)

https://github.com/K0Kosuki/3ano/blob/main/PDS/Voice%20test/Reconhecimento_de_Voz2.dia