



UNIVERSIDADE
LUSÓFONA

Classificação de géneros musicais com integração de bases de dados

Trabalho Final de curso

Relatório Intercalar 1º Semestre

Nome do Aluno: Eduardo Colaço Marques

Nome do Orientador: professor Fernando Teodósio

Trabalho Final de Curso | LEI | 10 de Setembro 2021

www.ulusofona.pt

Direitos de cópia

(Classificação de géneros musicais com integração de bases de dados), Copyright de *(Eduardo Colaço Marques)*, ULHT.

A Escola de Comunicação, Arquitectura, Artes e Tecnologias da Informação (ECATI) e a Universidade Lusófona de Humanidades e Tecnologias (ULHT) têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objectivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

Resumo

Neste relatório está descrito o projeto desenvolvido no âmbito do trabalho de final de curso atribuído pelo professor Fernando Teodósio.

Após várias iterações este trabalho obteve o objetivo do desenvolvimento em python de uma prova de conceito de um sistema de classificação de géneros musicais com recurso a uma rede neural convolucional[1] e inserção dinâmica em bases de dados .

A classificação no seu estado atual consegue analisar ficheiros do tipo “.mp3” e “.wav” através dos coeficientes cepstrais de frequência de mel[5] e passando esses valores por uma rede neural convolucional treinada com recurso ao conjunto de dados GTZAN[3] devolve uma previsão do género musical do ficheiro.

No que toca à inserção dos dados obtidos, não foi possível automatizar o processo nem criar compatibilidade para diferentes tipos de base de dados, no entanto é possível ao utilizador selecionar as qualquer tabela numa base de dados SQL e colocar uma entrada na mesma inclusive de ficheiros através da conversão dos mesmos para “*binary large objects*” ou *BLOBS*.

No futuro sinto que este projeto poderia continuar a ser desenvolvido tanto para melhorar a precisão da rede neural através de melhorias no algoritmo tanto para criar integração com diferentes tipos de bases de dados como “*MongoDB*”, “*MariaDB*” entre outras.

Palavras-chave: trabalho de final de curso, rede neural convolucional, bases de dados

Abstract

This document describes the final work for the course of computing engineering from the university Lusófona de Humanidades e Tecnologias.

After several iterations this project obtained its final objective in the form of the development in python of a proof-of-concept app that classifies musical genres through a convolutional neural network [1] and allows dynamic database insertions.

The classification in its current state can analyse *“.mp3” and “.wav” type files* by passing their mel frequency cepstral coefficients[5] through a neural network which was trained with aid from the GTZAN dataset[3] to return the audio file's musical genre

When it comes to database insertions it wasn't possible to automate the process nor add compatibility with different types of databases, however it's possible for the user to select any table in the database and make an entry including files by converting them to binary large objects or BLOBS

In the future I feel like this project could see continued development both to increase the neural network precision through algorithm improvements, but also to create integration with different types of databases such as MongoDB, MariaDB amongst many others

keywords: final project, convolutional neural network, databases

Índice

Resumo.....	3
Abstract	4
Índice	5
Lista de Figuras	6
Lista de Tabelas	7
1 Identificação do Problema	8
2 Levantamento e análise de requisitos	10
3 Viabilidade e Pertinência.....	12
4 Solução Desenvolvida.....	13
4.1. Conjunto de dados	14
4.2. Tratamento de Dados.....	15
4.3 Rede Neural.....	18
4.3.1 Rede Neural Convolucional	18
4.4 Bases de dados.....	20
4.5 Programa Desenvolvido	21
5 Benchmarking.....	23
6 Métodos e planeamento.....	24
7 Resultados	25
8 Calendário	26
9 Conclusão	27
Bibliografia	28
Glossário.....	29

Lista de Figuras

Figura 1. cotas do mercado de streaming de música no 1º quartil de 2021	8
Figura 2. Representação do sinal de som o ficheiro rock 17	15
Figura 3. espectograma do ficheiro rock 17	16
Figura 4. MFCC's do ficheiro rock 17	17
Figura 5. Tabela de testes criada na base de dados	20
Figura 6. Funcionamento da aplicação desenvolvida	22

Lista de Tabelas

Tabela 1- tabela de requisitos funcionais..... 10

Tabela 2-tabela de requisitos não funcionais..... 11

Tabela 3-tabela de tecnologias usadas..... 13

1 Identificação do Problema

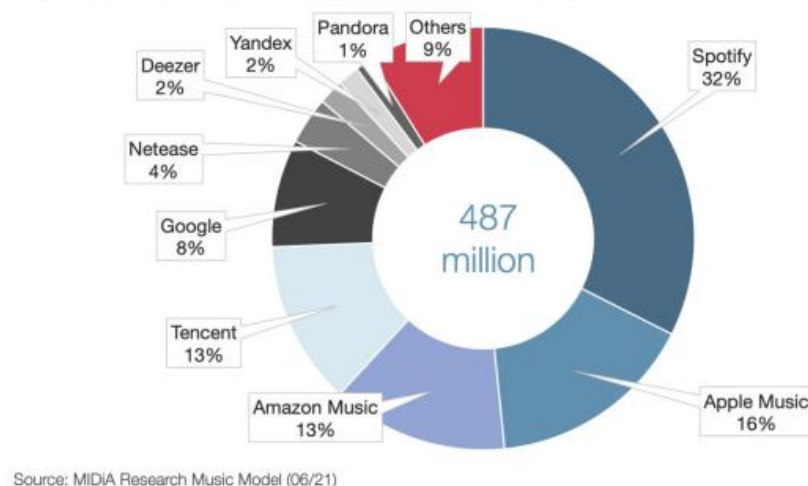
Este capítulo relata o enquadramento do problemas que identificamos para num capítulo seguinte desenvolver-mos a sua solução.

O desenvolvimento deste trabalho começou com o simples tema da classificação automática de géneros musicais mas após um breve estudo da arte foi nos possível concluir que a área de estudo já se encontrava bastante saturada sendo que grandes empresas já empregam tecnologias derivadas destes estudos como é o caso do *spotify* e do seu *echo nest*[4] que é um sistema que entre outras funções identifica e organiza as musicas presentes na base de dados por géneros musicais.

Decidimos após a conclusão anterior alterar a perspectiva com que abordáramos o trabalho, e após um estudo do mercado foi nos chegou-se á conclusão que mais de 50% do mercado de streaming de música pertence a grandes empresas[7]:

Spotify maintains subscriber market share lead but Amazon, Tencent and Google made big gains

Streaming music subscriber market shares, Q1 2021, global



MIDIA

Figura 1. cotas do mercado de streaming de música no 1º quartil de 2021

Empresas que como o exemplo analisado anteriormente, têm os seus sistemas privados de classificação e organização géneros musicais, sistemas demasiado dispendiosos para serem criados[10] para novas empresas que queiram entrar no mercado. Uma alternativa que seria possível para estas empresas seria a utilização de um dos vários projetos já existentes no entanto cada entrada na base de dados teria que ser feita à mão.

É neste contexto que surge a solução proposta por este relatório, isto é, um sistema que agrupe o processo de classificação e de inserção de dados nas bases de dados através de uma aplicação open-source desenvolvida em python com recurso a diversas bibliotecas pré-existentes de modo a que possa ser utilizado por empresas que se queiram estabelecer no mercado mas não tenham o mesmo capital da competição.

No entanto após desenvolvimento e conclusão do projeto devo salientar que o projeto existe num estado de prova-de-conceito e que carece ainda de bastante desenvolvimento sendo que

n me foi possível implementar alguns dos requisitos planeados inicialmente quer por falta de conhecimentos como por limitações relacionadas com o tempo para o desenvolvimento.

2 Levantamento e análise de requisitos

Este tópico é responsável pela listagem de requisitos necessários para o desenvolvimento do projeto.

Após uma reflexão sobre funções necessárias ao funcionamento da aplicação proposta foi necessário criar uma lista de requisitos para a mesma.

Tabela 1- tabela de requisitos funcionais

Tabela de requisitos			
Titulo	Descrição	Critério de aceitação	Implementação
Input de ficheiros na aplicação	A aplicação deve aceitar a introdução de ficheiros de som por parte do utilizador	A)O ficheiro tem formato correto	Concluída
Análise de ficheiros	A aplicação deve automaticamente analisar o conteúdo do ficheiro e se possível classificá-lo	A) O ficheiro tem de ter uma duração mínima de 1 minuto	Parcialmente
Input de dados numa base de dados	Após a análise do ficheiro permitir ao utilizador colocar os dados associados dentro de uma base de dados	A) Conexão à base de dados a funcionar corretamente B) Todos os dados obrigatórios estão presentes	Concluída
Visualização de itens presentes na base de dados	Permite ao utilizador verificar que musicas e que géneros já estão presentes na base de dados	A)Conexão à base de dados a funcionar corretamente	Não implementado

Tabela 2-tabela de requisitos não funcionais

Tabela de requisitos		
Titulo	Descrição	Implementação
Verificação de duplicados	A aplicação devera automaticamente verificar se existem itens duplicados dentro da base de dados	Não implementado
Navegabilidade	A aplicação deve ter um <i>UI</i> simples e intuitivo	Parcialmente
Garantir integridade de dados	O sistema deve garantir que os dados estão de acordo com os restantes da base de dados	Concluída

E após o desenvolvimento do projeto foi possível concluir que alguns dos requisitos não seriam possíveis de implementar por dificuldades técnicas ou por conflito com outros requisitos

3 Viabilidade e Pertinência

Neste capítulo verificamos a pertinência e viabilidade de mercado do trabalho em desenvolvimento através de um estudo da bibliografia

O trabalho desenvolvido infelizmente carece em termos de pertinência devido a não existir um cliente externo pelo que no seu desenvolvimento optamos por algo mais generalizado e maleável que possa futuramente ser trabalhado por outras entidades ou colegas.

Como foi possível analisar no capítulo anterior o mercado de streaming de música está dominado por grandes empresas[7] que por sua vez desenvolveram sistemas privados ou proprietários de organização e classificação de músicas dentro das suas próprias bases de dados, sistemas muitas vezes desenvolvidos pela própria empresa ou por subsidiárias cujo o capital é na ordem dos milhões[7].

Este trabalho encontra a sua viabilidade na existência de pequenas empresas que não consigam ou não estejam dispostas desenvolver sistemas próprios de classificação embora estas consistam menos de 9%[7] do mercado sinto que é justificável o desenvolvimento de ferramentas que estas possam usar.

No entanto o ponto anterior não justifica o trabalho por si só, sinto que uma das mais valias deste projeto é o seu desenvolvimento em open-source num repositório github permitindo a qualquer pessoa desenvolve-lo posteriormente inclusive futuros alunos.

Após testes preliminares a versão está funcional como protótipo ou prova-de-conceito, garantindo a análise relativamente precisa de músicas assim como a integridade de dados com as bases de dados usadas sendo que por enquanto apenas aceita ligações a bases de dados SQL o que demonstra a sua possível utilidade no futuro.

4 Solução Desenvolvida

Este capítulo tem como objetivo descrever os processos e recursos utilizados na resolução do problema apresentado, as suas características e o desenvolvimento de um programa que sirva como prova de conceito da solução .

Começando por justificar a seleção da feature utilizada na análise do som assim como as tecnologias utilizadas.

Tabela 3-tabela de tecnologias usadas

Tabela de Tecnologias usadas	
Nome	Descrição
Python	Linguagem de programação de alto nível
<i>keras</i>	Keras é uma biblioteca de Python que permite a criação de redes neurais e a implementação de algoritmos de <i>machine learning</i>
Librosa	Esta é uma biblioteca de Python que permite a leitura e tratamento de ficheiros de som assim como extração “features” dos mesmos
<i>Numpy</i>	É uma biblioteca vocacionada á realização de operações matemáticas de maior grau de complexidade
<i>Sklearn</i>	Biblioteca que permitiu fazer a divisão do conjunto de testes
<i>json</i>	Uma biblioteca que permite a leitura e tratamento de dados do tipo JSON
<i>Mysql connector</i>	Permite ao programa estabelecer ligações com uma base de dados e inclusive executar operações na mesma
<i>TKinter</i>	Biblioteca que permite a criação interfaces gráficas
<i>GTZAN dataset</i>	É um conjunto de dados que contem ficheiros de som de cerca de 30 segundos separados por género musical que será usado para treinar o algoritmo de deep learning

4.1. Conjunto de dados

Para garantir a precisão de um sistema de “deep learning” é necessário utilizar um conjunto de dados com elevado nível de precisão no seu processo de treino, preferencialmente este conjunto de dados deve já ter sido utilizado em estudos prévios de modo a garantir a sua validade.

Como tal decidimos escolher o conjunto de dados GTZAN já utilizado em diversos outros trabalhos[8][9]

que é composto por cerca de 100 amostras de 30 segundos para cada um dos 10 géneros representados, cada uma destas amostras tem uma frequência de 22050Hz/SampleRate.

Este Conjunto de dados também possui uma lista de espectrogramas de mel correspondentes a cada uma das amostras de música, também organizados por género, para análise visual.

4.2. Tratamento de Dados

Sinais de som podem ser representados graficamente pela sua amplitude no domínio do tempo através da sua “digitalização” com recurso á biblioteca librosa e a taxa de amostragem correta ,neste caso 22050Hz,

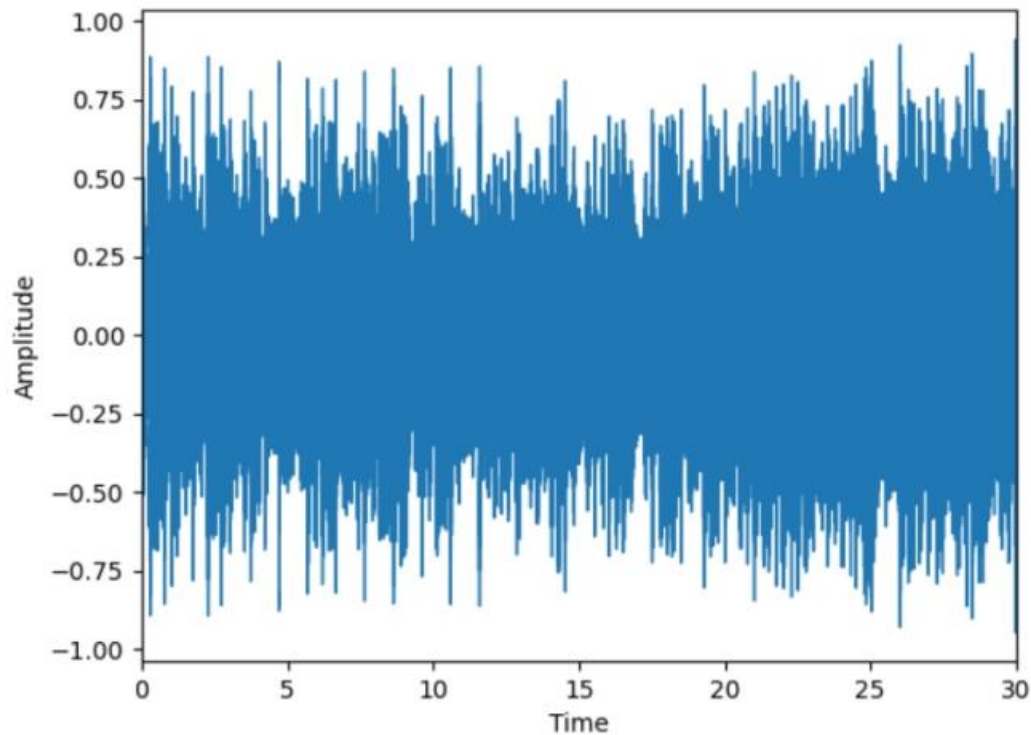


Figura 2. Representação do sinal de som o ficheiro rock 17

No entanto estes gráficos não possuem informação suficiente para ser futuramente analisada pela nossa rede neural temos ainda o facto que o sinal não deve ser analisado como um todo mas sim em partes cujo o tamanho é determinado pelo valores de “hop length”(tamanho de salto) e de “frame length”(tamanho de janela).

Mas gráficos tempo/Amplitude não têm informação suficiente para representar um sinal sonoro é necessário aplicar uma transformada rápida de Fourier para representar estes sinais no domínio da frequência obtemos assim um espectro que representa a magnitude da presença de determinadas frequências no sinal sonoro. No entanto estes espectros não contêm informação temporal para resolver este problema dividimos o sinal em “windows”(janelas)e aplicamos a transformada de Fourier a cada uma desta janelas , obtemos assim espectogramas que relacionam a frequência, tempo e magnitude[5].

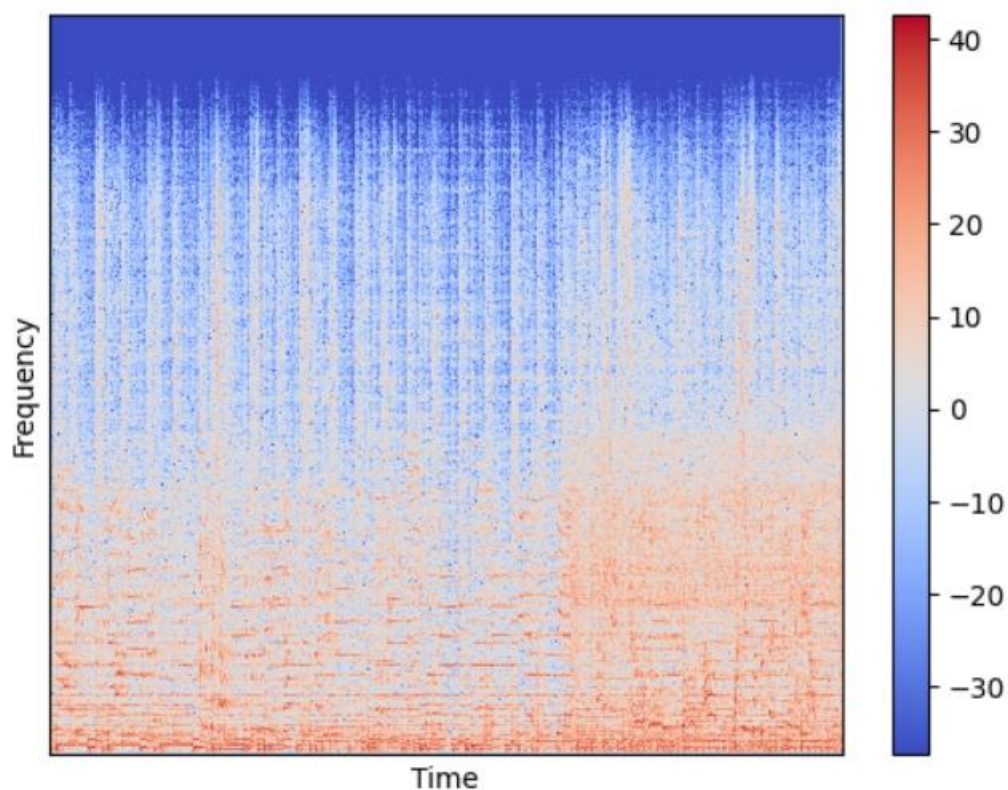


Figura 3. espectograma do ficheiro rock 17

Finalmente o ultimo passo do processo de tratamento do ficheiros sonoros é a obtenção da feature desejada isto é os MFCC's ou coeficientes cepstrais de mel , pretendemos obter esta feature pois permite obter resultados semelhantes ao sistema de percepção vocal humano.

MFCC's representam fenómenos distintos de som no cepstro de frequência de mel sendo que a frequência de mel é uma escala que relaciona a frequência de um som do modo como é interpretada pelo ouvido humano com a frequência real do mesmo e cepstro é obtido pela transformada inversa de Fourier do logaritmo da transformada de Fourier de um sinal.

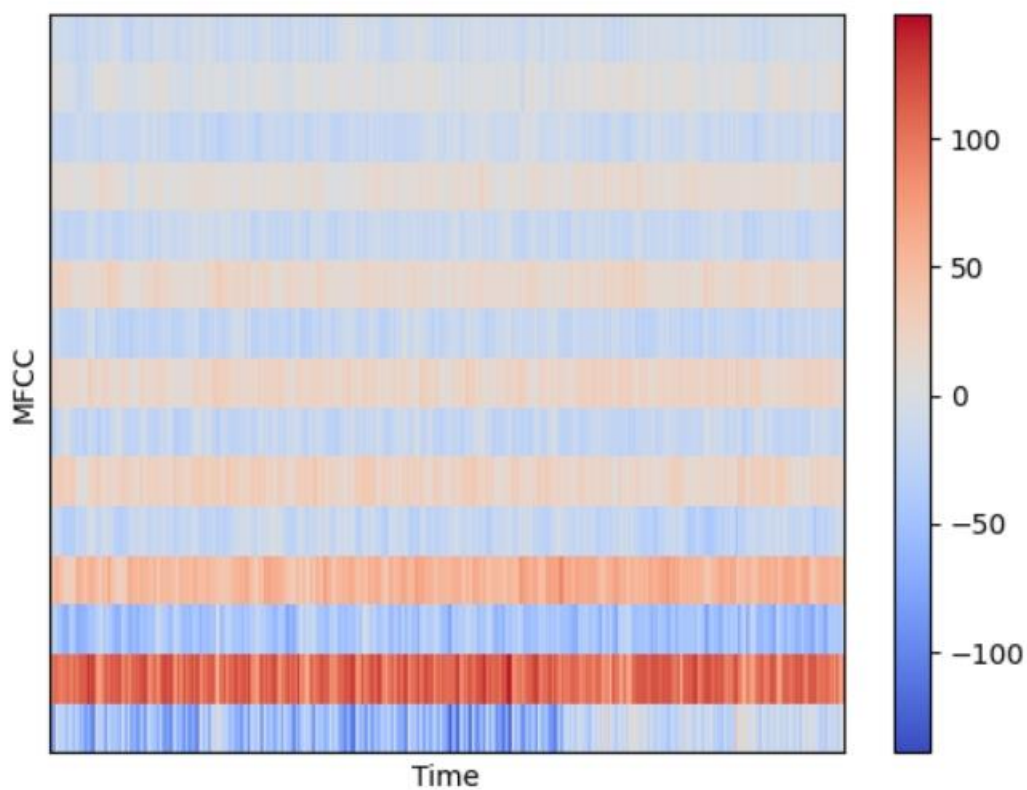


Figura 4. MFCC's do ficheiro rock 17

4.3 Rede Neural

Para procedermos à análise de ficheiros do nosso conjunto de dados usámos um sistema de “deep learning” mais especificamente uma rede neural convolucional implementado com recurso á biblioteca Keras, este sistema é preferível a outros sistemas como por exemplo perceptrons de múltiplas camadas pois permite uma melhor análise de imagens e futuramente a rede neural poderá ser treinada com recurso a imagens dos espectogramas de MFCC's presentes no conjunto de dados GTZAN[3] caso permita uma maior precisão na previsão de géneros musicais ao invés dos valores numéricos dos MFCC's[9].

4.3.1 Rede Neural Convolucional

Neste projecto foi implementado uma versão modificada da CNN utilizada por *Hasan Can Et al.*(2021)[9] com as seguintes camadas:

- Camada Convolucional:
 - 256 *neurónios*;
 - *Tamanho de Kernel : 3x3*;
 - *Activação: relu*;

- Camada Convolucional:
 - 256 *neurónios*;
 - *Tamanho de Kernel : 3x3*;
 - *Activação: relu*;

- Camada de *Pooling*:
 - *Tamanho de Pool: 3x3*;
 - *Strides(avanços): relu*;

- Camada Convolucional:
 - 256 *neurónios*;
 - *Tamanho de Kernel : 3x3*;

- Activação: *relu*;
- Camada de *Pooling*:
 - *Tamanho de Pool: 3x3*;
 - *Strides(avanços): relu*;
- Camada de normalização de lote(*batch*);
- Camada de achatamento;
- Camada Convolutacional:
 - 256 *neurónios*;
 - Ativação: *relu*;
- Camada Convolutacional:
 - 128 *neurónios*;
 - Ativação: *relu*;
- Camada Convolutacional:
 - 10 *neurónios*;
 - Ativação: *softmax*;

As camadas convolucionais funcionam passando filtros sobre os Inputs recebidos;[11]

As camadas de *Pooling* funcionam reduzindo os dados recebidos das camadas convolucionais mas mantendo os valores mais importantes;[11]

A camada de normalização de lote tem por objetivo standardizar os inputs de tal modo que tenham uma média de 0 e desvio padrão de 1;[11]

As camadas densas tem como objetivo a redução outputs até ao número de géneros musicais a classificar nest caso 10.[11]

4.4 Bases de dados

Como prova de conceito é necessário implementar uma base de dados experimental, neste caso foi criada uma base de dados SQL muito rudimentar através do programa MYSQL workbench onde foi criada uma tabela com o seguinte formato :

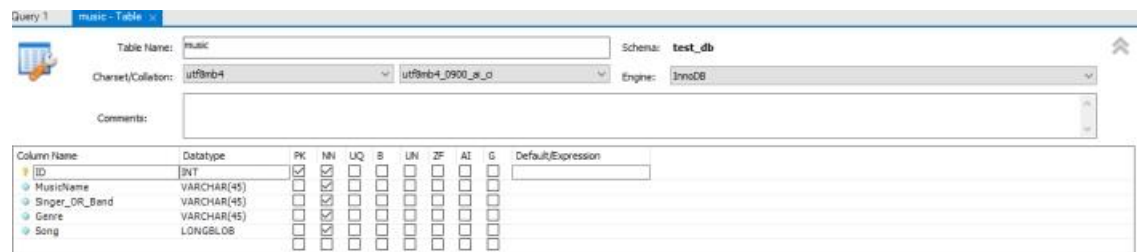


Figura 5. Tabela de testes criada na base de dados

Este tipos de dados foram escolhidos de forma arbitrária por sentir que seriam a escolha mais lógica para o que é pretendido.

Para o armazenamento das musicas foi escolhido fazer a conversão do ficheiro para dados de tipo LongBlob que são dados binários que suporta até 4 GB de informação enquanto que as restantes informações são guardadas como *varchar* por consistirem em apenas texto.

Um alternativa possível de armazenamento dos ficheiros de música seria guardar a sua localização em disco na entrada da base de dados

4.5 Programa Desenvolvido

O programa criado permite a qualquer utilizador usar o modelo de rede neural desenvolvido e fazer uma previsão de género de um ficheiro *mp3* ou *wav* que tenha á sua disposição, o programa permite ainda que o utilizador se ligue a uma tabela de base de dados desde que possua o url da base de dados o nome da mesma e credenciais válidas.

Após efetuar ligação á base de dados o utilizador poderá escolher qual a tabela da base de dados em que queira fazer uma entrada, o programa consegue ainda “ler” as colunas da tabela e os seus tipos de dados e permitir ao utilizador fazer uma entrada de forma dinâmica desde que os tipos de dados introduzidos correspondam aos da coluna correspondente. Neste momento o programa só consegue trabalhar com dados do tipo varchar, int e Blob um potencial objectivo numa iteração futura do trabalho seria adicionar compatibilidade com mais tipos de bases de dados e diferentes tipos de dados.

É de salientar que o programa também tem uma interface gráfica relativamente simples criada com recurso á biblioteca Tkinter que também poderia ser um ponto a melhorar em futuras iterações

TFC

C:/Users/eduar/Desktop/TFC/Data/genres_original/jazz/jazz.00053.wav

The song is classified as: jazz

select an audio file

classify on genre

DB Endpoint

localhost

DB Name

test_db

Username e Password

root

Connect

music

ID

MusicName

Singer_OR_Band

Genre

Song

mudar o ficheiro

Make database entry

Figura 6. Funcionamento da aplicação desenvolvida

5 Benchmarking

Neste capítulo fazemos uma comparação com a *concorrência* e outros trabalhos já existentes que possam ter o mesmo tema.

Após ter sido efetuada uma pesquisa foi possível denotar a existência de vários trabalhos já integrados nesta temática com diferentes abordagens inclusive alguns com já alguns anos o que torna a inovação de métodos algo complicada.

Um exemplo muito semelhante ao que estava planeado inicialmente para este projeto é o trabalho “Features for the Classification and Clustering of Music in Symbolic Format”[13] que estuda a classificação de música em apenas 4 géneros musicais classificados como “melodia, harmonia, baixo e bateria” com recurso a machine learning.

Um estudo bastante semelhante a este no que toca á metodologia foi “Automatic music genre classification and it’s relation with music education”[9] que inclusive foi publicado durante o desenvolvimento deste projecto.

Um estudo que efetua uma abordagem bastante diferente está intitulado “CLASSIFYING MUSIC BY GENRE USING THE WAVELET PACKET TRANSFORM AND A ROUND-ROBIN ENSEMBLE”[12] que usa a transformada de wavelet discreta e uma classificação round-robin para determinar o género de música a partir de um ficheiro de som.

No entanto este trabalho separa-se dos restantes pela integração de um sistema de comunicação com bases de dados juntamente com o sistema de classificação de géneros algo que não me foi possível encontrar disponível no mercado.

6 Métodos e planeamento

Neste capítulo irei explicar o meu método de trabalho e o planeamento.

Inicialmente abordei este trabalho com um nível de conhecimento que não se adequava corretamente inclusive no calendário proposto inclui objetivo que se vieram a provar irrelevantes para o projeto desenvolvido.

Outro problema demonstrado no desenvolvimento deste trabalho foi a falta de comunicação da minha parte com o meu orientador, por motivos que não irão aqui ser elaborados.

Na primeira iteração deste trabalho submetido demonstrava muito pouco conhecimento sobre os temas abrangidos nomeadamente tratamento de som e redes neuronais algo que eventualmente consegui ultrapassar mesmo que por muito pouco.

Sinto ainda que infelizmente não consegui cumprir a maioria dos objetivos presentes no calendário de trabalho devido à minha própria ética de trabalho que infelizmente deixou muito a desejar algo que infelizmente se manteve á iteração final do trabalho.

7 Resultados

Neste capítulo irei apresentar os resultados derivados do desenvolvimento deste trabalho.

O principal resultado deste trabalho foi a pequena aplicação que poderá servir como prova de conceito para trabalhos futuros

No entanto no que toca aos requisitos cumpridos mesmo que parcialmente obtivemos os seguintes resultados:

- **Input de ficheiros na aplicação-** requisito foi cumprido no aspeto que a aplicação consegue aceitar ficheiros do tipo .mp3 e .wav para classificação
- **Análise de ficheiros** -Mesmo que limitado a ficheiros do tipo .mp3 e .wav a aplicação consegue proceder á classificação de qualquer ficheiro de som submetido
- **Input de dados numa base de dados-** Requisito foi cumprido mesmo que parcialmente permitindo ao utilizador fazer entradas em bases de dados de tipo SQL
- **Navegabilidade-** A interface de utilizador acabou por ficar com um *design* bastante simples que facilita a navegação do utilizador
- **Garantir a integridade de dados-** O requisito foi cumprido parcialmente pois neste momento a aplicação apenas garante esta integridade para dados do tipo “varchar”, “Int” e “Blob”

8 Calendário

Estudo sobre Teoria Musical:

-Início: 1 de Dezembro de 2020;

-Fim:20 de Dezembro de 2020;

Criação de diagramas:

-Início:21 de Dezembro de 2020;

-Fim:1 de Janeiro de 2021;

Início do desenvolvimento da aplicação:

-Início:7 de Janeiro de 2021;

-Fim:30 de Março de 2021;

Criação de um base de dados:

-Início:1 de Abril de 2021;

-Fim:13 de Abril de 2021;

Integração da aplicação e base de dados:

-Início:14 de Abril de 2021;

-Fim:6 de Maio de 2021;

Entregas:

-1º relatório intermédio: 22 de Janeiro de 2021;

-relatório intercalar de 2º semestre: 23 de Maio de 2021;

-Entrega final: 23 de Julho de 2021;

9 Conclusão

Sinto que este trabalho poderia ter sido muito melhor realizado especialmente no entanto sinto que cumpri o objetivo inicial mesmo que não tenha sido da melhor forma.

Este trabalho no seu estado final terá valor apenas como prova de conceito de que o que é proposto é possível e que para ter viabilidade de mercado seria necessário desenvolvimento por alguém com mais experiência na área

No entanto sinto que também foi uma oportunidade de aprendizagem pois trabalhei com tecnologias que nunca tinha explorado anteriormente nomeadamente a API de deep learning keras e a linguagem Python em si que até ao início deste semestre não tinha utilizado, sinto no entanto que a maior dificuldade encontrada neste trabalho foi os conceitos em volta da análise de som que mesmo agora no final do trabalho não sinto que estejam completamente integrados.

Bibliografia

- [1] Pt.wikipedia.org, 2021, Rede neural convolucional, [online] available at:
https://pt.wikipedia.org/wiki/Rede_neural_convolucional
- [2] Pt.wikipedia.org, 2021, Mel-frequency cepstrum, [online] available at:
https://en.wikipedia.org/wiki/Mel-frequency_cepstrum
- [3] kaggle.com, 2021, GTZAN dataset, [online], available at:
<https://www.kaggle.com/andradaolteanu/gtzan-dataset-music-genre-classification>
- [4] craft.co, 2021, echoNest, [online], available at:
https://en.wikipedia.org/wiki/The_Echo_Nest
- [5] medium.com, 2021, MFCC [online] available at:
<https://medium.com/prathena/the-dummys-guide-to-mfcc-aceab2450fd>
- [6] Pt.wikipedia.org, 2021, Multilayer Perceptron, [online] available at:
https://en.wikipedia.org/wiki/Multilayer_perceptron
- [7] midiaresearch.com, 2021, Market Shares, [online] available at:
<https://www.midiaresearch.com/blog/global-music-subscriber-market-shares-q1-2021>
- [8] researchgate.net, 2021 MFCC Study [online] available at:
https://www.researchgate.net/publication/272620724_Music_Classification_based_on_MFCC_Variants_and_Amplitude_Variation_Pattern_A_Hierarchical_Approach
- [9] researchgate.net, 2021, hasan et al, [online] available at:
https://www.researchgate.net/publication/350947333_Automatic_Music_Genre_Classification_and_Its_Relation_with_Music_Education
- [10] musicbusinessworldwide.com, 2021, echo nest price [online] available at:
<https://www.musicbusinessworldwide.com/spotify-acquired-echo-nest-just-e50m/>
- [11] keras.io, 2021, keras documentation [online] available at:
<https://keras.io/api/>
- [12] music.mcgill.ca, 2021, Marco et al [online], available at:
http://www.music.mcgill.ca/~ich/classes/mumt611_08/similarity/GrimaldiGenre.pdf
- [13] ul.pt, 2021, Bernardo et al [online], available at:
<https://repositorio.ul.pt/handle/10451/5138>

Glossário

LEI Licenciatura em Engenharia Informática

LIG Licenciatura em Informática de Gestão

TFC Trabalho Final de Curso