NATIONAL CENTRE FOR
SCIENTIFIC RESEARCH "DEMOKRITOS"

Deep learning music information retrieval for content-based movie recommendation

Master Thesis

KARYDIS G. ATHANASIOS
dsc17008@uop.gr

Institute of Informatics & Telecommunications
NCSR Demokritos

Supervisor:
Dr. Theodoros Giannakopoulos

September 14, 2020

Student: ______________________________

Karydis Athanasios

Approved: ______________________________

Supervisor: Theodoros Giannakopoulos, Post-Doctoral Researcher

Approved: ______________________________

Examiner: Klampanos Iraklis - Angelos, Post-Doctoral Researcher

Approved: ______________________________

Examiner: Moscholios Ioannis, Associate professor

Acknowledgements

I would like to thank my supervisor Dr. Theodoros Giannakopoulos for his dedicated support and guidance. Theodoros continuously provided encouragement and was always willing and enthusiastic to assist in any way while his insight and knowledge into audio analysis steered me through this research.

Abstract

A recommendation system is a system that provides suggestions to users for certain resources like movies. Movie recommendation systems usually predict what movies a user would like based on the attributes present in previously liked movies. Content-based movie recommendation systems are important because they base their recommendations mostly on the content of the movies such as music, speech and subtitles. Music, in particular, is an import aspect of a movie, as, in many cases, it may reach the audience emotionally beyond the ability of picture and sound.

In this thesis, both machine learning and deep learning are used to extract valuable information from movie music, in order to create a good base for a content-based recommendation system. Towards the end, a dataset is built containing the predicted values of the aforementioned information of 145 movies of 12 different directors and a collection of metadata.

At a first stage, a simple music detector based on SVM classifier and hand crafted features is trained and applied to movies in order to detect music segments of each movie. Then 4 deep learning models are trained to classify 4 music attributes, namely: danceability, valence, energy and musical genres. In addition, an interactive web application was created with dash Python web framework and plotly for data investigation and a series of plots were produced through a mining procedure in the aforementioned dataset.

All the developed methods described in this thesis are openly available in the following GitHub Repo `https://github.com/thkaridis/MScThesis`.

Contents

Chapter 1
Introduction

1.1 Problem Definition

In recent decades, movie industry is getting more and more prosperous. Therefore, movie recommender system[1] has become more and more popular as a research topic. Those systems are based commonly on the user's feedback on movies, ignoring content features. Although the aforementioned strategy is successful, it suffers from various problems such as the sparsity problem (SP) [2], the cold start problem (CSP)[2] and the popularity-bias problem (PBP)[2]. SP problem refers to poor recommendations which is due to system's lack of meta information on either the items or the users. In addition, CSP is a special case of SP. More specifically, it refers to newly added movies or new users that recently joined and for which the system has no meta information at all. Finally, PBP problem refers to the fact that the system tends to recommend frequently the most popular items, while unpopular or unknown are not recommended at all.

Content-based models[3] are the answer to the aforementioned problems. Those models can be characterised as a very promising approach, even though identifying good representation of items is not that trivial. Through the past years, several techniques have been used in order to improve the content representation of items in content-based recommender systems. It is clear that deep learning and more specifically Convolutional Neural Networks are gaining ground in the battle for better representations to be extracted.

1.2 Related Work

Recommendation is a hot topic in various application areas and not restricted only to music. Some very well-known recommender systems[4] [5] are those used at Netflix.com[6] and at Amazon.com[7]. There are three different approaches[8] towards recommendation systems, content based filtering[2] [3], collaborative filtering[9] [2], and finally the hybrid approaches [10] [2], which provide more accurate and effective recommendations.

Concerning recommendation systems, there have been a great number of studies that focus on different approaches of recommendations such as multimodal video recommendation [11] [12], multimodal emotion classification [13], content analysis based on multimodal features [14]. Nevertheless, most of the research is based mainly on collaborative filtering [15]. Also there are studies that focus on specific modalities of a movie such as gender representation [16] [17], text [18], emotion extraction [19]. Furthermore, there are several approaches that based only on visual features [20]. DeepRecVis, is a novel approach that represents items through features extracted from key-frames of the movie trailers, leveraging these features in a content-based recommender system. [21] In addition, a multi-modal content-based movie recommender system that replaces human-generated metadata with content descriptions automatically extracted from the visual and audio channels of a video is proposed. [22]

Concerning music recommendation there is a very interesting work that makes use of Convolutional Neural Networks to predict attributes from music audio signals [23], while there is a really good overview concerning the state of the art of employing deep learning for music recommendation. [24]. Also, a Fully Content-based Movie Recommender System which trains a neural network model, Word2Vec CBOW, with content information and takes advantage of the linear relationship of learned feature to calculate the similarity between movies seems quite interesting. [25] In addition, a lot of hybrid approaches of recommender systems exist based on social movie networks or the enhancement of collaborative filtering with sentiment classification of movie reviews [26] [27]. Moreover, there is a very interesting work that demonstrates the extraction of multimodal representation models of movies, based on textual information from subtitles combined with audio and visual channels [28].

Social recommender systems are also a very hot topic towards recommendation systems. More specifically, Graph Neural Networks provide great potential to advance social recommendation. [29] Concerning music classification, a novel method of using the spiking neural networks and the electroencephalograph processing techniques to recognize emotion states is proposed. [30]. A really fresh idea concerning music emotions is using triplet neural networks to the regression task of music emotion prediction. [31]

Considering all the above, the domain of recommendation systems has a great number of different research topics. As a consequence, only a small subset of all the related was presented.

1.3 Project Overview

The main goal of the current research is to take as input a collection of movies and create an enhanced dataset that combines metadata of movies, extracted from IMDB[32] and predictions of relevant music features[33] exported from the music parts of those movies. The produced dataset can be used as a good base for a content-based movie recommendation system. Also, the system will try to reveal any correlations between music data, metadata and the directors of the movies. In the following paragraphs, a high level overview will be presented in order to clarify both the main tasks and the techniques used within this project.

First, as mentioned before one of the main tasks of this project is the collection of metadata of all movies for which the investigation is going to take place. So, a scraper has been developed in order to collect all the appropriate information from IMDB[32], one of the world's most popular and authoritative source for movies that contains a great variety of information concerning movies such as user ratings, movies' genres, actors, locations, directors and so on.

The next goal concerns the identification of the parts of a movie that contain music. This step is crucial considering the fact that the audio features will be extracted only by the music segments of a movie. So, a Support Vector Machine classifier[34] will be trained in order to be able to characterise each second of the audio of a movie as Music, Speech or Others. For this purpose the external library pyAudioAnalysis[35], a Python library for audio feature extraction, classification, segmentation and applications, will be used.

Considering that all music parts of a movie can be traced using the SVM classifier, the next goal is to choose the most representative items from a great variety of features that spotify provides. A wide research concluded that the most representatives features for the current classification task are danceability, valence and energy which describe how suitable a track is for dancing based on a combination of musical elements, a perceptual measure of intensity and activity and the musical positiveness conveyed by a track respectively[33]. In addition, genre of each movie will be taken into consideration. So, for each one of those features, a classifier will be trained using Mel-Spectrograms and Convolutional Neural Networks. The classifier will be able to predict a value for each one of the features for all the movies under investigation.

After training all the classifiers above, the system has all the tools needed to take a collection of movies as input, find the music segments of each movie and predict a value for each one of the selected music attributes for each of those segments. Continuing, aggregations of segments' results of each movie will take place in order to export a total result for each of the movies. The main tasks described above can be seen in the following conceptual diagram.

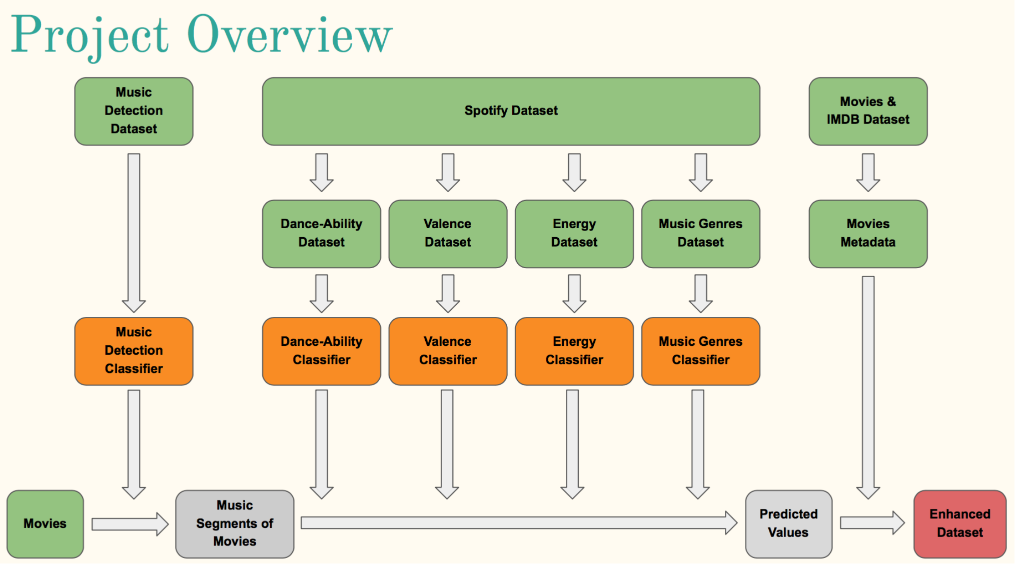


Figure 1: Enhanced Dataset - Conceptual Diagram

All those results are visualised in a dashboard application, created by Plotly[36] and the Python framework Dash[37] which is built on top of Flask, Plotly.js, and React.js.

Last but not least, mining techniques will be performed to the combination of metadata and music features predicted by the system (enhanced dataset) in order to extract correlations between selected features and investigate whether directors can be visually separated taking into consideration both the aggregated results of their movies and their metadata.

Chapter 2

Datasets

In this chapter all datasets that are going to be used for this current research will be presented. A conceptual diagram is presented to make an introduction to the reader concerning the content of the aforementioned datasets.

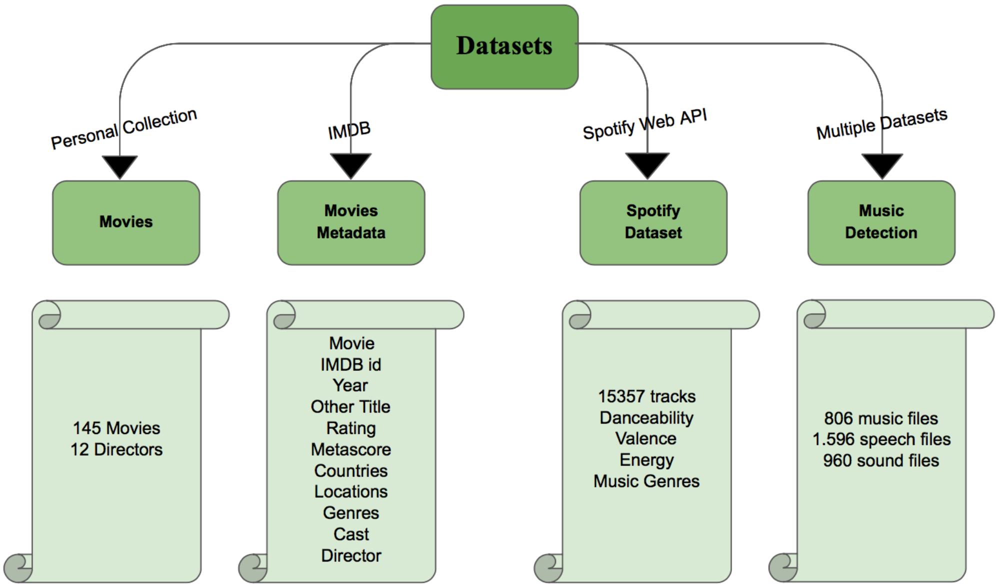


Figure 1: Datasets - Conceptual Diagram

2.1 Movies Dataset

This dataset is actually a personal collection of 145 movies from 12 different directors, that can be seen in the following table.

Movies Dataset	
Director	Movies
Alfred Hitchcock	Dial M For Murder, Family Plot, Foreign Correspondent, Frenzy, I Confess, Jamaica Inn, Lifeboat, Mr Mrs Smith, Rear Window, Rope, Saboteur, Shadow of a Doubt, Spellbound, Stage Fright, Strangers on a Train, Suspicion, The 39 Steps, The Lady Vanishes, The Man Who Knew Too Much, The Paradine Case, The Trouble With Harry, The Wrong Man, To Catch a Thief, Under Capricorn, Vertigo, Young and Innocent
Christopher Nolan	Batman Begins, Inception, Insomnia, Interstellar, The Dark Knight Rises, The Prestige
Clint Eastwood	Pale Rider, Bloodwork, Gran Torino, High Plains Drifter, Play Misty For Me, The Gauntlet, The Outlaw Josey Wales, A Perfect World, Absolute Power, Heartbreak Ridge, Million Dollar Baby, Space Cowboys, Sudden Impact, Unforgiven, White Hunter Black Heart, Firefox, The Rookie
Coen Brothers	Barton Fink, Fargo, Miller's Crossing, No Country For Old Men, Raising Arizona, The Big Lebowski, The Ladykillers
Francis Ford Coppola	The Godfather Part II, The Godfather Part III, The Godfather
George Lucas	Star Wars Episode I The Phantom Menace, Star Wars Episode II Attack of the Clones, Star Wars Episode III Revenge of the Sith, Star Wars Episode IV A New Hope
Pedro Almodovar	Dark Habits, Kika, La flor de mi secreto, La ley del deseo (Law of Desire), La piel que habito, Laberinto de pasiones, Live Flesh, Los abrazos rotos, Los amantes pasajeros, Matador, Mujeres al borde de un ataque de nervios, Que he hecho yo para merecer esto, Tacones lejanos, Talk to Her, Tie Me Up Tie Me Down, Todo sobre mi madre
Peter Jackson	The hobbit An unexpected journey, The hobbit The battle of the five armies, The hobbit The desolation of Smaug, The Lord of the Rings The Fellowship Of The Ring, The Lord of the Rings The Return Of the king, The Lord of the Rings The Two Towers

Polanski	Cul de Sac, Repulsion, The Fearless Vampire Killers, The Tenant
Quentin Tarantino	Death Proof, Django Unchained, Inglourious Basterds, Jackie Brown, Kill Bill 1, Kill Bill 2, Pulp Fiction, Reservoir Dogs
Tim Burton	Alice in Wonderland, Batman Returns, Batman, Beetlejuice, Big Eyes, Corpse Bride, Dark Shadows, Edward Scissorhands, Frankenweenie, Sweeney Todd The Demon Barber Of Fleet Street
Woody Allen	A Midsummer Night's Sex Comedy, Annie Hall, Another Woman, Bananas, Broadway Danny Rose, Bullets Over Broadway, Cassandra's Dream, Celebrity, Crimes and Misdemeanors, Deconstructing Harry, Everything You Always Wanted To Know About Sex But Were Afraid to Ask, Hannah And Her Sisters, Hollywood Ending, Husbands and Wives, Interiors, Love and Death, Manhattan Murder Mystery, Manhattan, Match Point, Melinda and Melinda, Mighty Aphrodite, New York Stories, Radio Days, Scoop, September, Shadows and Fog, Sleeper, Small Time Crooks, Stardust Memories, Sweet and Lowdown, Take the Money and Run, The Curse of the Jade Scorpion, The Purple Rose of Cairo, Vicky Cristina Barcelona, What's Up Tiger Lily, Whatever Works, Zelig

Table 1: Movies Dataset (per Director)

2.2 Movies Metadata Dataset

In this section the procedure that have been followed in order to collect metadata for the movies of the previous section will be described.

2.2.1 IMDB Dataset

IMDB[32] is one of the world's most popular and authoritative source for movies that contains a great variety of information concerning movies such as user ratings, movies' genres, actors, locations, directors and so on. In other words, it can be characterised as a very representative source of movies metadata.

Luckily, subsets of IMDB data[38] are available for access to customers for personal and non-commercial use. More specifically, title.basics.tsv.gz file contains 5885.795 rows. Each row has all the information concerning a movie except from the first row which is the header of the file. The following table shows all properties included in each row of the file and a small description of the type of the data.

title.basics.tsv.gz (IMDB Dataset)		
Properties	Data type	Description
tconst	string	alphanumeric unique identifier of the title
titleType	string	type of the movie (movie, short, tvseries, etc)
primaryTitle	string	the more popular title
originalTitle	string	original title, in the original language
isAdult	boolean	0: non-adult title; 1: adult title
startYear	YYYY	represents the release year of a title.
endYear	YYYY	primary runtime of the title, in minutes
genres	string array	includes up to three genres associated with the title

Table 2: IMDB Datasets' properties[38]

2.2.2 IMDB Scraper

An IMDB scraper is a system that is capable to access IMDB website[32], parse the content of a specific movie web page and write all information needed into an exported file.

First of all the system loads the IMDB dataset[38] described in the previous section, filters only the movie records, erasing all the other and exports onlyMovies.csv file. This file has no header line and contains a movie in each line with all the information included in the parent dataset, separated by the character ",". Also, the user has to define the path that contains all movies for investigation separated by director's name e.g. */movieForThesis/Alfred Hitchcock/.

The system finds all movies' titles from the aforementioned path, performs cleaning of the titles and checks if those titles exist in the "primaryTitle", or "originalTitle" columns of the produced onlyMovies.csv dataset. In case the movie title exists, the alphanumeric unique identifier of the title "tconst" is used by the system in order to access the link http://www.imdb.com/tconst and begin parsing the HTML page of the movie using BeautifulSoup[39], a Python library for pulling data out of HTML and XML files. The same procedure is followed for all movies and the system finally exports the metadata dataset in both csv and json form. In both files each movie contains the information mentioned in the following table.

Movie	IMDB id	IMDB link
Other Title	Year	Rating
Metascore	Countries	Locations
Genres	Cast	Director

Table 3: Metadata Datasets' properties

The implementation of IMDB Scrapper (collectMoviesMetadata.py) can be found in `https://github.com/thkaridis/MScThesis/tree/master/moviesMetadata`. In order to run the python file just follow the steps described inside the instructions.txt file.

2.3 Music Analysis Data

Spotify[40] is a well known application that provides an easy way to find the right music or podcast for every moment on a great variety of electronic devices such as smartphones, tablets, laptops etc. The most impressive about spotify is that it contains a great variety of audio attributes for each song included in the platform.

More specifically, there are 12 audio characteristics for each track, including confidence measures like acousticness, liveness, speechiness and instrumentalness, perceptual measures like energy, loudness, danceability and valence and descriptors like duration, tempo, key, and mode[33]. After a wide research in those attributes, we concluded that the most representative for the current classification task are Danceability, Valence and Energy. In addition to the aforementioned features, music genre will be one of the features for investigation. In the following table you may find a small description for each one of those features.

Audio Attributes	
Audio Attribute	Description
danceability	Describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. The value can be between 0.0 and 1.0 where a value of 0.0 is least danceable and 1.0 is most danceable.
energy	It ranges from 0.0 to 1.0 and represents a measure of intensity. More specifically, the songs that are more energetic, feel fast, loud, and noisy.

valence	It ranges from 0.0 to 1.0 and describes the positiveness or negativeness of a song. High valence means more positive while songs with low valence sound more negative, depressed and angry.
genres	string that describes the music type of each track.

Table 4: Audio Attributes[33]

Considering the fact that the number of genres is really high, it is imperative to group them into a smaller number of hyper genres. In order to succeed in this task all genres were passed through a python implementation that makes use of K-Means clustering[34], one of the simplest and popular unsupervised machine learning algorithms.

A cluster consists of data points gathered together because of certain similarities. The user defines a target number k, which refers to the number of centroids needed. Every data point is allocated to each of the clusters through reducing the in-cluster sum of squares. In other words, the K-means algorithm identifies k number of centroids, and then allocates every data point to the nearest cluster, while keeping the centroids as small as possible.

The result of the aforementioned algorithm maps each genre to one of the 10 hyper genres. It has to be mentioned that the result was reviewed and some changes were made manually towards result's improvement. The results can be seen in the following table.

Music Genre	Sub - Genres
Blues	blues-rock, memphis blues, louisiana blues, electric blues, modern blues, traditional blues, soul blues, new orleans blues, piedmont blues, jump blues, country blues, british blues, harmonica blues, delta blues, chicago blues, blues, piano blues, acoustic blues
Classical	classical, modern classical, hindustani classical, indian classical, classical period, baroque, classical flute, classical piano, concert piano
Country	country rock, country gospel, gospel, gospel reggae, country, country road, alternative country, outlaw country, contemporary country, country dawn, folk, appalachian folk, arab folk, freak folk, folk-pop, corsican folk, indie anthem-folk, british folk, indie folk, folk christmas, anti-folk, ectofolk, cowboy western, new americana

Rock	folk rock, dance rock, rock steady, experimental rock, psychedelic rock, indie rock, norwegian rock, boston rock, symphonic rock, industrial rock, pop rock, hungarian rock, belgian rock, suomi rock, lovers rock, rock, kiwi rock, roots rock, progressive rock, sleaze rock, glam rock, modern country rock, classic funk rock, comedy rock, art rock, irish rock, swedish hard rock, turkish rock, album rock, rap rock, soft rock, classic garage rock, garage rock, french rock, australian alternative rock, swedish alternative rock, celtic rock, crack rock steady, wedish indie rock, stoner rock, classic rock, math rock, hard rock, funk rock, protopunk, danspunk, punk, skate punk, emo punk, german punk, brazilian punk, garage punk, power-pop punk, ska punk, horror punk, folk punk, french punk, celtic punk, punk christmas, deep surf music, surf music
Metal	dark black metal, german metal, pagan black metal, gothic metal, chaotic black metal, melodic death metal, symphonic black metal, industrial metal, canadian metal, avantgarde metal, doom metal, cyber metal, groove metal, gothic symphonic metal, neo-trad metal, alternative metal, christian metal, jazz metal, glam metal, speed metal, finnish metal, black metal, metal, progressive metal, neo classical metal, nu metal, rap metal, atmospheric black metal, brutal death metal, power metal, funk metal, death metal, folk metal, crust punk, death core, anarcho-punk, hardcore punk, gothic americana, black thrash, thrash core, crossover thrash
Pop	swedish pop, teen pop, swedish folk pop, bubblegum pop, arab pop, pop, danish pop, deep brazilian pop, canadian pop, deep indian pop, viral pop, stomp pop, australian pop, french pop, turkish pop, classic swedish pop, classic norwegian pop, deep german pop rock, mande pop, pop punk, pop rap, classic venezuelan pop, russian pop, post-teen pop, dance pop, latin pop, hip pop, acoustic pop, underground pop rap, pop christmas, french indie pop, classic turkish pop, german pop, candy pop, classic belgian pop, bow pop, brill building pop, indian pop, deep pop r&b, new wave pop, norwegian pop, italian pop, grunge pop, alternative pop, polynesian pop, chamber pop, indie pop, britpop, europop
Dance	disco, italian disco, deep italo disco, post-disco, deep disco, nu disco, rock-and-roll, new jack swing, traditional swing, electro swing, salsa, deep eurodance, australian dance, full on, dancehall, intelligent dance music, eurodance, swedish eurodance, italo dance, alternative dance, bubblegum dance, belly dance, latin christmas, flamenco, latin, tango, rockabilly, samba, mambo

Jazz	latin jazz, new orleans jazz, jazz blues, smooth jazz, jazz christmas, electro jazz, vocal jazz, gypsy jazz, jazz funk, jazz bass, contemporary jazz, finnish jazz, acid jazz, swedish jazz, jazz, dark jazz, nu jazz, indie jazz, turkish jazz, soul jazz, cool jazz, northern soul, neo soul, soul christmas, memphis soul, chicago soul, soul, big room, deep big room, neurofunk, deep funk, funk, g funk, liquid funk, cumbia funk, p funk, baile funk, uk funky, funky breaks, japanese jazztronica, hard bop
Hiphop	abstract hip hop, polish hip hop, german hip hop, swedish hip hop, austrian hip hop, southern hip hop, desi hip hop, alternative hip hop, detroit hip hop, hip-hop, hardcore hip hop, hip hop, deep swedish hip hop, christian hip hop, old school hip hop, east coast hip hop, gangster rap, underground rap, dirty south rap, rap, chicano rap, west coast rap, alternative r&b, indie r&b, r&b, deep indie r&b, chillhop
Electronic	new wave, deep new wave, uk dub, dub, dubstep, minimal dub, psychedelic trance, deep uplifting trance, progressive trance house, uplifting trance, bubble trance, trance, progressive trance, progressive uplifting trance, deep progressive trance, electro, electro dub, retro electro, electronic, vintage french electronic, latin electronica, electronic trap, tribal house, progressive electro house, deep soul house, deep groove house, tropical house, deep house, deep euro house, pop house, disco house, dutch house, electro house, minimal tech house, float house, deep tropical house, vocal house, deep melodic euro house, acid house, tech house, progressive house, hard house, deep disco house, filter house, chicago house, funky tech house, house, hip house, minimal techno, german techno, destroy techno, detroit techno, deep minimal techno, acid techno, hardcore techno, techno, bass trap, deep trap, deep southern trap, trap music, west coast trap, dwn trap, progressive psytrance, trip hop, steampunk, french reggae, roots reggae, reggae, traditional reggae, polish reggae, skinhead reggae, reggae fusion, microhouse, progressive deathcore, electroclash, freestyle, drum and bass, breakbeat, deep chill-out, ska

Table 5: Genres & Sub-genres mapper

2.3.1 Spotify Dataset

In order to create spotify dataset, music files from a personal collection have been used. Moreover, metadata for the aforementioned music files have been gathered through Spotify Web API[41]. More specifically, Spotify Web API returns metadata of desirable tracks enclosed in a JSON file concerning music artists, albums, and tracks, directly from the Spotify Data Catalogue.

The handmade spotify dataset contains 15357 tracks and a variety of metadata from Spotify Web API some of which can be seen in the following figure.

Spotify Dataset		
Audio Attributes	Data type	Example
spotify-duration (ms)	integer	406013
spotify-artistName	string	Johann Sebastian Bach
spotify-mode	integer	1
spotify-albumName	string	Bach: Six Concertos ...
spotify-loudness	float	-14.237
spotify-timesignature	integer	3
spotify-valence	float	0.621
spotify-energy	float	0.292
spotify-instrumentaln	float	0.00034
spotify-liveness	float	0.0947
spotify-acousticness	float	0.895
spotify-speechiness	float	0.0413
spotify-trackName	string	Brandenburg Concerto No. 4
spotify-key	integer	6
spotify-danceability	float	0.459
spotify-genre04	string	romantic
spotify-genre03	string	classical christmas
spotify-genre02	string	classical
spotify-genre01	string	baroque
spotify-genre05	string	
spotify-tempo	float	96.407

Table 6: Selected attributes of Spotify Dataset

From the produced dataset, the attributes that are going to be used in the next steps are: spotify-duration, spotify-valence, spotify-energy, spotify-danceability and spotify-genre0x (where x ranges from 1 to 5).

2.3.2 Balanced Datasets Creation

The spotify dataset will be used to train a classifier for each of the attributes selected (danceability, valence, energy, genres). It is clear that spotify dataset is not balanced concerning each feature. As a consequence, for each one of the attributes, a balanced subset of volume of approximately 1000 instances from the whole dataset will be created in order to perform the training procedure.

The 70% of the records will be used for training while 30% of them for testing purposes.

Let's see the procedure followed in order to create a balance dataset for danceability. The first step is to filter Spotify dataset and keep only records that contain spotify-danceability and at the same time spotify-duration exceeds the limit of 2 minutes. Continuing, the selected records will be shuffled and divided into two equal parts. From the first half train dataset will be produced while the second one will generate the test set. The aforementioned division into two parts ensures that there will be no track containing in both train and test sets and avoids overfitting of Convolution Neural Network in a specific song. Both train and test will be created taking into consideration the danceability values.

As already described, danceablility, valence and energy values range from 0.0 to 1.0. Consequently, the fact that classification and not regression will be performed to the dataset, the values of the aforementioned features can not be continuous floats. As a result an integer (0, 1, 2) will be assigned to each one of the tracks for each one of the features. Obviously, the tracks that have danceability in range [0, 0.33), "0"value will be assigned, those with range [0.33, 0.66] will take the value "1" and "2" will be assigned to the rest. In case of genres, each genre will be represented as a integer: 'blues': 0, 'rock': 1, 'pop': 2, 'metal': 3, 'jazz': 4,'classical': 5, 'electronic': 6, 'hiphop': 7, 'country': 8, 'dance': 9. So, at this point a train dataset of volume of 700 tracks and a test dataset of volume of 300 tracks have been created. Those datasets are balanced in respect to danceability feature.

The exact procedure is followed for each one of the features (genres, danceability, valence, energy) concluding in 4 balanced datasets.

2.4 Music Detection Dataset

The dataset that will be used for the training of a classifier capable to detect music segments of a movie, contains audio files exported from more than 50 movies[42]. Those files are separated in three folders with regards to the type of sounds they are consisting of. More specifically the three folders or the three classes are Music, Speech and Others. Music contains files that consist of music segments of movies, speech folder contains speech segments and screams from movies and lastly others folder contains movie segments such as shots, fights etc. In order to empower the music class, considering the fact that the purpose of the training classifier is to classify music segments of a movie correctly over the other 2 classes, some music segments from GTZAN Genre Collection Dataset[43] were added to the music folder. As a result the dataset contains 1.806 music files, 1.596 speech files and 960 files of other sounds. A summary of all the above is shown in the following table.

Segments			
Type	Count	Mean Duration(s)	Description
Music	1,806	8.55	music (movies) & songs
Speech	1,596	2.48	speech,screams (movies)
Others	960	2.56	shots, fights (movies)

Table 7: Summary of Music Detection's Dataset

Chapter 3

Music Detection

In this section Music Detection will be performed. Music Detection means that the system is capable of separating music from speech and other audio sounds of an audio file. Considering both the volume of the dataset and that the task is quite simple lead us to use hand-crafted features and Support Vector Machine (SVM) in order to succeed in the aforementioned purpose of classifying each second of an audio file as Music, Speech or Others. The dataset that will be used has been presented in chapter 2.

3.1 Audio Features

Feature extraction is the process that a system extracts audio features from audio files. More specifically the 34 different audio features that will be extracted from each one of the short-term windows or frames are shown in the following table. The short-term and mid-term techniques will be described in chapter 3.1.1.

Audio Features	
Index	Audio Feature
1	Zero Crossing Rate
2	Energy
3	Entropy of Energy
4	Spectral Centroid
5	Spectral Spread
6	Spectral Entropy
7	Spectral Flux
8	Spectral Rolloff
9-21	MFCCs
22-33	Chroma Vector
34	Chroma Deviation

Table 8: Audio Features

Those audio features are divided into two categories, the time-domain features which are directly extracted from the raw signal and the frequency-domain features which are based on the magnitude of the Discrete Fourier Transform (DFT). The time-domain features are Energy, Zero Crossing Rate and Entropy of Energy while all the rest are characterised as frequency-domain features.[44]

Zero Crossing Rate: The rate of sign-changes of the signal during the duration of a particular frame. It can be characterised as the measurement of noisiness. High values of this feature indicates noisy signals.

$$Z(i) = \frac{1}{2W_L} \sum_{n=1}^{W_L} |sgn[x_i(n)] - sgn[x_i(n-1)]|$$

where

$$sgn[x_i(n)] = \left\{ \begin{array}{ll} 1, & x_i(n) \geq 0 \\ -1, & x_i(n) < 0 \end{array} \right\}$$

Energy: The sum of squares of the signal values, normalized by the respective frame length.

$$E(i) = \frac{1}{W_L} \sum_{n=1}^{W_L} |xi(n)|^2$$

Energy Entropy: The entropy of sub-frames' normalized energies. It can be interpreted as a measure of abrupt changes in the signal's energy.

Spectral Centroid: The center of gravity of the spectrum.

$$C_i = \frac{\sum_{k=1}^{Wf_L} (k+1) X_i(k)}{\sum_{k=1}^{Wf_L} X_i(k)}$$

Spectral spread: The second central moment of the spectrum.

$$S_i = \sqrt{\frac{\sum_{k=1}^{Wf_L} ((k+1) - C_i)^2 X_i(k)}{\sum_{k=1}^{Wf_L} X_i(k)}}$$

Spectral entropy: Entropy of the normalized spectral energies for a set of sub-frames. The procedure for spectral entropy calculation is to divide spectrum into L sub-bands, compute normalized sub-band energies (Ef) and finally compute entropy.

$$H = -\sum_{f=0}^{L-1} n_f \cdot log_2(n_f)$$

where

$$n_f = \frac{E_f}{\sum_{f=0}^{L-1} E_f}, f = 0, ..., L-1$$

Spectral Flux: The squared difference between the normalized magnitudes of the spectral of two successive frames.

$$Fl_{(i,i-1)} = \sum_{k=1}^{Wf_L} (EN_i(k) - EN_{i-1}(k))^2$$

where

$$EN_i(k) = \frac{X_i(k)}{\sum_{l=1}^{Wf_L} X_i(l)}$$

Spectral Rolloff: The frequency below which a percentage of the magnitude distribution of the spectrum is concentrated.

Mel-Frequency Cepstral Coefficients (MFCCs): Mel Frequency Cepstral Coefficients form a cepstral representation where the frequency bands are not linear but distributed according to the mel-scale. Usually the first 13 MFCCs are selected as they carry enough discriminative information. The following figure describes the procedure needed to compute cepstrum.

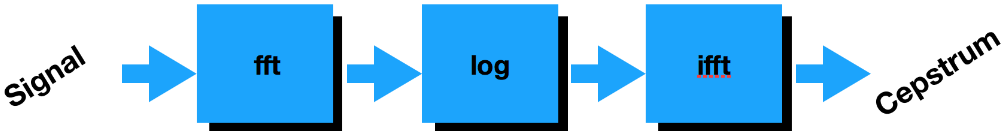


Figure 3: Cepstrum Computation Procedure

Chroma Vector: A 12-element representation of the spectral energy where the bins represent the 12 equal-tempered pitch classes of western-type music.

$$v_k = \sum_{n \in S_k} \frac{X_i(n)}{N_k}, k \in 0...11$$

where Sk is the set of frequencies for the k-th bin (representing DFT coefficients)

Chroma Deviation: The standard deviation of the 12 chroma coefficients.

3.1.1 Mid-term and short-term techniques

In order to compute the features described in the previous chapter there are many different techniques that can be followed. More specifically, mid-term and short-term techniques are mainly used to extract those features. According to the mid-term technique, the audio signal is divided into non overlapping mid-term windows that last 1 second. For each one of mid-term segments, short-term technique will be performed. Short-term basis defines that the audio signal is divided into short-term windows (frames) of 50ms and for each one them, all 34 features are calculated and presented as a feature vector. Therefore, each mid-term segment or each second is presented by 20 feature vectors of 34 features. For each one of those vectors mean and standard deviation are computed for each of the 20 values. The result of the aforementioned procedure is a mid-term statistic feature vector of 68 features. [44].

3.1.2 Support Vector Machine - Implementation

Support vector machines is highly preferred by the majority of the machine learning applications as it produces significant accuracy with less computation power. Abbreviated as SVM can be used to both regression and classification tasks.

The purpose of the support vector machine algorithm is to find a hyperplane or decision boundary in an N-dimensional space, where N is the number of features, that classifies the data points. Considering the fact that there are countless hyperplanes that can achieve this goal, SVM searches for the one that maximizes the margin or the distance between the data and the plane and thus the distance between the data of different class. The dimensions of a hyperplane is defined by the number of features. In one dimension, a hyperplane is called a point. In two dimensions it is clearly a line while in three dimensions it is a plane. While the number of the features is increased, it becomes really difficult to imagine the shape of the plane. The objective of an SVM classifier is to find the optimal separating hyperplane by correctly classifying the training data and it is the one which will generalize better with unseen data[34].

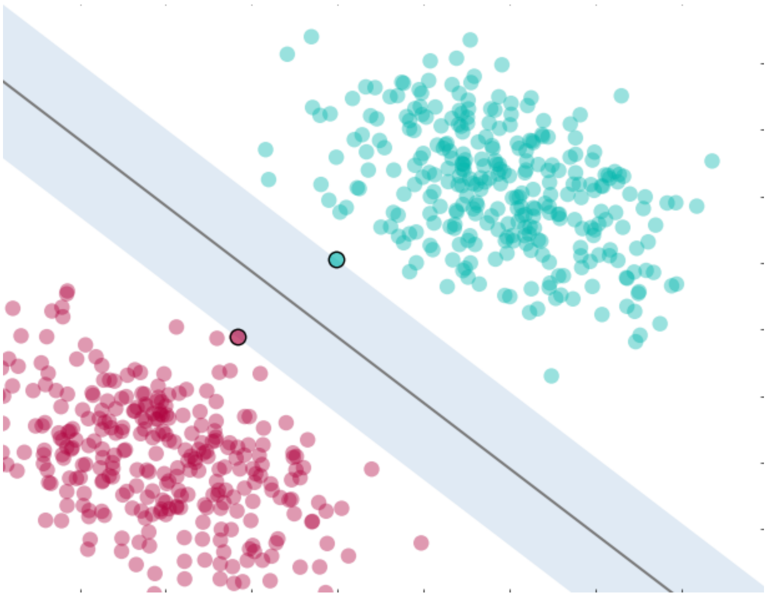


Figure 4: SVM & Hyperplanes[45]

Last but not least, cross-validation procedure will be performed in order to optimise the classifier. During the evaluation step the input consists of arbitrary-sized segments. Suppose that you have as input some segments that last 3, 4 and 8 seconds. Those segments have 3, 4 and 8 statistic mid term vectors of 68 values respectively. For each one of those segments long term averaging of the 68 values of each statistic mid term feature vector will be performed. Finally, F1 score is extracted for each audio class and the best average F1 score is used as criteria for parameter selection.

3.2 Implementation Details

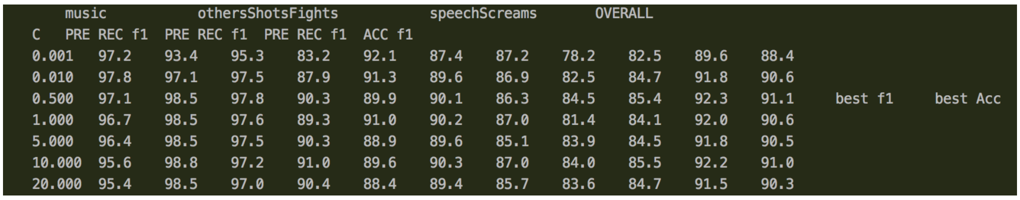
In order to accomplish all the above, the external library called pyAudioAnalysis will be used. PyAudioAnalysis[35] is an Open-Source Python Library for Audio Signal Analysis that provides a wide range of audio analysis procedures such as feature extraction, classification of audio signals, supervised and unsupervised segmentation and content visualization. It has been already used in several research applications concerning audio such as smart-home functionalities through audio event detection, music segmentation, multi modal content-based movie recommendation and others.

3.3 Results

In order to train the classifier that separates Music, Speech and Others for each audio file containing into the dataset, the parameters of pyAudioAnalysis (mid window, mid step, short window, short step) should be defined. Many attempts were performed with different parameters, in order to find the best combination of parameters that leads to the best classifier.

Parameter	Value
Mid Window	1.0
Mid Step	1.0
Short Window	0.05
Short Step	0.05

Table 9: Parameters of feature extraction & SVM training



```
      music           othersShotsFights           speechScreams       OVERALL
   C    PRE REC f1  PRE REC f1  PRE REC f1  ACC f1
   0.001   97.2    93.4    95.3    83.2    92.1    87.4    87.2    78.2    82.5    89.6    88.4
   0.010   97.8    97.1    97.5    87.9    91.3    89.6    86.9    82.5    84.7    91.8    90.6
   0.500   97.1    98.5    97.8    90.3    89.9    90.1    86.3    84.5    85.4    92.3    91.1      best f1      best Acc
   1.000   96.7    98.5    97.6    89.3    91.0    90.2    87.0    81.4    84.1    92.0    90.6
   5.000   96.4    98.5    97.5    90.3    88.9    89.6    85.1    83.9    84.5    91.8    90.5
   10.000  95.6    98.8    97.2    91.0    89.6    90.3    87.0    84.0    85.5    92.2    91.0
   20.000  95.4    98.5    97.0    90.4    88.4    89.4    85.7    83.6    84.7    91.5    90.3
```

Figure 5: SVM F1 score

Confusion Matrix			
	Music	Other	Speech
Music	40.78	0.53	0.11
Other	0.85	32.93	2.84
Speech	0.39	3.02	18.56

Table 10: SVM training - Confusion Matrix

Observing the confusion matrix above, it is clear that the trained classifier predicts correctly and assigns the majority of test audio files (unknown) to the right class.

Chapter 4

Music Information Extraction

As mentioned in section 2.3.2, 4 balanced datasets have been exported from the Spotify dataset. Each dataset contains audio files that will be used as input to Convolutional Neural Networks in order to train 4 different classifiers that will be able to predict danceability, valence, energy and music genre values of a movie's music. In the following sections, the procedure needed to train the aforementioned classifiers will be presented.

4.1 Feature Extraction

WAV files pass through a function that converts their rate to 8kHz and channel stereo to mono. The signal of each one of the files will be loaded through scipy.io.wavFile, a function from scipy software[46] that returns the sample rate (in samples/sec) and data from a WAV file. After a normalisation of the loaded signal, it will be divided in 10-second segments using 80.000 as window length and only the first 80 seconds of the track will be kept for investigation while the rest will be discarded. In other words each track will be splitted in 8 10-second segments. Each audio segment is converted into a spectogram, which is a visual representation of spectrum of frequencies over time. A spectrogram is squared magnitude of the short term Fourier transform (STFT) of the audio signal. Spectrogram is squashed using mel scale to convert the audio frequencies into a representation that conforms psycho-acoustic observations, something a human is more able to understand. The Mel Scale, from the aspect of mathematics, is actually the result of a non-linear transformation of the frequency scale. This procedure is performed through the built in function of librosa library[47]. The parameters concerning the transformation are window length, which indicates the window of time to perform Fourier Transform on and hop length which is the number of samples between successive frames. Finally, Mel Spectrograms produced by librosa are scaled by a log function. As a result of this transformation each 10-second segment audio file is converted to a Mel-Spectrogram of shape (128, 100) where 128 is the number of the Mel Coefficients and the number 100 is the result of the division of 10sec by 0.1 second step. The following figure shows an example of a Mel-Spectrogram produced by a 10-sec segment.

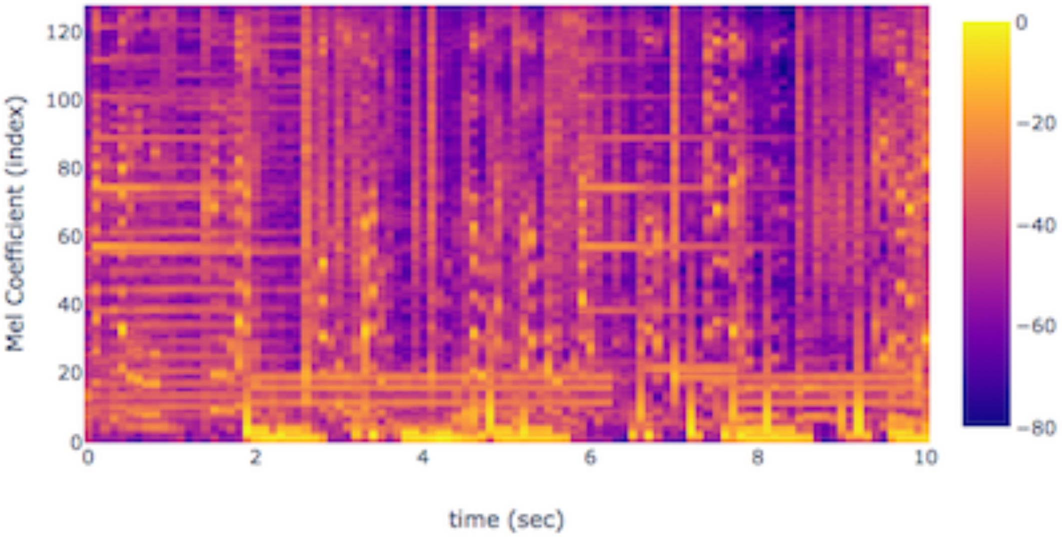


Figure 6: Example of a Mel-Spectrogram produced by a 10-sec segment

4.2 Classification: Training

One question that may arises is why do we need CNNs? Considering that Mel-Spectrogram is a visual representation of audio, and considering the fact that CNNs work efficiently when the input has to do with images, justifies this decision. In this section, the theory behind the Convolutional Neural Networks (CNN)[48] and all the design decisions concerning the CNN used will be described. It has to be mentioned that there is a great variety of choices that could be made to improve the result of the training procedure, but this is not the purpose of this current research. In the next paragraphs, basic principles of Convolutional Neural Networks will be succinctly described.

The Convolutional Neural Network (CNN) is a class of deep learning neural networks, representing a breakthrough concerning image recognition. They're mostly used to analyze images and are also used in image classification. They can be found in a variety of applications, from Facebook's photo tagging to self-driving cars. Their two major advantages are velocity and effectiveness. Technically, each input image will pass through a series of convolution layers with filters, Pooling, fully connected layers and Softmax function to classify a class with probability between 0 and 1.

Convolution is the first layer of a CNN and preserves the relationship between pixels by learning image features using small squares of input data. More specifically, it is a mathematical operation that takes as input an image matrix and a filter or kernel. The convolution of a matrix multiplies with a filter matrix (weight matrix) is called feature map. The aforementioned convolution can perform various operations such as edge detection, noise blurring etc. Weights are learnt such that the loss function is minimized similar to Machine Learning principles. Therefore weights are learnt to extract features from the original image which lead the network to correct predictions. When multiple convolutional layers exist, the initial layer extract more generic features, while as the network gets deeper, the features extracted by the weight matrices are more complex.

One thing to keep in mind is that the depth dimension of the weight would be same as the depth dimension of the input image. The weight extends to the entire depth of the input image. Therefore, convolution with a single weight matrix would result into a convolved output with a single depth dimension. In most cases multiple filters of the same dimensions are applied together. The outputs are stacked together forming the depth dimension of the convolved image.

Pooling layers are used in order to reduce the number of parameters when the images are too large. In other words, this section performs sub sampling or down sampling by reducing the dimensionality and at the same time retaining important information.

Dropout ignores neurons during the training phase of a set of neurons, randomly chosen. More specifically, at each training stage, individual nodes are either dropped out of the net with probability 1-p or kept with probability p, in order to reduce the network.

After multiple layers of convolution and padding, the output should be formed as a class. However, to generate the final output a fully connected layer should be applied with flattened input in order to generate an output equal to the number of classes needed. In addition, there is a great variety of activation functions that can be used to introduce non linearity. One of the most well known is Rectified Linear Units (ReLU), which returns 0 when receiving any negative input and returns the value for any positive one. Mathematically, it can be seen as
$f(x) = \max(0,x)$.

The output layer has a loss function like categorical cross-entropy, to compute the error in prediction. Once the forward pass is completed, the back-propagation begins to update the weights and biases for error and loss reduction. Optimiser such as Stochastic Gradient Descent (SGD) can be used in order to tie together the loss function and model parameters by updating the model in response to the output of the loss function. Optimization is actually a type of searching process called learning. The optimization algorithm is called gradient descent, where gradient refers to the calculation of an error gradient and descent to the moving down along that slope towards some minimum level of error. The batch size is a hyper-parameter that controls the number of training samples to work through before the model's internal parameters are updated while the number of epochs is a hyper-parameter that controls the number of complete passes through the training dataset. Considering all the analysis above[48] towards basic CNN elements, the model designed for this current research can be seen in the next figure.

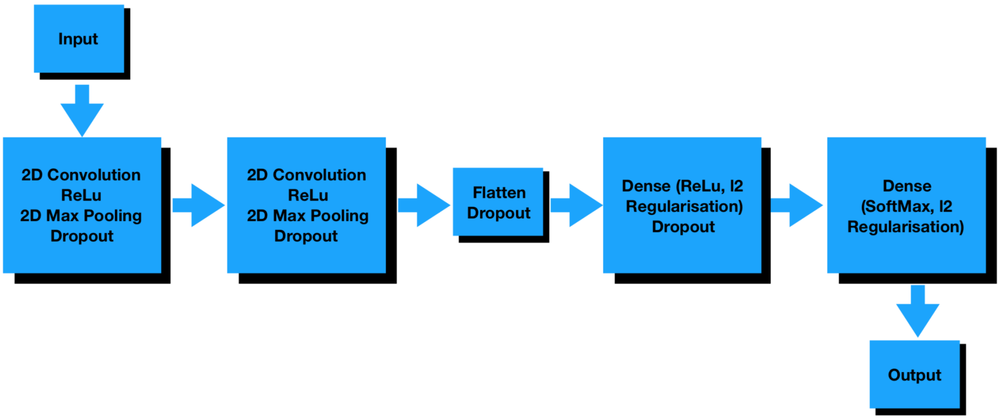


Figure 7: Convolutional Neural Network - Architecture

The shape of the input of Convolutional Neural Network is

X: (# of images)x 128 x 100 x 1)
Y: (# of images) x (of classes)

In other words, X input contain images (Mel-Spectrogram) while Y contains the values of the chosen feature. The model uses 2 2D CNNs sequentially. Those two blocks have different number of (3x3) filters, 16 and 32 respectively with (1x1) strides and same padding which tries to pad evenly left and right, but if the amount of columns to be added is odd, it will add the extra column to the right. Both of them apply RELU and 2D Max Pooling in order to reduce spatial dimensions of the image and avoid overfitting. The output from the those blocks is flattened and fed into a Dense layer of 64 units with RELU activation and L2 regularization. The final output layer of the model is a Dense Layer where the number of units is the same as the number of classes (3 for danceability, valence or energy and 10 for music genres). In the current layer Softmax activation is performed in order to return a probability for each value of the features and L2 regularization as well. The model is trained using SGD optimizer [48]with a learning rate of 0.001 while the loss function is categorical cross entropy. The model has been trained for 120 epochs and learning rate was reduced whether the validation accuracy plateaued for at least 3 epochs(patience).

The CNN, described in Figure 7, will be used to train the 4 different models needed. The only difference is the number of classes of the models of danceability, valence and energy ('low': 0, 'medium': 1, 'high': 2) and the genres model ('blues': 0, 'rock': 1, 'pop': 2, 'metal': 3, 'jazz': 4, 'classical': 5, 'electronic': 6, 'hiphop': 7, 'country': 8, 'dance': 9).

4.3 Experimental Results

This section contains the results of training procedure (described in the previous section) concerning danceability, valence, energy and music genres. The following graphical illustrations show the progress of both accuracy and loss during training phase through epochs (training cycles) and the confusion matrix of each one of the models.

Confusion matrix is a performance measurement for machine learning classification problems where output can be two or more classes. It is actually a table with 4 different combinations of predicted and true values. Both accuracy and confusion matrix are used to define how well the model performs in an unknown dataset (testing data).

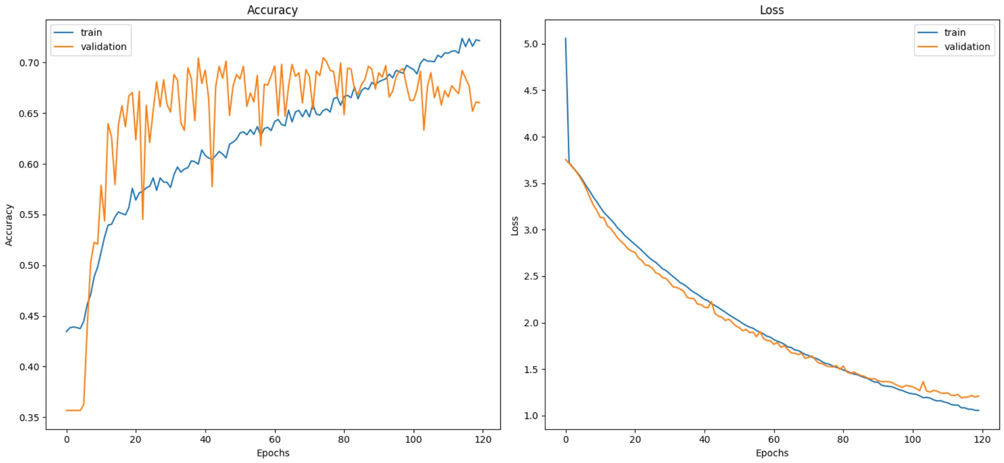


Figure 8: Danceability Feature - Accuracy over epochs

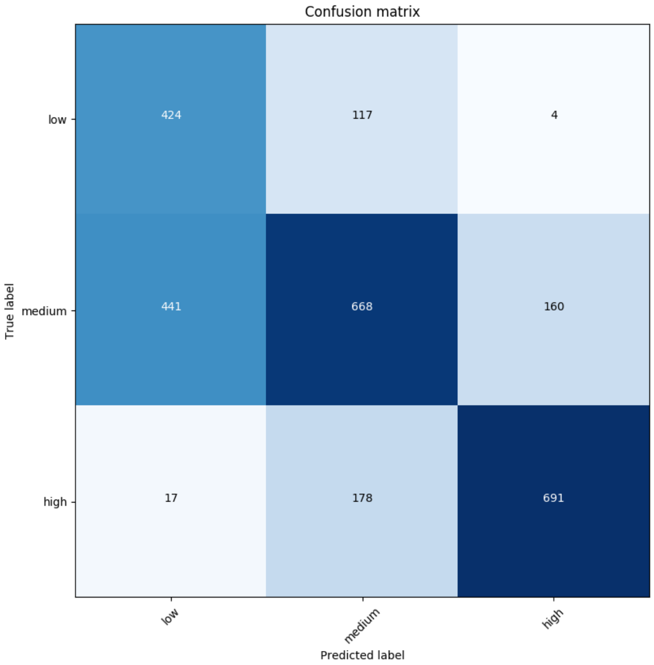


Figure 9: Danceability Feature - Confusion Matrix

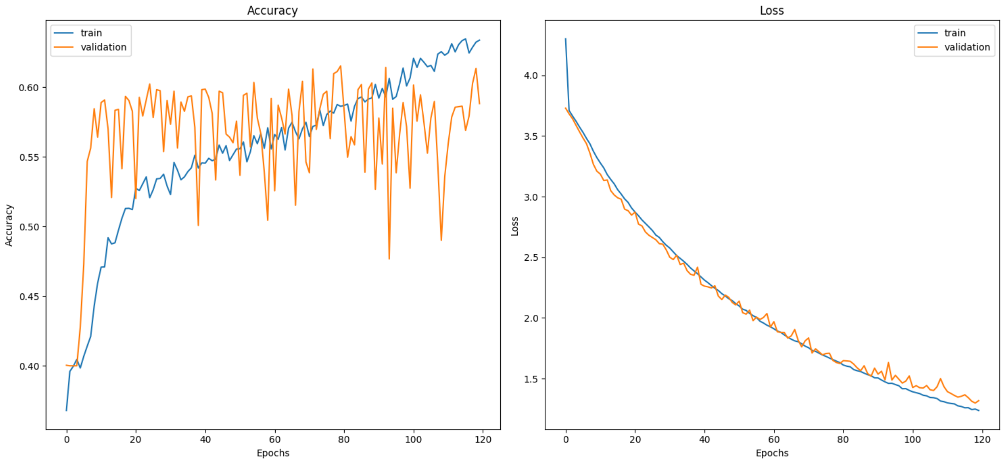


Figure 10: Valence Feature - Accuracy over epochs

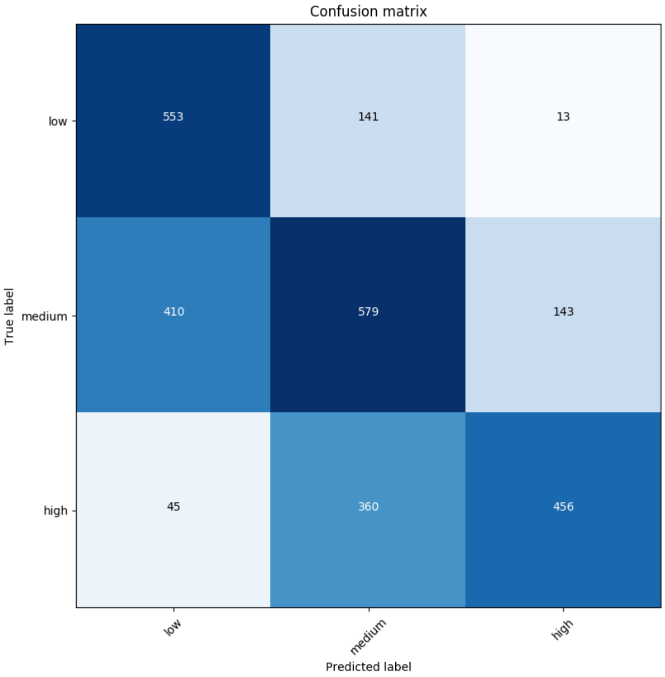


Figure 11: Valence Feature - Confusion Matrix

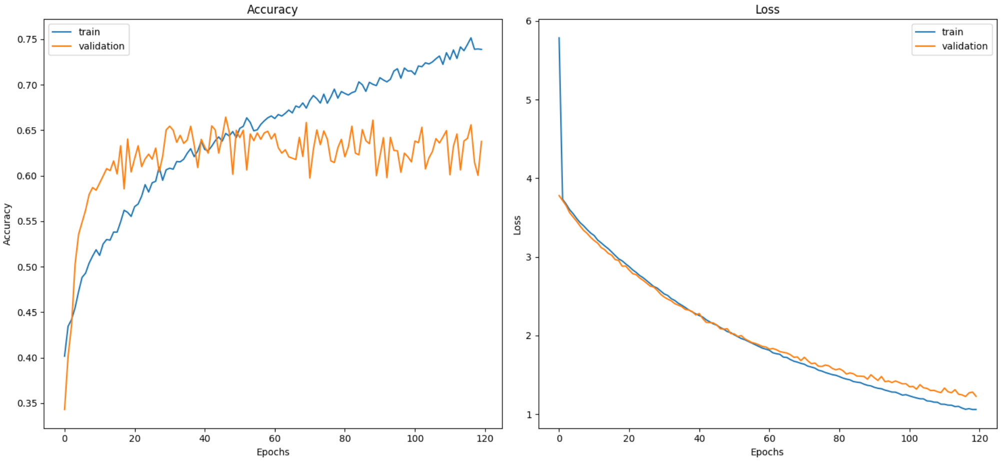


Figure 12: Energy Feature - Accuracy over epochs

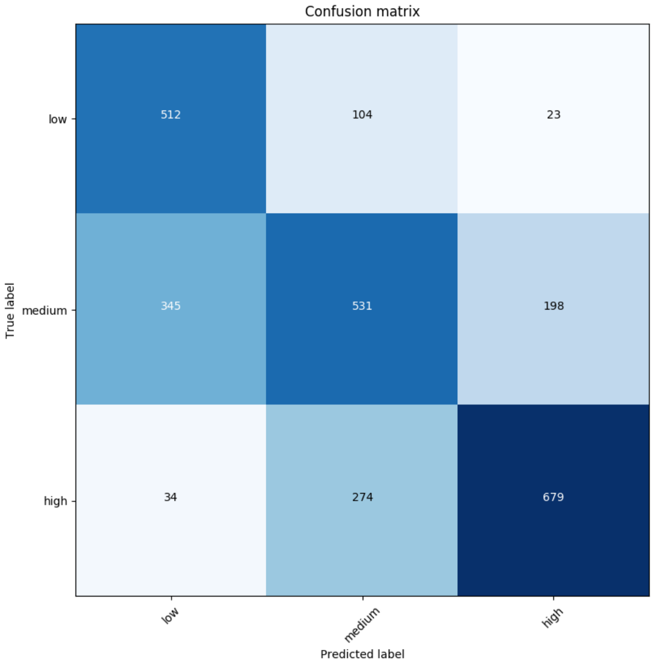


Figure 13: Energy Feature - Confusion Matrix

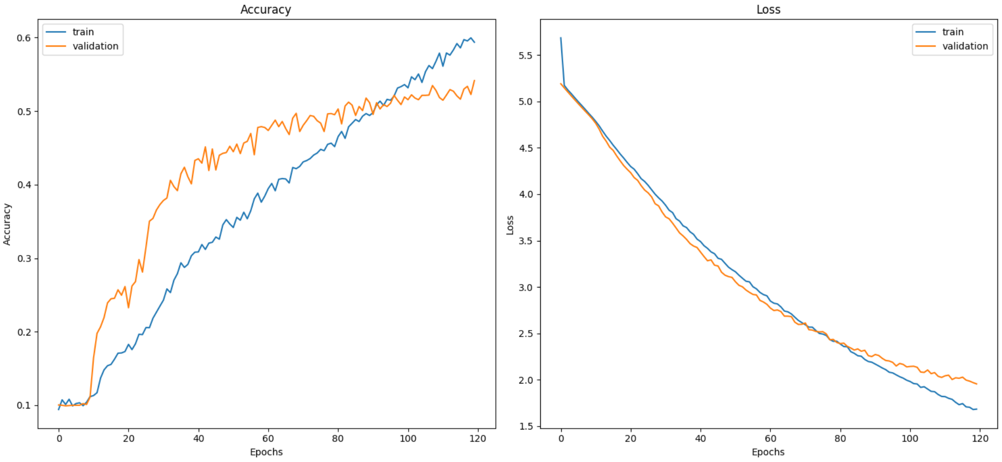


Figure 14: Music Genres Feature - Accuracy over epochs

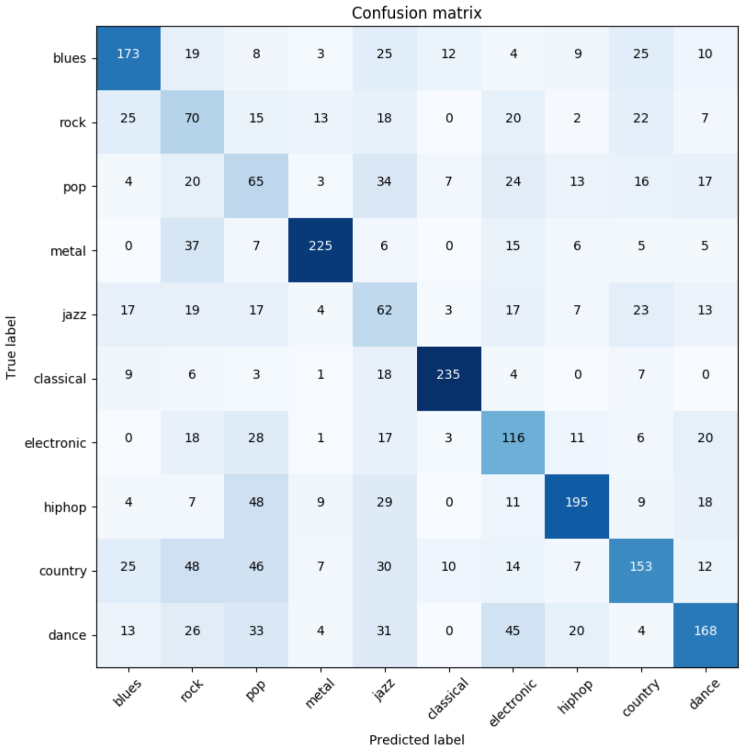


Figure 15: Music Genres Feature - Confusion Matrix

The results concerning the accuracy of all the trained models are summarising in the following table.

Trained Model	Accuracy (approximately)
Danceability	72%
Valence	70%
Energy	75%
Music Genres	60%

Table 11: Summary of Models' accuracy

It is obvious that all trained models predict correctly more than half of the unknown instances. Also confusion matrices show that there is not a class that most of the instances of the test file are miss predicted. It should be mentioned that changing the parameters of the Convolutional Neural Networks used for training the aforementioned models, may improve the accuracy score. This could be an extension of this research. Nevertheless, for the current research the accuracy of each model is sufficient considering that the purpose is not to build the perfect model but to show that the combination of all these models may lead to a good result.

Chapter 5

End to End Movie Music Analytics

The remaining task is to combine all those models trained in the chapters above and the metadata dataset in order to begin movie predictions and achieve the goal of this current research. The following table summarises all the tools gathered till now.

Trained models & Datasets	
Models & Datasets	Description
Movies Dataset	A collection of movies from several directors
Movies Metadata	Contains all metadata of movies of interest
Music Detection Model	Model that can classify each second of an audio file as Music, Speech or Other sound
Danceability Model	Model that is capable to predict the danceability of an audio segment
Valence Model	Model that is capable to predict the valence of an audio segment
Energy Model	Model that is capable to predict the energy of an audio segment
Music Genres Model	Model that is capable to predict the music genres of an audio segment

Table 12: Trained models & Datasets

Considering all the tools described above, the main task of this research should be defined. More specifically, each one of the movies, described in section 2.1 will be passed through music detection model to isolate the music parts that last more than 10 seconds. Although, the evaluation of music detection has reached 91% F1 score (Paragraph 3.1.2), we have the impression that during this phase this percentage will be decreased because the data have no arbitrary size in order to make use of long-averaging technique. More specifically, the input segments have fixed size of 1 second. Nevertheless, there is no way to spot the difference considering the fact that evaluation is not performed at this stage.

Continuing, those parts will be passed through all the models trained in order to obtain predictions for danceability, valence, energy and music genres for each one of the 10-second segments. Appropriate aggregations of the values of all those segments for each movie will be performed to obtain the prediction values in a movie level. Last but not least, both the predicted values and the metadata dataset of each movie will be combined in one dataset. Mining procedure will be performed to the aforementioned dataset and a series of visualisations will be created to reveal any hidden correlations between the attributes.

5.1 Implementation

For each one of the movies described above, audio is isolated and wav files are produced. Continuing, those wav files pass through a function that converts their rate to 8kHz and channel stereo to mono. All produced audio files of each director pass through the entire implementation sequentially. Let's follow the flow for an example.

Assuming that the procedure begins for the movie I Confess directed by Alfred Hitchcock. The audio file is extracted from the movie and transformed to the desirable rate and number of channels. The signal of the movie is extracted and passes through the SVM classifier. The model returns a sequence of seconds of the movie with a tag on each one among Music, Speech and Others. Continuing, the system finds all music segments consisting of 10 or more successive music segments. In the next step the system has to create the Mel-Spectrograms for each 10 seconds segment found. But what will happen with the music segments that last more than 10 seconds? In order to avoid losing seconds of music, we decided to take all those segments and perform a windowing procedure. For example a segment that lasts 12 seconds will pass through an algorithm that make use of a window of 10 seconds length sliding upon the audio seconds with step one. The result is the production of three 10 second segments. The procedure is visualised in the following figure.

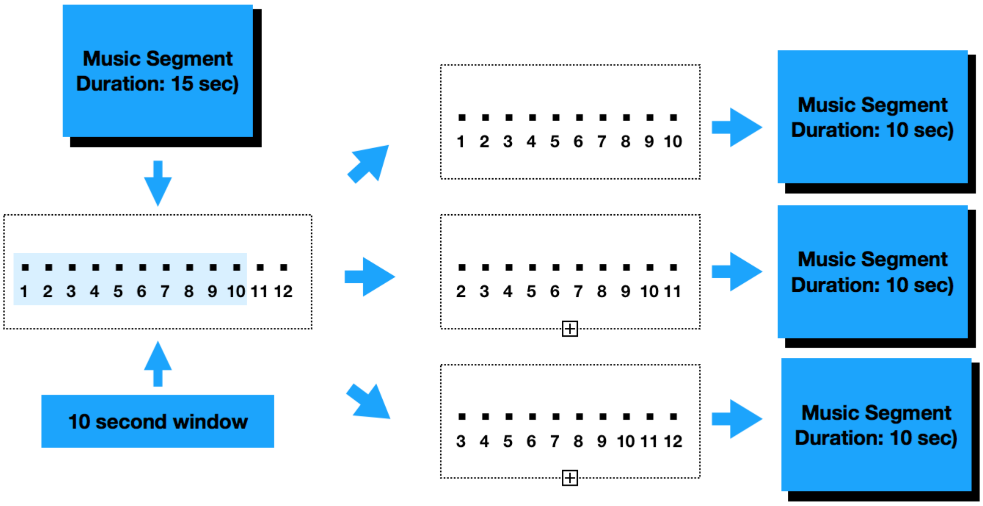


Figure 16: Perform windowing in a 13 second audio segment

At this point all the produced audio segments (10 second) of the movie will be transformed to Mel-Spectrograms (1 mel per 10-second segment) that will pass through the classifiers of danceability, valence, energy and music genres. Each one of the classifiers will produce a value for each Mel-Spectrogram or each 10-second audio segment and the probability of this value. In the following figure all the possible values for music genre, danceability, valence and energy are shown.

Trained models & Datasets	
Feature	Possible Values
Danceability	Low, Medium, High
Valence	Negative, Neutral, Positive
Energy	Low, Medium, High
Music Genres	Blues, Rock, Pop, Metal, Jazz, Classical, Electronic, Hiphop, Country, Dance

Table 13: Possible Values of Predictions

Segments whose duration is more than 10 seconds, are splitted to segments with duration of 10 seconds in order to take the predictions of danceability, valence, energy and music genres. Now it is time to group again the results of those segments keeping the most dominant (frequent) values. Continuing the previous example, lets say that the results of the predictions were those shown in the following table.

Features & Predictions of a 12 second segment				
Segments(sec)	Danceability	Valence	Energy	Music Genres
1-10	Low	Medium	Medium	Blues
2-11	Medium	Medium	High	Classical
3-12	Low	High	Medium	Blues

Table 14: Example - predictions of a 12 second Segment

So, the system keeps the most frequent values and assigns a weight in each segment which is actually its duration.

Features & Aggregated Predictions of a 12 second segment					
Segments(sec)	Danceability	Valence	Energy	Music Genres	Weight
1-12	Low	Medium	Medium	Blues	12

Table 15: Example - aggregated predictions of a 12 second Segment

The information of each movie in segment layer has been extracted and it is time to perform appropriate aggregations to the segments of each movie in order to produce results concerning the whole movie. The aggregation procedure is really simple. Actually, the same values of each label are counted and the result is divided with the summation of the segment weights. It is obvious that the results of all values of each label should sum to 100. Assuming that the results for a movie concerning danceability after the aforementioned aggregation are Low: 70, Medium 15 and High 15. Those numbers indicate that the 70% of the music segments are predicted to be Low while 15% of them are Medium and the rest High concerning danceability.

The procedure described in this section is performed to each movie of the dataset. The result of the aforementioned procedure performed to I Confess can be seen in the following table.

movie: I Confess	
Label	Value
Blues	14
Rock	2
Pop	0
Metal	0
Jazz	31
Classical	51
Electronic	1
Hiphop	0
Country	1
Dance	0
Dance Low	67
Dance Med	31
Dance High	2
Valence Negative	80
Valence Neutral	20
Valence Positive	0
Energy Low	60
Energy Med	15
Energy High	25
Music Duration	31
Speech Duration	63
Other Duration	6
Director	Alfred Hitchcock

Table 16: Movie Predictions (I Confess)

Concluding, all predictions per movie and the metadata are merged to a dataset which will be used for mining and visualising purposes. Below you can see all information from the enhanced dataset concerning the movie I Confess.

movie: I Confess	
Movie	I Confess
Blues	14
Rock	2
Pop	0
Metal	0
Jazz	31
Classical	51
Electronic	1
Hiphop	0
Country	1
Dance	0
Music Duration	31
Speech Duration	63
Other Duration	6
Director	Alfred Hitchcock
Year	1953
Rating	7.3
Metascore	73
Country	USA
Locations	Maison Hearn, Grande Allée Est, Quebec, Canada
Movie Genres	Crime, Drama, Thriller
Cast	Montgomery Clift, Anne Baxter, Karl Malden
Energy	Low
Danceability	Low
Valence	Negative

Table 17: Enhanced Dataset Example (I Confess)

5.2 Validity Evaluation - Dash Application

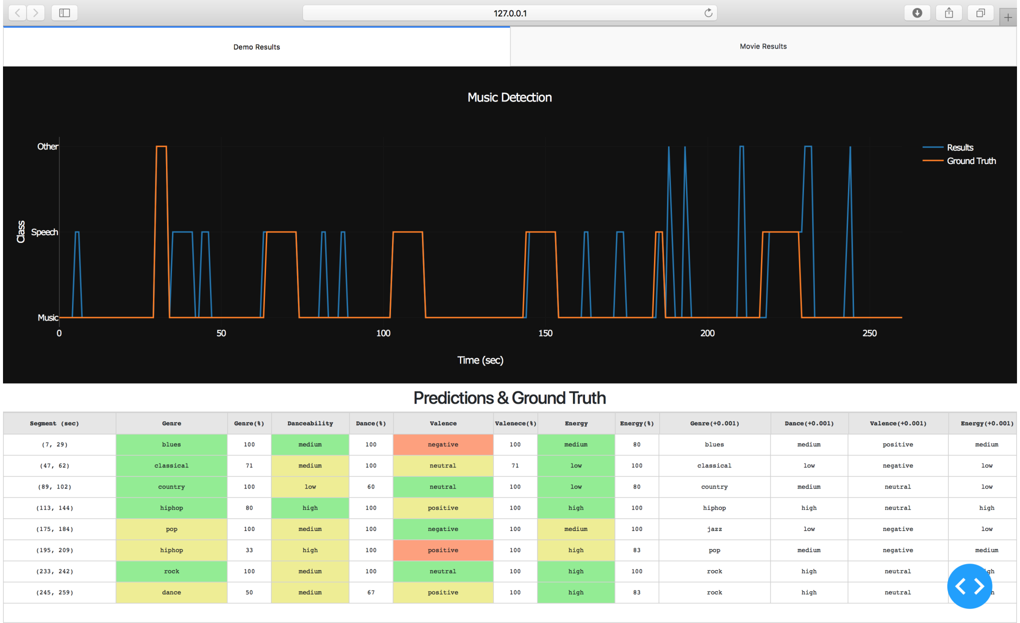
In order to validate if the models trained in the previous sections can make right predictions, a synthetic audio file was created. This audio file lasts 4min and 20 seconds and was created by combining several speech segments from movies, parts of 7 songs (blues, classic, country, hiphop, jazz, pop, rock) and a file with various sounds. The evaluation using this synthetic file is really easy considering the fact that the predictions can be compared to the ground truth values. The results of this comparison are visualised in a dash application.

Data visualisation is the graphical representation of information and data. Using various visual elements such as charts, graphs, maps, data provide an accessible way to see and reveal trends, outliers and patterns in data. Considering the volume of big data, data visualisations are essential for analysing massive amounts of information and making data-driven decisions.

In order to present some of the results of the enhanced dataset produced and the results of validity evaluation procedure a dash application[37] is produced, which is a productive Python framework for building web applications. Dash is written on top of Flask, Plotly.js, and React.js and it helps python developers to create interactive web-based applications.

More specifically this application produced, has two different tabs. The first one refers to the results of the validity evaluation procedure while the second one presents the values predicted to the whole movie dataset in both segment and movie layer.

The first tab contains an interactive graph which shows the results of music detection classifier and the ground truth values and a table that compares the results of the classifiers of danceability, valence, energy and music genres with ground truth values. The first graph is a XY-Diagram where X-axis is the class (Music, Speech, Other) and Y-axis is time. In this graph the reader may see with a blue line the predicted classes from the music detection classifier through time and with an orange line the real classes through time in the same diagram. In this way, it is really easy for anyone to understand that music detection is working quite well considering that most of the prediction classes made by the classifier are correct. Continuing, the table shows all music segments of the synthetic audio file and all predicted values made by the classifiers concerning danceability, valence, energy and music genres with their probabilities and the ground truth values as well. Those predictions that are correct in regards to ground truth values are painted with green color, those who are simple errors with a yellow color while the extreme errors are painted in red. In the following figure the first tab of this small application is presented.



Predictions & Ground Truth

Segment (sec)	Genre	Genre(%)	Danceability	Dance(%)	Valence	Valenece(%)	Energy	Energy(%)	Genre(+0.001)	Dance(+0.001)	Valence(+0.001)	Energy(+0.001)
(7, 29)	blues	100	medium	100	negative	100	medium	80	blues	medium	positive	medium
(47, 62)	classical	71	medium	100	neutral	71	low	100	classical	low	negative	low
(89, 102)	country	100	low	60	neutral	100	low	80	country	medium	neutral	low
(113, 144)	hiphop	80	high	100	positive	100	high	100	hiphop	high	neutral	high
(175, 184)	pop	100	medium	100	negative	100	medium	100	jazz	low	negative	low
(195, 209)	hiphop	33	high	100	positive	100	high	83	pop	medium	negative	medium
(233, 242)	rock	100	medium	100	neutral	100	high	100	rock	high	neutral	igh
(245, 259)	dance	50	medium	67	positive	100	high	83	rock	high	neutral	h

Figure 17: Dash Application - Evaluation Procedure

By observing the table of the application, it is easy to understand that results are really good. Not only the majority of the predicted values are correct but there are not many extreme errors. As already described, danceability, valence and energy can take three different values. A mismatch is either a simple error or an extreme error. An extreme error is for example when the classifier predicts Low value of danceability instead of High. In case ground truth value is Medium, it is called a simple error. It is obvious that the best classifier according to this simple synthetic file is the one that predicts energy values while the worst one is the one responsible for valence where 2 extreme errors exist.

5.3 Dashboard Application - Visualisations

As mentioned in the previous section, the Dash application produced, has two different tabs. The second tab of the application contains the prediction values for all movies both in segment and movie level. This tab contains 3 empty fields.

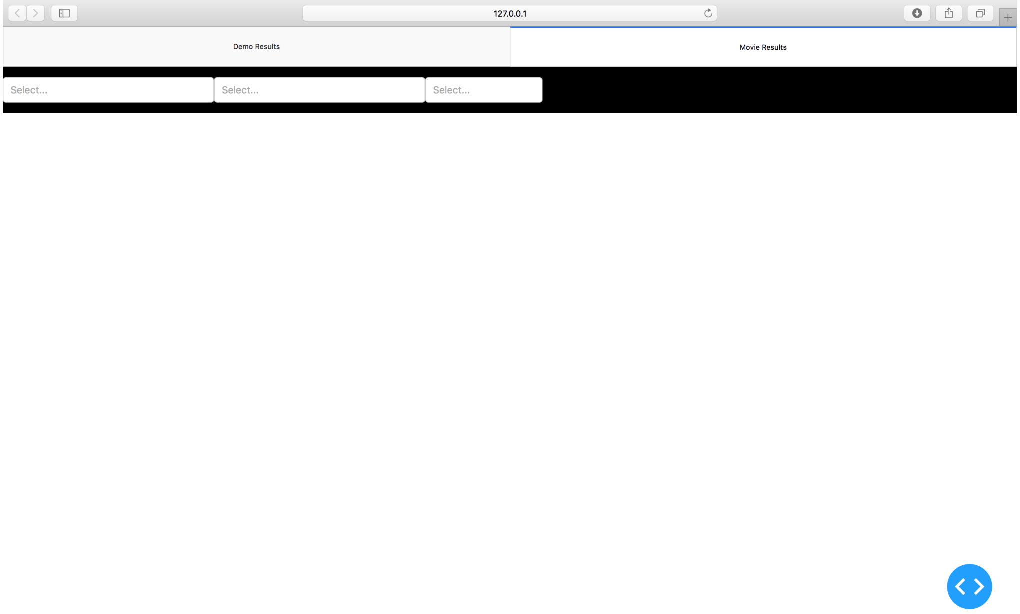


Figure 18: Dash Application (Movie Results) - Demo Example 1

On click on each one of the fields a drop down list will appear. A click to the first one reveals all the directors for investigation.

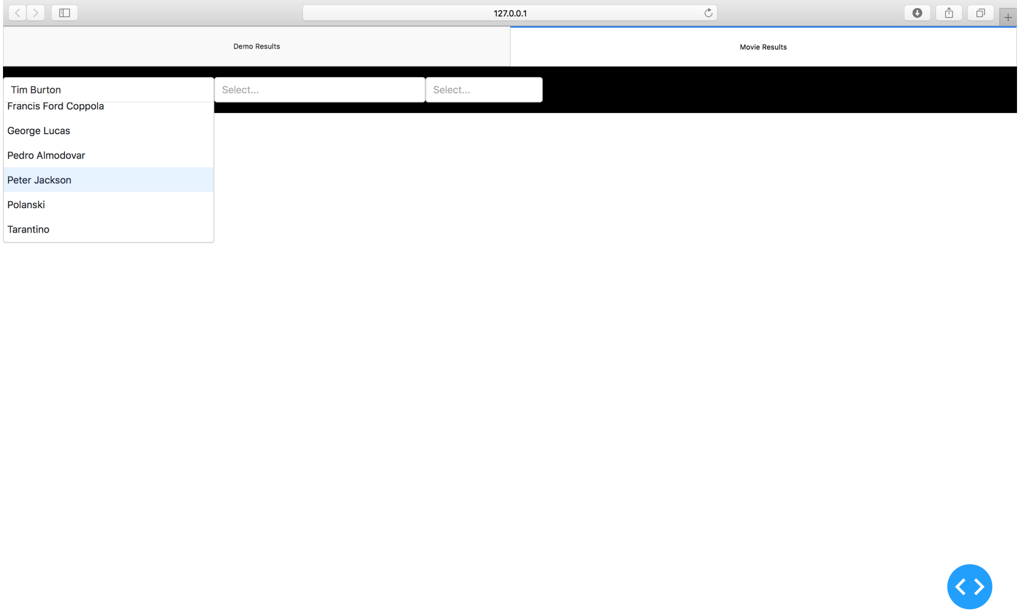


Figure 19: Dash Application (Movie Results) - Demo Example 2

When a director is selected, the user may click on the second field and all movies of the selected director will appear.

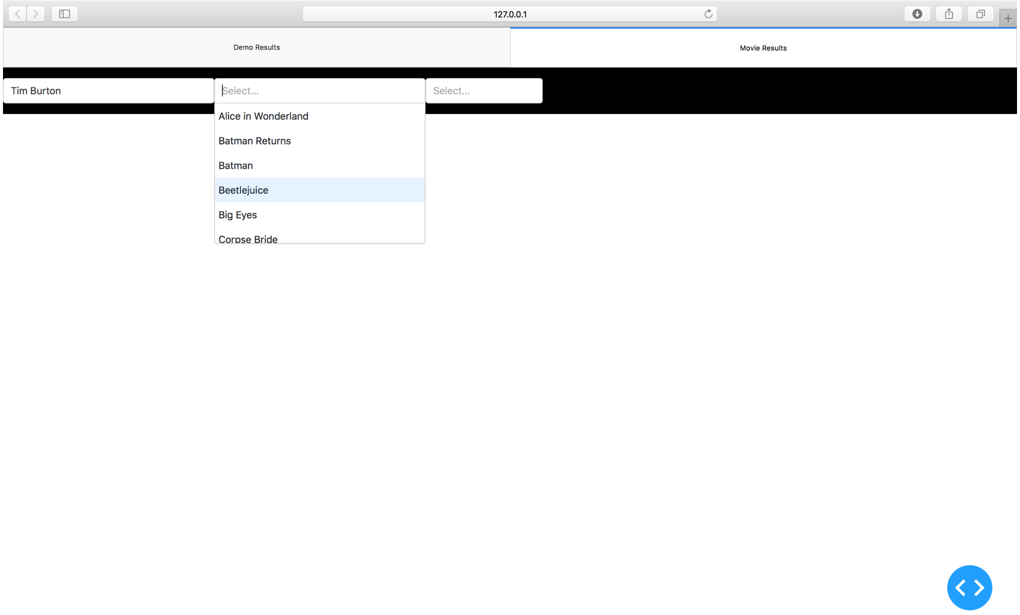


Figure 20: Dash Application (Movie Results) - Demo Example 3

Last but not least, the third drop down list contains 2 values, Segment Layer and Movie Layer.

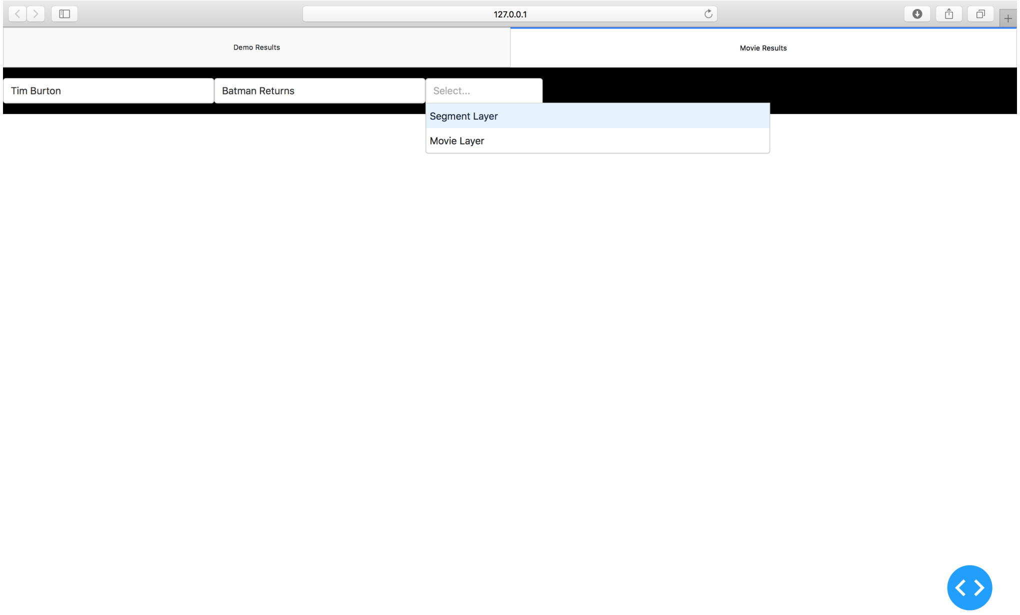
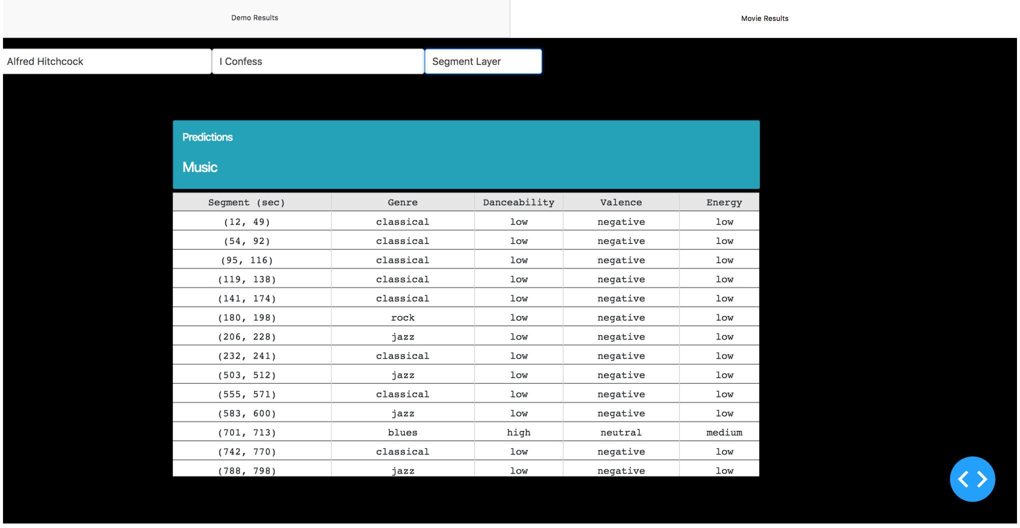


Figure 21: Dash Application (Movie Results) - Demo Example 4

By choosing the Segment Layer value, a table with 5 columns will appear, in which each row represents the predictions of a music segment concerning genre, danceability, valence and energy.



Predictions

Music

Segment (sec)	Genre	Danceability	Valence	Energy
(12, 49)	classical	low	negative	low
(54, 92)	classical	low	negative	low
(95, 116)	classical	low	negative	low
(119, 138)	classical	low	negative	low
(141, 174)	classical	low	negative	low
(180, 198)	rock	low	negative	low
(206, 228)	jazz	low	negative	low
(232, 241)	classical	low	negative	low
(503, 512)	jazz	low	negative	low
(555, 571)	classical	low	negative	low
(583, 600)	jazz	low	negative	low
(701, 713)	blues	high	neutral	medium
(742, 770)	classical	low	negative	low
(788, 798)	jazz	low	negative	low

Figure 22: Dash Application (Movie Results) - Demo Example 5

Moreover, choosing the Movie Layer value, 4 cards will become visible to the user.

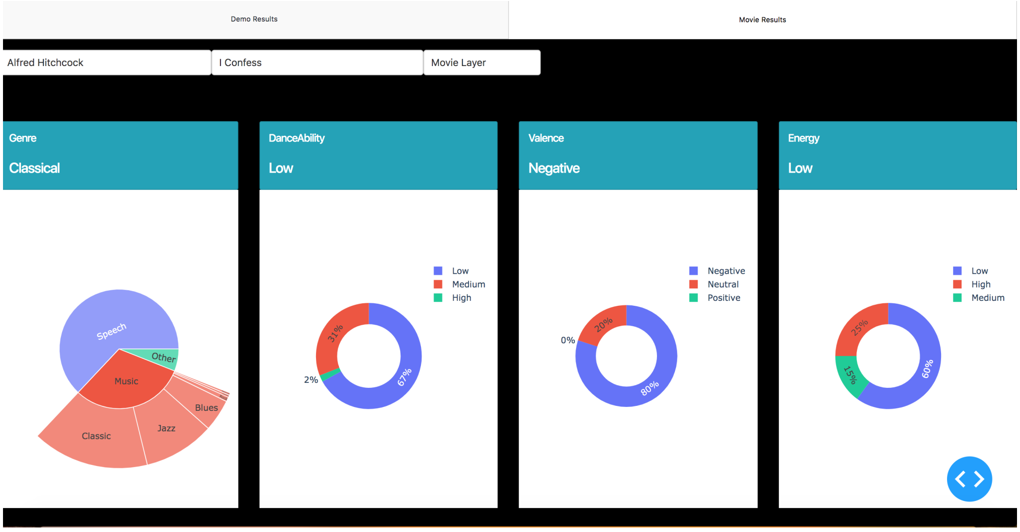


Figure 23: Dash Application (Movie Results) - Demo Example 6

From left to the right the cards illustrate the aggregated results concerning music genres, danceability, valence and energy of the movie. At the top of each card the feature in which the card refers to and the result concerning the current feature are declared. The illustration in the white part of each card shows the analysis of the result. The visualisation of the music genres is a 2 level pie graph. The core of the pie is a circle divided in three parts (Music, Speech, Other) e.g. the size of music slice indicates the percentage of music in the entire movie and so on. In addition, the external side of music part is covered by a second pie which shows the contribution of the different types of music in music of this movie. On click to the core of this pie, as shown in the next figure, the pies' parts of speech and other will be vanished and the music will cover the whole circle in order to make easier for user to observe the contribution of the types of music to the movie's music .

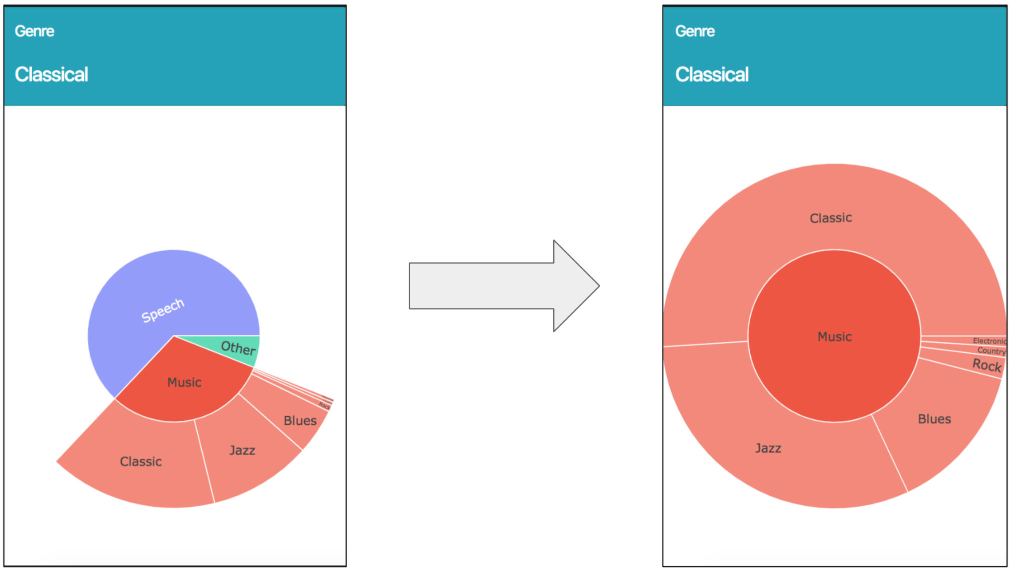


Figure 24: Dash Application (Movie Results) - Demo Example 7

Each one of the cards concerning danceability, valence and energy contain a ring which is divided in three parts. It is obvious that the value of the feature that covers the biggest part of the ring is the dominated value for this feature.

Chapter 6

Mining & Visualising Musical Content and Movie's Metadata

While the enhanced dataset, consisting of all the predicted music attributes and movies metadata, has been created through all the steps described in the previous chapters, it is time to perform mining procedure in order to reveal any hidden correlations, patterns in data, etc. The results of the aforementioned mining procedure that seem interesting will be presented through a series of visualisations created by plotly[36]. The main purpose of this research is to find correlations between directors and the predicted music attributes and to search if the usage of those attributes combined with movie metadata are capable of visually separate directors in 2D space.

First of all, in order to reveal any correlations between the features (music features & metadata) all data will be loaded to a Pandas dataframe and preprocess will be performed in order to plot them in the same scale. For example Metascore (IMDB metric scores from an external website) and mean of the users' scores, Rating don't have the same scale. So Metascore will be multiplied by ten. Also all values of Metascore that are empty will be filled with the value of Rating. In addition the categorical attributes Countries, Valence, Energy, Danceability and Director will be transported to integers in order to become comparable e.g. Valence of value Low, Medium or High which will become 0, 1 or 2 respectively. Movie Genres is an attribute that indicates the type of movie. This attribute contains more than one value. In order to transport this feature in a way that it can be used, comma separator will be eliminated and concatenation of the values into a string will be performed. In this way every movie has a string as a value and so it can be transported to an integer. The reason that this approach is going to work is because of the observation that the same combinations of types exist in different movies such as Crime, Thriller or Comedy, Romance and so on. Last but not least, as described before in this dataset all music genres were kept for each movie. At this point, it was decided that only those genres that contribute more than 20% of the total genres of each movie will be used. In order to achieve that, genres of each movie were filtered by the contribution and were concatenated into a string so as to be able to become integers.

Now that all the aforementioned attributes are comparable to each other, pairwise Spearman's correlation[49] will be performed. Spearman's rank-order correlation is the non parametric version of the Pearson product-moment correlation. Spearman's correlation coefficient, measures the strength and direction of the linear association between two ranked variables. The heatmap below shows the correlation results of all attributes.

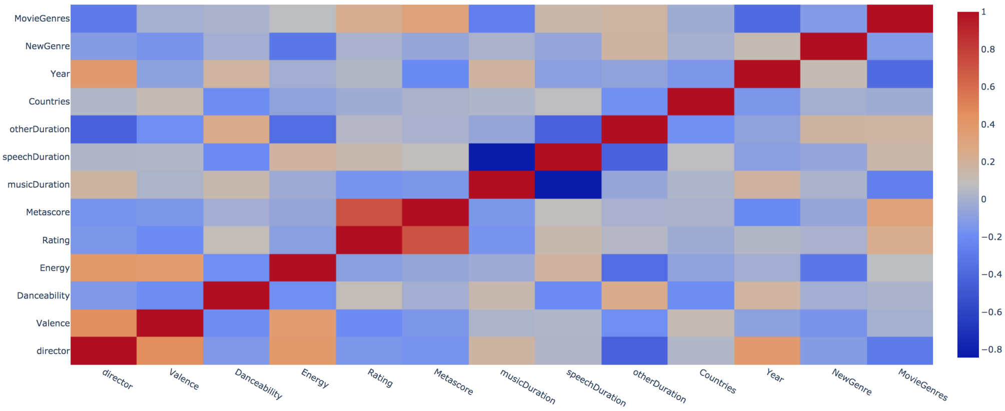


Figure 25: Heatmap - Spearman's correlation Results

By observing the heatmap above, there are some really interesting conclusions that can be made concerning the correlation between various attributes. Firstly, a result that was expected is that there is a strong positive correlation between rating and metascore.

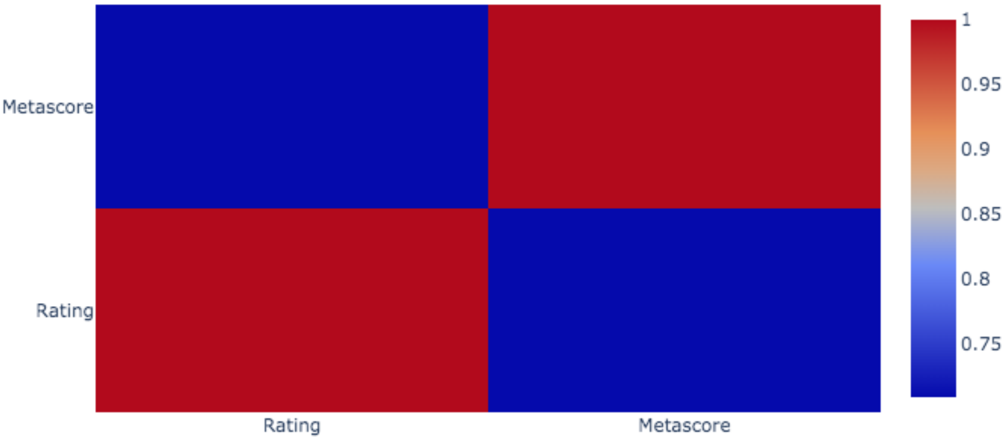


Figure 26: HeatMap - Spearman's correlation Results

An other strong positive correlation is observed between director and the year movies of the director were released which is quite interesting.

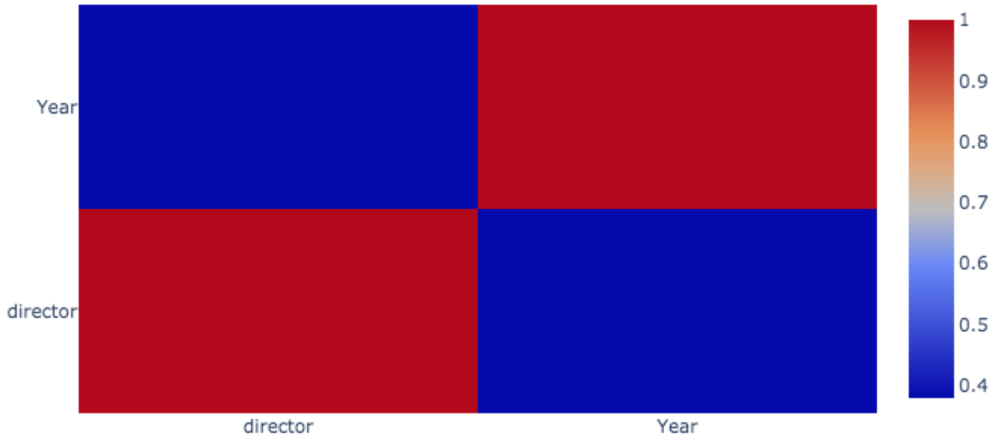


Figure 27: HeatMap - Spearman's correlation Results

Continuing, as expected, the duration of music, speech and others are correlated in some way. More specifically, it seems that the duration of music and speech are strongly negative correlated, while the duration of other sounds seems to be quite independent of the other two duration values.

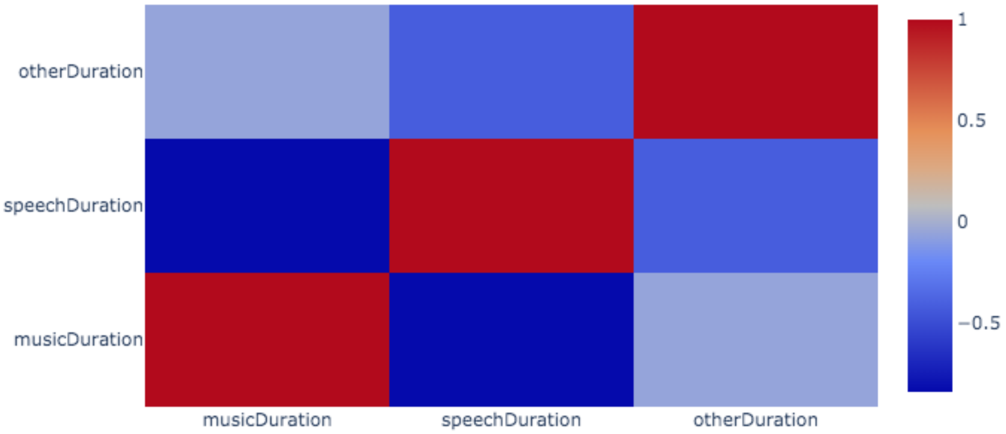


Figure 28: Heatmap - Spearman's correlation Results

The following heatmap shows how correlated are the audio features from music of the movies and the director. It seems that danceability is negatively correlated to both valence and energy. The most interesting observation is that director is positively correlated with valence and energy and also comparing to danceability there is a slightly negative correlation.

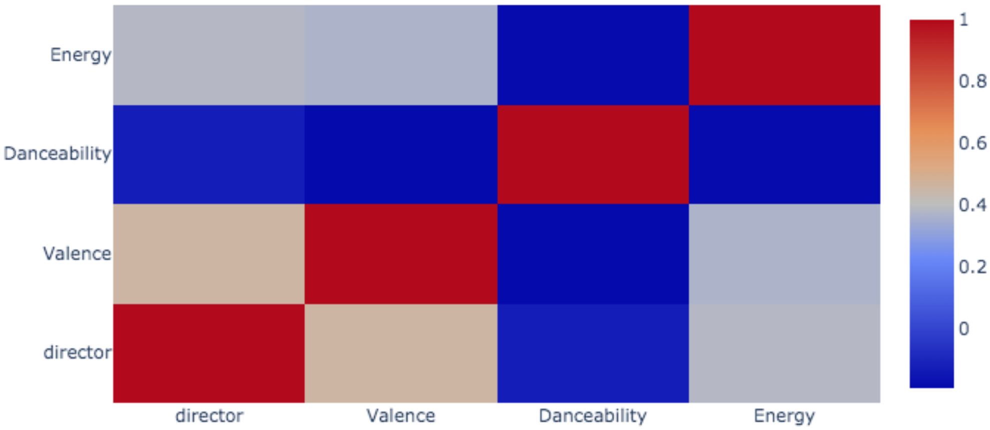


Figure 29: Heatmap - Spearman's correlation Results

Last but not least, type of movie genres and the director seems to have a small negative correlation.

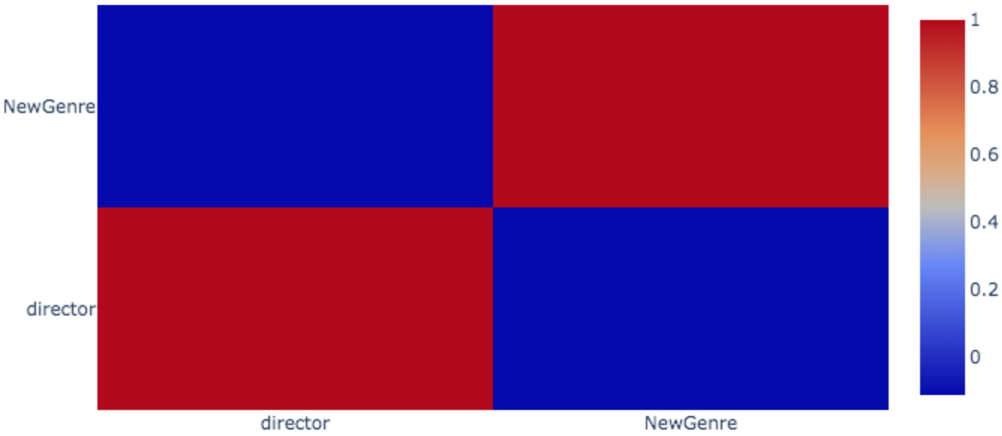


Figure 30: Heatmap - Spearman's correlation Results

For the next series of plots, Principal Component Analysis or PCA[34] which is a widely used technique for dimensionality reduction of large data set will be used for both metadata and audio features. Music features (Blues, Rock, Pop, Metal, Jazz, Classical, Electronic, Hiphop, Country, Dance,musicDuration, speechDuration, otherDuration, Energy, Danceability, Valence) will be reduced to 2 through the use of PCA. Those two new features will be X and Y axis of the following plot, while directors will define the color of the points of this 2D plot.

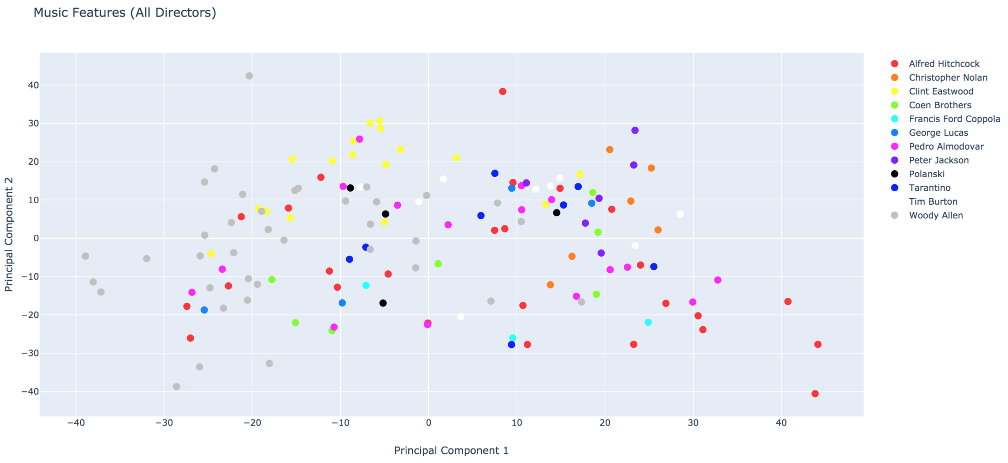


Figure 31: 2D Representation of Music Features reduced in 2 with PCA

The same procedure will be followed for the metadata features (Rating, Metascore, Countries, Year, MovieGenres) of all movies.

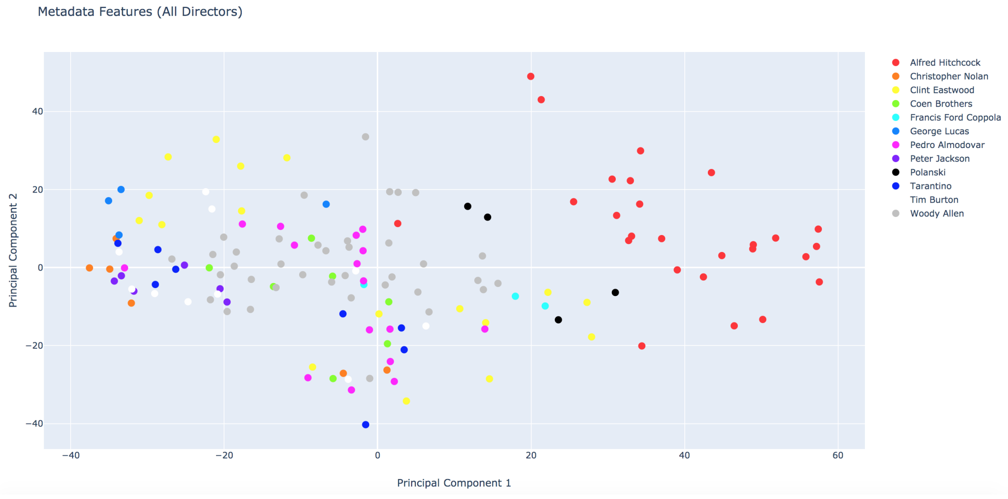


Figure 32: 2D Representation of Metadata Features reduced in 2 with PCA

For the next representation, both metadata and music features are going to be plotted in the same figure. To accomplish this goal music features are reduced to 1 through PCA and 1 more for all metadata features. Music Features will be used as Y-axis, Metadata as X-axis while color of the points is changed with respect to the directors of the movies.

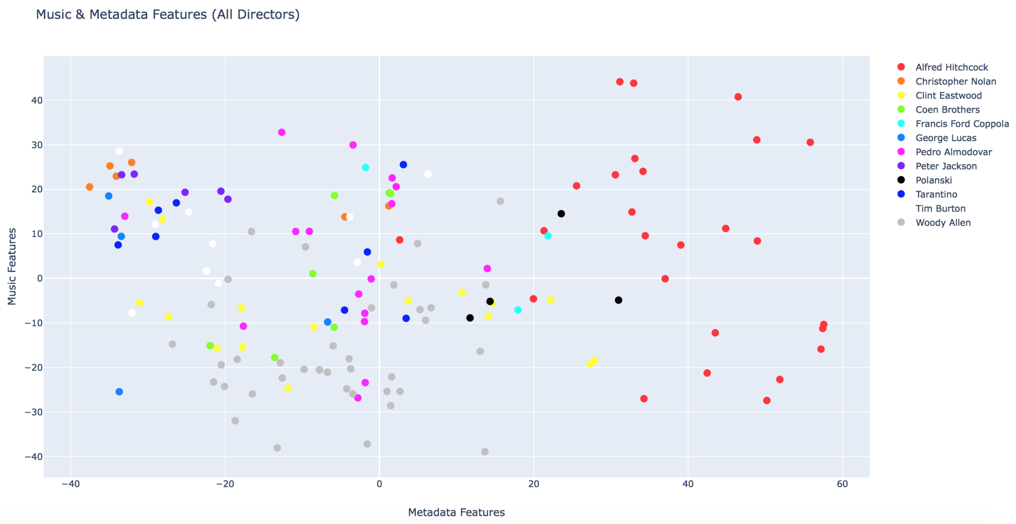


Figure 33: 2D Representation of Music and Metadata Features

Taking into account the visualisation above, it seems that all movies of the same director tend to be gathered together in a region of this 2D representation. In order to make clear this deduction, decision surfaces will be used. More specifically, an SVM classifier will be trained by the data and a grid of points spanning the entire space will be created. Continuing, predictions of each observation in the grid will be performed using SVM and the results will be plotted. In this way, background color of the data points will be colored with the same color with the points that dominate that region in a more slight tone.

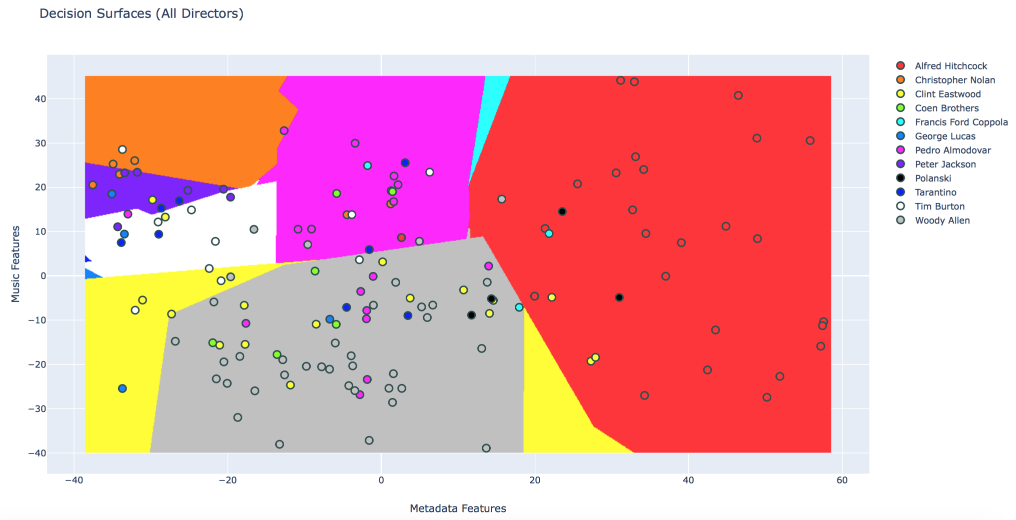


Figure 34: 2D Representation of Music and Metadata Features with Decision Surfaces - All Directors

Although the directors seem to be slightly separated when metadata and music features are combined, it is clear from the previous illustration that when decision surfaces are performed to all directors in the same scatter plot things seem like a mess. So, in the next three figures a subset of directors will be used in order to achieve more clear illustrations.

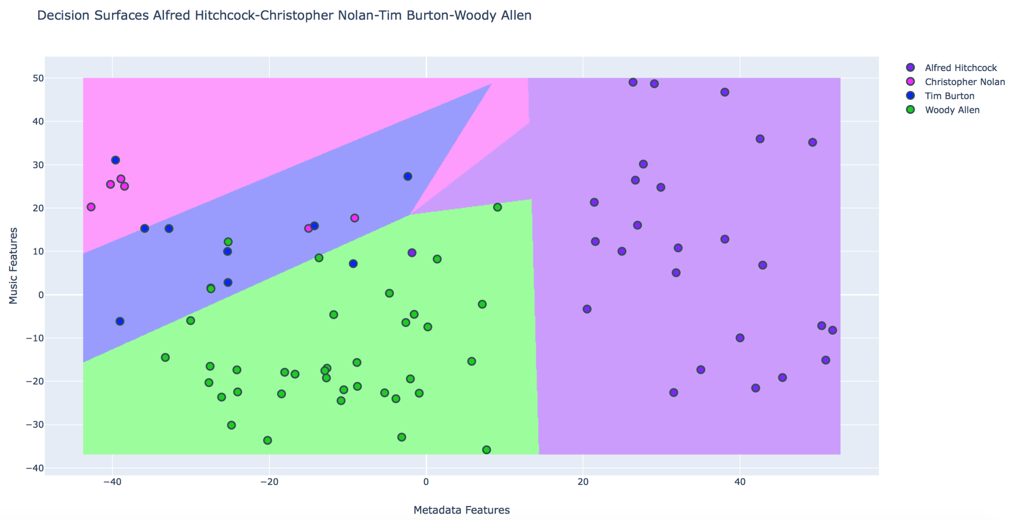


Figure 35: 2D Representation of Music and Metadata Features with Decision Surfaces - Alfred Hitchcock, Christopher Nolan, Tib Burton, Woody Allen

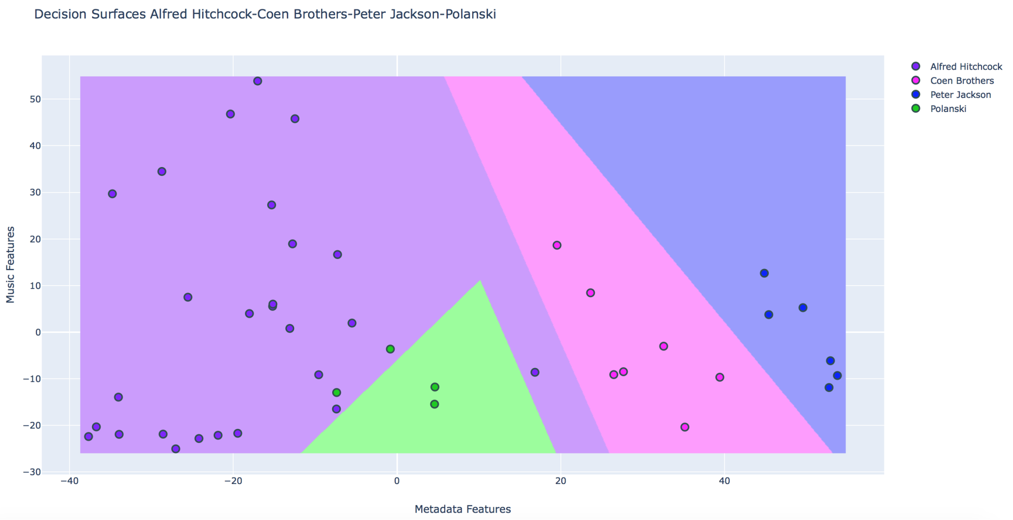


Figure 36: 2D Representation of Music and Metadata Features with Decision Surfaces - Alfred Hitchcock, Coen Brothers, Peter Jachson, Polanski

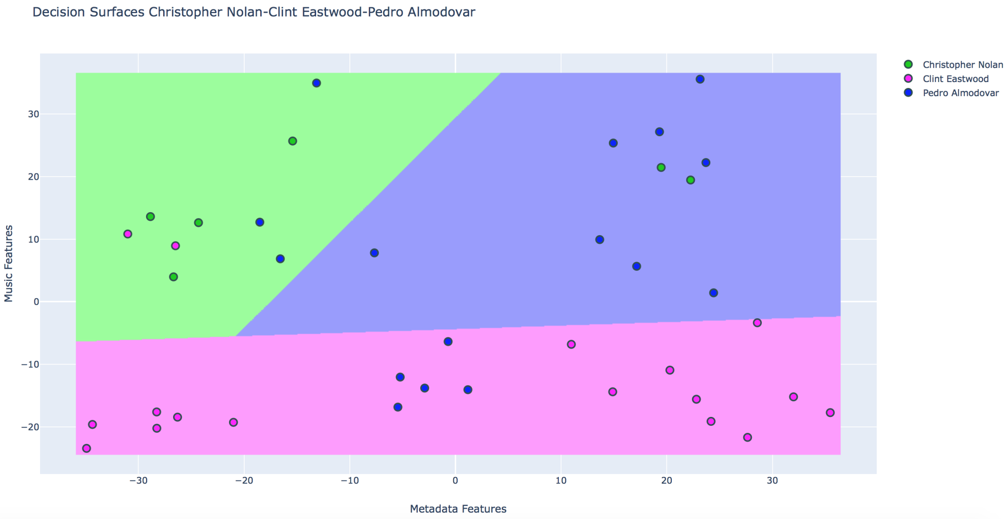


Figure 37: 2D Representation of Music and Metadata Features with Decision Surfaces - Christopher Nolan, Clint Eastwood, Pedro Almodovar

Regarding all the above visualisations, it seems that there are hidden correlations between several attributes of the enhanced dataset. Also, the suspicion that the combination of metadata and music features of a collection of movies may lead to some conclusions regarding the directors of the movies is confirmed.

Chapter 7
Discussion

A recommendation system has the ability to recommend items to users that may interested in from a great pool of possible choices. Content-based recommendation systems are trying to recommend items similar to those a given user has shown an interest in the past. More specifically a movie content-based recommender is trying to match the attributes (ratings, directors of movies chosen, etc) of a user's profile with the attributes (music, speech valence, subtitles, etc) of the content of the preferred movies, in order to recommend to the user new interesting items.

Several studies show that music affects many parts a human's brain very deeply. It creates strong feelings and a lot of memories. Considering the impact of music to humans' lives it justifies why music is one of the most important aspects. It may reach an audience emotionally beyond the ability of picture and sound.

Like in many other research areas, deep learning is increasingly adopted in music recommendation systems. In this thesis, both machine learning and deep learning are used to extract valuable information from music of several movies, in order to create a good base for a recommendation system. The target of this current research is to conclude to a dataset containing the predicted values of several music attributes of 146 movies of 12 different directors and a collection of movies' metadata which can be used as a good base for a recommendation system.

Firstly, a simple music detector based on SVM classifier and hand crafted features is trained and applied to a collection of movies in order to detect music over speech and other sounds for each one of them. Then 4 deep learning models based on Convolution Neural Networks are trained to classify 4 spotify features (danceability, valence, energy, music genres) of the music parts of each one of the movies found in the aforementioned collection. In addition, a synthetic audio file is created for evaluation purposes. This file passes through the entire implementation and the predicted values are compared to the the ground truth values in order to evaluate the classifiers that are going to be used to real movies. Continuously, an interactive web application was created with dash framework and plotly in order to present results in a more user friendly way and facilitate the investigation of the data. Last but not least, data mining is performed to the aforementioned dataset and a series of plots are produced revealing correlations and patterns of the attributes enclosed in this dataset.

Concerning future work, the existing implementation could be refactored in order to be able to predict more spotify audio features such as acousticness, instrumentalness, liveness, loudness and so on. Moving a step forward, the music features of the movies could be combined with visual features from movies' images, text features extracted from the subtitles and the emotion detection of the actors in order to design the ultimate content-based recommendation system. Considering the results of this thesis, correlation found between director and the spotify features seems really interesting and could be a good start for a research around this topic. The fact that directors are slightly separated visually in a 2D scatter plot of music and metadata features of movies make us wonder if it is possible to predict the director of a movie taking into consideration only music content of a movie and its metadata. Concluding, the contribution of this research is the design and implementation of an end to end system that takes as input a collection of movies and creates a unique dataset that contains metadata of the movies and predictions of audio features of music parts of the movie. This dataset would be a good base for a movie content-base recommendation system.

Bibliography

[1] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, "Recommender systems: An introduction." USA: Cambridge University Press, 2010.

[2] S. Khusro, Z. Ali, and I. Ullah, "Recommender systems: Issues, challenges, and research opportunities," 02 2016, pp. 1179–1189.

[3] S. Mohanty, J. Chatterjee, S. Jain, A. Elngar, and P. Gupta, *Recommender System with Machine Learning and Artificial Intelligence: Practical Tools and Applications in Medical, Agricultural and Other Industries.* Wiley, 2020. [Online]. Available: https://books.google.gr/books?id=36TqDwAAQBAJ

[4] R. Burke, A. Felfernig, and M. Göker, "Recommender systems: An overview," vol. 32, 09 2011, pp. 13–18.

[5] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, "Recommender systems," vol. 24, 01 2010.

[6] [Online]. Available: http://www.netflix.com/

[7] [Online]. Available: http://www.amazon.com/

[8] H. Gaudani, "A review paper on machine learning based recommendation system," vol. 2, 12 2014, pp. 3955–3961.

[9] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, "Collaborative filtering recommender systems," vol. 4, no. 2, 2011, pp. 81–173. [Online]. Available: http://dx.doi.org/10.1561/1100000009

[10] F. Mansur, V. Patel, and M. Patel, "A review on recommender systems," in *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, 2017, pp. 1–6.

[11] B. Yang, T. Mei, X.-S. Hua, L. Yang, S.-Q. Yang, and M. Li, "Online video recommendation based on multimodal fusion and relevance feedback," 07 2007, pp. 73–80.

[12] T. Mei, B. Yang, X.-S. Hua, and S. Li, "Contextual video recommendation by multimodal relevance and user feedback," *ACM Trans. Inf. Syst.*, vol. 29, p. 10, 04 2011.

[13] S. N. Tiwari, N. Q. K. Duong, F. Lefebvre, C.-H. Demarty, B. Huet, and L. Chevallier, "DEEP FEATURES FOR MULTIMODAL EMOTION CLASSIFICATION," Mar. 2016, working paper or preprint. [Online]. Available: https://hal.inria.fr/hal-01289191

[14] S. AshwinT, S. Saran, and G. R. M. Reddy, "Video affective content analysis based on multimodal features using a novel hybrid svm-rbm classifier," *2016 IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics Engineering (UPCON)*, pp. 416–421, 2016.

[15] Z. Wang, X. Yu, N. Feng, and Z. Wang, "An improved collaborative movie recommendation system using computational intelligence," *Journal of Visual Languages Computing*, vol. 25, 10 2014.

[16] T. Guha, C.-W. Huang, N. Kumar, Y. Zhu, and S. S. Narayanan, "Gender representation in cinematic content: A multimodal approach," in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, ser. ICMI '15. New York, NY, USA: Association for Computing Machinery, 2015, p. 31–34. [Online]. Available: https://doi.org/10.1145/2818346.2820778

[17] S. Reddy, S. Nalluri, S. Kunisetti, S. Ashok, and B. Venkatesh, "Content-based movie recommendation system using genre correlation," in *Smart Intelligent Computing and Applications*. Springer Singapore, 2019, pp. 391–397.

[18] C. Dupuy, F. Bach, and C. Diot, "Qualitative and descriptive topic extraction from movie reviews using lda," 07 2017, pp. 91–106.

[19] N. Malandrakis, A. Potamianos, G. Evangelopoulos, and A. Zlatintsi, "A supervised approach to movie emotion tracking," *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2376–2379, 2011.

[20] Y. Deldjoo, M. Elahi, P. Cremonesi, F. Garzotto, P. Piazzolla, and M. Quadrana, "Content-based video recommendation system based on stylistic visual features," *Journal on Data Semantics*, vol. 5, pp. 1–15, 01 2016.

[21] R. J. R. Filho, J. Wehrmann, and R. C. Barros, "Leveraging deep visual features for content-based movie recommender systems," in *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017, pp. 604–611.

[22] Y. Deldjoo, M. G. Constantin, H. Eghbal-Zadeh, B. Ionescu, M. Schedl, and P. Cremonesi, "Audio-visual encoding of multimedia content for enhancing movie recommendations," in *Proceedings of the 12th ACM Conference on Recommender Systems*, ser. RecSys '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 455–459. [Online]. Available: https://doi.org/10.1145/3240323.3240407

[23] A. van den Oord, S. Dieleman, and B. Schrauwen, "Deep content-based music recommendation," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 2643–2651. [Online]. Available: http://papers.nips.cc/paper/5004-deep-content-based-music-recommendation.pdf

[24] M. Schedl, "Deep learning in music recommendation systems," *Frontiers in Applied Mathematics and Statistics*, vol. 5, p. 44, 2019. [Online]. Available: https://www.frontiersin.org/article/10.3389/fams.2019.00044

[25] H.-W. Chen, Y.-L. Wu, M.-K. Hor, and C.-Y. Tang, "Fully content-based movie recommender system with feature extraction using neural network," *2017 International Conference on Machine Learning and Cybernetics (ICMLC)*, vol. 2, pp. 504–509, 2017.

[26] S. Wei, X. Zheng, D. Chen, and C. Chen, "A hybrid approach for movie recommendation via tags and ratings," *Electron. Commer. Rec. Appl.*, vol. 18, no. C, p. 83–94, Jul. 2016. [Online]. Available: https://doi.org/10.1016/j.elerap.2016.01.003

[27] V. K. Singh, M. Mukherjee, and G. K. Mehta, in *Multi-disciplinary Trends in Artificial Intelligence.*

[28] K. Bougiatiotis and T. Giannakopoulos, "Enhanced movie content similarity based on textual, auditory and visual information," *Expert Systems with Applications*, vol. 96, 11 2017.

[29] W. Fan, Y. Ma, Q. Li, Y. He, E. Zhao, J. Tang, and D. Yin, "Graph neural networks for social recommendation," *WWW '19: The Web Conference on The World Wide Web Conference WWW 2019*, pp. 417–426, 2019.

[30] Y. Luo, Q. Fu, J. Xie, Y. Qin, G. Wu, J. Liu, F. Jiang, Y. Cao, and X. Ding, "Eeg-based emotion classification using spiking neural networks," *IEEE Access*, vol. 8, pp. 46 007–46 016, 2020.

[31] K. W. Cheuk, Y.-J. Luo, G. Roig, and D. Herremans, "Regression-based music emotion prediction using triplet neural networks," *Proceedings - ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing*, 07 2020.

[32] [Online]. Available: https://www.imdb.com/

[33] [Online]. Available: https://developer.spotify.com/documentation/web-api/reference/tracks/get-audio-features/

[34] C. M. Bishop, "Pattern recognition and machine learning (information science and statistics)." Berlin, Heidelberg: Springer-Verlag, 2006.

[35] T. Giannakopoulos, “pyaudioanalysis: An open-source python library for audio signal analysis,” *PLOS ONE*, vol. 10, p. e0144610, 12 2015.

[36] [Online]. Available: https://plotly.com

[37] [Online]. Available: https://dash.plotly.com

[38] [Online]. Available: https://www.imdb.com/interfaces/

[39] [Online]. Available: https://www.crummy.com/software/BeautifulSoup/bs4/doc/

[40] [Online]. Available: https://www.spotify.com/us/

[41] [Online]. Available: https://developer.spotify.com/documentation/web-api/

[42] T. Giannakopoulos, “Study and application of acoustic information for the detection of harmful content, and fusion with visual information. department of informatics and telecommunications, vol. phd. university of athens, greece.” 2009.

[43] G. Tzanetakis and P. Cook, “Musical genre classification of audio signals,” *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.

[44] [Online]. Available: https://github.com/tyiannak/multimodalAnalysis

[45] [Online]. Available: https://towardsdatascience.com/demystifying-support-vector-machines-8453b39f7368

[46] [Online]. Available: https://www.scipy.org

[47] [Online]. Available: https://librosa.org

[48] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[49] [Online]. Available: https://statistics.laerd.com/statistical-guides/spearmans-rank-order-correlation-statistical-guide.php