ATHENS UNIVERSITY OF ECONOMIS & BUSINESS
DEPARTMENT OF INFORMATICS
MSc IN DATA SCIENCE, FULL TIME 2017-2018

"Emotion-aware content representation and retrieval for movie dialogues"

Capstone Project

Varvara Samoili

Supervisors:

Dr. Ion Androutsopoulos
Associate Professor, Department of Informatics at the Athens University of Economics & Business

Dr. Theodoros Giannakopoulos
Director of Machine Learning at Behavioral Signals, Inc.

In association with:

Behavioral Signals, Inc.

Athens, December 2018

This page is intentionally left blank.

Abstract

The vast variety of information available on today's web has created a need for state-of-the-art Recommender Systems. Apart from collaborative methods, which are based on modeling the similarities between the preferences of different users, content-based retrieval applications for Recommender Systems and User Profiling, particularly in the area of movie recommendations, are also important. Their contribution becomes even more valuable when, apart from static metadata, they also use underlying information related to the content consumed by the user. Furthermore, during the last two decades, Emotion Recognition has peaked the interest of researchers involved in Speech and Text Analytics. Meanwhile, emotion and the way it is conveyed is particularly important in films, as it undoubtedly plays a major role in the final aesthetic result. This leads us to believe that speech emotion in movie dialogues can act as an extra 'dimension' in content-based movie retrieval and recommendation, resulting in emotion-aware content-based movie retrieval.

In this work, we show how specific high-level attributes, which derive from speech emotion estimates in movie dialogues, can constitute a discriminative factor when separating movie content. This is demonstrated through the use of an open and widely used dialogue benchmark, which first undergoes appropriate preprocessing. Experiments show that while, on one hand, emotion-based information alone is not a reliable enough factor for movie retrieval, there is, nonetheless, a statistically significant correlation between the 'emotion-aware' features and high-level movie attributes. Further research should be conducted in order to explore fusion methods of this emotion-based information along with metadata, as well as other types of content-based information (music, vision, etc.) towards the improvement of recommender systems.

Περίληψη

Ο τεράστιος όγκος πληροφορίας που είναι διαθέσιμος στον παγκόσμιο ιστό στις μέρες μας, έχει δημιουργήσει την ανάγκη για τεχνολογικά εξελιγμένα Συστήματα Σύστασης (Recommender Systems). Εκτός από τις τεχνικές Συνεργατικού Φιλτραρίσματος (Collaborative Filtering), στο χώρο των Συστημάτων Σύστασης και ανάλυσης των προφίλ χρηστών, και ιδιαίτερα στον τομέα των κινηματογραφικών ταινιών, υπάρχει επιπλέον ανάγκη για εφαρμογές ανάκτησης οι οποίες βασίζονται στο περιεχόμενο. Η συμβολή τους είναι ιδιαίτερα χρήσιμη όταν, εκτός από στατική πληροφορία από μεταδεδομένα, χρησιμοποιούν και υποκείμενη πληροφορία που σχετίζεται με το περιεχόμενο που καταναλώνει ο χρήστης. Επιπλέον, κατά τη διάρκεια των δύο τελευταίων δεκαετιών, η Αναγνώριση Συναισθήματος έχει προσελκύσει το ενδιαφέρον των ερευνητών που ασχολούνται με ανάλυση φωνής και κειμένου. Σε αυτό το πλαίσιο, το συναίσθημα και ο τρόπος με τον οποίο αυτό εκφράζεται είναι ιδιαίτερης σημασίας στον τομέα των κινηματογραφικών ταινιών, αφού αναμφίβολα διαδραματίζουν πρωταρχικό ρόλο στο τελικό αισθητικό αποτέλεσμα. Αυτός είναι λόγος να πιστεύουμε ότι το συναίσθημα που προκύπτει από τη φωνή στους διαλόγους ταινιών μπορεί να χρησιμοποιηθεί ως μια επιπλέον 'διάσταση' στην ανάκτηση ταινιών βάσει περιεχομένου, έχοντας ως αποτέλεσμα την 'ανάκτηση ταινιών βάσει συναισθηματικού περιεχομένου'.

Στην παρούσα εργασία, δείχνουμε πώς γνωρίσματα υψηλού επιπέδου, που απορρέουν από εκτιμήσεις ανάλυσης συναισθήματος σε φωνή, μπορούν να αποτελέσουν διακριτικό παράγοντα στον διαχωρισμό περιεχομένου ταινιών. Αυτό επιδεικνύεται με τη χρήση ενός ανοιχτού και ευρέως χρησιμοποιούμενου συνόλου δεδομένων διαλόγου, αφού πρώτα έχει υποστεί κατάλληλη προεργασία. Τα πειράματα δείχνουν πως, παρόλο που η πληροφορία που βασίζεται στο συναίσθημα δεν είναι ικανή να χρησιμοποιηθεί ως μοναδικό κριτήριο στην ανάκτηση ταινιών, υπάρχει στατιστικά σημαντική συσχέτιση μεταξύ των συναισθηματικών χαρακτηριστικών και των υψηλού-επιπέδου χαρακτηριστικών της ταινίας. Θα ήταν χρήσιμο να διεξαχθεί μελλοντική διερεύνηση μεθόδων σύντηξης της συναισθηματικής πληροφορίας με μεταδεδομένα, καθώς και άλλα είδη πληροφορίας βασισμένης στο περιεχόμενο (μουσική, εικόνα, κ.λπ.) προς βελτίωση των Συστημάτων Σύστασης.

Acknowledgements

This Capstone Project was realized thanks to and in association with Behavioral Signals, Inc. and particularly thanks to Dr. Nassos Katsamanis, VP of Engineering, and Dr. Theodoros Giannakopoulos, Director of Machine Learning. I would like to thank them both for giving me the opportunity to work on the extremely fascinating field of audio and speech analysis, which is completely new to me.

I would like to give special thanks to Dr. T. Giannakopoulos for his invaluable assistance and devotion to the project, which would have been impossible to complete without his contribution. I would also like to thank my professor, Dr. Ion Androutsopoulos, Associate Professor at the Athens University of Economics & Business, for his guidance throughout the course of this trimester and for always being available to answer my questions.

Last but not least, I would like to thank my parents, Kassiani & Stamatis, for always being there for me and supporting me in every aspect of my life.

Table of Contents

List of Figures

1 Introduction

Today's web, or 'Web 2.0' as called by many, is the new era of Web characterized by an unprecedented information overload, in which content is perpetually distributed and shared not only by a few content creators, as was the case in the past, but also by individual users. In this ocean of information, no content entity comes bare, but rather followed by a swarm of associated metadata to label it, group it with and distinguish it from other entities. In this environment, state-of-the-art systems which take advantage of this type of data for this purpose ('Recommender Systems') have flourished during the recent years.

In the area of motion pictures, Recommender Systems, in general, rely on Collaborative Filtering or Content-based filtering in order to decide on recommendations for a specific user. Collaborative Filtering works based on the assumption that a user A who has consumed similar content to a user B and rated it similarly, is very likely to also want to consume content new to them, but which the user B has already rated positively. User-based Collaborative Filtering systems, despite their popularity, face multiple challenges, such as sparsity, scalability [1] and cold-start [2]. Content-based filtering relies exclusively on information that is specific to the item and the user profile in order to decide on recommendations. 'Content' includes anything from tags, keywords, and general item-specific text (metadata) to more complex content such as music or vision.

Current technologies mainly rely on metadata for the purposes of indexing, and recommendation in hopes that there is a strong correlation between the textual data surrounding media and the media themselves. However, this is not always the case; metadata may be noisy, incomplete or they simply do not tell the whole truth. With that in mind, it seems that it would be useful to try to extract more and better information from more complex content than metadata text.

Meanwhile, Emotion Recognition both in Speech and in Text Analytics, is an area that has attracted a lot of interest during the past two decades. The main motive behind Emotion Recognition in speech has been the improvement of human – machine interaction. Although Automatic Speech Recognition has shown impressive results during the last few years, the inability of the computer to understand the underlying emotional content of spoken utterances keeps us still away from this target. Considerable research is being conducted in the field of Speech Emotion Recognition at the moment, the applications of which are numerous [3]. It is interesting to note that very recently advanced models making use of CNNs and LSTMs have been tested in this field and have shown competitive results [4]. Also, attempts are made at Multimodal Emotion Recognition using fusion features, e.g. from audio and video simultaneously [5]. There are, however, a few reasons why models that have been proposed in the past may have failed to effectively serve this cause so far; these may

have to do with a lack of uniformity in the way they are evaluated, the absence of a universally acceptable testing environment and a lack of cross-domain evaluation methods [6].

Apart from human – machine interaction, emotion information can be extremely useful in content-based representation of media. As far as motion pictures are concerned, it is an indisputable fact that emotion plays a very important role in the aesthetic result of a movie and it is reasonable to believe that this contribution to the aesthetic result might be a deciding factor when recommending a title to a specific user. All in all, emotion can be considered a type of content, extractable from a movie, which carries useful information about the latter and can be used to assess its similarity or dissimilarity to other movies. This information, potentially fused with other types of content-based information, may be used in User Profiling, Movie Indexing or Recommender Systems.

In this work we combine data from a very widely-used, open dataset containing various metadata and scripted dialogues for a number of well-known movies, the Cornell movie-dialogues corpus [7], with a movie repository which contains well-known titles as well, in video format, along with their subtitles. The purpose is to extract movie dialogues in audio format, based on which emotion-aware features will be constructed through a speech emotion recognition classification model, property of Behavioral Signals, Inc. The main goal is to evaluate how these emotion-aware features perform at representing, describing and assessing similarity among movie titles in a way that they would make them useful in applications such as recommendations, user profiling, indexing, etc.

A flowchart of the entire project is depicted in Figure 1. The coding portion of this work is written in the Python programming language.

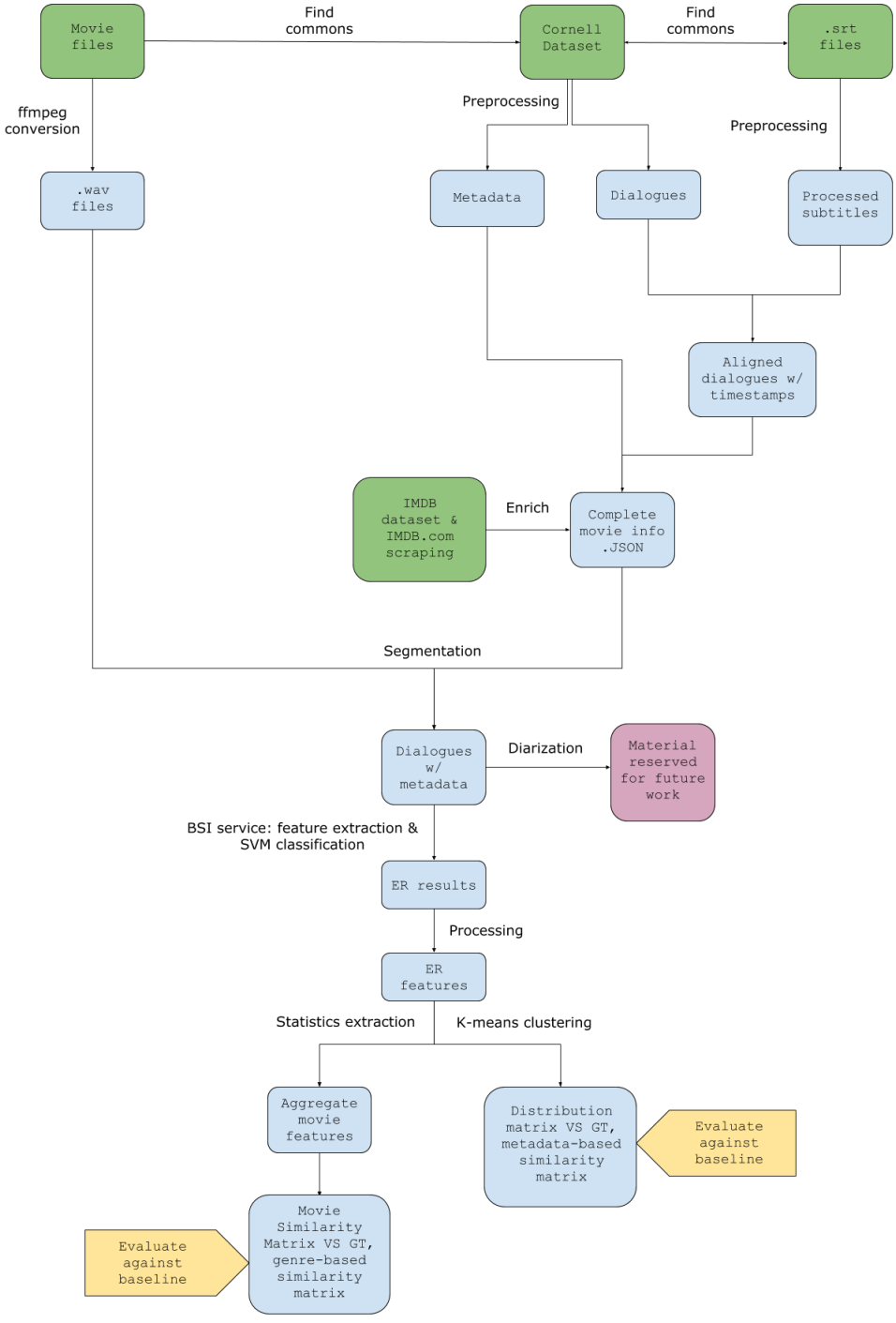

Figure 1: Project flowchart

2 Data

For the analysis we are going to pursue in this work, two types of data had to be combined, movie files and movie metadata. Movie files in .mp4 and .mkv file format, in English, along with their corresponding subtitles in .srt file format, were provided by the company itself. In case any movie file lacked its associated .srt files, those were found on websites that distribute them freely [8, 9] and were synchronized by the author to make sure that they were perfectly aligned to the audio. For movie metadata, the Cornell movie-dialogues corpus [7] was used, which contains both movie-level metadata for a large number of movies, as well as dialogue-level metadata, with character annotations, for fictional dialogues that can be heard during the movie.

Since the two datasets do not contain the exact same titles, the final dataset used for our analysis resulted from the intersection of the two aforementioned ones. After getting rid of titles unsuitable for our task (e.g. movies with subpar .srt files or none at all, musicals, etc.) the final dataset, before any preprocessing, is made up of 225 titles. The information that it contains can be summed up as follows:

- Movie title
- Release year
- IMDb rating
- Number of IMDb votes
- Genres
- Character metadata:
  - Name
  - Gender
  - Position on movie credits
- Dialogue metadata:
  - Actual utterances (lines) based on script
  - Movie character corresponding to each utterance
- Movie file
- Subtitles file

As the Cornell Movie-Dialogues Corpus does not currently include any information regarding movie directors or lead actors, this information was extracted from the IMDb Datasets [10], subsets of which are freely available on the IMDb website for personal or non-commercial use and are updated daily. Naturally, the titles contained in our dataset were not guaranteed to be found in these particular subsets and thus, some information had to be scraped from the website itself, using the associated tags which are common between the website content and the free datasets. As a result, our movie dataset was augmented by the following metadata:

- Directors
- Lead Actors
- Lead Roles

3 Preprocessing

We now have at our disposal movie-specific and dialogue-specific information, as well as actual movie parts, in some video format, along with their corresponding subtitles. Naturally, the video contains a lot of information, only part of which is of importance to the audio analysis - sentiment recognition task, so the first step in the preprocessing stage is to get rid of useless information. To this end, the FFmpeg [11] open source multimedia framework is used to convert all video files to audio files (.wav format, 44,100Hz sampling rate, 16bit depth, single channel), thus getting rid of all visual information.

With regard to the metadata, all the movie-specific and dialogue-specific information are organized in a .JSON data format, the structure of which is described, using an actual example of a movie entity, in Figure 2. In this example, only one scirpted dialogue is shown for visualization purposes. Most fields are self explanatory; the 'time_s', 'time_e' and 'full_wav' fields refer to the alignment stage of the data processing and are filled during that stage, a process which is described in detail in Section 4.1. 'Tag' is an auxiliary field and refers to the unique IMDb tag of the movie. The 'directors', 'actors' and 'roles' fields are all filled using IMDb data, either by scraping the free subsets provided or by crawling the movie pages directly, as mentioned above.

It is worth noting that, while we hope to find the roles and associating actors in the lead roles and actors of the IMDb dataset, this is not guaranteed to happen for every single dialogue. In the example shown in Figure 2, this particular dialogue happens to take place between two lesser roles, which do not appear in the 'roles' field of this movie. Luckily, the majority of the dialogues were successfully annotated, enabling us to take advantage of this extra piece of information.

```
[{
        "movie_id": "m2",
        "movie_title": "15 minutes",
        "file_path": "./Movies/15 minutes",
        "movie_filename": "15 minutes.wav",
        "srt_filename": "15 minutes.srt",
        "dialogs": [{
                "dialog_id": 3,
                "time_s": 5911.2,
                "time_e": 5916.164,
                "lines": [{
                        "line_id": "L3459",
                        "text": "I brought you some letters.  It's really fan mail.  Women
                        mostly.
                                One wants to buy you clothes, another sent a check. Another
                                wants a check.",
                        "main_speaker_name": "CUTLER"
                    }, {
                        "line_id": "L3460",
                        "text": "You bring the cigarettes?",
                        "main_speaker_name": "EMIL"
                    }, {
                        "line_id": "L3461",
                        "text": "Oh, sure.",
                        "main_speaker_name": "CUTLER"
                    }
                ],
                "full_wav": "./Movies/15 minutes/shorts\\15 minutes_5911.2_5916.16.wav",
            }, {
            ...
            }, {
            ...
            }
        ],
        "tag": "tt0179626",
        "directors": ["John Herzfeld"],
        "actors": ["Robert De Niro", "Edward Burns", "Kelsey Grammer", "Avery Brooks"],
        "genre": ["thriller", "action", "drama", "crime"],
        "roles": [["Detective Eddie Flemming"], ["Fire Marshal Jordy Warsaw"], ["Robert
        Hawkins"], ["Detective Leon Jackson"]]
    },{
    .....
    },{
    .....
    },
]
```


Figure 2: Metadata .JSON structure

4 High-level data processing

As one would expect, an one to three hour long movie generally contains a variety of events, not just dialogues. Music, singing, background noise, long periods of silence, etc. are all present in most mainstream movies. These are not only redundant for our task, but also impeding, in some cases, which is why they need to be cut off of the original audio.

4.1 Dialogue level audio-text alignment

Audio segmentation is the process of dividing an audio file into multiple smaller segments, which are considered homogeneous with regard to some objective, such as speaker, emotion, scene or audio event. Depending on the application, this may mean that they all contain speech versus other segments that contain silence or music, they belong to the same speaker versus other speakers, or they follow a recurring pattern that is present in other parts of the audio (e.g. a song chorus), etc. In audio analysis, audio segmentation can be an unsupervised or a semi-supervised process and can even make use of prior information that may be available about the segments (e.g. number of speakers).

The main type of segmentation that the movie audio needs to undergo is a dialogue-level segmentation so that it can be matched to its corresponding dialogue-based metadata. The desired result of this process is a .wav segment that corresponds to a single scripted dialogue, i.e. if a movie is made up of 10 dialogues which are present in the corpus, the result is 10 .wav segments, each one of which is several seconds long and contains exactly one dialogue. In theory this could be carried out by a speech segmentation machine learning algorithm, but choosing this method poses two problems. First of all, there is no available annotated dataset for a ML algorithm to learn off of, so the task would have to be unsupervised and on top of that, there is no way to reliably evaluate the result. One could also make a case for high heterogeneity within each cluster, e.g. singing versus speaking, speech with various levels of energy, different types of background noise, etc. Secondly, even if we did have annotated datasets at our disposal for classification, a movie audio is sure to contain multiple different classes (speech, music, silence, special effects, etc.), rendering the classification task that much harder.

Here, the problem of audio segmentation for dialogue extraction is solved by using a simple model guided by the dialogue text provided by the Cornell Movie-Dialogues Corpus. This task may appear simple seeing that synchronized subtitles are available for each audio file, but it is not trivial. It is worth noting here that a .srt file contains the start and end timestamps between which the subtitle should appear on the screen, as seen in Figure 3. It would seem that a simple search of whole dialogues in the .srt

text would be enough, except that, in reality, a scripted dialogue is rarely identical to the actual lines uttered during filming, which means that the two texts are pretty similar but far from identical. An example of this can be seen in Figure 4.

```
731
01:03:31,006 --> 01:03:33,925
Is it hard to get to be in
the Secret Service?
```

Figure 3: Example of subtitle in a .srt file

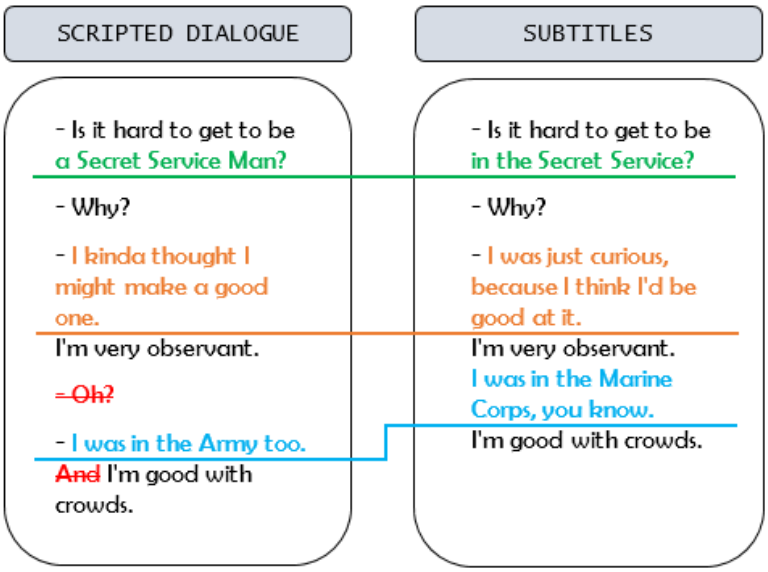

Figure 4: Example of differences between a scripted dialogue and its corresponding movie subtitles

There are at least three identifiable types of transformations that can occur in the scripted dialogue; insertion, deletion and substitution. Insertion and deletion refer to adding or removing entire sentences, respectively, while substitution refers to syntactically altering a sentence in a way that preserves the meaning. Except for the obvious effect these kind of transformations have on the scripted text, there is also

the problem of altering the speaker turn, which is the hardest one to overcome when it comes to this kind of text alignment.

During the dialogue level audio-text alignment the goal is not to exactly capture all the subtitles that correspond to the scripted dialogue; capturing part of the dialogue is good enough, even if some of the beginning or ending tokens are "cropped". The goal here is, mainly, to ensure that the captured segment is an actual dialogue and secondly to match the metadata that belong to the scripted dialogue to a part of the .srt file, which logically shares the same metadata.

In order to achieve this alignment, a simple model is used. First of all, both the contents of the .srt file and the scripted dialogue are converted to lower case, tokenized and stripped of punctuation marks. The whole scripted dialogue is then compared to each subtitle separately based on a simple sentence similarity metric:

$$Sentence\_Similarity = \frac{length(SD\_tokens \cap srt\_tokens)}{length(srt\_tokens)}$$

Subtitles that achieve higher similarity than a preset threshold are candidate matches to the scripted dialogue. It is worth noting that a metric such as Jaccard similarity, although more intuitive, would not work here, since the length of each subtitle and, by extension, the length of the union

$$length(srt\_tokens \cup SD\_tokens)$$

is not fixed, so setting a threshold is non-trivial. Through trial & error we concluded that a threshold of 0.8 is a good value for maintaining an extremely high precision (over 99%) and a pretty acceptable recall of approximately 47%.

4.2 Audio Segmentation

Based on the results of the dialogue level audio-text alignment, the information needed for the segmentation is now available. Using the start and end timestamps of the first and last subtitles that make up the dialogue, respectively, we can now produce one .wav segment per dialogue using the FFmpeg tool. A different number of segments is produced for each movie, depending on how many scripted dialogue excerpts are available for it in the corpus; this number can vary from 5 to 181. The new .wav files share the same features as their 'parent' .wav files (44,100Hz sampling rate, 16bit depth, single channel).

4.3 Emotion-aware content representation

As mentioned above, the scope of this work is to evaluate emotion-based labels as features as far as their descriptive and discriminative capability goes, specifically in

the task of predicting movie similarity and clustering movies with regard to their metadata. This information of emotion-based labels is, of course, neither present in nor directly derivable from our current dataset.

In order to extract this kind of information, an emotion recognition classifier is needed, which will accept as input appropriate features and return proper emotion-based labels. Choosing the type of features for the classification task and defining the desired emotion output is far from trivial.

There is a large theoretical background behind what types of features should be used in audio analysis depending on the problem at hand; not all audio signals are the same but, also, different types of information can and should be extracted from a single piece of audio signal in order to serve the particular task.

However, feature selection and model development for emotion recognition audio analysis is outside the scope of this work; Behavioral Signals has provided the author with a pre-trained classification service that accepts as input audio segments, extracts the required hand-cracted features and returns a .JSON file with various data, results of the emotion recognition task. The details behind the service are withheld from the author and constitute literary property of the company, however there is some information available regarding the types of features that are extracted from the audio segments.

There are three 'families' of hand-crafted features that are computed by the service. We are not going to go into detail with regard to their natural significance or how they are calculated, since these topics are exhaustively covered in the literature [12, 13], but it is useful to give some representative examples:

1. Time Domain Features:
   (a) Energy
   (b) Zero-Crossing Rate
   (c) Entropy of Energy
2. Spectral Domain Features:
   (a) Spectral Centroid & Spread
   (b) Spectral Entropy
   (c) Spectral Flux
   (d) Spectral Rolloff
3. Mel-frequency Cepstral Coefficients (MFCCs): Although MFCCs typically fall under the category of Spectral Domain Features, they are referred to separately for two main reasons; firstly, their natural significance is not as straightforward

as the aforementioned quantities and secondly, they are extremely popular in the field of speech processing. They are calculated using the cepstral representation of the audio signal, namely the Inverse Fourier Transform of the logarithm of its spectrum.

With respect to the classification model, a combination of typical statistical classifiers on hand-crafted features and deep audio feature extractors was used, which was trained on movie dialogues, but has undergone cross-domain evaluation (movies, phone calls, dialogues and human - computer interaction). The trained version of this model was used to extract the emotion-based labels.

With all that said, it would seem as if the emotion recognition model output is predetermined for us, with no room for emotion-based feature selection. This is not true, however, since the model provides an extremely information-heavy output and it is reasonable to believe that the model is not equally 'certain' about every piece of information that is part of this output. This is why the user needs to select wisely which pieces to keep and which ones to discard.

More specifically, for a particular segment, the model divides the audio signal into 'frames' of 100 milliseconds each and makes decisions for each frame. For example, for an audio segment (in our case, a dialogue) 4.5 seconds long, the model produces 45 frames, each accompanied by an emotion-aware associated decision. This decision is, as mentioned above, information-heavy; it consists of a variety of labels, not only emotion-based, but also gender, tone, language, etc. An example of the classification output for a single frame can be seen in Figure 5. The two labels which are of interest to us are 'arousal' and 'valence'.

Arousal and valence are not exactly straightforward emotions, but rather emotion dimensions that are derived from the psychology of production and perception of emotions. Both arousal and valence are explicitly defined psychological terms; delving into and grasping their definitions and their importance to affective science is beyond the scope of this work, but details can be found in the literature [14]. However, in very broad terms, one could describe valence as the 'pleasure' (positive valence) or 'displeasure' (negative valence) behind an emotion and arousal as the 'calmness' (negative arousal) or 'excitement' (positive arousal) of an emotion. Not only is it possible to map high-level straightforward emotions, like happiness or anger, in terms of the valence-arousal duo, but we could also use the latter to furthermore describe events, objects or situations, as shown in Figure 6.

Figure 5: Example of the Emotion Recognition Service output for a single frame. The .JSON structure is split in three columns for visualization purposes. Note that 'positivity' is a label for valence and 'strength' is a label for arousal.

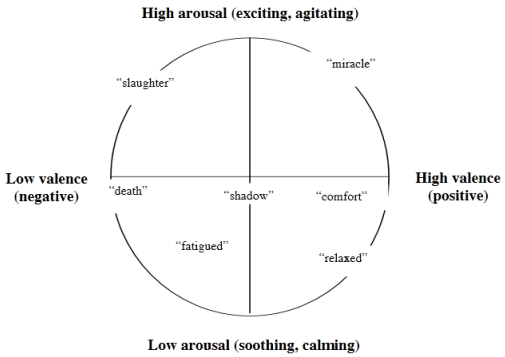

Figure 6: 2-dim emotion description using valence and arousal dimensions of emotion. Image taken from Kensinger [15].

Having experimented with different audio features, classification models and emotion representations as class labels, the company's machine learning experts have concluded that, up until now at least, emotion representation in terms of valence and arousal can, in general, achieve higher classification accuracies than discrete emotions (e.g. 'anger', 'sadness', etc.) can, at least up until now. The models developed by the company can boast one of the highest performances in terms of emotion recognition based on speech at the moment when compared to state-of-the-art technologies, which is around 60%. Acquiring more and better cross-domain data is going to enable them to build even better models which are certainly going to achieve higher accuracies, even for discrete-emotion-based classification and this is one of the company's main plans in the near future.

4.4 Feature extraction

Back to our particular problem, we have yet to define features to evaluate the emotion representation described in the previous section. We are now looking at one .JSON output per dialogue, each one of which contains a variety of information concerning each 100ms-long frame of the dialogue segment. This information needs to undergo some processing to be useful to our task.

The first course of action is to discard all information besides valence ('positivity') and arousal ('strength') by parsing the .JSON files. As a result of this process, the information is still divided in a number of frames for each dialogue segment. However, we only need a single vector representation for each dialogue segment, so these results have to be aggregated in some manner.

Since this is the result of a classification model, valence and arousal are not continuous quantities, but rather have discrete values of 0.0, 0.5 and 1.0 which correspond to 'negative', 'neutral' and 'positive', respectively. This is the case for both emotion

dimensions. The aggregation to a dialogue-specific vector is performed by calculating the percentage of the occurrence of each label for a specific emotion dimension against the other two labels. For example, if the output consists of 100 frames, 70 of which were labeled as negative, 20 as neutral and 10 as positive, valence-wise, then the result is of the form:

[0.7, 0.2, 0.1]

The same goes for the arousal emotion dimension; if, for the same dialogue, we encounter 12 negative, 28 neutral and 60 positive labels, arousal-wise, the array representation now becomes:

[[0.70, 0.20, 0.10], [0.12, 0.28, 0.60]]

However, since each of these two vectors always sum up to one, one of the three elements is redundant, as it is linearly dependent on the other two. We can, of course, arbitrarily choose which one to remove; here the neutral label is dropped from both vectors and the final, flattened vector becomes:

[0.70, 0.10, 0.12, 0.60]

Apart from the feature representation of a single dialogue, a feature representation of the entire movie is also needed. For this purpose, an aggregation on all the feature vectors of the movie is performed. Here, simply calculating the mean of each feature is not enough, since this would strip the final feature vector (from now on referred to as 'aggregate feature vector') of the information regarding the distribution of emotion in the movie. It is also not reasonable to assume that a movie is described in each entirety by one particular emotion or emotion dimension on average. For example, in an action movie we expect to encounter a number of dialogues with a high value of the arousal-positive feature. However, there will surely be dialogues non action-based, with negative or neutral arousal. This is why we need to describe the distribution of each particular feature by appropriate statistics, not just the mean. Here, a total of 4 statistics are used, namely the mean, variance, 25^{th} and 75^{th} percentile. The final aggregate feature vector is a 1x16 dimension vector which can be summed up as follows:

[val_neg_mean, val_neg_variance, val_neg_25th_perc, val_neg_75th_perc,

val_pos_mean, val_pos_variance, val_pos_25th_perc, val_pos_75th_perc,

ar_neg_mean, ar_neg_variance, ar_neg_25th_perc, ar_neg_75th_perc,

ar_pos_mean, ar_pos_variance, ar_pos_25th_perc, ar_pos_75th_perc,]

where 'val' is short for valence, 'ar' for arousal, 'neg' for negative, 'pos' for positive and 'perc' for percentile.

5 Evaluation methods

Before getting into what methods are going to be used in order to evaluate the emotion-aware features we now have at our disposal, it is important to discuss the concept of similarity among movies. Similarity is far from trivial to define in this context, as it is a rather subjective; different people, when presented with an example movie title, will come up with different answers when asked which titles they consider similar to the example title. Two titles may be deemed similar when sharing common genres, a common director, or just because they are considered 'classics', or both have a lot of award nominations. In this work, as far as similarity goes, it is assumed that titles that share similar metadata (directors, actors, genres), in general, are similar. Metadata is therefore treated as a 'ground truth' against which we are going to evaluate the emotion-aware features that were introduced in the previous section of this work. The hypothesis is that these emotion-aware features are going to be able to simulate to an extent this metadata-derived similarity.

All in all, we need to propose a methodology which involves a model, a baseline model and a ground truth; the evaluation process is made up of two general steps, each of which is comprised of these three elements. The general evaluation framework is depicted in Figure 7.

As a first step, a model based on the concepts of Information Retrieval is proposed – from now own referred to as 'Emotion-Aware Movie Retrieval' or 'EAMR' – in order to evaluate how the features perform at predicting movie similarity. For this purpose, the cosine similarity metric is going to be used, as given by the following equation:

$$Cosine\ Similarity\ =\ 1 - \frac{a \cdot b}{||a|| \cdot ||b||}$$

where a, b the feature vectors corresponding to the two entities between which the similarity is calculated. Let us note that the choice of cosine similarity over other similarity metrics, such as the Euclidean distance for example, is arbitrary.

The EAMR model involves constructing features based on the available ground truth information and calculating the cosine similarity between all pairs of titles for which we have emotion-based information available. Each movie is represented by a binary feature vector and each metadata category acts as a categorical variable; for instance, if a movie is labeled as 'action' and 'adventure' genre-wise, then its ground truth feature vector will be 1 in the 'action' and 'adventure' indices and 0 elsewhere. This can of course be extended for every piece of metadata-based information there is available about this particular title. In this model, only movie genres are considered part of the ground truth; what is more, because movie genres were deemed too specific for this purpose, aggregation is performed on all unique genres that can be found in the metadata, so that they form 'teams' of genres that are more general, yet preserve

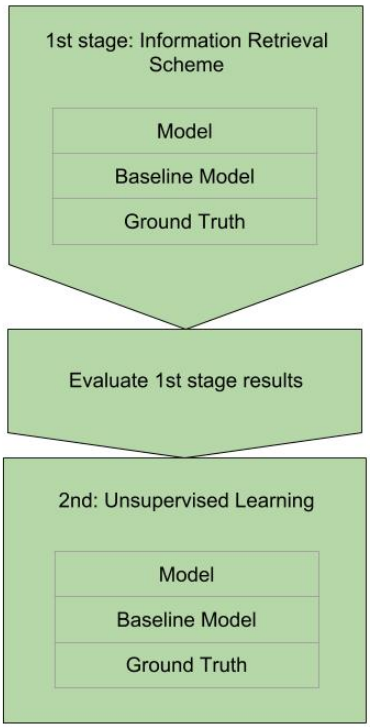

Figure 7: Evaluation framework

the logical similarity that a viewer might expect to identify among the unique genres. In this manner, if a title belongs to the 'fantasy' and 'action' genres, its ground truth feature vector in this case will be as follows:

$$\mathbf{[1,\ 1,\ 0,\ 0]}$$

The aggregation performed on the unique genres is shown in Figure 8. All in all, the similarity which is calculated in this manner acts as the ideal similarity-target we would like to be able to reach using our emotion-aware features.

Following that, using the aggregate feature vectors that were introduced in the previous section in order to represent entire titles, we recompute the similarity between all title pairs and compare against the ground truth. In order to evaluate the performance of the model, a baseline model is needed; in our case, assigning a random feature vector to each title is pretty reasonable. Using the randomly assigned feature vectors, the similarity is calculated and compared against the ground truth.

If the first step in the evaluation process supports the claim that emotion-aware features are fit, to an extent, to predict similarity among titles, we go on to the second step, which is an unsupervised learning model (clustering); what we want to assess here is the ability of the emotion-aware features to assign a specific title's dialogues to the same cluster as the metadata-based features, which act as the ground truth. For this purpose, the very popular K-means algorithm is used for clustering, as established in [16] and the implementation used is the scikit-learn one [17]. As baseline, a random distribution of each dialogue to one of the k clusters is used. The actual significance of each cluster is not important, as long as one is able to identify a unique cluster across the three different results: model, ground truth and baseline.

```
{
  'team_1': ['crime', 'thriller', 'action', 'war', 'horror', 'mystery',
             'animation', 'noir'],
  'team_2': ['fantasy', 'adventure', 'sci-fi'],
  'team_3': ['biography', 'drama', 'history', 'western'],
  'team_4': ['comedy', 'family', "sport"]
}
```

Figure 8: Aggregation performed on the available movie genres

6 Results & Discussion

6.1 Evaluation Metrics

Before presenting detailed evaluation results, let us first refer to the evaluation metrics used for this task. During the first stage, namely the EAMR model, as described in the previous section, typical Information Retrieval metrics are used, most of which are based on the most popular evaluation metrics used in Collaborative Filtering Recommender Systems, namely precision and recall [18].

6.1.1 Precision@k & Mean Precision@k

In the context of evaluating an information retrieval system, Precision@k in general, shows how many documents are relevant out of the k first that the model recommended, as shown below:

$$\frac{\#\ of\ recommended\ items\ that\ are\ relevant@k}{\#\ of\ recommended\ items@k}$$

In our case, k is chosen to be 5, arbitrarily, but still far from 77, which is our movie total. Also, the number of "recommended items", which is essentially the number of similar titles as proposed by the model, is always 5 in our case, since all titles are assigned a similarity number. So, for example, if the model finds 3 titles, in the first 5 places, in common with the ground truth, then the precision is 3/5, or 0.6. When evaluating our model, there corresponds one Precision@k value to each title, so the Mean Precision@k is also calculated, which is the mean over all titles. In the case of the baseline model, the Precision@k for each title is calculated over 100 iterations and the overall mean is assigned to each title; then the Mean Precision@k is calculated as before.

6.1.2 Success@k

This metric is the most lenient among the three, as it is only concerned with the most relevant item appearing anywhere in the first k items recommended by the model; it is defined as the ratio of the number of titles that had the most relevant item appear in the first k (here, 5) recommendations over the number of all available titles:

$$\frac{\#\ of\ titles\ with\ 1^{st}\ most\ relevant\ appearing\ in\ first\ k\ recommendations}{\#\ of\ all\ titles}$$

In contrast to the two aforementioned metrics, this is calculated over all titles, so once for the main model and once for the baseline.

6.1.3 Mean Reciprocal Rank (MRR)

The Reciprocal Rank of a query, in general, is the multiplicative inverse of the rank where the model found the most relevant item. For example, if ground-truth-wise the most relevant item is found in 5^{th} position by the model, then RR is 1/5 or 0.2 for this particular query. When looking at all queries, the Mean Reciprocal Rank is defined as the mean of the Reciprocal Ranks. In our case, the metric assesses at which rank each individual most similar title was found by our model and calculates the overall mean as follows:

$$MRR = \frac{1}{n\_titles} \sum_{i=1}^{n\_titles} \frac{1}{rank_i}$$

6.1.4 Silhouette

Silhouette analysis is used for the 2^{nd} part of the modeling process, the K-Means Clustering. The Mean Silhouette Coefficient metric used in this work is the one introduced in [19] and the implementation used is the scikit-learn one [17].

The Silhouette Coefficient is calculated for each sample and uses the mean intra-cluster distance, a, and the mean nearest-cluster distance, b, as follows:

$$SC = \frac{b - a}{max(a, b)}$$

Silhouette Coefficient assumes values in the interval [-1,1], 1 indicating a point far away from the neighbouring clusters, 0 indicating a point very close to the decision boundary and -1 indicating a strong probability that the point is wrongly assigned to the particular cluster. Here we are going to use the Mean Silhouette Coefficient of all clusters to decide on the number of clusters in which the K-Means algorithm is going to divide the dataset items (or the individual dialogues).

6.2 Results

6.2.1 Emotion-Aware Movie Retrieval Model

The results from calculating the cosine similarity between each pair of titles can be summed up in a similarity matrix. Both the matrix rows and columns are made up of the 77 available titles, hence the matrix is symmetrical and its diagonal is 1 everywhere, as expected. The cosine similarity is calculated, as mentioned in the previous section, between two titles' feature vectors for all possible pairs.

Here, it is assumed that the ground truth, as far as cosine similarity between two titles goes, is the value calculated based on the titles' 'ground truth feature vectors'.

Based on these, a 'ground truth' similarity matrix is constructed, which acts as the ground truth for evaluating our emotion-aware features.

A cropped version of the matrix, that involves only 10 out of 77 the titles, can be seen in Figure 9. The complete matrix is 77x77 in size, so the crop serves visualization purposes. Also, instead of the actual movie titles, the movie IDs are depicted on each axis. A comprehensive table containing all 77 movie IDs and their corresponding titles can be found in Appendix A, for reference.

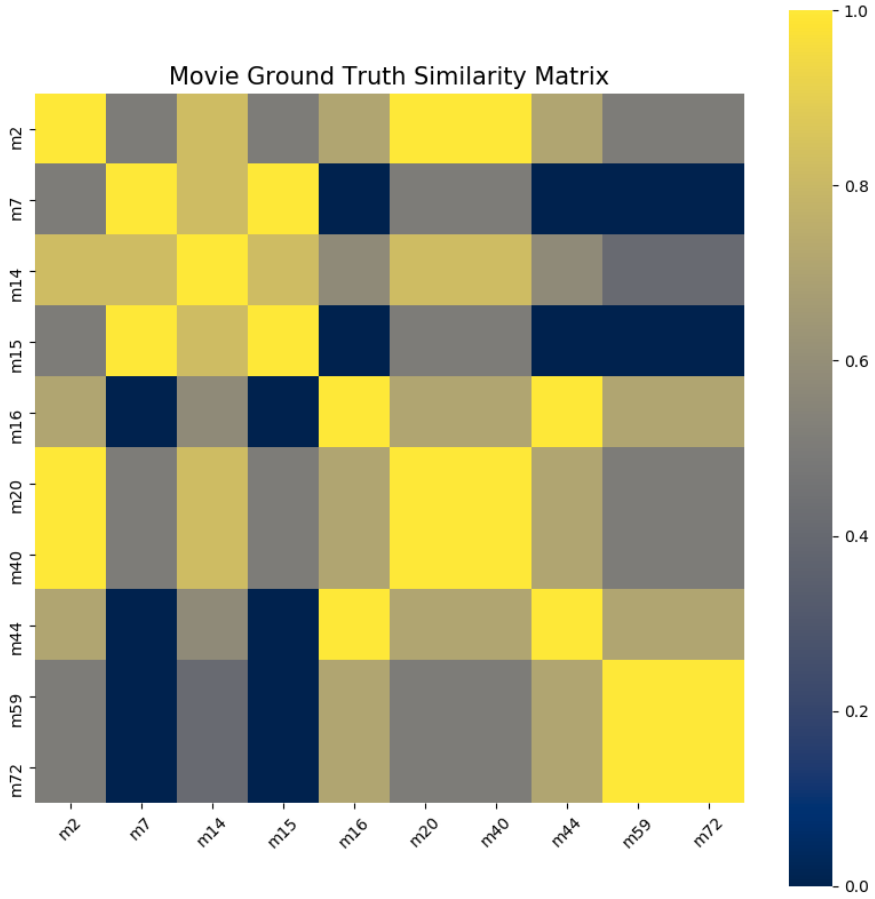

Figure 9: Cropped 'Ground Truth' similarity matrix

Two more such matrices are produced, one based on the emotion-aware features and one based on randomly generated features; the latter serves as the baseline model. The metrics which were introduced in Section 6.1 are used to evaluate each model and the results are presented in Table 1. Note that the results for the Baseline Model are overall means over 100 iterations.

With regard to Precision@5 we can also see more detailed results of the experiment in Figure 10, where results for each title are documented. As far as the random results go, only one random instance out of the 100 iterations is depicted in the barplot.

Table 1: Results from the EAMR Model and the corresponding Baseline Model

Metric	EAMR Model	Baseline Model
Precision@5	0.1117	0.0621
Success@5	0.1300	0.0606
MRR	0.0834	0.0641

There are a few things we can derive from these results. Firstly, based on the Success@5 values, when looking at the similarity matrix produced by the EAMR model, if one chooses a title and looks at the 5 most similar ones as proposed by the model, they are more than twice as likely (0.1300) to find the actually most similar title in them, than trusting the baseline model (0.0606). What is more, based on the Precision@5 value, one is going to find almost twice as many (0.1117) titles in the top 5, that actually belong there, trusting the EAMR model than trusting the baseline model (0.0621). Finally, When ranking the actually most similar title, the EAMR model performs slightly better (0.0834) than the baseline model (0.0641), but enough to be able to distinguish it from randomness.

Figure 10 further supports these results; it depicts in how many more distinct cases (titles) and by how much the EAMR model (blue bars) outperforms the baseline model (orange bars).

Similar results with 'ground truth' feature vectors were produced, where actors, directors and individual genres are taken into consideration and no aggregation is performed. Since they are not substantially different from the ones presented here, we have no reason to believe that one ground truth is any different from the other. This is probably due to the fact that 77 titles are rather few, so there are not a lot of titles that share common directors or actors. This means that the genres carry more weight as features than actors or directors do. At the Clustering stage of this process, the former ones are used.

Some qualitative examples of the EAMR model's performance are summed up

in Figure 11. Query movies are denoted with black, relevant hits are denoted with green, while irrelevant ones with red. Most results are self-explanatory; the one that could potentially stand out is 'Tombstone', a 1993 movie starring Kurt Russell, which would probably not be relevant to the other two in its cluster genre-wise, since it is a classic western film. Emotion-wise, however, it is a crime-heavy movie, the dialogues of which are expected to convey emotion dimensions close to other crime movies in its bubble (low valence, high arousal). Another such example is 'The night of the Hunter' in the Horror group. Genre-wise this movie would never have been found similar to the other items of the group, since it is a film-noir movie by genre. However, emotion-wise, the film has a strong haunted underlay with suspenseful dialogues that typically characterize horror films and it definitely belongs to this group based on its aesthetic result. Examples such as these hint at the necessity of content-based recommendations in the movie recommender systems field.

Another interesting visual representation of the results is depicted in Figure 12. The plot is a result of projection of the 'aggregate' emotion-aware features on a 2D space. Examples in green show titles that were grouped together based on their emotional content and are indeed similar. Examples in grey are outliers that should have been found closer to other groups. Finally, examples in red are mostly titles irrelevant to each other that were grouped together. For example, 'Shakespear in love' is a 1998 drama - romance, which is grouped together with 'Total Recall', a 2012 action packed adventure and 'The adventures of Ford Fairlane', a 1990 action packed adventure-comedy, which are obviously similar to each other. Furthermore, 'Gandhi', a 1982 biographical title of the famous politician - activist, hardly resembles the sci-fi/action titles it is grouped with. The rightmost group is rather interesting; its place on the 2D projection hints at high values in one of the two emotional dimensions, probably arousal. Taking theese particular titles into consideration, this leads us to believe that this is not an example of a failed emotional classification, but rather a true case where emotional content may not be the best judge of general thematic content.

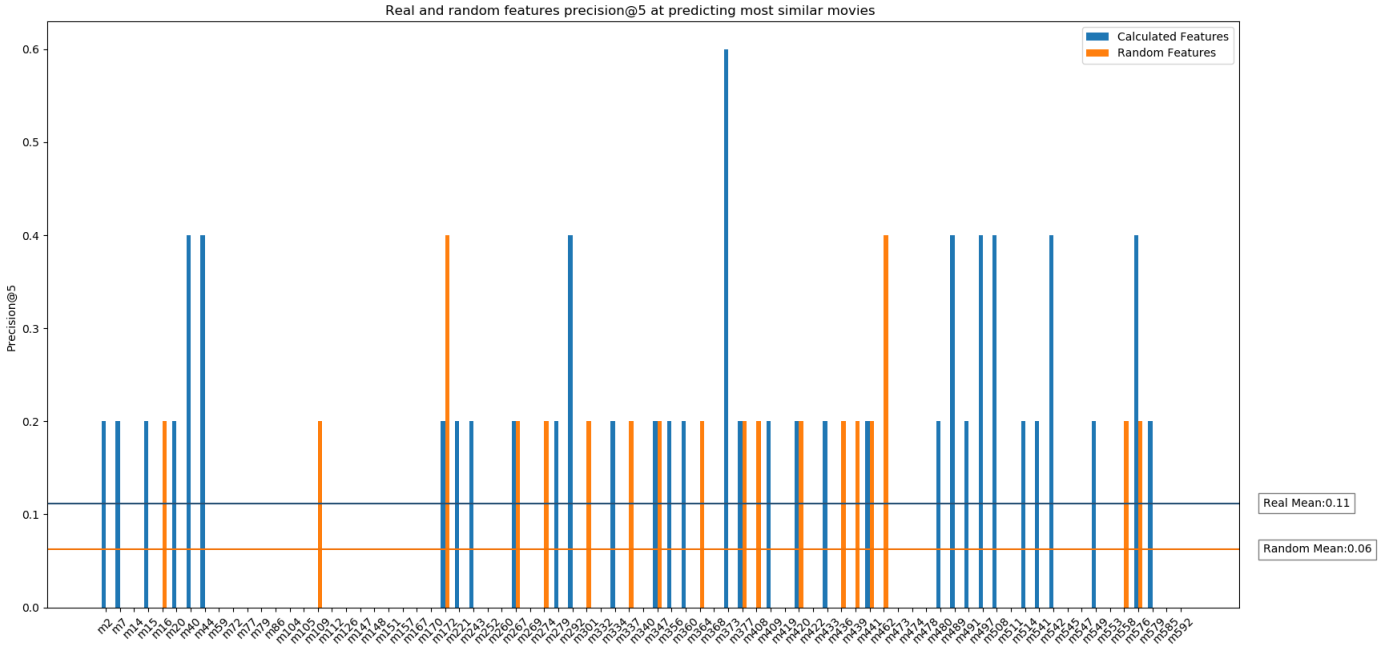

Figure 10: EAMR Model (blue) & Baseline Model (orange) precision@5 for all individual titles

Figure 11: Qualitative results of the EAMR model. Queries are denoted with black, relevant hits with green, while irrelevant ones with red.

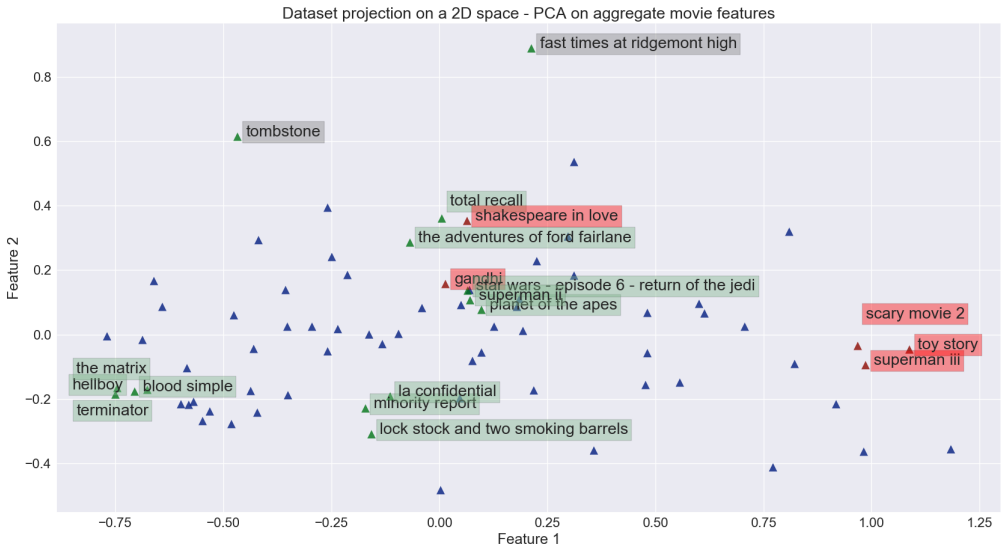

Figure 12: 2D representation of the 'aggregate' emotion-aware features after PCA. Some representative examples are shown in labels. Green ones denote relevance, while red ones, irrelevance; grey markers note some outliers;

All in all, the evaluation of the EAMR model dictates that there is some use-

ful information in the emotion-aware features and definitely some correlation to the metadata-based features. Based on these results, it is reasonable to continue on with the 2nd stage of the modeling process, which is the application of an Unsupervised Learning Model, as described in the previous section.

6.2.2 Unsupervised Learning Model

This part of the modeling process follows a similar logic, as described in the previous section, in the sense that there are three types of features based on which the K-Means algorithm is going to divide the dialogue set. The result of each clustering process is a distribution matrix, which shows what portion of dialogues from each title is assigned to which cluster.

The 'ground truth' distribution matrix is the one produced by the clustering performed using the available metadata for each title as feature vectors, in a manner similar to what was described in section 5, but without the aggregation; in the feature vector there is now information about actors, directors and individual genres.

The model's distribution matrix is produced by the clustering performed using the standard emotion-aware feature vectors for each dialogue, not the aggregate ones, for obvious reasons. The goal is to see how many titles the model assigns to the same cluster as the ground truth clustering (majority of dialogues).

The baseline distribution matrix is a randomly produced distribution matrix. Each title - cluster pair is assigned a random value in the interval [0,1], from a uniform distribution in such a way that, for each title, the values sum to 1.

First, the algorithm is tuned in order to determine the optimal number of clusters based on the Mean Silhouette Coefficient. The options are 2, 3 and 4 clusters; 4 is the number of the 'genre teams' that were produced from the aggregation, as described in 5, which are considered ground truth, so it is reasonable to believe that the maximum number of clusters coincides with this number. Table 2 shows the Mean Silhouette Coefficient for each number of clusters.

Table 2: Mean Silhouette Coefficient

# of Clusters	Mean Silhouette Coefficient
2	0.4303
3	0.4660
4	0.4217

The highest Mean Silhouette Coefficient is achieved when selecting 3 clusters. Firstly, the ground truth features are used to train a K-Means algorithm, setting the number of clusters at 3 and based on the 4746 ground truth feature vectors available in the training set; the output is the ground truth distribution matrix. Following that, the main model is similarly applied, but now the new K-Means algorithm is trained based on the 4746 emotion-aware feature vectors; the main model's distribution matrix is compared to the ground truth distribution matrix. Only the 'successful' distributions are documented and visualized. In Figure 13 the ER (main model) distribution matrix versus the GT distribution matrix are depicted, while in Figure 14 the corresponding matrices for GT versus (one instance of) the baseline model are shown.

In order to make sure that the clusters are indeed aligned, the following experiment is performed; the K-Means algorithm is applied to the dataset, firstly based on ground truth features only and then based on features that contained both the ground truth and the emotion-based information. When comparing the resulting distribution matrices, almost none of the titles is re-assigned on a different cluster. The clustering remains almost exactly the same qualitatively, with only a few quantitative changes.

Back to the results at hand, it is obvious that the emotion-aware features outperform the baseline at assigning the majority of each title's dialogues to the correct cluster. The ER features assign 32 out of 77 titles correctly, while the random features, when tested over 100 iterations, assign 25 out of 77 titles correctly, on average (as expected).

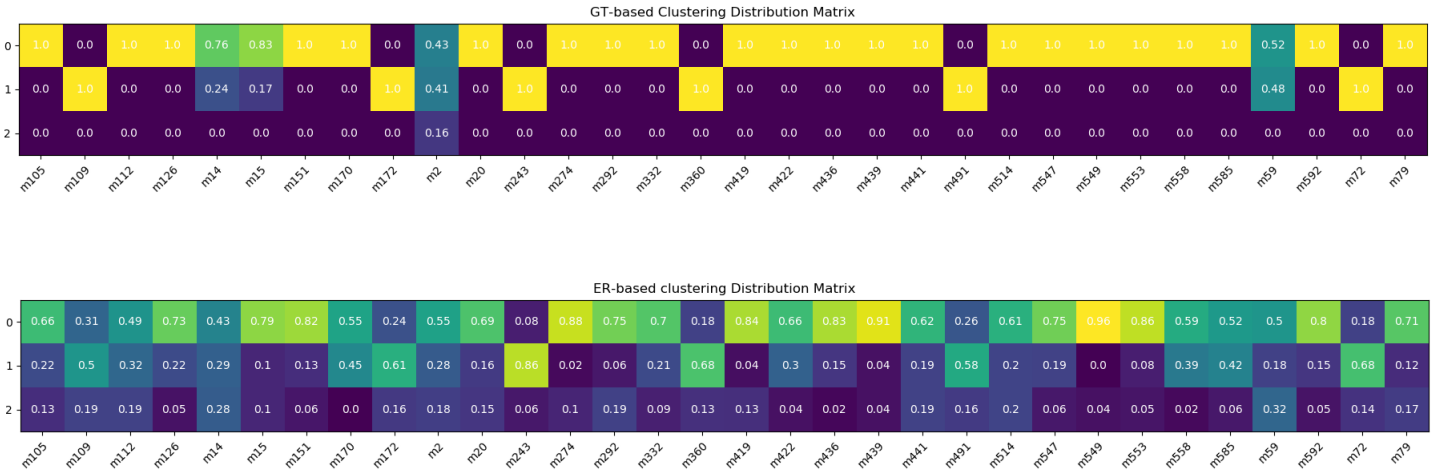

Figure 13: Ground Truth Distribution Matrix vs Emotion-Aware Distribution Matrix - only the correct matches are shown

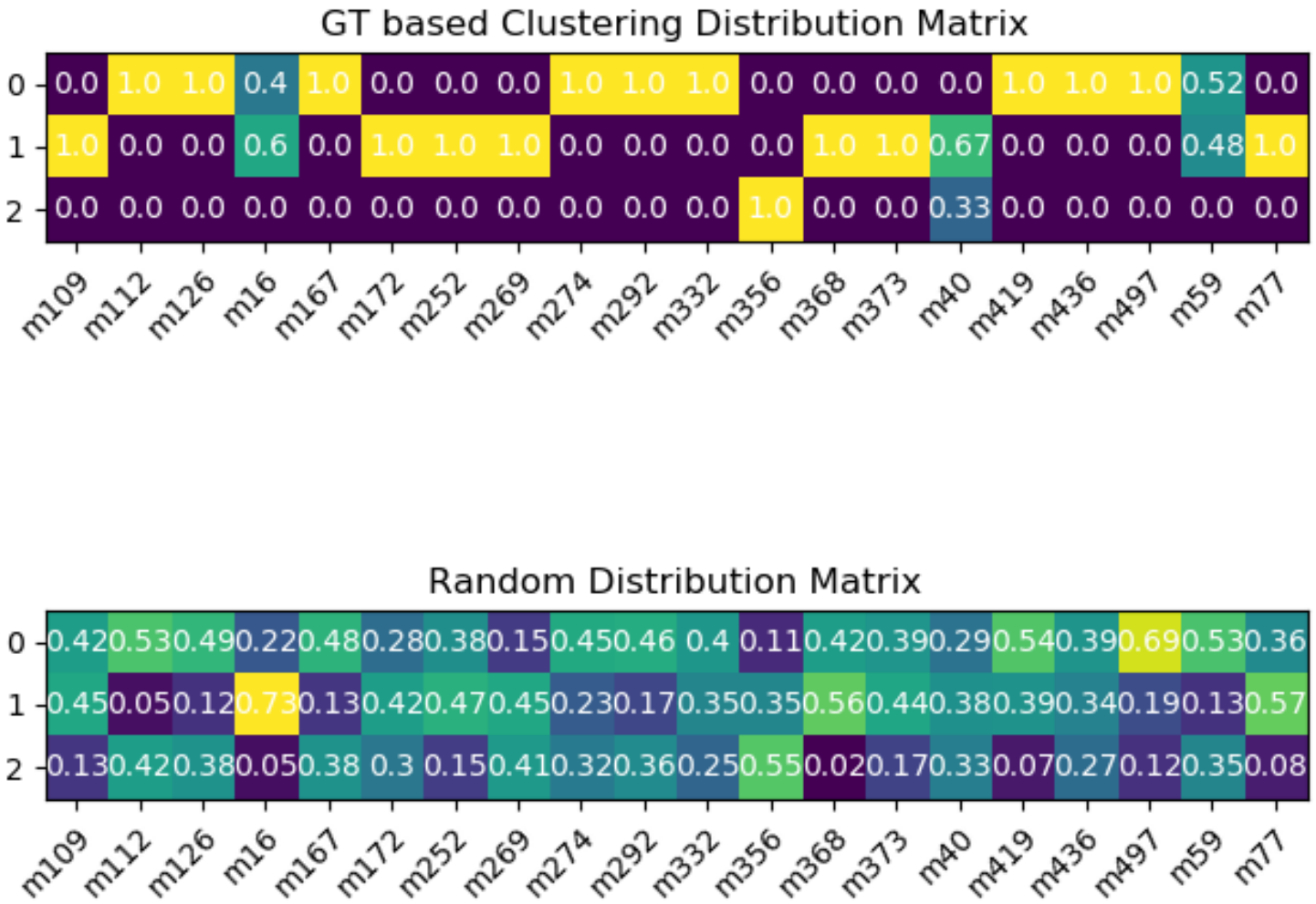

Figure 14: Ground Truth Distribution Matrix vs Random Distribution Matrix - only the correct matches are shown

7 Conclusions & Future Work

The results show that emotion-based information, at present, is not close to being able to represent the movie content by itself. Emotion-aware features cannot yet act as the sole descriptive or discriminative factor among movies. However, results strongly hint at the presence of a statistically significant correlation between emotion-aware features and high-level textual information attached to movies.

It goes without saying that this research area would greatly benefit from more robust emotion recognition classification models with better accuracy. Although the resources the company has provided the author are state-of-the-art in this regard, the accuracy that the speech emotion recognition models achieve, in general, is not great; this is due to the fact that the models used to extract these results where "off-the-shelf", for the experimentation to be conducted without any domain adaptation and the setup to be as generic and realistic as possible. However, Behavioral Signals is already working on more advanced models that involve deep learning methods which stand to achieve better overall performance in the near future; the incorporation of more and of better quality annotated datasets, which the company is in the process of obtaining, are sure to greatly contribute to this end. When the performance of emotion recognition classification models in the movies domain has reached a higher level, the company is planning on revisiting models such as the ones proposed in this work in order to review and re-evaluate their efficiency.

Another great challenge we had to face in this work is the lack of good enough ground truth information to evaluate our emotion-aware features against. Choosing metadata as our ground truth, although our only option in this case, sets a restrictive upper bound to our model's performance. Retrieval based on metadata is far from perfect and this is the reason why it would be valuable to see what other kinds of information can be extracted from emotion content which is not present in the metadata. Unfortunately, having to set up the model in this manner confines this knowledge. This area could greatly benefit from further research on fusion models that make use of both metadata and other textual information with emotion recognition features in order to assess potential improvement in content-based retrieval and recommendation technologies.

Lastly, a very valuable result that could have been produced if the utterance level audio-text alignment could be completed would be the assignment of emotion profiles to different actors and maybe documenting the progression and variation of emotion during the course of a movie. A larger movie dataset, where there would be multiple movies starring a unique actor or directed by a unique director, could also enable this documentation across multiple movies through the years and, furthermore, extract similar information about directors and associate emotion to directing styles. Since the utterance level text-audio alignment was not completed during this particular

capstone project, this could be the premise of a future project. For some thoughts on how we were to go about it, one can refer to Appendix B.

References

[1] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM, 2001.

[2] Francesco Ricci, Lior Rokach, and Bracha Shapira. Recommender systems: introduction and challenges. In *Recommender systems handbook*, pages 1–34. Springer, 2015.

[3] Shashidhar G Koolagudi and K Sreenivasa Rao. Emotion recognition from speech: a review. *International journal of speech technology*, 15(2):99–117, 2012.

[4] George Trigeorgis, Fabien Ringeval, Raymond Brueckner, Erik Marchi, Mihalis A Nicolaou, Björn Schuller, and Stefanos Zafeiriou. Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 5200–5204. IEEE, 2016.

[5] Marco Paleari and Benoit Huet. Toward emotion indexing of multimedia excerpts. In *Content-Based Multimedia Indexing, 2008. CBMI 2008. International Workshop on*, pages 425–432. IEEE, 2008.

[6] Christos-Nikolaos Anagnostopoulos, Theodoros Iliou, and Ioannis Giannoukos. Features and classifiers for emotion recognition from speech: a survey from 2000 to 2011. *Artificial Intelligence Review*, 43(2):155–177, 2015.

[7] Cristian Danescu-Niculescu-Mizil and Lillian Lee. Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011*, 2011.

[8] BS Player Subtitles. http://www.bsplayer-subtitles.com/. Accessed: 10/09/2018.

[9] opensubtitles. https://www.opensubtitles.org. Accessed: 10/09/2018.

[10] the Internet Movie Database. https://www.imdb.com/. Accessed: 25/10/2018.

[11] FFmpeg Team. FFmpeg. *URL http://FFmpeg. org*, 2013.

[12] Theodoros Giannakopoulos and Aggelos Pikrakis. *Introduction to audio analysis: a MATLAB® approach*. Academic Press, 2014.

[13] Hyoung-Gook Kim, Nicolas Moreau, and Thomas Sikora. *MPEG-7 audio and beyond: Audio content indexing and retrieval*. John Wiley & Sons, 2006.

[14] Lisa Feldman Barrett. Discrete emotions or dimensions? the role of valence focus and arousal focus. *Cognition & Emotion*, 12(4):579–599, 1998.

[15] Elizabeth A Kensinger. Remembering emotional experiences: The contribution of valence and arousal. *Reviews in the Neurosciences*, 15(4):241–252, 2004.

[16] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[17] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[18] Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen, and John T Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):5–53, 2004.

[19] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[20] Theodoros Giannakopoulos. pyaudioanalysis: An open-source python library for audio signal analysis. *PloS one*, 10(12):e0144610, 2015.

[21] Theodoros Giannakopoulos and Sergios Petridis. Fisher linear semi-discriminant analysis for speaker diarization. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(7):1913–1922, 2012.

A Movie titles list

Table 3: Movie titles for main analysis and their corresponding movie IDs.

Movie ID	Movie Title
m2	15 minutes
m7	a nightmare on elm street 4 the dream master
m14	alien nation
m15	aliens
m16	Amadeus
m20	American psycho
m22	Austin Powers - international man of mystery
m40	Braveheart
m42	Casablanca
m44	The cider house rules
m50	donnie darko
m59	Fast times at Ridgemont high
m65	From dusk till dawn
m72	Ghost world
m77	The graduate
m79	The grifters
m86	Hellboy
m91	Hope and glory
m98	Indiana Jones and the last crusade
m103	It happened one night
m104	JFK
m105	Jackie Brown
m109	Juno
m112	Knight moves
m126	Minority report
m147	The night of the hunter
m148	A nightmare on elm street
m151	No country for old men
m157	The patriot
m167	Rear window
m170	Reservoir dogs
m172	Scary movie 2
m221	Total recall

m243	Annie Hall
m249	As good as it gets
m252	A walk to remember
m260	Batman and Robin
m267	Being John Malkovich
m269	The big Lebowski
m274	Blood simple
m279	The bridges of Madison County
m288	Bull Durham
m289	Casino
m292	The crying game
m296	Chinatown
m301	A clockwork orange
m318	Dead poets society
m323	Die hard
m324	Dog day afternoon
m325	Domino
m332	L.A. confidential
m334	The English patient
m337	Star Wars - episode 5: The empire strikes back
m340	Excalibur
m347	Fargo
m356	The adventures of Ford Fairlane
m358	Frequency
m360	Jason lives - Friday the 13^{th} part VI
m362	Jason takes Manhattan - Friday the 13^{th} part VIII
m364	Gandhi
m368	Glengarry Glen Ross
m373	Good Will Hunting
m377	Hackers
m396	I walked with a zombie
m398	Insomnia
m400	I still know what you did last summer
m402	It's a wonderful life
m408	Jerry Maguire
m409	Jurassic Park III
m419	The silence of the lambs

m420	The last of the Mohicans
m422	Lock stock and two smoking barrels
m433	The Matrix
m436	Memento
m439	Midnight express
m441	Misery
m444	Moonstruck
m445	Monty Python and the holy grail
m448	Mulholland Dr.
m462	Notting Hill
m473	Planet of the apes
m474	Platoon
m478	Predator
m480	The princess bride
m489	Star Wars - episode 6: Return of the Jedi
m491	Rocky
m497	Rush hour 1
m508	Se7en
m511	Shakespeare in love
m514	The shining
m541	Superman iii
m542	Superman ii
m545	The sweet hereafter
m547	Terminator 2: Judgement day
m549	Terminator
m553	The man who wasn't there
m558	The third man
m576	Tombstone
m579	Toy story
m585	True lies
m591	Unforgiven
m592	The usual suspects
m594	Vertigo
m595	Very bad things

B Utterance level audio-text alignment

The analysis described in the main text was based on the assumption that a single movie dialogue can be described by a single emotion, which can be expressed by a pair of values in terms of valence and arousal. This is not a farfetched assumption to make, especially for movie dialogues, where each dialogue usually encloses a logical topic. However, an utterance level audio-text alignment would further enable us to dissect each dialogue audio into unique lines, or 'utterances' and assign a specific emotion to each of them.

The utterance level audio-text alignment was attempted but not completed during the course of this project. This is due to difficulties which generally have to do with the high dissimilarity between the scripted dialogues in the Cornell dataset and the movie subtitles which represent what the actual utterances during the movie were, as mentioned in Section 4 of the main text. One of the challenges was the fact that each line in a dialogue may correspond to multiple subtitles in the .srt file and punctuation in subtitles is rather rare, so a one-to-one matching or a matching based on sentence tokenization could not be performed. This is why during the dialogue level alignment the matching was performed using the entire dialogue to 'fish' subtitles out of the .srt file.

Another problem is the complete absence of some lines in the .srt and vice versa. A deciding factor in the success of the dialogue level alignment was the 'filling' of intermediate subtitles, which were not matched based on their common tokens with the dialogue, in areas where their neighboring subtitles were matched. This is unfortunately not an option in the utterance level alignment, due to absent lines and absent subtitles. Since the problem did not seem to be solvable with text manipulation, an idea was to combine Speaker Diarization along with Automatic Speech Recognition in order to dissect the dialogues extracted from the dialogue level alignment.

Speaker Diarization is the process of partitioning an input audio signal into homogeneous segments according to the speaker identity, while Automatic Speech Recognition (or ASR for short) is the conversion of spoken utterances into written text. The idea is that Speech Diarization could provide us with the knowledge of how many actual lines the dialogue is made up of, but since this would hold no information about speaker turn, ASR could recognize part, or all, of the utterances being spoken so that they could then be matched to some of the lines in the dialogue. Still, while ASR works very well at recognizing voice, especially in the English language, even when encountering funny accents, Diarization cannot yet boast great accuracy levels.

Despite that, Diarization was indeed applied to all extracted dialogue segments in order to assess its performance. The Diarization module used for this purpose is part of the open source Python Audio Analysis library (or pyAudioAnalysis) [20]. The Diarization implemented in this library is a variant of the method proposed in [21].

Three of the four steps from which the method is comprised are used here, namely automatic audio feature extraction, k-means clustering and smoothing. Fortunately, the number of speakers is known and fixed in our case, since the available dialogues always take place between two people, which greatly helps the performance of the k-means clustering.

The library also provides a built-in module for evaluating the results of the diarization, provided that there is a ground truth to compare them to. To this end, the author annotated 30 dialogue segments, noting the start and end time of each speaker in each one and labeling them. Since the library enables us to simultaneously run the diarization module and evaluate it, we first need to tune the model, which requires 3 parameters, two of which are independent, namely mid-term window size and short-term window size; the mid-term window step is set to always be half of the mid-term window size per the creator's advice. The function is tuned with short-term window size values from 0.02 to 0.2 with a step of 0.1 and a mid-term window size from 0.5 to 1.5 with a step of 0.1 seconds. Cluster Purity and Speaker Purity are documented for all parameter combinations. The way these two metrics are defined in this particular model, they closely resemble standard precision & recall metrics, respectively, from which one can calculate the f1 score and produce a PRF curve. F1 score is defined here as follows:

$$f1 = \frac{2 * (cl\_purity * sp\_purity)}{(cl\_purity + sp\_purity)}$$

Results are shown in Figure 15. The Cluster Purity – Speaker Purity line is marked with blue, while the f1 score – Speaker Purity one is marked with orange. The values represent percentages.

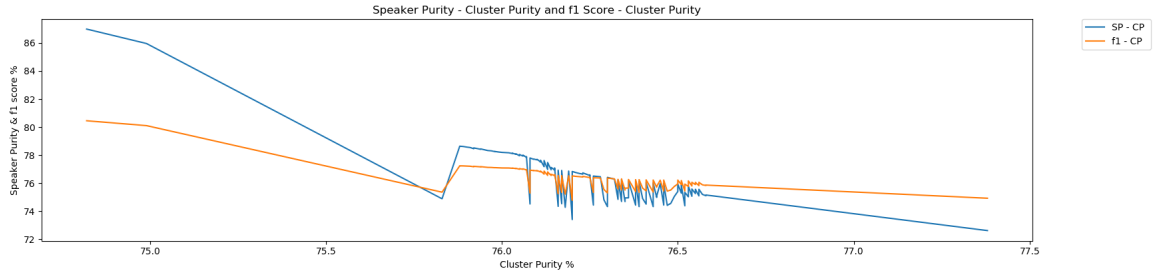

Figure 15: Speaker Purity & f1 score versus Cluster Purity for different values of mt_size, st_size

Based on these results, we choose the pair of mid-term window and short-term window that results in the best f1 score. In our case the best pair was a mid-term

window size of 1.5sec and a short-term window size of 0.020, which achieved a Mean Cluster Purity of 74.99% and a Mean Speaker Purity of 85.96%.

These results were deemed borderline acceptable for the purpose the diarization was needed. However, since this branch of the analysis did not continue any further, it is not known how it would perform on the utterance level audio-text alignment task. This could be an item for future research.