Video Clip Retrieval

Authors: Boura Tatiana, Sideras Andreas
Instructor: Giannakopoulos Theodoros

MSc in Artificial Intelligence, NCSR Demokritos & University of Piraeus
Multimodal Machine Learning

---

1 Introduction

In this assignment, we are presented with the following task: given a video clip, we must retrieve the most similar video clips to it from a data collection at our disposal. There are many ways to approach this task, such as using standard recommender system techniques, treating it as a classification problem and employing neural networks to solve it, etc. Our approach is based on multimodal machine learning techniques that take into consideration the visual, audio, and lyrical aspects of a video clip. Specifically, in Section 2 we extract features from the aforementioned three modalities of the video clips available to us. Then, in Section 3, we apply representation learning to these extracted features to obtain their embeddings. To retrieve the most similar video clips to a queried one, we find its closest neighbors using a similarity metric, in Section 4. We then evaluate this method, in Section 5 and present a visual representation of the embeddings to confirm the correctness of our approach.

2 Feature Extraction

To create our database, we utilized the YouTube API to download playlists from YouTube. In total, we obtained 1363 unique video clips, which we subsequently categorized into the following 9 classes: *blues-jazz, country, hip-hop, indie-folk, metal, pop-rock, punk, soul, and UK drill.* In this section, we will outline the process by which we extracted features from these video clips using all three modalities.

2.1 Audio Features

For each video in our collection, we computed 257 mid-term audio features using Librosa[5]. The feature vector for each song included the beats per minute (bpm) feature, as well as the statistics such as mean, median, standard deviation and 75th percentile of the following short-term sequences:

- Zero crossing rate
- Mel spectogram and dB-scaled spectrogram
- Harmonics
- Perceptual shock wave
- Spectral centroids and their 1^{st} and 2^{nd} derivatives
- Spectral flux and Spectral rolloff
- Spectral bandwidth 2,3 and 4
- 12 Chroma features
- 13 MFCCs and their 1^{st} and 2^{nd} derivatives

2.2 Video Features

To obtain the features for the video portion of the video clip, we leverage the capabilities of the pre-trained model called 3D ResNet from the PyTorch Video library[2]. This model is specifically designed for video recognition tasks and is based on a 3D convolutional neural network as described in[3]. It was pre-trained on the Kinetics dataset[4], which con-

tains a large collection of video clips spanning various action classes.

The 3D ResNet architecture is an extension of the original ResNet model, initially proposed for image classification, to handle spatio-temporal data in videos. While traditional 2D convolutions are suitable for capturing spatial features in images, the 3D ResNet incorporates 3D convolutions, which take into account both spatial and temporal dimensions. These 3D convolutional layers are responsible for learning hierarchical representations of the video frames i.e. capturing motion and temporal dependencies present in the video sequence. They are interconnected through residual connections, enabling efficient gradient flow and facilitating the training of deep networks.

After forwarding our video clips through the aforementioned model, we obtain a video feature vector of dimension 400.

2.3 Text Features

To utilize this modality, we derived the lyrics of the videos in our collection through YouTube's API. Specifically, we accomplished this by searching for the title of each video clip on YouTube and retrieving all available versions until we found one that contained the lyrics. Following this procedure, we successfully obtained lyrics for a total of 821 songs.

After obtaining the lyrics of the songs, for each video clip, we obtained a single vector representation of all its lyrics, consisting of 768 dimensions, using the BERT model[1]. Bidirectional Encoder Representations from Transformers (BERT), is a pre-trained language model that utilizes a Transformer architecture. It is worth noting that BERT outputs a vector representation for each token in the input text, including a special token called [CLS]. The [CLS] token serves as an additional representation that captures the overall information and contextual understanding of the entire input sequence, making it suitable as a summary representation for the sentence or document. We utilize this token as the embedding representation for the text features.

2.4 Feature Vector

The final feature vector is obtained by concatenating and then standardising the representations of the three modalities. Recall that lyrics were available for only 821 songs, resulting in a data collection comprising 821 video clips, each represented by a feature vector of 1425 dimensions.

3 Representation Learning

To uncover the intrinsic dependencies between the 3 modalities, we use an AutoEncoder architecture, which learns a latent space representation that captures the underlying structure of the features. We train the AutoEncoder to map the 1425 features for each video clip to a lower-dimensional latent space of 256 features. It's architecture is presented in 1.

```
Autoencoder(
  (encoder0): Linear(in_features=1425,
      out_features=1024, bias=True)
  (activation): Tanh()
  (encoder1): Linear(in_features=1024,
      out_features=1024, bias=True)
  (activation): Tanh()
  (encoder2): Linear(in_features=1024,
      out_features=768, bias=True)
  (activation): Tanh()
  (encoder22): Linear(in_features=768,
      out_features=512, bias=True)
  (activation): Tanh()
  (encoder3): Linear(in_features=512,
      out_features=256, bias=True)
  (activation): Tanh()
  (decoder0): Linear(in_features=256,
      out_features=512, bias=True)
  (activation): Tanh()
  (decoder11): Linear(in_features=512,
      out_features=768, bias=True)
  (activation): Tanh()
  (decoder1): Linear(in_features=768,
      out_features=1024, bias=True)
  (activation): Tanh()
  (decoder2): Linear(in_features=1024,
      out_features=1024, bias=True)
  (activation): Tanh()
  (decoder3): Linear(in_features=1024,
      out_features=1425, bias=True)
)
```

Listing 1: AutoEncoder'a arcitecture

The training configuration of the AutoEncoder is presented in *Table* 1.

Training Parameters	Value
Batch Size	32
Epochs	600
Learning Rate	0.0001
Loss	MSE
Optimizer	Adam

Table 1: Training configuration of AutoEncoder

4 Video clip retrieval

After training the AutoEncoder, we compute the embedding for each video clip. This embedding is the 256 dimensional output of the encoder.

When a query/video clip is received, we obtain its latent representation and map it to the latent space along with the video clips in our collection. Subsequently, we calculate the distance between the query and the other video clips using the cosine similarity distance metric. The cosine similarity measures the similarity between two vectors by computing the cosine of the angle between them. It takes into account the direction and magnitude of the vectors, rather than just the Euclidean distance. A cosine similarity value of 1 indicates that the vectors are identical, while a value of 0 indicates no similarity. In our case, a higher cosine similarity score implies a higher degree of similarity between the query and the video clips, enabling us to retrieve the k-most similar video clips.

5 Experiments

We conducted a comprehensive comparison between the AutoEncoder method, the cosine similarity over the initial feature vectors, and the cosine similarity after applying PCA. Additionally, we employed clustering techniques to analyze the representations of the latent space. Evaluating the performance of our model posed a challenge, as it is difficult to provide a quantitative metric for this task. To gain deeper insights, we manually assessed selected video clips, focusing on evaluating the similarity of music genres and the overall resemblance of the videos. This assessment provided valuable qualitative feedback on the model's performance and its ability to capture genre similarity and visual similarities among video clips.

Let us now present two query examples and their evaluation.

5.1 Query example 1

The initial video clip we evaluated in our implementation was Falling In Reverse - "ZOMBIFIED", which belongs to the post-hardcore/metalcore genre. The song and the corresponding video clip revolve around a zombie apocalypse theme, characterized by dark colors and lyrics that align with the overall atmosphere.

Our implementation suggested the following 5 video clips as the most similar:

1. Avenged Sevenfold - Shepherd Of Fire
2. Nim Vind - "Where I'm From"
3. WATERPARKS - RITUAL
4. K.Flay - Raw Raw
5. Disturbed - Another Way To Die

All of the suggested video clips are categorized within various sub-genres of metal, ensuring a similar musical style to the query. Additionally, some of these video clips exhibit similar characteristics to the query, with corresponding video content that aligns with the theme and aesthetic elements.

If we do not compute the corresponding embedding for each video clip and instead use the initial features to calculate the similarity, the suggested video clips are as follows:

1. Blur - Parklife
2. Sonic Martini Music - User User User

3. The Calling - Wherever You Will Go
4. Sum 41 - In Too Deep
5. Green Day - Hitchin' A Ride

The suggested videos above primarily fall into the genres of rock, pop rock, punk rock/pop, which are not too far acoustically, yet clearly distinguishable from the query's genre. Interestingly, in three out of the five suggestions, the bands are displayed performing in a concert-like manner, which also happens in the queried video clip.

If we apply dimensionality reduction using PCA instead of computing the embeddings, the implementation suggests the following video clips:

1. Yellowcard - Southern Air
2. Sonic Martini Music - User User User
3. Matchbox Twenty - Don't Get Me Wrong
4. Destiny's Child - Soldier
5. Ice Cube, Mack 10, Ms. Toi - You Can Do It

These suggestions were the least relevant. The genres of the first three songs are pop-rock, while the remaining songs are hip-hop. Additionally, two of the suggested videos are lyric videos, and three of them feature the bands or individuals performing.

5.2 Query example 2

The second clip we evaluated was STORMZY - LONGEVITY FLOW, which is classified as a grime/hip-hop genre. In the video clip, the artist performs in a setting that resembles a party. It, also, a lot of people revolving around and interacting with the artist.

Using the embeddings to calculate the distance, the suggested video clips were:

1. Central Cee - Retail Therapy
2. Mike Jones - Back Then Part 2
3. Female Special - Plugged In w/ Fumez The Engineer
4. Cristale x Teezandos - Plugged In w/ Fumez The Engineer
5. Ying Yang Twins - Salt Shaker

The suggested songs are classified as either hip-hop or drill, both of which are genres very similar to the one in the query. In all of the videos, the artist takes center stage, which aligns with the nature of the queried video.

Using the raw features or applying dimensionality reductions in this example results in less accurate suggestions, as many of them belong to irrelevant genres such as country or pop.

5.3 Clustering of Latent Space

After training the AutoEncoder, we have computed as mentioned earlier an encoding vector for each video clip in our database. the clustering of these representations could provide several benefits and insights. We ran Kmeans with Euclidean distance as metric, and after applying the elbow method (*Figure* 1) we decided the final number of clusters, which is equal to 9.

Clustering the vector representations in the latent space obtained from the AutoEncoder can help verify if the latent space effectively groups together similar video clips. If the AutoEncoder is successful in capturing the underlying structure and patterns in the video clips, you would expect similar videos to have encoded representations that are close to each other in the latent space. Clustering algorithms can then help identify these groups or clusters of similar videos. We then sampled some video clips from each cluster and made some assumptions about the performance of our model. (*Figure* 2)

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

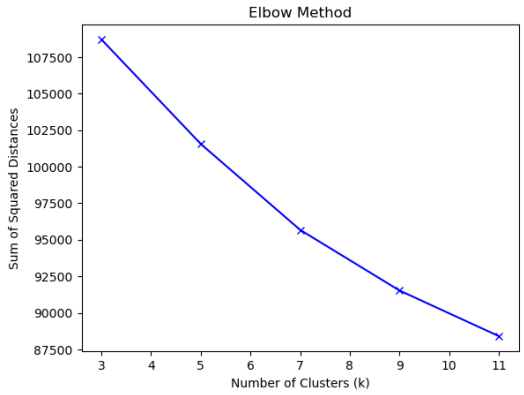

Figure 1: Finding best K value

cluster		
1	695	Luke Bryan - Roller Coaster (Official Music Vi...
	57	Choking Victim - War Story
	111	Craig Cardiff - Emm & May (Official Video)
	393	Coldplay - Adventure Of A Lifetime (Official V...
	626	Murray McLauchlan _ Down by the Henry Moore
2	667	Billy Ray Cyrus - Achy Breaky Heart (Official ...
	436	One Direction - You & I
	253	Chris Young - The Man I Want To Be (Official V...
	820	Sam Smith - Stay With Me ft. Mary J. Blige (Live)
	402	Maddie & Tae - Die From A Broken Heart (Offici...
3	351	(OOAK) Russ Millions x Buni x YV - Reggae & Ca...
	132	Florence + The Machine - Dog Days Are Over (20...
	407	Pennywise - 'Same Old Story'
	541	Coldplay - Hymn For The Weekend (Official Video)
	760	Sly Slick and Wicked - Somebody Please
4	440	Hate It Too - 'Twelve's the New Eight' Officia...
	794	Counting Crows - Mr. Jones (Official Music Video)
	23	DragonForce - Three Hammers (Live)
	379	OutKast -- Hey Ya lyrics
	121	Chris Lane - I Don't Know About You (Official ...
5	750	Grey Kingdom 'Sun Like Moon Light' - www.strea...
	590	Madison Violet - Small Of My Heart
	67	Johnny Flynn - Raising the Dead
	502	Mystery Train _ Playing For Change _ Live Outside
	237	Bukka White - Miss Leola
6	20	Huey - Pop, Lock & Drop It (Video Edit)
	41	50 Cent - Outta Control ft. Mobb Deep
	722	Lil Wayne - Fireman (Official Music Video)
	360	Eminem - Stan (Long Version) ft. Dido
	391	Ludacris - Rollout (My Business) (Official Mus...
7	19	Blur - Girls And Boys (Official Music Video)
	250	Central Cee - Retail Therapy [Music Video]
	725	The Struts - Kiss This (2014)
	362	Matchbox Twenty - Wild Dogs (Running in a Slow...
	129	Steel Panther - Party Like Tomorrow Is The End...
8	758	Morgan Evans - Over For You (Official Music Vi...
	526	Kelsie Kimberlin - Cosmopolitan Girl _ #stand...
	245	Queens Of The Stone Age - No One Knows (Offici...
	373	You Are The First, My Last, My Everything (Bar...
	336	Maroon 5 - Won't Go Home Without You (Official...
9	329	Motörhead – God Save The Queen (Official Video)
	18	Foo Fighters - Best Of You (Official Music Video)
	423	Jet - Are You Gonna Be My Girl
	170	Sex Pistols - God Save The Queen
	782	Social Distortion - Ball and Chain

Figure 2: Five samples from each cluster

[2] C. Feichtenhofer, H. Fan, J. Malik, and K. He. Slowfast networks for video recognition. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6201–6210, Los Alamitos, CA, USA, nov 2019. IEEE Computer Society.

[3] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition, 2019.

[4] Will Kay and et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[5] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8, 2015.