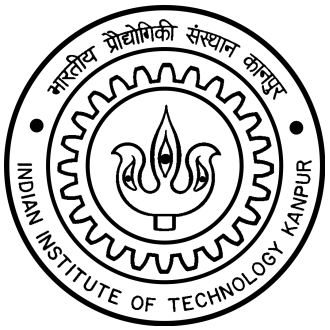

UGP - Report

Topic - Cross Modal Media Retrieval

Advisor:

Prof. Medha Atre

Ankit Bhardwaj - 150101

Amrit Singhal - 150092

Contents

Abstract

This project aims to utilize the the emotional information present in any media to perform cross-modal media retrieval efficiently. We present an implementation to extract this emotion from images as 2-dimensional vectors, and propose two methods to bring this emotion vectors from different medias in the same space, one being a statistical approach, and the other being a learning based approach. We also present a hypothesis, which allows us to establish a ground truth for the cross modal mapping. We perform extensive analysis of one of our proposals, and report the results obtained. We have also proposed and implemented a heuristic to retrieve cross-modal results efficiently.

1 Introduction

With the increasing amount of data being available in all formats, namely audios, images, videos and text, there is definitely a need to be able to process the similarity of data across formats, and perform efficient cross-modal retrieval. Traditional features such as pixel information for images or node information for audios, clearly cannot be used for this task. We need parameters that can be assessed across mediums. Our project deals with one specific medium: Emotion.

The emotional factor is a also important one from the aspect of retrieval. For example, if a user is searching for a sad song, then it is highly likely that he is feeling sad himself, and will be searching for sad videos or images as well. Clearly, emotion cannot be the sole factor for determining the cross-modal retrieval, but it surely is a important one.

2 Problem Statement

The projects aims to perform cross-modal media retrieval using emotional information from the data points in all modalities. The project dealt with extracting the emotion from the data, and bringing the emotion information extracted from different medias in different spaces, together into the same space, to allow cross-modal comparisons.

It is important to remember that we need to extract the emotion that the user feels on being presented with the data, and not necessarily the emotion present in the image, though in most cases, these two will coincide. For example, in the cases of images, we do not want to extract the emotion the person in the image is feeling, but the emotion that the user will feel on seeing that image. To that end, it is not even necessary that there be a person in the image. Even the images of animals or inanimate objects can arise emotions in the user. A similar logic holds for audios as well.

Also, the project involves not only extracting this information, and using it for retrieval, but it also includes handling this data efficiently, using appropriate data structures, to give search results at the minimum computational expense and time.

3 Representing Emotions

It is easy to say that we need to extract emotions from the data, but modelling this abstract entity of *emotion* in the mathematically appropriate fashion is a challenging task in itself.

1. **Discrete Models**
   We can either use a Bagging model, where the data is classified to belong to one of several emotion classes. While this model is good for image classification tasks, learning such a model would mean loosing in the proximity information of the data points. Any two data points will either belong to the same class, or not.

2. **Dimensional Models**
   These type of emotion models define emotions in one or more dimensions. Since each data point gets a value in a vector space, this type of emotion extraction allows the comparative approach that we need. There are numerous available dimensional models for emotions, but we use the **Circumplex Model** proposed by James Russell[2].

3.1 The Circumplex Model for Emotion

The model distributes the emotions in a two-dimensional vector space. The vertical axis models the arousal, and the horizontal axis models the valence.Fig. 1 shows how the emotions get distributed in this space.

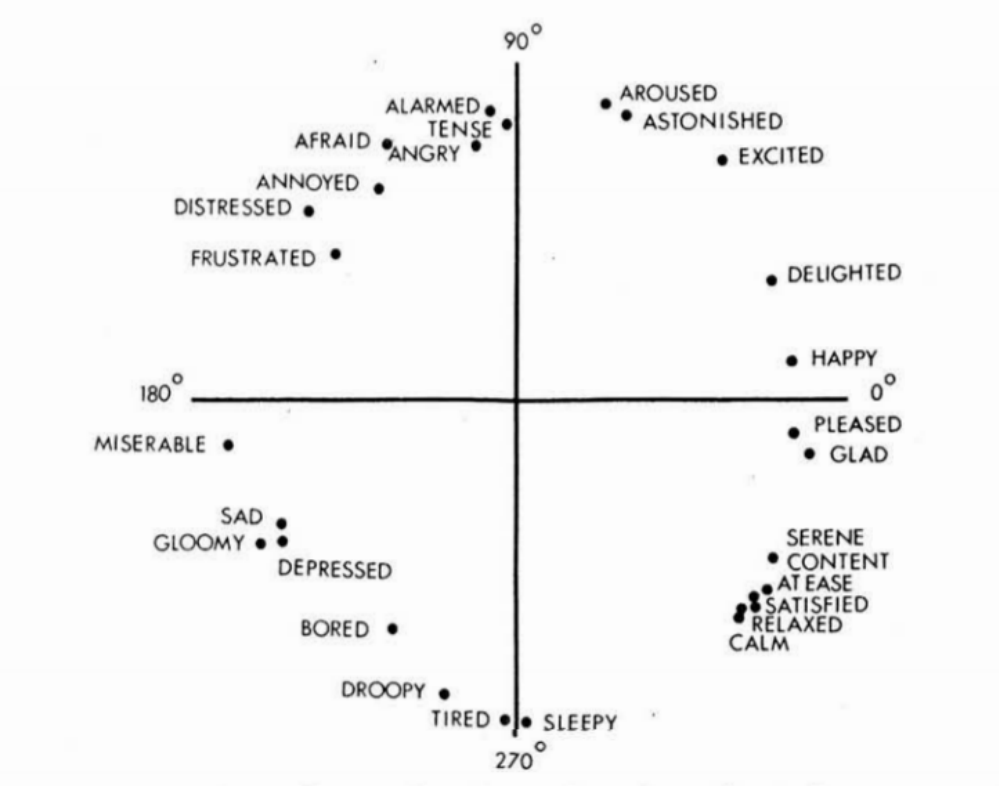

Figure 1: The Circumplex Model for Emotion Representation[2][1]

4 Existing Work on the Project

- This representation had already been chosen for the emotions, although any other has ot yet been tried yet. This is the first representation chosen, and other representations can be tried out if this fails.
- Also,there was a working implementation for the model to extract arousal-valence(A-V) values from audios, from the previous members of the project. It gives A-V values for each file at every 500ms time window, as well as A-V value for the complete audio file.

[1]James A. Russell. A circumplex model of affect.Journal of Personality and Social Psychology, 39:1161,1980

5 Our Contributions

- Literature Review to decide a model to extract A-V values from images, and implemented it.
- Proposing a heuristic for efficient search-retrieval algorithm for the cross modal data, and implemented it.
- Proposed a hypothesis for establishing ground truth mapping across different mediums.
- Proposed two different models to bring the different modalities into the same space using the proposed hypothesis.
- Implementing the proposed models. These can prove/disprove the hypothesis as well.

In the following sections, we will be explaining each of our contributions in greater detail.

6 The Image A-V Model

6.1 Architecture

This model was proposed by Kim et al.[1]. It is a emotion based feed forward deep neural network which produces the emotion values of a given image, in the form of continuous values in the 2-dimensional space (Valence and Arousal). The model takes as input a **1721**-dimensional feature vector for every image I, constructed as follows:

- Local features: We extract a 512-dimensional GIST vector, and a 59-dimensional local binary pattern(LBP)descriptor.
- Object level features: We extract a 1000 dimensional VGG feature vector, that represents the probabilities of different object categories.
- Semantic features: Even the background information in important in extracting the emotional information from the image. So, we also take a 150-dimensional feature vector that encapsulates this semantic information.
- All these separate features are simply concatenated together to obtain a feature vector for the image model.

The network consists of a input layer, three dense hidden layers, and a output layer.

	Input	H1	H2	H3	Output
Number of neurons in the layer	1721	3000	1000	1000	1

6.2 Implementation

The model was implemented in Python using the **Tensorflow** backend. The squared error between the output value and the ground truth was used as the loss function L. The model was trained with a batch size of 500, for 70000 iterations, where its loss function L decreased no more. Training was done using the Stochastic Gradient Descent Optimizer, with a constant learning rate of 0.0001 and momentum of 0.9.

The full Image_AV_model implementation can be found at https://github.com/amr4i/EmotionDetection.

7 The Search Retrieval System

For the purpose of achieving a mixed result on the search query, we have implemented an intelligent heuristic to give us cross modal results for given search queries.

7.1 Library used

Annoy - More information can be found on the given link:
https://github.com/spotify/annoy

7.2 Heuristic used

Theoretically annoy approximates the nearest neighbours through various optimization and methods which are not of serious interest. Further information can be found at the above link. Using the library, we get k nearest neighbours from image and music files separately, where k is the number of results we have to display in the search. For this purpose we build the ***annoy models*** for both image and audio files separately. We also receive the distances of these nearest neighbours as an array. Then, we travel over these arrays to select the k entries to be displayed. Also, once an entry is selected from the two arrays, we multiply the distances of the remaining items of the corresponding array by a constant factor. Then we repeat the process of finding the least distance neighbour (after progressing the pointer variable for the array from which the entry was selected). This pushes the points of the same class away so that the results can be made more balanced. This method makes sure that the entries among images or audios separately do not come out in a wrong order. The order between image and audio files will depend on the constant factor which also decide the quality of result. Unfortunately, this parameter is a hyper-parameter and can only be attuned once the dataset is finalized.

7.3 Implementation

https://github.com/ankitbha/NEMO

7.4 Time Experiment Results

Putting aside the pre-processing time which needs to be invested in a lump-sum, the method works extremely fast in practice. The results for around million entries can be obtained in 0.01 seconds. Thus the method is feasible for datasets with even billion entries.

8 Proposed Hypothesis for Ground truth establishment

8.1 Statement

We propose that *for any video segment, a user will feel the same emotion on being presented only with the images from that segment, or only with the audios from that segment separately.*, i.e. the audio in any video has the same average emotion in any time window, as the average emotion of the images that occur on the screen during that time.

8.2 Creating a Dataset

Proceeding with this assumption, we have manually created a small dataset of some song videos. We have made sure to keep our dataset diverse, by including different multiple emotion conveying videos, and different languages. We have then split these videos into video segments of length **15 seconds** each. The segments are now treated as independent data points. This gives a larger, and diverse dataset. Now, audio segments are extracted from each video, and that forms our audio dataset. Running this through the audio A-V

extraction model, we will get the audio A-V values for that segment. For the image counterpart, we do not take one image per segment, but instead we take 1 image per second of the video segment, and the average AV score over all images from a segment will form the image AV values for that segment.

8.3 Implementation Details

- Tool use for splitting: ffmpeg
- Tool used for extracting images: ffmpeg
- Tool used for extracting audios: avconv

Now, these audio A-V values and image A-V values are not in the same space. We now propose two methods for bringing them into the same space, to allow comparisons between them.

9 Transferring AV values from different modalities into the same space

9.1 Statistics based proposal

Using the dataset created in Section 8.2, we obtain some data points that can be related to each other in the image and audio datasets (say landmark points or anchor points). We can use these anchor points to perform some statistical manipulations to get a transforrmation of both data point sets to a same space. We have used these anchor points to perform a Procrustes analysis, and tried to fit them to the same space. Detailed analysis of this method is given in Section 10.

9.2 Learning based proposal

We already have separate models trained to predict the Arousal-Valence values for audios and images, in different modalities. Let call these models A_1 and I_1 respectively. Now, our aim is to get values that are in the same space.

For this, we plan to again learn these two models jointly, using videos as our training data.

We will separate the videos in audios, and image sequences, just like before, and further segment them into time frames of length l. We plan on tuning this value of l, but it will typically be around 10-15 secs. The length should be enough to be able to make sense of some emotion from it, but not too long to have a huge variation in the emotion being conveyed.

Now, we will run these audio segments through model A_1 to get the emotion value a_1, and we run some well-defined k images, based on the time frame, from the corresponding image sequence, and run them through model I_1 to obtain emotion values $i_{t1}, i_{t2}, ..., i_{t3}$ and take their (weighted) average to obtain the emotion value i_1 for the image sequence.

These video clip segments along with the a_1 and i_1 values for each segment, form the training set for our new model.

Now, we train the models again, to generate values in the same domain. Suppose the model predicts value a_2 for audio, and value i_2 for image, then we aim to minimize the loss function of the form:

$$L(a_2, i_2) = \Sigma_{trainData} \left(|a_2 - i_2|^2 + |f_1(a_1) - a_2|^2 + |f_2(i_1) - i_2|^2 \right)$$

The first term in the loss function is to get the corresponding image and audio values from the same segment as close as possible. Without any other restriction, this would lead to all values being converged

to one same point, loosing all essential information about the emotion. Now, we do need to maintain the emotion, so the second and the third terms are added to keep the new scores close to some representation of the original scores.

Here, non-linear f_js are needed for allowing non-linearity in the mapping. If we choose a linear function, then only a linear shift can be learned, which is equivalent to out first model. If we omit f_i altogether, then will try to get as close to the mean value between a_j and i_j, which is not what we want.

This should be able to accomplish our aim to get values in the same domain for images and audios, by enforcing that the values for images and audios with same emotion are close by, and maintaining the relative separation between audios and images among themselves. Rather than transforming one space to another, this aims to bring both the representations in a common space.

10 Procrustes Analysis and its Results

10.1 Idea

We have already told earlier in Section 9.1 that we can try to come up with a transformation that converts the A-V values from image space and audio space to a same space, or establishes the truth of our ground truth hypothesis. To make inquiries into this part we used the primary method of comparing two curves in statistics. (Since, we have no way of directly comparing two spaces, comparing the curves in two spaces should give some idea on this.)

So, we used the anchor points from the dataset from Section 8.2, that we obtained by splitting a video into its audio and image and running them through the different models that have been trained separately. It is important to note that this method is just an inquiry into the structure of the output spaces of these models and does not justify or falsify the earlier assumption. Thus, taking these points as anchor, we ran a Procrustes analysis to find the similarity between the output vectors in the two spaces from the different models.

Procrustes analysis is actually a statistical shape analysis tool, that first performs superposition optimally translating, rotating and uniformly scaling the objects, and then calculating the disparity between the resulting shapes. In our case, if the two outputs from the image AV model and audio AV model differ only linearly, then the Procrustes analysis will be able to align them and get a low disparity. Thus, this linear transformation can then be used to get them in the same space.

10.2 Results

We tested this method on 7 videos with enough variety (videos containing sad images with dull sounds, happy images with cheerful sounds, videos in different languages, etc.). We split up the videos into video segments of length 15 seconds each, obtaining 195 independent test points that we used as anchor points. Assuming that the curve formed by these 195 points represents the space of output for both the models, we performed Procrustes analysis on these points. **The disparity score that we obtained was 0.96.**

10.3 Implementation

The Python Library `scipy` was used to perform the Procrustes method. The method was given as input the two 195×2 shaped matrices, and gets as output two same shaped matrices A and I as before, and a disparity score. The output matrices are actually transformed versions of the original matrices, aligned together to the greatest degree possible using only linear transformations.

$$I \leftarrow \text{Standard Image matrix}$$

$$A \leftarrow \text{Standard Audio matrix}$$

Standard Matrix: A matrix M such that it is centered around origin and $Tr(MM^T) = 1$

10.4 Similarity Calculation

Let a_i and v_i be the transformed arousal and valence values of the i^{th} data point. Then, for both audio and image points,

$$\sum_{i=1}^{195}(a_i^2 + v_i^2) = 1$$

Let P_{im}^i and P_{au}^i correspond to the i^{th} point in the transformed image and audio space respectively. The disparity score is defined to be

$$Disp(I, A) = \sum_{i=1}^{195} dist^2(P_{im}^i, P_{au}^i) = 0.96$$

Now, for the average case, we can assume that all vectors have the same length, then

$$\forall i, len(a_i, v_i) = \frac{1}{\sqrt{195}}$$

Let α be the angle between the corresponding point vectors from the audio and image spaces. Then,

$$\forall i, dist(P_{im}^i, P_{au}^i) = 2\dot{l}en(a_i, v_i) * sin\left(\frac{\alpha}{2}\right) = \frac{2}{\sqrt{195}} sin\left(\frac{\alpha}{2}\right)$$

Since,

$$\sum_{i=1}^{195} dist^2(P_{im}, P_{au}) = 195 * \frac{4}{195} * sin^2\left(\frac{\alpha}{2}\right) = 0.96$$

$$sin(\frac{\alpha}{2}) = \sqrt{0.24}$$

$$cos(\alpha) = 1 - 2sin^2\left(\frac{\alpha}{2}\right) = 0.52$$

Thus the similarity between outputs of two models on average is not very good.

10.5 Inferences

The result obtained clearly shows that the output spaces of both models are fairly different and can not be transformed into one another with any *linear* transformations like scaling, shifting or rotation. We infer three possibilities from this result:

- The window size taken for the videos (10-15 seconds) is too large and causes significant deviation in experienced/measured emotions. Then, on taking the average over this diverse emotion values, the meaning is lost, and this leads to poor results.
- The transformation between the spaces is non-linear. In this case, our proposed learning based approach should work well.
- The actual ground truth values(which are not known to us) are incompatible with the hypothesis. It is basically equivalent to saying that hypothesis is wrong.

11 Future Work

The possibilities of future work are as follows:

- Checking the standard deviation on the emotion readings of both models to verify the first possibility mentioned in the inference section.
- Implement the proposed learning model, and run it to verify the second possibility from the inference section.
- If the ground truth hypothesis turns out to be false and we can create our own dataset suitable for the problem. We will need some ground truth definitely, because no method can bring the two output spaces together without some ground truth values.

References

[1] Hye-Rin Kim, Seon Joo Kim, and In-Kwon Lee. Building emotional machines: Recognizing image emotions through deep neural networks. *CoRR*, abs/1705.07543, 2017.

[2] James A. Russell. A circumplex model of affect. *Journal of Personality and Social Psychology*, 39:1161, 1980.