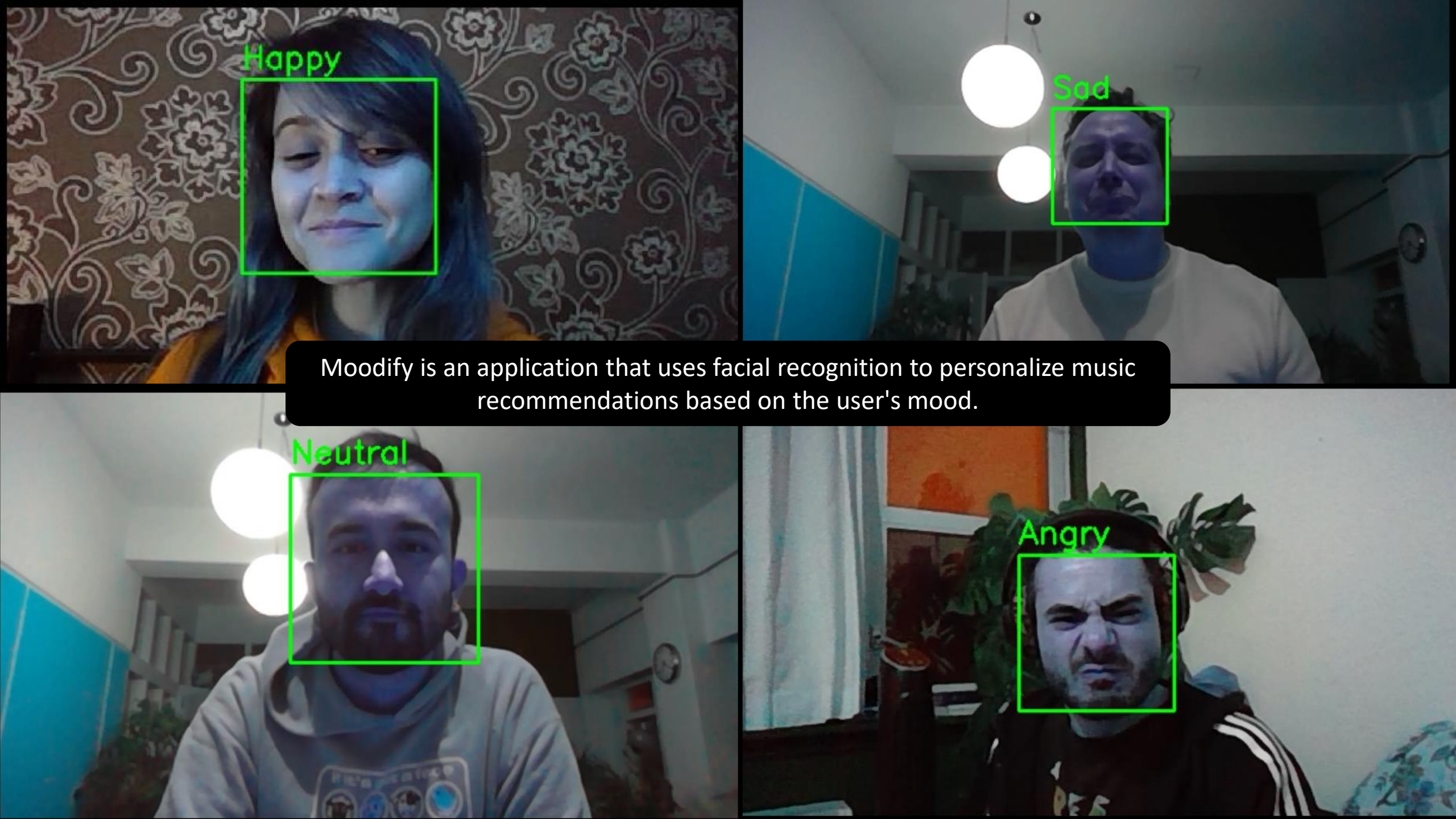
Moodify



Moodify is an application that uses facial recognition to personalize music recommendations based on the user's mood.

Building the app

1. Mood detection using computer vision and deep learning model

2. Recommendation system inside the Spotify app

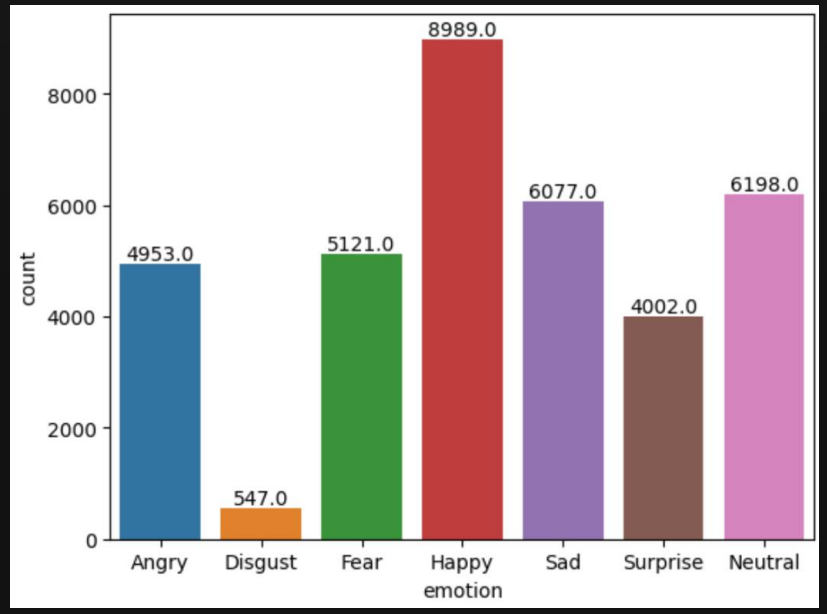
3. Improving songs labelling with unsupervised machine learning model

Mood detection system

The fer2013 dataset

	emotion	pixels	Usage
28188	6	232 232 229 216 194 169 184 205 210 197 167 13...	Training
7513	2	68 57 34 42 50 51 51 45 49 42 58 135 175 183 1...	Training
23435	3	85 71 64 65 63 60 58 63 59 59 63 65 62 52 40 4...	Training
13564	3	19 21 30 39 48 60 69 78 82 90 98 109 118 122 1...	Training
10214	0	255 253 255 192 96 94 90 85 94 118 127 132 154...	Training

Sample of Fer2013 image dataset



Distribution of images per label / ‘emotion’ in the fer2013 dataset

The CNN model

collection-training.ipynb M ● exploration.ipynb

Model > collection-training.ipynb > Create Img Recognition Model > #Callbacks: special functions that are called during the training at varios points.

+ Code + Markdown | Run All Clear All Outputs Restart | Variables Outline ··· recsys (Python 3.10.9)

```
 dropout_4 (Dropout)         (None, 128)               0

 out_layer (Dense)           (None, 4)                 516

=================================================================
Total params: 2,395,204
Trainable params: 2,393,156
Non-trainable params: 2,048
_________________________________________________________________

C:\Users\danid\AppData\Local\Temp\ipykernel_14200\1913433744.py:3: UserWarning: `Model.fit_generator` is deprecated and
  history = model.fit_generator(

Epoch 1/16
737/737 [==============================] - 777s 1s/step - loss: 1.5618 - accuracy: 0.3182 - val_loss: 1.2712 - val_accur
Epoch 2/16
737/737 [==============================] - 875s 1s/step - loss: 1.2779 - accuracy: 0.4111 - val_loss: 1.1011 - val_accur
Epoch 3/16
221/737 [=======>......................] - ETA: 10:19 - loss: 1.2032 - accuracy: 0.4535
```

CNN model being trained with fer2013 image dataset

Performance metrics

```
total wrong validation predictions: 636


              precision    recall  f1-score   support

           0       0.72      0.67      0.70       495
           1       0.92      0.90      0.91       899
           2       0.71      0.62      0.66       608
           3       0.63      0.76      0.69       620

    accuracy                           0.76      2622
   macro avg       0.74      0.74      0.74      2622
weighted avg       0.76      0.76      0.76      2622
```

Performance results from the CNN Model training.

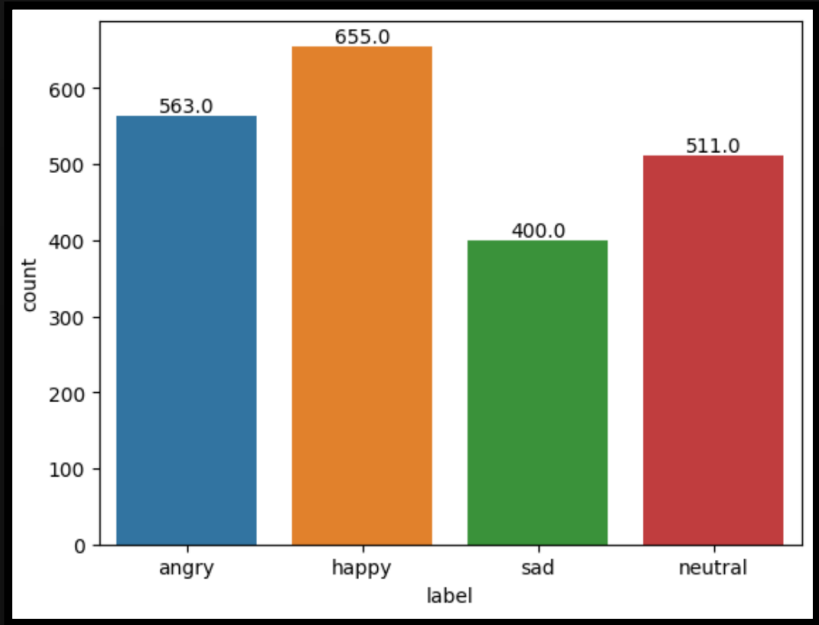

true:happy, pred:happy
true:happy, pred:happy
true:happy, pred:happy
true:happy, pred:happy
true:happy, pred:happy
true:happy, pred:happy
true:happy, pred:happy
true:happy, pred:happy
true:happy, pred:angry
true:sad, pred:sad
true:sad, pred:neutral
true:sad, pred:sad
true:sad, pred:neutral
true:sad, pred:angry
true:sad, pred:sad
true:sad, pred:sad
true:sad, pred:neutral
true:sad, pred:sad

Songs recommendation system

Iter. 1 – Song extraction from playlists

playlist_id	label	track_name	artist_name	album_name	track_link
cmh0D0Hq	angry	The Anthem	Good Charlotte	The Young and The Hopeless	https://open.spotify.com/track/0BRHnOFm6sjxN1i...
NKjOK0o75	happy	Walking On Sunshine	Katrina & The Waves	Katrina & The Waves	https://open.spotify.com/track/05wIrZSwuaVWhcv...
b6Ba0LuVc	happy	Call Me	Blues Trip	Call Me	https://open.spotify.com/track/1hc4YKpgFFFbK6Y...

Sample of the songs dataset. Extracted from mood playlists using the Spotify API.



Distribution of songs per label / ‘emotion’

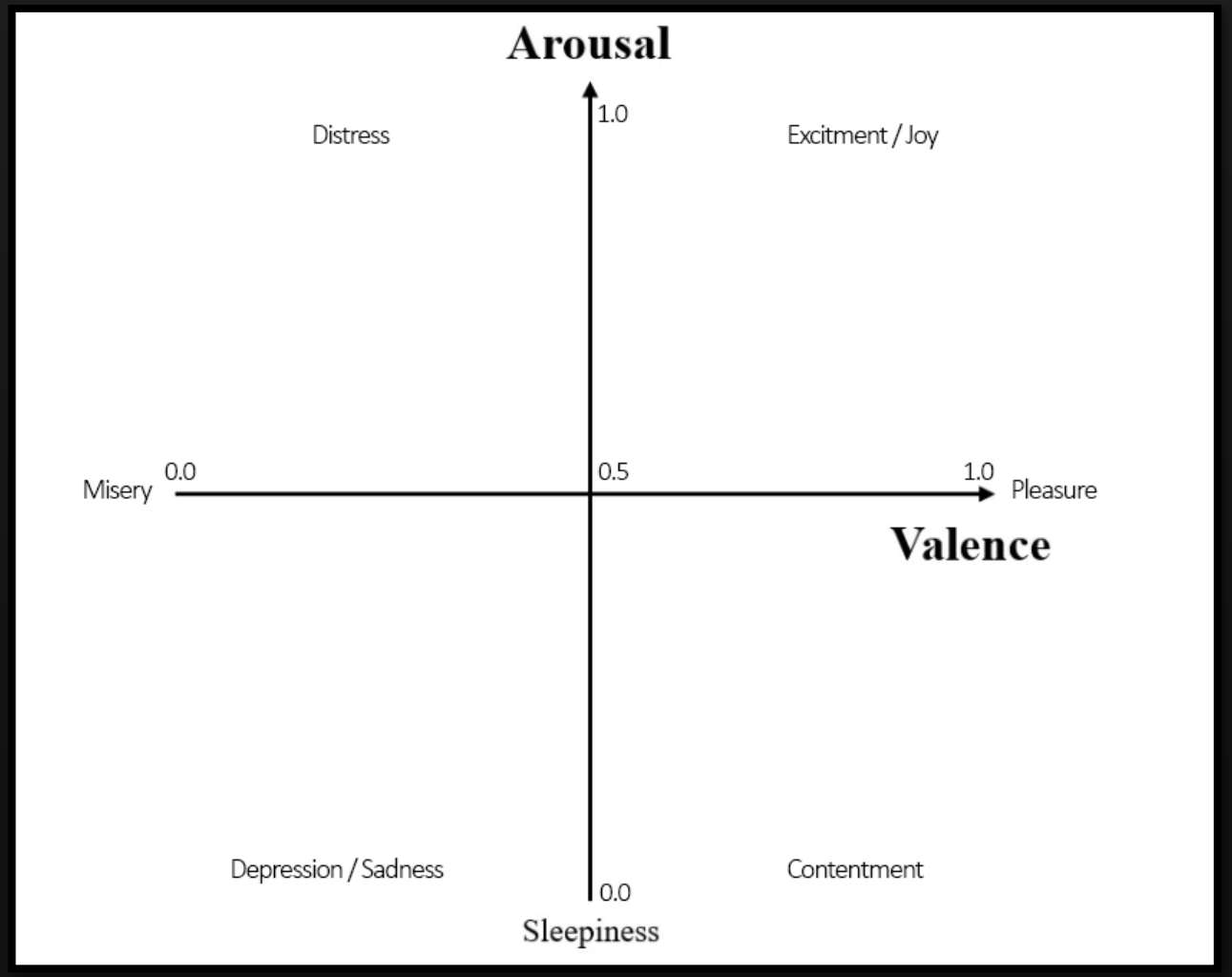
How can I scale this?

Iter. 2 – Queries inside the app

valence	year	acousticness	artists	danceability	duration_ms	energy
0.0594	1921	0.98200	['Sergei Rachmaninoff', 'James Levine', 'Berli...	0.279	831667	0.211
0.9630	1921	0.73200	['Dennis Day']	0.819	180533	0.341
0.0394	1921	0.96100	['KHP Kridhamardawa Karaton Ngayogyakarta Hadi...	0.328	500062	0.166

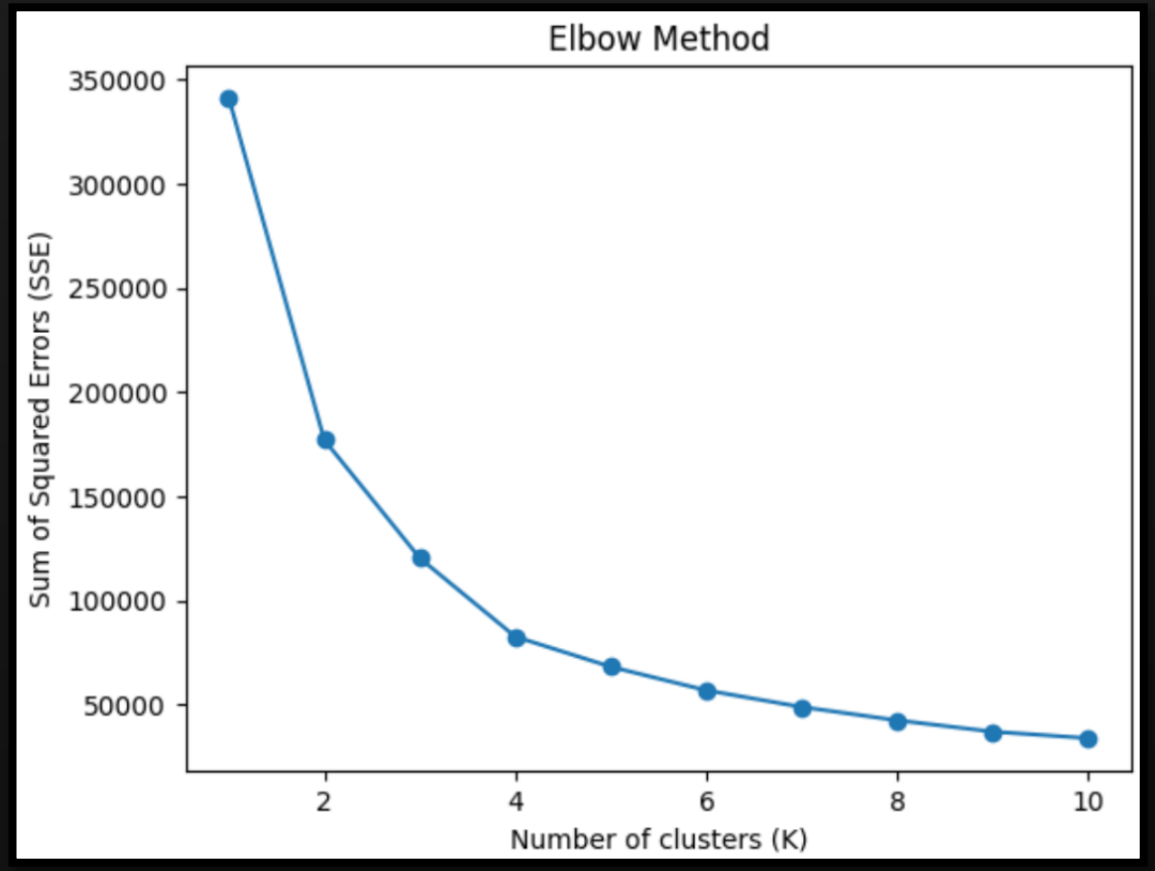
Sample of the Kaggle dataset. Contains a list of 171k songs and its ‘melody’ features.

Heuristic labelling



Rule of thumb labelling using Spotify Documentation and Valence-Arousal Plane by Russel

Iter. 3 – Unsupervised model for labelling



Cluster results

cluster	valence	energy
0	0.665325	0.325990
1	0.383620	0.741933
2	0.233710	0.213204
3	0.807275	0.714762

Clusters

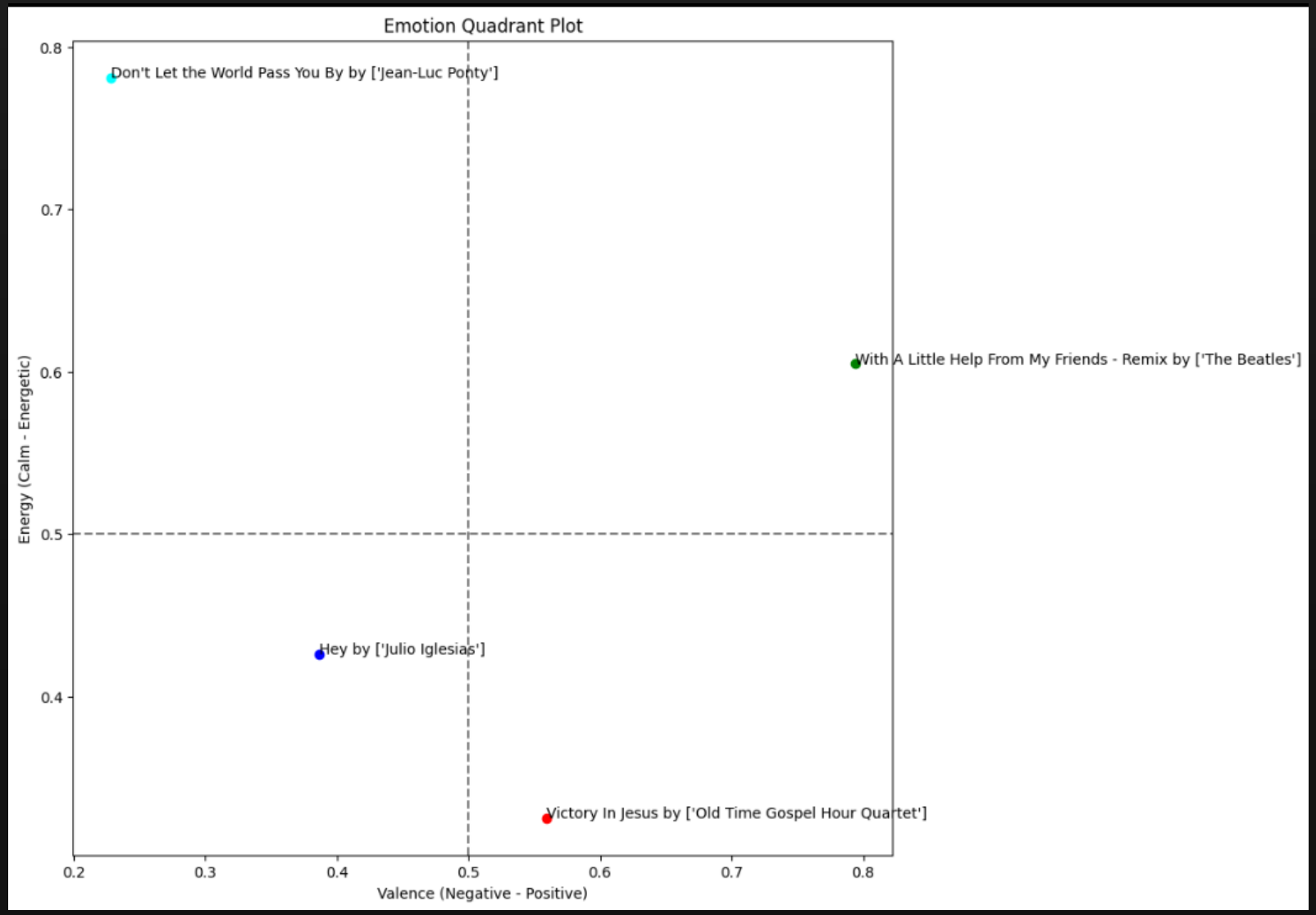
Cluster 0 - Neutral:
Valence: 0.655 (high)
Energy: 0.325 (neutral)
Explanation: Cluster 1 has neutral energy and neutral mean valence, indicating that the songs in this cluster could be classified as Neutral.

Cluster 1 - Angry:
Valence: 0.383 (neutral)
Energy: 0.741 (high)
Explanation: Cluster 0 has a high energy level and neutral valence, which suggests that the songs in this cluster could be classified as angry.

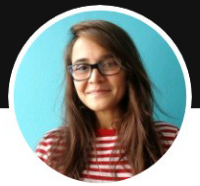
Cluster 2 - Sad:
Valence: 0.233 (low)
Energy: 0.213 (low)
Explanation: Cluster 2 has the lowest values for valence and energy which suggests that the songs in this cluster could be classified as sad.

Cluster 3 - Happy:
Valence: 0.807 (high)
Energy: 0.714 (high)
Explanation: Cluster 3 has high values for valence and energy indicating that the songs in this cluster could be classified as happy.

Samples from each cluster



Find me on LinkedIn ☺



Daniela Demarchi

Head of Product Dev. & Innovation
| Helping businesses find the righ...