Emotion Prediction Project

20191109 신승효

20191204 Oliveira De Souza Ventura Vittoria

20201118 이우

Contents

[1. Project Introduction 3](#_Toc58887083)

[2. Process 3](#_Toc58887084)

[3. Innovation 4](#_Toc58887085)

[4. Project Result 5](#_Toc58887086)

# Project Introduction

The project is called ‘Music recommendation by predicted emotion.’ We made a classification model for emotion prediction and connected it with an API created by Musicovery (<http://b2b.musicovery.com/>). This model is for a user who wants to listen to old memory songs based on their current emotional state. The purpose of this model can be summed up in three ways.

* Convenience: Users do not need to look for their music
* Nostalgia or Novel: For the elderly, the playlist can bring up an old memory. For the young, the playlist can give a chance to encounter new music.
* Entertainment: Finding their emotional state and choosing a song will be fun.

When a user gives a picture of himself, this app recommends multiple playlists to cheer him up. The user should then choose one song in the list, and it will directly show a song video from Youtube.

# Process

* 1. Main problem & Form

The main problems are “How to predict an emotion?” “How to recommend a music list that corresponds to mood?”. For the first problem, we use a classification model. For the second problem, we use the Musicovery API. The input is real human face pictures for the emotion prediction model from Kaggle(face expression recognition dataset). This model’s classification is an input to a music recommendation model, and an ultimate output is a music video on youtube chosen by the user choose.

However, this Musicovery code is a recommendation based on two variables. Also, API results are based on numbers, so there is no explicit separation between moods. Connecting the two models and separating the music were our new problems, and we’ll cover this in the next section.

* 1. Code process

The emotion prediction model

from fastai.vision import \*

from google.colab import drive

drive.mount('/content/gdrive')

path = Path('/content/gdrive/MyDrive/Data&AI/TeamProject/archive/images/category')

classes = ['angry','happy','sad']

for c in classes:

    print(c)

    verify\_images(path/c, delete=True, max\_size=500)

np.random.seed(42)

data = ImageDataBunch.from\_folder(path, train=".", valid\_pct=0.2,

        ds\_tfms=get\_transforms(), size=224, num\_workers=4,).normalize(imagenet\_stats)

learn = cnn\_learner(data, models.resnet34, metrics=error\_rate)

learn.fit\_one\_cycle(3)

interp = ClassificationInterpretation.from\_learner(learn)

interp.plot\_confusion\_matrix()

interp.plot\_top\_losses(9, figsize=(15,11))

#made a code for prediction

def imgPrediction():

    imgPath = input('Give a path of your image: ')

    img = open\_image(imgPath)

    img.show()

    pred,index,probs = learn.predict(img)

    classes = learn.data.classes

    predictedClass = classes[index]

    print('result: ',predictedClass)

imgPrediction()

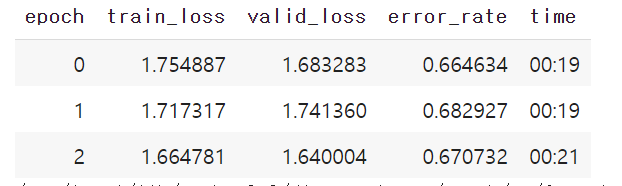
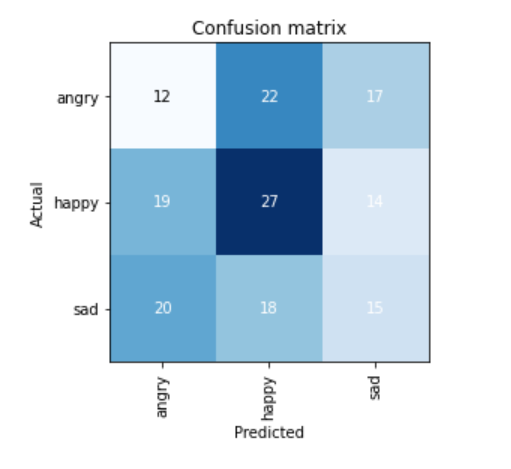
☞ Problem with prediction

Sometimes it is difficult to distinguish a person’s facial expression even from the actual human eye. The first problem of the project was this. The criteria for classifying the data itself were ambiguous. For example, see the following picture.

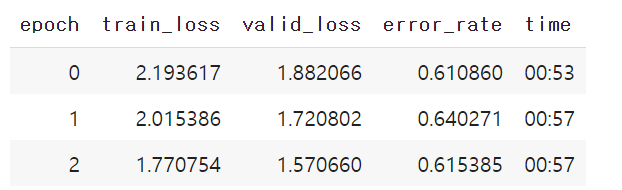
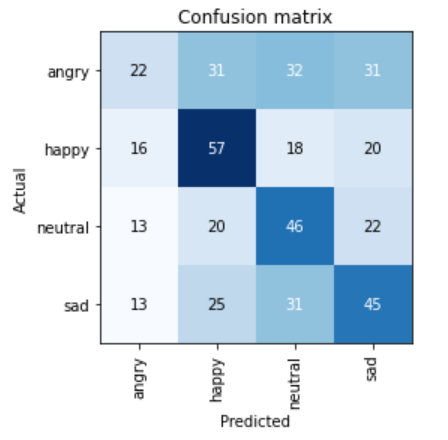


These pictures belong to the Angry category. However, when you look at these, you can see emotion as sad. It’s hard to label emotions in one category, but we’ve done the first model with not separating these pictures.

The dataset for the first model was 500 images for each category, and the result was as follows:



In the second model, we objectively reduced this dataset. Except for the confusing images, we organized the pictures that clearly show emotions from the human perspective. The results are as follows:

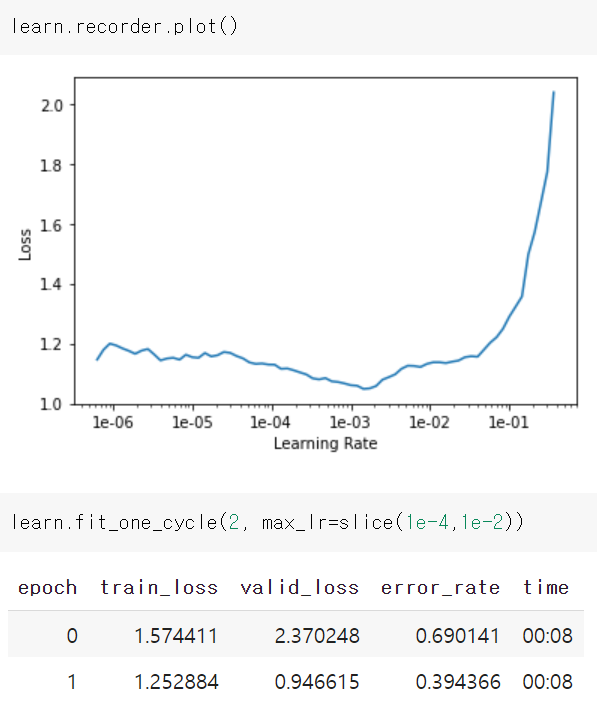
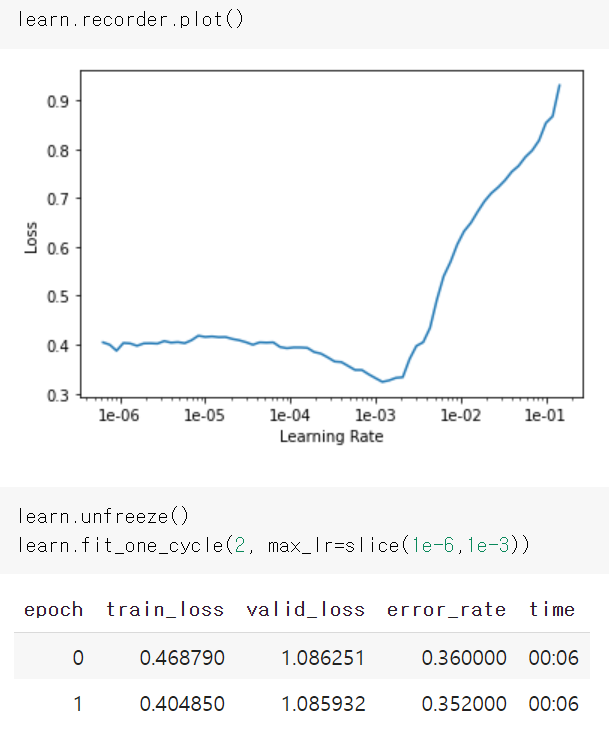


However, both models did not work well. The result kept predicting wrong, so we decided to try multiple ways and find the best model from that experience. We’ve noticed that it’s hard to say what’s the neutral emotion even with human eyes.

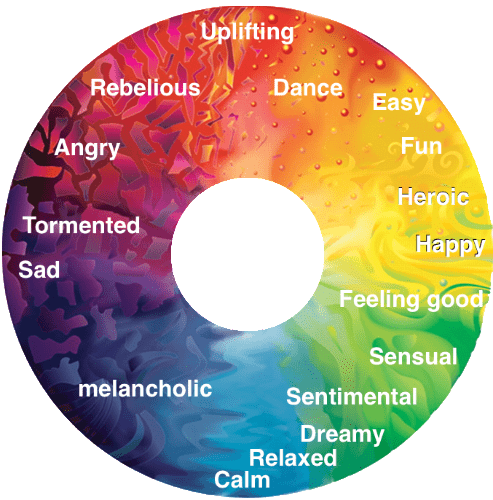
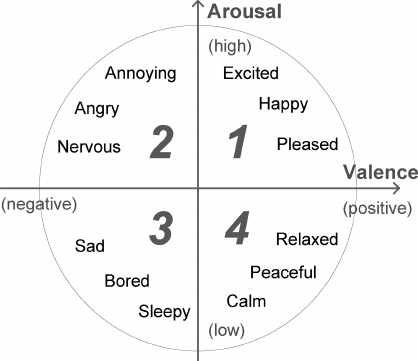
This is because the human face had a wide variety of features. Some people looked angry even when they were calmed because of their eyebrows or sharp eyes. Also, the high error was related to the neutral category a lot, so we decided to remove the neutral category and focus on predicting the three main category predictions.

We tried the 300 Kaggle dataset and unfroze it, so the error\_rate is lower. The prediction predicted the sad and angry right, but it did not predict happy at all. However, when we didn’t unfreeze it, it predicted the happy images fine. This result was confusing because the prediction kept changing. (first picture below).

Then, we tried using a google data because it has variable images. Also, we tested multiple times with three epochs and ten epochs. (second picture below) This is the best result. This could predict all three categories right even when we used images of ourselves. However, this result also changed when we tried again.



☞ Problem with connecting the two model

Musicovery code is a recommendation based on two variables. Also, the variables’ form is the number, so there is no explicit separation between moods. How to connect the two models and how to separate the music were our new problems. We solve this problem by playing with multiple numbers, like changing the variables and listening to each song it provided.

As you can see from this table, you can see that emotions are classified into four parts by two variables, Arousal and Valence. Here, the categories we have are happy, angry, and sad, corresponding to the first, second, and third quadrants. Therefore, we tried to extend one more category, which is ‘neutral.’ However, it was challenging to find neutral images, so we did not solve them in the end, as I mentioned previously.

Anyway, there are three quadrants of music, and it was impossible to say precisely what this mood value was because it had consecutive numbers. Therefore, we put in a lot of numbers, experimented with them, and set a range. Then we coded to provide a list of random numbers according to this range.

# Innovation

There are three points of difference in our project. First, in previous models, the dataset was made with only one person’s facial expression. On the contrary, our model is made up of photographs of countless people. Second, the user can upload their picture so it can be used for other purposes. For example, the user can grab a friend’s face, choose the right music, and share it. Third, the song recommendation API combining was an exciting challenge.

Furthermore, we thought about how we could improve the model. First, the music recommendation can be a neural network that can learn the user’s taste of songs by artist choice, melody, and genre. Then, the process of recommendation will become automatic. Second, the recommended music can be set differently by age. For example, it can recommend the remake version of an old song for the young. We also want to try the neutral emotion, but before that, it was hard to train the model even with three categories.

We look forward to seeing that our project has potential that other people can solve in the future.

# Project Result

This is the link to our final model.

Song Recommendation App Colab:

<https://colab.research.google.com/drive/13UD5FezHJMbsFsz0t-lXx5cSZ0zIJZnI?usp=sharing>

Song Recommendation App GitHub:

<https://github.com/vvrebellion/20191204_VITTORIA_Data-AI/blob/master/Song_Recommendation_App_.ipynb>

Emotion Classification Model Colab:

<https://colab.research.google.com/drive/1Sa3kIIvjRER3vX3mRp1gnnbEZDneYqsD?usp=sharing>

Emotion Classification Model GitHub:

<https://github.com/vvrebellion/20191204_VITTORIA_Data-AI/blob/master/EmotionPrediction_with_Google_Dataset.ipynb>

What we learned:

This project deals with human emotion recognition, which already has a better model. the differential point is that it presents the process of predicting emotions and responding to the user's condition. There was a lot to learn from the fact that music recommendations, the most representative technology of personalized services, were implemented in conjunction with machine learning.Discussion:

If the "typical features" of the emotion did not appear when judging it, it could not be assumed well. It is hard to classify emotions that are too sad or too happy, and also, in the case of weak smiles, the teeth were not revealed, and the eyes were open, making it difficult to judge with really happy emotions in data. Because sadness and anger are feelings that can coexist, so presenting the judgment results as one of the continuous values is also part of the improvement.Suggestions:

Research the effect of recommended music and learn the preferences of music that users listen to. The model could use reverse correction to the label of music using emotional judgment and music recommendations.