Machine Learning Engineer Nanodegree

Capstone Proposal

Luiz Picanço
March 19th, 2019

Proposal

Domain Background

The identification of the emotions associated with facial expressions is an intrinsically human trait, very important in social interactions and communication. Although a task considered easy for humans, it is considered very difficult for software to perform.

Software capable of performing this task satisfactorily has many applications, such as evaluating customer satisfaction in a market research and evaluation of a driver's mental fatigue driving a motor vehicle.

I consider this an interesting problem to solve, being in the field of computer vision, an area that I have a lot of interest.

This project is based on this papers: - Training Deep Networks for Facial Expression Recognition with Crowd-Sourced Label Distribution[1] - Challenges in Representation Learning: A report on three machine learning contests[2]

Problem Statement

The problem to be solved by this project is the identification of which type of emotion is associated with a human face. The types of emotions that can be identified are: neutral, happiness, surprise, sadness, anger, disgust, fear and contempt. One of the ways to solve the problem is with the use of CNN (convolutional neural network).

Datasets and Inputs

To solve the problem, 2 datasets will be used: FER2013 - Available on Kaggle, from the "Challenges in Representation Learning: Facial Expression Recognition Challenge" competition [3]. This dataset has 35,888 lines, with the following structure:

- emotion - Emotion numeric code associated with the image(0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral). It will not be used in this project.
- pixels - Image data in grayscale with 48x48 pixels.
- usage - Image utilization type(Training, PrivateTest, PublicTest). It will not be used in this project.

FER+ - Available on Github [4]. This dataset also has 35,888 rows, with the following structure: - usage - Same meaning of the usage column in FER2013 dataset - Image name - Image name - neutral, happiness, surprise, sadness, anger, disgust, fear, contempt, unknown, NF - Number of votes that each emotion received, associating it with the image

The FER+ is an enhanced version of the FER2013 dataset with 2 main advantages: - Each image has been labeled by 10 crowd-sourced taggers, providing better quality than the original FER labels - Since each image has 10 votes, each can have more than one emotion associated with it.

Solution Statement

For the solution of the proposed problem I intend to combine the two datasets, using only the pixels column of the first and the corresponding columns of the emotions of the second dataset.

This derived dataset will be divided into 3 parts: training, validation and test sets, being used in the training of a convolutional neural network. The output of the network will be a set of emotions associated with a probability.

Another approach can also be used by combining the two datasets using only the pixels column of the first and the column corresponding to the emotion with the highest votes in the second dataset. In this approach the output of the network will be an emotion.

Benchmark Model

Since this is a classification problem, I can measure the performance of the model based on the accuracy. With a naive approach it is possible to achieve 25% accuracy by always returning the happiness emotion. For this problem I intend to achieve at least 75% accuracy.

Evaluation Metrics

For the evaluation metrics I will use a confusion matrix to identify the false positives, false negatives, true positives and true negatives. Accuracy will also be used to calculate the correct predictions[5]:

Project Design

For approaching my solution I intend to use Python 3, Keras 2.2.x and Pandas, following this steps: 1. Read the 2 datasets 2. Perform an EDA on the two datasets 3. Join the first dataset with the second, creating a unique dataset, as stated in the "Datasets and Inputs" section. 4. Perform an EDA on the new dataset 5. Separate the dataset into 3 parts: training, evaluation and testing 6. Build a CNN model using Keras 7. Use the training and evaluation sets to train the model. 8. Test the network with the test set 9. Perform the evaluation metrics and compare with the benchmark model, using the test set

References

- [1] Training Deep Networks for Facial Expression Recognition with Crowd-Sourced Label Distribution1
- [2] Challenges in Representation Learning: A report on three machine learning contests2
- [3] Challenges in Representation Learning: Facial Expression Recognition Challenge (https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data)
- [4] FER+(https://github.com/Microsoft/FERPlus)
- [5] An introduction to ROC analysis(https://people.inf.elte.hu/kiss/11dwhdm/roc.pdf)