ESAPR

Emotional State Analyser for Pschyological Response

Interim Project Report

Submitted by

(Roll No: **12619001010**, Reg No: 029238 of 2019-2020)
(Roll No: **12619001087**, Reg No: 031543 of 2019-2020)
(Roll No: **12619001036**, Reg No: 013597 of 2019-2020)

Under the Supervision of

Prof. Mohuya Kar Byabartta

Department Of Computer Science And Engineering

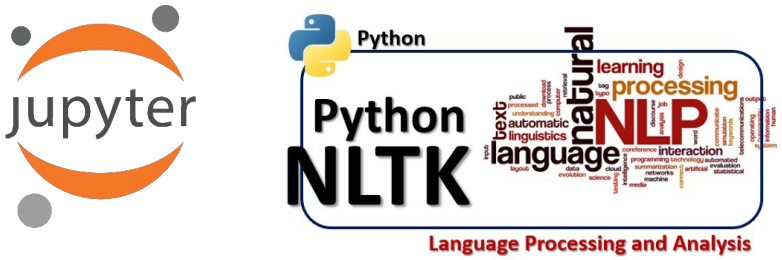
HERITAGE INSTITUTE OF TECHNOLOGY, KOLKATA

ABSTRACT

This project have the ultimate main to reduce psychological burden that people of these time face. At the end of the project we would have a working application platform which would be available in the form of Web and Android (may be iOS) as a subsystem. This is an emotion processing application which is going to identify the emotion of the user and can be used for recommending list of songs, web-articles as well as videos on the basis of it. We are exploring state of the art models in multimodal emotion recognition. We have chosen to explore textual, sound and video inputs and develop an ensemble model that gathers the information from all these sources and displays it in a clear and interpretable way.

We are naming this project in acronym of **ESAPR** (Emotional State Analyser for Pschyological Response).

Tools And Frameworks used till now



Work in Pipeline

Sentimental Analysis in real time

Multimodal Emotion Recognition Multimodal Emotion Recognition is a relatively new discipline that aims to include text inputs, as well as sound and video. This field has been rising with the development of social networks that gave researchers access to a vast amount of data. Recent studies have been exploring potential metrics to measure the coherence between emotions from the different channels. We are developing a multimodal emotion recognition platform to analyze the emotions of the user. Although we could have used simple emotion analyser, but the emotion analyser with single output emotion could have high degree of non-alignment with the facial emotion because a person may show a mixture of facial feature in the intense situations. We are going to explore several categorical targets depending on the input considered. We have chosen to diversify the data sources we used depending on the type of data considered.

Video input from a live webcam or stored from an MP4 or WAV file, from the images.

Video Analysis

Challenges such as emotion recognition can typically not be solved though classical machine learning techniques. All the recent research papers focus on several deep learning techniques, some of which include Artificial Neural Networks (ANN), Convolution Neural Network (CNN), Region-CNN (R-CNN), Fast R-CNN, Recurrent Neural Network (RNN) or Long Short-Term Memory (LSTM). The aim of the following section is to develop the bases that lead to Convolutional Neural Networks (CNN)

Data Source

For the video data sets, we are using the popular **FER2013** Kaggle Challenge data set. The data consists of 48x48 pixel grayscale images of faces. The faces have been automatically registered so that the face is more or less centered and occupies about the same amount of space in each image. The data set remains quite challenging to use, since there are empty pictures, or wrongly classified images.

- https://www.kaggle.com/c/challenges-inrepresentation-learning-facial-expression-recognition-challenge/data
- https://www.kaggle.com/competitions/challenges-in-representation-learning-facial-expression-recognition-challenge/data

Model

The model we have chosen is an **XCeption** model, since it outperformed the other approaches we developed so far.

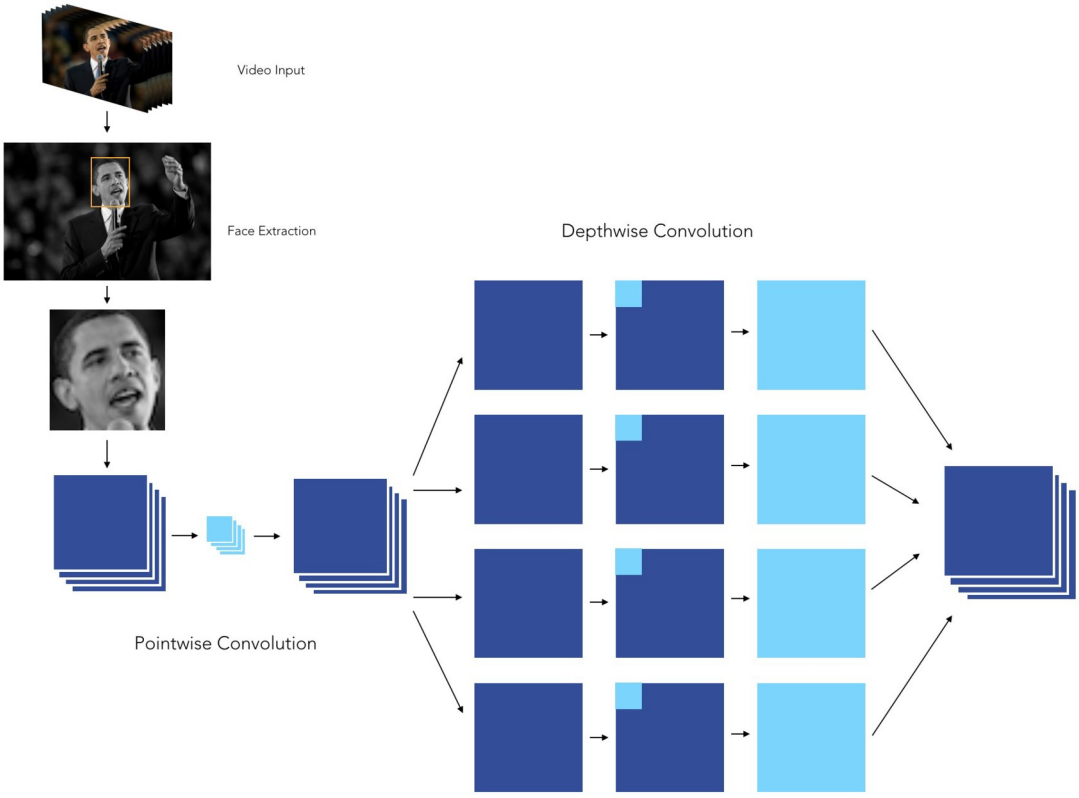
We tuned the model with :

- Data augmentation
- Early stopping
- Decreasing learning rate on plateau
- L2-Regularization
- Class weight balancing
- And kept the best model

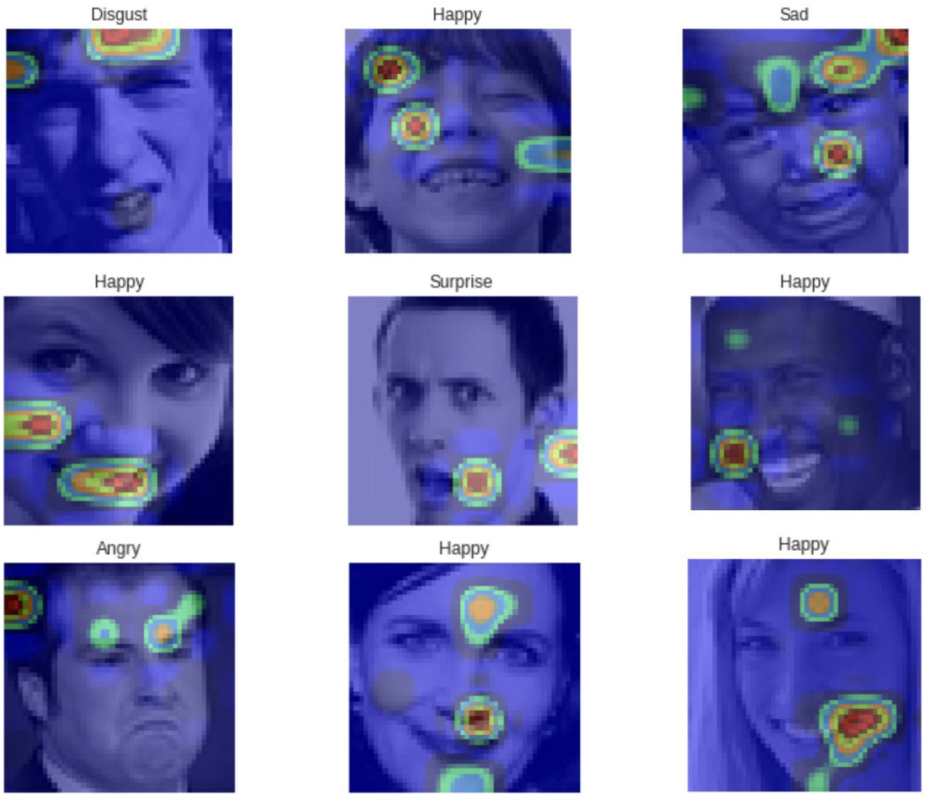
As you might have understood, the aim was to limit overfitting as much as possible in order to obtain a robust model.

- To know more on how we prevented overfitting, check this article : https://maelfabien.github.io/deeplearning/regu/
- To know more on the **Xception** model, check this article : https://maelfabien.github.io/deeplearning/xception/

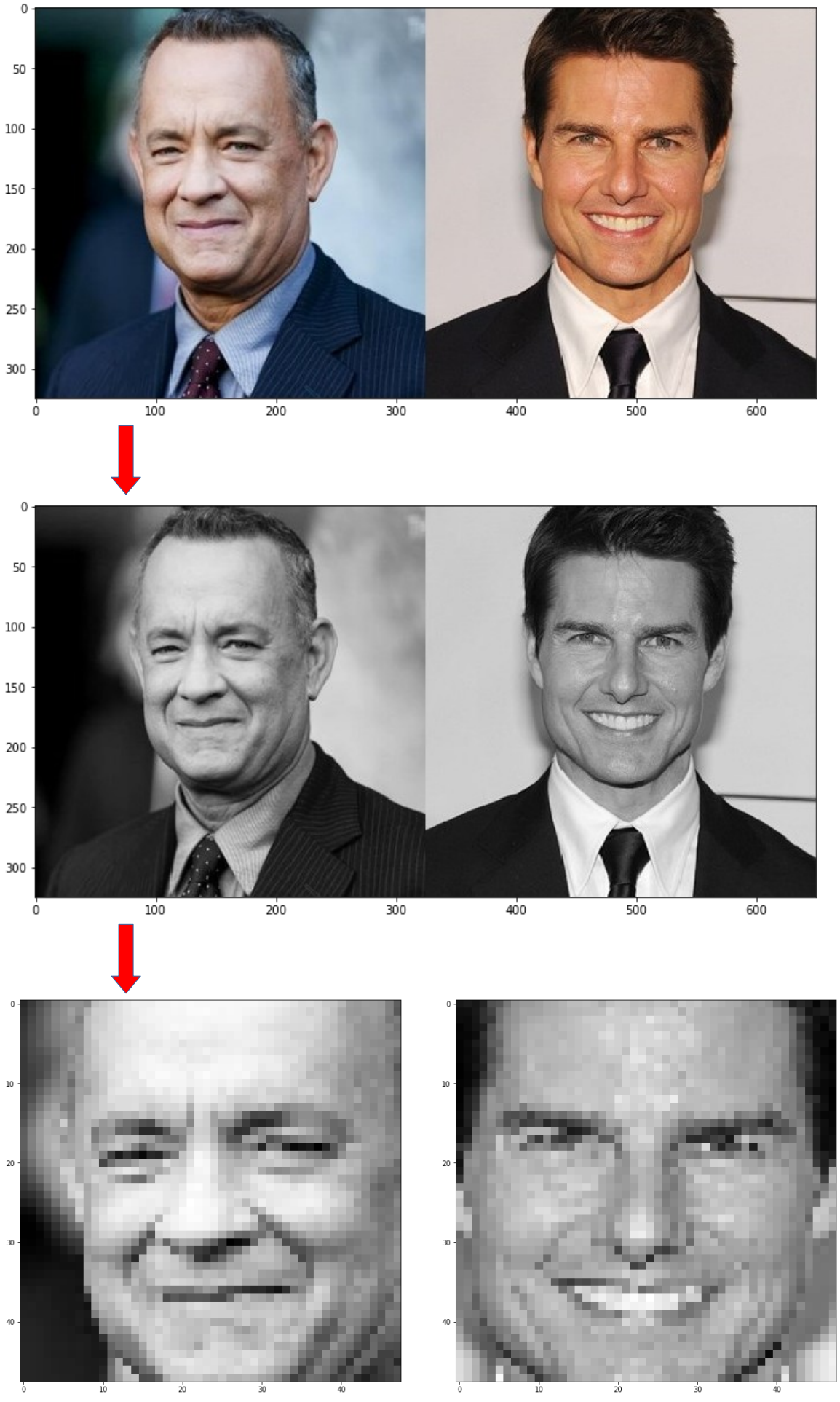
The XCeption architecture is based on DepthWise Separable convolutions that allow to train much fewer parameters, and therefore reduce training time on Colab's GPUs to less than 90 minutes.



When it comes to applying CNNs in real life application, being able to explain the results is a great challenge. We can indeed plot class activation maps, which display the pixels that have been activated by the last convolution layer. We notice how the pixels are being activated differently depending on the emotion being labeled. The happiness seems to depend on the pixels linked to the eyes and mouth, whereas the sadness or the anger seem for example to be more related to the eyebrows.



Overall Process



To enhance the overall process. These parameters must also be taken care of:

- Frequency of eye blink
- Detect Keypoints to plot them
- Face Alignment

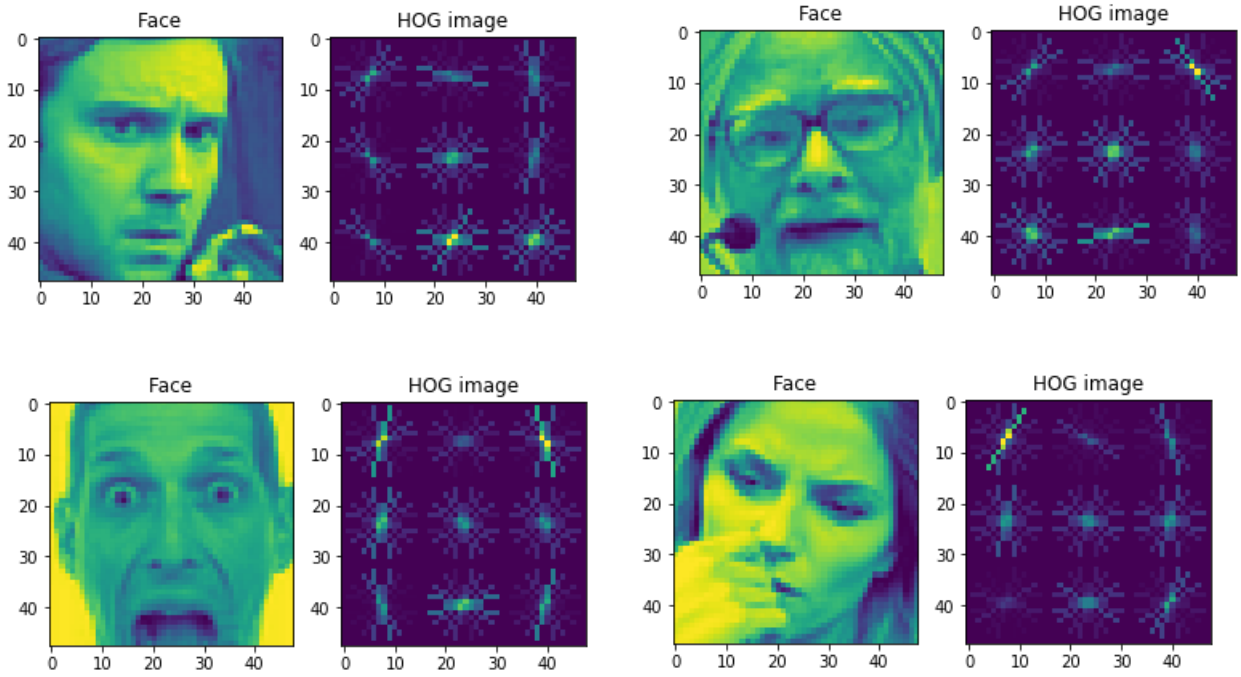
Results

The training dataset, as per the different emotional states.

1. Disgust
2. Fear
3. Happy
4. Sad
5. Surprise
6. Neutral

This simple architecture produces over 4,40,000 parameters to estimate. The computation time is around 8 hours on local machine. In order to prevent overtting, we also apply Keras built-in data generation module.

The histogram of oriented gradients **(HOG)** is a feature descriptor used in computer vision and image processing for the purpose of object detection. The technique counts occurrences of gradient orientation in localized portions of an image.



Text-based Emotion Classifier:

Problem Formulation:

Text messages omit *tone* and *emotion* - causing potential misinterpretation in the meaning of the text messages.

Data Source:

Source: https://www.kaggle.com/praveengovi/emotions-dataset-for-nlp

Libraries Used :

- sklearn
- nltk
- numpy
- pandas
- matplotlib

Speech Emotion Recognition

The aim of this section is to explore speech emotion recognition techniques from an audio recording. he data set used for training is the Ryerson Audio-Visual Database of Emotional Speech and Song: https://zenodo.org/record/1188976#.XA48aC17Q1J

Libraries Used :

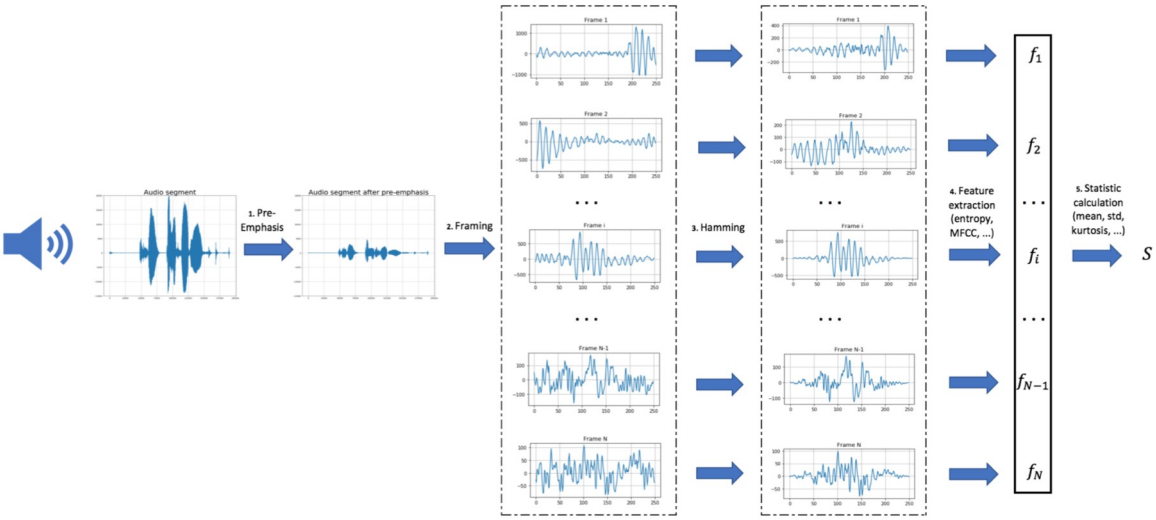
```
Python : 3.6.5
Scipy : 1.1.0
Scikit-learn : 0.20.1
Tensorflow : 1.12.0
Keras : 2.2.4
Numpy : 1.15.4
Librosa : 0.6.3
Pyaudio : 0.2.11
Ffmpeg : 4.0.2
```

Models Used :

SVM

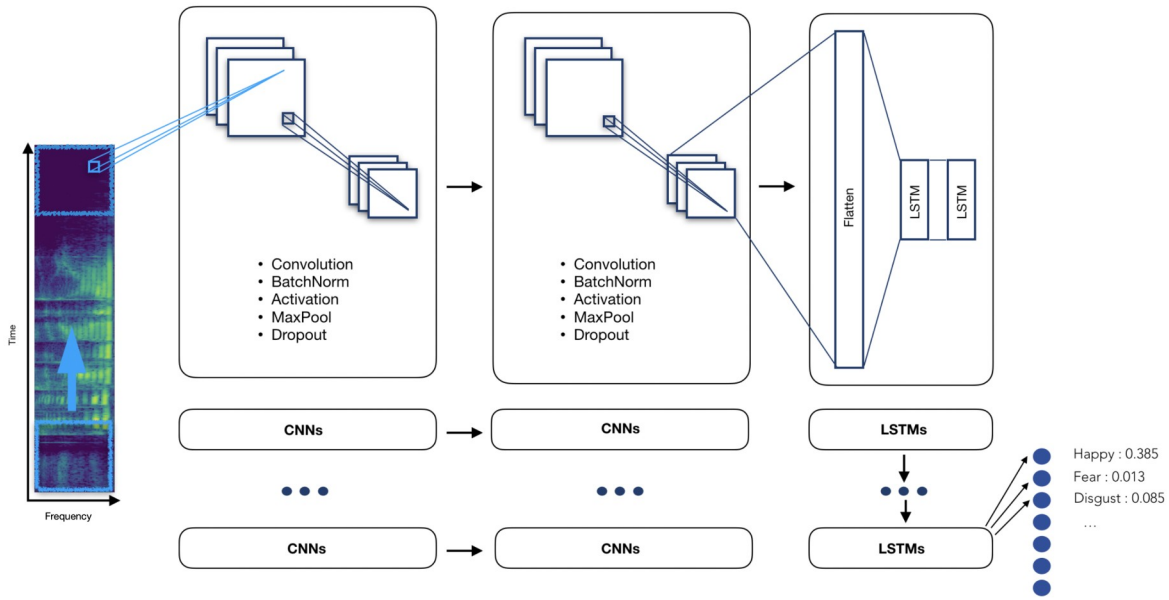
Classical approach for Speech Emotion Recognition consists in applying a series of filters on the audio signal and partitioning it into several windows (fixed size and time-step). Then, features from time domain (Zero Crossing Rate, Energy and Entropy of Energy) and frequency domain (Spectral entropy, centroid, spread, flux, rolloff and MFCCs) are extracted for each frame. We compute then the first derivatives of each of those features to capture frame to frame changes in the signal. Finally, we calculate the following global

statistics on these features: mean, median, standard deviation, kurtosis, skewness, 1% percentile, 99% percentile, min, max and range and train a simple SVM classifier with rbf kernel to predict the emotion detected in the voice.



TimeDistributed CNNs

The main idea of a Time Distributed Convolutional Neural Network is to apply a rolling window (fixed size and time-step) all along the log-mel-spectrogram. Each of these windows will be the entry of a convolutional neural network, composed by four Local Feature Learning Blocks (LFLBs) and the output of each of these convolutional networks will be fed into a recurrent neural network composed by 2 cells LSTM (Long Short Term Memory) to learn the long-term contextual dependencies. Finally, a fully connected layer with softmax activation is used to predict the emotion detected in the voice.



To limit overfitting during training phase, we split our data set into train (80%) and test set (20%). Following show results obtained on test set:

Model	Accuracy
SVM on global statistic features	68.3%
Time distributed CNNs	76.6%