Multi-modal LSTM Neural Network with Randomized Training for Personality-Traits Recognition

August 18, 2016

1 Team details

- Team name : evolgen
- Team leader name : Arulkumar S
- Team leader address, phone number and email

  Room #440, Sindhu Hostel,
  IIT Madras, Chennai, India - 600036
  Mobile: +918973234334
  Email: aruls@cse.iitm.ac.in

- Rest of the team members : Vismay Patel, Ashish Mishra
- Team website URL (if any)
- Affiliation : Computer Vision Lab, IIT Madras

2 Contribution details

- Title of the contribution :

  Multi-modal LSTM Neural Network with Randomized Training for Personality-Traits Recognition

- Final score :

  Validation score: 0.9133
  Extraversion: 0.914548
  Agreeableness : 0.915749
  Conscientiousness: 0.913594
  Neuroticism : 0.909814
  Openness: 0.913069

- General method description :

  Features used:

  The full-length audio and video are split into 6 non-overlapping equal-length partitions. From each partition of audio, a 68-dimensional mid-term feature vector (mean and standard deviation of features from 5.1) is extracted. In total, the audio features have 6 feature vectors of 68 dimensions. From the frames available in each partition of video, we extract the 3D Face-Aligned RGB frames each of dimension 112 x 112 x 3. These raw frames are used in our multimodal LSTM architecture (refer 1).

Multi-modal LSTM Neural Network:

The proposed network is a two-input LSTM architecture containing two branches for audio, video feature processing. As already mentioned, the given whole video (including audio) is splitted into 6 non-overlapping equi-length parts and the appropriate audio, video features are extracted. During training, the corresponding feature vector (the cumulative audio feature = 68 dimension, a single 3D Face-aligned image = 112x112x3 dimension) of every partition (in same temporal ordering) is passed to the LSTM network. We hypothesize that the model learns to recognize the personality-traits from the temporally-maintained sequence of audio+video data.

Randomized Training:

A video of length of 15 seconds will have 450 frames (30 frames / second). When we split the video as 6 non-overlapping equal-length partitions, each partition will have 75 3D Face-aligned RGB frames (2.5 seconds). During training, there is only one 3D Face-aligned frame per partition is selected randomly and passed to the model along with the audio feature. The hypothesis is that, the randomly selected 3D Face-aligned RGB images from every partition will capture the variations in facial expression and the audio feature will capture the tone-variations, voice-expression mappings. The training is done in a randomized fashion that at every training iteration, the audio-features remain same and the RGB Face images are randomly selected.

The parameters used for Stochastic Gradient Descent (SGD) are,
learning rate = 0.05
weight decay = 5e-4
momentum = 0.9
learning rate decay = 1e-4
batch size = 128

Testing:

Since, the model is trained in randomized way, the evaluation is run for 10 times and the average of 10 results is given as the final Personlaity-traits recognition values.

- References : refer 6.2
- Representative image / diagram of the method : refer figure 1
- Describe data preprocessing techniques applied (if any)

The data preprocessing of raw videos is applied for retrieving audio features and visual features separately:

  - **Audio features**

    We extract and use hand-crafted Audio features (refer the table 5.1) from the python-based Opensource audio-processing library 'pyAudioAnalysis'[2, 5]
    Specifically, we use mid-term (mean and standard deviation) features calculated by splitting the total length of Audio into 6 parts and calculating the mean, standard deviation of features in the table 5.1. The total feature dimensions of Audio per every video is,

    6 parts x [(34 dimension of mean) + (34 dimension of standard deviation)] = 6 x 68 dimensions

  - **Visual feature(s)**
    We use the images of 3D-aligned faces extracted using an open-source library called "OpenFace"[3, 1].

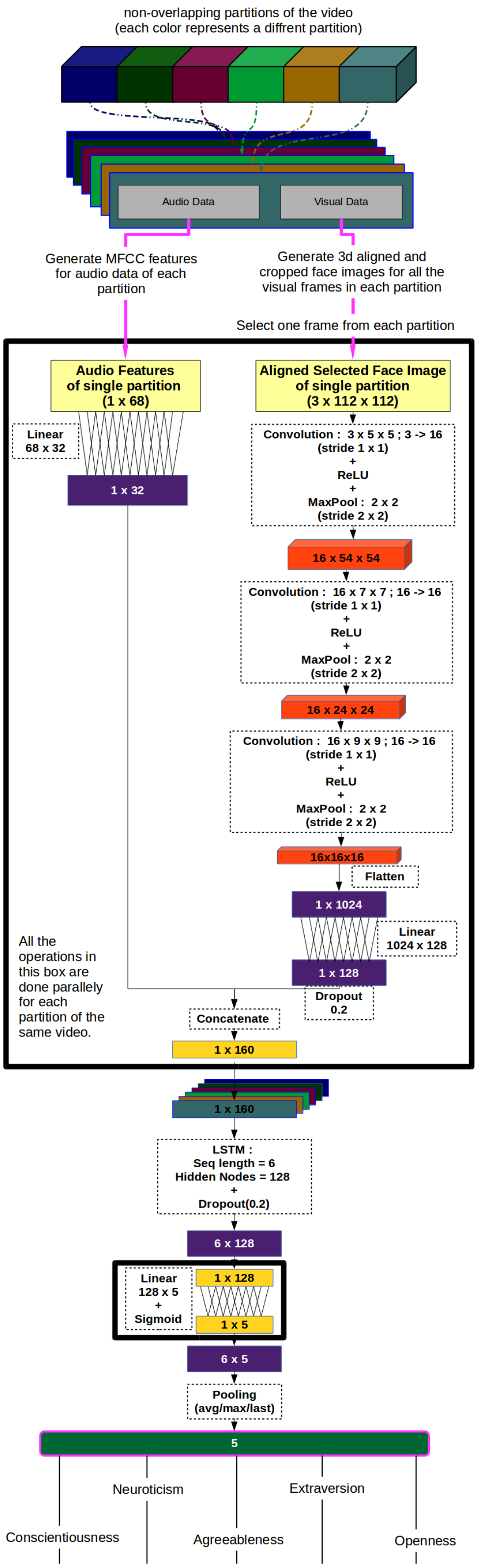
non-overlapping partitions of the video
(each color represents a diffrent partition)
Audio Data
Visual Data
Generate MFCC features for audio data of each partition
Generate 3d aligned and cropped face images for all the visual frames in each partition
Select one frame from each partition
Audio Features of single partition (1 x 68)
Aligned Selected Face Image of single partition (3 x 112 x 112)
Linear 68 x 32
1 x 32
Convolution : 3 x 5 x 5 ; 3 -> 16 (stride 1 x 1) + ReLU + MaxPool : 2 x 2 (stride 2 x 2)
16 x 54 x 54
Convolution : 16 x 7 x 7 ; 16 -> 16 (stride 1 x 1) + ReLU + MaxPool : 2 x 2 (stride 2 x 2)
16 x 24 x 24
Convolution : 16 x 9 x 9 ; 16 -> 16 (stride 1 x 1) + ReLU + MaxPool : 2 x 2 (stride 2 x 2)
16x16x16
Flatten
1 x 1024
Linear 1024 x 128
1 x 128
Dropout 0.2
All the operations in this box are done parallely for each partition of the same video.
Concatenate
1 x 160
1 x 160
LSTM : Seq length = 6 Hidden Nodes = 128 + Dropout(0.2)
6 x 128
Linear 128 x 5 + Sigmoid
1 x 128
1 x 5
6 x 5
Pooling (avg/max/last)
5
Neuroticism
Extraversion
Conscientiousness
Agreeableness
Openness

Figure 1: Multi-modal LSTM Neural Network architecture

3 Visual Analysis

3.1 Face Detection Stage

We use an open-source library called “OpenFace”[3, 1] for face landmark alignment. The library in turn uses **”dlib”[7, 6]** implementation for face detection.

3.2 Face Landmarks Alignment Stage

We use an open-source library called “OpenFace”[3, 1] for face landmark alignment. The library in turn uses **”Constrained Local Neural Fields”**[4] for landmark detection

4 Personality Trait recognition from Visual data

4.1 Features / Data representation

The extracted 3D Face-Aligned RGB images are of size 112 x 112 x 3.

Some examples are shown below:

(a) ADa9rkwF2uA.001

(b) iAgu-wDe2MQ.002

4.2 Dimensionality reduction

We are not using any dimensionality reduction technique explicitly.

4.3 Compositional model

not used

4.4 Learning strategy

We apply three consecutive Convolution, ReLU and Maxpooling layers (refer figure 1) on top of the 3D Face-Aligned images. In the later stage, the output of visual data features are concatenated with the audio data features and a multimodal network is trained using Stochastic Gradient Descent method.

4.5 Method complexity

The visual data processing branch in the model (1) contains 165728 parameters to be tuned.

5 Personality Trait recognition from Audio data

5.1 Features / Data representation

The full length audio is splitted into 6 non-overlapping parts and the mid-term features (namely, the mean and standard deviation of 34 dimensions = 68 dimensions of each part) are extracted using the library “pyAudioAnalysis” [5]. The 34-dimensional features include,

Feature ID	Feature Name	Description
1	Zero Crossing Rate	The rate of sign-changes of the signal during the duration of a particular frame.
2	Energy	The sum of squares of the signal values, normalized by the respective frame length.
3	Entropy of Energy	The entropy of sub-frames' normalized energies. It can be interpreted as a measure of abru
4	Spectral Centroid	The center of gravity of the spectrum.
5	Spectral Spread	The second central moment of the spectrum.
6	Spectral Entropy	Entropy of the normalized spectral energies for a set of sub-frames.
7	Spectral Flux	The squared difference between the normalized magnitudes of the spectra of the two succe
8	Spectral Rolloff	The frequency below which 90% of the magnitude distribution of the spectrum is concentr
9-21	MFCCs	Mel Frequency Cepstral Coefficients form a cepstral representation where the frequency ba
22-33	Chroma Vector	A 12-element representation of the spectral energy where the bins represent the 12 equal-t
34	Chroma Deviation	The standard deviation of the 12 chroma coefficients.

5.2 Dimensionality reduction

We are not using any dimensionality reduction technique explicitly.

5.3 Compositional model

not used.

5.4 Learning strategy

We apply a linear layer with 32 neurons on the audio features. In the later stage, we combine output of this linear layer (32 dimension) with the output of Visual branch of the neural network. The training is done with SGD optimizer.

5.5 Other techniques

Given a single video, we first split it into some fixed number of non-overlapping temporal blocks(6) and then calculate the hand-crafted features on each of these block independently. The audio features sequence is passed to the Neural network as per temporal ordering.

5.6 Method complexity

The audio-features-processing branch uses ((68 dimensions + 1 bias) x (32 dimensions)) 2208 parameters in total.

6 Multimodal Personality Trait recognition

6.1 Data Fusion Strategies

Late fusion strategy of audio, visual features is followed in the architecture (1). The audio, visual features are processed individually in the two-input model and then, the processed features are fused at the semantic phase. The fused features are passed through a LSTM for learning temporal structure and the final output of LSTM is used for calculating the personality-traits recognition values from a Sigmoidal regression.

6.2 Global Method Description

- Total method complexity: The model has 316549 parameters to be learned, in total.
- Which pre-trained or external methods have been used (for any stage, if any)

  –None–

- Which additional data has been used in addition to the provided ChaLearn training and validation data (at any stage, if any)

  –None–

- Qualitative advantages of the proposed solution

  we are effectively using only 6 random frames (temporal sequence is maintained) and the audio features from a video for predicting the Personality-traits recognition values. It essentially reflects the fact that the Neural network is able to learn effectively with only less exposure of the subject.

- Results of the comparison to other approaches (if any)

- Novelty degree of the solution and if is has been previously published

  - Randomized Training using only Minimal frames (6 frames) for whole video & Online batch generation:

  The 6 input frames for model is selected randomly from all available frames by keeping temporal ordering in mind. This randomized training "regularizes" the learning effectively and increases the generalizability of the model.

  - Average of 10 tests are taken as final result

  Since the training is done in randomized fashion, the testing is also carried out in the same randomized way and the average of 10 evaluations is given as the Personality-traits recognition results.

- Language and implementation details (including platform, memory, parallelization requirements):

  Lua based Torch7 is used for implementation
  Platform: Ubuntu 14.04
  GPU : GeForce GTX TITAN X
  CUDA: 7.5

- Human effort required for implementation, training and validation?

  5 person days of coding (implementation)

- Training/testing expended time?

  1 day for training the model & 10 minutes for testing

- General comments and impressions of the challenge? what do you expect from a new challenge in face and looking at people analysis? The videos provided are of mostly from Indoor conditions. It will be interesting to work on the videos in wild.

References

[1] Openface. `https://github.com/TadasBaltrusaitis/OpenFace`. a state-of-the art open source tool intended for facial landmark detection, head pose estimation, facial action unit recognition, and eye-gaze estimation.

[2] pyaudioanalysis. `https://github.com/tyiannak/pyAudioAnalysis`. an open Python library that provides a wide range of audio-related functionalities.

[3] Tadas Baltru, Peter Robinson, Louis-Philippe Morency, et al. Openface: an open source facial behavior analysis toolkit. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–10. IEEE, 2016.

[4] Tadas Baltrusaitis, Peter Robinson, and Louis-Philippe Morency. Constrained local neural fields for robust facial landmark detection in the wild. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 354–361, 2013.

[5] Theodoros Giannakopoulos. pyaudioanalysis: An open-source python library for audio signal analysis. *PloS one*, 10(12):e0144610, 2015.

[6] Davis E King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10(Jul):1755–1758, 2009.

[7] Davis E King. Max-margin object detection. *arXiv preprint arXiv:1502.00046*, 2015.