Abstract : UMONS submission for the OMG-Emotion Challenge

Jean-Benoit Delbrouck
TCTS Lab, University of Mons, Belgium
jean-benoit.delbrouck@umons.ac.be

Abstract

This paper describes the UMONS solution for the OMG-Emotion Challenge[1]. We explore a context-dependent architecture where the arousal and valence of an utterance are predicted according to its surrounding context (i.e. the preceding and following utterances of the video). We report an improvement when taking into account context for both unimodal and multimodal predictions. Our code is made publicly available[2].

1 Data

The organizers specially collected and annotated a One-Minute Gradual-Emotional Behavior dataset (OMG-Emotion dataset) for the challenge. The dataset is composed of Youtube videos chosen through keywords based on long-term emotional behaviors such as "monologues", "auditions", "dialogues" and "emotional scenes". An annotator has to watch a whole video in a sequence so that he takes into consideration the contextual information before annotating the arousal and valence for each utterance of a video. The dataset provided by the organizers contains a train split of 231 videos composed of 2442 utterances and validation split of 60 videos composed of 617 utterances. For each utterance, the gold arousal and valence level is given.

2 Architecture

Because context is taken into account during annotation, we propose a context-dependent architecture (Poria et al., 2017) where the arousal and valence of an utterance is predicted according to the surrounding context. Our model consists of three successive stages:

- A context-independent unimodal stage to extract linguistic, visual and acoustic features per utterance
- A context-dependent unimodal stage to extract linguistic, visual and acoustic features per video
- A context-dependent multimodal stage to make a final prediction per video

2.1 Context-independent Unimodal stage

Firstly, the unimodal features are extracted from each utterance separately. We use a mean square error as loss function :

$$\mathcal{L}_{mse} = \frac{1}{N} L^2_{p=2}(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{N} \sum_{i=1}^{N} (x_i - y_i)^2$$

where N is the number of utterances predicted, $\boldsymbol{x}$ the prediction vector for arousal or valence and y_i the ground truth vector.

Below, we explain the linguistic, visual and acoustic feature extraction methods.

2.1.1 Convolutional Neural Networks for Sentences

For each utterance, a transcription is given as a written sentence. We train a simple CNN with one layer of convolution (Kim, 2014) on top of word vectors obtained from an unsupervised neural language model (Mikolov et al., 2013). More precisely, we represent an utterance (here, a sentence) as a sequence of k-dimensional word2vec vectors concatenated. Each sentence is wrapped to a window of 50 words which serves as the input to the CNN. Our model has one convolutional layer of three kernels of size 3, 4 and 2 with 30, 30 and 60 feature maps respectively. We then apply a max-overtime pooling operation over the feature

[1] https://www2.informatik.uni-hamburg.de/wtm/OMG-EmotionChallenge/

[2] https://github.com/jbdel/OMG_UMONS_submission/

map and capture the most important feature, one with the highest value, for each feature map. Each kernel and max-pooling operation are interleaved with ReLu activation function. Finally, a fully connected network layer FC_{out} of size $[120 \rightarrow 2]$ predicts both arousal and valence of the utterance. We extract the 120-dimensional features of an utterance before the FC_{out} operation.

2.1.2 3D-CNN for visual input

In this section, we explain how we extract features of each utterance's video with a 3D-CNN (Ji et al., 2013). A video is a sequence of frames of size $W \times H \times 3$. The 3D convolution is achieved by convolving a 3D-kernel to the cube formed by stacking multiple successive video frames together. By this construction, the feature maps in the convolution layer is connected to multiple frames in the previous layer and therefore is able to capture the temporal information. In our experiments, we sample 32 frames of size 32×32 per video, equally inter-spaced, so that each video in the dataset $\in \mathbb{R}^{32\times32\times32\times3}$. Our CNN consists of 2 convolutional layers of 32 filters of size $5\times5\times5$. Each layer is followed by two max-pooling layers of size $4 \times 4 \times 4$ and $3 \times 3 \times 3$ respectively. Afterwards, two fully connected network layers FC_{out_1} $[864 \rightarrow 128]$ and FC_{out_2} $[128 \rightarrow 2]$ map the CNN outputs to a predicted arousal and valence level. We extract the 128-dimensional features of an utterance before the FC_{out_2} operation.

2.1.3 OpenSmile for audio input

For every utterance's video, we sample a Waveform Audio file at 16 KHz frequency and use OpenSmile (Eyben et al., 2010) to extract 6373 features from the IS13-ComParE configuration file. To reduce the number, we only select the k-best features based on univariate statistical regression tests where arousal and valence levels are the targets. We pick $k = 80$ for both arousal and valence tests and merge features indexes together. We ended up with 121 unique features per utterances.

2.2 Context-dependent Unimodal stage

In this section, we stack the utterances video-wise for each modality. Lets consider a modality m of utterance feature size k, a video $\boldsymbol{V}_i$ is the sequence of utterances vectors $(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_T)_i$ where $\boldsymbol{x}_j \in \mathbb{R}^k$ and T is the number of utterances in $\boldsymbol{V}_i$. We now have a set of modality videos $\mathcal{V}_m = (\boldsymbol{V}_1, \boldsymbol{V}_2, \ldots, \boldsymbol{V}_N)_m$ where N is number of video in the dataset.

In previous similar work (Poria et al., 2017), the video matrice $\boldsymbol{V}_i$ was the input of a bi-directional LSTM network to capture previous and following context. We argue that, especially if the video has many utterances, the context might be incomplete or inaccurate for a specific utterance. We tackle the problem by using self-attention (sometimes called intra-attention). This attention mechanism relates different positions of a single sequence in order to compute a representation of the sequence and has been successfully used in a variety of tasks (Parikh et al., 2016; Lin et al., 2017; Vaswani et al., 2017). More specifically, we use the "transformer" encoder with multi-head self-attention to compute our context-dependent unimodal video features.

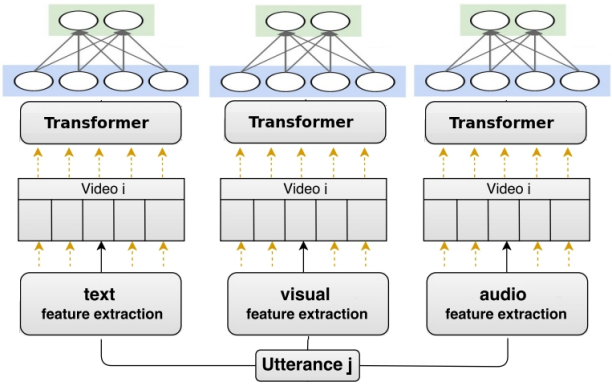

Figure 1: Overview of the Context-dependent Unimodal stage. Each utterance's arousal and valence level are predicted through a whole video

2.2.1 Transformer encoder

The encoder is composed of a stack of N identical blocks. Each block has two layers. The first layer is a multi-head self-attention mechanism, and the second is a fully connected feed-forward network. Each layer is followed by a normalization layer and employs a residual connection. The output of each layer can be rewritten as the following

$$\text{LayerNorm}(x + \text{layer}(x))$$

where layer(x) is the function implemented by the layer itself (multi-head attention or feed forward).

2.2.2 Multi-Head attention

Let d_k be the queries and keys dimension and d_v the values dimension, the attention function is the

dot products of the query with all keys, divide each by$\sqrt{d_k}$, and apply a softmax function to obtain the weights on the values :

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

Authors found it beneficial to linearly project the queries, keys and values h times with different learned linear projections to d_k, d_k and d_v dimensions. The output of the multi-head attention is the concatenation of the h number of d_v values.

We pick $d_k = d_v = 64, h = 8, N = 2$.

2.2.3 Dense output layer

The output of each utterance's transformer goes through a last fully connected layer FC_{out} of size $[m, 2]$ to predict both arousal and valence level. Because we make our prediction per video, we propose to include the concordance correlation coefficient in our loss function. We define $\mathcal{L}_{\text{ccc}} = 1 - p_c$ where

$$p_c = \frac{2\sigma_{xy}^2}{\sigma_x^2 + \sigma_y^2 + (\mu_x - \mu_y)^2}$$

We now want to minimize

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{mse}} + 0.25 \times \mathcal{L}_{\text{ccc}}$$

for both arousal and valence value. In addition to lead to better results, we found it to give the model more stability between evaluation and reproducibility between runs.

3 Context-dependent Multimodal stage

This section is similar to the previous section, except that we now have only one set of video $\mathcal{V} = (\boldsymbol{V}_1, \boldsymbol{V}_2, \ldots, \boldsymbol{V}_N)$ where each video $\boldsymbol{V}_i$ is composed of multimodal utterances $\boldsymbol{x}_j = (\boldsymbol{x}_{\text{linguistic}}, \boldsymbol{x}_{\text{visual}}, \boldsymbol{x}_{\text{audio}})$. In our experiments, we tried two types of fusion.

A Concatenation
We simply concatenate each modality utterance-wise. The utterance $\boldsymbol{x}_j$ can be rewritten $\boldsymbol{x}_j = \boldsymbol{x}_{\text{linguistic}} \oplus \boldsymbol{x}_{\text{visual}} \oplus \boldsymbol{x}_{\text{audio}}$ where $\oplus$ denotes concatenation.

B Multimodal Compact Bilinear Pooling
We would like each feature of each modality to combine with each others. We would learn a model $\boldsymbol{W}$ (here linear), i.e. $\boldsymbol{c} = \boldsymbol{W}[[\boldsymbol{x} \otimes \boldsymbol{y}] \otimes \boldsymbol{z}]$ where $\otimes$ is the outer-product operation and [] denotes linearizing the matrix in a vector. In our experiments, our modality feature size are 120, 128 and 121. If we want $c \in \mathbb{R}^{512}$, $\boldsymbol{W}$ would have 951 millions parameters. A multimodal compact bilinear pooling model (Fukui et al., 2016) can be learned by relying on the Count Sketch projection function (Charikar et al., 2002) to project the outer product to a lower dimensional space, which reduces the number of parameters in $\boldsymbol{W}$.

4 Results

We report our preliminary results in term of the concordance correlation coefficient metric.

Model	Mean CCC
Monomodal feature extraction	
Text - CNN	0.165
Audio - OpenSmile	0.150
Video - 3DCNN	0.186
Contextual monomodal	
Text	0.220
Audio	0.223
Video	0.227
Contextual multimodal	
T + A + V	0.274
T + A + V + CBP	0.301

References

Moses Charikar, Kevin Chen, and Martin Farach-Colton. 2002. Finding frequent items in data streams. In *International Colloquium on Automata, Languages, and Programming*, pages 693–703. Springer.

Florian Eyben, Martin Wöllmer, and Björn Schuller. 2010. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1459–1462. ACM.

Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. 2016. Multimodal compact bilinear pooling for visual question answering and visual grounding. *arXiv preprint arXiv:1606.01847*.

Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 2013. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Ankur P Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933*.

Soujanya Poria, Erik Cambria, Devamanyu Hazarika, Navonil Majumder, Amir Zadeh, and Louis-Philippe Morency. 2017. Context-dependent sentiment analysis in user-generated videos. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 873–883.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010.