From speech to facial activity: towards cross-modal sequence-to-sequence attention networks

Lukas Stappen, Vincent Karas, Nicholas Cummins, Fabien Ringeval, Klaus Scherer, Bjorn Schuller

Angaben zur Veröffentlichung / Publication details:

Stappen, Lukas, Vincent Karas, Nicholas Cummins, Fabien Ringeval, Klaus Scherer, and Bjorn Schuller. 2019. "From speech to facial activity: towards cross-modal sequence-to-sequence attention networks." In *21st International Workshop on Multimedia Signal Processing (MMSP), 27-29 September 2019, Kuala Lumpur, Malaysia*, 1–6. Piscataway, NJ: IEEE. https://doi.org/10.1109/mmsp.2019.8901779.

From Speech to Facial Activity: Towards Cross-modal Sequence-to-Sequence Attention Networks

Lukas Stappen[1], Vincent Karas[1], Nicholas Cummins[1], Fabien Ringeval[2], Klaus Scherer[3], Björn Schuller[1,4]

***Abstract*—Multimodal data sources offer the possibility to capture and model interactions between modalities, leading to an improved understanding of underlying relationships. In this regard, the work presented in this paper explores the relationship between facial muscle movements and speech signals. Specifically, we explore the efficacy of different sequence-to-sequence neural network architectures for the task of predicting *Facial Action Coding System Action Units* (AUs) from one of two acoustic feature representations extracted from speech signals, namely the *extended Geneva Minimalistic Acoustic Parameter Set* (eGeMAPs) or the *Interspeech Computational Paralinguistics Challenge features set* (ComParE). Furthermore, these architectures were enhanced by two different attention mechanisms (intra- and inter-attention) and various state-of-the-art network settings to improve prediction performance. Results indicate that a sequence-to-sequence model with inter-attention can achieve on average an Unweighted Average Recall (UAR) of 65.9 % for AU onset, 67.8 % for AU apex (both eGeMAPs), 79.7 % for AU offset and 65.3 % for AU occurrence (both ComParE) detection over all AUs.**

***Index Terms*—attention networks, facial action units, sequence to sequence, paralingustics**

I. Introduction

A challenge when processing multimodal data gathered from realistic settings is the varying quality of signal information over time [1]; i. e., a signal of interest may be of low quality, or even go missing, at specific time points. For example, in the context of emotion recognition from speech and facial expressions, the face might be partially occluded shaded or positioned at varying orientation, and speech might be noisy or simply missing when the person is silent [2]. However, if the relationships between modalities are known, a cross-modal algorithm could be used to estimate the information that the missing channel should probably contain.

[1] Lukas Stappen, Vincent Karas, Nicholas Cummins, Björn Schuller are with the ZD.B Chair of Embedded Intelligence for Health Care and Wellbeing, University of Augsburg, Germany. {lukas.stappen, vincent.karas, nicholas.cummins, bjoern.schuller}@informatik.uni-augsburg.de

[2] Fabien Ringeval is with Université Grenoble Alpes, CNRS, Grenoble INP, LIG, Grenoble, France. {fabien.ringeval}@imag.fr

[3] Klaus Scherer is with the Université de Genève, Swiss Center for Affective Sciences, Switzerland {klaus.scherer}@unige.ch

[4] Björn Schuller is also with GLAM – the Group on Language, Audio & Music, Imperial College London, UK.

The work presented in this paper focuses on predicting *Facial Action Coding System Action Units* (FACS AUs) [3] from speech signals. This task is essentially a sequence-to-sequence (seq2seq) problem, which can be approached by seq2seq models, i. e., by stacked or encoder-decoder architectures [4].In this paradigm, the encoder creates an abstract representation of the input. This representation is then converted into an output sequence, which can be different modality to the encoded signal, by the decoder. In an encoder-decoder model, attention mechanisms are now commonly used to assign weights to the steps of a sequence and help align both the encoding and decoding [5]. Further, this assignment also helps overcome memory bottlenecks associated with encoding longer sequences [6]. Attention-based models have successfully been applied in machine translation [6], [7], speech recognition [8], and emotion detection [9].

An initial attempt to predict *Action Units* (AU) within the Facial Action Coding System from the voice is presented in [2]. In this work, the authors used two acoustic feature sets in combination with either a *Support Vector Machine* (SVM) or a stacked *Long Short-term Memory* (LSTM) – *Recurrent Neural Network* (RNN) classifier [10], for estimating different AUs (sub-)classes, such as onset, apex, offset and occurrence. Other approaches in the literature include [11], [12], both of which use Bayesian networks to learn the relationship between specific AUs and phonemes.

In this paper, we follow up on [2] by utilising stacked seq2seq models and an encoder-decoder architecture with different attention modules for the same speech to AU cross-modal prediction. We first, on a single well-suited AU (AU17 chin raiser), evaluate the benefit of including different attention mechanisms. In particular, we explore one type of local attention, herein referred to as intra-attention (IntraAtt), and one type of inter- (or contextual) attention (InterAtt). These approaches are compared to a baseline of a stacked seq2seq model and an encoder-decoder model without attention. In a second series of experiments, we use the best-performing architecture from our first step to run extensive experiments on all AUs and compare with results in [2].

The InterAtt-based model, when compared to the baseline and previous studies, leads to an absolute improvement in UAR. It achieves, on average over all AUs, competitive results on onset (65.9 %, +3.7 %) and apex (67.8 %, +2.8 %)

(both eGEMAPS) as well as offset (79.7 %, +19.4 %) and occurrence (65.3 %, +2.4 %) (both COMPARE).

The rest of this paper is organised as follows: In section II, the seq2seq architectures are presented in detail. Section III describes the dataset and the features extracted from it. Our experiments are the subject of section IV. Section V reports and discusses the results. Section VI concludes the paper.

II. NETWORK ARCHITECTURES

In this section, the basic structures for understanding the encoder-decoder (Enc-Dec) (cf. Section II-A) approach, different attention mechanisms utilised for Enc-Dec and seq2seq architectures (cf. Section II-B), and some of the more advanced settings (cf. Section II-C) are explained.

A. Encoder-Decoder architectures

Our network architecture is mostly based on the seq2seq architecture with attention proposed for machine translation in [6]; it takes all inputs of one sequence as the input and passes it through the Enc-Dec structure (cf. Figure 1). An encoder g_e (with parameters H_e) transforms the input sequence $x^{(i)}$, with the maximum length of T_e time steps, into a higher-level representation h. This representation is then fed into a decoder layer g_d, with a maximum length of T_d time steps. The representation is incorporated with the previous states of the decoder s_{j-1} and optionally with the previously predicted target $\hat{y}_{j-1}$ to predict $\hat{y}_j$ for each time step j of the output sequence. This process is repeated for all input-output sequences, such that the entire data set can be represented as $D = \left\{\left(x^{(i)}, y^{(i)}\right), i = 1, ..., N\right\}$, where N is the total number of data points and $x^{(i)}$ a sequence of feature vectors with a sequence of corresponding labels $y^{(i)}$:

$$h_t^{(i)} = g_e\left(x_t^{(i)}\right), i \in \{1, ..., N\}, t \in \{1, ..., T_e\} \tag{1}$$

for the encoder, and:

$$s_j^{(i)} = g_d\left(h_j^{(i)}\right), i \in \{1, ..., N\}, j \in \{1, ..., T_d\} \tag{2}$$

for the decoder, while both can be any layer type, e. g., a $\overrightarrow{LSTM}$ layer and trained with any gradient descent optimiser.

In our case, the input sequence are audio features of d dimensions and the targets y_i are a sequence of one AU subclass per model.

B. Attention components

To improve prediction quality in seq2seq architectures, various attention mechanisms can be used. Common are IntraAtt that encodes a sequence step into a sequence step with respect to the surrounding steps and InterAtt to obtain a more meaningful high-level context vector for decoding by emphasising important hidden state vectors of the encoded input sequence.

Intra-Attention: The IntraAtt module captures the similarity of any time step (t) with respect to neighbouring steps (t') in a seq2seq layer. It is based on additive attention as in [13] incooperating the idea of an attention matrix A [14] but limiting the context to the steps within a local, non-parametric window w, similar to the local attention mechanism. It transforms the hidden state representation h_t of the encoder into a positional hidden state g (Note that we have dropped the bias terms for clarity; capital letter variables represent trainable parameters):

$$g_{t,t'} = tanh(W_t h_t + W_x h_{t'}) \tag{3}$$

$$\alpha_{t,t'} = \sigma((W_a g_{t,t'}) * L), \tag{4}$$

where W_t, W_x and W_a are independently learnt weight matrices, respectively, corresponding to the hidden states h_t, $h_{t'}$, $g_{t,t'}$ element-wise multiplied with L, a matrix of the shape [batch size, time steps, time steps] that contains binary values of the chosen window size w over all time steps, while w either considers the $w-1$ past steps or the past and previous $w/2$ steps, and $\sigma()$ is an element-wise sigmoid function. This step is followed by the usual normalisation procedure of the weighted summation to receive the hidden state representation of the current token.

Inter-Attention: In contrast, InterAtt is specifically designed for an encoder-decoder architecture (cf. Figure 1) to incooperate the encoder hidden states, the decoder hidden states and optionally the prediction of the previous time step similar to [6] using an additive scoring attention function:

$$e_{j,t} = V_a * tanh(W_a s_{j-1} + U_a h_t), \tag{5}$$

where this time the previous internal hidden state of the cell state s_{j-1} is utilised. Next, a softmax operation normalises the attention probabilities:

$$\alpha_{j,t} = \frac{exp(e_{j,t})}{\sum_{k=1}^{T} exp(e_{j,k})}, \tag{6}$$

and a context vector c is computed which is the weighted sum of the encoded sequence and the attention probabilities:

$$c_j = \sum_{t=1}^{T_e} \alpha_{j,t} h_t. \tag{7}$$

Decoder: For any audio feature input at position t in the sequence, the decoder uses the encoded sequence, the internal hidden state of the decoder cell s_{j-1} and optionally the previously predicted facial action unit $\hat{y}_{j-1}$, so that the proposed hidden state is:

$$\hat{s_j} = tanh(W_p \hat{y}_{j-1} + U_p[r_j s_{j-1}] + C_p c_j), \tag{8}$$

where, W_p, U_p and C_p are trainable weight matrices and r_j is from the updated reset gate to compute the new hidden state:

$$s_j = (1 - z_j) s_{j-1} + z_t \hat{s}_j, \tag{9}$$

where z_t is the result of the updated update gate, so that the final prediction is:

$$\hat{y}_j = \sigma(W_o \hat{y}_{j-1} + U_o s_j + C_o c_j). \tag{10}$$

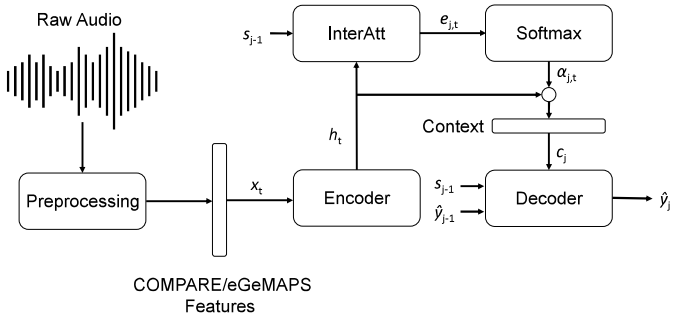

Fig. 1. Pipeline from raw audio files to prediction utilising the Encoder-Decoder architecture with an inter-attention module. The raw audio files are segmented and selected in the preprocessing followed by an extraction of the eGeMAPs and ComPare feature sets. Then, a sequence is encoded, a context vector is calculated and, finally, the decoded sequence is used for prediction.

C. Further advanced settings

Initial experiments showed that applying **Batch Normalisation** [15] after every layer adds an advantage especially in terms of a smoother training behaviour (loss curve), reduced training time by a factor of 4, and provides a more stable training outcome.

We use **Early Stopping** [16] to automatically stop the learning process. This has the advantage of helping to alleviate overleaf and identify the optimal number of epochs to train the model. Because loss and metrics can be spiky during the training process due to strongly imbalanced classes and the model often recovering from harmful epochs, we use additional parameters to control early stopping: The $patience$ parameter describes the number of epochs the model does not improve over before stopping. The $prev\_rounds$ parameter controls the number of epochs the model trains before starting to activate the patience. It also controls the number of last epochs the model looks back to calculate the average result of these rounds to compare this value to the result of the actual epoch within the patience mechanism. The summation of $patience$ and $prev\_rounds$ is also equal to the minimum number of epochs. Finally, the parameter $last\_top\_results$ describes the number of epochs for the look back to select the best performing epoch.

In order to reduce overfitting in a classification setting due to imbalanced classes, the data sets of the respective classes are weighted differently to adjust for the loss of a datapoint belonging to a specific class. Thi is acheived using a simple **class weight**:

$$W_k = N/(K * \sum_{n=1}^{N} N_k), \tag{11}$$

where N is the number of data points, $k \in \{1, ..., K\}$ a specific class, N_k the number of occurrences belonging to k, and W_k is the corresponding weight.

III. Data and Prepossessing

A. The Geneva Multimodal Emotion Portrayals corpus

The *Geneva Multimodal Emotion Portrayals* data set consists of 7 000 audio-video emotional portrayals of 18 emotions played by five female and five male French-speaking actors [17]. The recordings were made during continuous vocal interactions between actors and directors, which were then manually segmented. The recordings were made with three cameras with integrated microphones at a frame rate of 25 Hz, with one camera pointing frontally at the face. Additionally, one 44.1 kHz microphone was positioned at the left ear of the actor. A subset of 158 portrayals was designed for this particular speech-to-vison task by the authors of [2]. This subset consists of instances with a high agreement rate between human annotated labels – more than 5 % occurrence for all labels – and an equal amount of recordings from each speaker.

B. Preprocessing

The synchronisation between the audio and the video signal was performed manually to guarantee an optimal alignment of audio-based features and video-based labels. The audio data were normalised to 0 dB peak amplitude (loudness). Speaker-wise normalisation was also performed on the extracted low-level feature descriptors (cf. Section III-C) considering the activity of the facial action units labels to reduce the variance of features over the different actors.

C. Audio feature extraction

We extracted acoustic low-level descriptors (LLDs) using openSMILE [18]. The 130 dimensional ComPare features are considered as the de facto LLD standard feature set in computational paralinguistics [18]. This feature set consists of 4 prosodic and energy-related, 55 spectral and 6 voicing-related LLDs as well as their first order derivatives. The features were extracted using 60 ms frames with a Gaussian window function for voicing-related LLDs and a 25-ms frames using a Hamming window function for all others, overlapped and sampled at 100 Hz. A symmetric moving average window of 3 frames length and first order delta regression coefficients with a context window size of 2 frames for all LLD's were used for smoothing.

Similar to the ComPare feature set, we used the same tool with the same settings to extract the eGeMAPs features resulting in a 28 dimensional, handcrafted LLD feature set, sampled at 100 Hz using overlapping windows and the same symmetric moving average window for smoothing. For detailed descriptions the interested reader is referred to [2].

D. Labels: Facial Action Coding System (FACS)

The Facial Action Coding System is the common standard in emotion research for the systematic coding of different facial expressions based on the movements of 28 main individual facial muscles [3]. Typically, the intensity of an AU is indicated either on a scale from 0 to 5 or A to E, where the intensity increases with a higher number or letter.

In this work, to ensure comparability, we used the same labels as generated in [2]. The authors used two independent certified FACS encoders to accurately hand-annotate the eight most relevant AUs: Inner Brow Raiser (AU1), Outer Brow Raiser (AU2), Brow Lowerer (AU4), Cheek Raiser (AU6), Lid Tightener (AU7), Nose Wrinkler (AU10), Lip Corner Puller (AU12), Chin Raiser (AU17). Further, the originally continuous AUs were transformed into binary active and inactive labels for classification, while each Action Unit was specified into four classes:

1) Onset: The frame at which the first increasing change in appearance was observed (start of muscles activation). Initial periods with little change are also added if a granular increase occurs and no plateau or direct apex is reached.
2) Apex: Is defined as the peak (maximum of muscles activation) after a rise within the onset and before the subsequent decrease in intensity.
3) Offset: If the intensity decreases over at least two consecutive frames (release of muscles), this initiates the end of the apex and the beginning of an offset, which ends with the complete drop of the intensity to zero or with a new onset phase.
4) Occurrence: The duration of all appearances of onset, apex and offset in the sequence (activation of muscles activation).

IV. Experiments

This section outlines our key experiments as well as experimental and training settings.

A. Experiments

Two extensive series of experiments were carried out. In the first, a broad combination of the attention components, presented in Section II-B, is evaluated on one AU. This experiment provides a general overview of the performance of the components and the combination of these components for this task. First, preliminary experiments have shown that AU17 (chin raiser) provides the most stable results, and is the preferred class for this type of comparison.

In general, we evaluate two different seq2seq network architectures, two stacked bidirectional LSTMs (Stacked) plus a softmax prediction layer and the introduced Section II-A Enc-Dec architecture. We first evaluate both of them completely without attention (No-Att) to receive an initial baseline of the degree to which the advanced settings improve the results. Based on the stacked architecture four experiments are conducted: two experiments of the IntraAtt component with a bidirectional attention window bi of the sizes 5 and 15 are incorporated between the stacked layers, as well as two experiments with two IntraAtt components, each on the encoding and decoding sequence, whereby one is again bidirectional and one attends only the previous sequence steps pa.

This comparison is followed by one experiment with pure InterAtt between the encoder and decoder (Enc-InterAtt-Dec) and one in combination with IntraAtt (Enc-InterAtt-Dec-IntraAtt). Since the sequences are not particularly long, no distinction is made between training and inference time. Therefore, the real y_{t-1} is not used for training as is often the case, but only the predicted $\hat{y}_{t-1}$ for both training and inference.

In the second series of experiments, all AUs are evaluated by the best architecture and compared to the result of a 3-layer stacked LSTM architecture of [2]. One AU consists of the four sub-classes (occurrence, onset, offset, apex), and for each sub-class ten (the number of partitions) models per class were trained and evaluated. As a result, a total of 320 models (8 component combinations * 4 AU17 sub-classes * 10 partitions) for the first and 640 models (8 AUs * 4 AU sub-classes * 2 feature sets (eGeMAPs, ComParE) * 10 folds) for the second series of experiments were trained.

B. Experimental settings

Since this is a relatively small data set and in order to maintain comparability with [2], the speaker independent leave-one-speaker-out (LOSO) cross-validation method was chosen for partitioning.

We have used the Unweighted Average Recall (UAR) as the metric to evaluate the results of the models. Alongside the comparability to [2] and widespread use in audio tasks, e. g., in the INTERSPEECH ComParE challenges, this also has the advantage that strongly imbalanced classes do not falsify the result (binary chance level is $50\,\%$). The UAR is calculated on the real length of the sequence (T_e = T_d), although the maximum T is fixed for training to a larger number of time steps. Since we have multiple partitions per class, the displayed result of one class is the mean UAR of all partitions.

All experiments were implemented with Keras custom layers.

C. Training settings

Due to the simple cross-validation with LOSO partitioning, a tuning of the respective networks is only appropriate to a limited extent. For this reason, we have not performed a comprehensive hyperparameter tuning but applied generic and frequently used techniques in seq2seq learning.

The maximum number of input and output time steps is always 100, where T_e^i is equal to T_d^i. Shorter sequences were brought to their maximum length by zero padding. Since T^i sometimes differs considerably from sequence to sequence, we used **masking** on the padded time steps to handle variable length inputs and skipped updating weights for these time steps in all downstream layers including all attention components.

The individual networks converge very differently depending on the AU, the partitions, the unequal class distribution and the initialisation. This variability forces the use of early stopping to prevent overfitting. In this regard, we set the maximum number of epochs to 100 with $patience$ = 5. The result is compared with the average value, calculated using

TABLE I
PERFORMANCE OF ALL MECHANISMS EXEMPLIFIED BY AU17 WITH COMPARE FEATURES ON ONSET, APEX, OFFSET, AND OCCURRENCE (OCCUR.). ALL RESULTS ARE GIVEN IN UNWEIGHTED AVERAGE RECALL IN %.

Architectures		AU17			
Name	**Window**	**Onset**	**Apex**	**Offset**	**Occur.**
Stacked-NoAtt	-	75.8	73.1	80.2	75.4
Stacked-IntraAtt	bi = 5	62.9	62.1	66.8	61.9
Stacked-IntraAtt	bi = 15	79.1	70.3	74.9	70.2
2xStacked-IntraAtt	pa = 15	76.9	76.3	80.7	65.7
2xStacked-IntraAtt	bi = 15	75.7	72.8	75.4	**76.6**
Enc-Dec-NoAtt	-	77.0	76.8	83.3	73.8
Enc-InterAtt-Dec	-	**80.5**	**77.6**	**89.2**	73.3
Enc-InterAtt-Dec-IntraAtt	pa = 15	76.2	76.4	83.9	71.1

$prev\_rounds$ = 15. Further, we store a running total of the best result, setting $last\_top\_results$ = 10.

For the training of the networks, we used only bidirectional LSTM(s) for both stacking and encoder layer. The extracted eGEMAPS feature set dimension is relatively small ($d = 28$). We, therefore, limited the number of neurons in the hidden layers to 20 to avoid very fast overfitting and, thus, fluctuating training results. For a fair comparison, we use the same value for the COMPARE feature set.

Preliminary experiments showed that merging the hidden states using sum and average of the bidirectional layers additionally stabilised the training, while $concat$ had a destabilising effect. We chose sum for all our experiments. For the same reasons, we applied an L2 kernel regulariser of 1e-1, an L1 bias regulariser of 1e-4, and an attention regulariser weight of 1e-4 for IntraAtt, and 0.2 dropout to all layers.

Finally, we used binary cross-entropy as the objective function and optimised the training process with an Adam optimiser, a learning rate of 1e-4, and a batch size of 32.

V. RESULTS AND DISCUSSION

A. Results

First, the results of the first series of experiments (AU17, chin raiser) are discussed, followed by the comparison of all AUs on the two feature sets COMPARE and eGEMAPS of the second series.

The first series of experiments (Table I) shows that the IntraAtt configurations (Stacked-IntraAttbi bi=5, StackedIntraAtt bi=15), yield better performance if the attention window is wider. The double stacked BiLSTM-IntraAtt improves results for apex, offset, and occurrence (Stacked-IntraAtt vs 2xStacked-IntraAtt, both bi=15). If in this configuration only the context of past steps is considered (pa=15), onset (76.9 %), apex (76.3 %), and offset (80.7 %) increase further. However, the overall best result for occurrence, 76.6 %, was achieved using a bidirectional attention window (2xStacked-IntraAtt, bi=15).

In general, it is noteable that model configurations that achieve good results in the sub-classes (onset, apex, and offset) do not necessarily do so in the overall class occurrence. We speculate that this is due to the more complex structure in the prediction of occurrence, which changes much more frequently (binary) and is, therefore, less constant than the sub-classes. A similar pattern can be seen in the Stacked vs Enc-Dec architectures. Furthermore, it seems more difficult to the attention mechanisms to learn these transitions from the neighbouring time steps, in which case the Stacked IntraAtt has only slight advantages over the non-attention version.

The Enc-Dec architecture, which uses exclusively InterAtt, results in the best performance for onset (80.5 %), apex (77.6 %) and offset (89.2 %) in this series of experiments. Due to the high computation effort, only this architecture is evaluated in the second part.

The results of the second series of experiments (Table II) demonstrate improvements across almost all AUs compared to the architecture of [2]. A reason for this seems to be both the architecture with InterAtt and the state-of-the-art network settings, namely, batch normalisation, early stopping, class weights and masking (see also baseline results without attention from the first experiment series, which also show improved results).

The average results over all AUs utilising the eGEMAPS feature set achieved an improved UAR of onset (65.9 %, +3.7 %), apex (67.8 %, +2.8 %) and offset (77.8 %, +16.4 %). Similarly good results for the sub-classes were achieved by the models using the COMPARE features with onset (63.9 % +1.9 %), apex (66.2 %, +3.1 %) and offset (79.7 %, +19.4 %). In particular, the offset class stands out across all AUs and feature sets. In contrast, the results of the overall class occurrence are mixed and in line with the results of the first series of experiments, with an absolute average improvement of +2.4 % for COMPARE and a deterioration of -1.2 % for eGEMAPS compared to [2]. Looking at the individual AUs in detail shows a strong improvement in the prediction of AU7 (Lid Tightener), AU12 (Lip Corner Puller) and AU17 (Chin Raiser) across almost all sub-classes and for both feature sets. For the eGEMAPS feature set, the occurrence of all brow related AUs (1, 2, 4) and the apex of AU1 and AU2 deteriorated slightly, while onset and offset improved with these AUs having already achieved far above-average results in [2].

B. Limitations of the Proposed Approach

We observed during the system development and first evaluations that the training was not always stable and could

TABLE II

PERFORMANCE OF THE ENCODER-DECODER MODEL WITH INTER-ATTENTION ON ALL AU CLASSES AND THE TWO FEATURE SETS INCLUDING A ONE TO ONE ABSOLUTE COMPARISON TO THE STACKED THREE LAYER LSTM IN [2] IN BRACKETS. ALL RESULTS ARE GIVEN IN UNWEIGHTED AVERAGE RECALL IN %. THE AUS ARE AS DEFINED IN SECTION III-D: INNER BROW RAISER (AU1), OUTER BROW RAISER (AU2), BROW LOWERER (AU4), CHEEK RAISER (AU6), LID TIGHTENER (AU7), NOSE WRINKLER (AU10), LIP CORNER PULLER (AU12), AND CHIN RAISER (AU17).

	COMPARE				eGEMAPS			
AU	**Onset**	**Apex**	**Offset**	**Occurrence**	**Onset**	**Apex**	**Offset**	**Occurrence**
1	64.2 (+2.3)	68.1 (+3.6)	80.0 (+20.5)	69.7 (+2.6)	69.7 (+7.6)	67.1 (-0.8)	81.0 (+19.8)	63.8 (-3.8)
2	60.9 (-3.4)	73.3 (+5.9)	70.9 (+8.6)	68.1 (-1.2)	64.8 (+0.2)	63.2 (-7.8)	66.5 (+3.5)	65.2 (-5.7)
4	68.9 (+7.5)	65.8 (-0.4)	75.6 (+12.3)	67.6 (+3.0)	64.7 (+2.8)	72.9 (+6.2)	73.3 (+10.1)	61.5 (-6.4)
6	65.1 (+1.1)	64.2 (-1.0)	90.3 (+26.9)	64.0 (+0.2)	65.7 (+1.4)	72.9 (+5.2)	81.4 (+18.2)	62.2 (-1.0)
7	62.0 (-0.7)	65.3 (+8.7)	85.6 (+27.3)	60.3 (+5.9)	62.8 (+0.0)	65.7 (+6.0)	69.7 (+8.2)	57.8 (+5.1)
10	55.7 (-6.1)	56.2 (-4.3)	73.8 (+17.4)	60.1 (-0.3)	59.5 (-2.5)	62.6 (+1.5)	91.1 (+33.1)	57.8 (-2.5)
12	53.6 (-5.7)	59.5 (-0.4)	72.0 (+15.3)	59.0 (+0.4)	60.5 (+2.5)	67.2 (+6.3)	79.6 (+21.8)	62.0 (+3.3)
17	80.5 (+20)	77.6 (+12.8)	89.2 (+26.9)	73.3 (+8.7)	79.6 (+18.0)	70.6 (+5.6)	79.8 (+16.7)	67.1 (+1.3)
Avg.	63.9 (+1.9)	66.2 (+3.1)	**79.7** (+19.4)	**65.3** (+2.4)	**65.9** (+3.7)	**67.8** (+2.8)	77.8 (+16.4)	62.2 (-1.2)

produce results of varying quality. The LOSO and the small data set limited the amount of hyperparameter optimisation we could realistically achieve.

VI. CONCLUSION AND FUTURE WORK

In this paper, we demonstrated that recurrent encoder-decoder architectures with attention could be used to predict FAUs from audio. We used a well-suited AU (AU17, chin raiser) to compare local and contextual attention, herein referred to as intra-attention (IntraAtt) and inter-attention (InterAtt) respectively, to a baseline model without attention. The architecture that exclusively used InterAtt performed best in terms of UAR for onset (80.5 %), apex (77.6 %) and offset (89.2 %). We then used this architecture to predict all AUs and compared the results to those presented in [2]. There was a significant improvement of UAR utilising the eGEMAPS and COMPARE feature set for the sub-classes onset, apex, and offset, while the results for the general class occurrence were mixed.

Our future work will involve applying the approach presented in this paper to data recorded in-the-wild [19]. Due to the considerably larger data set, this work could also be extended by implementing more complex attention-based architectures i. e., tensor2tensor [20]. Moreover, we intend to predict FACS regression scores instead of performing binary classification.

ACKNOWLEDGMENT

This research has received funding from the EUs Horizon 2020 Programme under grant agreement No. 826506 (sustAGE) and BMW Group research.

REFERENCES

[1] T. Baltrušaitis, C. Ahuja, and L.-P. Morency, "Multimodal machine learning: A survey and taxonomy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 2, pp. 423–443, 2019.

[2] F. Ringeval, E. Marchi, M. Mehu, K. Scherer, and B. Schuller, "Face reading from speech – predicting facial action units from audio cues," in *Proc. of the Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[3] P. Ekman and W. V. Friesen, *Manual for the facial action coding system*, Consulting Psychologists Press, 1978.

[4] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in Neural Information Processing Systems 27*, pp. 3104–3112. 2014.

[5] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems 28*, pp. 577–585. 2015.

[6] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[7] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," in *Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing*, October 2014, pp. 1724–1734.

[8] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Proc. of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 4960–4964.

[9] D. Hazarika, S. Gorantla, S. Poria, and R. Zimmermann, "Self-attentive feature-level fusion for multimodal emotion detection," in *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, April 2018, pp. 196–201.

[10] F. Weninger, J. Bergmann, and B. Schuller, "Introducing currennt: The munich open-source cuda recurrent neural network toolkit," *The Journal of Machine Learning Research*, vol. 16, no. 1, pp. 547–551, 2015.

[11] Z. Meng, S. Han, and Y. Tong, "Listen to your face: Inferring facial action units from audio channel," *IEEE Transactions on Affective Computing*, 2018.

[12] Z. Meng, S. Han, P. Liu, and Y. Tong, "Improving speech related facial action unit recognition by audiovisual information fusion," *IEEE Transactions on Cybernetics*, vol. 49, pp. 3293–3306, 2018.

[13] J. Cheng, L. Dong, and M. Lapata, "Long short-term memory-networks for machine reading," *arXiv preprint arXiv:1601.06733*, 2016.

[14] G. Zheng, S. Mukherjee, X. L. Dong, and F. Li, "Opentag: Open attribute value extraction from product profiles," in *Proc. of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 1049–1058.

[15] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[16] L. Prechelt, "Early stopping-but when?," in *Neural Networks: Tricks of the trade*, pp. 55–69. Springer, 1998.

[17] T. Bänziger and K. R. Scherer, "Introducing the geneva multimodal emotion portrayal (gemep) corpus," *Blueprint for affective computing: A sourcebook*, pp. 271–294, 2010.

[18] F. Eyben, F. Weninger, F. Groß, and B. Schuller, "Recent Developments in openSMILE, the Munich Open-Source Multimedia Feature Extractor," in *Proc. of MM 2013*, Barcelona, Spain, 2013, pp. 835–838, ACM.

[19] J. Kossaifi, R. Walecki, Y. Panagakis, J. Shen, M. Schmitt, F. Ringeval, J. Han, V. Pandit, B. Schuller, K. Star, et al., "Sewa db: A rich database for audio-visual emotion and sentiment research in the wild," *arXiv preprint arXiv:1901.02839*, 2019.

[20] A. Vaswani, S. Bengio, E. Brevdo, F. Chollet, A. N. Gomez, S. Gouws, L. Jones, L. Kaiser, N. Kalchbrenner, N. Parmar, et al., "Tensor2tensor for neural machine translation," *arXiv preprint arXiv:1803.07416*, 2018.