AN AUTOMATED AUDIO CAPTIONING APPROACH UTILISING A RESNET-BASED ENCODER

Technical Report

Alexander Gebhard[1], *Andreas Triantafyllopoulos*[1,2], *Alice Baird*[1], *Björn Schuller*[1,2,3]

[1]EIHW – Chair of Embedded Intelligence for Healthcare and Wellbeing,
University of Augsburg, Augsburg, Germany
[2]audEERING GmbH, Gilching, Germany
[3]GLAM – Group on Language, Audio, and Music, Imperial College, London, UK

ABSTRACT

In this report, we present our submission system to TASK6 of the DCASE2021 Challenge. The main module is based on the baseline architecture for the automated audio captioning (AAC) task, which was provided by the challenge organisers. We exchange the encoder part of the baseline architecture and replace it by a Residual Neural Network (ResNet)-18 encoder adapted to the AAC task. Results from our proposed architecture have shown an average increase of 35.7% over the baseline system, reaching a BLEU_1 score of 0.449 on the development set, demonstrating the effectiveness of the proposed encoder for this task.

Index Terms— automated audio captioning, DCASE Challenge

1. INTRODUCTION

Automatic captioning is well established in the video domain, but is still relatively new in the field of intelligent audio research [1]. Captioning systems in the video domain have been proven useful for several applications, e. g., accessibility [2] or vision-based security systems [3]. Automated audio captioning (AAC) has recently begun to gain traction in the computer audition community as well [4], particularly since the 2020 DCASE challenge where the task was first introduced [5, 6]. In a nutshell, AAC is the process of automatically generating textual descriptions of an audio scene [7].

In this contribution, we present an encoder-decoder architecture based on the DCASE Task6 baseline system. The main adaption for our proposed architecture is the substitution of the Recurrent Neural Network (RNN)-based encoder with a Residual Neural Network (ResNet)-18 Convolutional Neural Network (CNN) one. The main motivation to include ResNet-18 for the task of AAC in the architecture is its strong performance in video-captioning [8] and the robust results obtained in recent audio-domain studies [9, 10], as well as multimodal approaches [11].

This report is organised as follows; firstly, in Section 2 and Section 3, we detail the Clotho-v2.1 data and features used for our experiments. We then give full detail of our proposed architecture for the task of AAC in Section 4. Subsequently, we present our experiments and results on this task in Section 5. Finally, we conclude our findings, and suggest future work avenues for the proposed system in Section 6.

2. DATASET

The dataset used in this work is the official DCASE2021 AAC dataset, namely CLOTHO-V2.1, which is an extension of the original CLOTHO-V1 [12]. The dataset consists of audio samples with 15 to 30 seconds duration, each audio sample having five captions of eight to 20 words in length. In total, there are 6 974 audio samples with 34 870 captions in the full CLOTHO dataset, considering both versions. CLOTHO-V2.1 is divided into four splits: development, validation, evaluation, and testing. There are audio samples for all four splits, however, for the purposes of the challenge, the captions are withheld for the testing split. To avoid possible dependency on particular words, it is ensured that no word appears in the evaluation, validation, or testing split which is not part of the development split. In most of the cases, the words are also proportionally distributed among the splits, with 55 % in the development set and 15 % in each of the three other sets.

Due to time constraints and since the baseline system was still aligned with CLOTHO-V1, we had to exclude the validation split and could only use the development, evaluation, and testing splits for our approach. In regard to this, the development split was used as our training set and the evaluation set was utilised to measure the performance of the network and calculate the metrics, such that we could compare the results with the baseline. That is, the results are reported on the official Clotho v2.1 evaluation split, which is referred to as development-test split by the challenge organisers. Finally, the testing set of Clotho was used to predict the captions for the challenge submission.

Table 1 provides a short overview of the different naming conventions of the data splits and displays the amount of files which were available to the authors of this report in each data split.

3. FEATURES

The model is trained on audio data with a duration of 15–30 seconds. All audio files are in wav format, are re-sampled (for consistency) to 44.1 kHz, 16-bit, and converted to MONO during the loading process of the architecture. For feature extraction, we utilise the function provided by the baseline system, extracting log-Mel spectrograms. The features are extracted by using a Hanning window function with a window size of 23 ms (1 024 samples) and 50 % overlap (512 samples). From each frame, we extract $n_{mel} = 64$ Log-Mel bands. The extracted audio features are then transformed into a matrix $X = [x_1, x_2, ..., x_T]$, where T is

CLOTHO	DCASE	$\sum$
development	development-training	3 839
validation	development-validation	1 045
evaluation	development-testing	1 045
testing	evaluation	1 043

Table 1: Naming conventions of the data splits w. r. t. the AAC task as well as total ($\sum$) audio samples used for the current experiments w. r. t. each data split.

the number of frames that the input audio data is divided into and $x_t \in \mathbb{R}^{64}$ is a vector containing the log-Mel bands in frame t. These feature matrices are the input to the network. Since the duration of the audio data varies between 15–30 seconds, the features are padded with zeros at the front, such that all the input audio features sequences in a batch have the same amount of vectors. The output word sequences are padded with <EOS> tokens at the end, so all output sequences will have the same amount of words.

4. ARCHITECTURE

The goal for the proposed system is to take an audio file of 44.1 kHz sampling frequency as an input and create a caption for it (i. e., a textual description).

The proposed architecture in this report is an adaptation of the baseline system for the AAC task, which is a sequence-to-sequence architecture and consists of an encoder and a decoder. The original baseline encoder consists of three bi-directional Gated Recurrent Units (GRUs) and outputs the summary of the input sequence of features. However, our encoder does not use any RNNs at all, and is a solely CNN-based approach, since it is an adaptation of the ResNet-18 architecture proposed in [13]. The decoder part remains the same as in the baseline system, i. e., consists of one GRU and one linear layer, which outputs the probability for each of the unique words.

Our adjustments to the ResNet-based encoder are as follows: we exclude the last two layers, i. e., the final dense layer of the ResNet, as well as the preceding average pooling layer, and replace this with our own linear layer which has an input dimensionality of 1 024 and an output dimensionality of 512. Thus, it transforms the extracted features such that it can be fed to the GRU, which is the first layer of the decoder and has an input dimensionality of 512. The output of the GRU is then handed over to the classifier (a linear layer), in order to obtain the probability for each of the unique words.

The other ResNet-layers of our encoder are adopted as described in the original paper from He et al. [13]. The first convolutional layer applies a 2D convolution over the input features with 1 input channel in our case, using 64 filters with a kernel size of (7×7) and a stride of (2×2); the utilised activation function being rectified linear unit (ReLU). Afterwards Batch Normalisation (BN) and MaxPooling are applied. This is followed by the big four convolutional blocks of the ResNet-18 architecture and finally the linear layer. For the structure of the original ResNet-architecture, please have a look at the original paper of He et al. in [13]. In Figure 1, you can see an overview scheme of the proposed network.

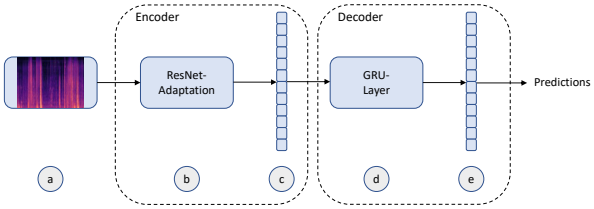

Figure 1: An overview of the proposed pipeline. (a) The log-Mel spectrogram of an audio file serves as input to the network, (b) the adapted ResNet-architecture is used for feature extraction, (c) the linear layer transforms the features for the decoder, (d) the GRU layer creates an output for each time step, and (e) the linear layer generates predictions of the unique words. The encoder comprises (b) and (c), while (d) and (e) form the decoder part.

Metrics	Baseline	Proposed System	% Increase
$BLEU_1$	.378	.449	18.8
$BLEU_2$	.119	.167	40.3
$BLEU_3$	.050	.068	36.0
$BLEU_4$	.017	.029	70.6
$ROUGE_L$	.263	.284	8.0
METEOR	.078	.097	24.4
CIDEr	.075	.098	30.6
SPICE	.028	.043	53.6
SPIDEr	.051	.071	39.2
μ	-	-	35.7

Table 2: Results on the development-test (evaluation) for the DCASE2021 AAC Dataset. Reporting all the baseline metrics provided by the challenge organisers, as well as the percentage (%) of increase and the mean (μ) increase across all metrics.

5. EXPERIMENTS

5.1. Training

The model is trained for 300 epochs with a categorical cross-entropy loss, and the best model is selected based on the training loss. We use a batch size of 16 and a learning rate of 0.0001 with the Adam optimiser [14]. We use gradient clipping, similar to Tran et al. in [15], such that the 2-norm of the gradients does not exceed the value of 1.

5.2. Results

Table 2 shows the results of our approach compared to the baseline system. As can be inferred from the table, every score metric was increased. Therefore, it appears that the ResNet encoder extracts useful features from the spectrograms which are better suited than the features extracted by the multi-layer GRU of the baseline encoder. This suggests that ResNet-based CNN encoders are better suited to this task and should be further explored as follow-up work.

When looking at the different scores obtained by this system, it is noticeable that the percentage increase was the highest for the $BLEU_4$ score. Additionally, the $BLEU_2$ and $BLEU_3$ score increased considerably more than the $BLEU_1$ norm, which is a robust

indication that our approach improves on the main task for producing sequences of words, occurring within a given window with a window size of 2 - 4 words. However, since the BLEU_1 score was also increased noticeably, the number of predicted candidate words which occur in the reference caption was boosted in general.
The least increase could be achieved for the ROUGE_L score, which is a Longest Common Subsequence (LCS) based statistics. That is, the natural sentence level structure of long cooccuring n-gram sequences was improved, but not substantially. However, the remaining two n-gram based scores METEOR and CIDEr did show an improvement, suggesting that the predicted captions were syntactically enhanced.
Moreover, the non n-gram based score SPICE which takes the semantic structure of a caption into account was increased as well, which indicates that also the meaning of the predicted captions was improved. This would mean that both syntax and semantics of the predicted captions have been boosted, something also affirmed by the increased SPIDEr score, which considers both semantics and syntax, respectively.

6. CONCLUSION

The current work investigates the use of a ResNet-based encoder for the task of AAC. The baseline fully-RNN system is thus adapted to a hybrid Convolutional Recurrent Neural Network (CRNN) approach, which shows better results. The performance of the method can be further improved through use of higher-capacity ResNet encoders, potentially pre-trained on different tasks, state-of-the-art data augmentation approaches such as SpecAugment [16], and features better suited to intelligent audio analysis tasks such as openL3 [17, 18], DeepSpectrum [19], or auDeep [20].

7. ACKNOWLEDGMENT

This project received funding from the DFG's Reinhart Koselleck project No. 442218748 (AUDI0NOMOUS), as well as the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957337, project MARVEL.

8. References

[1] B. Schuller, S. Amiriparian, G. Keren, A. Baird, M. Schmitt, and N. Cummins, "The next generation of audio intelligence: A survey-based perspective on improving audio analysis," in *Proceedings of the International Symposium on Auditory and Audiological Research*, vol. 7, 2019, pp. 101–112.

[2] M. R. Morris, A. Zolyomi, C. Yao, S. Bahram, J. P. Bigham, and S. K. Kane, "" with most of it being pictures now, i rarely use it" understanding twitter's evolving accessibility to blind users," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 2016, pp. 5506–5516.

[3] R. S. Kumawat and S. Iniyan, "A survey on public safety systems inside atm,"

[4] Y. Koizumi, R. Masumura, K. Nishida, M. Yasuda, and S. Saito, "A transformer-based audio captioning model with keyword estimation," *arXiv preprint arXiv:2007.00222*, 2020.

[5] Y. Koizumi, D. Takeuchi, Y. Ohishi, N. Harada, and K. Kashino, "The ntt dcase2020 challenge task 6 system: Automated audio captioning with keywords and sentence length estimation," *arXiv preprint arXiv:2007.00225*, 2020.

[6] K. Drossos, S. Adavanne, and T. Virtanen, "Automated audio captioning with recurrent neural networks," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, New York, U.S.A., Oct. 2017. [Online]. Available: https://arxiv.org/abs/1706.10006.

[7] ——, "Automated audio captioning with recurrent neural networks," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, IEEE, 2017, pp. 374–378.

[8] A. Kapadi, C. R. Kavimandan, C. S. Mandke, and S. Chaudhari, "Wildlife video captioning based on resnet and lstm," *Modern Approaches in Machine Learning and Cognitive Science: A Walkthrough: Latest Trends in AI, Volume 2*, p. 353,

[9] C. Bergler, H. Schröter, R. X. Cheng, V. Barth, M. Weber, E. Nöth, H. Hofer, and A. Maier, "Orca-spot: An automatic killer whale sound detection toolkit using deep learning," *Scientific reports*, vol. 9, no. 1, pp. 1–17, 2019.

[10] M. Gerczuk, S. Amiriparian, S. Ottl, and B. Schuller, "Emonet: A transfer learning framework for multi-corpus speech emotion recognition," *arXiv preprint arXiv:2103.08310*, 2021.

[11] O. Köpüklü, M. Taseska, and G. Rigoll, "How to design a three-stage architecture for audio-visual active speaker detection in the wild," *arXiv preprint arXiv:2106.03932*, 2021.

[12] K. Drossos, S. Lipping, and T. Virtanen, "Clotho: An audio captioning dataset," in *45th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Barcelona, Spain, May 2020. [Online]. Available: https://arxiv.org/abs/1910.09387.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[14] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[15] A. Tran, K. Drossos, and T. Virtanen, "Wavetransformer: An architecture for audio captioning based on learning temporal and time-frequency information," in *29th European Signal Processing Conference (EUSIPCO)*, 2021.

[16] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *Proc. Interspeech 2019*, pp. 2613–2617, 2019.

[17] R. Arandjelovic and A. Zisserman, "Look, listen and learn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 609–617.

[18] J. Cramer, H.-H. Wu, J. Salamon, and J. P. Bello, "Look, listen, and learn more: Design choices for deep audio embeddings," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 3852–3856.

[19] S. Amiriparian, M. Gerczuk, S. Ottl, N. Cummins, M. Freitag, S. Pugachevskiy, A. Baird, and B. Schuller, "Snore sound classification using image-based deep spectrum features," en, in *Interspeech 2017*, ISCA, Aug. 2017, pp. 3512–3516.

[20] M. Freitag, S. Amiriparian, S. Pugachevskiy, N. Cummins, and B. Schuller, "Audeep: Unsupervised learning of representations from audio with deep recurrent neural networks," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 6340–6344, 2017.