TASK 6 DCASE 2020: LISTEN CAREFULLY AND TELL: AN AUDIO CAPTIONING SYSTEM BASED ON RESIDUAL LEARNING AND GAMMATONE AUDIO REPRESENTATION

Technical Report

Javier Naranjo-Alcazar[1 *], *Sergi Perez-Castanos*[*], *Pedro Zuccarello*[1], *Maximo Cobos*[1]

[1] Universitat de València, Burjassot, Spain, {janal2@alumni.uv.es}, {maximo.cobos, pedro.zuccarello}@uv.es

ABSTRACT

Automated audio captioning is machine listening task whose goal is to describe an audio using free text. An automated audio captioning system has to be implemented as it accepts an audio as input and outputs a textual description, that is, the caption of the signal. This task can be useful in many applications such as automatic content description or machine-to-machine interaction. In this technical report, a automatic audio captioning based on residual learning on the encoder phase is proposed. The encoder phase implemented via different Residual Networks configurations. The decoder phase (create the caption) is run using recurrent layers plus attention mechanism. The audio representation chosen has been Gammatone. Results show that the framework proposed in this work surpass the baseline system improving all metrics.

Index Terms— Audio captioning, Residual learning, Attention, Encoder-Decoder, Gammatone

1. INTRODUCTION

Audio captioning is a novel machine listening task that was first presented in [1]. Audio captioning can be understood as a intermodal translation. Its goal is to create an autonomous and smart description on an audio signal. The state-of-the-art solution employs an encoder-decoder architecture [2]. The encoder block embeds the audio representation (i.e. log Mel-Spectrogram representation) into a lower dimensionality feature map while the decoder creates a sequence of words from that new representation, that is, the smart caption. This caption must be as close as possible as human performance which means that captions must be structured according to the language in which it is being described. This task differs from other classic machine listening tasks such as audio tagging or sound event detection. Audio captioning is not intended to assign labels to audio or to calculate onset and offset times.

First approximations to autonomous captioning where done in image domain [3, 4] followed by autonomous video captioning [5, 6]. In [3] this problem was addressed for the first time. Captioning, whether in the image or audio domain, can be interpreted as an artificial intelligence problem that connects two fields. In the case of the image domain, computer vision and language processing techniques must be merge. In audio domain, the representation of the audio, the processing of this representation (similar to computer vision techniques) and the language processing must be taken into account. The intuition on which this work is based is in the work done in sentence translation where a sentence must be translated from an initial language to a target language. The emergence of recurrent neural networks (RNNs) led to the creation of simpler solutions (without reordering or individual processing of each word) and maintained state-of-the-art performance [7, 8]. In this type of problem, an RNN encodes the input sentence into a fixed-size codification and a RNN decoder generates the translated sentence in the target language. As image captioning takes as input a fixed size image, in [3] it is decided to replace the encoder block by a convolutional neural network (CNNs). This type of network has shown good results in extracting descriptive information about the images. Thus, a network of this nature will be used to encode the images for the decoder, which in this case is a recurrent network.

Autonomous captioning frameworks (regardless the data domain, e.g. image or audio) can be divided in two blocks as mentioned before: encoder and decoder. Going a little more in detail in each of the parts can be stated:

- **Encoder**: it processes the input data e.g. RGB image and creates more sophisticated high level representations from the input. This block is the one that has produced the most different solutions. Some state-of-the-art works propose that the encoder be by convolutional layers [3, 5] and others by recurring layers [8, 7, 1].
- **Decoder**: it takes the encoder's output and creates the final caption. This block is usually implemented using a Recurrent Neural Network (RNN). The final layer is usually a fully connected layer [8, 7] with the number of units equal to number of possible words [1].

As it can be observed, an autonomous captioning system can be reformulated as one that given an entry $\mathbf{X}$, the system is able to obtain the most relevant characteristics that allow it to title the entry through a series of meaningful words lexicon.

Once explained in a generic way how the audio captioning system (input-output configuration) is composed. There are a series of metrics that allow evaluating the performance of the proposed system. These metrics are: BLEU [9], ROUGE$_L$ [10], METEOR [11], CIDERr [12], SPICE and SPIDEr [13].

In order to validate automated audio captioning systems the Clotho dataset will be used [2]. This is the first captioning dataset manually labeled using only audio data information. All the information on how it has been labeled and the postprocessing procedure can be found in [14]. This dataset is made up of audios from 15 seconds long to 30 seconds long. Each audio has 5 captions that can vary from 8 to 20 words. There are a total of 4981 audio samples and therefore 24905 captions. All audios are from the Freesound platform and their titles have been made using the Amazon Mechanical Turk tool with annotators from English-speaking countries.

The total number of possible words in the dataset is 4365. In turn, the dataset is divided into 3 parts: development, evaluation and

[*] Equal contribution

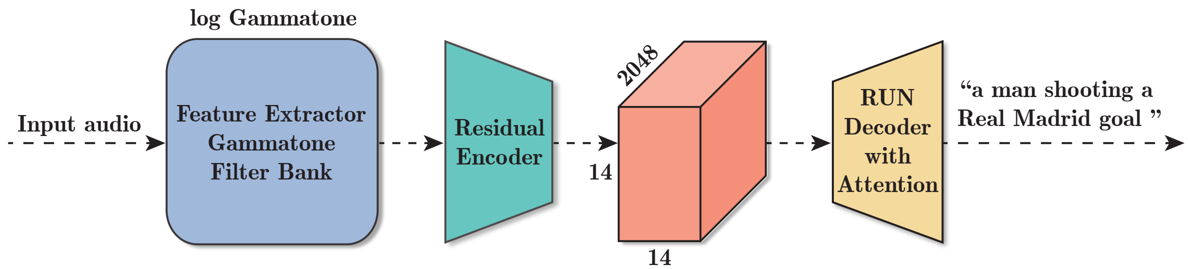

Figure 1: Proposed system representation. Audio is first represented using Gammatone filter bank. Later, a richer representation of the audio is obtained by means of a residual encoder. The final caption is obtained by using a recurrent decoder with the attention mechanism.

testing. Some of the details of this separation are that only an audio sample can appear in one of the partitions and not in two. There is no word that appears only in a split. The appearance of the words is proportional to the partition percentage: 60 % development, 20 % evaluation and 20 % testing.

During the development phase of the contest, the development and evaluation partitions have been released. The development split consists of 2893 audio clips and 14465 captions. This partition is used to train the system. The evaluation split consists of 1045 audio clips and 5225 captions. This partition is used to validate the training and check its generalization. This block is used to choose the model that will be used to predict once the testing partition is released. All partitions take into account the consideration of the proportional appearance of words in each split as explained before.

The aim of this work is to propose a solution based on an encoder-decoder structure where the encoder corresponds to a convolutional neural network and the decoder to a recurrent neural network. To achieve more accurate results the decoder implements an attention mechanism. An analysis of different state-of-the-art residual networks as possible encoders is carried out. In addition, the representation of the audio is done using the Gammatone filter bank.

The rest of the paper is organized as follows. Section 2 explains the baseline method and the proposed method in this submission. Section 3 shows the results obtained by all the experiments compared to the baseline. Finally, Section 4 concludes this work.

2. METHOD

2.1. Baseline system

The starting point of this task consists in an encoder-decoder structure formed by a Recurrent Neural Network (RNN) in both blocks. The encoder corresponds to a RNN of 3 bi-directional GRU layers. All GRU layers have the same number of features, 256. The decoder in implemented with just one GRU layer of 256 features and a classification layer of 4637 (length necessary to represent all possible words in one-hot encoding).

The input is a log-Mel Spectrogram audio representation with 64 Mel filters. The temporal bins are obtained using a 46 ms window length with 50% overlap. Representation are padded with zeros at the beginning when the audio length is less than the maximum possible in the batch.

A post-processing for captions is needed in order to train the system. The steps carried out are:

- Remove all punctuacion
- Change all letters to small case
- Tokenization, that is, assign specific index identification to each word
- Add start of sequence token, i.e. `<sos>`
- Add end of squence token and pad when necessary to the maximum caption length possible, i.e. `<eos>`

The autoncoder is optimized during training using Adam optimizer [15]. The batch size is set to 16 samples and the number of epochs to 300. The loss function is set to cross-entropy loss. The learning rate start with a value of 10^{-4} and before every weight update, the 2-norm of the gradients clipped using as a threshold the value of 2.

2.2. Proposed system

The system proposed in this work is based on [3, 16]. The main idea on [16] is that the captioning system learn where to look in the representation in order to predict the next sequence word. The attention mechanism forces the system to look for the relevant part in the encoding feature map. A full representation of the proposed framework can be seen in Figure 1.

In this work, the encoder block is a CNN. Residual networks have been the choice. Residual network were first introduced in [17]. Residual layers are designed in order to approximate a residual function as $\mathcal{F}(\mathbf{X}) \coloneqq \mathcal{H}(\mathbf{X}) - \mathbf{X}$, where $\mathcal{H}$ represents the feature to be fit by a set of stacked layers and $\mathbf{X}$ represents the input to the first of such stacked layers. Therefore, $\mathcal{H}$ can be expressed as $\mathcal{H}(\mathbf{X}) = \mathcal{F}(\mathbf{X}) + \mathbf{X}$. The reason why this kind of CNN have been chosen corresponds to the idea that the network training might be easier if optimizing a residual mapping instead of an unreferenced one, as in a classical CNN [18]. This residual learning can be easily implemented by adding a shortcut connection that would perform as identity mapping, that is, adding the input $\mathbf{X}$ to the output of the residual block $\mathcal{F}(\mathbf{X})$. In this work, different state-of-the-art Residual networks have been implemented as encoder block [17, 19, 20].

The decoder is implemented via RNN. As explained before, this block task is to look at the encoder's output and generate the final caption word by word. The decoder is implemented with a LSTM

System	$BLEU_1$	$BLEU_2$	$BLEU_3$	$BLEU_4$	$ROUGE_L$	Cider	SPICE	SPIDEr
Baseline	0.389	0.136	0.055	0.015	0.262	0.074	0.033	0.054
Enc-50-DecAtt	0.4528	0.2058	0.0976	0.0488	0.3066	0.1220	0.0602	0.0911
Enc-101-DecAtt	0.4635	0.2168	0.1074	0.0562	0.3133	0.1438	0.0648	0.1043
Enc-152-DecAtt	0.4480	0.2077	0.1023	0.0537	0.3100	0.1240	0.0628	0.0934
Enc-Wide101-DecAtt	0.4452	0.2053	0.1049	0.0572	0.3092	0.1253	0.0641	0.0947

Table 1: Results with different Residual configurations

layer. The final layer of the decoder corresponds to a Dense layer of 4637 units with sigmoid activation that indicates the probability of each word to be used as caption. This decoder also implement an attention mechanism to allow the encoder to analyze different parts of the encoder's output as it has to predict the next word. In practice, this is a smart weighted average across all encoder's outputs feature maps. The higher weights will indicated relevancy in that features. This averaged feature map is passed as input to generate the next word. The attention mechanism takes into account the sequence created at the specific moment before predicting the next word and looks for the next part of the encoder's output to be more relevant. This is done by a a DNN with a softmax activation at the last layer in order to assign the relevancy of each part of the encoder's output.

The input of this framework has been decided to be the Gammatone audio representation [21, 22]. The number of frequency bins is set to 64 as well as the baseline system. All post-processing of the captions is done with the same procedure as the baseline. Training procedure also reamins the same as baseline system. This work implementation can be found in the following link*.

3. RESULTS

Table 1 shows the results obtained for the different implementations submitted to this challenge. As it can be appreciated, the study module in this challenge has been the encoder. For this, different state-of-the-art architectures incorporating residual learning have been implemented. The Resnet50, Resnet101 and Resnet152 residual architectures were proposed in the original paper where the residual learning was first presented [17]. The idea behind them is the same and they are implemented with the same convolutional blocks placed in the same sequential way. That is, they have the same kernel size or number of filters. The difference between them is how many times each convolutional block is repeated. Thus Resnet50 is a 50-layer residual network, Resnet101 is a 101-layer network, and Resnet152 is a 152-layer network.

Since the introduction of this type of network, a multitude of combinations or studies have been proposed to improve the behaviour of the residual network, either in reducing the number of layers or the training time by maintaining or improving the results obtained. One solution that aims to maintain the benefits of residual learning but with less deep networks is the Wide Residual Networks [20]. In this submission, the Resnet101 network has been tested but implementing the modifications presented in [20].

Table 1 presents all the metrics obtained on the validation set except the METEOR metric. If we analyze the SPIDEr metric (since it is the one that is going to be used to rank the systems in the challenge) it can be noticed that the networks with 101 layers are those that show a better behavior. The system with the Resnet101 encoder obtains a value of 0.1043, doubling the result presented in the baseline. As it can be observed, the system that implements the Resnet152 network shows a worse result, which may be a case of overfitting. On the other hand, Resnet50 shows the worst result, being in this case an example that the system is too tiny to extract relevant features from the input Gammatone spectrogram. On the other hand, the encoder with wide residual learning does not achieve any improvement in the classic residual architecture.

For more clarity, a table is presented with the relationship between the system studied and the label used in the challenge submission

Encoder used	Submission name
Enc-50	Naranjo-Alcazar_UV_task6_1
Enc-101	Naranjo-Alcazar_UV_task6_2
Enc-152	Naranjo-Alcazar_UV_task6_3
Enc-Wide101	Naranjo-Alcazar_UV_task6_4

Table 2: Relationship between the name of the submission and the implementation explained in this paper.

4. CONCLUSION

Audio captioning is a very novel task in the field of machine listening. Automated captioning is a problem that has been getting the attention of the image research community for a few years now. Thanks to the recent release of a dataset specially designed for audio captioning and the proposal of this task, the first novel solutions will be proposed. In this work, a state-of-the-art image captioning network is implemented in the problem of audio captioning by making a study in the encoder block that is in charge of extracting the information from the audio. It has been decided to change the state of the art representation based on Mel filters and to use the Gammatone filter bank that has shown better results in other tasks performed by this same team.

5. ACKNOWLEDGEMENTS

The participation of Javier Naranjo-Alcazar and Dr. Pedro Zuccarello in this work is partially supported by Torres Quevedo fellowships DIN2018-009982 and PTQ-17-09106 respectively from the Spanish Ministry of Science, Innovation and Universities.

*https://github.com/sergipc22/dcase20_task6/tree/develop

6. REFERENCES

[1] 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

[2] 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

[3] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3156–3164.

[4] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4651–4659.

[5] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko, "Translating videos to natural language using deep recurrent neural networks," *arXiv preprint arXiv:1412.4729*, 2014.

[6] A. Shin, K. Ohnishi, and T. Harada, "Beyond caption to narrative: Video captioning with multiple sentences," in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 3364–3368.

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

[8] 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," *arXiv preprint arXiv:1406.1078*, 2014.

[9] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.

[10] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, July 2004, pp. 74–81. [Online]. Available: https://www.aclweb.org/anthology/W04-1013

[11] A. Lavie and A. Agarwal, "METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments," in *Proceedings of the Second Workshop on Statistical Machine Translation*. Prague, Czech Republic: Association for Computational Linguistics, June 2007, pp. 228–231. [Online]. Available: https://www.aclweb.org/anthology/W07-0734

[12] R. Vedantam, C. L. Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 4566–4575.

[13] S. Liu, Z. Zhu, N. Ye, S. Guadarrama, and K. Murphy, "Improved image captioning via policy gradient optimization of spider," 10 2017, pp. 873–881.

[14] S. Lipping, K. Drossos, and T. Virtanen, "Crowdsourcing a dataset of audio captions," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, Nov. 2019. [Online]. Available: https://arxiv.org/abs/1907.09238

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

[16] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, 2015, pp. 2048–2057.

[17] 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.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[19] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.

[20] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016.

[21] S. Tabibi, A. Kegel, W. K. Lai, and N. Dillier, "Investigating the use of a gammatone filterbank for a cochlear implant coding strategy," *Journal of neuroscience methods*, vol. 277, pp. 63–74, 2017.

[22] Z. Zhang, S. Xu, S. Cao, and S. Zhang, "Deep convolutional neural network with mixup for environmental sound classification," in *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*. Springer, 2018, pp. 356–367.