MULTI-TASK REGULARIZATION BASED ON INFREQUENT CLASSES FOR AUDIO CAPTIONING

Emre Çakır, Konstantinos Drossos, and Tuomas Virtanen

Audio Research Group
Tampere University
Tampere, Finland
{firstname.lastname}@tuni.fi

ABSTRACT

Audio captioning is a multi-modal task, focusing on using natural language for describing the contents of general audio. Most audio captioning methods are based on deep neural networks, employing an encoder-decoder scheme and a dataset with audio clips and corresponding natural language descriptions (i.e. captions). A significant challenge for audio captioning is the distribution of words in the captions: some words are very frequent but acoustically non-informative, i.e. the function words (e.g. "a", "the"), and other words are infrequent but informative, i.e. the content words (e.g. adjectives, nouns). In this paper we propose two methods to mitigate this class imbalance problem. First, in an autoencoder setting for audio captioning, we weigh each word's contribution to the training loss inversely proportional to its number of occurrences in the whole dataset. Secondly, in addition to multi-class, word-level audio captioning task, we define a multi-label side task based on clip-level content word detection by training a separate decoder. We use the loss from the second task to regularize the jointly trained encoder for the audio captioning task. We evaluate our method using Clotho, a recently published, wide-scale audio captioning dataset, and our results show an increase of 37% relative improvement with SPIDEr metric over the baseline method.

Index Terms— audio captioning, Clotho, multi-task, regularization, content words, infrequent classes

1. INTRODUCTION

Audio captioning is the novel task of automatically generating textual descriptions (i.e. captions) of the contents of general audio recordings [1, 2]. Audio captioning started in 2017 [3], and it can be considered as an inter-modal translation task, where the humanly perceived information in the audio signal is translated to text. For example, given an audio recording, a caption can be "*the wind blows while cars are passing by*" or "*a man alternates between talking and flapping a piece of cloth in the air three times*"[1].

Existing audio captioning methods are deep neural networks (DNNs) based, mostly employing the sequence-to-sequence paradigm. An encoder gets as an input the audio sequence, processes it, and outputs a sequence of learned feature vectors. Then, the output sequence of the encoder is aligned with the targeted output sequence of the decoder, typically by two alternative methods. The first, is the one proposed in [4], where the encoder outputs a fixed length vector, and this vector is used as an input to the decoder and for every time-step of the output sequence. The second method for sequence alignment, is through the attention mechanism proposed in [5], where for each time-step of the output sequence, the alignment mechanism calculates a weighted sum of the output sequence of the encoder, conditioned on the state of the decoder. For example, in [3] the method uses an encoder-decoder scheme, utilizing a multi-layered and recurrent neural network (RNN) based encoder and an RNN-based decoder. The encoder gets as an input the audio signal, and its output is processed by the attention mechanism presented in [5]. The output of the attention mechanism is used as an input to an RNN-based decoder, followed by a classifier which outputs the predicted words at each time-step of the output sequence. Study [6] presents another method, where the input sequence is encoded to a fixed length vector, through an RNN-based encoder and by a time-averaging of the output of the encoder. Then, the fixed length vector is used as an input to an RNN-based decoder, for every time-step of the output sequence, similarly to [4]. Again, a classifier predicts the output words. In addition, the work in [7] presents an approach where a VGGish encoder is used to process the input audio sequence. The output of the VGGish-based encoder is processed by an attention mechanism, and a sub-sequence RNN-based decoder followed by a classifier, outputs the predicted words. Finally, an autoencoder with attention mechanism is also employed in [1] for establishing initial results for the audio captioning dataset called Clotho. Clotho is a novel audio captioning dataset, employing around 5000 clips with five captions for each audio clip, amounting to a total of around 25 000 audio clips and caption examples. Clotho is built with emphasis on diversity and robustness, utilizing established good practises for dataset creation from the machine translation and image captioning communities [1, 2]. Clotho offers captions that are sanitized from speech transcription, typos, and named entities, provides splits with no hapax-legomena (i.e. words appearing only once in a split) [8] and it is used in the audio captioning task of the DCASE 2020 challenge[2].

Though, in many natural language-based datasets (like Clotho) it is observed that there is a typical class imbalance [9, 10]. The function words, i.e. the articles (e.g. "a"/"an", "the"), prepositions (e.g. "in", "over", "from", "about"), and conjunctions (e.g. "and", "or", "the", "until") appear in overwhelming amounts compared to the other words, i.e. the content words. The frequency of appearance of content words, most likely will cause the common machine

The authors would like to acknowledge CSC-IT Center for Science, Finland, for computational resources.

[1] Actual captions from Clotho dataset.

[2] http://dcase.community/challenge2020/task-automatic-audio-captioning

learning optimization methods, such as gradient descent, to overfit to them, since they are the ones that affect the most to the loss function. This is especially undesirable for audio captioning in two major ways. Function words most often do not possess any information about the audio content, which makes it even harder to map the acoustic features to these most common words, i.e. classes. In addition, the class imbalance between function and content words, prevents the learning of the acoustically more informative, content words, since they contribute less to the total learning loss. On the other hand, a valid caption generated by an automatic audio captioning system must anyway include the function words in the appropriate places to be grammatically correct.

To tackle the above, we draw inspiration from traditional natural language processing techniques. Specifically, we consider the cases where the class imbalance between function and content words is treated with employing weights of the loss for the words [9, 11], and we propose a novel regularizing method for the encoder that process the audio sequence, employing a data preprocessing and a multi-task learning set-up. We first identify the function and content words. Then, additionally to the task of predicting the proper sequence of words (i.e. caption) for a given audio input, we utilize an extra learning signal for the encoder. This signal emerges from an extra decoder followed by a classifier, which try to predict the content words for the corresponding input.

The rest of the paper is organized as follows. In Section 2 is our proposed method and Section 3 describes the followed evaluation procedure. The obtained results and their discussion are in Section 4. Section 5 concludes the paper and proposes future research directions.

2. PROPOSED METHOD

The proposed method consists of two stages: feature and target extraction, and deep learning based sequence-to-sequence classifier. Given an audio recording, our method first extracts the acoustic features, and then uses a recurrent neural network based autoencoder with two separate decoders to generate an audio content description as a sentence. While the system is trained with both caption decoder and content word decoder, only caption decoder is used to obtain the generated content description. The system overview is given in Figure 1.

The proposed method is based on the baseline system for audio captioning task of the DCASE 2020 challenge[2], and includes several extensions to this work. In order to alleviate the class imbalance problem on the learning of acoustically informative words, we propose two main extensions: multi-task regularization based on content words, and loss weighting based on word frequency. For the rest of the paper, these methods are referred as CWR-CAPS (content word regularized captioning system) and CWR-WL-CAPS (content word regularization with weighted loss captioning system). The open-source code repository for this work, written as an extension to official challenge baseline system, is available in [3].

2.1. Feature and target extraction

We use log mel band energies as acoustic features. Since Clotho dataset comes with varying length audio recordings, zero-padding is applied at the end for the shorter recordings in each batch. As a result, the acoustic features $\mathbf{X} \in \mathbb{R}^{N \times T}$ for each recording are obtained, where N is the number of bands, and T is the maximum number of frames in a recording for a given batch.

Each caption is pre-processed by making all the words lowercase, removing the punctuation and adding start- and end-of-sequence tokens ([SOS] and [EOS]). The target outputs for captions of a recording is a matrix $\mathbf{Y} \in \mathbb{R}^{K \times T'}$, where K is the number of unique words in the dataset including [SOS] and [EOS], and T' is the length of the longest caption target output vector in a batch. Each column of $\mathbf{Y}$ is a one-hot vector representing the index of a word in the caption at each timestep. Similar with the input features, the target output length is varying among the recordings, therefore the shorter target outputs in each batch are padded with [EOS] tokens.

We define a second set of the target outputs for audio captioning, namely *content words*, to be used to introduce regularization over the encoder outputs. The content words are defined as the set of words in the given dataset of captions, excluding the prepositions, articles, conjunctions and auxiliary verbs. The list of content words used in this work can be found in the open-source code repository [3]. As a result, for each given caption, we obtain a multi-label encoded binary content word vector $\mathbf{y}' \in \mathbb{R}^{K'}$. If the i^{th} content word is present in the caption, then $\mathbf{y}'_i$ is set to 1, and 0 vice versa.

2.2. Sequence-to-sequence classifier

The proposed method is a sequence-to-sequence deep learning classifier with two sets of target outputs: captions and content words. The input to the system is log mel band energy features. This input is fed to an encoder block which consists of bidirectional Gated Recurrent Unit (GRU) [12] layers. Dropout [13] is applied after each GRU layer.

The output of the encoder is then fed to two separate decoder branches, namely caption decoder and content word decoder. The difference between captions and content words, and how the content words are obtained are explained in Section 2.1. Both decoder blocks include a single unidirectional GRU layer, and a fully connected (FC) layer. The FC layers in both decoder blocks apply the same set of weights over the RNN outputs at each timestep.

The differences between the caption and content word decoder processes are as follows. The main difference is the nonlinearity applied to the weighted outputs. In the case of caption decoder, the FC layer nonlinearity is softmax, whereas for the content word decoder, it is sigmoid function to allow multiple content word outputs being detected for the same input. Another difference is that the final outputs for the content word decoder are collapsed in time axis by taking the maximum value over time, in order to obtain a single probability vector for the whole recording. The caption decoder outputs are also treated as probabilities at the frame level, where each column represents the probabilities of the words at a given frame (timestep). During inference, the caption decoder output is determined as the word with the highest probability at each time step.

2.3. Training

The sequence-to-sequence classifier is trained using Adam gradient optimizer [14]. The upper boundary of the squared norm of the gradients is selected as 1 to prevent exploding gradients. The classifier is trained using a patience scheme, where the training is aborted if the SPIDEr metric (explained in Section 3.2) for the evaluation

[3] https://github.com/emrcak/dcase-2020-baseline/tree/sed_caps

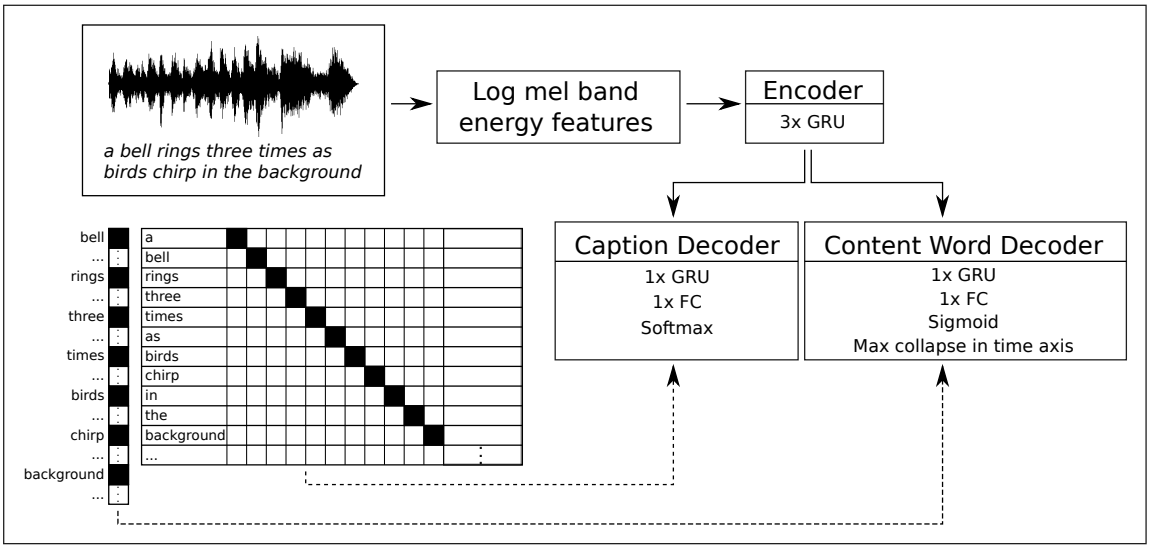

Figure 1: System overview.

dataset does not improve for certain number of epochs. As the final model, we use the model from the epoch with the best validation SPIDEr score.

Weighted non-negative log likelihood and cross-entropy loss are used as objective loss functions for caption decoder and content word decoder outputs, respectively. The purpose of using the additional loss from the content words is to regularize the encoder to produce intermediate representations that contain more information on the content words. Our empirical analysis show that the magnitude of loss corresponding to the content words is consistently about 10% of the captioning loss over the training. Therefore, this additional loss does not dominate the whole training and can indeed be seen as acting as a regularizer over the encoder.

The class weights for the non-negative log likelihood loss of the caption decoder output are determined as inversely proportional to the amount of each word's occurrences in the development dataset. This leads the classifier to avoid overfitting on more common but less informative words, due to their smaller weight on the total loss function. This method also provides a better matching with some of the commonly used captioning evaluation metrics such as CIDEr [15], which uses Term Frequency Inverse Document Frequency (TF-IDF) [16] weighting that puts more emphasis on the detection of the less common words.

2.4. Other changes to baseline method

Apart from the proposed content word based regularization and weighted loss schemes, there are a few additional changes made to baseline system. For the baseline system, the input features from the shorter (in time) recordings are padded with zero vectors for the beginning timesteps to have an equal sized feature matrix between the examples in a batch. In order to better match the target outputs being padded at the end, we move the input feature padding also to the end. In addition, validation SPIDEr score based early stopping is added to the baseline system. Also, the gradients are reset after processing each batch (this was initially missing from the baseline system - hence from baseline results -, but later added to baseline code repository).

3. EVALUATION

3.1. Dataset

In correspondence with DCASE 2020 challenge task on audio captioning (task 6), Clotho [17] dataset is used for development and evaluation. Clotho consists of 15 to 30 seconds long recordings collected from FreeSound platform [4], and each recording is annotated with five different captions using crowd-sourcing. In this work, development split of Clotho is used for training the systems, and the performance is evaluated using the evaluation split.

3.2. Evaluation Metrics

For the assessment of the performance of our method, we employ the proposed metrics from the audio captioning task at DCASE 2020 challenge[2]. These metrics can be divided in two categories. Firstly there are the machine translation metrics, which are BLEU_n [18], ROUGE_L [19], and METEOR [20]. BLEU_n calculates a weighted geometric mean of the precision of n-grams (typically $n \in \{1, 2, 3, 4\}$) between predicted and ground truth captions, ROUGE_L calculates an F-measure using the longest common sub-sequence (also between predicted and ground truth captions), and METEOR is based on a harmonic mean of the precision and recall of segments, from the predicted and ground truth captions. Then, there are the captioning metrics which are the CIDEr [21], SPICE [22], and the linear combination of these two metrics called SPIDEr [23]. CIDEr uses a weighted sum of the cosine similarity of n-grams, between the predicted and ground truth captions, and SPICE measures how well the predicted caption recovered objects, scenes, and relationships of those, according to the ground truth caption. SPIDEr is the average of CIDEr and SPICE, and it evaluates both fluency and semantic properties of the predicted captions [23].

[4] https://freesound.org/

Metric	Baseline	CWR-CAPS	CWR-WL-CAPS
B_1	38.9	39.0	40.9
B_2	13.6	14.3	15.6
B_3	5.5	6.3	7.3
B_4	1.5	2.4	3.0
R	26.2	27.0	27.8
M	8.4	8.5	8.8
CIDEr	7.4	8.9	**10.7**
SPICE	3.3	3.6	**4.0**
SPIDEr	5.4	6.3	**7.4**

Table 1: Percentage results for baseline, CWR-CAPS and CWR-WL-CAPS. B_N stands for BLEU, R is for ROUGE, and M is for METEOR.

3.3. Hyperparameters

The specific hyperparameters used in this work are as follows. The feature extraction and the model architecture hyperparameters are kept the same with the baseline method for better comparability. The number of log mel bands for feature extraction is selected as 64, and Hamming window of 46 ms length with 50% overlap is used for frame division. the total number of content words is 88. For the encoder, we use three bidirectional GRU layers with 512 units each. The dropout probability used in the encoder is 0.25. For the decoder, we use one GRU layer with 512 units. For the autoencoder classifier training, the batch size is selected as 32 and the Adam learning rate is selected as 10^{-4}. The maximum number of training epochs is set to 300, with 100 patience epochs before aborting.

4. RESULTS AND DISCUSSION

The performance results for CWR-CAPS and CWR-WL-CAPS with the DCASE 2020 challenge official metrics are given in Table 1. CWR-WL-CAPS method offers 37% relative increase on SPIDEr compared to baseline, and also performs better on other metrics. Moreover, comparing CWR-CAPS and CWR-WL-CAPS, there is a considerable benefit for using weighted loss for the caption outputs. The benefit is more evident in CIDEr metric compared to SPICE. This is also consistent with the theoretical expectations, due to TF-IDF weighting in CIDEr calculation (as mentioned in Section 2.3).

While both methods perform better than the baseline, the produced captions still mostly lack the structure of a grammatically valid sentence. Even though the non content word contribution to the objective loss is decreased significantly, words such as *is*, *are*, and *and* appear repetitively towards the end of many of the produced captions. This can be attributed to the fact that both caption and content word autoencoders aim to map acoustic features to the words, without e.g. a language-model based prior. Longer-term temporal modeling of the captions can be improved with a language model, trained with the given captions and also with external text material, which would be then used together with the autoencoder acoustic model, and the outputs would be produced using e.g. beam search algorithm [24]. We currently consider this approach as the future work for this task.

5. CONCLUSIONS

In this paper, we propose two methods for automated audio captioning. These methods are based on content word regularization and weighted objective loss, both using recurrent neural network based autoencoder. This work is evaluated in the framework of DCASE 2020 challenge task on audio captioning, and both proposed methods provide a considerable boost over the baseline results of the challenge. Still, the produced captions mostly lack the correct English grammatical structure, and addressing this problem using external language models is planned as future work.

6. REFERENCES

[1] K. Drossos, S. Lipping, and T. Virtanen, "Clotho: An audio captioning dataset," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 736–740.

[2] S. Lipping, K. Drossos, and T. Virtanen, "Crowdsourcing a dataset of audio captions," in *Detection and Classification of Acoustic Scenes and Events (DCASE) 2019*, Oct. 2019.

[3] K. Drossos, S. Adavanne, and T. Virtanen, "Automated audio captioning with recurrent neural networks," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2017, pp. 374–378.

[4] 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 *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1724–1734. [Online]. Available: https://www.aclweb.org/anthology/D14-1179

[5] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proceedings of the International Conference on Learning Representation (ICLR)*, 2014.

[6] M. Wu, H. Dinkel, and K. Yu, "Audio caption: Listen and tell," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 830–834.

[7] C. D. Kim, B. Kim, H. Lee, and G. Kim, "AudioCaps: Generating captions for audios in the wild," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Jun. 2019, pp. 119–132.

[8] I.-I. Popescu and G. Altmann, "Hapax legomena and language typology," *Journal of Quantitative Linguistics*, vol. 15, no. 4, pp. 370–378, 2008.

[9] C. Padurariu and M. E. Breaban, "Dealing with data imbalance in text classification," *Procedia Computer Science*, vol. 159, pp. 736 – 745, 2019, knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 23rd International Conference KES2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050919314152

[10] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks*, vol. 106, pp. 249–259, 2018.

[11] V. S. Sheng and C. X. Ling, "Thresholding for making classifiers cost-sensitive," in *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 1*, ser. AAAI'06. AAAI Press, 2006, p. 476–481.

[12] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8)*, 2014.

[13] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting." *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[14] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International conference on learning representations (ICLR)*, 2015.

[15] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4566–4575.

[16] S. Robertson, "Understanding inverse document frequency: on theoretical arguments for IDF," *Journal of documentation*, 2004.

[17] 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

[18] 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.

[19] 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

[20] 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*, 2007, pp. 228–231.

[21] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "CIDEr: Consensus-based image description evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2015, pp. 4566–4575.

[22] P. Anderson, B. Fernando, M. Johnson, and S. Gould, "SPICE: Semantic propositional image caption evaluation," in *European Conference on Computer Vision*. Springer, 2016, pp. 382–398.

[23] S. Liu, Z. Zhu, N. Ye, S. Guadarrama, and K. Murphy, "Improved image captioning via policy gradient optimization of spider," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 873–881.

[24] C. Tillmann and H. Ney, "Word reordering and a dynamic programming beam search algorithm for statistical machine translation," *Computational linguistics*, vol. 29, no. 1, pp. 97–133, 2003.