ISSN: 2278-4632
Vol-11 Issue-01 2021

CRNN AND GRU REINFORCEMENT LEARNING FOR AUDIO CAPTIONING

Dr. B.SUBRAHMANYESWARA RAO[1], Mr. KURELLA AYYAPPA RAVI KIRAN[2], Mr. LEGALLA ANJANEYULU[3]
PROFESSOR[1], ASSISTANT PROFESSOR[2,3], DEPARTMENT OF ECE,
SWARNANDHRA COLLEGE OF ENGINEERING AND TECHNOLOGY, NARASAPUR

ABSTRACT

The goal of audio captioning is to create a natural sentence that describes the information contained in an audio recording. To solve this multi-modal challenge, this research suggests using a powearful CRNN encoder in conjunction with a GRU decoder. To generate richer and more accurate captions, reinforcemint learning is also explored in addition to traditional cross-entropy. Our method achieves a relative improvement of at least 34% on all metrics presented, outperforming the baseline model significantly. The results show that a Spider of 0 is achieved by our suggested CRNNGRU model with reinforcement learning.

REINFORCEMENT LEARNING IN CRNN AND GRU FOR AUDIO CAPTIONING

The Clotho evaluation set has 190 points. Additional data augmentation raises the performance to 0.223. Using only 5 million parameters, we ranked second on five metrics in the DCASE challenge Task 6—inclouding BLEU, ROUGE-L, and METEOR—while keeping a modest model size. We placed fourth based on Spider. Index Terms: convolitional recurrent neural networks, audio captioning, and reinforcement learning

INTRODUCTION

The Clotho evaluation set has 190 points. Additional data augmentation raises the performance to 0.223. Using only 5 million parameters, we ranked second on five metrics in the DCASE challenge Task 6—inclouding BLEU, ROUGE-L, and METEOR—while keeping a modest model size. We placed fourth based on Spider. Index Terms: convolitional recurrent neural networks, audio captioning, and reinforcement learning but he questioned whether widely used machine translation criteria could accurately assess the overall performance.

The key point of contention is that, despite their approach producing quantifiably near-human performance through objective measurements, human review frequently finds the generated sentences to be less valuable. Exposure bias is present in audio captionIng, just like it is in other text production jobs like machine translation and image captioning. Usually taught in a "teacher forcing" manner, neural network-based models seek to maximize the probability of a future ground-truth word given the current ground-truth word. Nevertheless, the model can only infer the future word during inference by using its own predicted current word; ground-truth annotations are only available during trainIng. During the test, this causes an accumulation of errors. One additionalA policy is defined by the model parameters, and its action is reflected in the current generated word selection. The incentive is derived from the sampled sentence's evaluation scores (BLEU, METEOR, Cider, etc.). Using the reward, policy-gradient [9] is used to estimate the gradient of the agent parameters. By leveraging incentives from greedy-sampled words as the baseline to lower the high varyacne of payouts, work in [10] enhances this technique. Instead of selecting words at random from the action space, subsequent work in [11] also uses actor-critic approaches [12] to determine the worth of generated words. In this work, we investigate audio captioning using the self-critical sequence training (SCST) method (as suggested in [10]). The format of this paper is as follows: in Section 2, we present our

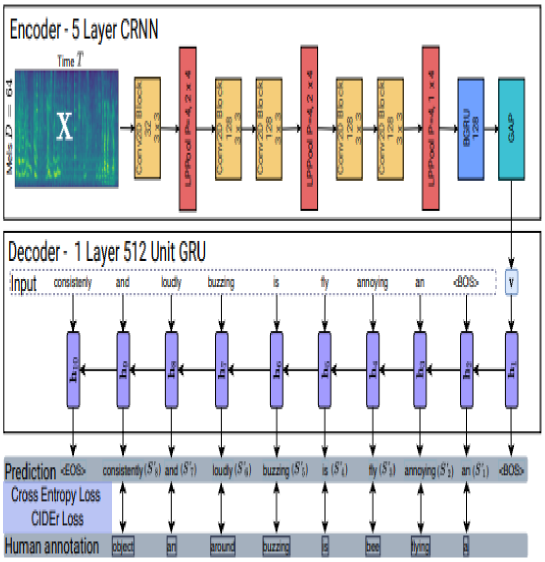

Figure 1: Our proposed encoder-decoder architecture. The encoder is a CRNN model which outputs a fixed-sized 256-dimensional embedding v after a global average pooling layer (GAP).

A convolution block refers to an initial batch normalization, then a convolution, and lastly, a Leakier (slope −0.1) activation. The numbers in each block represent the output channel size and the kernel size. For example, ”32, 3 × 3” means the convolution layer has 32 output channels with a kernel size of 3 × 3. All convolutions use padding in order to preserve the input size. Then a GRU decoder utilizes this audio embedding v or embedding of the word S 0 t at each time-step, to predict the next word S 0 t+1.

APPROACH

Similar to previous audio captioning frameworks [3], our approach follows a standard encoder-decoder model (see Equation (1)).

$$\begin{aligned} \mathbf{v} &= \mathrm{Enc}(\mathbf{X}) \\ [S'_1, \ldots, S'_T] &= \mathrm{Dec}(\mathbf{v}) \end{aligned} \qquad (1)$$

The encoder (Enc) is fed an audio-spectrogram (X) and produces a fixed-sized vector representation v, which the decoder (Dec) uses to predict the caption sentence. Specifically, the decoder generates a single word-token S 0 t for each time-step it up until an end of sentence () token is seen (see Figure 1). In audio captioning, decoding differs between training and evalauction stages:

$$\ell_{\mathrm{XE}}(\theta; S, \mathbf{v}) = -\sum_{t=1}^{T} \log p(S_t | \theta; \mathbf{v}) \qquad (2)$$

Equation (2) describes how Dec generates word-tokens during training, given the embedding v and human-annotated data S, under the supervision of a cross-entropy (XE) loss when transcriptions are available. Since there are no transcriptions accessible for evaluation and testing, word tokens are sampled from the decoder using the audio embedding method. This explanation makes it clear that the resulting sentence quality is directly influenced by the quality of v. Thus, the encoder design and the loss function are the two primary areas where our technique differs from other approaches. We replace the traditional GRU encoder with a resilient convolutional recurrent neural network (CRNN) in order to address concerns that previous encoder models (GRU) may not be sufficient to create a robust vector representation. In Figure 1, you can see our framework. Furthermore, there may be drawbacks to traditional XE training. individuaally, syntactically incorrect sentences can be produced. Finally, because the model must accurately replicate a sentence word by word rather than permitting semantically identical but differently worded sentences, optimizing XE necessarily results in repetitive sentences.

For audio captioning, we use reinforcement learning. We can directly back-propagate a measure (like BLEU or Cider) in the form of a reward through reinforcement learning. Formally, we train the model to minimize the sentence S 0 negative reward from a single sample.

$$\ell_{\mathrm{RL}}(\theta; \mathbf{v}) = -r(S'), S' \sim p(S'|\theta; \mathbf{v}) \qquad (3)$$

where S 0 = [S 0 1, S0 2, . . ., S0 T]. By incorporating the policy gradient method with baseline normalization, the parameter gradients can be estimated as follows:

$$\nabla_\theta \ell(\theta; \mathbf{v}) = -(r(S') - b)\nabla_\theta \log p(S'|\theta; \mathbf{v}), S' \sim p(S'|\theta; \mathbf{v}) \qquad (4)$$

here b is a pre-defined baseline normalization constant to reduce the high variance brought by sampling [12]. We set b as the greedy decoding reward because of its effectiveness in image captioning [10].

Models

Coder A CRNN model, which is our suggested encoder, has been successful in localizing sound events [13, 14]. The architecture is a five-layer CNN with three-by-three convolutions that is summarized into three blocks and followed by L4-Norm pooling. The temporal dimension is subsampled by a factor of four in the CNN blocks. The penultimate CNN output is followed by a Bigram, which effectively ends our model's ability to precisely localize sounds. Finally, we eliminate all time-variability to a single, time-independent representation v ∈ R 256 by applying a global average pooling (GAP) layer. With 679k parameters, the encoder is relatively lightweight, using only 2.7 MB of storage space. Decoder A decoder uses a fixed-sized embedding in the context of audio captioning and seeks to generate a

EXPERIMENTS

Collection Clotho [2, 15] is provided by the challenge for the audio captioning assignment. There are 4981 audio samples in all, with durations evenly spaced between 15 and 30 seconds. Every audio sample was taken from the Freedsound repository. Each sample has been annotated by five native English speakers, resulting in a total of 24905 captions available. After post-processing, captions are checked to make sure they contain eight to twenty words each and don't contain any unique terms, named entities, or speech transcription. With a ratio of 60%-20%-20%, the dataset is formally divided into three sets: development, assessment, and testing. Our audio captioning model is trained on the development and evaluation sets in the challenge, and it is evaluated on the testing set.

Data pre-processing

As our default input feature, we extract the 64-dimensional log-Mel spectrogram (LMS). Here, a single frame is retrieved every 20 ms using a Hann window size of 40 ms and a 2048-point Fourier transform. For every audio input, this yields an X ∈ R T ×D log-Mel spectrogram feature, where T is the number of frames and D = 64. Additionally, the development set's mean and standard deviation are used to normalize the input feature. To reduce the vocabulary size, we convert all letters to lowercase and eliminate punctuation from each caption in the dataset. We add unique tokens "" and "" to captions to indicate the start and finish of phrases. 90% of the available training data is divided into a model training portion.

Evaluation metrics

Our model-generated captions are assessed using eight objective metrics in total: BLEU@1-4 grams [4], METEOR [7], RougeL [5], Cider [6], and SPICE [16]. The mean of Cider and SPICE is referred to as an additional Spider metric.

Training details

We challenge the following four models with our predictions:

- Base CRNN-B. Our standard CRNN-GRU encoder-decoder model is this one.

- Word, CRNN-W. In this case, Word2Vec word-embeddings trained on the deelopement set captions are used to initialize the decoder word-embeddings.

- Ensemble CRNN-E. In this case, the output level CRNN-B and CRNN-W findings are fused.

- Reinforcement (CRNN-R). Here, we use reinforcement learning to fine-tune CRNN-W. The following goes into more depth about each contribution. XE instruction Teacher forcing is used in XE training to expedite the learning process. Every epoch, we assess the model on the validation set and choose the model with the highest BLEU4 score as the best model. We utilize Adam [17] optimizer with an initial learning rate of 5 × 10−4 and train the model for 20 epochs. The amount of the batch

RESULTS

Results

Our findings on the Clotho evaluation set are displayed in Table 1 along with a comparison to the DCASE challenge baseline, which employs a three-layer Bigram encoder and a two-layer Bigram decoder. As can be seen, our initial CRNN-B model outperforms the baseline by a large margin, indicating that a strong encoder may actually improve captioning performance. CRNN-W outperforms CRNN-B on most metrics, except for Cider and METEOR, when initialized with word embeddings using Word2Vec, which was trained on the development set captions. In terms of performance, CRNN-E performs better than CRNN-B and CRNN-W. Our top-performing model is CRNN-R. It's noteworthy that BLEU3 and BLEU4 exhibit a higher relative improvement than Cider, despite the fact that CRNN-R is tuned for Cider scores. The ROUGEL and

CONCLUSION

In this research, we offer a new method for audio captioning that involves both a reinforcement learning framework and a CRNN encoder front-end. The Clotho dataset is used to train models for audio captioning. The Clotho evaluation set results indicate that while reinforcement learning further increases the perromance considerably across all parameters, the CRNN encoder is essential for extracting meaningful audio embeddings for captioning. In the DCASE2020 task 6 challenge testing set, our method achieved a computative result on all metrics except Cider, placing it in fourth place. Notably, with the fewest parameters (5 million), our method delivers the best non-ensemble outcome without data augmentstation. We detect an additional boost in regrads to the Spider score on the evaluation set by leveraging Specie data augmentation further.

REFERENCES

[1] *"Automated audie captioning with recurrent neural networks," by K. Dross, S. Advance, and T. Virtanen, in 2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), October 2017, pp. 374–378.*

[2] "Crowdsourcing a dataset of audio captions," S. Lipping, K. Dross, and T. Virtanen, Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop (DCASE), Nov. 2019. [Online]. Currently accessible at https://arxiv.org/abs/1907.09238

The IEEE International Conference on Acoustics, Speech, and Signal Processing - Proceedings, vol. 2019-May, Institute of Electrical and Electronics Engineers Inc., may 2019, pp. 830–834. [3] M. Wu, H. Dinkel, and K. Yu, "Audio Caption: Listen and Tell."

[4] "Bleu: a method for automatic evaluation of machine translation," by K. Papini, S. Roukoops, T. Ward, and W.-J. Zhu