AUTOMATIC AUDIO CAPTIONING SYSTEM BASED ON CONVOLUTIONAL NEURAL NETWORK

Technical Report

Qianyang Wu
University of Electronic Science and Technology of China
Communication Engineering Dept
Chengdu,China
wuqianyang@std.uestc.edu.cn

Shengqi Tao
University of Electronic Science and Technology of China
Communication Engineering Dept
Chengdu,China
taosq@std.uestc.edu.cn

Xingyu Yang
University of Electronic Science and Technology of China
Communication Engineering Dept
Chengdu,China
yangxingyu@std.uestc.edu.cn

ABSTRACT

Automated audio captioning has been a new issue in natural language processing (NLP) for recent years. The key point of automatic audio captioning system is that it describes non-audio signals in the form of natural language. The system should take audio as input, and output as descriptive audio sentences. Most of approaches use seq2seq model with RNNs as both the encoder and decoder. It results in considerable time to get the training process finished. This paper proposed a neural network with CNN as the encoder and GRU as the decoder. Encoder is based on VGG16, which has deeper networks and three fully-connected layers. Despite the low accuracy of prediction, our model decreases the training time significantly. It proves that the application of CNN can be a choice for automated audio captioning.

Index Terms—audio captioning, sequence-to-sequence model, CNN, GRU

1. INTRODUCTION

In our daily life, sound signal is one of the sources that we receive the most information from the outside world. The automated audio captioning problem refers to the task of describing an audio signal in text. It can visualize sound information, and through its description of sound signal, we can receive sound information in other ways. The automatic audio captioning system for this task is not a conventional voice-to-text system, it is an intermodal translation task. The input of the system is not only the voice signal issued by human beings, but also the audio signal frequently occurring in life scenes. The output of the system is the sentence describing the input audio, rather than the speech text or object label, which contains more audio information. The system model focuses more on the recognition of human perception information of ordinary audio signals and expression through text. This information includes acoustic events, acoustic scenes, spatio-temporal relationships between sound sources, differences between foreground and background, concepts, and identification of physical properties of objects and environments. Given an audio signal, the system can identify the context in which the audio was emitted and how it was emitted, with uncertainty as to the outcome [1].
Image captioning has gone further than audio captioning. It is a combination of image processing and natural language processing, and has broad application prospects as well as audio captioning. The solution to the image captioning problem is to use the encoder-decoder structure or a structure very close to it. Image captioning has already reached a high accuracy. However, the problem of audio captioning is vague compared to images, and the recognition of audio is even subjective to different people, making it difficult to determine the best match. For most efficient system for image captioning, CNN is used as the encoder to extract image features and generate semantic vector, and RNN is used as the decoder to transform the vector into sequenced words list **错误!未找到引用源。**, **错误!未找到引用源。**.

Because of the run time is too long for RNNs, the baseline system **错误!未找到引用源。** is not convenient to tune parameters. Considering the parallel computing performance of CNNs, we proposed the model whose encoder is a convolutional neural network. For mel spectrum is a common method to extract features that is more in line with human auditory sense, it can be used as the first features extractor. Next, we consider the log mel-bands energies spectrum as an image with only one channel and put it into a CNN to extract visualized auditory characteristics. It is an innovative attempt to treat audio as a kind of "image" and input it to the CNN inside the encoder. Firstly, we chose VGG16 [5] as the encoder for it is easily to realize compared with other latest CNNs. But VGG16 is still too deep for this task, so based on its structure, we tried to reduce some convolutional layers and cut down filters for each layer. And we also tune the output dimension of fully-connected layers to fit the decoder, which is the same as baseline's decoder. It accepts semantic vector and output sequenced words list. Details will be introduced in next two sections.

2. MODEL

Our proposed method takes an audio file as input with 44.1kHz sampling frequency and 16 bits sample width and creates a textual description for it. At first, a matrix of features, $\mathbf{X} \in \mathbb{R}^{T \times M}$ are extracted from audio files. The row of $\mathbf{X}$ contains the features in a certain frame and the column contains the features of mel-spectrum. Outputs $\mathbf{Y} \in \mathbb{R}^{N \times N_{words}}$, implies the probability distribution over unique words. At last, we pick the most propable word to create a sequence of words, which is the output caption of the audio file.

For feature extraction, input audio file is framed with Hanning window of 1024 samples and 50% overlap. From each frame we extracted $\mathrm{M} = 64$ log mel-bands energies. With the extracted

audio features, we obtain the matrix of features $\mathbf{X} \in \mathbb{R}^{T \times M}$, and T is the number of frames we divide.

The neural network consists of an encoder and a decoder. Encoder is a CNN mainly including five convolution layers and three fully-connected layers. We use ReLU as its activation function. Convolution layers don't change the size of input, and max-pooling layers will compress the input with the stride equals kernel's size. We put $\mathbf{X}$ into the CNNs as an image with single channel, and features of this image is extracted through convolution layers and maxpooling layers. Then these features will transform into semantic vector $\mathbf{v} \in \mathbb{R}^{N \times 1}$ after passing three fully-connected layers and a softmax layer.

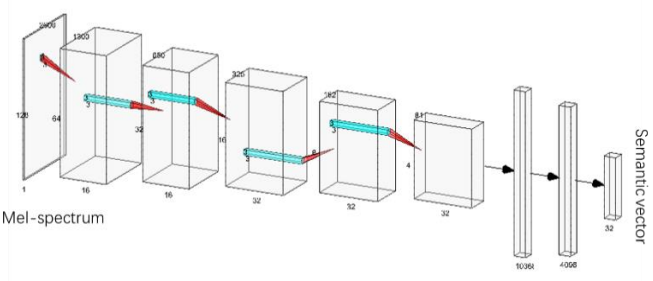

Figure 1. Illustration of the encoder

Decoder is a single-layered GRU followed by a fully-connected layer with linear activation. The GRU layer has 256 cells. It takes as input the semantic vector $\mathbf{v}$ and produces the output $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N], \mathbf{y_i} \in \mathbb{R}^{N_{words}}$, which contains the probability distribution over N_{words} unique words. For each $\mathbf{y_i}$, we select the word with the highest probability and the sequence of N words is the caption predicted for the audio file.

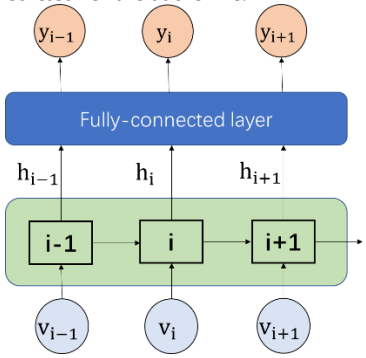

Figure 2. Block diagram of the decoder

The amount of the total parameters of this network is 60730943. The encoder and decoder are jointly trained using Adam optimizer with the default parameters mentioned in the original paper [7], and categorical cross entropy loss. Dropout rate is 0.25 in the GRU of the decoder, and batch size is 16. The code is developed using the Pytorch framework.

Table 1. Structure of the encoder

Conv2d-16(ReLU)
Maxpool
Conv2d-16(ReLU)
Maxpool
Conv2d-32(ReLU)
Maxpool
Conv2d-32(ReLU)
Maxpool
Conv2d-32(ReLU)
Maxpool
Average-pool
FC-4096
FC-4096
FC-32
Log softmax

Table 2. Experimental conditions

Dimension of mel-bands energies	128
GRU cells	256
Batch size	16
Total epochs	200
Optimizer	Adam
Learning rate	1e-5

3. EVALUATION

To evaluate the effectiveness of the proposed model, we conducted an experiment. Next, we present the evaluation results of our model.

3.1. Dataset

We used the Clotho dataset provided by DCASE2020 Task 6. The development-training split of Clotho consists of 2893 audio samples and two CSV files, and the evaluation dataset consists of 1043 audio samples and one CSV file. The sampling rate of these audio data is 44.1khz and the length is 15s to 30s. Features extracted are 128 log mel-band energies. Table 3 shows the score of the employed metrics for the baseline.

Table 3. Evaluation scores for the baseline system

Metric	Value
BLEU-1	0.389
BLEU-2	0.136
BLEU-3	0.055
BLEU-4	0.015
ROUGE-L	0.262
METEOR	0.084
CIDEr	0.074
SPICE	0.033
SPIDEr	0.054

3.2. Objective Scores

Table 4 shows the scores of each assessment indicator under the assessment dataset using the model we proposed.

Table 4. Evaluation scores for the system we proposed

Metric	Value
BLEU-1	0.379
BLEU-2	0.020
BLEU-3	0.000
BLEU-4	0.000

ROUGE-L	0.261
METEOR	0.063
CIDEr	0.024
SPICE	0.001
SPIDEr	0.012

3.3. Subjective Analysis

Compared with the baseline system [3] assessment indicators, the following is an analysis of the evaluation results for our system.

According to the description of the original paper of audio captioning, BLEU[*] is a precision-based metric. It calculates a weighted geometric mean of a modified precision of n-grams between predicted and ground truth captions. Typical lengths for n-grams are one to four, resulting in BLEU1/2/3/4, respectively. For BLEU1/2/3/4, scores of our system are lower than that of the baseline system. METEOR[*] calculates a harmonic mean of precision and recall of segments of the captions be-tween the predicted and ground truth captions. METEOR is essentially an optimization improvement of BLEU. Based on accuracy and recall, the baseline system METEOR score is higher, which is better than our system. CIDEr[*] calculates a weighted sum of the cosine similarity between the predicted and ground truth captions. The CIDEr indicator score is higher, the better. From this point of view, the baseline system is still better. ROUGEL[*] is a Longest Common Subsequence (LCS) based metric. It calculates an F-measure using LCS between the predicted and ground truth caption. ROUGE is a similar measurement method based on the recall rate. The higher the score, the better the performance. This indicator score is only 0.001 between our system and the baseline system, which is not much different. The SPICE indicator baseline system scores higher, so it is still better than our system [1].

Because CNN can realize parallel computing and RNN can't use it, the model we proposed cost less time to train. As showed in Table 5, training time of our model is about only one third of baseline's for every epoch, which indicates that the proposed model is superior to the baseline in terms of training time.

Table 5. Comparison of training time between baseline and proposed model

	Baseline	Proposed model
CPU	Intel Core i5-9600K	
RAM	64G	
GPU	NVIDIA Geforce RTX 2080 SUPER	
Memory of GPU	8G	
Layers of encoder	3	8
Layers of decoder	2	2
Training time of every epoch	300s~350s	100s~110s

4. CONCLUTION

This paper proposed a captioning system that converts audio signals into corresponding descriptive natural languages. It uses a seq2seq-based encoder-decoder model. The encoder uses CNN, has five convolutional layers to filter features, and some ReLU layer serves as a correction unit and pooling operation. The decoder uses a multi-layer RNN. Besides, one-hot code encoding method is used for word representation, and the value of the discrete feature is expanded into the Euclidean space. Then the evaluation result using cross-entropy can be used to obtain a satisfactory accuracy rate. Although the final test results are not as good as the baseline system, the proposed system reduces much run time for researchers. Furthermore, the system proves that it can be a choice to use CNN as the encoder for audio captioning.

5. REFERENCES

[1] http://dcase.community/workshop2020/.

[2] http://www.ieee.org/web/publications/rights/copyright-main.

[3] Fu, Kun , et al. "Aligning Where to See and What to Tell: Image Captioning with Region-Based Attention and Scene-Specific Contexts." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.12(2017):2321-2334.

[4] Li, Linghui , et al. "GLA: Global-local Attention for Image Description." *IEEE Transactions on Multimedia* (2017):1-1.

[5] Simonyan, Karen , and A. Zisserman . "Very Deep Convolutional Networks for Large-Scale Image Recognition." *Computer Science* (2014).

[6] Drossos, Konstantinos , S. Adavanne , and T. Virtanen . "Automated Audio Captioning with Recurrent Neural Networks." *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), 2017* IEEE, 2017.

[7] Kingma, Diederik , and J. Ba . "Adam: A Method for Stochastic Optimization." *Computer Science* (2014).

* URL : https : //arxiv.org/abs/1706.10006