AudioCaps: Generating Captions for Audios in The Wild

Chris Dongjoo Kim **Byeongchang Kim** **Hyunmin Lee** **Gunhee Kim**
Department of Computer Science and Engineering & Center for Superintelligence
Seoul National University, Seoul, Korea
{cdjkim,byeongchang.kim}@vision.snu.ac.kr {lhm1442,gunhee}@snu.ac.kr

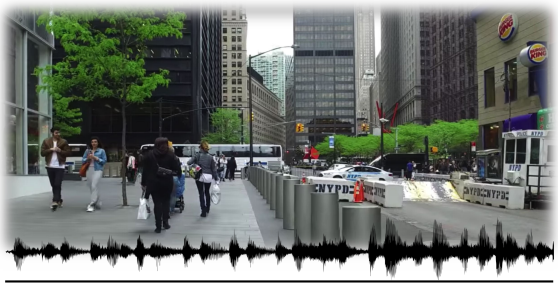
Abstract

We explore the problem of *audio captioning*[1]: generating natural language description for any kind of audio in the wild, which has been surprisingly unexplored in previous research. We contribute a large-scale dataset of 46K audio clips with human-written text pairs collected via crowdsourcing on the AudioSet dataset (Gemmeke et al., 2017). Our thorough empirical studies not only show that our collected captions are indeed loyal to the audio inputs but also discover what forms of audio representation and captioning models are effective for audio captioning. From extensive experiments, we also propose two novel components that are integrable with any attention-based captioning model to help improve audio captioning performance: the top-down multi-scale encoder and aligned semantic attention.

1 Introduction

Captioning, the task of translating a multimedia input source into natural language, has been substantially studied over the past few years. The vast majority of the journey has been through the visual senses ranging from static images to videos. Yet, the exploration into the auditory sense has been circumscribed to human speech transcription (Panayotov et al., 2015; Nagrani et al., 2017), leaving the basic natural form of sound in an uncharted territory of the captioning research.

Recently, sound event detection has gained much attention such as DCASE challenges (Mesaros et al., 2017) along with the release of a large scale *AudioSet* dataset (Gemmeke et al., 2017). However, sound classification (*e.g.* predicting multiple labels for a given sound) and event detection (*e.g.* localizing the sound of interest in a clip) may not be sufficient for a full understanding of the sound. Instead, a natural sentence offers a greater freedom to express a sound, because it allows to characterize objects along with their states, properties, actions and interactions. For example, suppose that suddenly sirens are ringing in the downtown area. As a natural reaction, people may notice the presence of an emergency vehicle, even though they are unable to see any flashing lights nor feel the rush of wind from a passing vehicle. Instead of simply tagging this sound as *ambulance* or *siren*, it is more informative to describe which direction the sound is coming from or whether the source of the sound is moving closer or further away, as shown in Figure 1.



[Audio Classification] rumble | vehicle | speech | car | outside

[Video Captioning] A bus passing by with some people walking by in the afternoon.

[Audio Captioning] A muffled rumble with man and woman talking in the background while a siren blares in the distance.

Figure 1: Comparison of audio captioning with audio classification and video captioning tasks.

To that end, we address the *audio captioning* problem for audios in the wild, which has not been studied yet, to the best of our knowledge. This work focuses on one of the most important bases toward this research direction, *contributing a large-scale dataset*. The overarching sources of in-the-wild sounds are grounded on the *AudioSet* (Gemmeke et al., 2017), so far the largest collection of sound events collected from Youtube

[1]For a live demo and details, https://audiocaps.github.io

Proceedings of NAACL-HLT 2019, pages 119–132
Minneapolis, Minnesota, June 2 - June 7, 2019.

videos. We newly collect human-written sentences for a subset of AudioSet audio clips via crowdsourcing on Amazon Mechanical Turk (section 3). We also develop two simple yet effective techniques to generate captions through the joint use of multi-level pretrained features and better attention mechanism named aligned-semantic attention (section 4). Lastly, we perform experiments contrasting between video-based captions and audio-focused captions by employing a variety of features and captioning models (section 5).

The contributions of this work are as follows.

1. To the best of our knowledge, this work is the first attempt to address the audio captioning task for sound in the wild. We contribute its first large-scale dataset named *AudioCaps*, which consists of 46K pairs of audio clips and text description.

2. We perform thorough empirical studies not only to show that our collected captions are indeed true to the audio inputs and but also to discover what forms of audio representations and captioning models are effective. For example, we observe that the embeddings from large-scale pretrained VGGish (Hershey et al., 2017) are powerful in describing the audio input, and both temporal and semantic attention are helpful to enhance captioning performance.

3. From extensive experiments, we propose two simple yet effective technical components that further improve audio captioning performance: the *top-down multi-scale encoder* that enables the joint use of multi-level features and *aligned semantic attention* that advances the consistency between semantic attention and spatial/temporal attention.

2 Related Work

Speech recognition and separation. One of the most eminent tasks for audio understanding may be speech recognition, the task of recognizing and translating human spoken language into text with less emphasis on background sound that may coexist. A multitude of datasets exist for such task *e.g.* Speech Commands dataset (Warden, 2018), Common Voice dataset (Mozilla, 2017), Librispeech (Panayotov et al., 2015), LS Speech (Ito, 2017). As one of similar lineage, automatic speech separation forks an input audio signal into several individual speech sources (Hershey et al., 2016; Ephrat et al., 2018). To most of these tasks, in the wild sound is deemed as background noise to be removed as an obstructer of speech recognition. On the other hand, our work puts the spotlight on these neglected sounds and express them through natural language.

Audio classification and sound event detection. This line of tasks emphasizes categorizing a sound into a set of predefined classes. There exist a number of datasets to aid in achieving this goal, including DCASE series (Stowell et al., 2015; Mesaros et al., 2016, 2017), UrbanSound8k (Salamon et al., 2014), ESC (Piczak, 2015). AudioSet (Gemmeke et al., 2017) is an audio event dataset collected from Youtube that is unsurpassed in terms of coverage and size, structured with an ontology containing 527 classes. Another predominant large-scale dataset is Freesound (Fonseca et al., 2017). It consists of audio samples from *freesound.org* recordings based on the preceding AudioSet ontology. In contrast to audio classification, which uniquely map the audio to a set of labels, our task generates a descriptive sentence. Hence, it needs to not only detect salient sounds of classes but also explores their states, properties, actions or interactions.

Captioning tasks and datasets. The vast majority of captioning tasks and datasets focus on the visual domain. Image captioning generates text description of an image, and numerous datasets are proposed, such as Flickr 8k (Rashtchian et al., 2010), Flickr 30k (Young et al., 2014), MS COCO (Lin et al., 2014), DenseCap (Johnson et al., 2016) and Conceptual Captions (Sharma et al., 2018). Akin to the image captioning is video captioning, for which there are many datasets too, including MSVD (Guadarrama et al., 2013), MSR-VTT (Xu et al., 2016), LSMDC (Rohrbach et al., 2017) and ActivityNet Captions (Krishna et al., 2017).Compared to previous captioning tasks and datasets, our work confines the problem by focusing on in the wild audio inputs.

Recently, there have been some efforts to solve video captioning with audio input (Hori et al., 2017, 2018; Wang et al., 2018). However, the audio input merely serves as auxiliary features for video captioning, and as a result, it only marginally improves the performance (*e.g.* BLEU-4 score: 39.6 (video only) vs. 40.3 (video + MFCC) (Wang et al., 2018)). These results are

partly culpable to dataset collection, where the annotators mostly rely on the video input. On the contrary, our collection induces the annotators to mainly abide to audio, hence, increasing the dependency of written text on the audio input as can be shown in our survey analysis in Figure 5.

3 The Audio Captioning Dataset

Our *AudioCaps* dataset entails 46K audio caption pairs. Table 1 outlines its key statistics. The audio sources are rooted in *AudioSet* (Gemmeke et al., 2017), a large-scale audio event dataset, from which we draft the *AudioCaps*, as discussed below. We present more details of data collection and statistics in the Appendix.

3.1 AudioSet Tailoring

It is important to select qualified audio clips as the first step of dataset collection. The chosen categories of clips must be well-rounded in coverage of naturally occurring audios, be relevant to practical applications and appear with high frequency. To that end, we tailor the AudioSet dataset (Gemmeke et al., 2017) that comprises 1,789,621 human-labeled 10 second YouTube excerpts with an ontology of 527 audio event categories. However, an immediate collection of captions from these audios pose several difficulties: (i) too many audio clips, (ii) inconsistent level of abstraction among the classes, (iii) distribution bias of some labels and (iv) noisy labels that are only noticeable from visual cues. We circumvent these issues through a controlled sampling process as described below.

Among 527 audio event categories of AudioSet, we first exclude all the labels whose number of clips are less than 1,000 to promote a balanced distribution within the dataset. We also remove all 151 labels in the *music* super-category, because they are often indiscernible even for a human. For example, a human with no expertise can hardly discriminate the sound of *Guitar* from *Banjo*. Thus, we set aside the musical territory for future exploration. We further discard categories if they do not satisfy the following two constraints. The word labels should be identifiable solely from sound (i) without requiring visuals (*e.g.* remove the category *inside small room*) and (ii) without requiring any expertise (*e.g.* remove *power windows* and *electric windows* because their distinction may be possible only for car experts). Finally, we select 75 word labels derived from 7 augmented super-categories as avoiding the sharp skewness in the word labels (*e.g.* 48.5% clips include *speech* label). We limit the number of instances per category to 2,000 by sampling with preference to audio clips associated with more word labels to prioritize the audios with diverse content. The final number of audio clips is about 115K, from which we obtain captions for 46K as the first version.

Split	# clips	# captions	# words/caption	# labels/clip
Train	38,118	38,118	8.79 (8)	4.25 (4)
Val	500	2,500	10.12 (9)	4.06 (3)
Test	979	4,895	10.43 (9)	4.03 (3)
Total	39,597	45,513	9.03 (9)	4.22 (4)

Table 1: Some statistics of *AudioCaps* dataset. We also show average and median (in parentheses) values. labels refer to the semantic attributes.

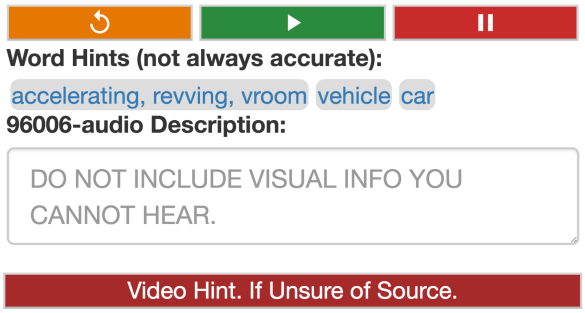


Figure 2: The AMT interface for audio annotation.

3.2 Audio Annotation

The collected captions should be precise, specific, diverse, expressive, large-scale and correlated with the paired audios with minimal visual presumptions. Such complex nature of our requirements necessitates employing crowdworkers through Amazon Mechanical Turk (AMT). Some qualification measures are set for the crowdworkers, such as they should hold a +95% HIT approval rate and the total number of approved HITs that are greater than 1,000 and be located at one of [AU, CA, GB, NZ, US]. In total, 108 caption writing workers and 3 caption reviewing workers participate and are compensated at 10 cents per clip.

Annotation Interface. Figure 2 shows our annotation interface, which is designed to minimize the visual presumption while maintaining diversity. Each task page consists of an audio clip of about 10 seconds, word hints and video hints.

The *word hints* are the word labels that are provided by AudioSet for the clip and are employed

Below officers creep toward the entrance the door and points a gun

(a) LSMDC

A black and white video of about actors

(b) MSR-VTT

A train is approaching with a low rumble and rhythmic click and squeal

(c) AudioCaps

Figure 3: Comparison between two video captioning datasets and AudioCaps. The text from (a) LSMDC (Rohrbach et al., 2017) and (b) MSR-VTT (Xu et al., 2016) includes multiple visually grounded vocabularies (indicated in blue), whereas the text from (c) AudioCaps contains vocabularies relying on auditory cues (in red).

as hints to the crowdworkers. Even to humans, recognizing the true identity of a sound can be ambiguous, and thus the word hints act as a precursor to *accurately* guide the crowdworkers during the description process, while staying aloof from visual bias. Another benefit is that the diversity of the word labels may also enrich the expressiveness of the description. Also derived from AudioSet, the *video hints* are provided as a stronger hint for sounds that are too difficult even to the human ear or for clips associated with some erroneous or missing word hints (weak labels). We advise the workers to use them as a last resort measure.

Some instructions[2] are also provided to demarcate crowdworkers' descriptions as follows. (i) Do not include the words for visuals in the video that are not present in the sound. (ii) Ignore speech semantics. (iii) When applicable, be detailed and expressive. (iv) Do not be imaginative and be literal and present with the descriptions.

Quality Control. We use a qualification test to discern many crowdworkers who frequently violate the given instructions (*e.g.* transcribing instead of describing, just enumerating provided word hints or writing visual captions). Interested crowdworkers must participate in the test and submit a response, which the authors manually check and approve if they are eligible. We employ three additional workers to verify the data in accordance to our guidelines. In order to maintain high approval rates, we periodically blacklist malicious crowdworkers while granting reasonable incentives to benevolent workers.

[2] https://audiocaps.github.io/instruction_only.html.

3.3 Post-processing

We exclude the period symbol from all the captions, convert numbers to words using `num2words`[3] and correct grammar errors by `languagetool`[4]. We then tokenize words with `spacy` [5]. Finally, we build a dictionary $\mathcal{V}$ with a size of 4506 by choosing all the unique tokens.

3.4 Comparison with Other Datasets

Figure 3 qualitatively compares some caption examples between our *AudioCaps* and two captioning datasets with audio: LSMDC (Rohrbach et al., 2017) and MSR-VTT (Xu et al., 2016). Since both LSMDC and MSR-VTT focus more on describing videos than audios, their captions are characterized by visually grounded vocabularies (blue). On the other hand, the captions of *AudioCaps* accompany sound-based vocabularies (red).

4 Approach

We present a hierarchical captioning model that can attend to the fine details of the audio. The backbone of our model is an LSTM (Hochreiter and Schmidhuber, 1997) that we fortify with two novel components which are easily integrable with any attention-based captioning model. The *top-down multi-scale encoder* enables the contextual use of multi-level features, and the *aligned semantic attention* enhances the consistency between semantic attention and temporal attention (see Figure 4). Our experiments in section 5.3 show that these two techniques lead to non-trivial performance improvement.

[3] https://github.com/savoirfairelinux/num2words.

[4] https://github.com/languagetool-org/languagetool.

[5] https://spacy.io.

The input to our model are mel-frequency cepstral coefficient (MFCC) audio features (Davis and Mermelstein, 1980) and the output is a sequence of words $\{y_m\}_{m=1}^{M}$, each of which is a symbol from the dictionary. For text representation, we use `fastText` (Bojanowski et al., 2016) trained on the Common Crawl corpus to initialize the word embedding matrix $\mathbf{W}_{emb}$, which is fine-tuned with the model during training. We represent word sequences (*e.g.* attribute words for semantic attention and output words for answer captions) in a distributional space as $\{\mathbf{d}_n\}_{n=1}^{N}$ with $\mathbf{d}_n = \mathbf{W}_{emb}\mathbf{w}_n$ where $\mathbf{w}_n$ is a one-hot vector for n-th word in the word sequence and $\mathbf{d}_n \in \mathbb{R}^{300}$.

4.1 Top-down Multi-scale Encoder

Unlike speech data, sound in the wild is not always continuous. It can be often brief, noisy, occluded, in-the-distance and randomly sparsed throughout the audio. Hence, the lower-level features can be useful to capture such characteristics of natural sound, although they may lack the semantics of the higher-level features. Thus, the joint use of these two levels of features can be mutually beneficial.

The top-down multi-scale encoder takes as input the two-level audio embedding $\{\mathbf{f}_t\}_{t=1}^{T}$, $\{\mathbf{c}_t\}_{t=1}^{T}$ and generates the fused encoding vector, where T is the sequence length of the audio. For input, we use the features from the two layers of the pretrained VGGish network (Hershey et al., 2017): the `fc2` vector $\{\mathbf{f}_t\}_{t=1}^{T}$ as a high-level semantic feature, and the `conv4` vector $\{\mathbf{c}_t\}_{t=1}^{T}$ as a mid-level feature.

The first level of hierarchy encodes high-level features $\{\mathbf{f}_t\}_{t=1}^{T}$ using a bi-directional LSTM. We regard the last hidden state as the global audio embedding $\mathbf{h}^{ctxt} \in \mathbb{R}^{I}$:

$$\overleftrightarrow{\mathbf{h}}_t^{a1} = \text{biLSTM}(\mathbf{f}_t, \overrightarrow{\mathbf{h}}_{t-1}^{a1}, \overleftarrow{\mathbf{h}}_{t+1}^{a1}), \quad (1)$$

$$\mathbf{h}^{ctxt} = \mathbf{W}_c[\overrightarrow{\mathbf{h}}_T^{a1}; \overleftarrow{\mathbf{h}}_1^{a1}] + \mathbf{b}_c, \quad (2)$$

where $\mathbf{W}_c \in \mathbb{R}^{I \times D^1}$ and $\mathbf{b}_c \in \mathbb{R}^{I}$ are parameters, I is the dimension of input to the next layer and D^1 is the dimension of the first layer hidden states.

We then reshape and encode mid-level features $\{\mathbf{c}_t\}_{t=1}^{T} \in \mathbb{R}^{512}$ using another bi-directional LSTM. In order to inject the global semantics, we perform an element-wise addition of $\mathbf{h}^{ctxt}$ to the mid-level feature along the time axis, and feed them into the bi-directional LSTM one at a time,

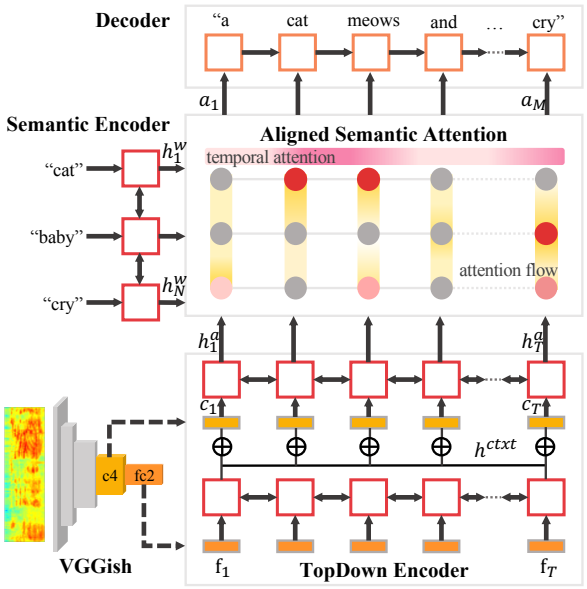


Figure 4: The audio captioning model with *top-down multi-scale encoder* and *aligned semantic attention*.

producing a hidden state $\overleftrightarrow{\mathbf{h}}_t^{a2} \in \mathbb{R}^{D^2}$ at each step:

$$\overleftrightarrow{\mathbf{h}}_t^{a2} = \text{biLSTM}(\mathbf{c}_t + \mathbf{h}^{ctxt}, \overrightarrow{\mathbf{h}}_{t-1}^{a2}, \overleftarrow{\mathbf{h}}_{t+1}^{a2}). \quad (3)$$

4.2 Aligned Semantic Attention

In many captioning models (You et al., 2016; Yu et al., 2017; Laokulrat et al., 2018; Long et al., 2018), semantic attention has been independently used from temporal/spatial attention. However, it can be troublesome because there may exist some discrepancies between the two attentions *i.e.* they do not attend to the same part of the input. For instance, given an audio of a cat meowing and a baby crying, temporal attention may attend to the *crying baby* while semantic attention attends to the word *cat*. We propose a simple yet effective approach that implicitly forces both semantic and temporal/spatial attention to be correctly aligned to one another to maximize the mutual consistency.

For semantic attention, we extract a set of N attribute words for each audio: following You et al. (2016), we retrieve the nearest training audio from the subset of AudioSet and transfer its labels as attribute words. We encode each attribute word vector using a bi-directional LSTM (named *semantic encoder*):

$$\overleftrightarrow{\mathbf{h}}_n^{w} = \text{biLSTM}(\mathbf{d}_n, \overrightarrow{\mathbf{h}}_{n-1}^{w}, \overleftarrow{\mathbf{h}}_{n+1}^{w}), \quad (4)$$

where $\mathbf{d}_n$ is the input text representation of the attribute word sequence. We then align these semantic word features $\overleftrightarrow{\mathbf{h}}_n^{w}$ to the temporal axis of the

audio features $\overleftrightarrow{\mathbf{h}}_t^{a2}$ via the attention flow layer (Seo et al., 2017). For notational simplicity, we omit the bidirectional arrow in the following.

Attention flow layer. We first compute the similarity matrix, $\mathbf{S} \in \mathbb{R}^{T \times N}$ between each pair of audio and word features using the score function $\alpha(\mathbf{h}_t^{a2}, \mathbf{h}_n^w) \in \mathbb{R}$:

$$\alpha(\mathbf{h}_t^{a2}, \mathbf{h}_n^w) = \mathbf{W}_\alpha[\mathbf{h}_t^{a2}; \mathbf{h}_n^w; \mathbf{h}_t^{a2} \circ \mathbf{h}_n^w], \quad (5)$$

$$\mathbf{S}_{tn} = \alpha(\mathbf{h}_t^{a2}, \mathbf{h}_n^w), \quad (6)$$

where $\circ$ is element-wise multiplication.

We then use $\mathbf{S}$ to obtain the attentions and the attended vectors in two directions: word-to-audio $\{\tilde{\mathbf{h}}_t^w\}_{t=1}^T \in \mathbb{R}^{D^2}$ and audio-to-word $\tilde{\mathbf{h}}^{a2} \in \mathbb{R}^{D^2}$:

$$\mathbf{a}_t = \text{softmax}(\mathbf{S}_{t:}), \qquad \tilde{\mathbf{h}}_t^w = \sum_n \mathbf{a}_{tn}\mathbf{h}_n^w, \quad (7)$$

$$\mathbf{b} = \text{softmax}(\max_{row}(\mathbf{S})), \quad \tilde{\mathbf{h}}^{a2} = \sum_t \mathbf{b}_t \mathbf{h}_t^{a2}, \quad (8)$$

where $\mathbf{a}_t \in \mathbb{R}^N$, $\mathbf{b} \in \mathbb{R}^T$.

Lastly, we concatenate them into $\{\mathbf{h}_t^{flow}\}_{t=1}^T \in \mathbb{R}^{4D^2}$, while keeping the temporal axis intact:

$$\mathbf{h}_t^{flow} = [\mathbf{h}_t^{a2}; \tilde{\mathbf{h}}_t^w; \mathbf{h}_t^{a2} \circ \tilde{\mathbf{h}}_t^w; \mathbf{h}_t^{a2} \circ \tilde{\mathbf{h}}^{a2}]. \quad (9)$$

Temporal attention over attention flow. We now have an embedding that aligns the semantic features of words with the time steps of audio features. Subsequently, we apply temporal attention over it; the attention weight is calculated as in Luong et al. (2015). Specifically, we use the global method for each t in $\{\mathbf{h}_t^{flow}\}_{t=1}^T$:

$$\alpha_m = \text{align}(\mathbf{h}_m^{dec}, \mathbf{h}_t^{flow}), \quad (10)$$

$$\mathbf{c}_m = \sum_t \alpha_{mt}\mathbf{h}_t^{flow}, \quad (11)$$

$$\mathbf{a}_m = \tanh(\mathbf{W}_{dec}[\mathbf{c}_m; \mathbf{h}_m^{dec}]), \quad (12)$$

where $\mathbf{h}_m^{dec} \in \mathbb{R}^{D^o}$ is the state of the decoder LSTM, $\mathbf{c}_m \in \mathbb{R}^{4D^2}$ is the context vector, $\alpha_m \in \mathbb{R}^T$ is the attention mask, and $\mathbf{W}_{dec} \in \mathbb{R}^{D^o \times (4D^2 + D^o)}$ is a parameter.

Next, we obtain the output word probability:

$$\mathbf{s}_m = \text{softmax}(\mathbf{W}_o \mathbf{a}_m) \quad (13)$$

where $\mathbf{W}_o \in \mathbb{R}^{V \times D^o}$. Finally, we select the output word as $y_{m+1} = \text{argmax}_{s \in \mathcal{V}}(\mathbf{s}_m)$. We repeat this process until y_{m+1} reaches an EOS token.

The model is trained to maximize the log-likelihood assigned to the target labels via the softmax as done in most captioning models.

5 Evaluation

We perform several quantitative evaluations to provide more insights about our *AudioCaps* dataset. Specifically, our experiments are designed to answer the following questions:

1. Are the collected captions indeed faithful to the audio inputs?
2. Which audio features are useful for audio captioning on our dataset?
3. What techniques can improve the performance of audio captioning?

We present further implementation details and more experimental results in the Appendix. Some resulting audio-caption pairs can be found at https://audiocaps.github.io/supp.

Before presenting the results of our experiments on these three questions, we first explain the experimental setting and baseline models.

5.1 Experimental Setting

Evaluation metrics. Audio captioning can be quantitatively evaluated by the language similarity between the predicted sentences and the ground-truths (GTs) such as BLEU (Papineni et al., 2002), CIDEr (Vedantam et al., 2015), METEOR (Banerjee and Lavie, 2005), ROUGE-L (Lin, 2004) and SPICE (Anderson et al., 2016). In all metrics, higher scores indicate better performance.

Audio features. Audios are resampled to 16kHz, and stereo is converted into mono by averaging both channels. We zero-pad clips that are shorter than 10 seconds and extract three levels of audio features. For the low-level audio feature, the lengthy raw audios are average-pooled by the WaveNet encoder as in Engel et al. (2017). For the mid-level feature, mel-frequency cepstral coefficients (MFCC) (Davis and Mermelstein, 1980) are extracted using `librosa` (McFee et al., 2015) with a window size of 1024, an overlap of 360 and the number of frames at 240, and encoded further with a bi-directional LSTM followed by a gated convolutional encoder (Xu et al., 2018). Lastly, we use two high-level features: the 24th output layer of SoundNet[6] (Aytar et al., 2016) with a (10×1024) dimension and the final output embedding of VGGish[7] (Hershey et al., 2017) with a (10×128) dimension of (time $\times$ embedding).

[6] https://github.com/cvondrick/soundnet.

[7] https://github.com/tensorflow/models/tree/master/research/audioset.

Video features. To contrast with video captioning datasets, we also extract video features at the frame-level and at the sequence-level from YouTube clips. For frame features, we use VGG16 (Simonyan and Zisserman, 2015) pretrained on the ILSVRC-2014 dataset (Russakovsky et al., 2015). For sequence features, we use C3D[8] (Tran et al., 2015) pretrained on the Sport1M dataset (Karpathy et al., 2014). We extract subsequent frames with 50% overlap centered at each time step on the input clips for AudioSet videos, while proceeding with no overlap for MSR-VTT clips as in the original paper. We sample videos at 25fps.

5.2 Baselines

Retrieval methods. As straightforward baselines, we test the 1-nearest search with audio features, denoted by `1NN-MFCC`, `1NN-SoundNet` and `1NN-VGGish`. For a query audio, we find its closest training audio using the ℓ_2 distance on the features and return its text as a prediction. We mean-pool all the audio features over time, because it empirically leads to a strong performance.

LSTM methods. As simple generative baselines, we test with the LSTM decoder, denoted by `-LSTM` postfix, where the encoded audio feature is set as the initial state of the LSTM. For instance, `WaveNet-LSTM` is the model with the WaveNet encoder and the LSTM decoder. We use a single-layer LSTM with dropout (Srivastava et al., 2014) and layer normalization (Ba et al., 2016).

Attention models. We test two popular attention models developed in video captioning research: (i) `TempAtt` (Luong et al., 2015; Yao et al., 2016) generates captions by selectively attending to audio features over time, and (ii) `SemAtt` (You et al., 2016) creates text attending to attribute words as secondary information.

Our models. We denote our top-down multi-scale encoder as the prefix `TopDown-` and aligned semantic attention as `AlignedAtt-`.

Upper-bounds. Given that each test data has five human-generated captions, we perform cross validation on the five GT captions as an upper-bound of performance denoted as `Human`. We regard one of five human annotations as model prediction and compute the performance metric with the other four as ground-truths. After doing this on each of five, we then average the scores.

[8] https://github.com/facebook/C3D.

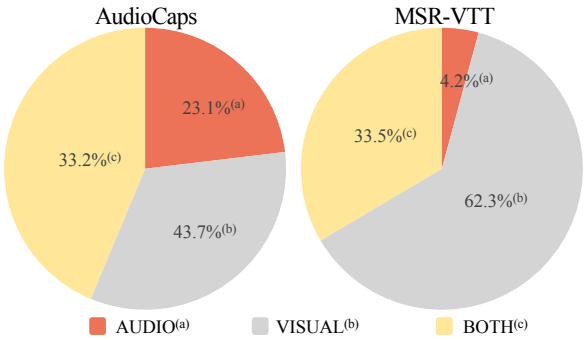


Figure 5: Comparison of vocabulary tag distribution between AudioCaps and MSR-VTT.

5.3 Results

We discuss experimental results in response to the three questions regarding the AudioCaps dataset.

5.3.1 Audio *vs* Video Captioning

We first evaluate whether the collected audio-based captions are indeed loyal to the audio clips. As one possible method to validate it, we perform comparative experiments with the video-oriented MSR-VTT dataset (Xu et al., 2016). Note that MSR-VTT and AudioCaps both provide pairs of audio clips and its corresponding videos, allowing us to perform this comparative study. We hypothesize that the captions from MSR-VTT would not coherently map to audio features, because they are written mainly based on the visual information. In contrast, AudioCaps captions would be better aligned to audio features than visual features.

The results in Table 4 support our hypothesis. In MSR-VTT, the video-based captioning model `C3D-LSTM` attains better scores than the preceding three audio-captioning models `*-LSTM`, while in AudioCaps the video-based model performs far worse than the audio models. This may be due to our collection method of AudioCaps, which encourages turkers to submit the descriptions based on the audio rather than the visual.

Vocabulary comparison. We also make comparisons between AudioCaps and MSR-VTT in terms of vocabulary usage in the captions. We select the 1,800 most frequent vocabularies of verbs, adjectives and adverbs from each dataset, and run a user study in which three different workers are asked to categorize each sampled word into one of (*Audio, Visual, Both, Not Applicable*). The category label per word is decided by a majority vote of three workers' opinions. We use AMT once more to collect the unbiased opinions. In or-

Methods	B-1	B-2	B-3	B-4	METEOR	CIDEr	ROUGE-L	SPICE
1NN-MFCC	34.1	17.8	10.0	5.3	9.9	8.7	23.4	4.7
1NN-SoundNet (Aytar et al., 2016)	39.1	22.0	12.9	7.6	12.0	16.4	27.2	6.9
1NN-VGGish (Hershey et al., 2017)	44.2	26.5	15.8	9.0	15.1	25.2	31.2	9.2
WaveNet-LSTM (Engel et al., 2017)	48.9	31.5	20.2	13.0	13.8	29.6	35.5	9.0
MFCC-LSTM (Xu et al., 2018)	57.3	40.0	26.8	16.4	18.4	44.8	41.1	11.5
SoundNet-LSTM (Aytar et al., 2016)	54.0	38.0	26.4	17.6	16.5	43.2	39.2	10.8
VGGish-LSTM (Hershey et al., 2017)	58.7	42.3	29.8	20.4	18.7	50.4	42.6	13.0
TempAtt-WaveNet-LSTM (Luong et al., 2015)	50.7	34.3	22.9	14.8	14.8	28.2	36.4	8.6
TempAtt-MFCC-LSTM (Luong et al., 2015)	57.7	40.7	27.6	17.9	18.2	49.3	41.8	12.4
TempAtt-SoundNet-LSTM (Luong et al., 2015)	55.5	37.4	24.8	15.8	17.0	43.4	40.0	11.6
TempAtt-VGGish(FC2)-LSTM (Luong et al., 2015)	61.3	43.2	29.6	19.5	19.3	50.9	43.5	13.5
TempAtt-VGGish(C4)-LSTM (Luong et al., 2015)	61.8	44.5	30.7	20.4	19.4	55.3	44.0	13.2
TempAtt-VGGish(C3)-LSTM (Luong et al., 2015)	61.2	44.1	30.3	20.9	19.0	52.3	43.7	13.0
TopDown-VGGish(FC2,C4)-LSTM	62.9	45.1	31.5	21.4	19.9	57.7	44.8	14.3
TopDown-VGGish(FC2,C4,C3)-LSTM	60.9	43.7	30.7	20.8	20.0	55.8	43.7	13.6
TopDown-SemTempAtt(1NN) (You et al., 2016)	62.2	44.9	31.3	20.9	20.2	58.1	44.9	13.6
TopDown-AlignedAtt(1NN)	61.4	44.6	**31.7**	**21.9**	**20.3**	**59.3**	**45.0**	**14.4**
Human	65.4	48.9	37.3	29.1	28.8	91.3	49.6	21.6

Table 2: Captioning results of different methods on *AudioCaps* measured by language similarity metrics.

Methods	B-1	B-2	B-3	B-4	METEOR	CIDEr	ROUGE-L	SPICE
SemTempAtt(1NN)-VGGish-LSTM (You et al., 2016)	62.2	44.5	31.0	20.5	19.3	52.5	44.0	13.7
AlignedAtt(1NN)-VGGish-LSTM	62.0	45.1	32.0	21.6	19.6	56.1	44.4	13.5
SemTempAtt(GT)-VGGish-LSTM (You et al., 2016)	67.0	50.3	36.4	24.8	22.5	72.0	48.3	16.3
AlignedAtt(GT)-VGGish-LSTM	69.1	52.3	38.0	26.1	23.6	77.7	49.6	17.2

Table 3: Upper-bound of *aligned semantic attention* by language similarity metrics.

	MSR-VTT		**AudioCaps**	
Methods	METEOR	CIDEr	METEOR	CIDEr
MFCC-LSTM	21.4	19.2	18.2	49.3
SoundNet-LSTM	20.0	14.7	17.0	43.4
VGGish-LSTM	22.8	26.1	19.3	50.9
C3D-LSTM	24.8	36.8	15.9	42.7
Gap (Audio - Video)	-2.0	-10.7	**+3.4**	**+8.2**

Table 4: Comparison of captioning results between video-based and audio-based datasets. The first three methods perform captioning using only audios while the last method C3D-LSTM, only use videos. The gaps empirically show how much *AudioCaps* is audio-oriented in contrast to MSR-VTT.

der to guarantee thoughtful submissions, we ask the workers to provide a description using the word. We compensate $0.05 per word to English-speaking workers with a 95% approval rate.

Figure 5 shows that AudioCaps has more vocabularies tagged as *Audio* (*e.g. neighs, rustling*) by 18.9%p more than MSR-VTT. Furthermore, 56.3% of the total vocabularies in AudioCaps are categorized as audio-related, that is, labeled as *Audio* or *Both* (*e.g. vibrating, applauds*). Hence, this vocabulary comparison result reassures that AudioCaps is more audio-oriented than MSR-VTT.

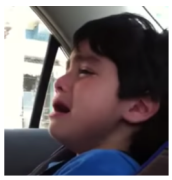
5.3.2 Comparison of Audio Features

The methods in the second group of Table 2 are compared to investigate which audio features are more suitable for captioning on AudioCaps. The best results are obtained by VGGish-LSTM. This may be because VGGish is pretrained on YouTube audio clips, similar to AudioCaps. Although the topics of YouTube are extremely diverse, the domain proximity may help VGGish learn more utilizable features for AudioCaps. SoundNet-LSTM shows inferior performance compared to VGGish-LSTM, one possible reason being because it is pretrained with Flickr videos, which are rather distant in domain from the source of our dataset, in terms of topic diversity and the amount of possible noise. MFCC-LSTM does not perform as well as VGGish-LSTM, even with the similar convolutional recurrent encoder. This result hints that pretraining with a proper dataset is essential for audio captioning. A comparison between MFCC-LSTM and WaveNet-LSTM reveals that using MFCC is better than directly taking raw waveform as input. The raw waveform is relatively long (>500× longer than MFCC); hence, it may pose a difficulty for RNN-based encoders to precisely represent the whole audio context.

5.3.3 Comparison of Models

Temporal attention consistently boosts the captioning performance of the LSTM decoder in all audio features, as shown in the models with TempAtt- prefix in Table 2. No-

(Ours) a large explosion followed by a loud pop
(1NN-VGGish) a man speaking followed by a loud bang
(SemTempAtt) a large explosion followed by a loud splash and thunder
(GT) a whooshing noise followed by an explosion



(Ours) a man and woman talking, then a baby crying
(1NN-VGGish) an engine runs, and people speak
(SemTempAtt) a dog barks and a woman laughs
(GT) a child cries followed by a man and woman speaking and then the child resumes sobbing

(Ours) a small motor is running, whirring occurs, and a high-pitched whine is present
(1NN-VGGish) a drill is operated, then a man speaks and restarts the drill
(SemTempAtt) a small motor running and a man speaking
(GT) a drone whirring followed by a crashing sound

(Ours) a truck engine is running, a siren is occurring, and an adult male speaks
(1NN-VGGish) a loud fire engine is followed by an emergency siren
(SemTempAtt) emergency sirens and a siren blaring
(GT) a large engine passes as people speak followed by a siren

Figure 6: Four examples of audio captioning with captured video frames, grouthtruths (GT), and generated captions by our method (Ours) and baselines. They can be heard at https://audiocaps.github.io/supp.

tably, a large performance gain is observed for `TempAtt-MFCC-LSTM`. This may be because MFCC features are transformed to temporally longer features than SoundNet and VGGish features ($240 > 10$), and thus allow temporal attention to better aid the model and bypass the vanishing gradient problem.

The semantic attention is also favorable for captioning performance, as `SemTempAtt(1NN)-VGGish-LSTM` in Table 3 slightly outperforms `TempAtt-VGGish(FC2)-LSTM` in Table 2. That is, the additional use of semantic attention enhances the temporal attention model. Obviously, when using GT labels instead of 1NN retrieved labels as attribute words, the performance increases much, hinting that better semantic attributes are more synergetic with the aligned attention.

The comparison between different layers (C4, C3, FC2) confirms the effectiveness of jointly using multi-level features. The fused features by the top-down multi-scale encoder (*i.e.* `TopDown-`) prove the most beneficial as they outperform their counterparts in Table 2. However, a stack of (FC2,C4) layers performs the best, while the three layer stack is slightly inferior, presumably due to overfitting and weak information flow between the upper and lower levels of the stacks. Finally, our best performing model is `TopDown-AlignedAtt` where both the top-down multi-scale encoder and aligned semantic attention are jointly used. We postulate that the two techniques synergize well thanks to rich information provided by `TopDown` allowing for better attention alignment.

5.3.4 Captioning Examples

Figure 6 shows selected examples of audio captioning. In each set, we show a video frame, GT and text descriptions generated by our method and baselines. Many audio clips consist of sounds with multiple sources in sequence, for which baselines often omit some details or mistakenly order the event sequence, whereas our model is better at capturing the details in the correct order.

6 Conclusion

We addressed a new problem of audio captioning for sound in the wild. Via Amazon Mechanical Turk, we contributed a large-scale dataset named *AudioCaps*, consisting of 46K pairs of audio clips and human-written text. In our experiments, we showed that the collected captions were indeed faithful to the audio inputs as well as improve the captions by two newly proposed components: the top-down multi-scale encoder and aligned semantic attention.

There are several possible directions beyond this work. First, we can further expand the scope of *AudioCaps*. Second, our model is integrable with speech counterparts to achieve more complete auditory captioning tasks.

Acknowledgments

We would like to thank SNU Vision & Learning Lab members and Yunseok Jang for the helpful comments and discussions. This work is supported by Kakao and Kakao Brain corporations and the international cooperation program by the NRF of Korea (NRF-2018K2A9A2A11080927). Gunhee Kim is the corresponding author.

References

Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. 2018. The Conversation: Deep Audio-Visual Speech Enhancement. In *Interspeech*.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. SPICE: Semantic Propositional Image Caption Evaluation. In *ECCV*.

Relja Arandjelovic and Andrew Zisserman. 2018. Objects that Sound. In *ECCV*.

Yusuf Aytar, Carl Vondrick, and Antonio Torralba. 2016. SoundNet: Learning Sound Representations from Unlabeled Video. In *NIPS*.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer Normalization. In *Stat*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *ACL Workshop MTSumm*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching Word Vectors with Subword Information. In *TACL*.

Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. 2018. VoxCeleb2: Deep Speaker Recognition. In *Interspeech*.

Steven B Davis and Paul Mermelstein. 1980. Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences. In *TASSP*.

Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan. 2017. Neural Audio Synthesis of Musical Notes with WaveNet Autoencoders. In *ICML*.

Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T Freeman, and Michael Rubinstein. 2018. Looking to Listen at the Cocktail Party: A Speaker-Independent Audio-Visual Model for Speech Separation. In *SIGGRAPH*.

Eduardo Fonseca, Jordi Pons Puig, Xavier Favory, Frederic Font Corbera, Dmitry Bogdanov, Andres Ferraro, Sergio Oramas, Alastair Porter, and Xavier Serra. 2017. Freesound Datasets: A Platform for the Creation of Open Audio Datasets. In *ISMIR*.

Ruohan Gao, Rogerio Feris, and Kristen Grauman. 2018. Learning to Separate Object Sounds by Watching Unlabeled Video. In *ECCV*.

Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. 2017. Audio Set: An Ontology and Human-labeled Dataset for Audio Events. In *ICASSP*.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *AISTATS*.

Sergio Guadarrama, Niveda Krishnamoorthy, Girish Malkarnenkar, Subhashini Venugopalan, Raymond Mooney, Trevor Darrell, and Kate Saenko. 2013. YouTube2Text: Recognizing and Describing Arbitrary Activities Using Semantic Hierarchies and Zero-Shot Recognition. In *ICCV*.

John R Hershey, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe. 2016. Deep Clustering: Discriminative Embeddings for Segmentation and Separation. In *ICASSP*.

Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al. 2017. CNN Architectures for Large-Scale Audio Classification. In *ICASSP*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*.

Chiori Hori, Huda Alamri, Jue Wang, Gordon Wichern, Takaaki Hori, Anoop Cherian, Tim K. Marks, Vincent Cartillier, Raphael Gontijo Lopes, Abhishek Das, Irfan Essa, Dhruv Batra, and Devi Parikh. 2018. End-to-End Audio Visual Scene-Aware Dialog using Multimodal Attention-based Video Features. In *arXiv:1806.08409*.

Chiori Hori, Takaaki Hori, Teng-Yok Lee, Ziming Zhang, Bret Harsham, John R Hershey, Tim K Marks, and Kazuhiko Sumi. 2017. Attention-based Multimodal Fusion for Video Description. In *ICCV*.

Keith Ito. 2017. The LJ Speech Dataset. https://keithito.com/LJ-Speech-Dataset/.

Justin Johnson, Andrej Karpathy, and Li Fei-Fei. 2016. DenseCap: Fully Convolutional Localization Networks for Dense Captioning. In *CVPR*.

Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. 2014. Large-Scale Video Classification with Convolutional Neural Networks. In *CVPR*.

Diederik Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017. Dense-Captioning Events in Videos. In *ICCV*.

Natsuda Laokulrat, Naoaki Okazaki, and Hideki Nakayama. 2018. Incorporating Semantic Attention in Video Description Generation. In *LREC*.

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *TSBO*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft COCO: Common objects in Context. In *ECCV*.

Xiang Long, Chuang Gan, and Gerard de Melo. 2018. Video Captioning with Multi-Faceted Attention. *TACL*.

Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. In *EMNLP*.

Brian McFee, Colin Raffel, Dawen Liang, Daniel Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. 2015. librosa: Audio and Music Signal Analysis in Python. In *SCIPY*.

Annamaria Mesaros, Toni Heittola, Aleksandr Diment, Benjamin Elizalde, Ankit Shah, Emmanuel Vincent, Bhiksha Raj, and Tuomas Virtanen. 2017. DCASE 2017 Challenge Setup: Tasks, Datasets and Baseline System. In *DCASE*.

Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. 2016. TUT Database for Acoustic Scene Classification and Sound Event Detection. In *EUSIPCO*.

Mozilla. 2017. Mozilla Common Voice. https://voice.mozilla.org/.

Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. 2017. VoxCeleb: A Large-Scale Speaker Identification Dataset. In *Interspeech*.

Andrew Owens and Alexei A. Efros. 2018. Audio-Visual Scene Analysis with Self-Supervised Multisensory Features. In *ECCV*.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An ASR Corpus Based on Public Domain Audio Books. In *ICASSP*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *ACL*.

Karol J Piczak. 2015. ESC: Dataset for Environmental Sound Classification. In *ACM MM*.

Cyrus Rashtchian, Peter Young, Micah Hodosh, and Julia Hockenmaier. 2010. Collecting Image Annotations Using Amazon's Mechanical Turk. In *NAACL-HLT*.

Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Christopher Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele. 2017. Movie Description. In *IJCV*.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *IJCV*.

Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. 2014. A Dataset and Taxonomy for Urban Sound Research. In *ACM MM*.

Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon. 2018. Learning to Localize Sound Source in Visual Scenes. In *CVPR*.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional Attention Flow for Machine Comprehension. In *ICLR*.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning. In *ACL*.

Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *ICLR*.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. In *JMLR*.

Dan Stowell, Dimitrios Giannoulis, Emmanouil Benetos, Mathieu Lagrange, and Mark D Plumbley. 2015. Detection and Classification of Acoustic Scenes and Events. In *IEEE Transactions on Multimedia*.

Du Tran, Lubomir D Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning Spatiotemporal Features with 3D Convolutional Networks. In *ICCV*.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. CIDEr: Consensus-based Image Description Evaluation. In *CVPR*.

Xin Wang, Yuan-Fang Wang, and William Yang Wang. 2018. Watch, Listen, and Describe: Globally and Locally Aligned Cross-Modal Attentions for Video Captioning. In *NAACL-HLT*.

Pete Warden. 2018. Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition. In *arXiv:1804.03209*.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. In *CVPR*.

Yong Xu, Qiuqiang Kong, Wenwu Wang, and Mark D Plumbley. 2018. Large-Scale Weakly Supervised Audio Classification using Gated Convolutional Neural Network. In *ICASSP*.

Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. 2016. Describing Videos by Exploiting Temporal Structure. In *ICCV*.

Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. 2016. Image Captioning with Semantic Attention. In *CVPR*.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From Image Descriptions to Visual Denotations: New Similarity Metrics for Semantic Inference over Event Descriptions. In *TACL*.

Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. 2017. End-to-End Concept Word Detection for Video Captioning, Retrieval, and Question Answering. In *CVPR*.

Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. 2018. The Sound of Pixels. In *ECCV*.

Appendix

In the supplemental material, we enlist the following which may shed further insights:

- Additional related work [section A]
- Additional dataset analysis [section B]
- Training Details [section C]

A Related Work

Audio-Visual correspondence. Over the past year, a great interest has been shone to the interconnection of auditory and visual senses. The task of localizing the sound source within the visual input have been actively explored (Nagrani et al., 2017; Chung et al., 2018; Senocak et al., 2018; Afouras et al., 2018; Gao et al., 2018; Arandjelovic and Zisserman, 2018; Zhao et al., 2018), along with blind source separation aided by visual features (Ephrat et al., 2018) and learning of audio-visual multisensory representation (Owens and Efros, 2018). These previous studies compensate the lack of information in the auditory input with visual information, whereas this work focuses solely on the auditory input to generate informative descriptions.

B Dataset

The full ontology of selected labels is outlined in Figure 7.

Figure 8 shows the number of clips per word label. The original AudioSet has an extreme label bias. For instance, a difference of 660,282 between the average of top 3 most common and average of top 3 most uncommon classes. Whereas our dataset at the moment has a difference of 971. Notice the label bias is significantly reduced in comparison to the original AudioSet. We plan to reduce this further in the upcoming releases.

Table 5 compares our audio captioning dataset with some representative benchmarks of video captioning: MSR-VTT (Xu et al., 2016) and LSMDC (Rohrbach et al., 2017). One interesting property of our dataset is that the portion of verbs in the vocabularies are larger than the others. This may imply that the captions describe what is *happening* rather than what *is* in the content.

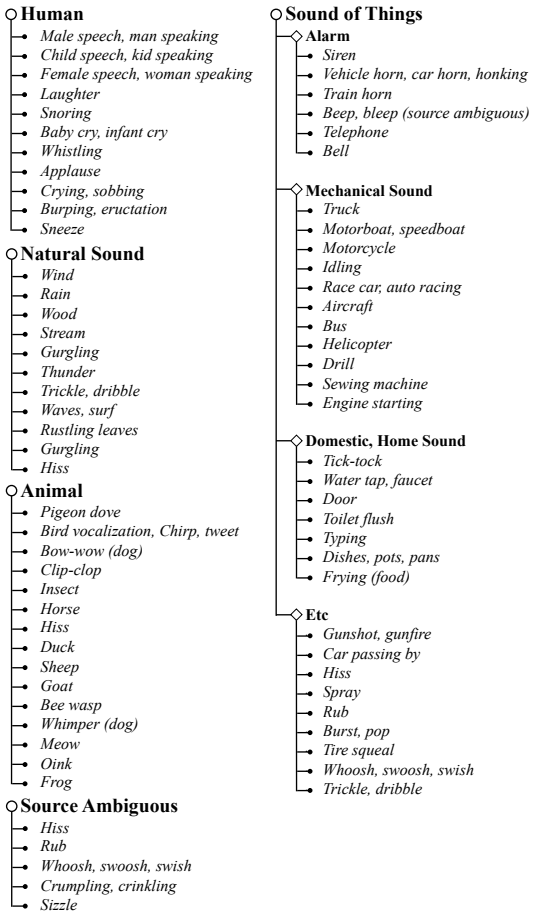


Figure 7: The curated ontology for *AudioCaps* on the basis of *AudioSet*.

C Training Details

All the parameters are initialized with Xavier method (Glorot and Bengio, 2010). We apply the Adam optimizer (Kingma and Ba, 2015) with $\beta_1 = 0.9, \beta_2 = 0.999$ and $\epsilon = 1e - 8$.

Dataset	Clips	Sentences	Unique clips	Tokens	Vocabs	Nouns	Verbs	Adjectives	Adverbs	Duration(h)
MSR-VTT	10,000	200,000	7,180	1,856,523	29,316	16,437	6,379	3,761	872	41.2
LSMDC	128,085	128,118	200	1,157,155	22,500	12,181	3,394	5,633	1,292	147
AudioCaps	39,106	43,022	39,106	567,927	4,506	2,747	1,825	766	353	108.6

Table 5: Comparison of *AudioCaps* with MSR-VTT (Xu et al., 2016), LSMDC (Rohrbach et al., 2017).

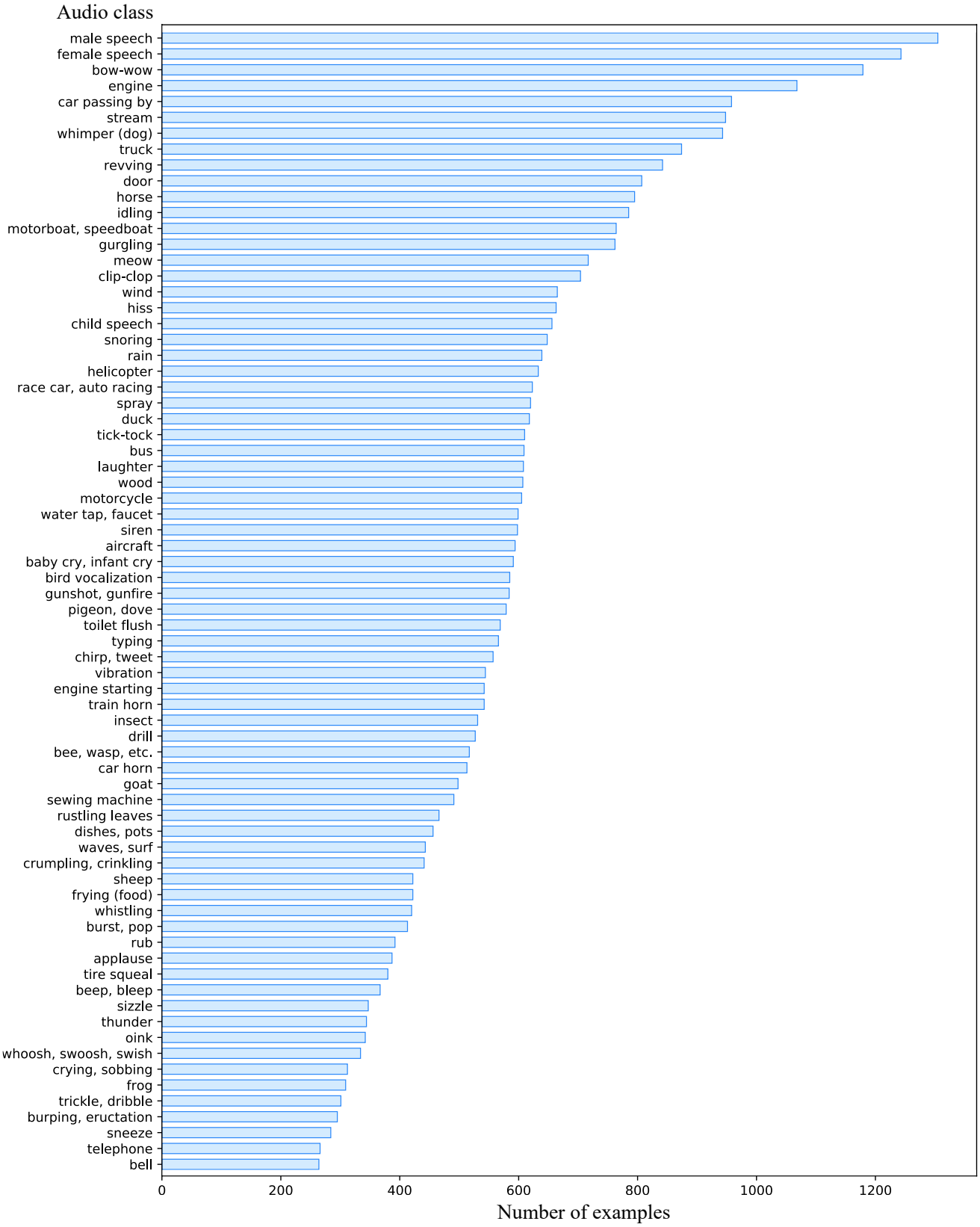


Figure 8: The frequencies of annotated instances per category (*i.e.* word labels) for *AudioCaps*.

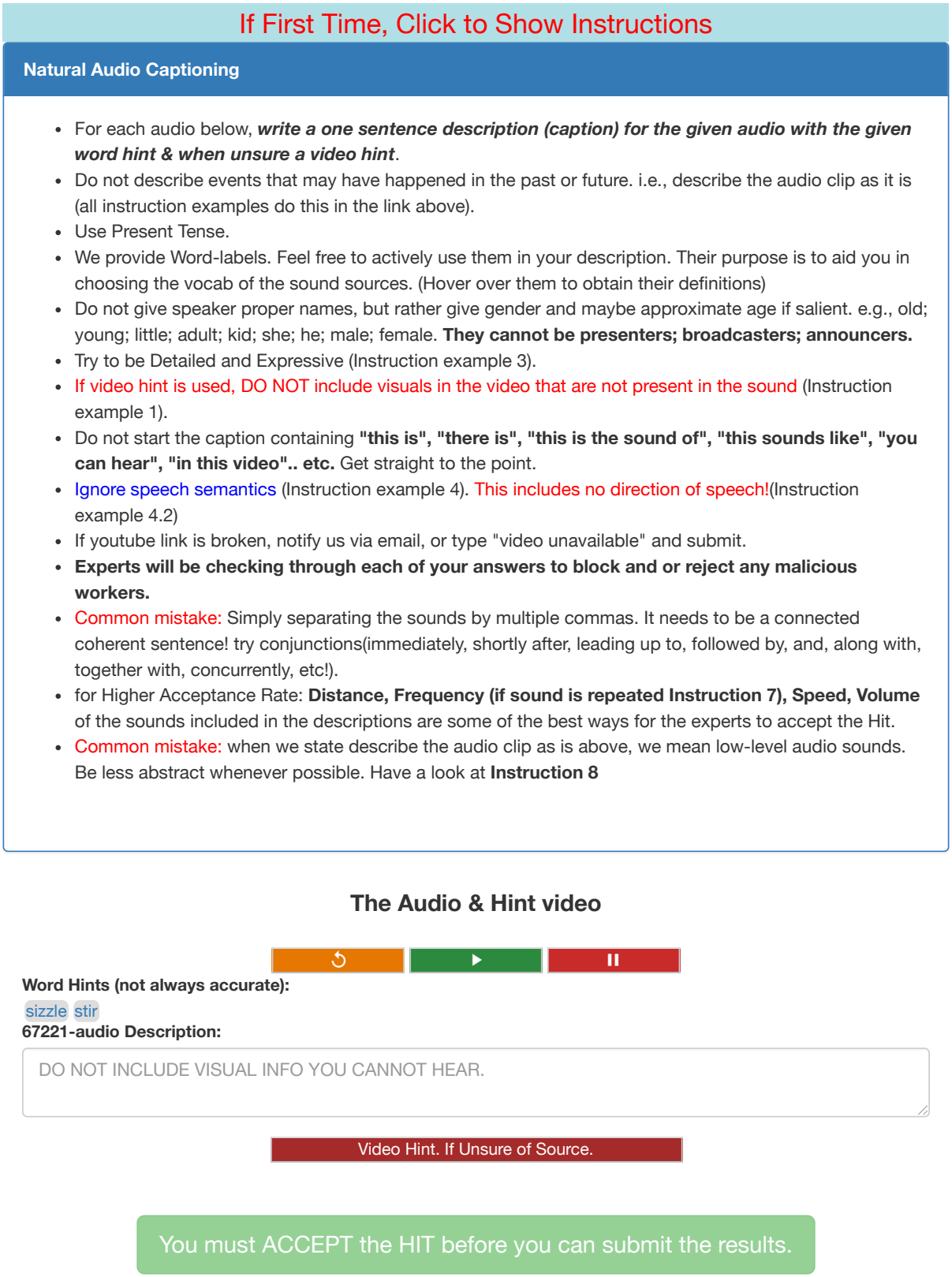


Figure 9: The AMT interface for sentence annotation with instructions.