

Towards Weakly Supervised Text-to-Audio Grounding

Xuenan Xu, *Student Member, IEEE*, Ziyang Ma, *Student Member, IEEE*, Mengyue Wu, *Member, IEEE* and Kai Yu, *Senior Member, IEEE*

Abstract—Text-to-audio grounding (TAG) task aims to predict the onsets and offsets of sound events described by natural language. This task can facilitate applications such as multimodal information retrieval. This paper focuses on weakly-supervised text-to-audio grounding (WSTAG), where frame-level annotations of sound events are unavailable, and only the caption of a whole audio clip can be utilized for training. WSTAG is superior to strongly-supervised approaches in its scalability to large audio-text datasets. Two WSTAG frameworks are studied in this paper: sentence-level and phrase-level. First, we analyze the limitations of mean pooling used in the previous WSTAG approach and investigate the effects of different pooling strategies. We then propose phrase-level WSTAG to use matching labels between audio clips and phrases for training. Advanced negative sampling strategies and self-supervision are proposed to enhance the accuracy of the weak labels and provide pseudo strong labels. Experimental results show that our system significantly outperforms previous WSTAG methods. Finally, we conduct extensive experiments to analyze the effects of several factors on phrase-level WSTAG. The code and models are available at <https://github.com/wsntxxn/TextToAudioGrounding>.

Index Terms—text-to-audio grounding, weakly-supervised learning, negative sampling, audio-text representation, clustering

I. INTRODUCTION

WITH the development of deep learning and the accessibility of large-scale datasets, audio understanding has achieved remarkable success. Many works focus on audio understanding tasks such as Acoustic Scene Classification (ASC) [1] and Sound Event Detection (SED) [2], [3], where audio recordings are classified into categories in a closed set, e.g., speech, music. However, such closed-set classification systems cannot handle complicated requirements like detecting the third beeping sound in an audio recording. We proposed text-to-audio grounding (TAG) [4] to overcome the limitations of SED systems, since TAG aims to detect sound events described by natural language queries. TAG can be potentially useful in human-machine interaction applications and cross-modal retrieval systems. It has also contributed to captioning evaluation [5]. In computer vision, visual grounding [6], [7] is a fundamental task bridging vision and language. It is a building block of grounded cross-modal tasks like grounded captioning [8] and has been extensively investigated [9]–[12]. However, as a similar task in the audio domain, TAG has not received as much attention. In this paper, we address the TAG task to fill in this gap.

TAG can be trained in two paradigms: strongly-supervised TAG (SSTAG) and weakly-supervised TAG (WSTAG). SSTAG uses strong annotations, providing the onsets and

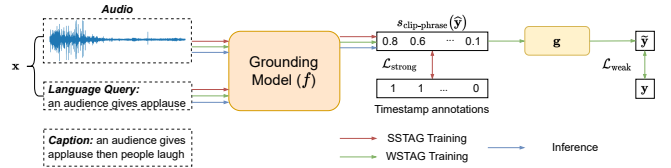


Fig. 1. Comparison between SSTAG and WSTAG.

offsets of queried events during training. By contrast, WSTAG only has access to the corresponding caption of each audio while timestamp annotations are unavailable. Figure 1 illustrates the difference between the two paradigms. Since frame-level supervision signals are provided during SSTAG training, it can achieve better performance than WSTAG. However, manual strong labeling is expensive and time-consuming, limiting the scalability of SSTAG. WSTAG can be applied to general audio captioning datasets [13], [14]. This paper focuses on WSTAG for its scalability.

Here we give a high-level description of weakly-supervised learning to highlight the challenges of WSTAG. In weakly-supervised learning, the model is trained to predict the desired \hat{y} from the input x . However, only pairs of (x, y) are available during training, where y is the label. Therefore during training \hat{y} is fed to an extra module g to obtain \tilde{y} so that the model can be trained by minimizing \tilde{y} and y . During inference, the extra module is no longer needed. The process can be formulated as the following.

Training:

$$\begin{aligned} \hat{y} &= f(x), & \tilde{y} &= g(\hat{y}) \\ \ell &= \mathcal{L}(y, \tilde{y}) \end{aligned} \quad (1)$$

Inference:

$$\hat{y} = f(x) \quad (2)$$

Specifically, in WSTAG, x is (audio, query) while \hat{y} is the similarity s_{fp} between each audio frame and phrase query. Previous WSTAG works [15] used the sentence-level audio-caption correspondence as y , which we refer to as **sentence-level** WSTAG. g includes two steps: pooling along the audio frames to obtain the clip-phrase similarity s_{cp} from s_{fp} and pooling along phrases to obtain the clip-sentence similarity s_{cs} . In previous works [15] mean pooling was used in both steps. However, this approach violates the multiple instance learning (SMI) assumption of audio description: s_{cp} is high if the phrase occurs in at least one frame, i.e., at least one s_{fp} is high. According to the definition of weakly-supervised

learning, g plays a crucial role since it bridges \hat{y} and y . Specifically, in sentence-level WSTAG, the two pooling steps are critical since they bridge the desired fine-grained *frame-phrase* correspondence and the coarse but available *clip-caption* correspondence. Therefore, in this paper, we explore several pooling strategies and show their influence on sentence-level WSTAG performance.

Furthermore, the training/test mismatch in sentence-level WSTAG hinders performance. The two pooling steps in g are employed to bridge two natural levels of mismatch: clip/frame in audio and sentence/phrase in text. In this paper, we attempt to narrow down the mismatch to one level by eliminating the sentence/phrase mismatch. The phrase-level audio-phrase correspondence is used as y . g includes only audio pooling. We refer to this paradigm as *phrase-level* WSTAG. In phrase-level WSTAG, y is 1 for positive audio-phrase pairs while 0 for negative ones. “Positive/negative” indicates the sound event described by the phrase is present/absent in the audio clip. To further improve phrase-level WSTAG, we propose two techniques: 1) advanced negative sampling strategies, including similarity-based and clustering-based, to enhance the accuracy of y ; 2) self-supervision, where a pre-trained WSTAG model is utilized to refine y and provide pseudo labels for s_{fp} . The proposed phrase-level WSTAG, along with the two techniques, brings significant improvement over the sentence-level WSTAG baseline.

In the conference version [16], we analyze the influence of several pooling strategies in sentence-level WSTAG where text pooling is used in both training and inference. This paper uses a simplified sentence-level framework where text pooling is not needed during inference and compares the effects of pooling strategies in the new framework. Additionally, we extend our work to a new phrase-level WSTAG framework with advanced techniques and analysis. Our contributions are summarized as follows:

- We comprehensively analyze the unique challenge faced in WSTAG: the granularity discrepancy in cross-modal alignment between training and test, and improve WSTAG performance accordingly.
- Based on the analysis, we improve sentence-level WSTAG to phrase-level WSTAG to narrow down the training/test discrepancy from the textual modality.
- We propose two techniques to improve phrase-level WSTAG, where advanced sampling strategies provide more accurate weak labels and self-supervision further narrows down the discrepancy from the audio modality.
- Experiments show our model achieves state-of-the-art (SOTA) WSTAG performance, with comparable performance to SSTAG methods, and generalizes well to SED datasets.

II. RELATED WORK

In this section, we briefly introduce works related to WSTAG from three themes: *weakly-supervised visual grounding*, *weakly-supervised sound event detection*, and *audio-centric text representation learning*.

A. Weakly-supervised Visual Grounding

Visual grounding is analogous to audio grounding. It includes two types of tasks: visual object grounding [6], [17], [18], and video moment localization [7], [19], [20]. The former requires localizing the objects described by natural language in an image or a video spatially while the latter requires localizing the event described by natural language in a video temporally. Similar to TAG, high-quality strongly-annotated data with bounding boxes or timestamps is scarce due to expensive labor costs. Therefore, several works explore weakly-supervised visual grounding approaches [9], [10], [21] using sentence-level or phrase-level weak labels. Some works use the sentence-level image/video-sentence correspondence to align the regions in images (or segments in videos) and phrases in sentences by contrastive learning [22]–[24], stimulating a similar setting in WSTAG as described in Section III. Other works explore using the phrase-level supervision signal for training by reconstructing the phrase queries [25], [26] or discriminating negative queries from positive ones [27]. In these works, the phrase-level image/video-phrase correspondence is taken as the training label. Therefore, phrase-level approaches use more fine-grained visual-text correspondence for training compared with sentence-level ones. Inspired by these works, we propose phrase-level WSTAG in Section IV.

B. Weakly-supervised Sound Event Detection

Weakly-supervised learning has attracted much attention for its potential application on large-scale weakly-annotated data [28]–[30]. In the field of SED, weakly-supervised sound event detection (WSSSED) has also been explored extensively [31]–[34]. It shares a similar goal with WSTAG in detecting onsets and offsets of sound events. However, in WSSSED the sound events are from a pre-defined class set instead of natural language descriptions. The weakly-annotated data provide only the presence or absence of sound events but the exact timestamps are unavailable. According to the weakly-supervised learning paradigm, \hat{y} and \tilde{y} are the estimated frame-level and clip-level event probabilities, respectively. Current WSSSED approaches use neural networks to predict \hat{y} and then pool it along the temporal axis to obtain \tilde{y} . In this work, we adopt the convolutional recurrent neural network (CRNN), which is a popular backbone in WSSSED, as the WSTAG audio encoder. We also investigate the effectiveness of several WSSSED temporal pooling strategies in WSTAG. It should be noted that due to the fixed class set, in WSSSED negative (absent) event classes are in fact provided for each data sample. In contrast, in WSTAG there is not such an explicit class set so we need to sample negative phrases, i.e., phrases that do not occur in the audio, for each audio sample.

C. Audio-centric Text Representation Learning

As a cross-modal task, TAG’s performance highly relies on efficient representation learning. With the development of Transformer-based architectures, large-scale unlabeled data are exploited to extract efficient text [35]–[37], speech [38]–[41] and visual [42], [43] representations. Deep Transformers

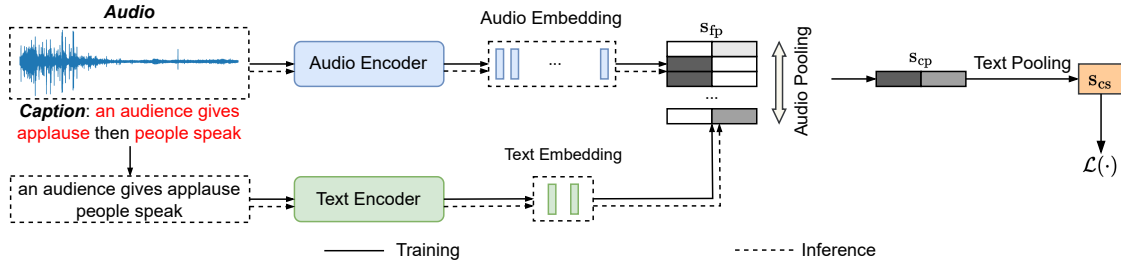


Fig. 2. Sentence-level WSTAG. For an audio-caption pair, the frame-phrase similarities s_{fp} are calculated. During training, audio pooling and text pooling transform them into the clip-sentence similarity s_{cs} for loss calculation. During inference, s_{fp} are taken as outputs.

are trained by masked language modeling (MLM) on the text data, significantly enhancing the performance of downstream language understanding tasks. Moreover, researchers also investigate learning audio-centric text representation [44] to facilitate audio-text cross-modal tasks like audio retrieval. Recently, contrastive language-audio pre-training (CLAP) has been proposed to extract robust audio and text embeddings from audio-text datasets [45]–[47]. With the large-scale audio-text pre-training, the text encoder is able to extract audio-centric representations from the text. Phrases describing the same or acoustically similar events are also close in the text embedding space. In this work, we use the text representations learned by contrastive pre-training to sample negative phrases in the phrase-level WSTAG.

III. SENTENCE-LEVEL WEAKLY-SUPERVISED TEXT-TO-AUDIO GROUNDING

In this section, we first formulate sentence-level WSTAG since it is the basic form of WSTAG and serve as a natural form to investigate the bridge between \hat{y} and y via audio and text pooling strategies (see Equation (1)). Then we elaborate pooling strategies with regards to their implications in aligning audio and text.

A. Framework

Sentence-level WSTAG uses sentence-level supervision signals for training. \hat{y} in Equation (1) is the clip-sentence similarity s_{cs} and y is the ground truth clip-sentence correspondence. As Figure 2 shows, for an audio-caption pair $(\mathcal{A}, \mathcal{T})$, two embedding sequences $\{\mathbf{a}_t\}_{t=1}^T$ and $\{\mathbf{t}_n\}_{n=1}^N$ are obtained. The similarity between the t -th audio frame and the n -th phrase is calculated as:

$$s_{fp}(t, n) = \sigma(\mathbf{a}_t \cdot \mathbf{t}_n) \quad (3)$$

, where $\mathbf{a}_t, \mathbf{t}_n \in \mathbb{R}^e$ and e is the embedding size. Sigmoid activation restricts s_{fp} to $[0, 1]^{T \times N}$ as it denotes the event probability. During training, audio pooling and text pooling transform s_{fp} to the clip-sentence similarity s_{cs} . Audio pooling summarizes s_{fp} into the clip-phrase similarity $s_{cp} \in [0, 1]^N$ while text pooling summarizes s_{cp} into s_{cs} :

$$\begin{aligned} s_{cp}(n) &= \text{Pool}_A(s_{fp}(1, n), s_{fp}(2, n), \dots, s_{fp}(T, n)) \\ s_{cs} &= \text{Pool}_T(s_{cp}(1), s_{cp}(2), \dots, s_{cp}(N)) \end{aligned} \quad (4)$$

During inference, audio pooling and text pooling are not required. We use the max margin ranking loss [48] for training.

For a minibatch with a batch size B , the loss encourages the similarities of positive audio-caption pairs to be higher than any negative pairs by at least the margin m :

$$\begin{aligned} \mathcal{L} &= \frac{1}{B} \sum_i \sum_{j \neq i} \mathcal{L}_t(i, j) + \mathcal{L}_a(i, j) \\ \mathcal{L}_t(i, j) &= \max(0, m + s_{cs}(i, j) - s_{cs}(i, i)) \\ \mathcal{L}_a(i, j) &= \max(0, m + s_{cs}(j, i) - s_{cs}(i, i)) \end{aligned} \quad (5)$$

where $s_{cs}(i, j)$ denotes the similarity between i -th audio clip and j -th sentence.

In the previous sentence-level WSTAG framework [16], the model calculates frame-word similarities. During inference, the frame-phrase similarities are obtained via another text pooling. Our new sentence-level WSTAG framework directly calculates frame-phrase similarities so text pooling is no longer needed during inference. It is more aligned with other methods since all other methods calculate frame-phrase similarities.

B. Pooling Strategies

Audio Pooling: Previous work [15] used mean pooling for Pool_A . As elaborated in [16], mean pooling potentially violates the standard SMI assumption of sounds so we investigate temporal pooling strategies utilized in WSSSED [49], including **mean pooling**, **max pooling**, **linear softmax pooling** and **exponential softmax pooling**.

$$\begin{aligned} \text{Mean: } s_{cp} &= \frac{1}{T} \sum_t s_{fp}(t) & \text{Max: } s_{cp} &= \max_t s_{fp}(t) \\ \text{Linear softmax: } s_{cp} &= \frac{\sum_t s_{fp}^2(t)}{\sum_t s_{fp}(t)} \\ \text{Exp. softmax: } s_{cp} &= \frac{\sum_t s_{fp}(t) \exp(s_{fp}(t))}{\sum_t \exp(s_{fp}(t))} \end{aligned} \quad (6)$$

Linear softmax achieves a strong performance in WSSSED [49]. However, in WSTAG, queries are free text instead of fixed classes and the loss function is also different. Therefore, linear softmax does not necessarily work well in this scenario.

Text Pooling: Text pooling aims to transform s_{cp} into s_{cs} . s_{cs} should be higher if more phrases present high similarity scores with the audio clip so **sum pooling** seems suitable, which is also used in several weakly-supervised visual grounding works [9], [50]. However, since s_{cp} is bound to be positive, s_{cs} cannot be penalized if there are irrelevant phrases in the caption under sum pooling. To mitigate this, we also incorporate **mean pooling** for text pooling.

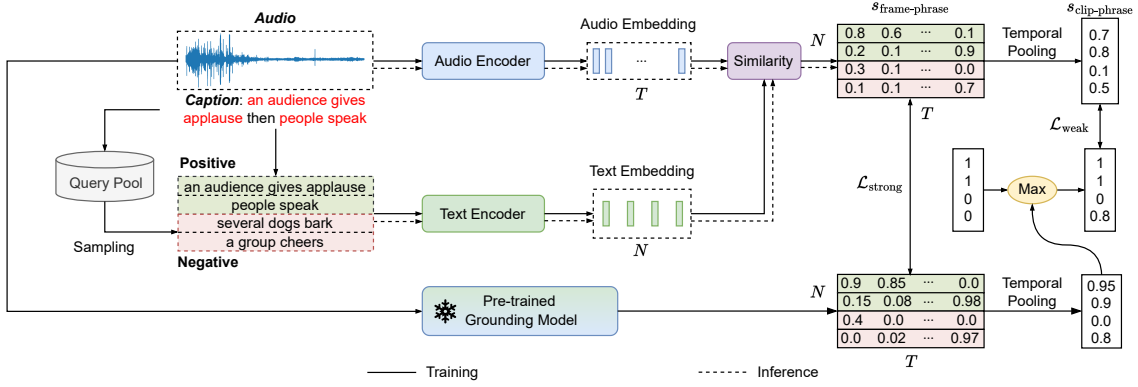


Fig. 3. The proposed phrase-level WSTAG approach. For an audio-caption pair, the training data contain both extracted positive phrases and sampled negative ones. A pre-trained WSTAG model is utilized to provide self-supervision. It adopts the same architecture as the WSTAG model to be trained and is trained using only $\mathcal{L}_{\text{weak}}$.

IV. PHRASE-LEVEL WEAKLY-SUPERVISED TEXT-TO-AUDIO GROUNDING

In phrase-level WSTAG, \hat{y} in Equation (1) is still the frame-phrase similarity s_{fp} but \hat{y} is the clip-phrase similarity s_{cp} . Phrase-level WSTAG is a combination of SSTAG and WSSD. SSTAG uses frame-level labels for training. Phrase-level WSTAG adapts the SSTAG framework to clip-level labels. As Figure 3 shows, a training data sample contains an audio clip and N phrases. Similar to sentence-level WSTAG, s_{cp} is calculated:

$$s_{\text{cp}}(n) = \text{Pool}(s_{\text{fp}}(1, n), s_{\text{fp}}(2, n), \dots, s_{\text{fp}}(T, n)) \quad (7)$$

Similar to WSSD, the training loss is the clip-level binary cross entropy (BCE) loss between s_{cp} and the label $y \in \{0, 1\}^N$:

$$\mathcal{L} = -\frac{1}{N} \sum_{n=1}^N y(n) \log(s_{\text{cp}}(n)) + (1 - y(n)) \log(1 - s_{\text{cp}}(n)) \quad (8)$$

However, phrases from the caption corresponding to the audio (positive phrases) are not enough for training. Since y for positive phrases is 1, the model will trivially predict probabilities close to 1 if only positive phrases are used for training. In contrast, in SSTAG there are negative frame-phrase pairs (the phrase is not present in the frame) so strong labels contain 0. In WSSD, the clip-level labels of an audio clip also contain 0 for absent events. Therefore, negative phrases, i.e., phrases not present in the audio, are necessary for phrase-level WSTAG.

A straightforward approach to negative phrase sampling is to randomly sample phrases from other captions. However, such phrases are probably not negative for the audio (we call them “**false negative phrases**”). A preliminary analysis shows that over 5% of all phrases describe the sound of male speech. If an audio clip contains the sound of a man speaking, phrases describing the same event are likely sampled as negative ones when the number of sampled phrases becomes large. Therefore, we propose advanced **sampling strategies** namely similarity- and clustering-based sampling in Section IV-A, to ensure that sampled phrases are truly negative. In addition,

to further enhance the label quality and provide frame-level supervision in the weakly-supervised setting, we propose **self-supervision** where a pre-trained WSTAG model with the same architecture guides training.

Algorithm 1: Similarity-based Negative Sampling.

Input: Audio captioning data sample $(a, c)_{i=1}^N$, query number n , query pool Q , threshold τ , batch size b , similarity function $\text{sim}(\cdot)$.
Output: WSTAG data sample (a, q, y) .

- 1 Extract phrases $\{q_1, q_2, \dots, q_{n_p}\}$ as Q_p from c ;
- 2 $Q_n \leftarrow \emptyset$;
- 3 **while** $|Q_n| < n - n_p$ **do**
- 4 Randomly sample $B \subset Q \setminus \{Q_p \cup Q_n\}$, $|B| = b$;
- 5 **for** $j = 1$ **to** b **do**
- 6 $s_j \leftarrow \max_q \{\text{sim}(B_j, q) \text{ for } q \text{ in } Q_p\}$;
- 7 **if** $s_j < \tau$ **then**
- 8 $Q_n \leftarrow Q_n \cup \{q_j\}$;
- 9 **end**
- 10 **if** $|Q_n| = n - n_p$ **then**
- 11 **break**
- 12 **end**
- 13 **end**
- 14 **end**
- 15 $q \leftarrow Q_p \cup Q_n$;
- 16 $y \leftarrow [0] * n$;
- 17 $y[: n_p] = 1$;
- 18 **return** (a, q, y) ;

A. Advanced Sampling Strategy

a) *Similarity-based sampling*: Similarity-based sampling is illustrated in Algorithm 1. It calculates the similarities between sampled phrases and positive phrases. Then phrases that are not negative enough are filtered out by a threshold. Since the size of the query pool can be very large, we adopt an efficient batch subset selection algorithm. It keeps sampling negative queries in batches until n queries are sampled. We calculate similarities between phrases by transforming phrases into audio-centric text representations introduced in Section II-C. We first train a bi-encoder using Contrastive Language-Audio Pre-training (CLAP) on the WSTAG dataset. Then the text encoder of CLAP is used to extract audio-centric

embeddings. It should be noted that although we use ‘‘CLAP’’, it is **trained solely on the WSTAG dataset without using extra data**. Acoustically similar phrase pairs are also close to each other in this text embedding space. Such a similarity-based sampling prevents phrases describing events in the audio clip from being sampled as negative phrases.

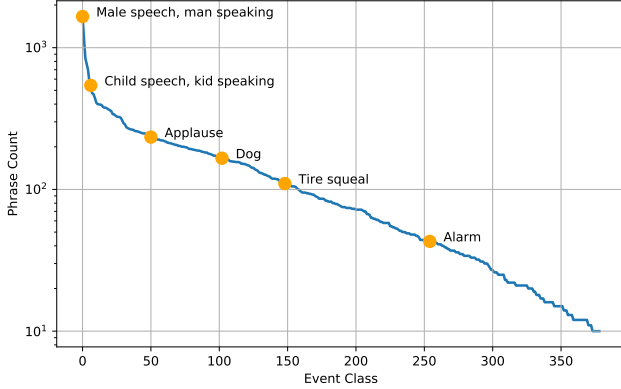


Fig. 4. The event distribution of phrases. Each phrase is mapped to its most acoustically similar AudioSet event. The phrase count of each class is plotted.

Algorithm 2: Clustering-based Negative Sampling.

Input: Audio captioning data sample (a, c) , query number n , clusters $C = \{s_i\}_{i=1}^{n_c}$, mapping function $M(\cdot)$ from phrases to clusters.

Output: WSTAG data sample (a, q, y) .

```

1 Extract phrases  $\{q_1, q_2, \dots, q_{n_p}\}$  from  $c$ ;
2  $C_p \leftarrow \cup\{M(q_1), M(q_2), \dots, M(q_{n_p})\}$ ;
3  $C_n \leftarrow C \setminus C_p$ ;
4  $Q_n \leftarrow \emptyset$ ;
5  $n_{sample} \leftarrow \min(|C_n|, n - n_p)$ ;
6 for  $j = 1$  to  $n_{sample}$  do
7   Randomly sample  $q_j$  from  $C_n[j]$ ;
8    $Q_n \leftarrow Q_n \cup \{q_j\}$ ;
9 end
10 if  $n - n_p > |C_n|$  then
11   for  $j = 1$  to  $n - n_p - |C_n|$  do
12     Randomly sample  $q_j$  from  $C_n[j \bmod |C_n|]$ ;
13      $Q_n \leftarrow Q_n \cup \{q_j\}$ ;
14   end
15 end
16  $q \leftarrow Q_p \cup Q_n$ ;
17  $y \leftarrow [0] * n$ ;
18  $y[:n_p] = 1$ ;
19 return  $(a, q, y)$ ;
```

b) Clustering-based sampling: Although similarity-based sampling filters out false negative phrases, the event distribution of sampled phrases is unbalanced. We map all phrases to AudioSet [51] classes based on the text similarity and plot the event distribution in Figure 4. There are over 1,000 phrases mapped to the sound of male speaking while the most infrequent class matches only 10 phrases. Therefore, it is highly probable that phrases describing infrequent events are seldom sampled by similarity-based sampling. In contrast, in WSED, labels of all events are always available. To make the event distribution more balanced, we propose clustering-based sampling, as Algorithm 2 shows. We first do clustering

on all phrases using their CLAP embeddings. In this way, phrases belonging to the same sound event or sharing the same acoustic characteristics are grouped into the same cluster. Negative phrases are sampled from each cluster iteratively to make the event distribution as even as possible.

B. Self-Supervision

The proposed sampling strategies are effective in achieving high-quality and balanced negative sampling. However, due to its weakly-supervised nature, noise still exists in clip-phrase matching labels, mostly derived from the following two aspects:

- The text matching or clustering based on CLAP embeddings is not perfect, so false negative phrases are not filtered out or assigned to the wrong groups.
- Some sound events are not included in the corresponding caption. Annotators are likely to focus only on prominent events while neglecting background sounds like wind blowing. As a result, phrases describing these events are mistakenly sampled.

Self-training and distillation has been found useful in approaches like noisy student training [52], [53]. We adopt a similar approach by taking soft labels estimated by a WSTAG teacher model for supervision and define it as WSTAG self-supervision. Although noise is inevitable in the training data, the teacher model is able to reasonably estimate clip-phrase similarities, especially for sampled negative phrases. The sufficient supervision of positive phrases, which are almost accurate, enables the model to recognize sound events when they are present. Therefore, a pre-trained model G is utilized to refine y in Equation (8). Furthermore, in addition to clip-level similarities, G also predicts frame-level ones. We also employ these frame-level similarities as training labels, thereby mitigating the drawback of WSTAG that frame-level supervision is lacking. In this way, the weakly-annotated data is leveraged so that G provides self-supervision.

Formally, G estimates the probabilities y_{self} of phrases \mathcal{P} given the audio \mathcal{A} . Then the maximum value of y and aggregated clip-level predictions $\hat{y}_{self} \in [0, 1]^N$ is used as the refined label $y_{refined} \in [0, 1]^N$:

$$\begin{aligned}
 y_{self}(t, n) &= G(\mathcal{A}, \mathcal{P}) \\
 \hat{y}_{self}(n) &= \text{Pool}_T(y_{self}(1, n), y_{self}(2, n), \dots, y_{self}(T, n)) \\
 y_{refined}(n) &= \max(y(n), \hat{y}_{self}(n))
 \end{aligned} \quad (9)$$

$y_{refined}$ replaces y in Equation (8) to calculate the weak loss \mathcal{L}_{weak} . The strong loss \mathcal{L}_{strong} is calculated on the frame level and the training loss is the combination of these two:

$$\begin{aligned}
 \mathcal{L}_{strong}(t, n) &= -y_{self}(t, n) \log(\text{sfp}(t, n)) - (1 - y_{self}(t, n)) \log(1 - \text{sfp}(t, n)) \\
 \mathcal{L} &= \frac{1}{N \cdot T} \sum_{n=1}^N \sum_{t=1}^T \mathcal{L}_{strong}(t, n) + \frac{1}{N} \sum_{n=1}^N \mathcal{L}_{weak}(n)
 \end{aligned} \quad (10)$$

Here G adopts exactly the same architecture as the target model. Therefore, the training contains two stages: 1) train a WSTAG model G using Equation (8); 2) use G as a teacher to train the student model using Equation (9).

V. EXPERIMENTAL SETUP

A. Datasets

In this work, all models are trained on AudioCaps [13], an audio captioning dataset without frame-level labels. There are 49501, 2475, and 4820 audio-caption pairs in the training, validation, and test set of our downloaded version, respectively. The three sets are merged and re-split into a training set and a validation set with 1,000 audio-caption pairs. The audio-centric text encoder used in phrase-level WSTAG is also trained on AudioCaps. We use AudioGrounding [4] for evaluation. AudioGrounding is a subset of AudioCaps augmented with human-annotated onsets and offsets of phrases. When training models on AudioCaps, audio files in AudioGrounding test set are eliminated. We only use the test set for evaluation. Although all audio-text datasets can be utilized for training, we only use AudioCaps in this work since AudioGrounding is its subset. Text descriptions from other datasets are found to have a different style from AudioCaps so incorporating these datasets may not be helpful.

B. Model Architectures

In this work, we use the same architectures for all approaches. The audio encoder is a CRNN with 8 CNN layers and a bidirectional GRU (BiGRU). The CNN structure is similar to CNN10 in PANNs [54] with the modification that the temporal down-sampling ratio is 4 instead of 16 to preserve a high time resolution. We pre-train the audio encoder on AudioSet to enhance its ability to recognize sound events. Since queries are mostly short phrases in AudioGrounding, we use a single randomly initialized word embedding layer with mean pooling as the text encoder.

CLAP is trained on the same dataset as WSTAG training. We follow the configuration in [55], where CNN14 in PANNs and BERT_{MEDIUM} are adopted as the audio encoder and text encoder. During WSTAG training, we use the embedding after the text projection layer as the phrase embedding, with a size of 1024.

C. Hyper-parameters

We extract 64-dimensional log mel-spectrograms (LMS) as the audio feature. The short-time Fourier transformation (STFT) window size and window shift are 32 ms and 10 ms. The hidden size of the BiGRU is 256. Dimensions of the audio and text embeddings are both 512. The model is trained for at most 100 epochs with an early stop patience of 10 epochs and a batch size of 32, using Adam optimization. The learning rate is initially 0.001 and reduced to $\frac{1}{10}$ if the validation loss does not improve for 3 epochs.

D. Evaluation Metrics

PSDS: Following previous practices [4], [56], polyphonic sound detection score (PSDS) [57] is used for evaluation. It is the area under the PSD-ROC curve, which measures the relationship between the true positive rate (TPR) and the false positive rate (FPR). Since the ground truth class of each phrase is unavailable, cross trigger is not considered here. We

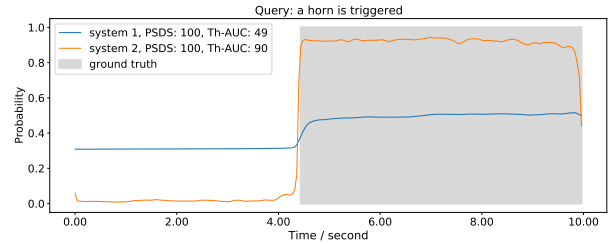


Fig. 5. A comparison of evaluation on a sample using PSDS and Th-AUC. PSDS measures the performance under the best threshold while Th-AUC measures the performance over all possible thresholds.

set $\rho_{DTC} = \rho_{GTC} = 0.5$, meaning that an output segment will be determined as correct if its overlap ratio with the ground truth segment is over 0.5. However, e_{max} is changed from 100 to 800, which is the maximum FPR value when calculating PSDS. In SED, FPR is calculated as the average value of each class but in TAG FPR values of all classes are summed up since the ground truth class is unavailable. Therefore the maximum tolerant FPR should be increased. PSDS is calculated using the toolkit from [58].

Th-AUC: In PSDS evaluation, the TPR-FPR curve is plotted by calculating detection metrics under different thresholds. However, we find that it does not measure the model’s robustness against different thresholds since it considers only the optimal threshold for a specific FPR value to ensure the monotonicity of the PSD-ROC curve [57]. For example, in Figure 5 two systems can both perform well under appropriate thresholds so their PSDS scores are the same. However, system 2 performs better than system 1 under most thresholds. To measure the model’s robustness under a large scope of thresholds, we propose the area under F_1 -threshold curve (Th-AUC). It measures the average performance under all possible thresholds. A significant gap between the performance of two systems in Figure 5 is shown in terms of Th-AUC.

For both PSDS and Th-AUC, two results are reported. One is calculated on the whole test set while the other is on a short-duration subset. The subset is built by selecting segments that last for less than half of the whole audio clip for a more stringent evaluation. Since many events last for the whole audio clip, models predicting high probabilities regardless of inputs still achieve high PSDS scores on the whole test set. On the short-duration subset, such models will get low scores. Therefore, the subset can validate models’ performance better.

VI. RESULTS

In this section, we report our experimental results. We provide an extensive analysis of the influencing factors and robustness of the proposed WSTAG approaches. Finally, a few qualitative results are given to show the performance and possible limits of our method.

A. Influence of Pooling Strategies on Sentence-level WSTAG

First, like the previous investigation on pooling strategies [16], we analyze the influence of pooling strategies on sentence-level WSTAG. Results are presented in the upper

TABLE I
WSTAG RESULTS USING DIFFERENT POOLING STRATEGIES.

	Pooling Strategy	Whole		Short	
		PSDS	Th-AUC	PSDS	Th-AUC
Audio Pooling	Mean	29.9	38.0	5.5	3.7
	Max	42.7	48.9	33.8	42.4
	Linear Softmax	32.4	40.4	6.2	8.7
	Exp. Softmax	30.0	38.4	5.9	4.3
Text Pooling	Mean	42.7	48.9	33.8	42.4
	Sum	42.7	43.9	33.4	38.1

part of Table I. We keep mean pooling for text the same. The conclusion for the new framework aligns with that in the previous work that max pooling shows a significant advantage over other strategies. As analyzed in [16], only one frame has a non-zero gradient using max pooling so it performs better under the weakly-supervised setting where gradients are likely to be misleading. Then, the effect of text pooling is analyzed. In contrast with the finding that sum pooling performs better in [16], mean pooling achieves superior performance in the new framework. This may be attributed to the disparities in the granularity of similarity calculation in the two frameworks. The old framework calculates frame-word similarities so functional words affect the aggregated similarity to a large extent. However, the new framework directly calculates frame-phrase similarities so the model may be trained to ignore functional words when encoding phrases. As a result, the negative effect of mean pooling is alleviated. Besides, similar PSDS scores but the gap in Th-AUC scores show that the two pooling strategies perform similarly under the optimal threshold but mean pooling performs better in a larger scope of thresholds. This indicates that in sum pooling, models are trained to assign lower scores to positive phrase queries to reduce the score of negative captions with both positive and negative phrases, leading to their worse robustness against thresholds.

B. Comparison between Approaches

In the upper half of Table II, we present the performance of our proposed WSTAG approaches. We use the same architecture for all approaches. Compared with the baseline sentence-level WSTAG using mean pooling for both audio and text, significant improvement is achieved by replacing audio mean pooling with max pooling. For phrase-level WSTAG, we use a phrase number n (see Algorithm 2) of 32 in all experiments. The similarity threshold τ in similarity-based sampling is 0.5. Clustering-based sampling uses k-means clustering with a cluster number of 32. All models use linear softmax pooling. Results show that phrase-level WSTAG is superior to sentence-level approaches. Even the most straightforward random sampling can achieve comparable performance to the best-performing sentence-level WSTAG. With the proposed advanced sampling strategies based on similarity and clustering, significant improvement is achieved. An absolute 10% improvement in terms of $\text{PSDS}_{\text{whole}}$ is achieved, indicating that our proposed sampling strategies significantly enhance the quality of training labels. Clustering-based sampling shows an advantage over similarity-based sampling. This validates the

benefit of a more balanced distribution of events corresponding to negative phrases.

Regardless of the sampling strategy, the involvement of self-supervision brings substantial performance improvement. The refinement of clip-level labels and the availability of frame-level labels from a pre-trained model further improves the label quality. In particular, the introduction of frame-level pseudo labels enables the model to receive frame-level supervision without ground truth frame-level labels, significantly enhancing the temporal localization accuracy.

C. Comparison with Previous Methods

We list several previous TAG methods for comparison in the lower half of Table II. Since there are few works on TAG, we include methods adapted from related works for comparison. Conditional CDur uses a method similar to [59] to fuse text into the SED model. The text feature is added to each output channel of feature maps to predict queries. CRNN-w2vmean and CRNN-QGCA are adapted from [4] and [56] by replacing the CDur audio encoder with the CRNN in this work. CRNN-BERT further improves CRNN-w2vmean by using a frozen $\text{BERT}_{\text{BASE}}$ text encoder.

These adapted methods provide much stronger baselines. Our proposed WSTAG model achieves competitive performance in terms of PSDS while outperforming all SSTAG methods in Th-AUC. WSTAG automatically learns event patterns from weak labels and leverages a large amount of audio-text data to avoid the drawback of imprecise timestamp annotations of short events. Therefore, WSTAG demonstrates a greater advantage in short-duration event detection. Additionally, the imprecision of timestamp annotations around event boundaries may cause SSTAG models to produce outputs that are more inclined towards moderate values, thus lacking robustness to threshold variations. In contrast, WSTAG mitigates this issue, achieving favorable results across a wider range of thresholds and thereby attaining a higher Th-AUC.

D. Evaluation on SED

To assess the generalizability of our model, we further evaluate our model on the DESED test set [61], a sound event detection dataset. Following [60], we calculate F_1 scores under PSDS settings, as Th-AUC does. We compare our model with PANNs [54] and HTS-AT [60] as they are not trained directly on DESED either. Our model is trained on AudioCaps, which is only $\frac{1}{20}$ of AudioSet, where PANNs and HTS-AT are trained. The result in Table III shows that our method efficiently leverages text annotation of audio clips, outperforming PANNs and HTS-AT significantly with much smaller training data. For common sounds like speech and dog, our model performs well on DESED without being exposed to its training set, validating the robustness of our model.

E. Analysis on Influencing Factors of Phrase-level WSTAG

In this part, we extensively analyze the influence of several factors on phrase-level WSTAG. When we analyze one factor, we keep all other settings the same as that in the previous

TABLE II
RESULTS OF DIFFERENT WSTAG APPROACHES. THE BEST WSTAG AND SSTAG RESULTS ARE HIGHLIGHTED IN BOLD. FOR ALL WSTAG METHODS, WE USE THE SAME MODEL ARCHITECTURE.

Approach		Whole		Short	
		PSDS	Th-AUC	PSDS	Th-AUC
<i>Our Proposed WSTAG</i>					
Sentence-level WSTAG (Ours)	A-Mean + T-Mean	29.9	38.0	5.5	3.7
	A-Max + T-Mean	42.7	48.9	33.8	42.4
Phrase-level WSTAG (Ours)	Random Sampling	43.7	46.5	34.5	43.2
	+ Self-Supervision	48.7	50.5	40.1	46.7
	Similarity-based Sampling	52.6	53.9	43.7	48.5
	+ Self-Supervision	55.7	57.1	46.2	50.8
Phrase-level WSTAG (Ours)	Clustering-based Sampling	52.9	54.2	44.4	49.3
	+ Self-Supervision	56.5	57.1	47.6	51.7
<i>Baseline TAG Methods (from Previous Works or Adaptation)</i>					
Previous WSTAG Methods	CDur-word Alignment [15]	35.2	35.4	6.8	5.3
Previous SSTAG Methods	CDur-w2vmean [4]	56.0	40.2	39.8	23.0
	CDur-QGCA [56]	57.4	47.2	43.2	31.5
	Conditional CDur (Adapted from [59])	57.6	48.8	43.1	35.7
	CRNN-w2vmean	58.3	42.3	43.3	25.6
	CRNN-BERT	60.4	56.0	48.5	44.3
	CRNN-QGCA	62.8	53.1	49.8	38.4

TABLE III
SED PERFORMANCE IN TERMS OF F₁ SCORE ON THE DESED TEST SET. WE ONLY LIST THE AVERAGE SCORE AND SCORES OF SOME CLASSES DUE TO LIMITED SPACE.

Model	Speech	Dog	Cat	Water	Frying	Average
PANNs [54]	69.7	35.8	36.3	30.6	9.3	35.1
HTS-AT [60]	46.8	48.0	67.7	43.0	60.3	48.4
WSTAG (Ours)	84.5	80.8	84.2	34.5	54.6	58.0

part. For simplicity, we do not involve self-supervision since it requires two training stages and we only list the results of PSDS scores on the short-duration subset ($PSDS_{short}$).

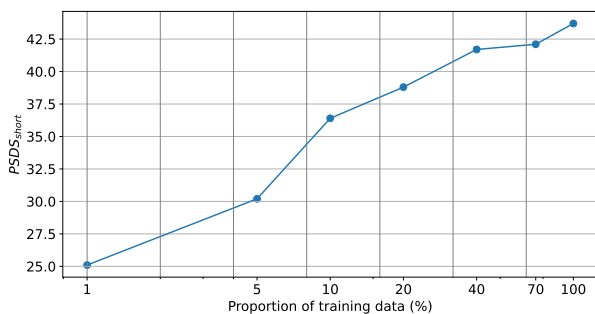


Fig. 6. Analysis of WSTAG using different amounts of training data.

1) *Data Size*: Since the advantage of WSTAG is to leverage large-scale weakly-annotated audio-text data for training, we analyze the influence of data size on the performance. For simplicity, the experiments are based similarity-based sampling strategy so the training of clustering models using different portion of data is not needed. in Figure 6. Generally, WSTAG performance improves with an increase in the dataset size. When the dataset is relatively small, such as 5% of the total, the performance gains from increasing the dataset size

are quite pronounced. However, as the dataset size grows to around 50% (25K pairs), the performance improvement becomes limited. Since the dataset size is already much larger than the strongly-supervised one (5K pairs) with most categories covered, further increasing the size may have a limited impact on frequent categories. In addition, the improvement in infrequent categories contributes relatively little to the overall performance, resulting in limited overall gains.

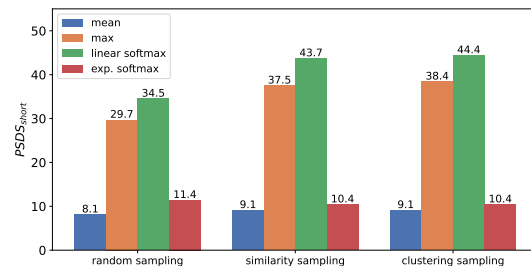


Fig. 7. Comparison of different pooling strategies.

2) *Pooling Strategies*: Figure 7 shows the performance of phrase-level WSTAG approaches using different pooling strategies. Same as audio pooling in sentence-level WSTAG, we compare strategies in Section III-B. The result is similar to sentence-level WSTAG in that mean pooling performs the worst. The violation of the SMI assumption dramatically hurts the performance. However, linear softmax consistently outperforms other strategies in phrase-level WSTAG. Max pooling still achieves good performance but lags behind linear softmax to a large extent. This comparison between sentence- and phrase-level WSTAG results suggests that pooling strategies heavily rely on suitable loss functions to work well: linear softmax is compatible with BCE loss while max pooling performs well under the contrastive loss.

3) *Phrase Numbers*: We analyze the influence of using different phrase numbers n . More negative phrases are sampled as

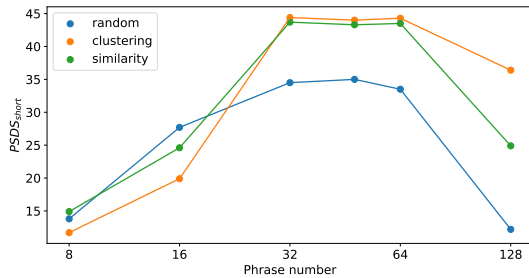


Fig. 8. Comparison of different phrase numbers.

n increases. Results are shown in Figure 8. For either sampling strategy, the phrase number imposes a significant impact on the performance and the optimal phrase number is around 32 and 64. If n is too small, phrases describing various sound events are not sampled during training. However, when n is too large, the probability that false negative phrases are sampled also becomes larger. It is observed that as n increases to 128, the advantage of the advanced sampling strategies over random sampling grows bigger. However, when n is much lower than the optimal value, the clustering-based sampling strategy leads to the worst result. This may be caused by the distribution of events. As n is much lower than the total sound event number, clustering-based sampling can only sample phrases covering a part of sound events. Even the most frequent events cannot be detected well. However, random sampling has more access to these frequent events. Therefore, random sampling achieves better performance when n is small.

TABLE IV
COMPARISON OF DIFFERENT PHRASE EMBEDDINGS.

Sampling Strategy	Similarity Sampling	Clustering Sampling
SBERT _{roberta-large}	42.2	28.3
CLAP _{bert-medium}	43.7	44.4

4) *Phrase Embeddings*: Here we analyze the influence of different phrase embeddings on negative sampling strategies in Table IV. We compare a pure semantic encoder, Sentence-BERT [36] with the audio-centric CLAP. We use Roberta_{LARGE} as its embedding size is the same as CLAP. Without audio-text learning, Sentence-BERT gives the similarity between phrases solely based on the semantic meaning. Therefore, CLAP consistently achieves better results than Sentence-BERT. Especially in clustering-based sampling, phrases from clusters of positive phrases are never sampled. The reliability of clustering is crucial to the result so better text embeddings can improve the performance significantly.

5) *Cluster Numbers*: The number of cluster centers n_c is an important factor in clustering. We present the results of clustering-based sampling using different n_c in Figure 9. Here the k-means clustering algorithm is exclusively used. We also plot the clustering inertia for reference. It measures the distance of samples to their closest cluster center. Although the inertia keeps decreasing as n_c grows, WSTAG performance starts declining after n_c reaches 32. Too small or too large n_c hurt the performance. When n_c is too small, phrases

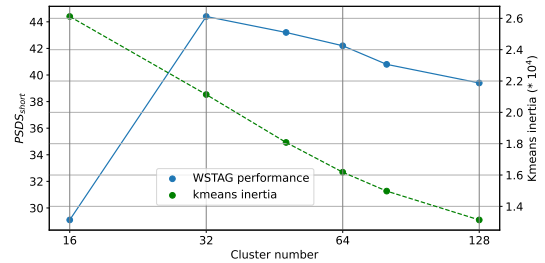


Fig. 9. Comparison of different cluster numbers in clustering-based sampling.

describing the same general sound but belonging to different fine-grained sounds are grouped into the same cluster (called false-merging). For example, phrases describing all animals are grouped into the same cluster when n_c is extremely small. During evaluation, there will be many false positive predictions for fine-grained sound event, e.g., dog barking or pig oinking. Therefore, the performance declines dramatically with only 16 clusters. In contrast, a large cluster number will force phrases describing the same sound event to group into different clusters (we call it false-splitting), especially for sound events with a large number of phrases like male speech. The model will thus estimate lower probabilities for these events when they are present. It can be observed from the comparison of performance under small and large n_c that the false-merging problem is more harmful than false-splitting.

TABLE V
COMPARISON OF DIFFERENT CLUSTER METHODS IN CLUSTERING-BASED SAMPLING.

Clustering Method	Kmeans	Spectral	Agglomerative
PSDS _{short}	44.4	41.5	38.2

6) *Clustering Algorithms*: Since the clustering result is critical to clustering-based sampling, we analyze the influence of different clustering algorithms. The result is shown in Table V. Besides k-means clustering, we also investigate spectral clustering and agglomerative clustering. K-means performs the best. It may be attributed to the fact that k-means is a general clustering algorithm but the other two are more suitable for specific data types. Compared with phrase embeddings and cluster numbers, our approach is not sensitive to the clustering algorithm.

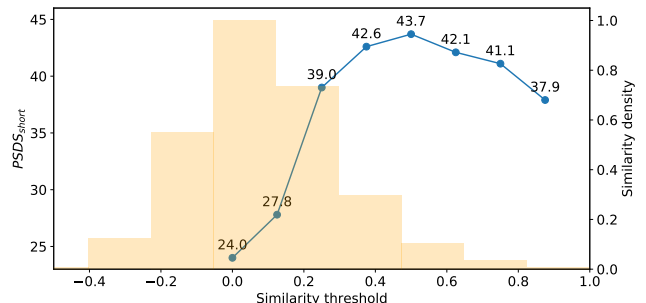


Fig. 10. Comparison of different similarity thresholds τ in similarity-based sampling.

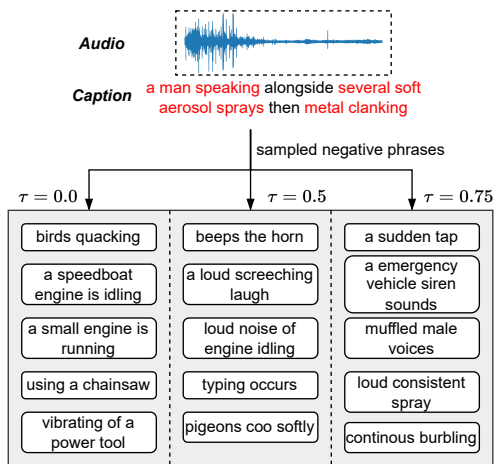


Fig. 11. An example of negative phrases sampled under different similarity thresholds τ .

7) *Similarity Thresholds*: For similarity-based sampling, an important hyper-parameter is the similarity τ used to determine whether a phrase is negative for an audio clip. Its influence is investigated and presented in Figure 10. The distribution of similarities between different phrases is also plotted in terms of the normalized density. $\tau = 0.5$ achieves the best result. When τ becomes larger, the algorithm fails to filter out many false negative phrases. When τ is not large enough, there is a risk of ignoring hard negative phrases (e.g., ignoring negative phrases like “another man speaking” in a recording of a single man speaking). However, the inclusion of false negative phrases (e.g., taking “male voice” as negative) has a larger influence than ignoring hard negative phrases, as indicated by the drop in the performance when τ keeps increasing. The discrimination of hard negative phrases may need a better CLAP.

If τ becomes too small, the number of candidate phrases also decreases a lot. Although there are still a large number of phrase pairs with negative similarities, phrases corresponding to many sound events are never sampled under $\tau = 0$, resulting in poor performance though the sampled phrases are definitely negative. An example is shown in Figure 11. When $\tau = 0.0$, the diversity of sampled phrases is limited: 4 of 5 randomly sampled phrases describe the humming or vibration of engines. When τ increases to 0.75, sampled phrases become much more diverse but false negative phrases are also sampled, e.g., “loud consistent spray”. $\tau = 0.75$ performs much better than $\tau = 0.0$, suggesting that the diversity of sampled phrases and corresponding events is more important than the accuracy of phrase labels.

F. Qualitative Results

In this part, we provide an analysis of our proposed WSTAG system through visualization. We first compare our best-performing WSTAG system with the baseline WSTAG and SSTAG systems. The baseline WSTAG system is the sentence-level one with mean pooling while our proposed WSTAG system is the phrase-level one with clustering-based sampling and self-supervision. Then we provide several qualitative data samples, including both successful and failure cases, to show

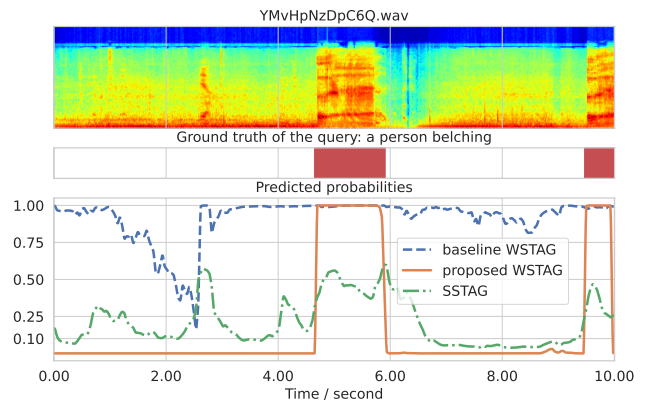


Fig. 12. An audio-phrase pair sample comparison on (1) the baseline WSTAG system; (2) our proposed WSTAG system; and (3) SSTAG system. Best viewed in color.

the model’s strengths and limitations. Finally, we compare our WSTAG system with an SED system to highlight the advantage of TAG over SED.

We randomly sample an audio-phrase pair from the short-duration subset and the comparison of different approaches is shown in Figure 12. The phrase query “a person belching” corresponds to the class “Hiccup” in AudioSet, which is an infrequent class with only 931 samples in the total 2.1 M audio clips. The baseline system is sentence-level WSTAG with mean pooling for both audio and text. It fails to detect the segments since it assigns high probabilities to all non-silent segments, regardless of the sound type. The violation of SMI assumption results in its bad performance. Although trained on strongly-annotated data, SSTAG is incapable of producing perfect predictions neither. This may be attributed to the scarcity of the event “Hiccup” in the strongly-annotated dataset. In contrast, our proposed phrase-level WSTAG system accurately predicts the onsets and offsets. Since the acoustic characteristics of hiccups are notable, WSTAG systems are trained to recognize such sounds from a variety of audio clips.

Furthermore, several examples of our proposed system’s predictions are shown in Figure 13. The first three are successful cases, where accurate predictions can be obtained by using a standard threshold of 0.5 and post-processing like median filtering. As WSTAG methods do not involve timestamp labels for training, the model learns sound patterns from the comparison between recordings with and without specific events. No preference like merging short segments is introduced in this training paradigm. Therefore, for short and loud sounds, the probability predicted by the model oscillates sharply between high and low values, as shown in the first example. The rest two examples are typical failure cases. The query of the first example is “something crunches”, which is often in the background with a low loudness. It is difficult for the model to learn the characteristics of such background sounds with the interference of predominant sounds. In the last example, the caption is a detailed description: “... and a man talking followed by another man talking”, with the phrase query “another man talking”. It requires the model to recognize that there are two men speaking and that the

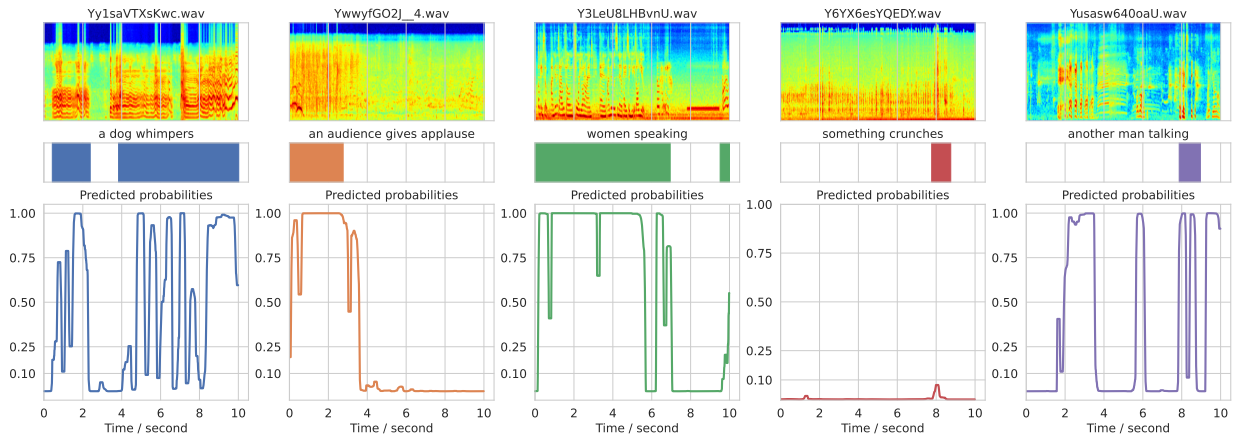


Fig. 13. Predictions of our proposed WSTAG system for five samples, including 3 successful cases and 2 failure cases.

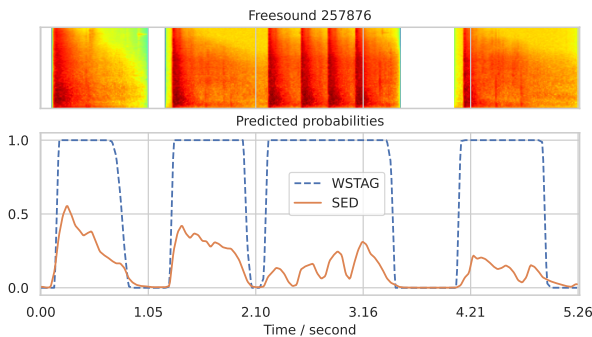


Fig. 14. Comparison between WSTAG and SED models on a Freesound sample.

desired output is the speech segment of the second man. The model predicts all speech segments. The behavior is attributed to the characteristics of training data: most captions just describe present sound events without details such as the speaker number or the temporal relationship between events. Since AudioCaps only cover a limited set of frequent, daily non-musical sounds, it is challenging for our WSTAG to detect rare, unseen sounds accurately. These failure cases indicate the limitations of our model in detecting background sounds and handling queries that seldom occur in the training set, including unseen sound categories and queries with details like the speaker identity.

Finally, we present the advantage of TAG over SED models by comparing their predictions. We choose a gunshot sample from out-of-distribution Freesound and the comparison is shown in Figure 14. The SED model is trained on strongly-annotated AudioSet subset [62], which is twice the size of AudioCaps. In spite of the larger training data size, the SED model gives relatively low probabilities when the gunshot occurs. However, the prediction of the WSTAG model is near optimal. We speculate the problem of SED originated from AudioSet labels [63], sound events such as gunshot are sometimes missing in annotations. During training, probabilities predicted for these events in data containing them are discouraged,

resulting in lower predicted probabilities when these events occur during inference. In contrast, annotators tend not to overlook prominent foreground sound events when describing an audio clip, thus avoiding this problem. Therefore, the advantage of natural language, compared with categorical systems, leads to TAG’s advantage over SED.

VII. CONCLUSION

In this paper, we explore TAG, the intersection of natural language processing and sound event detection, which aims to predict timestamps for sound events described by language. A major challenge in this field is the scarcity of strongly-labeled data, underscoring the importance of WSTAG research. With a lower requirement for annotations, WSTAG can make use of large-scale audio-text datasets for training. We first analyze the limitations of sentence-level WSTAG pooling strategies and investigate different pooling strategies. The best results are achieved with audio max pooling and text mean pooling. We then propose phrase-level WSTAG to narrow the training/test gap in the textual modality. Furthermore, we propose advanced negative sampling strategies and self-supervision to improve weak label accuracy and reduce the gap in the audio modality. Our phrase-level WSTAG outperforms the SSTAG system with the same architecture and is close to the SOTA SSTAG system, with strong performance on short-duration sounds. The evaluation on an unseen SED dataset validates the generalization ability of our model. We also analyze several factors impacting phrase-level WSTAG and visualize the advantages and limitations of our approach.

ACKNOWLEDGMENT

This work has been supported by Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102) and Jiangsu Technology Project (No.BE2022059-2). We would like to thank Mark D. Plumbley, who provided valuable insights and assistance in polishing this manuscript.

REFERENCES

- [1] Y. Tan, H. Ai, S. Li, and M. D. Plumbley, “Acoustic scene classification across cities and devices via feature disentanglement,” *IEEE/ACM Trans. Audio, Speech Lang. Process.*, 2024.

- [2] L. Xu, L. Wang, S. Bi, H. Liu, and J. Wang, "Semi-supervised sound event detection with pre-trained model," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2023, pp. 1–5.
- [3] L. Gao, Q. Mao, and M. Dong, "On local temporal embedding for semi-supervised sound event detection," *IEEE/ACM Trans. Audio, Speech Lang. Process.*, 2024.
- [4] X. Xu, H. Dinkel, M. Wu, and K. Yu, "Text-to-audio grounding: Building correspondence between captions and sound events," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2021, pp. 606–610.
- [5] S. Bhosale, R. Chakraborty, and S. K. Kopparapu, "A novel metric for evaluating audio caption similarity," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2023, pp. 1–5.
- [6] B. A. Plummer, L. Wang, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik, "Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2641–2649.
- [7] J. Gao, C. Sun, Z. Yang, and R. Nevatia, "TALL: Temporal activity localization via language query," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5267–5275.
- [8] L. Zhou, Y. Kalantidis, X. Chen, J. J. Corso, and M. Rohrbach, "Grounded video description," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 6578–6587.
- [9] L. Zhou, N. Louis, and J. J. Corso, "Weakly-supervised video object grounding from text by loss weighting and object interaction," in *Brit. Mach. Vis. Conf.*, 2018.
- [10] Y. Wang, J. Deng, W. Zhou, and H. Li, "Weakly supervised temporal adjacent network for language grounding," *IEEE Trans. Multimedia*, vol. 24, pp. 3276–3286, 2021.
- [11] Y. Song, R. Zhang, Z. Chen, X. Wan, and G. Li, "Advancing visual grounding with scene knowledge: Benchmark and method," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 15 039–15 049.
- [12] Y. Huang, L. Yang, and Y. Sato, "Weakly supervised temporal sentence grounding with uncertainty-guided self-training," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 18 908–18 918.
- [13] C. D. Kim, B. Kim, H. Lee, and G. Kim, "AudioCaps: Generating captions for audios in the wild," in *Proc. Conf. North Am. Chapter Assoc. Comput. Linguist.*, 2019, pp. 119–132.
- [14] K. Drossos, S. Lipping, and T. Virtanen, "Clotho: An audio captioning dataset," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2020, pp. 736–740.
- [15] H. Xie, O. Räsänen, K. Drossos, and T. Virtanen, "Unsupervised audio-caption aligning learns correspondences between individual sound events and textual phrases," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2022, pp. 8867–8871.
- [16] X. Xu, M. Wu, and K. Yu, "Investigating pooling strategies and loss functions for weakly-supervised text-to-audio grounding via contrastive learning," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. Workshop.* IEEE, 2023, pp. 1–5.
- [17] A. B. Vasudevan, D. Dai, and L. Van Gool, "Object referring in videos with language and human gaze," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4129–4138.
- [18] L. Yang, Z. Zhang, Z. Qi, Y. Xu, W. Liu, Y. Shan, B. Li, W. Yang, P. Li, Y. Wang *et al.*, "Exploiting contextual objects and relations for 3d visual grounding," *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, vol. 36, 2024.
- [19] L. A. Hendricks, O. Wang, E. Shechtman, J. Sivic, T. Darrell, and B. Russell, "Localizing moments in video with temporal language," in *Proc. Conf. Empir. Methods Nat. Lang. Process.*, 2018.
- [20] M. Liu, L. Nie, Y. Wang, M. Wang, and Y. Rui, "A survey on video moment localization," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–37, 2023.
- [21] R. Zhang, C. Wang, and C.-L. Liu, "Cycle-consistent weakly supervised visual grounding with individual and contextual representations," *IEEE Transactions on Image Processing*, 2023.
- [22] S. Datta, K. Sikka, A. Roy, K. Ahuja, D. Parikh, and A. Divakaran, "Align2ground: Weakly supervised phrase grounding guided by image-caption alignment," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 2601–2610.
- [23] C. Da, Y. Zhang, Y. Zheng, P. Pan, Y. Xu, and C. Pan, "Asynce: Disentangling false-positives for weakly-supervised video grounding," in *Proc. ACM Int. Conf. Multimedia*, 2021, pp. 1129–1137.
- [24] N. C. Mithun, S. Paul, and A. K. Roy-Chowdhury, "Weakly supervised video moment retrieval from text queries," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 11 592–11 601.
- [25] A. Rohrbach, M. Rohrbach, R. Hu, T. Darrell, and B. Schiele, "Grounding of textual phrases in images by reconstruction," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 817–834.
- [26] X. Liu, L. Li, S. Wang, Z.-J. Zha, D. Meng, and Q. Huang, "Adaptive reconstruction network for weakly supervised referring expression grounding," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 2611–2620.
- [27] T. Gupta, A. Vahdat, G. Chechik, X. Yang, J. Kautz, and D. Hoiem, "Contrastive learning for weakly supervised phrase grounding," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2020, pp. 752–768.
- [28] L. Zhou, C. Gong, Z. Liu, and K. Fu, "SAL: Selection and attention losses for weakly supervised semantic segmentation," *IEEE Trans. Multimedia*, vol. 23, pp. 1035–1048, 2020.
- [29] Z. Ren, S. Wang, and Y. Zhang, "Weakly supervised machine learning," *CAAI Transactions on Intelligence Technology*, vol. 8, no. 3, pp. 549–580, 2023.
- [30] C. Gong, J. Yang, J. You, and M. Sugiyama, "Centroid estimation with guaranteed efficiency: A general framework for weakly supervised learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 2841–2855, 2020.
- [31] H. Dinkel, M. Wu, and K. Yu, "Towards duration robust weakly supervised sound event detection," *IEEE/ACM Trans. Audio, Speech Lang. Process.*, vol. 29, pp. 887–900, 2021.
- [32] Y. Xin, D. Yang, and Y. Zou, "Background-aware Modeling for Weakly Supervised Sound Event Detection," in *Proc. ISCA Annu. Conf. Int. Speech Commun. Assoc.*, 2023, pp. 1199–1203.
- [33] J. Gao, M. Chen, and C. Xu, "Collecting cross-modal presence-absence evidence for weakly-supervised audio-visual event perception," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 18 827–18 836.
- [34] K. Li, S. Yang, L. Zhao, and W. Wang, "Weakly labeled sound event detection with a capsule-transformer model," *Digital Signal Processing*, vol. 146, p. 104347, 2024.
- [35] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Am. Chapter Assoc. Comput. Linguist.*, 2019, pp. 4171–4186.
- [36] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese bert-networks," in *Proc. Conf. Empir. Methods Nat. Lang. Process.*, 2019, pp. 3982–3992.
- [37] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "Data2vec: A general framework for self-supervised learning in speech, vision and language," *Proc. of ICML*, 2022.
- [38] A. Baevski, S. Schneider, and M. Auli, "vq-wav2vec: Self-supervised learning of discrete speech representations," in *Proc. of ICLR*, 2019.
- [39] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Proc. of NeurIPS*, 2020.
- [40] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Trans. Audio, Speech Lang. Process.*, 2021.
- [41] Z. Ma, Z. Zheng, C. Tang, Y. Wang, and X. Chen, "MT4SSL: Boosting Self-Supervised Speech Representation Learning by Integrating Multiple Targets," in *Proc. ISCA Annu. Conf. Int. Speech Commun. Assoc.*, 2023, pp. 82–86.
- [42] H. Bao, L. Dong, S. Piao, and F. Wei, "BEiT: BERT pre-training of image transformers," in *Proc. Int. Conf. Learn. Representations*, 2021.
- [43] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. of CVPR*, 2022.
- [44] A. Vijayakumar, R. Vedantam, and D. Parikh, "Sound-word2vec: Learning word representations grounded in sounds," in *Proc. Conf. Empir. Methods Nat. Lang. Process.*, 2017, pp. 920–925.
- [45] Y. Wu, K. Chen, T. Zhang, Y. Hui, T. Berg-Kirkpatrick, and S. Dubnov, "Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2023, pp. 1–5.
- [46] X. Mei, C. Meng, H. Liu, Q. Kong, T. Ko, C. Zhao, M. D. Plumbley, Y. Zou, and W. Wang, "Wavcaps: A ChatGPT-assisted weakly-labelled audio captioning dataset for audio-language multimodal research," *arXiv preprint arXiv:2303.17395*, 2023.
- [47] B. Elizalde, S. Deshmukh, and H. Wang, "Natural language supervision for general-purpose audio representations," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2024, pp. 336–340.
- [48] A. Karpathy, A. Joulin, and L. F. Fei-Fei, "Deep fragment embeddings for bidirectional image sentence mapping," *Advances in neural information processing systems*, vol. 27, 2014.

- [49] Y. Wang, J. Li, and F. Metze, “A comparison of five multiple instance learning pooling functions for sound event detection with weak labeling,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2019, pp. 31–35.
- [50] A. Karpathy and L. Fei-Fei, “Deep visual-semantic alignments for generating image descriptions,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3128–3137.
- [51] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, “Audio set: An ontology and human-labeled dataset for audio events,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2017, pp. 776–780.
- [52] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le, “Self-training with noisy student improves imagenet classification,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10 687–10 698.
- [53] D. S. Park, Y. Zhang, Y. Jia, W. Han, C.-C. Chiu, B. Li, Y. Wu, and Q. V. Le, “Improved noisy student training for automatic speech recognition,” in *Proc. ISCA Annu. Conf. Int. Speech Commun. Assoc.*, 2020, pp. 2817–2821.
- [54] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, “PANNs: Large-scale pretrained audio neural networks for audio pattern recognition,” *IEEE/ACM Trans. Audio, Speech Lang. Process.*, vol. 28, pp. 2880–2894, 2020.
- [55] X. Xu, Z. Xie, M. Wu, and K. Yu, “The SJTU system for DCASE2022 challenge task 6: Audio captioning with audio-text retrieval pre-training,” DCASE2022 Challenge, Tech. Rep., 2022.
- [56] H. Tang, J. Zhu, Q. Zheng, and Z. Cheng, “Query-graph with cross-gating attention model for text-to-audio grounding,” *arXiv preprint arXiv:2106.14136*, 2021.
- [57] Ć. Bilen, G. Ferroni, F. Tuveri, J. Azcarreta, and S. Krstulović, “A framework for the robust evaluation of sound event detection,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2020, pp. 61–65.
- [58] J. Ebberts, R. Haeb-Umbach, and R. Serizel, “Threshold independent evaluation of sound event detection scores,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2022, pp. 1021–1025.
- [59] Q. Kong, Y. Wang, X. Song, Y. Cao, W. Wang, and M. D. Plumbley, “Source separation with weakly labelled data: An approach to computational auditory scene analysis,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2020, pp. 101–105.
- [60] K. Chen, X. Du, B. Zhu, Z. Ma, T. Berg-Kirkpatrick, and S. Dubnov, “HTS-AT: A hierarchical token-semantic audio transformer for sound classification and detection,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2022, pp. 646–650.
- [61] R. Serizel, N. Turpault, A. Shah, and J. Salamon, “Sound event detection in synthetic domestic environments,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2020, pp. 86–90.
- [62] S. Hershey, D. P. Ellis, E. Fonseca, A. Jansen, C. Liu, R. C. Moore, and M. Plakal, “The benefit of temporally-strong labels in audio event classification,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2021, pp. 366–370.
- [63] Y. Gong, Y.-A. Chung, and J. Glass, “PSLA: Improving audio tagging with pretraining, sampling, labeling, and aggregation,” *IEEE/ACM Trans. Audio, Speech Lang. Process.*, vol. 29, pp. 3292–3306, 2021.



Xuenan Xu received his B.S. degree from Shanghai Jiao Tong University in 2019. He is currently working towards his Ph.D. degree with the Department of Computer Science and Engineering, Shanghai Jiao Tong University. His main research interests include audio understanding and generation and multi-modal learning.



Ziyang Ma received the B.Eng. degree in computer science from Shandong University in 2022. He is currently working toward the Ph.D. degree with the Department of Computer Science and Engineering, Shanghai Jiao Tong University. His research interests focus on speech, language, audio and music processing with Self-Supervised Learning (SSL) and Large Language Model (LLM).



Mengyue Wu received her B.S. and B.A. from Beijing Normal University in 2011 and was awarded Ph.D. from the University of Melbourne in 2017. She is currently an Assistant Professor in Computer Science and Engineering Department, Shanghai Jiao Tong University, China. Her main research interests lie in the area of audio- and language- based human machine interaction including audio processing, multimedia processing, and medical application of these technologies.



Kai Yu is a professor at Computer Science and Engineering Department, Shanghai Jiao Tong University, China. He received his B.Eng. and M.Sc. from Tsinghua University, China in 1999 and 2002, respectively. He then joined the Machine Intelligence Lab at the Engineering Department at Cambridge University, U.K., where he obtained his Ph.D. degree in 2006. His main research interests lie in the area of speech-based human machine interaction including speech recognition, synthesis, language understanding and dialogue management. He is a member of the IEEE Speech and Language Processing Technical Committee.