Supplementary Materials: Modality-Independent Teachers Meet Weakly-Supervised Audio-Visual Event Parser

Anonymous Author(s)
Affiliation
Address
email

A Caption Construction and Threshold Determination in VALOR

We provide detailed explanations on how we devise input captions for each event to be used with CLIP and CLAP. For the CLIP's input captions, we add the prompt "A photo of" before each event name and modify some of the captions to make them sound reasonable, *e.g.* changing "A photo of speech" to "A photo of people talking." As for CLAP, we add the prompt "This is a sound of" before each event name. All input captions devised for CLAP and CLIP are included in Table 1 for reference.

Furthermore, the determination of class-dependent threshold values, $\boldsymbol{\theta}^{\text{CLIP}}$ for CLIP and $\boldsymbol{\theta}^{\text{CLAP}}$ for CLAP, is based on the visual and audio segment-level F-score, respectively. This score is achieved by comparing the segment-level pseudo labels generated by the respective models against the ground truth labels.

Table 1: **The List of Input Captions and Thresholds for CLIP and CLAP.** We add the prompt "A photo of" before each event name to make CLIP's input captions and the prompt "This is a sound of" to make CLAP's input captions.

Events	Input Captions		thresholds $\boldsymbol{\theta}$	
	CLIP	CLAP	$\boldsymbol{\theta}^{\text{CLIP}}$	$\boldsymbol{\theta}^{\text{CLAP}}$
Speech	A photo of people talking.	This is a sound of speech	20	0
Car	A photo of a car.	This is a sound of car	15	0
Cheering	A photo of people cheering.	This is a sound of cheering	18	1
Dog	A photo of a dog.	This is a sound of dog	14	4
Cat	A photo of a cat.	This is a sound of cat	15	6
Frying_(food)	A photo of frying food.	This is a sound of frying (food)	18	-2
Basketball_bounce	A photo of people playing basketball.	This is a sound of basketball bounce	18	4
Fire_alarm	A photo of a fire alarm.	This is a sound of fire alarm	15	4
Chainsaw	A photo of a chainsaw.	This is a sound of chainsaw	15	2
Cello	A photo of a cello.	This is a sound of cello	15	2
Banjo	A photo of a banjo.	This is a sound of banjo	15	2
Singing	A photo of people singing.	This is a sound of singing	18	1
Chicken_rooster	A photo of a chicken or a rooster.	This is a sound of chicken, rooster	15	2
Violin_fiddle	A photo of a violin.	This is a sound of violin fiddle	15	3
Vacuum_cleaner	A photo of a vaccum cleaner.	This is a sound of vacuum cleaner	15	0
Baby_laughter	A photo of a laughing baby.	This is a sound of baby laughter	15	2
Accordion	A photo of an accordion.	This is a sound of accordion	15	2
Lawn_mower	A photo of a lawnmower.	This is a sound of lawn mower	15	2
Motorcycle	A photo of a motorcycle.	This is a sound of motorcycle	15	0
Helicopter	A photo of a helicopter.	This is a sound of helicopter	16	2
Acoustic_guitar	A photo of a acoustic guiter.	This is a sound of acoustic guitar	14	-1
Telephone_bell_ringing	A photo of a ringing telephone.	This is a sound of telephone bell ringing	15	2
Baby_cry_infant_cry	A photo of a crying baby.	This is a sound of baby cry, infant cry	15	3
Blender	A photo of a blender.	This is a sound of blender	15	3
Clapping	A photo of hands clapping.	This is a sound of clapping	18	0

Submitted to 37th Conference on Neural Information Processing Systems (NeurIPS 2023). Do not distribute.

B More AVVP Implementation Details

In our experiments, we apply two different model architectures: 1) the standard model architecture, which is employed in VALOR, consists of a single HAN layer with a hidden dimension of $= 512$; 2) the variant model architecture, which is used in VALOR+ and VALOR++, is a thinner yet deeper HAN model, comprising four HAN layers with a hidden dimension of $= 256$. Both models contain approximately the same number of trainable parameters. The above details are summarized in Table 2 below. The models are trained using the AdamW optimizer, configured with $\beta_1 = 0.5$, $\beta_2 = 0.999$, and weight decay set to 0.001. We employ a learning rate scheduling approach that initiates with a linear warm-up phase over 10 epochs, rises to the peak learning rate, and then decays according to a cosine annealing schedule to the minimum learning rate. We set the batch size to 64 and train for 60 epochs in total. We clip the gradient norm at 1.0 during training. We attach the code containing our model and loss functions to the supplementary files.

Table 2: **Two Different HAN Model Architectures.** The "standard" model architecture is used in VALOR. The "variant" model architecture is used in VALOR+ and VALOR++.

HAN model	standard	variant
Model Arch. Hyper-parameters		
hidden dim	512	256
hidden layers	1	4
trainable params	5.1M	5.05M
Training Hyper-parameters		
peak learning rate	1e-4	3e-4
min learning rate	1e-6	3e-6

C Additional Analysis: The Fidelity of Our Segment-level Labels

To examine the fidelity of our generated segment-level pseudo labels in both modalities, we compare our labels, $\hat{\boldsymbol{y}}_t^a$ and $\hat{\boldsymbol{y}}_t^v$, with naive segment-level labels, which are obtained by copying the video-level label $\boldsymbol{y}$ of a video and assigning it to all segments. In other words, we assume that an event occurs in both modalities and all segments if it occurs in the video. As depicted in Table 3, the segment-level F-scores of our generated segment-level audio and visual pseudo labels are superior to those of the respective naive ones. Notably, our segment-level visual F-score is 10 points higher than the naive one. Moreover, we evaluate the fidelity of the audio-visual event labels by performing element-wise AND operation on the segment-level audio and visual labels. The segment-level F-score of our audio-visual labels significantly surpasses that of the naive ones. These findings present the reliability of our segment-level pseudo labels, which can provide more accurate segment-level information to facilitate model training.

Table 3: **The Fidelity of Our Segment-level Labels.** We compare the segment-level labels generated from our method with the naive segment-level labels directly copied from the video-level labels, where we assume the events occurring in a video will occur in both modalities and every segment. We can see that the segment-level labels generated by our method VALOR are more accurate than the naive segment-level labels.

Methods	Audio	Visual	Audio-Visual
video labels as segment labels	79.33	69.30	60.69
VALOR-generated segment labels	84.92 (+5.59)	82.80 (+13.50)	76.37 (+15.68)

D VALOR with Pseudo Label Denoising

In this section, we explore the application of Pseudo Label Denoising (PLD), as proposed in VPLAN [5], to refine the segment-level labels generated by our method. The hyperparameters for the PLD, specifically $K = 4$ and $\alpha = 6$ for the visual modality, and $K = 10$ and $\alpha = 10$ for the audio modality, are chosen based on the visual and audio F-scores on the validation split. From

Table 4, we can see that PLD is less effective in refining our pseudo labels compared to VPLAN's pseudo labels (+1.5 v.s. +2.22 in segment-level metrics and +2.28 v.s. +3.41 in event-level metrics). However, it's worth noting the visual segment-level labels derived from our method **before** PLD are nearly as accurate as those from VPLAN **after** PLD (72.34 v.s. 72.51). Although we do implement PLD in the audio modality, no noticeable improvement is recorded for any audio pseudo labels. Referring to Table 5, the model trained with our denoised segment-level labels improves marginally. Nevertheless, we outperform VPLAN on Type@AV and Event@AV F-scores in segment-level and event-level metrics.

Table 4: **PLD refinement.** We evaluate the fidelity (F-score) of the segment-level pseudo labels before and after pseudo label denoising (PLD). PLD is less effective in refining our pseudo labels compared to VPLAN's pseudo labels. However, the visual segment-level labels generated from our method **before** PLD are nearly as accurate as those generated from VPLAN **after** PLD (72.34 v.s. 72.51). Results are reported on the validation split.

Methods	PLD	Audio		Visual	
		Seg	Event	Seg	Event
VALOR	✘	80.78	71.69	72.34	66.36
VALOR	✔	80.78	71.69	73.84 (+1.5)	68.64 (+2.28)
VPLAN [5]	✘	-	-	70.29	64.68
VPLAN [5]	✔	-	-	72.51 (+2.22)	68.09 (+3.41)

Table 5: **Results of Training with Denoised Labels.** We outperform VPLAN on Type@AV and Event@AV F-scores in segment-level and event-level metrics with and without PLD. Results are reported on the testing split.

Methods	PLD	Segment-level		Event-level	
		Type	Event	Type	Event
VALOR	✘	62.0	61.5	56.7	54.2
VALOR	✔	62.2	61.9	56.6	53.7
VPLAN [5]	✘	61.2	59.4	54.7	50.8
VPLAN [5]	✔	62.0	60.1	55.6	51.3

E Qualitative Comparison with Previous AVVP Works

Aside from quantitative comparison with previous AVVP works, we perform a qualitative evaluation as well. In Figure 1, we qualitatively compare with the baseline method HAN [4] and the state-of-the-art method JoMoLD [1]. In the top video example, JoMoLD erroneously predicts a "Speech" audio event, while all other methods accurately identify the audio events. In the bottom example, HAN produces identical temporal annotations for the "Speech" event in both modalities, despite the event only occurring audibly. Additionally, our method provides annotations that more closely align with the ground truth than either HAN or JoMoLD when the events occur intermittently, which is a challenging task for models to generate accurate predictions.

F More Audio-Visual Event Localization Details

Baseline Method We adopt the baseline model HAN to aggregate unimodal and cross-modal temporal information as we have done in the AVVP task. For brevity, we introduce our baseline method from the procedure after feature aggregation. The segment-level audio features and visual features, $\hat{\boldsymbol{f}}_t^a$ and $\hat{\boldsymbol{f}}_t^v$ ($\in \mathbb{R}^d$), output from HAN are processed through a 2-layer feed-forward network (FFN) to yield the unimodal segment-level predictions (logits), $\boldsymbol{z}_t^a$ and $\boldsymbol{z}_t^v$ ($\in \mathbb{R}^{(C+1)}$), respectively:

$$\boldsymbol{z}_t^m = \text{FFN}(\hat{\boldsymbol{f}}_t^m),\ m \in \{a, v\}, \tag{1}$$

where $C+1$ denotes the number of event classes and the "background" event. Since segment-level labels are not available in the weakly-supervised setting, we simply infer video-level logits

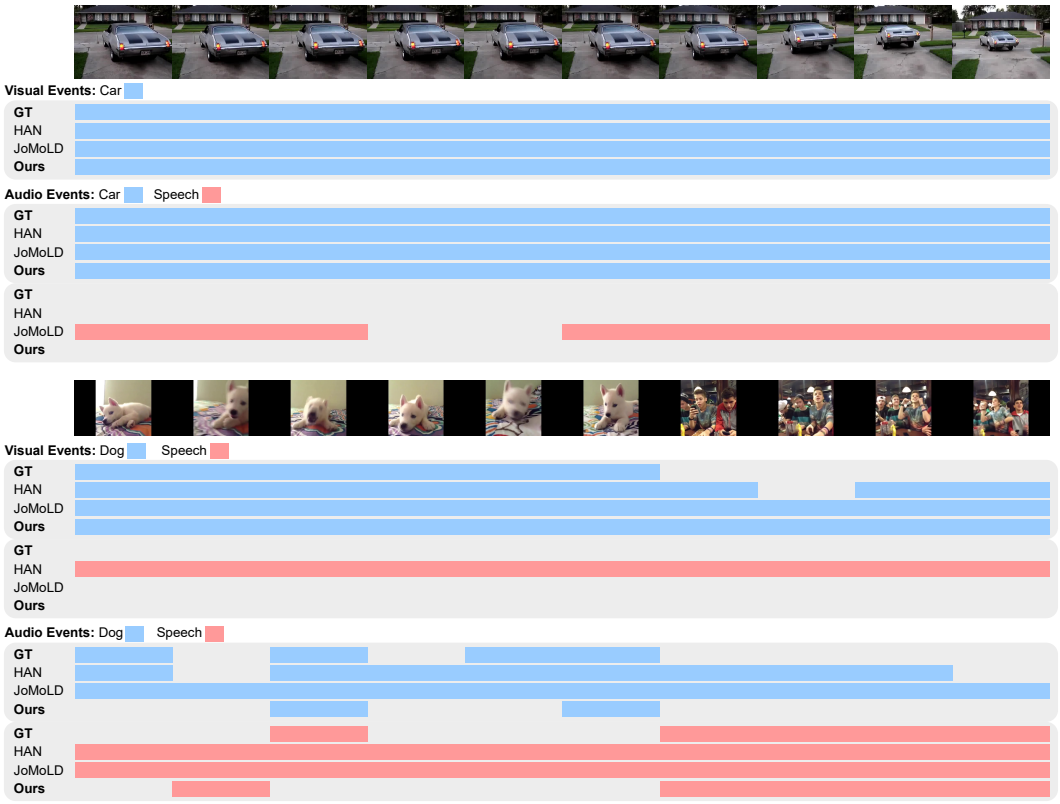

Figure 1: **Qualitative Comparison with Previous AVVP Works.** "GT" denotes the ground truth annotations. We compare with HAN [4] and JoMoLD [1]. In general, the predictions generated by our method VALOR are more accurate than those produced by the other methods.

$\boldsymbol{z} \in \mathbb{R}^{C+1}$ by averaging all logits over time dimension t and modality dimension m. Finally, the binary cross-entropy loss is applied to train the model:

$$\mathcal{L}_{\text{video}}^{\text{ave}} = \text{BCE}(\text{Sigmoid}(\boldsymbol{z}), \boldsymbol{y}), \quad \boldsymbol{z} = \frac{1}{2T} \sum_t \sum_m \boldsymbol{z}_t^m \tag{2}$$

Harvesting Training Signals The main idea of our method is to leverage large-scale open-vocabulary pre-trained models to provide modality-specific segment-level pseudo labels. We elaborate on how these pseudo labels are generated. Initially, segment-level audio logits and visual logits, $\boldsymbol{z}_t^{\text{CLAP}}$ and $\boldsymbol{z}_t^{\text{CLIP}}$ ($\in \mathbb{R}^C$), are generated in a manner identical to the AVVP task. Then, we use two sets of class-dependent thresholds, $\boldsymbol{\phi}^{\text{CLAP}}$ and $\boldsymbol{\phi}^{\text{CLIP}}$ ($\in \mathbb{R}^C$), to construct the uni-modal segment-level labels $\hat{\boldsymbol{y}}_t^a$ and $\hat{\boldsymbol{y}}_t^v$ ($\in \mathbb{R}^C$), respectively:

$$\hat{\boldsymbol{y}}_t^m = \boldsymbol{y} \wedge \{\boldsymbol{z}_t^P > \boldsymbol{\phi}^P\}, \ (m, P) \in \{(v, \text{CLIP}), (a, \text{CLAP})\} \tag{3}$$

In addition, we append an additional event "background" to the end of the segment-level labels $\hat{\boldsymbol{y}}_t^v$ to expand the dimension to $\mathbb{R}^{C+1}$. If $\hat{\boldsymbol{y}}_t^m$ consists solely of zeros, we assign the last dimension ("background") a value of one; otherwise, we assign it a value of zero. In other words, if an event could possibly occur in a video and the pre-trained model has a certain confidence that the event is present in a specific video segment, that segment will be labeled as containing the event; otherwise, the segment will be labeled as "background". Having prepared the segment-level pseudo labels $\hat{\boldsymbol{y}}_t^a$ and $\hat{\boldsymbol{y}}_t^v$, we compute binary cross-entropy loss in individual modality and combine them to optimize the whole model instead of using the video-level loss $\mathcal{L}_{\text{video}}^{\text{ave}}$:

$$\mathcal{L}_{\text{VALOR}}^{\text{ave}} = \text{BCE}(\text{Sigmoid}(\boldsymbol{z}_t^a), \hat{\boldsymbol{y}}_t^a) + \text{BCE}(\text{Sigmoid}(\boldsymbol{z}_t^v), \hat{\boldsymbol{y}}_t^v) \tag{4}$$

Dataset & Evaluation Metrics The *Audio-Visual Event (AVE) Dataset* [3] is composed of 4143 10-second video clips from AudioSet [2] that cover 28 real-world event categories, such as human

activities, musical instruments, vehicles, and animals. Each clip contains an event and is uniformly split into ten segments. Each segment is annotated with an event category if the event can be detected through both visual and auditory cues; otherwise, the segment is labeled as background. However, we only use video-level labels indicating which event occurs in the video during training. We follow [3] to split the AVE dataset into training, validation, and testing split and report the results on the testing split. Following the previous work [3], we use the accuracy of segment-level event category predictions as the evaluation metric.

Implementation Details The pre-trained large ViT-based CLIP and R(2+1)D are used to extract 2D and 3D visual features, respectively, which are then concatenated to represent low-level visual features. The pre-trained HTSAT-RoBERTa fusion-based CLAP is used to extract audio features. We adopt the standard HAN model (1-layer and 512-dim) in this task. AdamW optimizer with $\beta_1 = 0.5$, $\beta_1 = 0.999$, and weight decay $= 1e-3$ is used to train the model. A learning rate scheduling of linear warm-up for 10 epochs to the peak learning rate of $3e-4$ and cosine annealing decay to the minimum learning rate of $3e-6$ is adopted. The batch size and the number of total training epochs are 16 and 120, respectively. We clip the gradient norm at 1.0 during training.

References

[1] Haoyue Cheng, Zhaoyang Liu, Hang Zhou, Chen Qian, Wayne Wu, and Limin Wang. Joint-modal label denoising for weakly-supervised audio-visual video parsing. In *ECCV*, 2022. 3, 4

[2] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *ICASSP*, 2017. 4

[3] Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *ECCV*, 2018. 4, 5

[4] Yapeng Tian, Dingzeyu Li, and Chenliang Xu. Unified multisensory perception: Weakly-supervised audio-visual video parsing. In *ECCV*, 2020. 3, 4

[5] Jinxing Zhou, Dan Guo, Yiran Zhong, and Meng Wang. Improving audio-visual video parsing with pseudo visual labels. *arXiv preprint arXiv:2303.02344*, 2023. 2, 3