Supplementary Material:
VGGSound*er*: Audio-Visual Evaluations for Foundation Models

A. VGGSound*er*: Relabelling VGGSound

In the main paper, we highlighted several critical shortcomings of VGGSound, such as co-occurring classes, partially overlapping class definitions, multiple classes per sample, and modality misalignment. This appendix provides additional details about the relabelling process for obtaining the VGGSound*er* benchmark, addressing the specific issues identified in VGGSound.

A.1. Labelling of the gold-standard subset

As described in Sec. 4, we started by creating a high-quality reference subset (gold-standard) for reliable label verification. Four experienced computer vision researchers manually annotated randomly selected 10-second videos from the VGGSound test set. Annotators labelled classes clearly present either audibly, visually, or both. We ensured full class coverage by continuing the annotation process until all classes appeared at least once, resulting in 417 samples. These annotations were merged using majority voting. The annotation interface employed in this phase is illustrated in Fig. 5.

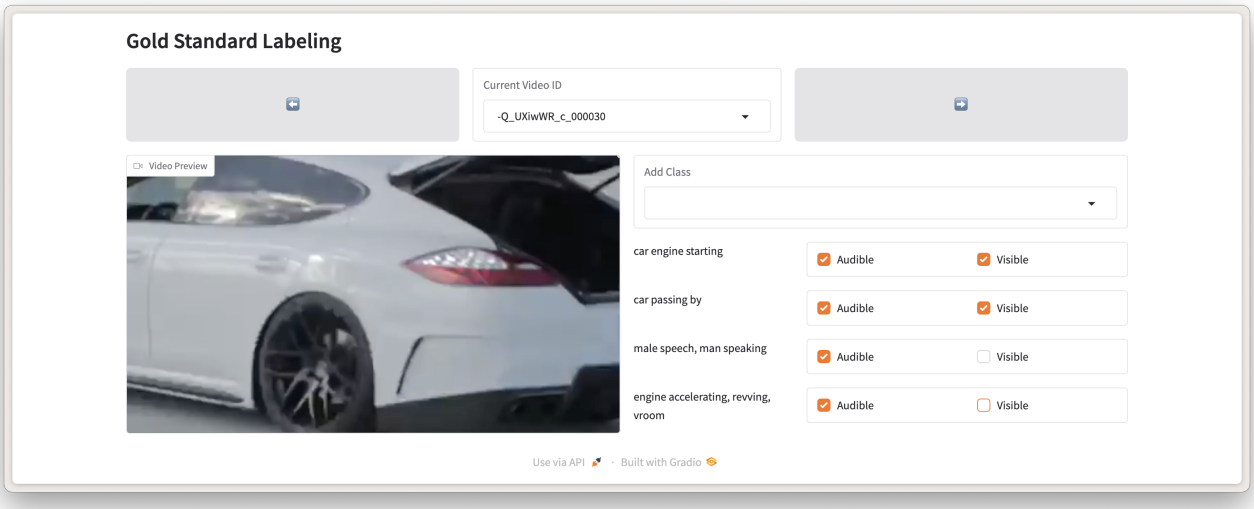


Figure 5. Interface used to annotate the gold standard set in-house.

The annotators were instructed to try to identify all audible and visible classes in the video, including hard cases when background music contains several instruments that compose the melody. For instance, a common instrument for the country music genre would be `playing the drum kit`, `female singing`, `male singing`, `playing the bass guitar`, `playing electric guitar` etc. The annotators are expected to do their best to identify all of the instruments.

Gold-standard samples serve as high-quality annotations for further labellers' cross-validation and automatic quality assessment. If a labeller shows a high agreement score with the gold-standard labels, we expect them to have high-quality labels outside of the gold-standard subset.

While analysing gold-standard labels, we made several interesting observations (see Tab. 5):

1. There is a significant portion of samples in the gold-standard set for which the original VGGSound labels (24.46%) are absent.
2. The proportion of classes that are only audible across all samples is significantly higher than that of the visible ones.

Metric	Value
Samples	417
Original class correct	283 (67.87%)
Original class audible	39 (9.35%)
Original class visible	22 (5.88%)
Original class absent	102 (24.46%)
Original class is only class	71 (17.03%)
Classes total	309
Classes only visible	6
Classes only audible	25
Average labels added per sample	1.39

Table 5. Relabelling statistics for the gold-standard subset.

While we cannot fix the second issue without substituting the dataset, the first issue quantifies the error introduced by VGGSound and its automatic labelling and verification and can be eliminated with human labelling.

We ran a second round of gold-standard annotations where one computer vision expert checked all 15446 samples and annotations in the VGGSound test set for their validity and enriched the correct labels with modality annotations. The interface for this annotation is illustrated in Fig. 6.

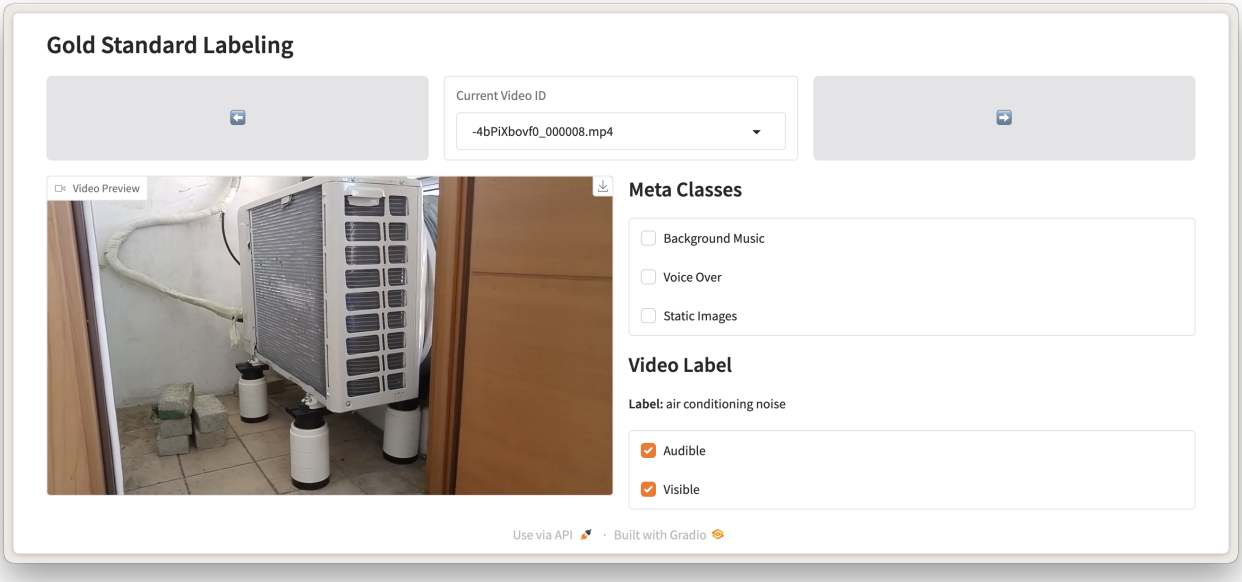


Figure 6. Interface used in-house to annotate the original labels in the VGGSound test set.

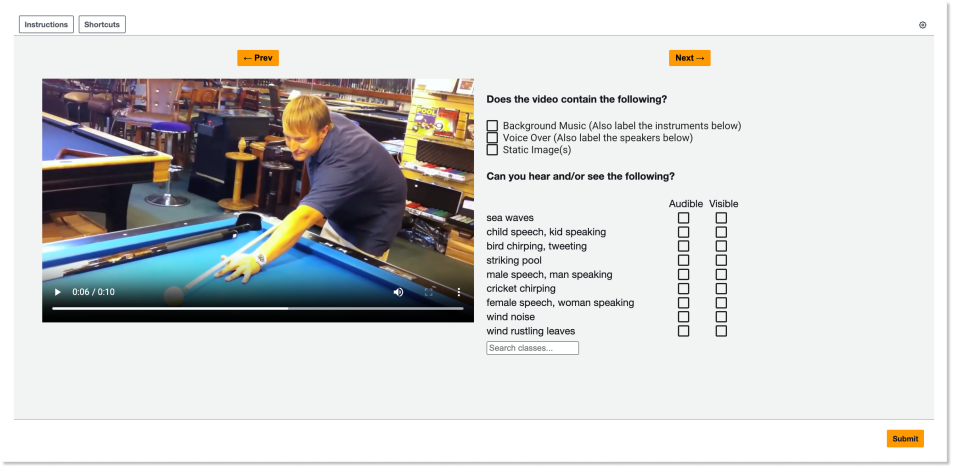
The second set of gold-standard labels firstly enriched the original VGGSound labels with modality annotations, but most importantly confirmed and further improved the estimates in Tab. 5 resulting in the following observation:

Around 48.43% of the original VGGSound test samples have either incorrect target labels or misaligned modalities.

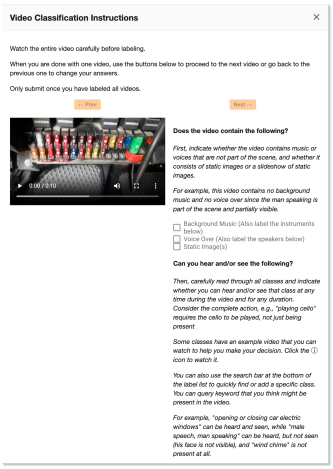
The two sets of gold-standard annotations, while having mixed reliability (cross-validation with four people vs. one person), serve as a strong grounding signal for our subsequent MTurk annotation pipelines.

A.2. Label proposals

To effectively scale human annotations to the entire test set, and to simplify the job for MTurk annotators, we introduced a label proposal generation strategy that combines state-of-the-art audio-visual model predictions with label heuristics. We



(a) **Example labelling interface for one video sample.**



(b) **Labelling instructions**

Figure 7. **Labelling interface and instructions for our full annotation pipeline that we ran on MTurk.** (a) Crowd workers are presented with a 10-second long video clip from the VGGSound test set, along with label proposals. They are tasked to select if those or additional VGGSound classes are audible or visible in the video clip. Furthermore, the workers are asked about meta-classes, such as background music, voice-over, and static images. They also have the option of searching for new classes that are missing in the proposals. (b) Labelling instructions provided to workers on Amazon Mechanical Turk before labelling the first video sample.

considered the following steps in our label proposals:

1. **Model predictions:**
   - We provide the original VGGSound label, extended with modality annotations curated by an in-house labeler, as well as the top-1 predictions of the following models [2] with visual and audio-visual inputs:
     - CAV-MAE
     - AVSiam
     - Equi-AV
     - DeepAVFusion
     - Gemini 1.5 Flash
     - Gemini 1.5 Pro
   - We further included the top-5 predictions when using audio inputs from the same models.
2. **Consensus labels:**
   - We created a secondary pool from the top-10 predictions across all modalities from AVSiam, CAV-MAE, and Equi-AV. Additionally, labels associated with the highest 60,000 logits or probabilities across the dataset were added.
   - Labels were proposed from this pool if at least two models independently agreed on their presence.
3. **Common classes:**
   - Regardless of model predictions, we always proposed frequently occurring classes such as:

`wind noise`, `wind rustling leaves`, `male speech, man speaking`, `female speech, woman speaking`, `child speech, kid speaking`, `bird chirping, tweeting`, `cricket chirping`, `sea waves`.

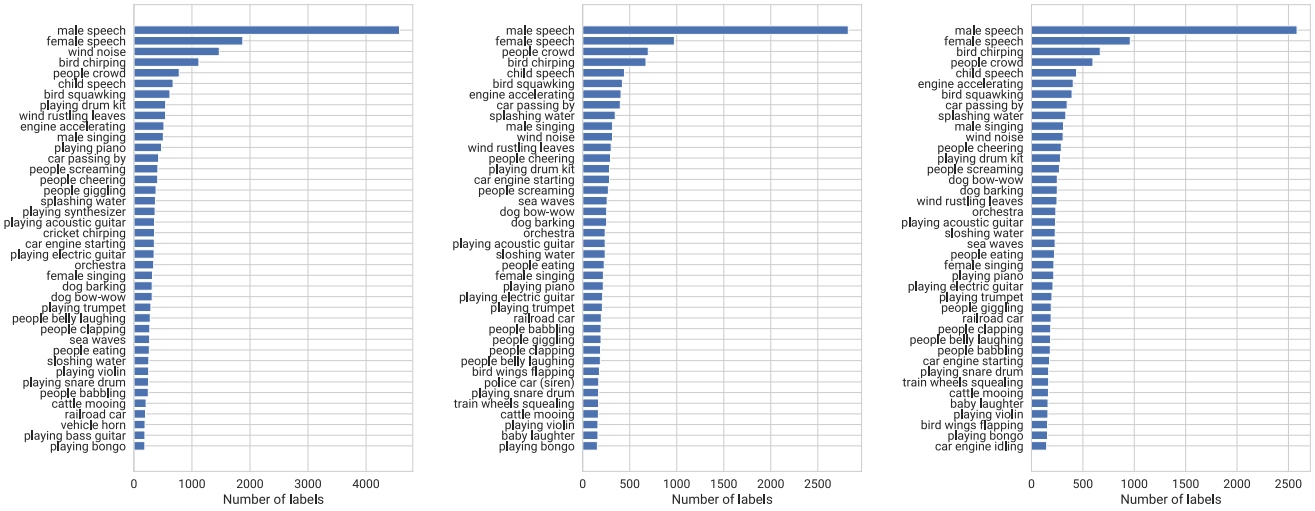
This strategic combination ensured an average of 30 proposals per video, achieving approximately 93% recall relative to the gold-standard set annotations.

A.3. Human labelling

Following our proposal strategy, we conducted extensive human annotation via Amazon Mechanical Turk (MTurk) to verify and expand the automatically generated proposals:

- **Worker qualifications:** Participation was restricted to AMT Masters with an approval rate above 98%.

[2] Gemini 2.0 Flash, VideoLLaMA-2, Unified-IO-2, Panda-GPT, and Ola were not used when the proposals were generated.



(a) *Audible* classes. (b) *Visible* classes. (c) *Audible and visible* classes.

Figure 8. Class label frequency in VGGSound*er* by modality.

- **Annotation interface:** Annotators reviewed each video to confirm the presence and modality (audible, visible, or both) of proposed labels. They could also suggest additional missing labels. Workers received detailed instructions (see Fig. 7b), and the annotation interface used is presented in Fig. 7a.
- **Quality control:** Videos were grouped into batches of 20, each containing two gold-standard samples as catch trials. Batches scoring below 25% F1-score on these catch trials were rejected and reassigned.

Our final pipeline ensured that each sample in the VGGSound*er* benchmark was labeled by at least three high-quality annotators.

A.4. Automatically added classes

To resolve overlapping and ambiguous class definitions discussed in Sec. 3, we automatically included synonymous classes and related superclasses whenever subclasses were deemed present. For instance, identifying `cow lowing` led to automatically including the superclass `cattle mooing`. A detailed overview of these automatically added classes and their relationships is provided in Tab. 6.

Class	Added Class
timpani	tympani
tympani	timpani
dog barking	dog bow-wow
dog bow-wow	dog barking
Barn swallow calling	Bird chirping, tweeting
Eagle screaming	Bird squawking
Canary calling	Bird chirping, tweeting
Mynah bird singing	Bird chirping, tweeting
Magpie calling	Bird squawking
Warbler chirping	Bird chirping, tweeting
Wood thrush calling	Bird chirping, tweeting
Goose honking	Bird squawking
Duck quacking	Bird squawking
Penguins braying	Bird squawking
Baltimore oriole calling	Bird chirping, tweeting
Crow cawing	Bird squawking
Airplane flyby	Airplane
Baby babbling	People babbling
Bull bellowing	Cattle mooing
Cow lowing	Cattle mooing
People eating noodle	People eating
People eating apple	People eating
Eating with cutlery	People eating
Bathroom ventilation fan running	Running electric fan
Striking bowling	Bowling impact

Table 6. Class mapping used to automatically add synonymous classes and superclasses.

B. Class label frequency in VGGSound*er*

Fig. 8 shows the frequency of the 40 most common class labels by modality. We observe that the label distribution appears to be very similar for visible classes and for classes that are audible and visible. This matches the label modality distribution in Figure 3B in the main paper. Furthermore, we observe that the class label `male speech` is occurs more frequently than `female speech`.

C. Model evaluations and input prompts

This section provides additional details about the evaluated models, input prompts and evaluation methodology used in the zero-shot and LLM-assisted evaluations described in Sec. 5 of the paper. Specifically, we detail the prompts and methods for generating classification predictions for the models in the Gemini family [73] and for the open-source foundational models VideoLLaMA-2 [21], Unified-IO 2 [53], PandaGPT [72], and Ola [52].

C.1. Models

CAV-MAE [33] combines contrastive learning with masked data modelling to obtain strong audio-visual embeddings, used for downstream retrieval and classification tasks. We use the multi-modal CAV-MAE-Scale+ model, pretrained on AudioSet and fine-tuned on VGGSound. Following [33], unimodal and multi-modal variants use original pretrained model but we fine-tune them on VGGSound only using the respective modality.

DeepAVFusion [57] integrates complementary features from the audio and visual modalities using a deep fusion mechanism, enhancing joint processing for classification tasks. We use publicly available checkpoints for unimodal and multi-modal models pre-trained on AudioSet and we then fine-tune them on VGGSound.

AV-Siam [49] uses a two-stream network to learn joint embeddings from audio and visual data. By maximising similarity for corresponding pairs and minimising it for non-corresponding pairs, the model captures meaningful relationships between modalities. We use public checkpoints of AV-Siam pre-trained on AudioSet, to then fine-tune it on VGGSound.

Equi-AV [43] is a transformer-based model that focusses on learning invariant embedding representations through an equivariant learning approach, making it robust to input variations. Again, we fine-tune original model pre-trained on AudioSet using unimodal or multi-modal VGGSound data.

Gemini 1.5 Flash, Gemini 1.5 Pro and Gemini 2.0 Flash [73] are mixture-of-experts transformer models that process both audio and visual information. For classification, the models are prompted to output class labels from the VGGSound class list that match the input video clip, along with a caption. Unlike models trained on VGGSound, the Gemini models are assumed to be free from VGGSound-specific biases. The complete input prompts are provided in Appendix C.

VideoLLaMA-2.1-AV (VideoLLaMA-2) [21] is a multi-modal foundation model that ingests audio and visual information in two branches that independently process vision-language and audio-language data. The two branches are connected via a language model. VideoLLaMA-2 exhibits strong results on audio-visual question-answering and captioning tasks. Details about the model and prompts used are detailed in Appendix C.

Unified-IO-2 [53] is a 7B-parameter autoregressive encoder–decoder model that tokenises text, images, audio, and discrete actions into one shared sequence, enabling "any-to-any" understanding and generation.

Panda-GPT [72] augments a frozen Vicuna-13B language model with ImageBind encoders by using a single linear projection and LoRA adapters. These are trained on only 160k image-text instruction pairs. Despite this lightweight fine-tuning, the model follows instructions across six modalities (image/video, audio, text, depth, thermal, IMU) and can seamlessly compose their semantics in zero-shot settings.

Ola [52] is an omni-modal 7B LLM that progressively aligns modalities—starting with image–text, and then adding speech and finally audio-visual video. It uses local–global attention fusion, dual audio encoders (Whisper [68] + BEATs [17]) and sentence-wise streaming speech decoding. This staged training yields balanced, competitive accuracy for image QA, video QA, and speech recognition.

Motivation for LLM-assisted evaluation

In Sec. 5, we briefly mentioned standard classification strategies for foundation models, such as:

- Directly asking for a class without providing a list of available classes *(direct)*,

Some models, such as VideoLLaMA-2, Unified-IO-2, and PandaGPT, were pretrained on VGGSound. For certain prompts, they return valid VGGSound classes, which makes character-level comparison feasible. However, their overall performance on VGGSound*er* is low, as most outputs are synonym classes not included in the original class set.

- Prepending a list of all available classes to the classification prompt *(zero-shot)*,

Here, we try to mitigate character-level comparison issues by prepending all 309 class names before the prompt: *"Annotate the video, explain in detail what is happening in the video. Use classes from the provided list in the captioning and also add*

yours." This approach works well for closed-source foundation models but performs extremely poorly on all open-source models, most likely due to their smaller effective context window.

- Asking 309 independent questions, one per class, for every sample *(multi-prompt)*.

This strategy avoids the context length limitation. Instead of including all class names at once, we ask 309 questions per sample, each with the prompt: *"Do you see or hear the following class 'class' in the video? Answer only with yes or no."* While this pipeline yields higher classification scores, it is computationally expensive and still fails to fully capture the video understanding capabilities of most open-source foundation models.

In conclusion, all the above strategies yield low performance (e.g., low F1 scores) and fail to reliably capture a model's video understanding. To address this, we adopt a hybrid approach: we use the *zero-shot* strategy for closed-source models and introduce an *LLM-assisted evaluation* protocol for open-source foundation models.

Gemini models The Gemini models can handle long prompts very well. Thus, to generate classification predictions with models from the Gemini family, we used a zero-shot evaluation protocol. Specifically, we provided the models with an input prompt, a list of all class names in VGGSound separated with commas, and an input video file. We used the following text template:

```
{CLASSES}
{VIDEO}
Annotate the video, explain in detail what is happening in the video.  Use classes from
the provided list in the captioning and also add yours.
```

LLM-assisted evaluation We evaluated all other foundation models using LLM-assisted evaluation.
Building on similar approaches,[3] we employ Qwen3 [82] (32B quantised to 8 bits) as our LLM for evaluating the alignment between model-generated outputs and the ground truth.
Specifically, for each sample, the open-source foundation models are asked the following questions depending on the input modality:

A:
```
What actions are being performed in this audio, explain all sounds and actions in the
audio?  Please provide a short answer.
```

V/AV:
```
What actions are being performed in this video, explain all sounds and actions in the
video?  Please provide a short answer.
```

The generated answer (video/audio captioning text) and the target labels (list of classes separated with comma) are then both supplied to the Qwen3 evaluator that receives the following system prompt.

LLM system prompt

```
You are an intelligent chatbot designed for evaluating the correctness of generative outputs
 for classification pairs. Your task is to compare the predicted answer with the correct
answer and determine if they match meaningfully. Heres how you can accomplish the task:

- Focus on the meaningful match between the predicted answer and the correct answer.
- Consider synonyms or paraphrases as valid matches.
- Evaluate the correctness of the prediction compared to the answer.
- The correct answer, might contain multiple classes. Treat them independently and evaluate
the correctness of all them w.r.t predicted answer.
```

[3] We found the VideoLLaMA-2 appendix [21], PointLLM appendix [81], and the Unified-IO 2 code base https://github.com/allenai/unified-io-2/blob/502ac4d81239f82c891a9f412b000c3c8d4e2946/t5x/examples/unified_io/data/prompt_dict.py to be very useful.

Provide your evaluation only as a yes/no and score where the score is an integer value between 0 and 5, with 5 indicating the highest meaningful match.

Please generate the response in the form of a Python dictionary string where names of classes are keys and values are dictionary strings with keys pred and score, where value of pred is a string of yes or no and value of score is in INTEGER, not STRING.

DO NOT PROVIDE ANY OTHER OUTPUT TEXT OR EXPLANATION. Only provide the Python dictionary string. For example, your response should look like this:

{"male speech, man speaking": {"pred": "yes", "score": 4}, "playing banjo": {"pred": "no", "score": 0}}

Example 1.

<Question>
Identify the main sounds present in the given audio clip with a few words.

<Correct Answers>
["cat caterwauling", "cat meowing"]

<Predicted Answer>
The main sounds present in the given audio clip are:

1. A ticking sound, possibly from a clock or timer.
2. A mechanical sound, which could be from a machine or device.
3. A human voice, which is speaking in the background.

Output: {"cat caterwauling": {"pred": "no", "score": 0}, "cat meowing": {"pred": "no", "score": 0}}

Example 2.

<Question>
What actions are being performed in this audio, explain all sounds and actions in the audio? Please provide a short answer.

<Correct Answers>
["cuckoo bird calling", "mynah bird singing", "bird chirping, tweeting"]

<Predicted Answer>
The audio features a cuckoo bird calling in the distance and some chirping and tweeting from smaller birds.

Output: {"cuckoo bird calling": {"pred": "yes", "score": 5}, "mynah bird singing": {"pred": "no", "score": 0}, "bird chirping, tweeting": {"pred": "no", "score": 5}}

Example 3.

<Question>
What actions are being performed in this video, explain all sounds and actions in the video? Please provide a short answer.

<Correct Answers>
["male speech, man speaking", "playing hammond organ"]

<Predicted Answer>

Models	Accuracy ↑			$\mu \downarrow$		
	a	v	av	μ_a	μ_v	$\mu_{A \cap V}$
CAV-MAE	59.05	45.57	65.08	4.71	4.84	0.67
DeepAVFusion	40.82	27.24	53.10	4.18	3.17	0.07
Equi-AV	46.68	24.84	50.08	6.91	5.51	0.98
AV-Siam	56.91	47.27	55.25	13.17	8.92	3.92
Gemini 1.5 Flash	0.31	22.12	23.60	1.51	4.17	0.09
Gemini 1.5 Pro	1.29	25.77	21.31	1.62	5.41	0.24
Gemini 2.0 Flash	5.70	20.29	19.39	2.50	4.77	0.63
VideoLLaMA 2	27.98	17.01	21.46	11.16	2.85	1.42
Unified-IO 2	32.28	20.24	52.40	4.88	3.42	0.87
PandaGPT	5.20	7.65	8.95	4.51	4.48	0.94
OLA	10.71	8.63	14.29	7.61	4.05	0.71

Figure 9. **Performance of state-of-the-art models on VGGSound.** We report top-1 classification accuracy for different input modalities (audio *A*, visual *V*, and audio and visual information *AV*). μ is modality confusion metric defined in Sec. 5

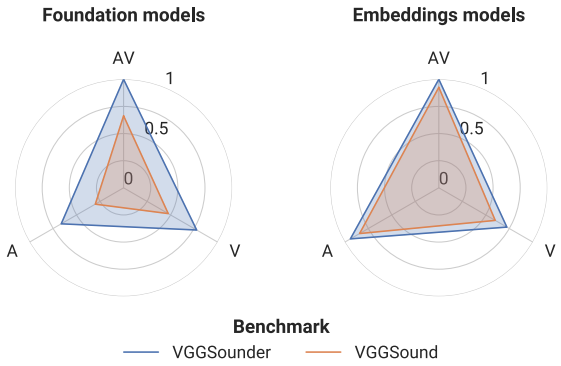


Figure 10. **Performance of state-of-the-art families on VGGSound compared to VGGSound*er*.** Radar plots illustrate the average F1-scores across modalities for two model families: "Foundation models" and "Embedding models" (Tab. 2).

```
The video shows a man who is playing regular piano and speaking with someone.

Output: {"male speech, man speaking": {"pred": "yes", "score": 5}, "playing hammond organ":
{"pred": "yes", "score": 3}}
```

User message template

```
<Question>
{QUESTION}

<Correct Answers>
{ANSWERS}

<Predicted Answer>
{CAPTION}
```

Qwen3 then outputs a Python-formatted dictionary mapping for each target class. The dictionary contains binary "pred" decision (yes/no) and a nuanced confidence score (0–5), accommodating synonymy and paraphrasing.

```
 {"male speech, man speaking": {"pred": "yes", "score": 5}, "playing hammond organ":
{"pred": "yes", "score": 3}}
```

This flexible scoring relaxes the strict label matching, yielding richer, semantically-aware assessments that better reflect human judgment and are align with recent "LLM-as-judge" [34] paradigms that have demonstrated enhanced correlation with human evaluators across a diverse set of tasks and domains.

D. Additional quantitative analysis

This appendix extends our quantitative analyses presented in Sec. 5.1 of the main paper, providing further insights into model behaviour on both VGGSound and the newly introduced VGGSound*er* benchmark.

D.1. Model performance on VGGSound

We present the classification performance of state-of-the-art models on the original VGGSound test data in Fig. 9. We observe that the multi-label hit accuracy on VGGSound*er* reported in Tab. 2 in the main paper significantly raises the performance across all models. This suggests that the models predict classes that were not present in the original VGGSound labelling, despite those being correct.

k	Model	Subset Accuracy ↑			F_1 ↑			Hit ↑		
		a	*v*	*av*	*a*	*v*	*av*	*a*	*v*	*AV*
3	CAV-MAE	0.99	0.56	1.09	39.10	35.58	42.92	81.18	72.14	82.55
	DeepAVFusion	0.22	0.08	0.68	28.07	22.43	37.36	65.15	50.86	74.75
	Equi-AV	0.55	0.24	0.34	33.50	22.78	34.06	73.82	48.13	70.78
	AV-Siam	0.83	0.77	0.74	37.36	37.05	40.91	79.32	73.21	79.83
5	CAV-MAE	0.04	0.04	0.03	35.14	30.79	36.00	87.04	78.73	87.64
	DeepAVFusion	0.00	0.00	0.00	24.91	19.48	31.06	72.24	58.48	80.38
	Equi-AV	0.01	0.01	0.00	30.06	20.44	28.62	80.64	55.67	77.09
	AV-Siam	0.02	0.04	0.02	33.13	31.88	34.67	84.81	79.57	85.53
10	CAV-MAE	0.00	0.00	0.00	25.61	21.66	24.36	91.64	85.27	92.01
	DeepAVFusion	0.00	0.00	0.00	18.36	14.23	21.06	80.35	67.87	85.70
	Equi-AV	0.00	0.00	0.00	22.39	15.24	19.84	87.41	65.49	83.77
	AV-Siam	0.00	0.00	0.00	24.15	22.12	23.94	90.11	85.86	90.99

Table 7. **Audio-visual video classification results on VGGSound*er* for $k \in \{3, 5, 10\}$.** The table is vertically grouped by k. Within each block, the four models are compared across the three metrics and input modalities.

Fig. 10 further compares the averaged F1-scores between the "Foundation model" and "Embedding model" families, highlighting that evaluations on the original VGGSound consistently underestimate model performance across all modalities when compared to evaluations on VGGSound*er*.

D.2. Co-occurrence matrix on VGGSound

To further illustrate the issue of class overlap described in Sec. 3 of our paper, we include an analysis of class co-occurrences in predictions by the CAV-MAE model [33]. Specifically, we provide a co-occurrence matrix highlighting frequent simultaneous predictions of certain classes. Notably, labels such as `playing drum kit` and `playing bass drum` are frequently predicted together, as they are not mutually exclusive. This analysis supports our identification of overlapping classes as a key limitation in the original VGGSound annotations and demonstrates the need for explicitly multi-label approaches in video classification tasks.

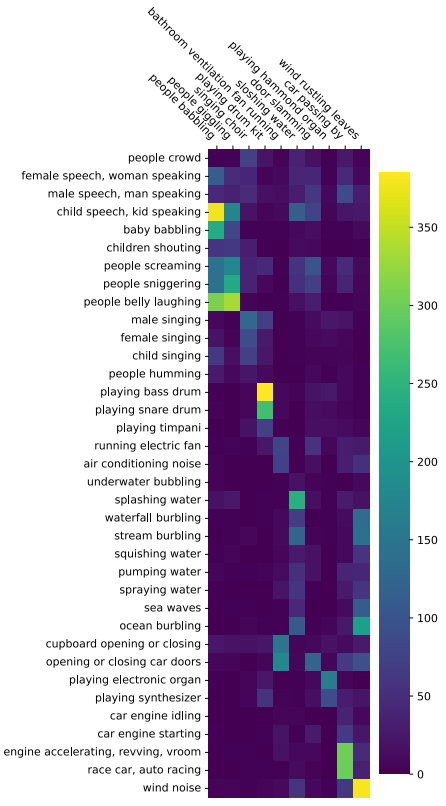


Figure 11. Co-occurrence counts among a subset of VGGSound classes, estimated on the VGGSound test set by the CAV-MAE model. Each cell indicates how frequently two classes appear together, highlighting labels that share overlapping acoustic cues (e.g., `playing drum kit` and `playing bass drum`). Best viewed zoomed in on a screen.

D.3. Classification results for other k

Tab. 7 extends the evaluation presented in the main paper by showing multi-label video classification results on VGGSound*er* for varying numbers of top-k predictions, specifically for $k \in \{3, 5, 10\}$. These additional results offer deeper insights into how model performance changes with an increasing number of predictions. Specifically, one can notice the opposite behaviour between the F1-score (goes down with k) and the Hit score (increases with k).

D.4. Performance on subsets of VGGSound*er*

To comprehensively evaluate model robustness in the presence of common confounders highlighted in Sec. 3 (i.e. meta-labels: *background music*, *static images*, and *voice over*), we present additional evaluations on distinct subsets of VGGSound*er*.

Specifically, Appendix D.4 and Tabs. 9 to 13 display the performance of state-of-the-art models on subsets only containing or fully excluding the meta-labels. These analyses confirm the importance of accounting for modality-specific and meta-label influences.

Impact of background music

	Subset Accuracy ↑			F_1 ↑					Hit ↑			μ ↓		
Model	*a*	*v*	*av*	*a*	*v*	*av*	$a(A\neg V)$	$v(V\neg A)$	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	10.80	19.17	23.84	31.03	31.29	38.60	17.83	22.26	55.96	44.57	54.02	4.26	6.92	0.87
DeepAVFusion	8.15	9.48	20.66	21.66	16.40	33.05	12.68	8.27	39.02	23.32	46.18	2.52	3.32	0.09
Equi-AV	9.11	10.11	19.70	25.33	17.86	32.13	14.75	11.52	45.67	25.44	44.98	5.39	6.09	1.09
AV-Siam	10.46	18.50	21.31	29.55	31.00	34.31	16.53	22.65	53.28	44.15	48.02	10.18	8.83	3.70
Gemini 1.5 Flash	1.15	13.91	14.75	13.31	34.57	38.36	11.49	22.10	30.47	44.24	53.50	11.53	3.65	0.78
Gemini 1.5 Pro	1.90	20.84	20.75	17.40	46.04	47.93	13.68	27.50	33.65	62.36	67.51	3.91	3.96	0.61
Gemini 2.0 Flash	1.08	11.32	10.44	11.33	32.14	32.97	9.84	21.84	18.62	39.81	43.11	2.31	4.39	0.83
VideoLLaMA 2	11.24	18.63	22.66	36.43	43.18	46.81	23.97	33.41	53.86	43.65	48.37	15.27	5.35	2.83
Unified-IO 2	9.11	13.45	24.49	28.90	29.07	44.69	20.77	22.97	42.55	28.82	52.24	5.92	5.87	1.57
PandaGPT	1.86	4.64	5.79	12.75	17.64	18.11	8.65	16.09	14.96	15.50	15.18	6.87	5.70	2.22
OLA	8.53	9.77	19.18	35.87	25.44	43.61	29.25	17.17	44.35	23.43	44.85	11.74	7.39	1.70

Table 8. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as containing *background music***. Similar to Table 1 in the main paper, we report multi-label classification metrics (subset accuracy, F_1-score, Hit accuracy, modality confusion (μ) for audio- $a(A)$, visual - $v(V)$, audio-visual - $av(AV)$, audio-only - $a(A\neg V)$ and video-only - $v(V\neg A)$ inputs.

	Subset Accuracy ↑			F_1 ↑					Hit ↑			μ ↓		
Model	*a*	*v*	*av*	*a*	*v*	*av*	$a(A\neg V)$	$v(V\neg A)$	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	13.19	19.23	24.49	34.46	34.91	42.62	13.94	19.00	62.29	53.44	64.17	3.58	6.43	0.77
DeepAVFusion	10.19	11.10	21.53	25.31	21.29	37.35	10.37	10.55	45.77	32.61	56.27	3.74	3.93	0.17
Equi-AV	11.60	10.52	20.00	29.39	20.42	34.69	12.55	10.65	53.12	31.26	52.24	6.97	7.13	1.38
AV-Siam	12.79	19.75	22.83	33.30	35.41	39.43	12.90	18.21	60.19	54.20	59.36	9.36	8.80	3.58
Gemini 1.5 Flash	1.78	14.44	16.44	14.49	36.98	42.52	15.61	21.61	32.73	47.36	59.10	10.22	4.25	0.77
Gemini 1.5 Pro	3.05	20.86	22.53	19.26	49.73	53.74	17.73	22.90	35.03	69.23	75.42	2.09	4.85	0.57
Gemini 2.0 Flash	1.85	12.54	12.69	11.80	34.08	36.45	6.19	18.90	18.51	43.83	47.72	2.39	5.43	1.00
VideoLLaMA 2	12.86	19.85	24.47	38.87	47.82	52.35	20.34	28.08	58.91	52.02	59.80	12.72	5.46	2.95
Unified-IO 2	11.94	11.56	25.61	35.31	27.92	48.89	21.38	16.53	54.39	31.05	65.11	8.70	5.16	1.79
PandaGPT	3.19	4.19	5.46	18.73	18.56	20.85	16.82	14.40	21.08	17.01	18.82	7.59	5.90	2.47
OLA	14.11	8.69	18.19	47.70	24.85	46.48	40.44	13.45	59.05	24.57	51.51	15.47	6.80	2.49

Table 9. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as *not* containing *background music***

A side-by-side inspection of the two subsets (Tab.D.4 vs. Tab.9) reveals several interesting points.

(i) Universal but modality-specific gains. Every method improves in terms of F_1 and *Hit* scores when the soundtrack is removed, that is especially clear for the *audio* input modality: for the embedding family we register jumps of up to $+5\%$ in F_1 for both audio and *visual* inputs. Consequently, joint audio–visual inputs rise in performance only slightly ($+3$–5%).

(ii) Same trend for foundation models, but with caveats. Foundation checkpoints with a meaningful audio encoder echo the pattern (Unified-IO2 $+7\%$, Ola $+12\%$); in contrast, the Gemini family remains audio-weak, suggesting that their publicly released models rely heavily on vision.

(iii) Intuition. Background music tends to mask class-specific foreground sounds; once that mask is removed the audio encoder can finally "hear" discriminative cues, whereas vision—being agnostic to the soundtrack—is affected only by the changed clip mix. With noisy audio, every model relies more on the *V* modality as a safety net, which explains why their baseline performance remains respectable despite the severe audio corruption.

Altogether, these observations confirm that background music constitutes a hard confounder, forcing models to rely on vision.

Impact of static images

Model	Subset Accuracy ↑			F_1 ↑					Hit ↑			μ ↓		
	a	*v*	*av*	*a*	*v*	*av*	$a(A\neg V)$	$v(V\neg A)$	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	22.13	19.48	27.42	38.24	27.21	37.62	35.18	15.03	61.20	34.74	47.28	4.22	6.85	0.35
DeepAVFusion	15.98	10.96	23.20	28.65	15.80	31.50	26.43	6.33	45.89	20.21	39.59	4.24	3.87	0.00
Equi-AV	19.00	10.39	22.50	32.59	14.24	31.33	30.14	9.25	52.15	18.18	39.37	5.62	4.22	0.35
AV-Siam	22.04	19.81	24.25	37.46	28.10	34.27	33.61	13.87	59.95	35.88	43.06	10.72	7.91	2.64
Gemini 1.5 Flash	1.43	13.47	15.64	9.35	29.51	33.27	8.15	18.63	20.25	30.19	40.60	10.90	4.57	0.88
Gemini 1.5 Pro	2.33	22.08	23.37	14.32	42.50	44.89	12.52	24.44	24.19	51.30	57.47	3.87	5.45	0.53
Gemini 2.0 Flash	3.85	10.88	13.88	13.54	26.91	31.64	12.82	13.75	19.53	29.87	38.31	1.93	3.87	0.88
VideoLLaMA 2	19.00	18.18	23.73	42.36	37.90	43.22	39.41	27.54	56.45	32.31	40.60	15.47	5.10	2.64
Unified-IO 2	17.92	9.58	28.47	35.87	22.12	45.95	33.45	15.93	47.49	18.34	47.80	7.38	3.34	1.05
PandaGPT	3.32	4.87	5.27	14.15	13.71	15.32	12.75	11.01	14.78	11.36	11.78	8.96	5.62	2.11
OLA	14.87	8.60	18.28	37.80	19.72	38.88	33.53	10.53	42.29	15.58	34.97	15.99	5.27	1.41

Table 10. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as containing *static images***

Model	Subset Accuracy ↑			F_1 ↑					Hit ↑			μ ↓		
	a	*v*	*av*	*a*	*v*	*av*	$a(A\neg V)$	$v(V\neg A)$	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	11.98	19.21	24.24	33.49	34.60	42.13	12.31	20.02	61.05	52.70	63.11	3.67	6.50	0.81
DeepAVFusion	9.30	10.81	21.30	24.31	20.67	36.84	8.96	10.33	44.34	31.48	55.19	3.50	3.82	0.16
Equi-AV	10.49	10.45	19.84	28.33	20.23	34.39	10.91	10.94	51.64	30.81	51.51	6.75	7.08	1.37
AV-Siam	11.57	19.52	22.49	32.24	34.95	38.76	11.26	19.51	58.76	53.23	58.06	9.44	8.85	3.64
Gemini 1.5 Flash	1.68	14.39	16.17	14.62	36.82	42.14	15.43	21.92	33.25	47.59	58.92	10.42	4.13	0.77
Gemini 1.5 Pro	2.87	20.80	22.17	19.22	49.36	53.07	17.24	23.81	35.60	68.82	74.80	2.34	4.66	0.58
Gemini 2.0 Flash	1.53	12.40	12.23	11.58	34.02	36.02	6.43	19.91	18.45	43.75	47.31	2.39	5.31	0.97
VideoLLaMA 2	12.04	19.70	24.18	38.13	47.41	51.78	18.90	29.21	58.05	51.42	58.61	13.06	5.46	2.94
Unified-IO 2	10.88	12.00	25.28	33.99	28.31	48.33	19.64	17.98	52.46	31.24	63.57	8.26	5.37	1.78
PandaGPT	2.91	4.24	5.53	17.83	18.58	20.60	15.01	15.00	20.29	17.00	18.49	7.40	5.87	2.44
OLA	12.88	8.89	18.36	46.03	25.12	46.27	38.29	14.45	57.30	24.78	51.05	14.78	6.97	2.40

Table 11. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as *not* containing *static images***

A side-by-side inspection of the "static image" split (Tab. 10) and its complement (Tab. 11) shows four salient effects.

(i) Vision takes the hit, audio steps up. Across the classic embedding models, the *visual* branch loses on average 6–7% absolute in F_1, while using *audio* inputs results in gains $+3$–6%. The same holds true for the joint audio-visual. Hit scores mirror the trend: Hit for visual inputs plunges by up to 20%, whereas Hit for audio inputs remains flat or edges upward for most of teh models.

(ii) Foundation models react unevenly. VideoLLaMA2 loses 8% on vision yet gains 3% on audio—whereas the vision-centric Gemini family suffers a broad decline, unable to compensate for the poor visual signal.

(iii) Intuition. Static clips provide far less discriminative visual evidence than genuine video, reducing motion and viewpoint cues. The audio track, in contrast, is untouched; consequently, models shift their reliance toward the acoustic channel, explaining the systematic audio gain and the parallel vision loss.

(iv) Modality-confusion drifts upward. With vision degraded, many architectures become more uncertain about which modality to trust; a few (AV-Siam) even over-correct, raising μ_A by $+1.8$ while slightly easing μ_V.

In sum, static imagery acts as the visual analogue to background music: it removes discriminative content in one modality (vision) and forces models to lean on the other (audio), exposing how well a model can rebalance modalities.

Impact of voice-over narration

Model	Subset Accuracy ↑			F_1 ↑					Hit ↑			μ ↓		
	a	*v*	*av*	*a*	*v*	*av*	$a(A\neg V)$	$v(V\neg A)$	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	2.78	14.34	17.38	26.68	28.36	35.37	11.19	25.50	51.65	43.55	53.62	4.21	7.06	0.65
DeepAVFusion	2.02	9.41	15.33	16.79	17.78	29.13	6.51	13.99	32.50	27.34	44.23	2.69	3.79	0.18
Equi-AV	3.51	7.75	14.53	22.45	15.44	28.14	9.62	10.74	43.45	23.71	42.65	7.47	6.35	1.07
AV-Siam	2.73	13.94	15.66	25.72	28.51	31.11	9.56	26.85	49.79	43.78	47.15	10.97	8.96	3.50
Gemini 1.5 Flash	5.26	9.43	10.85	29.43	30.61	34.01	27.55	23.61	63.71	40.02	48.28	22.18	4.57	1.36
Gemini 1.5 Pro	7.73	17.70	16.07	34.58	46.22	50.84	29.26	27.76	68.30	65.70	75.50	4.15	4.27	0.95
Gemini 2.0 Flash	0.72	7.40	8.36	11.87	27.47	29.85	6.10	21.62	19.95	36.38	40.57	2.79	5.81	1.30
VideoLLaMA 2	6.34	16.08	19.40	34.98	42.76	47.72	21.74	36.14	54.43	47.60	54.69	13.58	5.40	2.43
Unified-IO 2	4.69	9.08	18.15	29.81	24.80	40.94	23.17	21.65	48.61	27.13	53.44	9.49	5.16	2.02
PandaGPT	3.66	4.28	4.86	20.96	18.58	18.92	19.07	17.73	26.75	17.47	18.03	9.79	6.52	3.32
OLA	15.46	7.06	16.13	54.14	23.48	48.53	47.31	16.08	76.08	24.64	57.53	16.19	4.98	2.85

Table 12. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as containing *voice over* narrations**

Model	Subset Accuracy ↑			F_1 ↑					Hit ↑			μ ↓		
	a	*v*	*av*	*a*	*v*	*av*	$a(A\neg V)$	$v(V\neg A)$	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	14.18	19.92	25.38	34.92	35.19	42.94	15.67	18.90	62.44	53.10	63.70	3.62	6.44	0.81
DeepAVFusion	10.94	11.02	22.26	25.84	20.89	37.77	11.84	9.55	46.21	31.52	56.03	3.65	3.83	0.15
Equi-AV	12.23	10.83	20.73	29.58	20.67	35.18	13.80	10.84	52.88	31.18	52.19	6.59	7.04	1.37
AV-Siam	13.74	20.34	23.56	33.66	35.59	39.70	14.66	18.09	60.17	53.70	58.90	9.29	8.79	3.61
Gemini 1.5 Flash	1.14	15.06	16.91	12.18	37.42	42.94	12.38	21.45	27.71	47.79	59.56	8.76	4.09	0.69
Gemini 1.5 Pro	2.11	21.31	23.11	16.42	49.55	53.10	14.29	23.32	29.87	68.38	73.86	2.15	4.75	0.53
Gemini 2.0 Flash	1.84	13.04	12.87	11.68	34.66	36.73	7.38	19.23	18.32	44.10	47.84	2.31	5.17	0.92
VideoLLaMA 2	13.45	20.15	24.84	38.95	47.72	52.06	21.18	28.23	58.44	50.99	58.30	13.10	5.45	3.00
Unified-IO 2	12.37	12.29	26.46	34.79	28.58	49.28	20.85	17.37	52.60	31.17	64.26	8.05	5.30	1.71
PandaGPT	2.83	4.27	5.61	17.05	18.39	20.65	13.90	14.34	18.89	16.65	18.23	7.14	5.77	2.30
OLA	12.67	9.14	18.68	44.05	25.16	45.64	35.76	13.96	53.30	24.33	49.35	14.64	7.18	2.29

Table 13. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as *not* containing *voice over* narrations**

Considering the "voice-over" split (Tab. 12) with its complement (Tab. 13) exposes a two–way story that depends on how each model treats speech.

(i) Embedding models are confused. For all four embedding models, results when using *audio* inputs jump by roughly $+5$–10% in F_1 when the narration track is removed, and Hit for audio climbs in parallel. This confirms that spoken commentary *masks* class-specific sounds. However, once silenced, the models can finally "hear" the underlying events, again, similar to the background music meta-class.

(ii) Reduced performance for speech-centric foundation models. Gemini 1.5 and PandaGPT fail when narration disappears: F_1 for audio inputs plunges by around -17% and Hit for audio inputs drops by up to 39%. Our intuition is, that these models exploit the speech content as a shortcut.

(iii) Middle ground for broad-coverage LMMs. Unified-IO 2 and VideoLLaMA-2 are between the two extremes: they register a moderate audio lift (+4–5%) and a small visual bump (+1–2%), yielding a , $+1$–, 8% improvement in terms of F_1 score for audio-visual inputs. We hypothesise that their balanced training helps them survive the removal of speech while still profiting from the clearer acoustic scene.

(iv) Modality-confusion μ reacts in both directions. For speech-reliant models, clearer acoustics *reduce* uncertainty, whereas for event-focused encoders (trained on VGGSound) it slightly *rises* because the freshly revealing audio now dominates the fusion gate.

Taken together, voice-over narration acts as the mirror image of background music: it can be a *helpful shortcut* for speech-

aware foundation models, yet a *destructive mask* for sound classifiers trained on VGGSound.

Confounder–free subset

Model	Subset Accuracy ↑			F_1 ↑					Hit ↑			μ ↓		
	a	*v*	*av*	*a*	*v*	*av*	$a(A\neg V)$	$v(V\neg A)$	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A\cap V}$
CAV-MAE	13.51	19.53	24.90	34.80	35.59	43.21	11.71	18.99	62.87	54.68	65.26	3.52	6.42	0.80
DeepAVFusion	10.56	11.04	21.84	25.86	21.50	38.01	8.98	10.27	46.74	33.04	57.43	3.86	3.90	0.16
Equi-AV	11.75	10.73	20.19	29.57	20.97	35.17	10.67	10.77	53.42	32.22	53.11	6.85	7.31	1.42
AV-Siam	13.02	20.08	23.21	33.58	36.03	39.99	10.82	17.60	60.67	55.35	60.39	9.29	8.86	3.65
Gemini 1.5 Flash	1.27	14.93	16.90	12.86	37.67	43.46	14.05	21.55	29.32	48.58	60.70	8.79	4.13	0.71
Gemini 1.5 Pro	2.35	20.86	22.97	17.26	50.01	53.89	15.69	22.27	31.43	69.78	75.44	1.97	4.80	0.55
Gemini 2.0 Flash	1.85	13.01	12.93	11.73	34.83	37.09	5.89	18.93	18.38	44.86	48.56	2.39	5.36	0.95
VideoLLaMA 2	13.02	20.08	24.94	38.87	48.36	52.84	17.78	27.52	59.11	52.65	60.42	12.61	5.37	2.97
Unified-IO 2	11.94	11.76	25.96	35.18	28.25	49.42	18.84	16.01	54.07	31.62	66.07	8.41	5.24	1.75
PandaGPT	3.00	4.09	5.43	18.21	18.57	21.08	16.00	14.68	20.34	17.06	18.98	7.17	5.76	2.34
OLA	13.38	8.86	18.24	46.34	25.07	46.10	38.24	13.13	56.92	24.78	50.88	15.23	7.08	2.46

Table 14. **Audio-visual video classification results on the subset of VGGSound*er* that is labelled as *not* containing *background music, static images, or voice over* narrations**

The split that *simultaneously* excludes background music, static images, and voice-over narration (Tab. 14) serves as an upper-bound reference and reveals how each system performs when no major nuisance factor is present.

Removing *all* three meta-classes unlocks the highest scores yet observed and sharpens modality agreement.

D.5. Ablation study for additional labels in VGGSound*er*

In this section, we conduct an ablation study to quantify the benefits introduced by different components of our annotation pipeline described in Section 3. Specifically, we compare model performance on three variants of ground-truth labels: (a) Original VGGSound labels extended only with automatically added synonymous and superclass labels, (b) Original VGGSound labels extended exclusively with human annotations, (c) Original VGGSound labels extended comprehensively with both automatically added labels and human annotations (VGGSound*er*).
Detailed performance results in Tab. 16 and Tab. 17 demonstrate a consistent improvement across all evaluation metrics when employing the complete set of annotations (scenario c). This clearly illustrates the reduction in false-positive identifications and improved accuracy achieved through our annotation pipeline. This again highlights the importance of combining automated processes with thorough human verification in creating robust benchmarks for evaluating audio-visual models.

Author contributions This project was co-led and coordinated by ASK, DZ, TW, and AP. TW, DZ, and ASK analysed and identified issues with VGGSound. TW and DZ implemented the annotation pipeline with input from ASK, AP, and WB. TW ran annotations on MTurk and produced the final label set with support from DZ and input from ASK, AP, and WB. DZ implemented the in-house annotation pipelines with input from ASK, TW, and AP. DZ trained in-house versions of multiple models with help from ASK. DZ evaluated all models with support from ASK and input from TW and AP. TW, ASK, DZ, and AP wrote the manuscript with input from WB and MB. TW and DZ created the figures with feedback from ASK, AP, and WB. MB provided helpful feedback throughout the project.

Model	Subset Accuracy ↑			$\mathbf{F_1}$ ↑			Hit ↑			μ ↓		
	a	*v*	*av*	*a*	*v*	*av*	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A \cap V}$
Gemini 1.5 Flash	0.31	22.12	23.60	1.71	33.15	35.94	2.98	31.83	41.23	1.51	4.17	0.09
Gemini 1.5 Pro	1.29	25.77	21.31	4.43	36.41	35.62	6.11	41.72	45.70	1.62	5.41	0.24
Gemini 2.0 Flash	5.70	20.29	19.39	9.95	32.34	33.91	9.49	30.55	35.31	2.50	4.77	0.63
VideoLLaMA 2	27.98	17.01	21.46	41.32	31.46	36.80	30.05	22.72	27.90	11.16	2.85	1.42
Unified-IO 2	32.28	20.24	52.40	43.71	33.84	64.06	33.71	22.84	54.20	4.88	3.42	0.87
PandaGPT	5.20	7.65	8.95	12.68	16.83	19.55	8.54	11.23	13.30	4.51	4.48	0.94
OLA	10.71	8.63	14.29	23.33	17.81	28.86	18.06	10.95	22.41	7.61	4.05	0.71

Table 15. **Audio-visual video classification results on VGGSound + automatically added labels** inputs.

Model	Subset Accuracy ↑			$\mathbf{F_1}$ ↑			Hit ↑			μ ↓		
	a	*v*	*av*	*a*	*v*	*av*	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A \cap V}$
Gemini 1.5 Flash	1.67	14.49	16.42	14.42	36.72	41.93	32.25	46.43	57.59	10.48	4.19	0.80
Gemini 1.5 Pro	2.86	21.32	22.59	19.17	49.14	52.85	34.72	67.24	73.33	2.43	4.76	0.58
Gemini 2.0 Flash	1.83	12.83	12.66	11.75	33.97	36.00	18.29	42.71	46.44	2.35	5.34	0.95
VideoLLaMA 2	12.81	20.91	25.96	38.62	48.37	53.25	56.94	50.21	57.36	13.04	5.48	2.87
Unified-IO 2	11.58	12.09	26.31	34.35	28.63	49.79	51.44	30.52	62.38	8.15	5.32	1.74
PandaGPT	3.01	4.80	6.23	17.74	18.87	20.95	19.68	16.53	18.04	7.40	5.81	2.39
OLA	13.53	9.29	22.95	46.12	25.71	49.60	55.76	24.27	50.03	14.59	6.95	2.34

Table 16. **Audio-visual video classification results on VGGSound + human annotations**.

Model	Subset Accuracy ↑			$\mathbf{F_1}$ ↑			Hit ↑			μ ↓		
	a	*v*	*av*	*a*	*v*	*av*	*a*	*v*	*av*	μ_A	μ_V	$\mu_{A \cap V}$
Gemini 1.5 Flash	1.78	14.44	16.44	14.49	36.98	42.52	32.73	47.36	59.10	10.22	4.25	0.77
Gemini 1.5 Pro	3.05	20.86	22.53	19.26	49.73	53.74	35.03	69.23	75.42	2.09	4.85	0.57
Gemini 2.0 Flash	1.85	12.54	12.69	11.80	34.08	36.45	18.51	43.83	47.72	2.39	5.43	1.00
VideoLLaMA 2	12.86	19.85	24.47	38.87	47.82	52.35	58.91	52.02	59.80	12.72	5.46	2.95
Unified-IO 2	11.94	11.56	25.61	35.31	27.92	48.89	54.39	31.05	65.11	8.70	5.16	1.79
PandaGPT	3.19	4.19	5.46	18.73	18.56	20.85	21.08	17.01	18.82	7.59	5.90	2.47
OLA	14.11	8.69	18.19	47.70	24.85	46.48	59.05	24.57	51.51	15.47	6.80	2.49

Table 17. **Audio-visual video classification results on VGGSound + human annotations + automatically added labels**