AMUSE: Audio-Visual Benchmark and Alignment Framework for Agentic Multi-Speaker Understanding

Supplementary Material

In this supplementary material, we provide additional details about:

1 Supplementary Video
2 Additional Details About AMUSE
3 Post-Training Intuition
4 Additional Experiments
5 Additional Qualitative Results
6 User Study
7 RAFT Algorithm
8 Evaluation Details

1. Supplementary Video

In the supplementary video, we provide audio-visual examples for each task and compare the performance of different models across *zero-shot*, *guided* and *agentic* modes of evaluation. The video also explains how the perception tools are being leveraged at different instances to improve the overall model performance.

2. Additional Details About AMUSE

2.1. Dataset Construction

AMUSE is constructed through an automated pipeline by leveraging the metadata from the underlying datasets. The samples are derived from the ground-truth annotations available in the source datasets: AVA Active Speaker [2], VoxCeleb2 [1], FriendsMMC [5], AMI Meetings [3], and curated YouTube videos with manually verified metadata. Our pipeline defines explicit rules for (i) extracting valid temporal segments, (ii) mapping them to speakers or events, (iii) stitching or pairing segments across clips when required by the task, and (iv) generating template-based queries grounded in annotated timestamps, transcripts, and speaker identities. Below we describe the construction procedures for each task.

2.1.1. Speaker Temporal Grounding

For each video, we extract speech-active regions using the dataset-provided temporal annotations (e.g., AVA active-speaker labels, AMI speech segments etc). A valid grounding sample is formed by selecting a continuous utterance from a target speaker and preserving its exact start and end timestamps. The transcript for this interval is obtained by slicing the dataset transcript according to the same boundaries. All timestamps originate directly from annotation files. Questions are generated using templates that reference these grounded intervals, ensuring the textual prompt aligns perfectly with the annotated speech region.

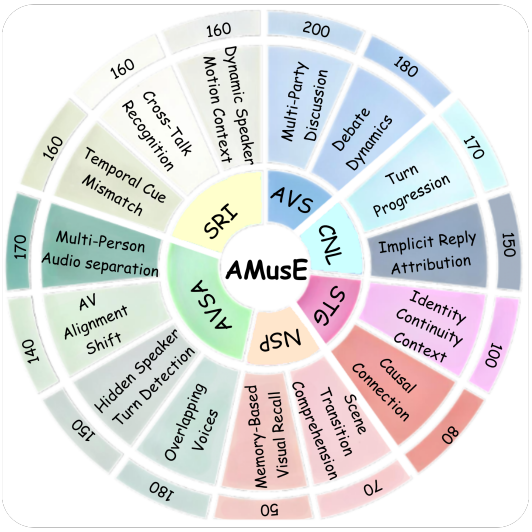

Figure 1. **AMUSE taxonomy.** Hierarchical taxonomy of 15 multi-speaker reasoning scenarios across 6 tasks highlighting the diversity and complexity of AMUSE conversations.

2.1.2. Audio-Visual Dialogue Summarization

Dialogue summarization samples are built by selecting multi-turn segments from the annotated transcripts. Each clip is formed by choosing a fixed-length temporal window (e.g., upto 40 seconds) and bundling all utterances occurring within that window. Speaker identities and utterance order are inherited from the transcript metadata. The transcript snippet is created by concatenating the turn-by-turn dialog text exactly as annotated, without modification or re-alignment. Summaries are then gener-

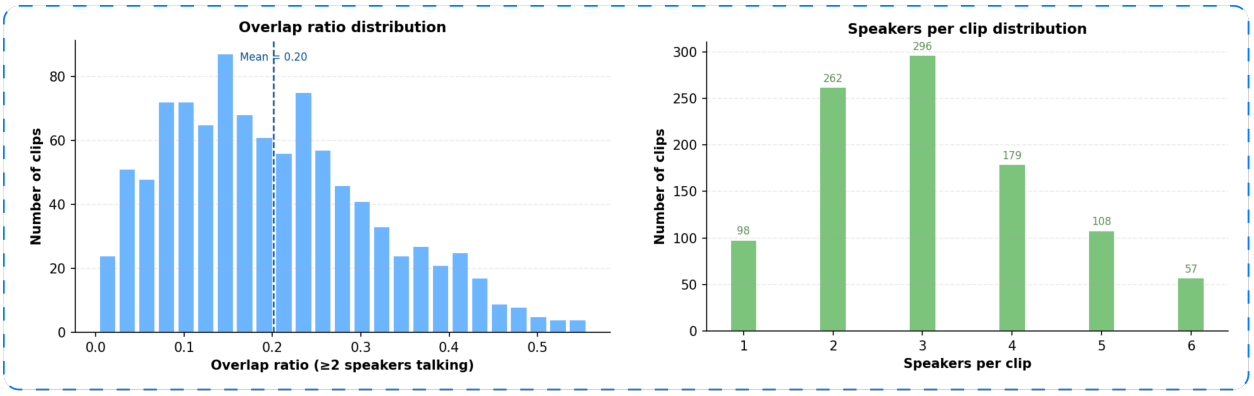

Figure 2. **Dataset statistics.** Distribution of multi-speaker overlap ratios (left) and number of visible speakers per clip (right) in AMUSE. The benchmark contains substantial speaker overlap and diverse group sizes, reflecting realistic multi-party conversational settings.

ated through structured templates that reference only the content contained in the selected segment.

2.1.3. AV Speaker Association

To build association samples, we first identify all speech segments from a given video along with their corresponding annotated speaker IDs. For each speech interval, we pair the audio-driven transcript slice with the video frames in which the same speaker appears, using the dataset-provided face track or region-of-interest metadata (e.g., bounding box indices in AVA or FriendsMMC). The pipeline directly maps the annotated speaker ID to the associated temporal window; no cross-modal embedding or recognition is involved. For each question, the template either asks the model to identify which speaker produced a given utterance or to decide whether a specific person in the scene said a particular line.

2.1.4. Next Speaker Prediction

This task is constructed using dialogue ordering rules. For any multi-speaker segment, we inspect the transcript metadata and detect the temporal boundary between two consecutive speakers. A training sample is created by selecting the preceding context (a sequence of utterances) and identifying the next annotated speaker as the prediction target. Only the transcript order and timestamps are used; no additional ASR or heuristic filtering is applied. If a segment contains overlapping or indistinguishable turns based on metadata alone, it is automatically discarded to maintain clean conversational structure.

2.1.5. Speaker Re-identification

Re-identification samples are constructed by pairing two segments of the same annotated speaker at different times within the same dataset. For each identity, we gather all timestamps where that speaker appears, select two non-overlapping intervals, and extract the corresponding transcript slices. Positive pairs reuse the same speaker ID; negative pairs are formed by pairing intervals belonging to distinct IDs. All identity information is taken strictly from the dataset's annotations (VoxCeleb2 IDs, FriendsMMC cast labels, AMI participant tags), with no external face verification. Only pairs with sufficient temporal separation or scene variation are retained to ensure meaningful samples.

2.1.6. Cross-Scene Narrative Linking

(i) Scene Selection and Grouping. We construct CSNL samples by identifying pairs or triplets of scenes from sitcoms, drama series, talk shows, and long-form conversational videos where the narrative in one segment depends on an event in another. Instead of using perceptual tools, we rely entirely on transcript metadata, speaker IDs, and annotated timestamps to detect recurring entities, callbacks, or references across non-contiguous segments. Scenes are grouped when (i) they share an annotated speaker or participant, (ii) the transcript explicitly references an earlier event, or (iii) later dialogue resolves or reacts to information introduced in a prior segment.

(ii) Template-Based Prompt and Option Design. For each grouped set of scenes, annotators write a question that explicitly requires linking the two temporally disjoint narrative events (e.g., "Why does the person in the red sweater react that way at the end of the video?"). All questions are grounded in the transcript segments extracted directly from timestamped annotations. Annotators then construct four answer choices (one correct, three distractors), ensuring that the correct answer can only be obtained by integrating information across scenes rather than relying on local, scene-specific cues. Distractors are crafted to be plausible based on individual scenes but incorrect when cross-scene reasoning is applied.

(iii) Coherence and Difficulty Verification. An annotator evaluates each CSNL item for narrative coherence, cross-

scene consistency, and difficulty using a Likert-scale rating. Items receiving low coherence scores, containing ambiguous distractors, or failing to require explicit cross-scene reasoning are removed. The final CSNL set therefore includes only those samples where the narrative link is unambiguous, grounded entirely in the annotated transcripts, and cannot be solved without integrating information across multiple scenes.

Rule-Based Engineering Pipeline. Across all tasks, the pipeline is governed by deterministic rules: (i) segments are extracted strictly using timestamp metadata; (ii) transcripts are produced by slicing and concatenating ground-truth tokens; (iii) speaker IDs originate solely from dataset annotations; (iv) multi-segment samples are stitched by pairing annotated intervals without any perceptual inference; and (v) every query is produced by a template that references only validated events, speakers, and timestamps.

Final Curation. We apply checks to ensure that the stitched intervals, paired speakers, and extracted transcripts are internally consistent (matching IDs, non-overlapping timestamps, correct temporal ordering). Ambiguous or borderline segments are pruned directly from the metadata rather than post-processed. This ensures AMUSE remains a clean, annotation-driven benchmark that reflects the structure and reliability of its underlying source datasets.

2.2. Dataset Breakdown

AMUSE integrates clips from AVA Active Speaker, VoxCeleb2, FriendsMMC, AMI Meetings, and curated YouTube videos to build six multimodal reasoning tasks. As summarized in Tab. 1, the first five tasks contain 400 samples each, while Cross-Scene Narrative Linking provides 100 multi-segment examples. Different datasets contribute complementary strengths AVA for active-speaker cues, VoxCeleb2 for identity-focused cases, FriendsMMC and AMI for rich multi-party dialogue, and YouTube for unconstrained scenarios. The dataset exhibits diverse speaker dynamics, with overlap ratios centered around 0.20 and clips containing 1–6 visible speakers (Fig. 2). The semantic wheel (Fig. 1) further shows that each task targets a distinct reasoning challenge, spanning grounding, turn-taking, association, identity persistence, and narrative linkage.

Task	AVA	VoxCeleb2	FriendsMMC	AMI	YouTube
Speaker Temporal Grounding	120	80	80	90	30
AV Dialogue Summarization	100	70	90	110	30
AV Speaker Association	130	90	70	80	30
Next Speaker Prediction	110	80	100	80	30
Speaker Re-identification	90	140	60	80	30
Cross-Scene Narrative Linking	0	0	70	0	30
Total	550	460	470	440	180

Table 1. Number of samples collected from each dataset for all six AMUSE tasks. The first five tasks contain 400 samples each, while Cross-Scene Narrative Linking contains 100 samples. Counts reflect how clips were sourced from AVA Active Speaker, VoxCeleb2, FriendsMMC, AMI Meetings, and curated YouTube videos.

2.3. Quality Control

2.3.1. Speaker Temporal Grounding

For each sample, we verify that the start and end timestamps fall strictly within valid speech-activity regions annotated in the source dataset. We also confirm that the speaker identity associated with the grounded span matches both the diarization labels and the face track present in the corresponding frames. Samples with boundary mismatches, off-by-one frame shifts, or incorrect speaker assignments are flagged and corrected by re-aligning timestamps using dataset-specific metadata (e.g., AVA Active Speaker [2] labels or AMI [3] segment boundaries).

2.3.2. Audio-Visual Dialogue Summarization

Each summary query is checked for semantic consistency with the transcript. We use a two-step validator: (i) template-level checks to ensure the summary references only events that occur within the clip, and (ii) LLM-based semantic alignment scoring to flag summaries that omit key events or hallucinate nonexistent content. Misaligned samples are regenerated by re-running the summarization template or corrected through constrained editing.

2.3.3. AV Speaker Association

We verify that the utterance-to-speaker mapping is consistent across modalities. Each audio segment is matched against the active face track using cross-modal similarity checks, which include active-speaker labels (AVA), speaker-ID metadata (VoxCeleb2 [1]), and video-based face clustering. Mismatches, such as a voice segment mapped to the wrong face, are automatically flagged using speaker-consistency rules and reprocessed by reassigning the correct identity or discarding ambiguous segments.

2.3.4. Next Speaker Prediction

For next-speaker queries, we inspect the dialogue ordering by validating that the predicted "next" speaker is temporally the next annotated speaker in the transcript and visually present in the upcoming frames. We also verify that no overlap, interruption, or missing segment disrupts the dialogue sequence. Samples with inconsistent ordering or missing participants are corrected through timeline realignment or regenerated from cleaner clips.

2.3.5. Speaker Re-identification

To ensure identity consistency across disjoint segments, each query pair is validated by comparing speaker embeddings from the original source dataset (e.g., VoxCeleb2

Tool	Inputs	Outputs	Example API-style call
Whisper	Audio file (wav/mp3); optional language tag.	Timestamped transcript; word segments; confidence scores.	asr = whisper.transcribe("clip.wav", model="large")
Pyannote	Audio waveform; diarization pipeline instance.	List of (start, end, speaker_id); overlap flags.	dia = pipeline("speaker-diarization")("clip.wav")
InsightFace	Video frames or mp4 file.	Face detections; embeddings; track IDs; bounding boxes with timestamps.	faces = insightface.get_faces("clip.mp4")
SyncNet	Video file and audio track.	AV sync offset; confidence score; mapping from speech segments to face tracks.	sync = syncnet.align(video="clip.mp4", audio="clip.wav")

Table 2. Inputs, outputs, and example API-style calls for the four perception tools used in AMUSE: Whisper (ASR), PyAnnote (speaker diarization), InsightFace (face tracking/recognition), and SyncNet (audio–visual synchronization).

[1] identity embeddings, FriendsMMC [5] cast labels, or AMI participant IDs). We confirm that the positive pairs indeed correspond to the same canonical identity, and that negative pairs have no shared speaker ID. Any identity drift (e.g., due to noisy face tracks or brief occlusions) is rectified by reassigning IDs or replacing the clip.

2.3.6. Cross-Scene Narrative Linking

Given the reasoning-heavy nature of this task, we perform additional checks to ensure narrative coherence between scenes. The validator ensures that the cross-scene question references entities or events that exist in both clips according to the annotated transcripts. We also use semantic matching to ensure that the link between scenes (e.g., shared speaker, consistent topic, or causal relation) is grounded in the metadata. Incorrect or weakly-connected samples are regenerated with stricter constraints on shared entities or topics.

Final Sanity Checks. Across all tasks, we perform automated mismatch detection to identify: (i) incorrect speaker labels, (ii) misaligned timestamps, (iii) invalid temporal spans, (iv) hallucinated entities in template-generated questions, and (v) multi-modal inconsistencies between audio, visual tracks, and transcripts. Detected issues are rectified either by re-extracting metadata from the raw sources, regenerating templates with stricter constraints, or manual inspection for borderline cases. This ensures that every AMUSE sample is temporally accurate, speaker-consistent, and semantically grounded.

2.4. Question Templates

Here we add question template for each task in Tab. 19 - Tab. 24. Each task in AMUSE is paired with a diverse set of carefully designed question templates that capture the core reasoning abilities required for that task. For example, Audio-Visual Dialogue Summarization includes prompts that ask annotators or models to restate, condense, or paraphrase a speaker's message within a specified temporal segment. Similarly, the other five tasks Speaker Temporal Grounding, AV Speaker Association, Next Speaker Prediction, Speaker Re-identification, and Cross-Scene Narrative Linking each use their own bank of structured question variants tailored to highlight temporal localization, identity matching, conversational dynamics, or narrative inference. Across tasks, these templates ensure broad semantic coverage, reduce prompt bias, and provide consistent evaluation signals while reflecting the natural variability of real-world multimodal queries.

3. Post-Training Intuition

While prior multimodal alignment methods rely on policy optimization (e.g., GRPO) or lightweight adapter tuning (e.g., LoRA), `RAFT` restructures post-training around three mathematically grounded components:

3.1. Self-Reflective Rewarding

For a multimodal input $x = (x^{(a)}, x^{(v)}, x^{(t)})$ and model output $(\hat{y}, r)$ consisting of an answer and a reasoning trace, we define the intrinsic reward as

$$R(x, \hat{y}, r) = \lambda_{\text{task}}\, s_{\text{task}}(\hat{y}) + \lambda_{\text{align}}\, s_{\text{align}}(x, r) + \lambda_{\text{conf}}\, s_{\text{conf}}(\hat{y}). \quad (1)$$

Here, s_{task} measures task-level correctness (e.g., answer accuracy or span IoU), s_{align} captures cross-modal consistency between referenced speakers/timestamps and the underlying audio–visual evidence, and s_{conf} penalizes unsupported over-confident predictions. All components are derived from the model's internal probabilities and alignment scores no external reward model is used. Intuitively, the model is rewarded when its predictions and explanations agree with the multimodal input and penalized when they contradict themselves.

STG	Metric	Value	Final IoU↑
Whisper	WER ↓	1.33	
Pyannote	DER ↓	1.23	51.02
SyncNet	Sync Error ↓	2.12	

Table 3. Tool-level metrics and final STG performance for Qwen3-Omni.

3.2. Selective Reasoning Adaptation

We decompose model parameters into $\theta = (\theta_{\text{base}}, \theta_{\text{cross}})$, where θ_{cross} corresponds to explicitly interpretable cross-modal reasoning blocks. During training, we mask gradients such that only cross-modal parameters are updated:

$$\tilde{\nabla}_{\theta_i}\mathcal{L} = \begin{cases} \nabla_{\theta_i}\mathcal{L}, & \theta_i \in \theta_{\text{cross}}, \\ 0, & \theta_i \in \theta_{\text{base}}. \end{cases} \tag{2}$$

This focuses adaptation on the specific components responsible for audio–visual–text reasoning, improving compute and data efficiency while avoiding catastrophic forgetting.

3.3. Temporal Coherence Constraint.

Let $h_t^{(a)}$, $h_t^{(v)}$, and $h_t^{(t)}$ denote audio, visual, and text embeddings extracted at time index t. We impose a temporal coherence loss:

$$\mathcal{L}_{\text{temp}} = \sum_t \left(\|h_t^{(a)} - h_t^{(v)}\|_2^2 + \|h_t^{(v)} - h_t^{(t)}\|_2^2 \right) + \sum_t \gamma \left\| \left(h_{t+1}^{(a)} - h_t^{(a)}\right) - \left(h_{t+1}^{(v)} - h_t^{(v)}\right) \right\|_2^2. \tag{3}$$

The first two terms encourage cross-modal agreement at each timestep, while the final term enforces consistent evolution over time. Intuitively, this prevents identity jumps, speaker drift, or abrupt modality mismatches critical for multi-speaker tracking, next-speaker prediction, and narrative coherence.

Together, these components provide a principled post-training paradigm in which RAFT optimizes a self-reflective reward, updates only interpretable cross-modal reasoning pathways, and maintains temporally coherent multimodal embeddings yielding efficient, stable, and grounded multimodal alignment.

4. Additional Experiments

4.1. RAFT Ablations and Robustness

4.1.1. Effect of Temporal Regularization

To isolate the contribution of the temporal regularizer $\mathcal{L}_{\text{temp}}$, we ablate it from the RAFT objective and retrain the models. Tab. 9 shows results on Speaker Temporal Grounding (STG) task; similar trends hold for other tasks.

AVS	Metric	Value	Final BLEU↑
Whisper	WER ↓	1.15	
Pyannote	Turn Acc ↑	89.29	48.08
InsightFace	ID Consistency ↑	90.33	

Table 4. Tool-level metrics and final AVS performance for Qwen3-Omni.

AVSA	Metric	Value	Final Acc↑
Whisper	Utterance Match ↑	92.93	
Pyannote	Speaker-ID Match ↑	94.22	46.98
InsightFace	Face-ID Match ↑	90.37	
SyncNet	AV Sync ↑	94.10	

Table 5. Tool-level metrics and final AV Speaker Association performance for Qwen3-Omni.

NSP	Metric	Value	Final Acc↑
Whisper	Context Coverage ↑	91.03	
Pyannote	Turn Ordering ↑	92.89	45.02
InsightFace	Visual ID Consistency ↑	93.20	

Table 6. Tool-level metrics and final Next Speaker Prediction performance for Qwen3-Omni.

SRI	Metric	Value	Final Acc↑
Whisper	Utterance Match ↑	93.27	
Pyannote	Identity Stability ↑	91.04	58.65
InsightFace	Embedding Match ↑	94.88	

Table 7. Tool-level metrics and final SRI performance for Qwen3-Omni.

CSNL	Metric	Value	Final Acc↑
Whisper	Transcript Match ↑	89.37	
Pyannote	Speaker Attribution ↑	90.04	49.76
InsightFace	Identity Continuity ↑	94.11	

Table 8. Tool-level metrics and final CSNL performance for Qwen3-Omni.

4.1.2. Softmax Temperature in RRO

Fig. 3 illustrates the effect of the RRO temperature β on average AMUSE performance. Extremely low or high values of β reduce stability by either under-emphasizing or overly sharpening the reward distribution. In contrast,

Model	Full RAFT	w/o $\mathcal{L}_{\text{temp}}$	Δ
Qwen3-Omni	56.3	51.5	−4.8
Qwen2.5-Omni	54.6	48.2	−6.4
CREMA	41.0	37.5	−3.5

Table 9. Ablation of the temporal regularizer $\mathcal{L}_{\text{temp}}$ on STG. We report Temporal IoU (higher is better).

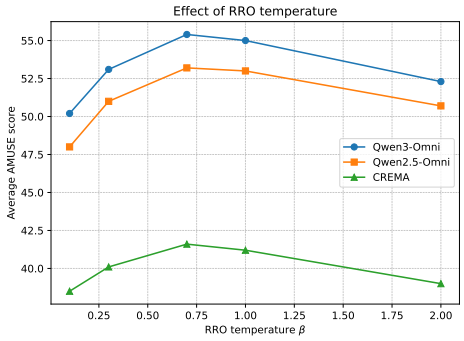

Figure 3. Effect of the RRO temperature β on average AMUSE performance. Extremely low or high values hurt stability, whereas a moderate range $\beta \in [0.3, 1.0]$ yields robust gains.

a moderate range ($\beta \in [0.3, 1.0]$) consistently yields higher scores across Qwen3-Omni, Qwen2.5-Omni, and CREMA, indicating that RRO benefits from controlled softmax weighting that strengthens perceptual correctness without amplifying noise.

The reflective reward, which aggregates multimodal consistency signals, is applied through reward-weighted regression and integrated into the full objective to encourage self-correction and reasoning consistency.

4.1.3. Parameter Efficiency of SRA

We compare RAFT with SRA to generic low-rank adaptation (LoRA) under different budgets of trainable parameters in Fig. 4. We measure the percentage of fine-tuned parameters relative to the backbone model.

Tab. 10 compares LoRA and SRA under different trainable–parameter budgets. SRA achieves comparable or higher average scores while using an order of magnitude fewer parameters. Notably, SRA-0.5% attains the best performance (54.1) despite training far fewer parameters than LoRA-5%.

4.2. Dataset-Centric Analyses

4.2.1. Speaker Overlap Difficulty

Fig. 5 shows how speaker overlap affects performance on AVSA and STG. As the proportion of time with two or more concurrent speakers increases, both tasks exhibit a clear performance drop, highlighting the difficulty of reasoning under dense multi-speaker interactions. Even with

Method	Trainable %	Avg. Score	Rel. Δ
LoRA-1%	1.0	51.2	–
LoRA-5%	5.0	53.5	+4.5
SRA-0.2%	0.2	52.4	+2.3
SRA-0.5%	0.5	**54.1**	+5.7

Table 10. Performance vs. trainable parameter budget. SRA matches or exceeds LoRA with an order of magnitude fewer parameters.

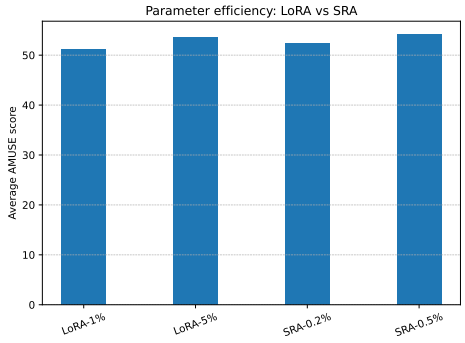

Figure 4. Average AMUSE score vs. fraction of trainable parameters for LoRA and SRA on Qwen3-Omni. RAFT with SRA achieves higher performance at significantly lower parameter budgets.

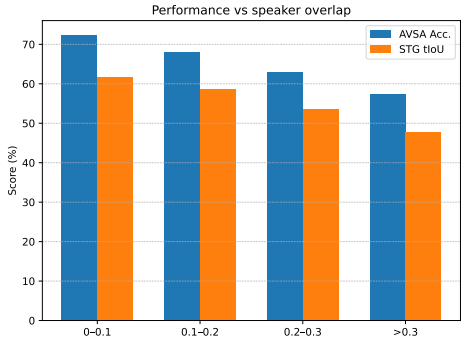

Figure 5. Performance as a function of speaker overlap ratio. Multi-speaker overlap substantially challenges even RAFT-trained models.

RAFT training, which improves grounding and temporal consistency, high-overlap scenarios remain challenging due to rapid turn-taking, overlapping utterances, and visual occlusions.

4.2.2. Number of Visible Speakers

Fig. 6 reports accuracy as a function of the number of visible speakers for AVSA and NSP. Both tasks show a consistent decline as scenes become more crowded, indicating the increased difficulty of tracking conversational roles and anticipating turn-taking when multiple

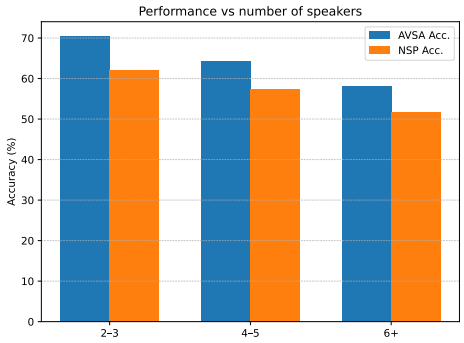

Figure 6. Accuracy vs. number of visible speakers on AVSA and NSP. Performance drops as the scene becomes more crowded.

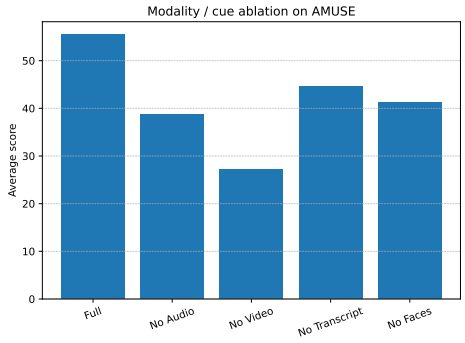

Figure 7. Effect of removing modalities and cues on average AMUSE performance. Multi-speaker reasoning is strongly multi-modal.

Duration	0–20s	20–40s	>40s
Avg. Score	56.8	54.2	49.7

Table 11. Effect of clip duration on average AMUSE performance.

participants are simultaneously visible. Even with RAFT, higher speaker density introduces more visual competition, overlapping cues, and ambiguous interaction patterns, which collectively reduce model accuracy.

4.2.3. Clip Duration vs. Accuracy

Tab.11 reports AMUSE performance as a function of clip duration. We observe a gradual decrease in accuracy as clips become longer, with the highest scores on short segments (0–20s) and the lowest on clips exceeding 40s. This trend reflects the increased reasoning difficulty in longer interactions, where models must track more speaker turns, maintain cross-modal coherence, and handle greater temporal dependencies.

4.3. Agentic Tool-Use and Cue Ablations

4.3.1. Tool Invocation Behavior

In agentic mode, the model decides when to call ASR (Whisper), speaker diarization (Pyannote), and face recognition (InsightFace). We report the performance of the tool call in Tab. 3 - Tab. 8 We report the fraction of examples in which the tool decisions match our oracle configuration in Tab. 12.

4.3.2. Modality and Cue Ablations

Fig. 7 and Tab. 13 present the effect of removing individual modalities and cues on average AMUSE performance. Removing audio or video causes the largest degradation, confirming that multi-speaker reasoning is fundamentally audio-visual and cannot be solved from transcripts. Eliminating transcripts or face crops also reduces accuracy, though to a lesser extent, indicating that all modalities contribute complementary cues. Together, these results highlight the strongly multi-modal nature of AMUSE and the importance of maintaining synchronized audio, visual, and textual information for robust reasoning.

4.4. Comparison to Other Alignment Objectives

Tab. 14 compares RAFT with standard RLHF approaches (PPO, DPO, and GRPO) across all six AMUSE tasks on Qwen3-Omni. RAFT consistently achieves the highest performance in every task, with especially strong gains in AVDS, NSP, SRID, and CSNL. These improvements highlight RAFT 's ability to provide more stable reward weighting, better multimodal grounding, and more coherent step-by-step reasoning than existing preference-optimization methods.

4.5. Semantic Metrics for AVDS

We report semantic-level metrics (BERTScore and MAUVE) for AVDS in Table 15, which capture semantic consistency beyond surface-level overlap. As shown, RAFT consistently improves both metrics, indicating better alignment with the underlying meaning of the ground-truth summaries rather than relying on lexical similarity alone. In particular, Qwen3-Omni achieves strong gains, demonstrating that RAFT enhances abstractive quality and semantic fidelity in generated summaries.

4.6. Modality Ablation on AMUSE

We report (Table 16) modality ablations on Qwen3-Omni. Single-modality inputs (text-only, audio-only, or visual-only) consistently underperform full audio-visual inference, indicating that unimodal signals are insufficient for AMUSE. In particular, transcript-level cues alone fail to capture speaker dynamics, temporal alignment, and cross-modal grounding required for accurate reasoning. In contrast, combining audio and visual inputs yields substantial gains, highlighting the importance of joint

Task	ASR Decision	Diarization Decision	Face-Track Decision
AVDS	92.1	88.4	75.6
AVSA	89.3	94.7	91.2
NSP	81.5	78.6	69.4
STG	87.9	96.1	93.2

Table 12. Tool selection correctness in RAFT agentic mode.

Setting	Avg. Score	Δ vs. Full
Full (A+V+T+F)	55.4	–
No Audio	42.7	−12.7
No Video	39.1	−16.3
No Transcript	48.5	−6.9
No Face Crops	50.2	−5.2

Table 13. Cue ablation on AMUSE (average score across tasks).

multimodal integration for robust multi-speaker understanding.

5. Additional Qualitative Results

Figures 8 and 9 illustrate qualitative comparisons across all six AMUSE tasks under Zero-Shot, Agentic w/o RAFT, and Agentic w/ RAFT modes. We observe consistent improvements in multimodal grounding, speaker attribution, and temporal consistency when using RAFT.

Fig. 8 (AVDS, AVSA, NSP). Zero-shot models frequently rely on textual priors and ignore speaker cues, leading to incorrect summaries, mismatched utterance–speaker assignments, and poor turn-taking prediction. Agentic inference without RAFT improves tool usage but remains unstable. In contrast, RAFT enables accurate identification of the correct speaker in AVDS, reliable association of utterances in AVSA, and context-aware prediction of the next speaker in NSP by enforcing structured reasoning and perceptual alignment.

Fig. 9 (SRID, STG, CSNL). For identity reasoning (SRID), non-RAFT agents confuse visually similar individuals, while RAFT reliably matches speakers across scenes. In STG, RAFT reduces temporal drift and accurately localizes when a specific person starts or stops speaking under heavy overlap. In CSNL, RAFT correctly links causally dependent events across disjoint scenes, avoiding shallow pattern matching. Overall, RAFT yields coherent, grounded, and stable multi-speaker reasoning, complementing the quantitative gains reported in the main paper.

6. User Study

Sample Curation Validity. Each dataset sample is manually reviewed by human annotators to ensure accuracy and clarity. Raters watch the full clip, verify transcripts, speaker identities, and temporal spans, and check that the question unambiguously matches the underlying audio–visual evidence. Samples with unclear boundaries, mismatched associations, or ambiguous narratives are corrected or discarded, ensuring that all items used for evaluation are high-quality and reliable.

Human Performance Estimation. To establish an upper bound on task difficulty, human raters also answer the evaluation questions themselves under the same conditions as the model. Annotators select answers for multiple-choice tasks or mark temporal segments for grounding tasks, providing a ceiling for achievable performance and helping distinguish true model errors from inherently ambiguous cases.

7. RAFT Algorithm

Algo. 1 summarizes the RAFT training procedure. The model first aligns its step-by-step reasoning to human supervision through the structured reasoning loss. It then samples multiple candidate responses and scores them with a perceptual reward to perform RRO, producing stable, reward-weighted updates. A temporal grounding regularizer enforces cross-modal synchrony across audio, visual, and textual streams. Finally, only the SRA adapter parameters are updated using the combined RAFT objective, enabling efficient and well-grounded multimodal reasoning.

8. Evaluations Details

8.1. Prompt Templates Across Evaluation Modes

We provide complete prompt templates for all three evaluation modes—zero-shot, guided, and agentic—including examples of how perception tools (ASR, diarization, face recognition, and AV sync) are incorporated. Zero-shot templates appear in the main paper; below we describe the guided and agentic settings. In the guided mode, all external tools (Whisper, PyAnnote, InsightFace, SyncNet) are executed offline and their outputs are inserted into the prompt as structured metadata, which the model must rely on without invoking tools itself. In the agentic mode, the model is instead given access to the full toolset and must autonomously decide when and how to

Method	AVDS (B@4) ↑	AVSA (Acc%) ↑	NSP (Acc%) ↑	SRID (Acc%) ↑	STG (Acc%) ↑	CSNL (Acc%) ↑
PPO	34.82	61.13	53.28	58.02	49.44	39.26
DPO	35.47	60.73	54.57	59.10	50.11	40.05
GRPO	36.77	62.23	55.49	60.34	51.84	41.35
RAFT (ours)	**54.54**	**54.22**	**56.73**	**62.53**	**56.33**	**57.26**

Table 14. Comparison of RAFT with PPO, DPO, and GRPO across AMUSE tasks on Qwen3-Omni. We report task-specific metrics. B@4: BLEU score.

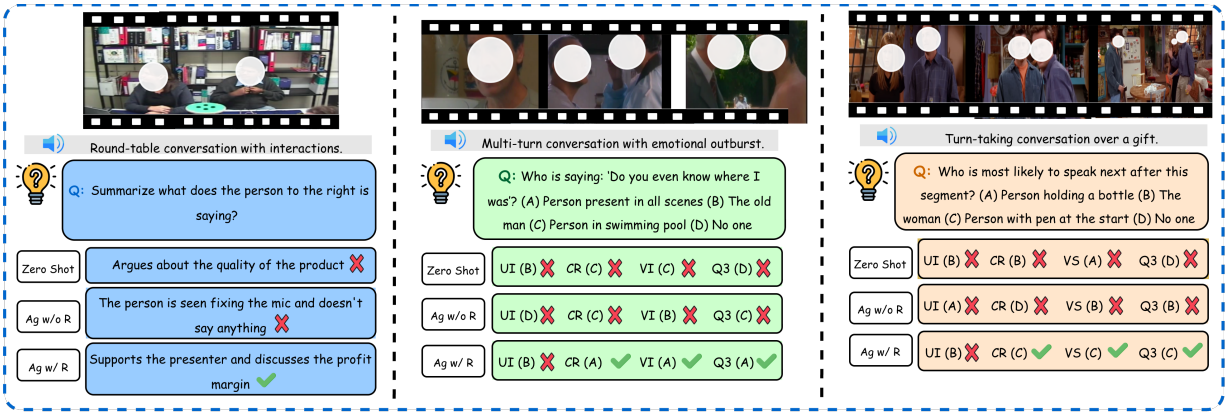

Figure 8. **Qualitative results 1.** Comparison on multi-speaker reasoning tasks: Audio-Visual Dialogue Summarization (left), Speaker Association (middle), and Next Speaker Prediction (right). UI: *Unified-IO2*, CR: *CREMA*, VS: *VideoSALMONN*, VI: *VITA*, Q2.5: *Qwen2.5-Omni*, and Q3: *Qwen3-Omni* under Zero-Shot, Agentic w/o RAFT, and Agentic w/ RAFT modes. Results for AVDS is for Qwen3-Omni.

Setting	BERT	MAUVE
Zero-shot	0.65	0.45
Guided	0.68	0.61
Agentic (no RRO)	0.69	0.59
RAFT-Full	**0.81**	**0.78**

Table 15. AVDS performance across settings.

Input	AVDS (GPT) ↑	AVSA (Acc) ↑
Text-only	3.9	26.8
Audio-only	4.4	21.9
Visual-only	4.1	30.1
Audio + Visual	**6.6**	**54.2**

Table 16. Modality ablations on Qwen3-Omni.

call tools, integrate their outputs, and perform multi-step reasoning to solve each task.

Guided Mode Prompt. In the guided mode, the model operates purely as a reasoning layer over precomputed structured information (Tab. 17). The template below is instantiated separately for each of the six AMUSE tasks.

Agentic Mode Prompt. In the autonomous mode, the model acts as a multimodal agent. It must plan, call tools selectively, incorporate returned evidence, and synthesize a final answer. The tool calling details are reported in Tab. 2 The same template is used across all AMUSE tasks. The prompt used is explained in Tab. 18.

8.2. More Details on LLM-based Choice Extraction

Choice extraction strategy. We adopt a two-stage procedure to robustly extract discrete choices from free-form AVLLM predictions. Although humans can easily infer the intended choice, rule-based matching is often brittle when faced with stylistic variation or incomplete responses. To ensure consistency across AVLLMs with diverse instruction-following abilities, we standardize the evaluation pipeline as follows:

Step 1. Prediction matching: We first apply a lightweight heuristic matching strategy to directly detect the choice label (e.g., 'A', 'B', 'C', 'D') from the model's output. If a valid label is found, it is used as the final prediction. If no reliable match is extracted, we proceed to the LLM-based extraction step.

Step 2. GPT-4 processing: Following prior benchmarks such as [4], GPT-4 serves as a dependable choice extractor. When Step 1 fails, we provide GPT-4 with the question, the list of answer choices, and the model's free-

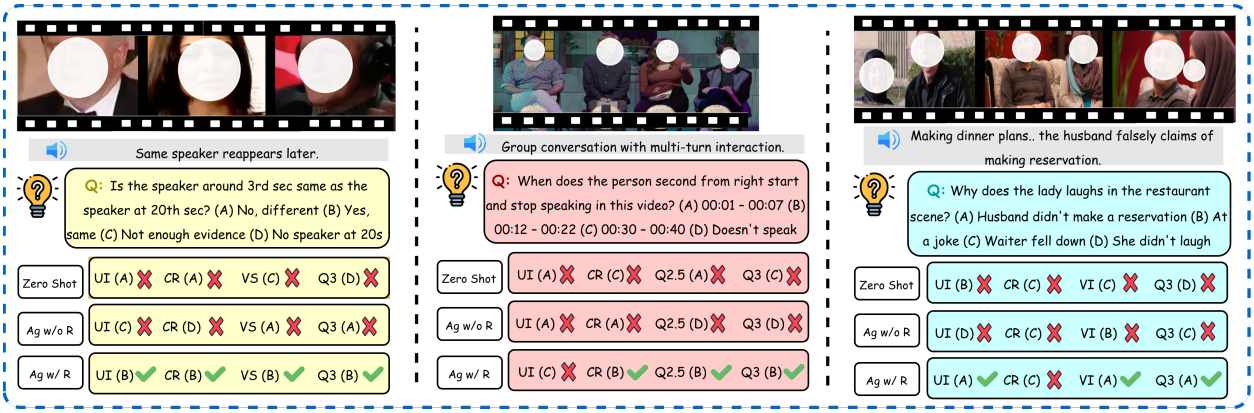

Figure 9. **Qualitative results 2.** Comparison on multi-speaker reasoning tasks: Speaker Re-identification (left), Temporal Grounding (middle), and Cross Scene Narrative Linking (right). UI: *Unified-IO2*, CR: *CREMA*, VS: *VideoSALMONN*, VI: *VITA*, Q2.5: *Qwen2.5-Omni*, and Q3: *Qwen3-Omni* under Zero-Shot, Agentic w/o `RAFT`, and Agentic w/ `RAFT` modes.

Algorithm 1 `RAFT`: Reasoning, Acting, and Feedback Training

Require: Dataset $\mathcal{D}$ of (x, y), policy π_θ, perceptual reward $R(x, y)$, weights α, β, samples K
Ensure: Updated parameters θ (SRA adapters)

Initialize θ of the base MLLM
Insert Selective Reasoning Adaptation (SRA) adapters in multimodal reasoning layers
while not converged **do**
  Sample minibatch $\{(x, y)\} \subset \mathcal{D}$
  // 1. Structured reasoning alignment
  Generate reasoning steps $y_k \sim \pi_\theta(\cdot \mid x)$
  $\mathcal{L}_{\text{align}} = -\sum_k \log \pi_\theta(y_k \mid x)$
  // 2. Reflective Reward Optimization (RRO)
  for each x in the minibatch **do**
    Sample $\{y_i\}_{i=1}^{K} \sim \pi_\theta(\cdot \mid x)$
    Compute rewards $r_i = R(x, y_i)$
  end for
  $\bar{r} = \frac{1}{K} \sum_i r_i$
  $w_i = \dfrac{\exp(\beta(r_i - \bar{r}))}{\sum_j \exp(\beta(r_j - \bar{r}))}$
  $\mathcal{J}_{\text{RRO}} = \sum_i w_i \log \pi_\theta(y_i \mid x)$
  // 3. Temporal grounding regularizer
  Extract embeddings $f_a(t), f_v(t), f_s(t), f_r(t)$
  $\mathcal{L}_{\text{temp}} = \sum_t \left(\|f_a(t) - f_v(t)\|_2^2 + \gamma \|f_s(t) - f_r(t)\|_2^2 \right)$
  // 4. `RAFT` objective and SRA update
  $\mathcal{L}_{\text{RAFT}} = \mathcal{L}_{\text{align}} + \alpha \mathcal{L}_{\text{temp}} - \beta \mathcal{J}_{\text{RRO}}$
  Update θ (SRA parameters only) by a gradient step on $\mathcal{L}_{\text{RAFT}}$
end while
return θ =0

form response, and instruct it to align the response with the most semantically similar option. If no option aligns, GPT-4 outputs "No match found". We additionally employ the CircularEval protocol [4] to ensure rigorous evaluation and to highlight performance differences among AVLLMs.

Response matching. We treat an option as selected whenever it is referenced through its isolated label (e.g., 'A') or standard labeled formats such as 'A) `<response>`', 'A. `<response>`', 'A, `<response>`', or '(A) `<response>`'—provided the `<response>` segment does not contain other option labels.

Where does heuristic matching fail? Heuristic matching commonly fails in two situations: (i) when the AVLLM does not commit to an answer and instead asks for clarification (e.g., "Apologies, could you clarify...?"), and (ii) when the model outputs multiple option labels simultaneously. In such cases, we defer to GPT-4 for choice extraction, as shown below.

Guided-mode prompt template

You are given a video clip along with structured information extracted using external audio-visual tools. All processing has already been completed. Use only the information shown below to solve the task.

Video metadata: short description, dataset source, and duration.
ASR transcript: transcript text with timestamps.
Speaker diarization: list of speech segments with start time, end time, and speaker identifiers.
Face tracks: list of track identifiers, their visible time spans, and bounding-box intervals.
Audio-visual alignment: optional synchronization offsets or scores.

Task description: one of the six AMUSE tasks (speaker temporal grounding, audio-visual dialogue summarization, speaker association, next speaker prediction, speaker re-identification, or cross-scene narrative linking).

Instructions:
(1) Read the task description carefully and determine what must be predicted.
(2) Use transcript segments, diarization labels, face-track identifiers, and alignment information as explicit evidence.
(3) Do not infer speakers, timestamps, or events that are not present in the structured fields.
(4) When reasoning about speakers or faces, always refer back to the provided identifiers.
(5) Produce a concise final answer and a short justification that cites the relevant segments or identifiers.

Table 17. Guided-mode prompt template. All external tools are executed before prompting, and their outputs are injected as structured text.

Autonomous (agentic) prompt template

You are an audio-visual reasoning agent. You can decide when and how to use the following tools in order to solve the task:

Whisper: transcribes the audio into text.
PyAnnote: produces speaker diarization with time-stamped segments.
InsightFace: provides face tracks and identity features for visible people.
SyncNet: estimates audio-visual synchronization between speech and faces.

Task description: one of the six AMUSE tasks (speaker temporal grounding, audio-visual dialogue summarization, speaker association, next speaker prediction, speaker re-identification, or cross-scene narrative linking).

Instructions:
(1) First restate the task in your own words and outline what information you need.
(2) Decide which tools are necessary to obtain that information and why; avoid unnecessary tool calls.
(3) Invoke tools sequentially, update your plan after each result, and decide whether additional calls are required.
(4) Treat tool outputs as ground-truth metadata and base your reasoning strictly on these results.
(5) Do not hallucinate speakers, timestamps, or events that are not supported by tool outputs or the video description.
(6) Once you have gathered sufficient evidence, produce a final answer along with a brief justification that explicitly cites the tool results and time segments you used.

Your response should therefore contain: (a) a short plan, (b) references to the tools you chose to use and their returned outputs, and (c) a final answer with clear, evidence-based reasoning.

Table 18. Autonomous-mode (agentic) prompt template. The model independently determines which tools to invoke and integrates their outputs.

Prompt Variants for Audio-Visual Dialogue Summarization

1. Summarize what the person in the <descriptor> says between <time_token> and <time_token>.
2. What is the main idea expressed by the speaker wearing a red shirt at <time_token>? Please summarize
3. Briefly summarize the key point the woman on the left conveys during <time_token>.
4. What is the speaker in the blue jacket trying to communicate at <time_token>?
5. Summarize the statement made by the man in the center into one sentence.
6. What message is the person standing on the right conveying at <time_token>?
7. Describe what the highlighted individual talks about during the segment starting at <time_token>.
8. Provide a brief summary of the response given by the seated person at <time_token>.
9. Rephrase the speaker's comment between <time_token> and <time_token> concisely.
10. What conclusion does the person in the black hoodie present during this segment?
11. In a few words, describe what the speaker with glasses emphasizes at <time_token>.
12. What information does the person in <outfit descriptor> share at <time_token>?
13. What does the dialogue turn from the woman on the right mainly focus on at <time_token>?
14. Summarize the viewpoint expressed by the man in the gray shirt in this part.
15. What does the person near the doorway explain during the segment starting at <time_token>?
16. Which topic does the speaker in the red dress address between <time_token> and <time_token>?
17. Summarize the line spoken by the person on the left couch at <time_token>.
18. What is the essence of the statement made by the speaker standing at the table?
19. Give a short paraphrase of what the person in the blue sweater says at <time_token>.
20. Summarize the main point communicated by the speaker facing the camera.

Table 19. Prompt variants for the **Audio-Visual Dialogue Summarization** task.

Prompt Variants for Audio-Visual Speaker Association

1. Who is speaking during the audio segment at <time_token>?
2. Match the utterance at <time_token> to the correct person in the scene.
3. Which individual is producing the speech at <time_token>?
4. Identify who is talking using lip movement and voice cues at <time_token>.
5. Who is talking while others remain silent at <time_token>?
6. Which person corresponds to the audio clip starting at <time_token>?
7. Who is the active speaker when the man in the red shirt moves his lips at <time_token>?
8. Whose voice do we hear when the woman seated on the left is shown at <time_token>?
9. Based on audio-visual cues, who is speaking while the man in the blue jacket appears at <time_token>?
10. Which on-screen person is talking during the segment at <time_token>?
11. Whose lip motion aligns with the spoken sentence at <time_token>?
12. Identify the speaker when the right-side participant is visible at <time_token>.
13. Which speaker's voice corresponds to the utterance at <time_token>?
14. Which person produces the spoken line heard at <time_token>?
15. Who is responsible for the highlighted phrase at <time_token>?
16. Who is delivering the dialogue while the person in the black jacket is centered at <time_token>?
17. Which person should be attributed as the speaker of the sentence aligned with the lip motion at <time_token>?
18. Whose mouth movement matches the audio when the left side of the table is shown at <time_token>?
19. Who is the speaker when the person in the white shirt appears at <time_token>?
20. Who produces the spoken line associated with the audio segment at <time_token>?

Table 20. Prompt variants for the **Audio-Visual Speaker Association** task.

Prompt Variants for Next Speaker Prediction

1. Based on the interaction up to <time_token>, who is most likely to speak next?
2. Who seems prepared to reply following the segment ending at <time_token>?
3. Which individual is most likely to take the next turn in the conversation?
4. Predict the next speaker among the on-screen participants.
5. Who appears ready to answer the question asked at <time_token>?
6. Which person is positioned to speak next, given their posture and gaze?
7. Whose body language suggests they are about to answer after <time_token>?
8. Considering the conversation flow, who is expected to continue the dialogue?
9. Who on the left side of the frame seems ready to speak next?
10. Which person on the right side of the table will likely speak after the current turn?
11. Who follows up the conversation after the speaker in the red shirt finishes at <time_token>?
12. Using gaze direction and facial expressions at <time_token>, who is likely to speak next?
13. Which person will likely contribute the next line following <time_token>?
14. Which participant seated on the couch is expected to speak next?
15. From the pattern of turn-taking up to <time_token>, who takes the next turn?
16. Who seems about to interject when the camera shows the group at <time_token>?
17. Which character is cueing up the next utterance, for example by leaning forward or opening their mouth?
18. Whose gestures indicate that they are preparing to speak next?
19. Who resumes the conversation after the short pause at <time_token>?
20. Who logically continues the dialogue when the question is directed towards the person in the blue sweater at <time_token>?

Table 21. Prompt variants for the **Next Speaker Prediction** task.

Prompt Variants for Speaker Temporal Grounding

1. At what <time_token> does the person in <descriptor> begin speaking?
2. When does the woman in the red dress start talking in the video?
3. Identify the <time_token> at which the man on the left first begins to speak.
4. Locate the moment the person in the blue shirt starts speaking.
5. At which <time_token> does their speech initiation occur?
6. Find the starting time of the speaker's voice for the person near the doorway.
7. When is the first audible word from the person sitting on the right side of the table?
8. Mark the <time_token> at which the person in the black hoodie begins their utterance.
9. What is the earliest <time_token> at which this person starts speaking?
10. Between which <time_token> values does the speaker's utterance begin?
11. Give the <time_token> where the woman in the center first speaks.
12. When does the speech associated with the man in the gray sweater start?
13. Identify the first frame in time (as <time_token>) when the speaker on the left starts talking.
14. At what <time_token> does the dialogue contribution of the person in the white shirt begin?
15. At what time does this speaker enter the conversation for the first time?
16. When is their first vocalization heard after they appear on screen?
17. What exact <time_token> corresponds to the onset of the speaker's voice?
18. Find the <time_token> where this speaker's sentence begins in the timeline.
19. At which <time_token> does the person standing at the counter start speaking?
20. Determine the onset <time_token> of the utterance produced by the speaker in <descriptor>.

Table 22. Prompt variants for the **Speaker Temporal Grounding** task.

Prompt Variants for Speaker Re-identification

1. Is the speaker at <time_token> the same person as the speaker at <time_token>?
2. Does the voice in the segment at <time_token> match the voice at <time_token>?
3. Are the speech segments at the two <time_token> values produced by the same individual?
4. Is the person wearing a red shirt at <time_token> the same as the speaker at <time_token>?
5. Compare the speaker at <time_token> with the speaker at <time_token>: are they the same person?
6. Do the face and voice at <time_token> correspond to the same identity as at <time_token>?
7. Are the appearances of the man on the left at the two <time_token> positions from the same person?
8. Is the person in the blue jacket at <time_token> the same speaker who talks at <time_token>?
9. Do the vocal patterns and facial cues at <time_token> and <time_token> indicate a single speaker identity?
10. Is the speaker near the doorway at <time_token> the same as the speaker at <time_token>?
11. Does the speaker at <time_token> match the person in the striped shirt speaking at <time_token>?
12. Are the speakers across the segments at <time_token> and <time_token> the same individual?
13. Is the woman on the right speaking at <time_token> the same woman speaking at <time_token>?
14. Are the vocal characteristics of the person in the black hoodie at <time_token> consistent with those at <time_token>?
15. Does the person speaking at the table at <time_token> correspond to the same identity speaking on the couch at <time_token>?
16. Is the speaker shown in close-up at <time_token> the same as the one talking in the wide shot at <time_token>?
17. Does the voice of the person in the red dress at <time_token> match the voice at <time_token>?
18. Do the face and audio cues of the man on the left at <time_token> indicate the same speaker identity as at <time_token>?
19. Does the person shown near the window at <time_token> match the speaker filmed near the table at <time_token>?
20. Are the dialogue segments at <time_token> and <time_token> delivered by the same speaker?

Table 23. Prompt variants for the **Speaker Re-identification** task.

Prompt Variants for Cross-scene Narrative Linking

1. How does the event at <time_token> connect to the reaction of the person in the red sweater at <time_token>?
2. What detail shown at <time_token> explains why the woman on the left reacts at <time_token>?
3. Why does the man in the blue shirt react the way he does at <time_token>, given what happened at <time_token>?
4. Which event at <time_token> provides context for the final scene at <time_token>?
5. What narrative link exists between the segment at <time_token> and the segment at <time_token>?
6. How does the phone call or object mention at <time_token> relate to the reaction of the person standing on the right at <time_token>?
7. What realization does the woman in the black jacket have at <time_token> based on something shown at <time_token>?
8. Explain what triggers the behavior of the person sitting on the couch at <time_token>, using clues from <time_token>.
9. Which event witnessed by the man on the far left at <time_token> leads to his action at <time_token>?
10. How does the interaction at <time_token> influence the character's response at <time_token>?
11. What causal link connects the scene at <time_token> with the outcome at <time_token>?
12. Which visual clue shown at <time_token> helps explain the reaction of the woman in the red dress at <time_token>?
13. How does the introduction of the <object> at <time_token> shape the character's interpretation at <time_token>?
14. Which detail noticed by the man on the right side of the frame at <time_token> sets up his reaction at <time_token>?
15. How does the sequence at <time_token> prepare the narrative moment occurring at <time_token>?
16. What information revealed at <time_token> does the woman in the blue sweater realize at <time_token>?
17. What continuity links the segment occurring at <time_token> with the one at <time_token>?
18. How do the events at <time_token> and <time_token> form a complete narrative arc?
19. What observation made by the person in the striped shirt at <time_token> is recalled at <time_token>?
20. What chain of events starting from the scene at <time_token> leads to the reaction of the person near the doorway at <time_token>?

Table 24. Prompt variants for the **Cross-scene Narrative Linking** task.

Choice extraction prompt for GPT-4

Can you help me match an answer with a set of options for a single-correct-answer question? I will provide a question, a set of options, and a model-generated response. Your task is to map the response to the most similar option. Output exactly one uppercase letter from {A, B, C, D, E}. If no option matches, respond with "No match found". Please avoid subjectivity and do not use external knowledge.

Example 1:

Question: What color is the man's shirt who is sitting left of the object making this sound?

Options: A. Green B. Red C. Yellow D. Black

Answer: The person sitting next to the record player is wearing a black shirt.

Your output: D

Example 2:

Question: What does the audio-visual event constitute?

Options: A. A dog barking at a cat B. A dog barking on being hit by a stick C. The dog is hungry D. The dog is chasing another dog

Answer: It is a wolf.

Your output: No match found

References

[1] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. "Voxceleb2: Deep speaker recognition". In: *arXiv preprint arXiv:1806.05622* (2018).

[2] Chunhui Gu et al. "Ava: A video dataset of spatio-temporally localized atomic visual actions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 6047–6056.

[3] Wessel Kraaij et al. "The AMI meeting corpus". In: *Proc. International Conference on Methods and Techniques in Behavioral Research*. 2005, pp. 1–4.

[4] Yuan Liu et al. "Mmbench: Is your multi-modal model an all-around player?" In: *arXiv preprint arXiv:2307.06281* (2023).

[5] Yueqian Wang et al. "Friends-mmc: A dataset for multi-modal multi-party conversation understanding". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 39. 24. 2025, pp. 25425–25433.