SPEECHLENS: MULTI-PERSPECTIVE SPEECH CLUSTERING WITH REASONING AUDIO-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Speech clustering is a fundamental task for organizing rapidly growing speech collections, supporting applications such as conversational analysis and speech-driven discovery. However, existing methods rely on fixed acoustic similarity metrics or ASR-based text pipelines, limiting their ability to reorganize the same audio collection under different user-specified perspectives, especially when clustering depends on both linguistic and paralinguistic cues. We introduce **speech multi-perspective clustering**, where a model directly partitions speech recordings according to a natural-language perspective while inferring both the number of clusters and their assignments. To study this setting, we construct SpeechLens-Bench, a benchmark spanning multiple application domains and evaluating both in-perspective and cross-perspective generalization. We further propose SpeechLens-R1, an end-to-end large audio-language model trained with reasoning distillation and preference optimization. Experiments show that SpeechLens-R1 consistently outperforms all baselines, improving overall ARI by **12.99 points** and V-measure by **11.62 points**. These results demonstrate the promise of native audio-language models for flexible, perspective-conditioned structure discovery over speech collections.

1 INTRODUCTION

Speech clustering aims to organize collections of speech recordings into coherent groups, and is a fundamental component for speech-driven analysis, retrieval, and discovery (Park et al., 2022; Casanueva et al., 2020; Larson & Jones, 2012; Hu et al., 2023; Clifton et al., 2020). As speech data rapidly grows in conversational agents, meetings, podcasts, and domain-specific audio archives, users increasingly need to reorganize the same collection according to different analytical goals. For example, the same set of recordings may be grouped by communicative intent, affective state, speaker-related traits, or background context. This requires clustering systems to flexibly adapt to user-specified criteria and to reason over both linguistic content and paralinguistic cues.

Existing approaches are limited in this setting. Traditional acoustic clustering methods operate directly on speech representations and are effective for predefined criteria such as speaker identity or acoustic similarity, but they typically rely on fixed similarity metrics and task-specific representations. Transcript-centric pipelines, which are the focus of the comparison in Fig. 1(a), provide a more semantic alternative. They transcribe each recording with an automatic speech recognition (ASR) model, encode the transcript with a sentence encoder (Reimers & Gurevych, 2019), and apply a classical clustering algorithm such as K-Means (Lloyd, 1982) or Gaussian mixture models (GMMs) (Dempster et al., 1977). A more recent line of work (Fig. 1(b)) replaces the clustering stage with a large language model (LLM) that performs clustering directly over the transcripts (Zhang et al., 2023; Viswanathan et al., 2024). Yet transcript-centric pipelines share a fundamental bottleneck: **ASR captures *what* is said but discards much of *how* it is delivered**, including prosody, speaker traits, emotion, etc. As a result, existing methods either specialize in fixed acoustic notions of similarity or reason over text-only representations, but lack a unified mechanism for reorganizing the same speech collection under flexible perspectives that depend on both linguistic and paralinguistic information.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Recent large audio-language models (LALMs) provide a promising foundation for this problem because they can process speech directly and jointly model semantic and acoustic information (Diao et al., 2025). Nevertheless, current LALMs are primarily evaluated on audio understanding and generation, rather than on structure discovery over speech collections. It remains unclear whether such models can interpret a natural-language clustering perspective, compare multiple speech segments under that perspective, infer the appropriate number of clusters, and produce a valid partition without relying on a separate clustering algorithm.

In this work, we introduce **speech multi-perspective clustering**, a new task that formulates speech clustering as perspective-conditioned structure discovery. Given a set of speech recordings and a natural-language clustering perspective, the model must infer both the number of clusters and the assignment of each recording. Unlike conventional settings where the similarity function or the number of clusters is fixed in advance, our setting allows the same speech collection to induce different valid partitions under different perspectives. This formulation directly tests whether models can use natural-language criteria to select the relevant linguistic and paralinguistic evidence for clustering. To support this task, we construct SpeechLens-Bench, **the first benchmark for multi-perspective speech clustering across diverse application domains.** It includes perspectives grounded in lexical-semantic reasoning as well as paralinguistic attributes, enabling evaluation of both text-like reasoning and audio-native perception. We further organize the benchmark into held-in and held-out perspectives, allowing us to measure in-perspective generalization under familiar criteria and cross-perspective generalization to unseen criteria.

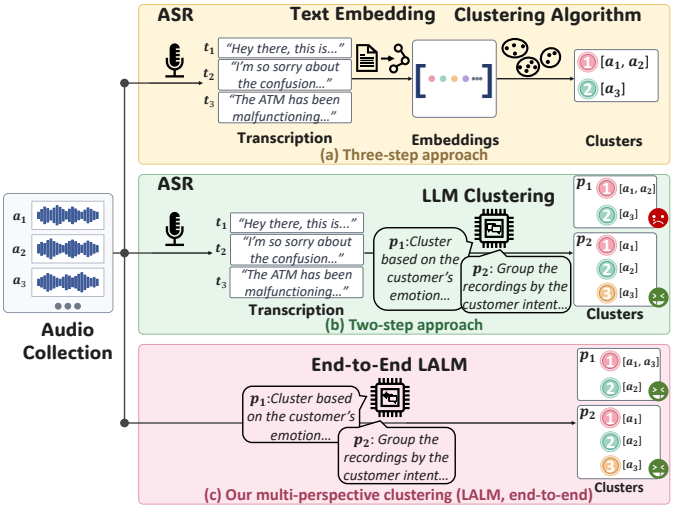


Figure 1: Three speech clustering paradigms. (a) Traditional pipelines transcribe speech, embed transcripts, and apply a clustering algorithm. (b) Transcript-based LLM clustering follows user-specified perspectives, but remains limited when the perspective depends on paralinguistic cues. (c) SpeechLens-R1 takes raw speech and natural-language perspectives as input, enabling the same speech collection to be clustered into different valid partitions under different perspectives.

We also propose **SpeechLens-R1, an end-to-end LALM-based clustering model** depicted in Fig. 1(c). SpeechLens-R1 takes raw speech segments and a natural-language perspective as input, and directly generates a structured clustering answer. To adapt LALMs to this task, we first train the model with reasoning distillation, which provides comparison-based supervision for perspective-conditioned clustering. We then apply direct preference optimization using valid but incorrect model-generated partitions as hard negatives, encouraging the model to prefer clustering decisions that better match the gold partition while preserving output validity.

Our contributions are summarized as follows:

- We formalize **speech multi-perspective clustering**, where recordings are clustered according to a natural-language perspective and the model must infer both cluster number and assignments.
- We introduce SpeechLens-Bench, a benchmark covering diverse domains, linguistic and paralinguistic perspectives, and held-in/held-out perspective splits.
- We propose SpeechLens-R1, an end-to-end LALM trained with reasoning distillation and preference optimization for perspective-conditioned speech clustering.
- Experiments demonstrate that SpeechLens-R1 consistently outperforms the baselines, improving overall ARI by **12.99** points and V-measure by **11.62** points.

2 BENCHMARK CONSTRUCTION

We introduce SpeechLens-Bench, a benchmark for evaluating whether models can cluster speech recordings according to natural-language perspectives. The benchmark is designed around two requirements. First, the clustering criterion should be flexible: the same speech collection may induce different valid partitions under different perspectives. Second, the benchmark should require both linguistic and paralinguistic reasoning, since many realistic speech-organization scenarios depend not only on ***what is said*** but also on ***how it is spoken***.

2.1 SOURCE CORPORA

We construct SpeechLens-Bench from four complementary corpora: ECHR (Poudyal et al., 2020), S&P 500 annual reports (jlohding, 2023), Banking77 (Casanueva et al., 2020), and MultiWOZ (Zang et al., 2020). These corpora cover legal cases, financial disclosures, banking service requests, and task-oriented dialogues, providing diverse domains and reasoning patterns. The original corpora are text-based or dialogue-based, so we convert them into speech recordings while preserving the clustering evidence required by each perspective. Corpus-specific construction details are provided in App. A.

2.2 PERSPECTIVE AND AUDIO SYNTHESIS

Fig. 2 illustrates the construction pipeline. For each corpus, we first induce candidate clustering perspectives from representative examples using an LLM. Each perspective defines a clustering criterion and a set of mutually exclusive categories. We then refine the proposed perspectives to ensure that categories are interpretable, comparable in granularity, and sufficiently supported by examples. After refinement, each retained perspective is converted into a natural-language clustering instruction that does not reveal category names.

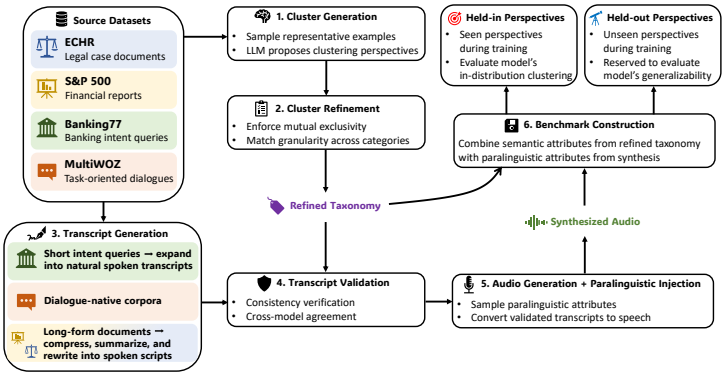


Figure 2: Overview of SpeechLens-Bench construction.

We next construct speech instances aligned with these perspectives. Depending on the source corpus, we either expand short user queries into spoken scripts, compress long documents into evidence-preserving spoken summaries, or reuse dialogue-native transcripts. To reduce lexical shortcuts, generated transcripts are not allowed to explicitly mention perspective names, category names, or near-verbatim instruction phrases. We validate each transcript by re-classifying it under the corresponding perspective and retaining only instances whose labels are consistently recovered.

Finally, validated transcripts are synthesized into speech using a text-to-speech (TTS) system. During synthesis, we inject controlled paralinguistic attributes such as emotion, speaker identity, speaker count, and background acoustic condition. For linguistic perspectives, these attributes are randomized and approximately balanced across categories to avoid spurious correlations. For paralinguistic perspectives, the target acoustic attribute defines the clustering criterion, while other attributes are randomized. This design allows SpeechLens-Bench to evaluate both lexical-semantic clustering and audio-native clustering. More details are discussed in App. A.7.

To minimize potential bias during construction, three authors independently reviewed the samples in SpeechLens-Bench. Specifically, they assessed the relevance of the clustering perspective, transcript accuracy, audio intelligibility, semantic-label correctness, and perceptibility of the injected paralinguistic attributes. A sample was retained only when the authors agreed that it satisfied these criteria; otherwise, it was discarded and regenerated.

2.3 BENCHMARK ORGANIZATION

We split clustering perspectives into *held-in* and *held-out* perspectives. Held-in perspectives are observed during training, including their instructions and underlying taxonomies. Held-out perspectives are reserved exclusively for evaluation, and their instructions, categories, and instances are never used during training. This perspective-level split allows us to distinguish in-perspective generalization from cross-perspective generalization.

Table 1: Statistics of SpeechLens-Bench. #Speech denotes the number of unique speech recordings, and #Data denotes the number of clustering instances.

Dataset	Train		$\mathbf{L}_0$		$\mathbf{L}_1$		$\mathbf{L}_2$	
	#Speech	#Data	#Speech	#Data	#Speech	#Data	#Speech	#Data
Banking77	641	4110	584	1863	519	1800	599	1849
MultiWOZ	466	1571	479	1617	440	1574	499	1578
ECHR	519	3795	478	1735	453	1665	478	1670
S&P 500	472	3907	404	1784	373	1700	396	1689

For held-in perspectives, speech recordings within each category are divided into a training pool and an evaluation pool. Training instances are created by sampling categories and then sampling speech recordings from each selected category. Evaluation instances are organized into three levels: ① $\mathbf{L}_0$: seen perspectives and seen speech recordings, but unseen category/speech combinations. This evaluates recombination robustness. ② $\mathbf{L}_1$: seen perspectives but unseen speech recordings and unseen combinations. This evaluates generalization to new speech under familiar perspectives. ③ $\mathbf{L}_2$: unseen perspectives, unseen speech recordings, and unseen combinations. This evaluates cross-perspective generalization. The statistics of SpeechLens-Bench are summarized in Tab. 1. Details about the benchmark organization are provided in App. B.

3 METHOD

We propose SpeechLens-R1, an end-to-end LALM for speech multi-perspective clustering. The model is trained in two stages: reasoning distillation, which teaches comparison-based clustering behavior, and preference optimization, which further aligns the model toward higher-quality clustering decisions.

3.1 PROBLEM FORMULATION

Let $\mathcal{D} = \{a_1, a_2, \ldots, a_n\}$ denote a collection of speech segments, where each a_i is an audio recording. Let p denote a natural-language clustering perspective that specifies the criterion for grouping the recordings. The goal is to produce a partition

$$\mathcal{C} = \{C_1, C_2, \ldots, C_K\}, \tag{1}$$

where each $C_k \subseteq [n]$ is a non-empty cluster and the partition satisfies

$$C_k \cap C_{k'} = \emptyset \quad (k \neq k'), \qquad \bigcup_{k=1}^{K} C_k = [n]. \tag{2}$$

The number of clusters K is not provided as input and must be inferred by the model:

$$(K, \mathcal{C}) = f_\theta(\mathcal{D}, p). \tag{3}$$

Because clustering is permutation-invariant, cluster names and cluster order do not affect correctness. We therefore evaluate model outputs as unordered partitions over item indices. In practice, each audio segment is serialized with a 1-based index, and the model is required to generate a valid answer block that assigns every index exactly once.

3.2 REASONING DISTILLATION

Directly training an LALM to output final partitions can encourage shallow pattern imitation and does not explicitly teach the model how to compare audio segments under a perspective. We therefore first perform reasoning distillation (RD). For each training instance, we construct a gold partition from the benchmark annotations and ask a teacher model to synthesize a concise reasoning trace that leads to the gold clustering decision. The resulting distillation dataset is

$$\mathcal{D}_{\mathrm{RD}} = \{(x^{(i)}, y^{(i)}_{\mathrm{trace}})\}_{i=1}^{M}, \tag{4}$$

where $x^{(i)} = (\mathcal{D}^{(i)}, p^{(i)})$ contains the indexed audio set and the clustering perspective, and $y^{(i)}_{\text{trace}}$ contains a reasoning trace followed by the final clustering answer. We generate complementary traces for linguistic and paralinguistic perspectives: linguistic traces are grounded in transcribed content, while paralinguistic traces are grounded in audible cues. All traces are filtered to ensure that their final partitions match the gold partition and that the reasoning is comparison-based, concise, and modality-consistent. More details are provided in App. C.1.

We train the model with standard autoregressive supervised fine-tuning (SFT):

$$\mathcal{L}_{\text{RD}}(\theta) = -\sum_{(x,y)\in\mathcal{D}_{\text{RD}}} \sum_{t=1}^{|y|} \log p_\theta(y_t \mid x, y_{<t}). \tag{5}$$

This stage teaches the model to interpret the perspective, compare recordings, infer the number of clusters, and produce a structurally valid partition.

3.3 Preference Optimization

After reasoning distillation, we further optimize the model with direct preference optimization (DPO) as illustrated in Fig. 3. The goal is to improve clustering decisions while preserving the output format. For each input x, we construct a preference pair (x, y^+, y^-), where y^+ is the canonical gold partition and y^- is a valid but incorrect partition sampled from the RD model. We only keep rejected answers that are parsable, assign every item exactly once, and differ from the gold partition. This focuses preference learning on clustering errors rather than formatting failures.

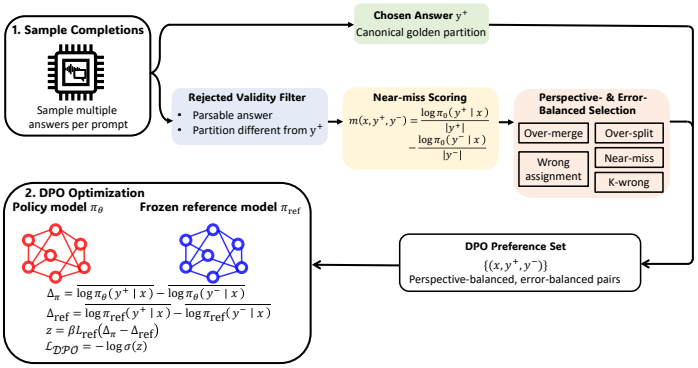


Figure 3: Overview of preference optimization pipeline.

To select informative pairs, we prioritize hard negatives that are close to the gold answer under the initial model or represent common structural clustering errors, such as over-merging, over-splitting, wrong cluster counts, or wrong item assignments. We also balance the selected pairs across perspectives and error types to avoid overfitting to a narrow class of mistakes. The detailed hard-pair selection procedure is described in App. C.2.

For each preference pair, we compute answer-token mean log-probabilities under the policy model π_θ and the frozen reference model π_{ref}. Let

$$\Delta_\pi = \log \pi_\theta(y^+ \mid x) - \log \pi_\theta(y^- \mid x), \quad \Delta_{\text{ref}} = \log \pi_{\text{ref}}(y^+ \mid x) - \log \pi_{\text{ref}}(y^- \mid x). \tag{6}$$

The DPO logit is

$$z = \beta L_{\text{ref}}(\Delta_\pi - \Delta_{\text{ref}}), \tag{7}$$

where L_{ref} is the average answer length in the DPO training set. The preference loss is

$$\mathcal{L}_{\text{DPO}} = -\log \sigma(z). \tag{8}$$

All log-probabilities are computed only over the canonical clustering answer. Thus, reasoning traces provide intermediate supervision during RD, while DPO directly optimizes the final decision.

4 Experimental Results

Baselines We compare our models with both previous paradigms and with different embedding-based clustering approaches. We use Whisper (Radford et al., 2023) as our standard ASR model to

Table 2: ARI comparison across four corpora. SpeechLens-R1 achieves the best overall ARI and demonstrates strong performance across most clustering settings. For embedding-based baselines, we report results with both K-Means and GMM clustering, while LLM/LALM-based methods directly produce clustering assignments. The best result is highlighted in **bold blue**, and the second-best result is underlined.

Method	Clustering	ECHR			S&P 500			Banking77			MultiWOZ			Average			Overall
		L_0	L_1	L_2	L_0	L_1	L_2	L_0	L_1	L_2	L_0	L_1	L_2	L_0	L_1	L_2	
ASR (Whisper (Radford et al., 2023)) + Text Embedding + Clustering																	
InBedder (Peng et al., 2024)	K-Means	4.98	4.24	8.82	9.90	5.71	5.42	8.29	6.83	6.83	4.73	4.77	8.81	6.97	5.39	7.47	6.61
	GMM	1.58	2.56	9.06	2.89	2.15	3.73	5.20	8.11	9.57	5.43	3.92	7.16	3.77	4.18	7.38	5.11
Instructor (Su et al., 2023)	K-Means	12.18	15.40	26.51	16.19	18.79	15.56	23.57	24.68	18.18	10.55	11.41	11.09	15.62	17.57	17.84	17.01
	GMM	8.44	9.68	17.30	9.37	9.46	6.98	14.79	13.13	14.65	7.30	7.66	6.48	9.97	9.98	11.35	10.44
Qwen3-Embedding (Zhang et al., 2025)	K-Means	11.70	9.89	15.37	11.20	11.92	7.26	16.94	16.62	16.90	8.55	12.66	8.34	12.10	12.77	11.97	12.28
	GMM	9.53	6.97	13.68	8.63	5.73	3.77	8.38	11.39	12.36	3.96	5.88	5.07	7.62	7.49	8.72	7.95
all-MiniLM-L6-v2 (Wang et al., 2020)	K-Means	12.93	14.32	24.95	16.19	14.36	13.89	19.72	23.28	18.51	9.81	9.96	14.47	14.66	15.48	17.96	16.03
	GMM	4.79	7.63	15.99	7.57	6.81	8.01	10.52	12.76	15.67	6.84	7.08	8.10	7.43	8.57	11.94	9.31
ASR (Whisper (Radford et al., 2023)) + LLM																	
GPT-4o (Hurst et al., 2024)	–	14.80	20.26	32.75	26.19	27.78	26.30	32.53	31.05	32.04	36.32	35.50	24.99	27.46	28.65	29.02	28.38
LALM																	
GPT-4o-audio-preview (OpenAI, 2025)	–	19.34	21.82	38.26	28.01	26.93	24.57	32.32	32.12	40.42	34.90	34.90	28.27	28.64	28.94	32.88	30.16
GPT-audio-1.5 (OpenAI, 2025)	–	18.76	20.20	40.24	25.56	27.75	26.71	29.77	29.57	46.79	37.99	37.21	40.82	28.02	28.68	38.64	31.78
Qwen3-omni-instruct-30B (Xu et al., 2025)	–	7.38	8.72	11.32	14.57	16.13	13.07	26.27	28.40	20.54	14.30	18.54	10.53	15.63	17.95	13.87	15.81
Audio Flamingo 3 (Goel et al., 2025)	–	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.57	1.17	0.98	0.14	0.29	0.25	0.23
Qwen2.5-omni (Qwen, 2025)	–	6.30	8.37	5.59	6.31	6.36	4.87	6.37	7.68	-0.79	13.97	15.25	8.58	8.24	9.42	4.56	7.41
SpeechLens-R1	–	**41.91**	**38.05**	**46.11**	**42.20**	**44.86**	**41.12**	**52.23**	**50.24**	**47.07**	**47.28**	**44.14**	**42.08**	**45.91**	**44.32**	**44.10**	**44.77**

Table 3: V-measure comparison across four corpora. The best result in each column is highlighted in **bold blue**, and the second-best result is underlined.

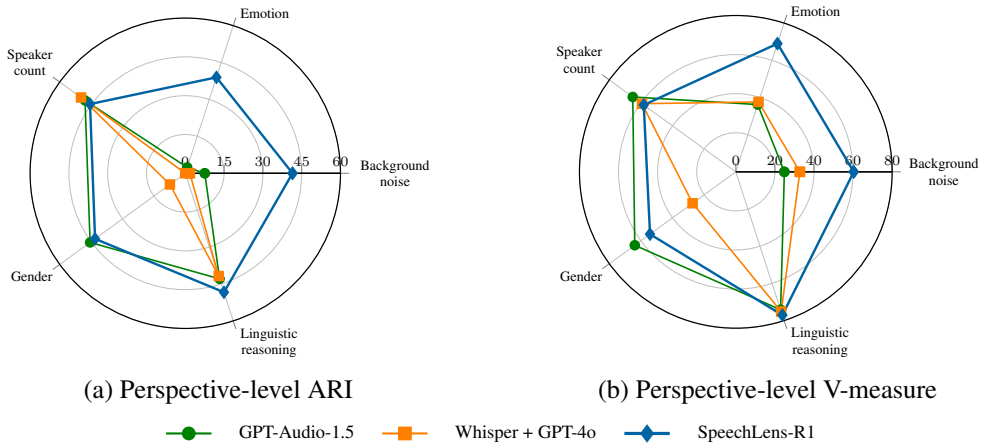
Method	Clustering	ECHR			S&P 500			Banking77			MultiWOZ			Average			Overall
		L_0	L_1	L_2	L_0	L_1	L_2	L_0	L_1	L_2	L_0	L_1	L_2	L_0	L_1	L_2	
ASR (Whisper (Radford et al., 2023)) + Text Embedding + Clustering																	
InBedder (Peng et al., 2024)	K-Means	50.71	51.51	57.84	53.69	50.88	48.79	52.13	52.87	50.09	30.24	29.38	29.55	46.69	46.16	46.57	46.47
	GMM	49.19	50.79	58.20	50.64	49.54	48.81	50.71	54.13	53.02	31.49	29.43	28.97	45.51	45.97	47.25	46.24
Instructor (Su et al., 2023)	K-Means	55.50	57.76	67.36	57.81	59.13	54.67	60.18	61.91	57.21	34.75	34.41	31.18	52.06	53.30	52.61	52.66
	GMM	53.50	55.07	62.85	54.41	53.58	50.40	56.26	57.21	55.96	32.98	32.15	28.51	49.29	49.50	49.43	49.41
Qwen3-Embedding (Zhang et al., 2025)	K-Means	55.12	54.87	61.72	55.11	54.51	50.05	57.61	58.61	56.49	33.51	35.38	29.19	50.34	50.84	49.36	50.18
	GMM	54.34	53.33	61.20	54.18	51.45	48.67	53.03	56.43	54.61	30.92	30.75	27.60	48.12	47.99	48.02	48.04
all-MiniLM-L6-v2 (Wang et al., 2020)	K-Means	55.68	57.33	66.83	57.97	56.61	53.83	58.34	61.16	57.07	34.22	33.67	33.55	51.55	52.19	52.82	52.19
	GMM	51.43	53.67	62.37	53.29	52.22	51.26	53.81	57.34	56.69	32.83	31.83	30.10	47.84	48.77	50.10	48.90
ASR (Whisper (Radford et al., 2023)) + LLM																	
GPT-4o (Hurst et al., 2024)	–	55.41	59.70	69.98	56.03	56.80	57.39	67.81	66.29	70.85	57.75	55.59	51.47	59.25	59.60	62.42	60.42
LALM																	
GPT-4o-audio-preview (OpenAI, 2025)	–	55.13	58.51	72.79	56.47	57.96	60.86	65.65	65.94	78.73	55.53	54.16	53.94	58.19	59.14	66.58	61.30
GPT-audio-1.5 (OpenAI, 2026)	–	53.73	53.60	73.52	54.54	53.39	62.09	66.59	64.10	**84.12**	57.68	57.11	61.23	58.14	57.05	70.24	61.81
Qwen3-omni-instruct-30B (Xu et al., 2025)	–	15.69	16.88	23.51	22.95	24.06	27.33	40.83	40.76	36.28	18.98	21.81	14.11	24.61	25.88	25.31	25.27
Audio Flamingo 3 (Goel et al., 2025)	–	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.66	1.31	1.10	0.17	0.33	0.28	0.26
Qwen2.5-omni (Qwen, 2025)	–	30.66	33.74	32.07	23.55	23.79	20.79	18.79	19.80	18.30	23.29	24.64	19.13	24.07	25.49	22.57	24.05
SpeechLens-R1	–	**72.77**	**75.30**	**78.08**	**75.02**	**73.32**	**71.61**	**79.88**	**78.79**	78.64	**68.91**	**67.19**	**61.67**	**74.15**	**73.65**	**72.50**	**73.43**

transcribe the audio. For three-step methods, we evaluate general-purpose embedding models (e.g., Qwen3-Embedding (Zhang et al., 2025), all-MiniLM-L6-v2 (Wang et al., 2020)) and instruction-tuned embedding models (e.g., InBedder (Peng et al., 2024), Instructor (Su et al., 2023)), each combined with classical clustering algorithms K-Means and GMMs. For instruction-tuned embedding models, we prepend the natural-language clustering perspective as the embedding instruction. For general-purpose embedding models, we encode the ASR transcript directly, as they do not support instruction-conditioned encoding. For two-step approaches, we use GPT-4o (Hurst et al., 2024) as the LLM for clustering. We also compare SpeechLens-R1 with off-the-shelf native LALMs/multimodal large language models (MLLMs): GPT-4o-audio-preview (OpenAI, 2025), GPT-audio-1.5 (OpenAI, 2026), Qwen3-omni-instruct-30B (Xu et al., 2025), Audio Flamingo 3 (Goel et al., 2025), Qwen2.5-omni (Qwen, 2025). For K-Means and GMM initialization, we provide the gold number of clusters to avoid confounding representation quality with cluster-number estimation. This gives these baselines an oracle advantage; in contrast, LLM/LALM-based methods must infer both the number of clusters and assignments from the input. More details are provided in App. D.

Implementation Details We adopt two metrics, i.e., V-Measure and Adjusted Rand Index (ARI), to evaluate the model performance, following established practice in clustering assessment (Cheng et al., 2023; Tipirneni et al., 2024; Liu et al., 2025). The formal definition of the metrics is included in App. F. We adopt Audio Flamingo 3 (Goel et al., 2025) as the base model due to its strong understanding and reasoning capabilities acquired during pre-training. The experiments are conducted on 4 NVIDIA H200 GPUs using the `TRL` library. App. E provides more details.

4.1 Main Results

Tabs. 2 and 3 compare SpeechLens-R1 against three categories of baselines: ASR-based text embedding followed by conventional clustering, ASR-based LLM clustering, and off-the-shelf native



(a) Perspective-level ARI (b) Perspective-level V-measure

Figure 4: Perspective-level performance averaged across the corpora. Each axis represents one clustering perspective, and each curve represents one model. Because polar coordinates cannot faithfully represent a negative radius, the ARI of -1.50 obtained by Whisper + GPT-4o for emotion clustering is displayed at 0. All other points show their exact values.

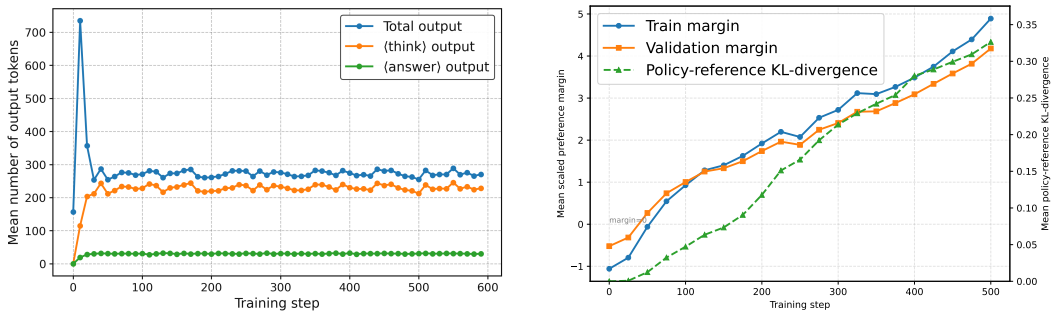
LALMs. Overall, SpeechLens-R1 achieves the best performance on both evaluation metrics, obtaining an overall ARI of 44.77 and an overall V-measure of 73.43. Notably, SpeechLens-R1 obtains the best ARI on all clustering settings and the best V-measure on 11 out of 12 settings. The gains are especially pronounced on ECHR, S&P 500, and Banking77, where SpeechLens-R1 consistently achieves the top results across most clustering conditions. Compared with the strongest competing system, GPT-audio-1.5 (OpenAI, 2026), SpeechLens-R1 improves the overall ARI by $+12.99$ absolute points. On V-measure, SpeechLens-R1 further surpasses the best baseline by $+11.62$ absolute points. These consistent improvements across two complementary clustering metrics demonstrate that SpeechLens-R1 not only recovers more accurate pairwise grouping structures, but also produces cluster assignments with better homogeneity and completeness. Additional results with Qwen2.5-Omni and failure analysis of Audio Flamingo 3 are provided in App. G.

Overall, the results validate the effectiveness of SpeechLens-R1 for speech multi-perspective clustering. The large margin over text-embedding baselines confirms the limitation of separating representation learning from clustering, while the improvement over ASR+LLM suggests that directly modeling speech can be beneficial for perspective-conditioned clustering, especially when the criterion depends on paralinguistic cues that may be lost in transcription. More importantly, the consistent gains over strong proprietary LALMs show that task-specific post-training enables SpeechLens-R1 to acquire clustering-oriented reasoning and decision-making capabilities beyond those of general-purpose LALMs.

4.2 Discussion

Perspective-level Performance The corpus-macro-averaged results are shown in Fig. 4. SpeechLens-R1 achieves the strongest average performance on background-noise, emotion, and linguistic-reasoning perspectives under both ARI and V-measure. The improvements are particularly obvious for background noise and emotion: SpeechLens-R1 obtains (41.45/60.35) and (39.00/69.00), respectively, compared with the strongest corresponding baseline results of (7.53/32.78) and (2.40/37.63). On linguistic reasoning, SpeechLens-R1 also achieves the best macro-average result of (48.38/77.10). These results indicate that the gains do not arise solely from lexical-semantic reasoning, but extend to perspectives that require direct modeling of acoustic and paralinguistic information.

The baselines exhibit more specialized behavior. Whisper+GPT-4o remains competitive on linguistic reasoning, where transcripts preserve much of the relevant clustering signal, but performs substantially worse on background noise, emotion, and gender. The performance of GPT-Audio-1.5 is considerably stronger but nevertheless uneven across perspectives, particularly for background noise and emotion. In contrast, SpeechLens-R1 exhibits a substantially more balanced profile: its average ARI ranges from 36.13 to 48.38 across the five perspectives, while its V-measure ranges from 54.25 to 77.10. This consistency suggests broader perspective-conditioned clustering ability rather than specialization toward either text-dominant or narrowly defined acoustic criteria.



(a) Response length decomposition.

(b) Preference margin and KL divergence.

Figure 5: (a) Evolution of mean output length, decomposed into total output tokens, <think> tokens, and <answer> tokens. (b) Evolution of the scaled preference margin and policy-reference KL divergence. The margin is computed as $m_{\text{scaled}} = L_{\text{ref}}\left(\overline{\log \pi_\theta(y^+ \mid x)} - \overline{\log \pi_\theta(y^- \mid x)}\right)$.

Ablation Study Tab. 4 studies the contribution of RD and DPO. The first row, which uses answer-only SFT with neither RD nor DPO, serves as the baseline. We observe that applying RD alone yields a modest improvement in overall ARI, increasing from 34.83 to 35.97, with a $+1.14$ absolute gain. However, its effect on V-measure is relatively small, improving the overall score by only $+0.28$. This suggests that RD can provide useful supervision for improving clustering assignments, but by itself, it does not consistently lead to better quality across all corpora.

Table 4: Ablation study of answer-only SFT, RD, and DPO. RD SFT denotes supervised fine-tuning with RD, while Answer-only DPO and RD+DPO denote DPO initialized from the corresponding checkpoints. Δ denotes the absolute improvement over the answer-only SFT baseline.

Metric	Corpus / Split	Answer-only SFT	RD SFT	Answer-only + DPO	RD + DPO
ARI	ECHR	33.17	33.85	35.38	**42.02**
	S&P 500	33.30	38.23	37.58	**42.73**
	Banking77	39.50	39.48	41.26	**49.85**
	MultiWOZ	33.34	32.34	34.07	**44.50**
	Overall	34.83	35.97	37.07	**44.77**
	Δ	–	+1.14	+2.24	**+9.94**
V-measure	ECHR	69.30	69.20	70.96	**75.38**
	S&P 500	66.27	69.88	69.90	**73.32**
	Banking77	71.21	72.51	74.13	**79.10**
	MultiWOZ	55.94	52.26	54.56	**65.92**
	Overall	65.68	65.96	67.39	**73.43**
	Δ	–	+0.28	+1.71	**+7.75**

DPO alone provides a stronger gain than RD alone. Compared with the baseline, DPO improves the overall ARI from 34.83 to 37.07, corresponding to a $+2.24$ absolute improvement, and improves the overall V-measure from 65.68 to 67.39, with a $+1.71$ gain. The improvement is especially clear on ECHR, S&P 500, and Banking77, indicating that preference optimization helps the model better align its outputs with desirable clustering structures. Nevertheless, DPO alone still shows limited improvement in some cases, such as MultiWOZ in V-measure, suggesting that preference learning without additional reasoning supervision may not fully resolve more challenging ambiguities.

The best performance is obtained when RD and DPO are combined. This setting achieves 44.77 overall ARI and 73.43 overall V-measure, outperforming the baseline by $+9.94$ and $+7.75$ absolute points, respectively. Importantly, the combined setting improves consistently across all four corpora for both metrics. Compared with DPO alone, adding RD further improves overall ARI by $+7.70$ and V-measure by $+6.04$, showing that RD and DPO are complementary rather than redundant. These results suggest that RD provides a stronger intermediate reasoning signal, while DPO further calibrates the model toward preferred clustering decisions. Together, they substantially improve both pairwise clustering agreement and cluster-level homogeneity/completeness, demonstrating the effectiveness of combining reasoning-based supervision with preference optimization.

Response-format Stabilization Fig. 5(a) shows the mean number of generated tokens across training steps, with separate measurements for the complete model output and for tokens enclosed within the prescribed <think> and <answer> fields. At the beginning of training, the total output length exhibits a sharp transient spike, whereas the counted <think> and <answer> tokens remain substantially lower. This discrepancy indicates that the model initially fails to consistently

follow the required response format, producing a large number of extraneous tokens outside the designated fields. After this short adaptation phase, the three curves stabilize: the <think> segment accounts for the majority of the response, and the <answer> segment remains compact and stable. These trends suggest that training rapidly improves structural adherence and suppresses uncontrolled generation, leading to a consistent output format throughout the remaining optimization process.

Preference Alignment and Controlled Policy Shift Fig. 5(b) shows that DPO training steadily increases the model's preference margin while also inducing a gradual divergence from the reference model. At the beginning, both training and validation margins are negative, indicating that the policy does not yet assign a higher likelihood to the chosen responses than to the rejected ones. As training proceeds, both margins rapidly cross the zero-margin boundary and continue to increase, suggesting that DPO effectively shifts the policy toward preference-aligned outputs. The validation margin closely tracks the training margin across training, with only a moderate gap emerging in later stages, which suggests that the learned preference signal transfers beyond the training probe rather than merely memorizing the optimization examples. Meanwhile, the policy-reference KL-divergence increases smoothly from near zero to a moderate value, indicating that the policy gradually departs from the reference model as preference optimization progresses. The KL curve grows in a controlled manner rather than exhibiting abrupt spikes, suggesting that the improvement is achieved through a stable distributional shift.

5 Related Work

Traditional and Embedding-based Clustering Clustering is a fundamental problem in machine learning, with classical methods such as K-Means (MacQueen, 1967; Lloyd, 1982) and GMMs (Dempster et al., 1977) widely used to partition data. Recent representation learning methods improve clustering by using pretrained encoders. In speech processing, wav2vec 2.0 (Baevski et al., 2020) and HuBERT (Hsu et al., 2021) learn rich acoustic representations. Instruction-aware embedding models, including Instructor (Su et al., 2023) and InBedder (Peng et al., 2024), further condition representations on natural-language instructions.

LLMs and Reasoning-based Clustering LLMs have recently been explored as components in clustering systems (Zhang et al., 2023; Viswanathan et al., 2024; De Raedt et al., 2023; Feng et al., 2024). More recently, reasoning-based clustering methods such as Cluster-R1 (Qing et al., 2026) formulate clustering as a generative reasoning task. These methods show the potential of using LLM reasoning for flexible structure discovery.

Audio-Language Models and Multimodal Reasoning Recent LALMs, such as SpeechT5 (Ao et al., 2022), AudioLM (Borsos et al., 2023), and Qwen2-Audio (Chu et al., 2024), enable end-to-end modeling of speech and text within unified frameworks. Despite this progress, prior work has largely focused on speech recognition, generation, or audio understanding, leaving structure discovery over speech collections underexplored. Our work addresses this gap by studying speech multi-perspective clustering with end-to-end LALMs.

6 Conclusion

We introduced speech multi-perspective clustering, a new task that requires models to organize a collection of speech recordings according to a natural-language perspective while inferring both the number of clusters and the cluster assignments. To support this task, we constructed SpeechLens-Bench, a benchmark spanning diverse application domains and combining linguistic and paralinguistic clustering perspectives. We further proposed SpeechLens-R1, an end-to-end LALM trained with reasoning distillation and preference optimization to perform perspective-conditioned structure discovery directly from speech. Experiments show that SpeechLens-R1 consistently outperforms the baselines across ARI and V-measure. These results suggest that native LALMs can serve as flexible clustering agents for speech collections, opening new directions for perspective-conditioned organization, retrieval, and analysis of audio data.

AI USE STATEMENT

In this work, we used generative AI tools for generating synthetic data sets, providing feedback on research methodology or experiments, implementing methods, cleaning and reformating dataset, supporting qualitative and thematic data analysis. We have not used generative AI tools for helping develop theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, interpreting results, and assisting with translation is not applicable to this work. Additionally, we used generative AI tools for editing software code, and brainstorming. We have reviewed all AI-assisted work. We checked the generated synthetic data sets and the code implemented by AI. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Junyi Ao, Rui Wang, Long Zhou, Chengyi Wang, Shuo Ren, Yu Wu, Shujie Liu, Tom Ko, Qing Li, Yu Zhang, et al. Speecht5: Unified-modal encoder-decoder pre-training for spoken language processing. In *Annual Meeting of the Association for Computational Linguistics*, 2022. Available: https://aclanthology.org/2022.acl-long.393.pdf.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Processing Systems*, 2020. Available: https://arxiv.org/pdf/2006.11477.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, et al. Audiolm: a language modeling approach to audio generation. *Transactions on Audio, Speech, and Language Processing*, 2023. Available: https://arxiv.org/pdf/2209.03143.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. Efficient intent detection with dual sentence encoders. In *2nd Workshop on Natural Language Processing for Conversational AI*, 2020. Available: https://aclanthology.org/2020.nlp4convai-1.5.pdf.

Luyao Cheng, Siqi Zheng, Qinglin Zhang, Hui Wang, Yafeng Chen, Qian Chen, and Shiliang Zhang. Improving speaker diarization using semantic information: joint pairwise constraints propagation. *arXiv preprint arXiv:2309.10456*, 2023. Available: https://arxiv.org/pdf/2309.10456.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, et al. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*, 2024. Available: http://arxiv.org/pdf/2407.10759.

Ann Clifton, Sravana Reddy, Yongze Yu, Aasish Pappu, Rezvaneh Rezapour, Hamed Bonab, Maria Eskevich, Gareth Jones, Jussi Karlgren, Ben Carterette, et al. 100,000 podcasts: A spoken english document corpus. In *International Conference on Computational Linguistics*, 2020. Available: https://aclanthology.org/2020.coling-main.519.pdf.

Maarten De Raedt, Fréderic Godin, Thomas Demeester, and Chris Develder. Idas: Intent discovery with abstractive summarization. In *5th Workshop on NLP for Conversational AI*, 2023. Available: https://aclanthology.org/2023.nlp4convai-1.7.pdf.

Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 1977. Available: https://www.ece.iastate.edu/~namrata/EE527_Spring08/Dempster77.pdf.

Xingjian Diao, Chunhui Zhang, Keyi Kong, Weiyi Wu, Chiyu Ma, Zhongyu Ouyang, Peijun Qing, Soroush Vosoughi, and Jiang Gui. Soundmind: Rl-incentivized logic reasoning for audio-language models. In *Conference on Empirical Methods in Natural Language Processing*, 2025. Available: https://aclanthology.org/2025.emnlp-main.27.pdf.

Zijin Feng, Luyang Lin, Lingzhi Wang, Hong Cheng, and Kam-Fai Wong. Llmedgerefine: Enhancing text clustering with llm-based boundary point refinement. In *Conference on Empirical Methods in Natural Language Processing*, 2024. Available: https://aclanthology.org/2024.emnlp-main.1025.pdf.

Arushi Goel, Sreyan Ghosh, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang-gil Lee, Chao-Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael Valle, et al. Audio flamingo 3: Advancing audio intelligence with fully open large audio language models. *arXiv preprint arXiv:2507.08128*, 2025. Available: https://arxiv.org/pdf/2507.08128.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *Transactions on Audio, Speech, and Language Processing*, 2021. Available: https://arxiv.org/pdf/2106.07447.

Hangrui Hu, Xinfa Zhu, Ting He, Dake Guo, Bin Zhang, Xiong Wang, Zhifang Guo, Ziyue Jiang, Hongkun Hao, Zishan Guo, et al. Qwen3-tts technical report. *arXiv preprint arXiv:2601.15621*, 2026. Available:https://arxiv.org/pdf/2601.15621.

Yebowen Hu, Timothy Ganter, Hanieh Deilamsalehy, Franck Dernoncourt, Hassan Foroosh, and Fei Liu. Meetingbank: A benchmark dataset for meeting summarization. In *Annual Meeting of the Association for Computational Linguistics*, 2023. Available: https://aclanthology.org/2023.acl-long.906.pdf.

Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 1985. Available: https://link.springer.com/article/10.1007/BF01908075.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. Available: https://arxiv.org/pdf/2410.21276.

jlohding. Sp500-edgar-10k: Annual reports of s&p 500 companies from sec filings, 2023. Available: https://huggingface.co/datasets/jlohding/sp500-edgar-10k.

Martha Larson and Gareth JF Jones. Spoken content retrieval: A survey of techniques and technologies. *Foundations and Trends® in Information Retrieval*, 2012. Available: https://www.emerald.com/ftinr/article-pdf/5/4-5/235/11085594/1500000020en.pdf.

Jianghan Liu, Ziyu Shang, Wenjun Ke, Peng Wang, Zhizhao Luo, Jiajun Liu, Guozheng Li, and Yining Li. Llm-guided semantic-aware clustering for topic modeling. In *Annual Meeting of the Association for Computational Linguistics*, 2025. Available: https://aclanthology.org/2025.acl-long.902.pdf.

Stuart Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 1982. Available: https://ieeexplore.ieee.org/document/1056489.

J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1967. Available: https://cir.nii.ac.jp/crid/1570572699967325952.

OpenAI. GPT-4o Audio Model, 2025. Available: https://developers.openai.com/api/docs/models/gpt-4o-audio-preview.

OpenAI. GPT-Audio 1.5 Model, 2026. Available: https://developers.openai.com/api/docs/models/gpt-audio-1.5.

Tae Jin Park, Naoyuki Kanda, Dimitrios Dimitriadis, Kyu J Han, Shinji Watanabe, and Shrikanth Narayanan. A review of speaker diarization: Recent advances with deep learning. *Computer Speech & Language*, 2022. Available: https://www.sciencedirect.com/science/article/abs/pii/S0885230821001121.

Letian Peng, Yuwei Zhang, Zilong Wang, Jayanth Srinivasa, Gaowen Liu, Zihan Wang, and Jingbo Shang. Answer is all you need: Instruction-following text embedding via answering the question. In *Annual Meeting of the Association for Computational Linguistics*, 2024. Available: https://aclanthology.org/2024.acl-long.27.pdf.

Prakash Poudyal, Jaromír Šavelka, Aagje Ieven, Marie Francine Moens, Teresa Goncalves, and Paulo Quaresma. Echr: Legal corpus for argument mining. In *7th Workshop on Argument Mining*, 2020. Available: https://aclanthology.org/2020.argmining-1.8.pdf.

Peijun Qing, Puneet Mathur, Nedim Lipka, Varun Manjunatha, Ryan Rossi, Franck Dernoncourt, Saeed Hassanpour, and Soroush Vosoughi. Cluster-r1: Large reasoning models are instruction-following clustering agents. *arXiv preprint arXiv:2603.23518*, 2026. Available: https://arxiv.org/pdf/2603.23518.

Team Qwen. Qwen2.5-omni technical report. *arXiv preprint arXiv:2503.20215*, 2025. Available: https://arxiv.org/pdf/2503.20215.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pp. 28492–28518, 2023. Available: https://arxiv.org/pdf/2212.04356.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 2019. Available: https://aclanthology.org/D19-1410.pdf.

Andrew Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 2007. Available: https://aclanthology.org/D07-1043.pdf.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A Smith, Luke Zettlemoyer, and Tao Yu. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, 2023. Available: https://aclanthology.org/2023.findings-acl.71.pdf.

Sindhu Tipirneni, Ravinarayana Adkathimar, Nurendra Choudhary, Gaurush Hiranandani, Rana Ali Amjad, Vassilis N Ioannidis, Changhe Yuan, and Chandan K Reddy. Context-aware clustering using large language models. *arXiv preprint arXiv:2405.00988*, 2024. Available: https://arxiv.org/pdf/2405.00988.

Vijay Viswanathan, Kiril Gashteovski, Carolin Lawrence, Tongshuang Wu, and Graham Neubig. Large language models enable few-shot clustering. *Transactions of the Association for Computational Linguistics*, 2024. Available: https://aclanthology.org/2024.tacl-1.18.pdf.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in Neural Information Processing Systems*, 2020. Available: https://proceedings.neurips.cc/paper/2020/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, et al. Qwen3-omni technical report. *arXiv preprint arXiv:2509.17765*, 2025. Available: https://arxiv.org/pdf/2509.17765.

Xiaoxue Zang, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Jianguo Zhang, and Jindong Chen. Multiwoz 2.2: A dialogue dataset with additional annotation corrections and state tracking baselines. In *2nd Workshop on Natural Language Processing for Conversational AI*, 2020. Available: https://aclanthology.org/2020.nlp4convai-1.13.pdf.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, et al. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025. Available: https://arxiv.org/pdf/2506.05176.

Yuwei Zhang, Zihan Wang, and Jingbo Shang. Clusterllm: Large language models as a guide for text clustering. In *Conference on Empirical Methods in Natural Language Processing*, 2023. Available: https://aclanthology.org/2023.emnlp-main.858.pdf.

CONTENTS OF APPENDIX

A Corpus-Specific Benchmark Construction **15**

A.1 Shared Construction Pipeline 15

A.2 Banking77 17

A.3 ECHR 18

A.4 S&P 500 19

A.5 MultiWOZ 19

A.6 Quality Control 20

A.7 Text-to-Speech system 21

B Benchmark Split Construction **22**

B.1 Held-in and Held-out Perspectives 22

B.2 Training Instance Construction 23

B.3 Evaluation Levels 23

B.4 Episode Construction and Benchmark Statistics 23

C Additional Method Details **25**

C.1 Reasoning Distillation Details 25

C.2 Preference Optimization Details 29

D Baseline Implementation Details **31**

E Implementation Config Details **32**

E.1 Reasoning Distillation 32

E.2 Direct Preference Optimization 32

F Evaluation Metrics **33**

G More Analysis **34**

G.1 Case Study 35

H Limitations **37**

I Ethical Considerations **38**

A CORPUS-SPECIFIC BENCHMARK CONSTRUCTION

This appendix provides additional details for the construction of SpeechLens-Bench. In the main paper, we use the term *clustering perspective* to denote a natural-language criterion for grouping audio recordings. In some data-generation prompts, we use the internal term *dimension*; each retained dimension is converted into a clustering perspective in the benchmark.

A.1 SHARED CONSTRUCTION PIPELINE

All corpora follow the same high-level pipeline: perspective induction, perspective refinement, transcript construction, transcript validation, and audio synthesis.

Perspective Induction For each source corpus, we sample representative examples and prompt an LLM to propose candidate clustering perspectives. Each candidate's perspective contains a short description and a set of category labels. The goal is to induce perspectives that reflect meaningful reasoning factors in the corpus, such as communicative intent, interaction pattern, procedural structure, discourse function, risk type, or consequence, rather than superficial lexical overlap.

Perspective generation prompt

```
You are an expert in taxonomy design. Given (i) a description of a
    dataset and (ii) example text entries from it, your task is to
    propose meaningful clustering dimensions that capture distinct
    reasoning-based perspectives.
For each clustering dimension:
1. Provide a concise description of what the dimension represents.
2. List the possible cluster labels under this dimension.
3. Justify why each cluster is distinct, highlighting reasoning
    factors such as cause, intent, context, or outcome.
Requirements:
    - Propose 5--8 clustering dimensions.
    - For each dimension, propose 4--8 mutually exclusive cluster
    labels.
    - Clusters should be collectively as exhaustive as possible.
    - Avoid vague labels such as "Other", "Misc", or "General".
For each dimension, provide:
    - id
    - name
    - description
    - categories
For each category, provide:
    - id
    - label
    - description (what it is + why it's distinct)
Output STRICTLY as JSON with key "dimensions".
(i) Dataset description: {DATASET_DESCRIPTION}
(ii) Example text entries:
{EXAMPLES}
Please generate the clustering dimensions and taxonomies.
The JSON schema must be:
{
  "dimensions": [
    {
      "id": "dim_01_xxx",
      "name": "...",
      "description": "...",
      "categories": [
        {
          "id": "cat_01_xxx",
          "label": "...",
          "description": "..."
```

```
          }
        ]
      }
    ]
}
```

Perspective Refinement We refine each proposed perspective to ensure that its categories are mutually exclusive, semantically interpretable, and comparable in granularity. We remove perspectives with vague category boundaries, highly imbalanced categories, insufficient instance support, or categories that can be solved through shallow lexical cues. For each retained perspective, we generate a neutral clustering instruction that describes the grouping criterion without revealing the underlying category names.

Perspective refinement prompt

```
You are a taxonomy refinement expert. Your goal is to revise the given
     taxonomy so that it is clear, consistent, and practically usable.
     Follow the requirements below:
Requirements:
    - Category name: Max {NAME_MAX_WORDS} words; concise, specific,
    and informative.
    - Description: Max {DESCRIPTION_MAX_WORDS} words; clearly explain
    what distinguishes this category.
    - Dimensions: Avoid near-duplicate dimensions that rely on the
    same primary signal. However, moderate correlation between
    dimensions is acceptable when conceptually justified.
    - Mutual exclusivity: Categories must not overlap or contradict
    each other. Within each dimension, categories should be defined so
     that most cases naturally fit ONLY ONE category.
    - Collective exhaustiveness: Categories together should cover all
    possible intents in the given context.
    - Granularity: All categories must be defined at the same level of
     specificity.
    - No vague labels: Avoid terms like "Other," "General," or "
    Miscellaneous."
    - Each category must be assignable using signals observable in the
     text or reasoning from the text.
[Dataset Context]
{DATASET_DESCRIPTION}
[Original Taxonomy]
{TAXONOMY_DRAFT}
Task:
1. Review the existing taxonomy and suggest improvements (e.g.,
    renaming, merging, splitting, or adding new categories).
2. Ensure each category has a clear reasoning justification and aligns
     with the data context.
3. If categories are missing, add enough to make the taxonomy
    collectively exhaustive.
4. Based on the refined taxonomy, check the instruction and provide a
    neutral clustering instruction that accurately reflects the
    categorization principle without revealing category labels. Do not
     include or hint at the actual category names in the instruction.
Instruction Style Requirements:
    - For each dimension, generate one neutral clustering instruction.
    - Each instruction MUST start with "Cluster" or "Group"
    - Keep each instruction within 25 words.
    - Do NOT mention the taxonomy, dimensions, or categories
    explicitly. Do NOT include or hint at any category names.
    - Emphasize selecting the single most dominant factor in the case.
Output Format:
```

```
Return a JSON object with:
1. "refined_dimensions": The updated taxonomy list (same structure as
    input dimensions: id, name, description, categories with id, label
    , description).
2. "clustering_instructions": For each dimension, a neutral
    instruction (list or dict keyed by dimension id).
3. "change_log": A brief explanation of what was merged/split/added
    and why.
```

Transcript Generation The transcript-generation procedure depends on the source format. For short intent queries, we expand the input into realistic spoken scripts. For long-form formal documents, we first compress the text into an evidence-preserving summary and then rewrite it into a spoken script. For dialogue-native corpora, we reuse the original dialogue structure. Across all corpora, the transcript is required to preserve the evidence needed for the assigned perspective labels.

Transcript Validation We validate each generated transcript by asking independently prompted classifiers to assign it to a category under the corresponding perspective. An instance is retained only when the predicted label agrees with the intended label. This filtering step reduces label drift introduced by expansion, compression, or spoken-script rewriting. We also filter transcripts that explicitly mention category names, perspective names, or near-verbatim instruction phrases, preventing models from solving the task through lexical shortcuts.

Audio Synthesis and Paralinguistic Injection Validated transcripts are converted into speech using a text-to-speech (TTS) system. More details are discussed in App. A.7. During synthesis, we inject controlled paralinguistic attributes, including emotion, speaker identity, speaker count, and background acoustic condition. For monologues, we synthesize a single speaker; for dialogues, we synthesize multiple speakers and concatenate turns according to the dialogue structure. For semantic perspectives, paralinguistic attributes are randomized and approximately balanced across categories. For paralinguistic perspectives, the corresponding acoustic attribute is used as the target clustering criterion, while other attributes are randomized to reduce confounding.

A.2 BANKING77

Banking77 (Casanueva et al., 2020) consists of short customer-service intent queries. Because the original queries are too short to serve as natural spoken recordings, we first classify each query under the refined perspectives and then expand classifiable queries into spoken scripts. The expanded scripts may be monologues, such as a customer describing a banking issue, or short dialogues, such as an exchange between a customer and an agent. The expansion prompt is constrained to preserve the assigned perspective labels while making the script realistic and conversational.

Script generation prompt for Banking77

```
You are a specialized Banking AI with two distinct modes: a precise
    Analyst and a creative Scriptwriter.
### Mode 1: Precise Analyst
Analyze the input sentence against the provided taxonomy.
- USE CATEGORY NAME ONLY for classification.
- CRITICAL: If the sentence cannot be clearly mapped to a category in
    ANY of the dimensions, set that dimension to `None`.
- If ANY dimension is `None`, the `speech_script` field MUST be `None
    `.
[Dataset Context]
{DATASET_DESCRIPTION}
[Taxonomy]
{TAXONOMY}
### Mode 2: Creative Scriptwriter
Expand the content into a 150-200 word TTS script.
```

```
Goal: Create a diverse, realistic banking scenario.
Ambiguity: The script must be high-fidelity to the classification
    results. Avoid any content that would make the categories hard to
    distinguish.
Diversity Encouragement:
    - Speaker Choice: Feel free to use a Monologue (e.g., a customer's
     thought process, a voicemail) or a Dialogue (e.g., customer vs.
    agent, two friends discussing a bank issue). Let the context
    decide.
    - Dialogue Format:
        - If 2 speakers: Use "Speaker A: ..." and "Speaker B: ..." on
    new lines.
        - If 1 speaker: Use "Narrator: ..."
    - Narrative Flow: You are NOT limited to an FAQ format. You may
    weave the original concern into a story, a phone call, or a help-
    desk interaction.
    - Naturalism: Use spoken-language markers (pauses, "uhm", "right",
     "okay"). The script should sound like it's happening in real life
    .
    - Tone: Match the tone to the 'Customer Harm and Urgency'
    dimension (e.g., calm for info, tense for security risks).
```

After expansion, we validate each script by re-classifying the generated transcript under the same perspective. Scripts are retained only if their transcript-level labels match the original query-level labels. The validated scripts are then synthesized into speech with sampled emotion, speaker identity, and background acoustic conditions. Speaker count is determined by the script format: monologues are synthesized with one speaker, while dialogues use multiple speakers.

Script validation prompt for Banking77

```
You are a specialized Banking AI classifier. Your task is to analyze
    the provided speech script and classify it based on the taxonomy
    dimensions.
Classification Rules:
    - For each dimension in the taxonomy, determine which category the
     script belongs to.
    - Use ONLY the CATEGORY NAME from the provided taxonomy.
    - If the script cannot be clearly mapped to a category in ANY
    dimension, set that dimension to "None".
    - Crucial: You must also include the "original_sentence" provided
    in the input in your final JSON output without any changes.
[Dataset Context]
{DATASET_DESCRIPTION}
[Taxonomy]
{TAXONOMY}
Return ONLY a JSON object containing the "classification_results".
```

A.3 ECHR

ECHR (Poudyal et al., 2020) contains long-form legal case documents. We first sample case documents and induce legal-case perspectives from representative examples. The retained perspectives capture substantive legal and factual reasoning factors, such as the interest at stake, the type of state action, the procedural context, the affected population, or the form of harm.

Because the original documents are too long and formal for direct speech synthesis, we compress each accepted case into a shorter evidence-preserving summary. The compression is label-aware: the summary must retain the information required to recover all accepted perspective labels. We then validate the compressed summary by re-classifying it under the same perspectives. Only summaries whose labels remain recoverable are rewritten into spoken scripts.

The final spoken scripts are synthesized with one to four speakers, sampled emotion, and background acoustic conditions. Paralinguistic attributes are assigned in an approximately balanced way within semantic strata to avoid creating shortcuts between acoustic factors and semantic labels.

A.4 S&P 500

The S&P 500 corpus (jlohding, 2023) consists of excerpts from SEC Form 10-K annual reports. The source texts cover company properties, legal proceedings, market information, dividends, and related stockholder matters. We induce clustering perspectives that reflect company-level patterns, such as asset ownership, geographic distribution, operational structure, litigation exposure, regulatory risk, shareholder payout policy, and capital-market behavior.

As with ECHR, the original disclosures are long and formal. We therefore compress each accepted text into a concise evidence-preserving summary and validate whether the accepted labels remain recoverable after compression. Validated summaries are then rewritten into spoken scripts. The synthesis stage injects emotion, speaker count, speaker identity, and background acoustic condition, while randomizing non-target paralinguistic attributes to reduce spurious correlations.

A.5 MULTIWOZ

MultiWOZ (Zang et al., 2020) is already dialogue-native, so it does not require expansion from short queries or compression from long documents. We sample dialogues and induce perspectives that capture task-oriented interaction patterns, user goals, constraint evolution, request structure, and dialogue-level behavior. Each dialogue is labeled under the refined perspectives by independently prompted classifiers, and we retain only dialogues with consistent label assignments.

The original dialogue text is reused as the transcript. During TTS, each speaker role is assigned a distinct voice, and dialogue turns are synthesized in order. We additionally inject emotion and background acoustic variation. This preserves the multi-turn structure of MultiWOZ while converting it into a speech-based benchmark instance.

Tab. 5 summarizes the corpus-specific transcript construction, validation, and TTS paths used in our synthesis pipeline.

Dataset description for taxonomy generation

```
ECHR: This dataset contains English-language case documents from the
    European Court of Human Rights (ECtHR). Each entry is a legal case
     narrative written in formal legal style, describing factual
    background, procedural history, evidence, and legal context. The
    text often includes details about actors (individuals, authorities
    , courts), actions (detention, investigations, speech restrictions
    , searches, employment disputes, etc.), timelines, and outcomes in
     domestic proceedings. The goal is to discover reasoning-based
    clustering perspectives for grouping cases. Dimensions should
    reflect substantive legal and factual reasoning (e.g., the right
    or interest at stake, the type of state action, procedural context
    , affected population, type of harm, or dispute structure), rather
     than superficial keywords or document length.
S&P 500: This dataset contains excerpts from SEC Form 10-K annual
    reports of S&P 500 companies. Each entry is a section from a 10-K
    filing, written in formal financial/legal style. The data includes
     three item types:
    - Properties: Descriptions of company properties, facilities, real
     estate holdings, and physical assets.
    - Legal Proceedings: Disclosures of litigation, regulatory
    proceedings, and legal risks.
    - Market for Registrar's Common Equity: Information on stock
    performance, dividends, equity markets, and related stockholder
    matters.
```

```
The goal is to discover clustering perspectives for grouping these S&P
     500 companies. Focus on company-level patterns reflected in the
    disclosures, such as:
    - asset ownership and utilization
    - geographic distribution of operations
    - industry-specific asset composition
    - exposure to litigation or regulatory risk
    - shareholder payout policies
    - capital structure or equity instruments
Banking77: This is a single-domain intent mining dataset consisting of
     user queries related to banking services. The dataset is
    characterized by subtle semantic differences between intents,
    where similar surface expressions may correspond to different
    underlying user goals. Focus on distinguishing fine-grained intent
     differences and capturing the exact user goal expressed in each
    query.
MultiWOZ: This is a multi-domain task-oriented dialogue dataset
    consisting of human-human conversations between a user and a
    system. Each dialogue spans one or more domains and involves
    completing tasks. The user's intent evolves across turns and may
    include constraints and requests. Focus on identifying the user's
    underlying intent at each turn and how it evolves across the
    dialogue, rather than treating each utterance independently.
```

Table 5: Corpus-specific transcript generation, validation, and TTS paths. Banking77 uses expansion from short intent queries; ECHR and S&P 500 use compression followed by spoken rewriting; MultiWOZ directly reuses native dialogues.

Corpus	Transcript Generation	Validation	TTS Path
Banking77	Classify each short intent query under the refined taxonomy, then expand each classifiable query into a natural spoken transcript (monologue or short dialogue).	Re-classify the expanded transcript under the same taxonomy and retain it only if all transcript-level labels exactly match the original query-level assignments.	Synthesize validated transcripts with sampled emotion and speaker identity; use one speaker for monologues and multiple speakers for dialogues; augment the waveform with background-noise conditions.
ECHR	Assign taxonomy labels to long legal case texts, compress each accepted case into an evidence-preserving summary, and rewrite the summary into a spoken script.	Use dual-model agreement for source-text labeling, then re-classify the compressed text and retain it only if all previously accepted dimensions remain recoverable after compression.	Synthesize accepted spoken scripts with one to four speakers, sampled emotion, and background-noise conditions, with approximate balancing of paralinguistic factors within semantic strata.
S&P 500	Assign taxonomy labels to 10-K disclosure segments, compress each accepted document into an evidence-preserving summary, and rewrite the summary into a spoken script.	Use dual-model agreement for source-text labeling, then re-classify the compressed text and retain it only if all accepted dimensions are preserved after compression.	Synthesize accepted spoken scripts with sampled emotion, speaker count, and background-noise condition, again using a balanced assignment scheme within semantic strata.
MultiWOZ	Directly reuse sampled human-written dialogues as transcripts after assigning taxonomy labels.	Retain only dialogues for which two independent classifiers produce identical label assignments across all taxonomy dimensions.	Synthesize the validated dialogue turn by turn with distinct speakers for different roles, while injecting sampled emotion and background-noise variation.

A.6 Quality Control

We apply several quality-control steps to make the benchmark reliable and to reduce shortcut learning.

First, each retained perspective must define categories that are mutually exclusive, interpretable, and sufficiently supported by examples. Perspectives with ambiguous decision boundaries, severe class imbalance, or categories that rely on superficial lexical cues are removed.

Second, transcript construction is validated by label-preservation checks. After expansion, compression, or rewriting, each transcript is re-classified under the corresponding perspective. Only transcripts whose labels can be recovered are retained.

Third, we filter explicit lexical leakage. Generated transcripts must not directly contain perspective names, category names, or near-verbatim instruction phrases. This prevents models from relying on trivial string matching.

Fourth, paralinguistic attributes are controlled during synthesis. For semantic perspectives, acoustic attributes are randomized and approximately balanced across categories. For paralinguistic perspectives, the target acoustic attribute defines the label, while other attributes are randomized. This reduces unintended correlations between semantic labels and acoustic conditions.

Tab. 6 shows the representative perspectives of each adopted corpus.

Table 6: Corpus-specific clustering perspectives.

Corpus	Held-in perspectives	Held-out perspectives
Banking77	• Cluster requests by the customer's primary intended outcome • Cluster requests by the main banking problem or request category • Cluster requests by the dominant urgency, severity, or risk level • Cluster clips by the dominant emotional tone in speech • Cluster clips by the exact number of distinct speakers • Cluster clips by the real-world background-noise scene	• Cluster requests by the banking capability or product domain most directly involved • Cluster requests by the primary route through which the issue would be resolved • Cluster clips by the dominant speaker-gender pattern
ECHR	• Cluster cases by the dominant State act or omission producing the alleged violation • Cluster cases by the procedural or situational context of the dispute • Cluster cases by the structure of the dispute and party configuration • Cluster clips by the dominant emotional tone in speech • Cluster clips by the exact number of distinct speakers • Cluster clips by the real-world background-noise scene	• Cluster cases by the principal protected interest implicated • Cluster cases by the main kind of harm or outcome alleged • Cluster cases by the most salient applicant status affecting power imbalance or risk • Cluster clips by the dominant speaker-gender pattern
S&P 500	• Cluster firms by the primary way key properties are controlled • Cluster firms by the geographic breadth and concentration of properties • Cluster firms by the dominant type of legal matter discussed • Cluster firms by the dominant signal about legal-matter status and potential financial impact • Cluster clips by the dominant emotional tone in speech • Cluster clips by the exact number of distinct speakers • Cluster clips by the real-world background-noise scene	• Cluster disclosures by the dominant type of physical properties central to operations • Cluster disclosures by the dominant approach to shareholder return • Cluster disclosures by the dominant approach to repurchases or equity structure • Cluster clips by the dominant speaker-gender pattern
MultiWOZ	• Cluster dialogues by the presence and source of explicit repair turns • Cluster dialogues by how users provide, accumulate, revise, or confirm constraints • Cluster dialogues by the balance between information-seeking and transaction-execution content • Cluster dialogues by whether the task succeeds directly, partially succeeds, fails, or is recovered through fallback • Cluster clips by the dominant emotional tone in speech • Cluster clips by the exact number of distinct speakers • Cluster clips by the real-world background-noise scene	• Cluster dialogues by how much earlier information must be remembered and reused later • Cluster dialogues by who drives the interaction and how strongly negotiation shapes the exchange • Cluster dialogues by whether the interaction stays within a narrow goal or couples multiple domains or subgoals • Cluster clips by the dominant speaker-gender pattern

A.7 TEXT-TO-SPEECH SYSTEM

We used Qwen3-TTS CustomVoice (Hu et al., 2026) to generate audio from the transcripts. During the synthesis, the model accepts, per utterance, the input text, a fixed language argument ("English" throughout the evaluation pipelines), target speakers, and a natural-language instruction that conditions the prosodic style.

Speaker Voices and Multi-Speaker Rendering Speaker identity is drawn from a fixed inventory of nine CustomVoice speakers: Vivian, Serena, Uncle_Fu, Dylan, Eric, Ryan, Aiden, Ono_Anna, and Sohee. During generation, each item is assigned a speaker count in $\{1, 2, 3, 4\}$, and a corresponding set of distinct speaker identities sampled from this pool, up to the size of the inventory.

Single-speaker items use the label "narrative", whereas multi-speaker items rotate through speakerA, speakerB, speakerC, and speakerD in strict cyclic order enforced by the generation prompt.

At synthesis time these abstract labels are bound to concrete Qwen speakers. For monologues, the full narrative text is rendered in a single call using the first assigned speaker identity. For conversations, the unique speaker labels are extracted in order of first appearance and mapped positionally onto the assigned speaker identities. Each dialogue segment is then synthesized independently with its bound speaker, and the resulting per-segment waveforms are concatenated with a fixed 0.3-second silence inserted between turns to demarcate speaker changes and impart conversational pacing.

Emotion Synthesis Emotional coloring is realized through instruction conditioning. Each item is assigned one of five categorical emotions: happiness, sadness, surprise, anger, or neutral. At synthesis time, the assigned emotion is mapped to a short natural-language directive that is passed as the instruct argument to the synthesizer (for example, "Speak in a happy, warm tone." for happiness).

Background-Noise Injection Following clean synthesis, each waveform is optionally degraded with an environmental background track to emulate realistic recording conditions. The noise taxonomy comprises four categories: clean (no degradation), indoor ambience, crowd noise, and traffic noise. For non-clean items, a noise file is drawn at random from the category-specific pool indexed over the configured noise directories. The chosen noise is converted to mono, resampled to the speech sample rate, and either randomly cropped or tiled with a random offset to match the speech length.

Mixing is performed at a controlled signal-to-noise ratio (SNR). The noise segment is first RMS-normalized, and its gain is then set so that the mixture attains the target SNR, following the relation

$$desired\_noise\_rms = \frac{speech\_rms}{10^{\frac{snr\_db}{20}}}. \tag{9}$$

To prevent clipping, the summed signal is peak-limited: if the maximum absolute amplitude exceeds 0.99, the mixture is rescaled by 0.99 / peak. If a per-item snr_db is absent, an SNR is drawn uniformly from a configurable range (default 10–20 dB).

Sampling and Stratified Assignment Rules The assignment of speaker count, emotion, and background-noise category is governed by a stratified, approximately uniform sampling scheme designed to mitigate joint skew across task types and taxonomy categories. Only items marked "accepted" with non-empty compressed text are eligible. Items are partitioned into strata defined by a stratum key.

Within each stratum of size m, a balanced routine allocates the m positions as evenly as possible over the candidate values of each attribute independently: it computes base, $rem = \textbf{divmod}(m, k)$ for k categories, assigns base occurrences to every category plus one extra to the first rem categories, and then shuffles the resulting multiset. The three per-attribute multisets—speaker counts, emotions, and noise categories—are zipped into triples, which are themselves shuffled to decorrelate the marginal assignments. Consequently, each attribute is close to uniformly distributed within every taxonomy stratum, rather than merely globally. An SNR value is drawn ($\textbf{uniform}(10.0, 20.0)$, rounded to two decimals) whenever the assigned noise category is not clean, and is left null otherwise.

B Benchmark Split Construction

B.1 Held-in and Held-out Perspectives

We partition perspectives rather than only individual audio recordings. Held-in perspectives are available during training, including their natural-language instructions and category taxonomies. Held-out perspectives are never used during training and are reserved for evaluating whether a model can adapt to novel clustering criteria. This design prevents evaluation from only measuring memorization of familiar label structures.

B.2 Training Instance Construction

For each held-in perspective, we split audio recordings within every category into a training pool and an evaluation pool. Training instances are generated by first sampling a subset of categories and then sampling multiple audio recordings from each selected category. This produces clustering instances with varying category combinations and cluster sizes. The model is not given the number of clusters; it must infer the number of clusters from the input audio and the natural-language perspective.

B.3 Evaluation Levels

The evaluation set is divided into three levels.

$\mathbf{L}_0$: seen perspectives and seen audio. $\mathbf{L}_0$ is constructed by re-sampling category and audio combinations from the training pool of held-in perspectives, while ensuring that the resulting clustering instances do not appear during training. This level evaluates recombination robustness: the model sees the same perspectives and audio recordings during training, but must solve new clustering combinations.

$\mathbf{L}_1$: seen perspectives and unseen audio. $\mathbf{L}_1$ is constructed from the evaluation pool of held-in perspectives. The clustering perspectives and their taxonomies are familiar, but none of the audio recordings appear in the training instances. This level evaluates whether the model can generalize to new recordings under familiar clustering criteria.

$\mathbf{L}_2$: unseen perspectives and unseen audio. $\mathbf{L}_2$ is drawn exclusively from held-out perspectives. The model has not observed the perspective instructions, category taxonomies, or audio recordings during training. This level evaluates cross-perspective generalization, namely whether the model can adapt to new natural-language clustering criteria over unseen audio collections.

B.4 Episode Construction and Benchmark Statistics

Episode formulation Each benchmark instance is formulated as an episode consisting of a natural-language clustering perspective p and n indexed audio clips, where $4 \leq n \leq 10$. The model input is represented as

$$(p,\, \{|i|\langle \text{audio}_i \rangle\}_{i=1}^{n}) \,,$$

and the target output is a partition $\mathcal{C} = \{C_1, \ldots, C_K\}$ over the n clips. The semantic category names used to construct the partition are hidden from the model. Moreover, the gold number of clusters K is not provided. The model must therefore jointly infer the partition cardinality and assign every input clip to exactly one cluster. This formulation evaluates both perspective-conditioned semantic understanding and variable-cardinality clustering.

Controlled episode sampling We construct episodes using a two-stage sampling procedure. First, category-specific audio pools are created for each clustering perspective. For held-in perspectives, the recordings are divided into training and evaluation pools, whereas held-out perspectives are reserved exclusively for evaluating $\mathbf{L}_2$ generalization. This separation ensures that generalization to unseen perspectives is evaluated independently from the construction of the SFT training episodes.

Second, episodes are generated through quota-aware perspective selection and adaptive balancing over the feasible values of K. Trivial single-cluster episodes are explicitly down-weighted, while categories with lower sampling coverage are assigned higher priority. After sampling the constituent categories, the total number of clips is drawn conditionally on K: episodes with smaller K preferentially contain 4–6 clips, whereas episodes with larger K may contain up to 10 clips. Candidates with invalid clip counts or duplicate episode signatures are rejected and resampled. This procedure balances category coverage while retaining natural variation in both episode size and partition cardinality.

Episode-level statistics As summarized in Tab. 7, the SFT training set contains 1876 episodes constructed from 2008 unique audio recordings. Based on the mean episode size, these episodes

Table 7: Episode-level statistics of the SFT training set. Entries for the final three columns are reported as mean (95th percentile, maximum).

Source	Episodes	Unique audios	Clips / episode	Gold K	Duration (s)
All SFT train	1,876	2,008	5.64 (7, 10)	3.37 (6, 9)	373.5 (524.6, 539.9)
ECHR	539	501	5.51 (7, 7)	3.37 (6, 7)	452.8 (531.4, 539.9)
S&P 500	571	455	5.69 (7, 9)	3.45 (6, 9)	411.4 (525.2, 539.2)
Banking77	523	618	5.88 (8, 10)	3.66 (7, 9)	306.3 (431.1, 538.1)
MultiWOZ	243	434	5.33 (7, 7)	2.56 (4, 5)	253.0 (372.3, 445.3)

comprise approximately 10.6K clip occurrences, corresponding to an average reuse factor of approximately $5.3\times$. Such recombination allows the same recording to participate in different episode configurations and prevents the training set from reducing to a fixed collection of partitions.

The training episodes contain 5.64 clips on average, with a 95th percentile of 7 clips, while the corresponding gold partitions contain 3.37 clusters on average and up to 9 clusters. The ratio between the aggregate mean episode size and mean K is approximately 1.67 clips per cluster, indicating that the benchmark predominantly evaluates compact, fine-grained partitions rather than a small number of large clusters. The average episode duration is 373.5 seconds (6.2 minutes), and the maximum duration is approximately 9 minutes. Thus, the task additionally requires reasoning over relatively long multi-audio contexts.

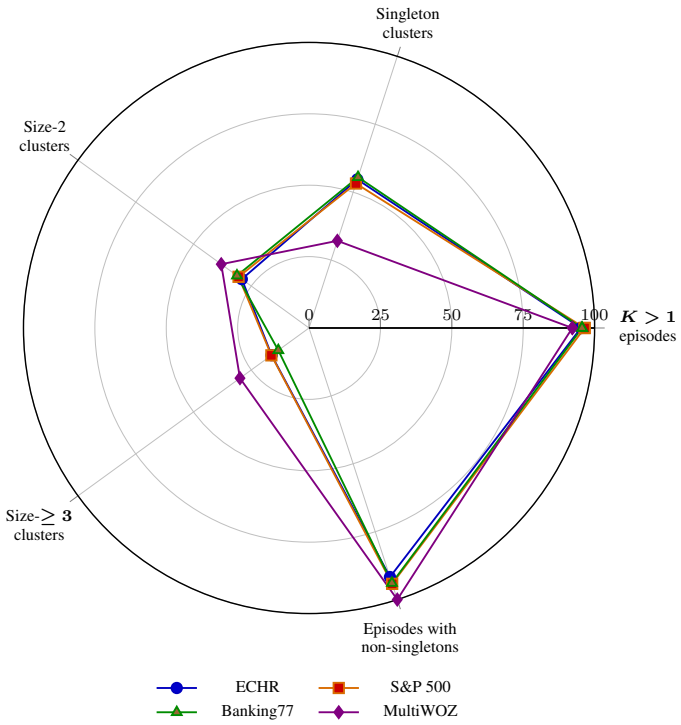


Figure 6: Source-specific episode and cluster profiles in the SFT training set. All axes report percentages. The first axis represents the percentage of multi-cluster episodes, i.e., $100\% - \Pr(K = 1)$. The three cluster-size axes form a compositional distribution and therefore sum to 100% for each source.

The individual sources exhibit complementary forms of complexity. Banking77 produces the largest episodes and the highest average partition cardinality, with 5.88 clips and 3.66 clusters per episode. ECHR instead contributes the longest temporal contexts, averaging 452.8 seconds despite containing fewer clips than Banking77. S&P 500 lies between these two regimes. MultiWOZ produces shorter episodes with a substantially smaller mean K of 2.56, yielding denser clusters with approximately 2.08 clips per cluster. These differences prevent the training distribution from being dominated by a single notion of episode difficulty.

Cluster-size distribution Across the complete training set, the 1876 episodes contain 6323 gold clusters. Singleton clusters account for 52.3% of these clusters, whereas clusters of size two and size three or larger account for 31.1% and 16.7%, respectively. The prevalence of singleton clusters is expected because a typical episode distributes only five to seven clips across multiple semantic categories. Importantly, the cluster-level singleton rate should not be interpreted as evidence of

episode-level degeneracy. Only 4.7% of the episodes have $K = 1$, while 94.1% contain at least one non-singleton cluster. Consequently, the vast majority of episodes require the model to identify at least one positive within-cluster relation while simultaneously separating clips belonging to different categories.

Fig. 6 visualizes the percentage-based cluster-structure statistics across the four sources. MultiWOZ exhibits a notably denser partition structure: only 32.1% of its gold clusters are singletons, compared with 53.2–55.5% for the other three sources, and 29.9% contain at least three clips. In contrast, Banking77 has the largest average K but the smallest proportion of size-three-or-larger clusters, indicating that its higher cardinality primarily arises from a larger number of fine-grained categories rather than larger within-category groups.

C Additional Method Details

This appendix provides additional details for the training procedure of SpeechLens-R1. Appendix C.1 describes reasoning distillation, and Appendix C.2 describes preference-pair construction and DPO training.

C.1 Reasoning Distillation Details

Gold Partition Construction For each training instance, benchmark annotations provide a category label for every audio segment under the corresponding clustering perspective. We convert these labels into a gold partition over item indices: two segments are assigned to the same cluster if and only if they share the same category label. This representation removes dependence on category names and cluster order, allowing supervision and evaluation to operate directly on partition structure.

Teacher Trace Generation Given an input audio set $\mathcal{D}^{(i)}$, a clustering perspective $p^{(i)}$, and the gold partition $\mathcal{C}^{(i)*}$, we ask a teacher model to generate a concise reasoning trace that naturally leads to the gold partition. The teacher is instructed to identify the grouping principle, compare segments across the set, resolve confusable cases, infer the number of clusters, and end with a final clustering block that assigns every item exactly once.

This procedure can be viewed as gold-constrained rationale synthesis. The teacher is not asked to discover the answer from scratch; instead, it is conditioned on the correct partition and asked to synthesize a plausible reasoning path consistent with that partition. This improves supervision quality by separating answer discovery from explanation generation.

Linguistic and Paralinguistic Traces We construct reasoning traces from two complementary views. For linguistic perspectives, we transcribe each audio segment and provide the teacher with indexed text inputs. The teacher then generates a semantically grounded rationale based on the lexical content. For paralinguistic perspectives, we provide indexed audio clips directly to an audio-capable teacher model. The teacher is encouraged to ground its reasoning in audible evidence. When both views are available for an instance, we retain both traces as complementary supervision.

Linguistic reasoning generation prompt

```
You are a clustering assistant for audio-style clustering.
Given a clustering goal and a list of indexed items (|1|. ..., |2|.
    ..., |3|. ...),
produce a reasoning process that clusters them correctly.
The clustering uses 1-based indexing for all items.
Think process of the task:
First, go through the items and identify the most plausible grouping
    principle under
the clustering goal. Form an initial view of how many clusters there
    may be and what
distinguishes them.
```

```
Then compare items across the set, focusing on which items belong
    together, which
ones are easily confusable, and what distinctions matter most.
If needed, include one brief moment of uncertainty, revision, or self-
    check, but
only when it naturally helps resolve a genuine grouping decision.
Finally, state the final grouping so that every item belongs to
    exactly one cluster.
Clustering Goal:
{INSTRUCTION}
Items:
{ENUMERATED_TEXT}
The correct final clustering results are:
<answer>
{CORRECT_ANSWER}
</answer>
Now produce a high-quality think process that naturally leads to the
    correct clustering.
IMPORTANT!!!
1. Write strictly from a direct listening perspective.
2. Do NOT mention text, transcripts, reading, or simulated listening.
3. Do NOT use phrasing that implies prior knowledge of the final
    answer (for example: "looking at the correct answer", "we must
    match", "the real clusters are", "since the answer is given").
4. Do NOT perform explicit sanity checks against the provided answer
    block.
5. The correct clustering should emerge after comparison and reasoning
    , not be stated immediately.
6. The reasoning should not start from fixed category names or
    predefined labels.
7. Cross-item comparison is required.
8. Include uncertainty, backtracking, or self-verification only when
    it naturally arises; do not force it.
9. Avoid rigid template repetition or purely item-by-item labeling.
10. Do not describe every item individually in sequence unless
    absolutely necessary; prioritize group-level comparison.
11. Keep the reasoning very concise, with high information density (
    useful grouping detail, not filler); the full reasoning should be
    between 120 and 200 words.
12. Avoid repeated paraphrases, repeated pairwise comparisons, or long
     explanations that do not add new grouping insight.
13. Each paragraph should contribute either:
    a grouping hypothesis,
    a comparison that separates or merges items,
    or a final assignment decision.
14. End with a clearly recoverable final grouping that covers all
    items exactly once.
Output format:
- Start with: Think process:
- End with a short final clustering block in this style:
Final clustering:
- Cluster 1: [ ... ]
- Cluster 2: [ ... ]
...
- Do not include anything outside the reasoning process.
```

Paralinguistic reasoning generation prompt

```
You are a clustering assistant for audio clustering.
Given a clustering goal and indexed audio clips (|1|, |2|, |3|, ...),
    listen to
```

```
the clips and produce a reasoning process that clusters them correctly
    .
The clustering uses 1-based indexing for all items.
Think process of the task:
First, listen through the clips and identify the most plausible
    grouping principle under the clustering goal. Form an initial view
     of how many clusters there may be and what audible cues
    distinguish them.
Then compare clips across the set, focusing on which clips belong
    together, which ones are easily confusable, and what distinctions
    matter most.
If needed, include one brief moment of uncertainty, revision, or self-
    check, but only when it naturally helps resolve a genuine grouping
     decision.
Finally, state the final grouping so that every clip belongs to
    exactly one cluster.
Clustering Goal:
{INSTRUCTION}
There are {N_ITEMS} clips in total, indexed from 1 to {N_ITEMS}.
The correct final clustering results are:
<answer>
{CORRECT_ANSWER}
</answer>
Now produce a high-quality think process that naturally leads to the
    correct clustering.
IMPORTANT!!!
1. Ground the reasoning in audible evidence such as voice
    characteristics, prosody, timing, overlap, background sounds,
    acoustic scene cues, or other perceptual clues relevant to the
    task.
2. Do NOT rely on semantic content unless the task itself is
    explicitly semantic.
3. Do NOT use phrasing that implies prior knowledge of the final
    answer.
4. Do NOT perform explicit sanity checks against the provided answer
    block.
5. The correct clustering should emerge after comparison and reasoning
    , not be stated immediately.
6. The reasoning should not start from fixed category names or
    predefined labels.
7. Cross-clip comparison is required.
8. Include uncertainty, backtracking, or self-verification only when
    it naturally arises; do not force it.
9. Avoid rigid template repetition or purely clip-by-clip labeling.
10. Do not describe every clip individually in sequence unless
    absolutely necessary; prioritize group-level comparison.
11. Keep the reasoning very concise, with high information density (
    useful grouping detail, not filler); the full reasoning should be
    between 120 and 200 words.
12. Avoid repeated paraphrases, repeated pairwise comparisons, or long
     explanations that do not add new grouping insight.
13. Each paragraph should contribute either:
    a grouping hypothesis,
    a comparison that separates or merges clips,
    or a final assignment decision.
14. End with a clearly recoverable final grouping that covers all
    clips exactly once.
Output format:
    - Start with: Think process:
    - End with a short final clustering block in this style:
Final clustering:
    - Cluster 1: [ ... ]
    - Cluster 2: [ ... ]
```

```
...
- Do not include anything outside the reasoning process.
```

Trace Constraints We impose several constraints on teacher-generated traces. First, the reasoning must be comparison-driven rather than a sequence of independent item labels. Second, it should explain why some segments belong together and why others should be separated. Third, it must not mention that the gold answer is provided or imply that the reasoning is merely matching a known solution. Fourth, it should be concise and information-dense, avoiding rigid templates or repeated paraphrases. Finally, paralinguistic traces must be grounded in audible evidence rather than generic semantic descriptions.

Verification and Filtering We apply a two-stage filtering procedure before adding a trace to the RD dataset. First, we parse the final clustering block and compare the implied partition with the gold partition, ignoring cluster order and cluster names. If rule-based parsing is inconclusive, we use an independent verifier model to determine whether the final grouping matches the gold partition. Second, we use a judge model to evaluate the reasoning quality, including completeness, logical coherence, comparison quality, decision quality, conciseness, and modality consistency. Only traces that pass both the partition-consistency check and the quality judgment are retained.

Partition checking prompt

```
You are a strict partition checker.
Your ONLY task is to decide whether the FINAL clustering assignment
    implied by the reasoning is exactly the same partition as the
    ground truth.
Rules:
1. Ignore narrative quality, fluency, confidence, or how convincing
    the explanation sounds.
2. Focus only on the final grouping the author commits to.
3. If the final grouping is ambiguous, incomplete, or not recoverable,
     output MISMATCH.
4. Cluster order does NOT matter.
5. Category names do NOT matter.
6. Only the partition over item indices matters.
Items are indexed 1..{n_items}, and each item must appear exactly once
    .
Ground truth (correct partition):
{correct_answer}
Reasoning chain to check:
{reasoning_chain}
Determine whether the final grouping implied by the reasoning matches
    the ground truth exactly as a partition over indices.
Output exactly:
VERDICT: MATCH or MISMATCH
REASON: one short sentence describing only the grouping comparison.
```

Reasoning quality evaluation prompt

```
You are an expert AI judge tasked with evaluating the quality and
    completeness of clustering reasoning chains.
Evaluate the reasoning chain using the following criteria:
1. Completeness:
   Does the reasoning provide a full path from initial observations to
     grouping decisions and a final assignment?
2. Logical Coherence:
   Does the reasoning proceed in a consistent and understandable
    sequence, without major contradictions or unexplained jumps?
```

```
3. Comparison Quality:
   Does the reasoning compare items across the set, rather than merely
     describing or labeling each item independently?
4. Decision Quality:
   Does the reasoning actually make grouping decisions, including
    separating confusable items or merging similar ones for clear
    reasons?
5. Exploration Quality:
   If uncertainty or ambiguity naturally arises, is it handled briefly
     and usefully? Reject fake or padded uncertainty.
6. Modality Consistency:
   - For linguistic reasoning outputs, the reasoning should still read
     like direct listening-based reasoning and should not mention
    reading or transcripts.
   - For paralinguistic reasoning outputs, the reasoning should be
    grounded in audible evidence rather than a generic semantic
    paraphrase.
7. Final Assignment Presence:
   Does the reasoning clearly imply a final clustering that assigns
    every indexed
   item exactly once?
8. Conciseness and Information Density:
   The reasoning should be concise and information-dense. Reject if it
     contains significant redundancy, repeated comparisons, repeated
    paraphrasing, or long passages that add little new grouping
    information. Do NOT reject solely because it is somewhat long if
    it remains efficient and informative.
Reject if any of the following is true:
    - The chain is too short, incomplete, or abruptly cut off
    - There is no real cross-item comparison
    - The reasoning is mostly template-like or mostly item-by-item
    labeling
    - It explicitly refers to reading, transcripts, or matching the
    provided answer
    - It forces artificial uncertainty or artificial self-correction
    - The final implied clustering is missing, incomplete, or does not
     cover all items exactly once
    - The chain is overly verbose relative to the amount of actual
    grouping insight
Reasoning chain to evaluate:
{reasoning_chain}
Response format:
ASSESSMENT: ACCEPT or REJECT
REASON: one concise explanation
```

Training Target Each retained RD target contains a reasoning trace followed by a final clustering answer. The model is trained with the autoregressive loss:

$$\mathcal{L}_{\mathrm{RD}}(\theta) = -\sum_{(x,y)\in\mathcal{D}_{\mathrm{RD}}} \sum_{t=1}^{|y|} \log p_\theta(y_t \mid x, y_{<t}). \tag{10}$$

Although the RD target includes reasoning, the final answer is always serialized in a canonical clustering format so that downstream evaluation can recover the predicted partition.

C.2 Preference Optimization Details

Preference-pair Construction For each training input x, we construct a preference pair (x, y^+, y^-). The chosen response y^+ is the canonical gold clustering answer derived from the benchmark annotation. The rejected response y^- is sampled from the RD model. A sampled response is eligible only if it satisfies three conditions: it is parsable as a clustering answer, it assigns

every item exactly once with no missing, duplicate, or unknown indices, and it produces a partition different from the gold partition.

This validity filter prevents DPO from being dominated by trivial formatting mistakes. Instead, the preference objective focuses on valid but incorrect clustering decisions.

Answer-token Margin To identify hard rejected answers, we compute the answer-token mean log-probability margin under the initial model π_0:

$$m(x, y^+, y^-) = \frac{\log \pi_0(y^+ \mid x)}{|y^+|} - \frac{\log \pi_0(y^- \mid x)}{|y^-|}. \tag{11}$$

A small or negative margin indicates that the initial model assigns comparable or higher likelihood to the rejected answer than to the gold answer. We therefore prioritize pairs with small margins, since they expose errors that the model is likely to make at inference time.

Clustering-quality Gap We also compute a clustering-quality gap

$$g = s(y^+) - s(y^-), \tag{12}$$

where $s(\cdot)$ is a clustering quality score used for candidate filtering. Candidates with extremely small gaps are removed because they may correspond to nearly equivalent partitions or ambiguous cases. The remaining candidates are valid but meaningfully worse than the gold clustering.

Structural Error Categories To balance the DPO data across different types of clustering errors, each valid rejected answer is assigned to one structural error category. Let K^+ and K^- denote the gold and predicted numbers of clusters. Let R_{same} be the recall of gold same-cluster pairs, and let R_{diff} be the recall of gold different-cluster pairs. We categorize rejected answers using the following deterministic rules:

$$e(y^-) = \begin{cases} \text{NEAR-MISS}, & 0.20 \le g \le 0.40 \ \wedge \ K^- = K^+, \\ \text{OVER-MERGE}, & K^- < K^+ \ \vee \ (R_{\text{same}} \ge 0.80 \wedge R_{\text{diff}} < 0.60), \\ \text{OVER-SPLIT}, & K^- > K^+ \ \vee \ (R_{\text{diff}} \ge 0.80 \wedge R_{\text{same}} < 0.60), \\ \text{K-WRONG}, & K^- \neq K^+, \\ \text{WRONG-ASSIGNMENT}, & \text{otherwise.} \end{cases} \tag{13}$$

NEAR-MISS cases have the correct number of clusters but imperfect assignments. OVER-MERGE cases collapse distinct gold clusters, while OVER-SPLIT cases fragment a gold cluster into multiple predicted clusters. K-WRONG captures remaining cluster-count errors, and WRONG-ASSIGNMENT captures valid partitions with incorrect item assignments.

Balanced Hard-pair Selection After assigning error categories, we select hard pairs with three goals. First, pairs should be difficult for the initial model, as indicated by a small answer-token margin. Second, selected pairs should cover different structural error types. Third, pairs should be balanced across clustering perspectives so that DPO does not overfit to a small number of frequent perspectives. We cap the number of rejected answers per prompt and use perspective- and error-balanced sampling to construct the final DPO dataset.

DPO Objective The policy model π_θ and reference model π_{ref} are initialized from the same RD checkpoint. The reference model is frozen during DPO. For each selected preference pair, we compute answer-token mean log-probabilities:

$$\Delta_\pi = \log \pi_\theta(y^+ \mid x) - \log \pi_\theta(y^- \mid x), \tag{14}$$

$$\Delta_{\text{ref}} = \log \pi_{\text{ref}}(y^+ \mid x) - \log \pi_{\text{ref}}(y^- \mid x). \tag{15}$$

The scaled DPO logit is

$$z = \beta L_{\text{ref}}(\Delta_\pi - \Delta_{\text{ref}}), \tag{16}$$

where β controls preference strength and L_{ref} is the average answer length in the DPO training set. The final loss is

$$\mathcal{L}_{\text{DPO}} = -\log \sigma(z). \tag{17}$$

All log-probabilities are computed only over the canonical answer block. This design aligns the model toward better clustering decisions while avoiding direct optimization over long free-form reasoning text.

D BASELINE IMPLEMENTATION DETAILS

We evaluate a set of transcript-embedding clustering baselines: audio-to-text conversion followed by embedding-based clustering. For each benchmark instance in the $\mathbf{L}_0$, $\mathbf{L}_1$, and $\mathbf{L}_2$ splits, the baseline first obtains a transcript for each audio segment. For each corpus, the default entrypoints use `OpenAI Whisper` to transcribe the audio on the fly; the latter transcripts are normalized by removing speaker-role prefixes and collapsing whitespace. Given the natural-language clustering perspective, each transcript is then converted into a task-conditioned text input using model-specific templates. We run four embedding backbones defined by the official baseline configuration: `hkunlp/instructor-large`, `BrandonZYW/llama-2-7b-InBedder`, `Qwen/Qwen3-Embedding-0.6B`, and `sentence-transformers/all-MiniLM-L6-v2`. For Instructor, the perspective is provided as an embedding instruction; for Qwen3-Embedding and all-MiniLM, the perspective and transcript are concatenated into a single task-aware query; for InBedder, the transcript and instruction are formatted as an input–instruction prompt and the final hidden representation after short generation is used as the embedding. All embeddings are L2-normalized by default. We then apply two standard clustering algorithms, K-Means and GMM, to the embeddings. Unless otherwise specified, the number of clusters is set to the number of gold clusters, giving these baselines an oracle cluster-count setting. K-Means is run with random seed 43 and automatic initialization when supported by the installed scikit-learn version; GMM is run with diagonal covariance, regularization coefficient 10^{-6}, and the same random seed. The resulting cluster assignments are compared against the gold labels using ARI and V-measure. Failed embedding or clustering runs are treated as invalid predictions and receive zero score in the aggregate evaluation.

For transcription-based LLM clustering, we follow the same ASR strategy as aforementioned, and use the following prompt to generate a clustering result by taking a collection of transcribed scripts.

Prompt for transcribed scripts reasoning

```
You are a clustering assistant to do audio clustering.
Given a clustering goal and a list of indexed audio:
First, listen to all audio recordings and think how can they be
    clustered based on the goal, determine the total number of
    clusters.
Then think about how to assign all audio recordings into these
    clusters.
Check the answer format before giving the final answer: every item
    must be assigned to exactly one cluster, and no item should appear
     in multiple clusters or be missing.
The reasoning and answer must be enclosed within <think> </think> and
    <answer> </answer> tags, respectively.
Final Output Format should be:
<think> assistant's reasoning process here </think>
<answer>
Total clusters: [N].
cluster1: [item_numbers separated by commas].
cluster2: [item_numbers separated by commas].
...
</answer>
Now, please follow the format for the following clustering task:
Goal: {CLUSTERING_PERSPECTIVE}
The following are transcribed texts from indexed audio segments.
    Please cluster these segments based on the transcribed content and
     task goal.
Transcribed Audio Text: {TEXTS}
```

For LALM baselines, we use a direct audio prompting protocol that requires the model to perform clustering from native audio inputs rather than from transcripts. The prompt explicitly instructs the model to listen to each audio segment in order, where the i-th audio segment corresponds to item i under 1-based indexing. This design ensures that the model conditions its prediction on the original speech signal and can exploit both linguistic content and paralinguistic cues when they are relevant to the specified perspective. To improve output consistency and facilitate automatic parsing, the prompt further requires the model to verify that every item is assigned to exactly one cluster, with no duplicated or missing items. The model is asked to produce its response in a structured format with separate reasoning and answer fields enclosed by `<think>` and `<answer>` tags. The final answer must report the inferred number of clusters and list the item indices assigned to each cluster. Unlike embedding-based baselines that assume an oracle number of clusters, this prompting protocol requires the LALM to jointly infer both the cluster count and the cluster assignments from the provided audio collection.

Prompt for native LALMs

```
You are a clustering assistant for native audio inputs.
You will receive a clustering goal and multiple audio segments
    attached as audio (not as transcripts). Listen to each segment in
    order; segment index i corresponds to item i (1-based numbering in
     the final answer).
Do not claim that you only have text--use the provided audio. If audio
     parts are present in the user message, you must base clustering
    on listening.
Check the answer format before giving the final answer:
Every item must be assigned to exactly one cluster,
and no item should appear in multiple clusters or be missing.
The reasoning and answer must be enclosed within <think> </think> and
    <answer> </answer> tags, respectively.
Final Output Format should be:
<think> assistant's reasoning process here </think>
<answer>
Total clusters: [N].
cluster1: [item_numbers separated by commas].
cluster2: [item_numbers separated by commas].
...
</answer>
Now, please follow the format for the following clustering task:
Goal: {CLUSTERING_PERSPECTIVE}
Audio: {SPEECH_CLIPS}
```

E Implementation Config Details

E.1 Reasoning Distillation

For the reasoning distillation, we train on the distilled trainset, which contains 1876 examples. The model is initialized from the 7B Audio Flamingo 3 and fine-tuned with LoRA for parameter-efficient adaptation. Training is conducted in `bfloat16`, with gradient checkpointing enabled to reduce memory usage, together with FlashAttention and DeepSpeed-based memory optimization. The micro-batch size is set to 1, and the run uses a cosine learning-rate schedule with a peak learning rate of 5×10^{-5}. We train for 6 epochs. This experiment is designed to preserve the full reasoning supervision signal during distillation and to improve the model's reasoning and response quality on audio-language instruction-following tasks.

E.2 Direct Preference Optimization

We perform scaled-mean DPO optimization on a screened on-policy preference set of 1,075 training pairs constructed from the held-in perspectives. Training is run with the official `trl.DPOTrainer` on 1 node with 4 GPUs, with per-device batch size 1 and gradient accumulation 3, giving an effective

global batch size of 12. The policy and reference are both initialized from the reasoning distilled model, and optimization uses scaled-mean DPO with $\beta = 0.5$ and a fixed length scale $L_{\text{ref}} = 34.0$. We train for 500 steps (with an epoch cap of 20.0), using a constant-with-warmup schedule with 5 warmup steps and learning rate 5×10^{-6}. Training uses `bfloat16`, gradient checkpointing, max grad norm 1.0, and a LoRA adapter applied to attention modules in the last 16 layers with rank 32, alpha 32, and dropout 0. The objective is computed on answer-only tokens, excluding prompt, padding, and non-answer tokens.

F Evaluation Metrics

Evaluating clustering quality is a fundamental yet non-trivial problem, as cluster labels are inherently unordered and lack a direct correspondence with ground-truth class labels. Consequently, effective evaluation metrics must be invariant to label permutations and capable of capturing different aspects of clustering structure. Broadly, external clustering metrics can be categorized into *information-theoretic* measures and *pair-counting* measures. In this work, we adopt two widely used and complementary metrics: *V-measure* (Rosenberg & Hirschberg, 2007), which is grounded in information theory, and *Adjusted Rand Index (ARI)* (Hubert & Arabie, 1985), which is based on pairwise assignment consistency.

V-measure is an entropy-based metric designed to quantify the agreement between predicted clusters and ground-truth classes through two desirable properties: *homogeneity* and *completeness*. Homogeneity measures whether each cluster contains only samples from a single class, thus penalizing cluster impurity. Completeness, on the other hand, evaluates whether all samples belonging to a given class are assigned to the same cluster, penalizing class fragmentation across multiple clusters.

These properties are formalized using conditional entropy. Let C denote the set of ground-truth labels and K denote the set of predicted clusters. The homogeneity score is defined as:

$$h = 1 - \frac{H(C \mid K)}{H(C)}, \tag{18}$$

which becomes 1 when each cluster contains only one class (i.e., zero conditional entropy). Similarly, completeness is defined as:

$$c = 1 - \frac{H(K \mid C)}{H(K)}, \tag{19}$$

which reaches 1 when all members of a class are assigned to a single cluster.

To balance these two criteria, V-measure computes their harmonic mean:

$$V = (1 + \beta) \cdot \frac{h \cdot c}{\beta \cdot h + c}, \tag{20}$$

where β controls the relative importance of completeness over homogeneity (typically $\beta = 1$). The harmonic mean ensures that a high V-measure score is achieved only when both homogeneity and completeness are simultaneously high. The resulting score lies in $[0, 1]$, with higher values indicating better alignment between clustering and ground truth.

Adjusted Rand Index evaluates clustering quality by considering all pairs of samples and measuring how consistently they are assigned in both the predicted clustering and the ground-truth labeling. Specifically, a pair of samples can either be assigned to the same cluster or to different clusters. ARI counts agreements and disagreements between the two partitions over all $\binom{n}{2}$ possible pairs.

The original Rand Index (RI) computes the fraction of agreeing pairs; however, it does not account for agreements that may occur by chance, especially when the number of clusters is large or unbalanced. ARI addresses this limitation by introducing a chance-adjusted normalization. Let n_{ij} denote the number of samples assigned to ground-truth class i and predicted cluster j, and define $a_i = \sum_j n_{ij}$ and $b_j = \sum_i n_{ij}$. The ARI is computed as:

$$\text{ARI} = \frac{\sum_{ij} \binom{n_{ij}}{2} - \frac{\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}}{\binom{n}{2}}}{\frac{1}{2}\left(\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right) - \frac{\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}}{\binom{n}{2}}}. \tag{21}$$

This normalization ensures that the expected ARI of random clusterings is approximately 0, providing a meaningful baseline. The ARI ranges from -1 to 1, where 1 indicates perfect agreement, 0 corresponds to random assignments, and negative values indicate worse-than-random clustering.

Complementary Perspectives V-measure and ARI capture complementary aspects of clustering quality. V-measure emphasizes global information consistency and is particularly sensitive to over-segmentation (low completeness) and mixed clusters (low homogeneity). In contrast, ARI focuses on pairwise consistency and is sensitive to both cluster size distribution and the relative placement of individual samples. Using both metrics provides a more comprehensive evaluation, especially in complex settings such as audio multi-perspective clustering, where both semantic purity and structural consistency are crucial.

G More Analysis

Comparison with ASR-based Pipelines Tables 2 and 3 show a clear gap between SpeechLens-R1 and ASR-based pipelines. Among embedding-based methods, the strongest overall result is achieved by Instructor with K-Means, reaching 17.01 ARI and 52.66 V-measure. Replacing the embedding-and-clustering stage with an LLM improves performance substantially: Whisper+GPT-4o obtains 28.38 overall ARI and 60.42 overall V-measure. However, SpeechLens-R1 still outperforms this ASR+LLM baseline by +16.39 ARI points and +13.01 V-measure points. This suggests that simply transcribing speech into text is insufficient for audio multi-perspective clustering. By directly operating on audio inputs, SpeechLens-R1 can exploit acoustic and paralinguistic cues that may be weakened or discarded during ASR.

Comparison with Native LALMs SpeechLens-R1 also substantially outperforms off-the-shelf native audio-language models. Among these baselines, GPT-audio-1.5 is the strongest overall system, achieving 31.78 ARI and 61.81 V-measure. In contrast, SpeechLens-R1 reaches 44.77 ARI and 73.43 V-measure, improving over GPT-audio-1.5 by +12.99 ARI points and +11.62 V-measure points. SpeechLens-R1 obtains the best ARI across all 12 corpus-level evaluation settings and the best V-measure on 11 out of 12 settings. These results indicate that general-purpose audio-language models do not automatically acquire reliable set-partitioning behavior, and that task-specific post-training is important for perspective-conditioned clustering.

Failure Mode of Audio Flamingo 3 Audio Flamingo 3 obtains very low scores in our evaluation, which is mainly due to its inability to reliably follow the required clustering-output protocol. In many cases, the model produces free-form responses that cannot be parsed into a valid partition, such as missing items, duplicated assignments, or outputs without an explicit cluster structure. To avoid underestimating the model solely due to rigid formatting constraints, we additionally applied an LLM-based extractor to recover clustering assignments from its responses when possible. However, the recovered partitions still led to very low clustering scores, suggesting that the failure is not merely a parsing artifact. Rather, off-the-shelf Audio Flamingo 3 lacks the task-specific behavior needed for perspective-conditioned set partitioning: it must interpret the clustering perspective, compare multiple audio segments jointly, infer the number of clusters, and produce a complete valid partition. This observation further motivates our reasoning distillation and preference optimization stages, which explicitly train SpeechLens-R1 to generate valid and clustering-aligned outputs.

Corpus-level Observations. The gains of SpeechLens-R1 are consistent across all four corpora, but the nature of the improvement differs by domain. On ECHR and S&P 500, SpeechLens-R1 substantially improves both ARI and V-measure, suggesting stronger ability to organize long-form legal and financial speech under different clustering perspectives. On Banking77, SpeechLens-R1 achieves especially strong ARI improvements, indicating that the model can distinguish fine-grained intent-oriented spoken utterances while also using audio-side information when required. On MultiWOZ, SpeechLens-R1 also outperforms all baselines in ARI and achieves the best V-measure across all three evaluation levels. This suggests that the proposed training pipeline improves not only paralinguistic perception but also dialogue-domain clustering, where the model must reason over interaction structure and pragmatic intent.

We also train the model with using Qwen2.5-omni as the base model. The evaluation results are included in Tab. 8.

Table 8: Performance of using Qwen2.5-omni as the base model. BG, Emo., Spk., Gen., and Reason denote background noise, emotion, speaker count, gender, and the average over linguistic reasoning perspectives.

Metric	Training	ECHR					S&P 500					Banking77					MultiWOZ					Overall				
		BG	Emo.	Spk.	Gen.	Reason	BG	Emo.	Spk.	Gen.	Reason	BG	Emo.	Spk.	Gen.	Reason	BG	Emo.	Spk.	Gen.	Reason	BG	Emo.	Spk.	Gen.	Reason
ARI	RD SFT	0.2947	0.2809	0.2682	0.2806	0.3872	0.3754	0.3096	0.3212	0.3539	0.3968	0.3380	0.2616	0.3025	0.3803	0.4749	0.3127	0.3664	0.0000	0.3823	0.4067	0.3302	0.3046	0.2230	0.3493	0.4164
	RD+DPO	0.3163	0.2882	0.2727	0.2993	0.3821	0.3965	0.3169	0.3293	0.3554	0.4014	0.3818	0.2437	0.3448	0.3786	0.4507	0.3407	0.3557	0.0000	0.3724	0.4010	0.3588	0.3012	0.2367	0.3514	0.4088
V-measure	RD SFT	0.6135	0.6772	0.6212	0.5756	0.8025	0.6460	0.6879	0.6294	0.5866	0.7717	0.5475	0.6769	0.4725	0.5801	0.8592	0.5115	0.7023	0.0000	0.5134	0.6240	0.5796	0.6861	0.4308	0.5639	0.7643
	RD+DPO	0.6221	0.6827	0.6223	0.5864	0.8000	0.6595	0.6920	0.6329	0.5837	0.7689	0.5834	0.6660	0.4921	0.5779	0.8485	0.5384	0.6985	0.0000	0.5141	0.6238	0.6008	0.6848	0.4368	0.5655	0.7603

G.1 Case Study

We present a case study from the ECHR domain to illustrate perspective-conditioned audio clustering. The model is asked to cluster legal cases by the main relationship between the applicant and the responsible actor(s), especially distinguishing direct State conduct from protection, enforcement, or regulatory failures. Although transcripts are shown for readability, SpeechLens-R1 receives the original audio recordings during inference.

As shown in Fig. 7, SpeechLens-R1 separates the seven cases into four relation-based groups: administrative or regulatory disputes, direct coercive State conduct, a private dispute involving judicial protection, and detention-condition complaints. This grouping reflects the intended perspective rather than superficial topical similarity. For example, police abuse and military police shooting are grouped together as direct State force, while customs seizure, land-transfer approval, and contaminated blood-product compensation are grouped as administrative or regulatory responsibility.

This case study suggests that SpeechLens-R1 can infer abstract relational structures from native audio inputs, supporting flexible clustering under user-specified perspectives.

`SPEECH_1`: [**Serena**: *In May 2005, the applicant was stopped on the street by police officers and taken into custody at the Sovetskiy district police station in Orsk. He tried to escape but was assaulted by the officers, who kicked him in the stomach.* **Vivian**: *That assault caused blunt abdominal trauma, including a ruptured intestine and serious health damage. He lost consciousness and was placed in a cell, where the police ignored his requests for medical help.* **Serena**: *The next day, he was finally hospitalized with internal bleeding and spent six weeks receiving care. Forensic reports confirmed his injuries were caused by blunt force trauma, ruling out any accidental causes.* **Vivian**: *Following this, the applicant brought a civil claim against the State authorities for ill-treatment and an ineffective investigation. The Leninskiy District Court partially granted compensation, finding the injuries occurred in police custody with no alternative explanation provided.* **Serena**: *That judgment was upheld on appeal, highlighting issues like direct police encounters, use of force by agents, physical abuse, and deprivation of liberty.* **Vivian**: *Yes, and the case also emphasized the failure of authorities to properly investigate the ill-treatment, which led to the civil litigation and compensation awarded to the applicant.*]

`SPEECH_2`: [**Sohee**: *The applicants, who are currently deprived of their liberty, have filed complaints against the detaining authority.* **Serena**: *Yes, their main concern is the inadequate and inhuman conditions they face while in detention.* **Sohee**: *Exactly. The details about each applicant and their specific applications are outlined in the appended table.* **Serena**: *Their grievances focus especially on the custody conditions and the care they receive, which they describe as poor.* **Sohee**: *They also highlight the ill-treatment they have suffered, which stems from the harsh detention environment.* **Serena**: *So, overall, their complaints paint a troubling picture of neglect and mistreatment within the detention facilities.*]

`SPEECH_3`: [**Sohee**: *The applicant, a U.S. citizen living in Munich, wanted to build a holiday home in Hopfgarten, Austria. She began negotiating to buy land there back in 1971.* **Dylan**: *Her purchase contract needed approval under the Tyrolean Real Property Transactions Act. At first, the local authority gave the green light for the sale.* **Ono_Anna**: *But then the Real Property Transactions Officer challenged this decision by appealing to the Regional Authority. The concern was that with 110 foreign landowners already in Hopfgarten, this sale might lead to foreign domination.* **Vivian**: *The Regional Authority agreed and refused to approve the transfer. They argued that the purchase could harm social and economic interests and noted the land was intended for a holiday home, not farming.* **Sohee**: *The applicant then took her case to the Constitutional Court, claiming her property rights and right to a fair court were violated. She also argued the Regional Authority wasn't*

You are a clustering assistant for native audio inputs. You will receive a clustering goal and multiple audio segments attached as audio (not as transcripts). Listen to each segment in order; segment index i corresponds to item i (1-based numbering in the final answer). Do not claim that you only have text–use the provided audio. If audio parts are present in the user message, you must base clustering on listening.

Check the answer format before giving the final answer: Every item must be assigned to exactly one cluster, and no item should appear in multiple clusters or be missing. The reasoning and answer must be enclosed within <think> </think> and <answer> </answer> tags, respectively.

Final Output Format should be: <think> assistant's reasoning process here </think> <answer>
Total clusters: [N].

Cluster1: [item_numbers separated by commas].

Cluster2: [item_numbers separated by commas].

... </answer>

Now, follow the format for the following clustering task: Goal: Cluster these legal cases by the main relationship between applicant and responsible actor(s), distinguishing direct State conduct from protection/enforcement failures. Audio: SPEECH_1, SPEECH_2, SPEECH_3, SPEECH_4, SPEECH_5, SPEECH_6, SPEECH_7

<think> I'm mainly hearing two big relationship types: cases where the harm comes straight from State agents' actions, versus cases where the State is involved more as a regulator/court system (or as custodian) rather than as the immediate perpetrator. Items 1 and 5 clearly sit together: both are on-the-spot encounters with police/military police using force. In 1 it's beating in custody plus delayed medical help; in 5 it's live fire during an arrest attempt leading to deaths. These feel like direct operational conduct by security forces, not just a regulatory decision. Item 2 is also about State control, but it's not a single arrest/force incident—it's ongoing detention conditions complained of against the detaining authority. That separates it from 1/5 into a custody-conditions bucket. Then there's a set where the applicant is essentially fighting the State as administrator/regulator: 7 (customs seizure/damage and compensation litigation), 3 (land transfer approval blocked by property-transaction authorities), 4 (infected blood products and administrative liability/compensation). I hesitated about 7 because a private company appears, but the core dispute is still with customs and State responsibility. Finally, 6 is different: a private insurance dispute where the State's role is providing a court process (with delay issues), not being the opposing actor. </think>
<answer>
Total clusters: [4].

Cluster 1: [1, 5]

Cluster 2: [2]

Cluster 3: [3, 4, 7]

Cluster 4: [6]</answer>

Figure 7: Case study

independent. **Dylan**: *However, the Constitutional Court dismissed her appeal. They confirmed the Regional Authority's independence and upheld their decision to block the land transfer.* **Ono_Anna**: *Meanwhile, the applicant and her family lived in Germany with temporary residence permits. She was even willing to apply for Austrian nationality to resolve the issue.*]

SPEECH_4: [**Ono_Anna**: *Mr. Jean-Marc Pailot, born in 1952, is a French clerical worker and haemophiliac who received multiple blood transfusions. On August 27, 1985, he tested positive for HIV. Seeking compensation, he approached the Minister for Solidarity, Health and Social Protection, but his claim was rejected. Mr. Pailot then took his case to the Châlons-sur-Marne Administrative Court, which referred the matter to the Conseil d'Etat. Initially, the Administrative Court held the State liable for infections from non-heat-treated blood products between March 12 and October 1, 1985, but expert evidence could not pinpoint the exact infection date. Appeals followed, involving*

the Deputy Minister for Health and the Minister of Employment and Social Affairs. Ultimately, the Conseil d'Etat overturned earlier rulings and found the State liable for infections occurring between November 22, 1984, and October 20, 1985, ordering compensation. The case was a legal dispute between an individual and regulatory authorities, with no indication of vulnerability beyond Mr. Pailot's medical condition, proceeding through administrative judicial review.]

SPEECH_5: [**Serena**: *On July 19, 1996, two unarmed conscripts, Mr. Angelov and Mr. Petkov, fled detention and were chased by four military police officers sent to arrest them.* **Uncle_Fu**: *Right, and these officers were told to use whatever means necessary. They found the men at their grandmother's house in Lesura, where the fugitives tried to escape through a window and over fences.* **Serena**: *Sergeant N. shouted "Stop, military police!" but didn't fire any shots. Then Major G. gave warnings and fired multiple shots with his automatic rifle, aiming at their feet to stop them.* **Uncle_Fu**: *Witnesses confirmed Major G. was the only one who fired live rounds. The others fired shots into the air. Both men were wounded but alive when taken to the hospital, though they died on the way.* **Serena**: *This incident involved direct police use of force that resulted in death, leading to complaints about police conduct. The officers knew who the fugitives were, and the shooting happened during a direct encounter.*]

SPEECH_6: [**Ono_Anna**: *The applicant was injured in a work accident back in 1993 and decided to sue the insurance company ZT for damages in civil court.* **Sohee**: *Between 1996 and 1999, she submitted eight preliminary filings, presented evidence, and requested six times that a hearing date be set. Eventually, three hearings took place, none adjourned at her request, and a medical expert was appointed to assess the case.* **Eric**: *The initial judgment partially upheld her claim, but in 1997, the presiding judge was replaced. After ZT appealed, the higher court partly allowed the appeal and sent the case back for re-examination.* **Ono_Anna**: *Following that, the applicant filed four rush notices to speed up the process. There were also decisions on costs, and ZT's appeals against those were rejected by 2002.* **Sohee**: *Throughout the dispute, which involved private parties, the court provided protection. Although there were procedural delays and remands, no specific vulnerability of the applicant was indicated during the proceedings.*]

SPEECH_7: [**Uncle_Fu**: *In 1997, the applicant was detained on suspicion of drug trafficking, and customs officers seized his car, belongings, documents, and money but refused to make an inventory. The damaged car was transferred by the Customs Service to a private company, which returned it missing parts, with torn documents and lost money. The applicant sued the Sverdlovsk Customs Service for compensation for both financial and non-financial damages, starting civil proceedings that were suspended while criminal investigations against two customs officers allegedly responsible for the damage were ongoing. The courts initially allowed some claims but later rejected them, with the criminal case still unresolved. After the applicant died in 2007, his mother and daughter joined the civil case, represented by his sister, and hearings resumed in 2009. This dispute involves an individual challenging a regulatory authority over property loss and inadequate compensation, with the civil litigation still ongoing in the first instance court.*]

H LIMITATIONS

SpeechLens-Bench is constructed through controlled synthesis. This design allows us to systematically evaluate whether models can follow different linguistic and paralinguistic clustering perspectives, but it may not fully capture the variability of real-world speech collections, such as spontaneous disfluencies, overlapping speakers, accents, reverberation, and uncontrolled background noise. Since several source corpora are originally text- or dialogue-based, our pipeline also relies on transcript expansion, compression, rewriting, and speech synthesis. Although we apply validation and filtering to preserve clustering evidence and reduce lexical shortcuts, the resulting audio may still be cleaner and more structured than naturally occurring recordings. In addition, our held-out perspectives evaluate cross-perspective generalization within the constructed benchmark; collecting real user-written perspectives, testing on broader speech domains, multilingual settings, and noisier recordings remain important future directions. We also focus on speech clustering rather than natural sound or music clustering, which involve different units, criteria, and modeling assumptions. This choice keeps the central challenge of this work focused on combining linguistic content with paralinguistic speech cues under user-specified clustering criteria.

Another limitation is scalability. SpeechLens-R1 is evaluated on moderate-size clustering instances where all audio segments can be jointly processed by the LALM. Current LALMs impose practical limits on the number and duration of audio clips that fit into the input context. As the collection grows, the model must compare increasingly many segment-level and group-level relations, while long recordings may require truncation, summarization, or chunking, potentially discarding useful clustering evidence. We view this limitation as largely orthogonal to the main goal of this work: establishing whether an LALM can interpret a user-specified perspective, compare multiple speech segments, infer the number of clusters, and produce a valid partition. Scaling to larger repositories may require hierarchical clustering, retrieval-augmented comparison, memory mechanisms, streaming inference, or future LALMs with longer audio context.

I Ethical Considerations

All experiments are conducted using publicly available datasets and models licensed for research use. Since our benchmark uses synthesized speech, it does not introduce new recordings of human speakers. However, speech clustering systems may be applied to sensitive attributes such as speaker traits or emotion. We therefore recommend that such systems be used with appropriate consent, privacy protection, and safeguards against profiling or unsupported demographic inference.