TOLD, THEN AWARE: ON-POLICY DISTILLATION FOR EMPATHETIC SPEECH-TO-SPEECH DIALOGUE

Anonymous authors
Paper under double-blind review

ABSTRACT

The same sentence spoken in a different tone calls for a different reply, yet end-to-end speech-to-speech (S2S) models usually respond to the words and largely ignore the voice, even when the delivery contradicts what is said. The common remedy is more empathetic dialogue data with paired annotations, but that data is scarce. We find the solution already latent in the model: a pretrained S2S model can carry emotion in its own speech, just not always the emotion empathy calls for, and it can answer empathetically both in content and in how it sounds as soon as the user's affect is explicitly stated in words. What is missing is not the capability but an objective that requires the voice to change the reply. We therefore adopt on-policy distillation (OPD): the teacher is prompted with the user's affect information as a training-time privilege, and this frozen privileged teacher provides dense per-position distributional supervision over both text and speech positions along the student's own sampled response trajectories. On SpeechParaling-Bench, our method on one backbone improves the Chinese score from 43.6 to 56.3 and the English score from 60.5 to 71.8, outperforming all evaluated systems. We observe consistent gains on EchoMind, VocalBench-zh, and with an alternative backbone. Furthermore, human evaluation confirms that the model trained via our approach achieves the best performance, demonstrating OPD's potential for generating empathetic spoken responses.

1 INTRODUCTION

S2S models can now hold open-ended conversations with humans and yield satisfying experiences. Speech carries information absent from text: if S2S models accurately capture paralinguistic cues such as affect from user speech and respond appropriately, human–machine interaction quality improves. Nevertheless, many S2S models cannot do this robustly. Even when generating emotionally expressive replies, they may misperceive user affect, and their performance degrades when vocal affect contradicts transcribed text sentiment. Consider a user saying *"I just opened the email, I got the offer"* in a flat, downcast voice. The words invite congratulation, the delivery does not, and current S2S models congratulate either way, spoken flat or spoken with delight: a documented tendency of speech language models to transcribe rather than listen (Chen et al., 2026a). Writing x for the utterance, y for the reply, $c(x)$ for the transcript and $\pi(y \mid x)$ for the distribution over replies a model defines, such models belong to

$$\Pi_0 = \{\pi \ : \ \pi(y \mid x) = \pi(y \mid c(x))\}, \tag{1}$$

the set of models whose replies are unchanged when the same words are read again in a different voice. We call this mismatch between what the model can extract from the voice and what its replies the *perception–behavior gap*. We ask what keeps models in this gap Π_0.

The gap is not a failure of perception. We freeze the model and fit linear classifiers along its audio path, from the input spectrogram to the representation the language model actually receives: accuracy rises sharply after the spectrogram and is unchanged by downsampling, so the affective state is still recoverable where the audio is handed to the language model (shown in Table 1). Whatever keeps the model in Π_0, it is not a loss of information in perception. Nor is the ability to respond missing. We compare two views of one frozen S2S model on the same audio: the *bare* view receives the audio alone, as at deployment, while the *told* view additionally receives, in text, a short description of how the user sounds. On a listening set of human recordings with sentences each spoken

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.

Table 1: Affect recovered by linear classifiers from frozen representations at four stages of the audio path. A logistic regression is fit to predict one of four affective states. Accuracy is the fraction of utterances whose state is predicted correctly, Macro-F1 averages F1 over the four classes, and chance is 0.25. Each sentence appears in every affective state and in every voice, so neither the words nor the speaker reveals the label, and every fold tests on a voice the classifier never heard. The last row is the representation the language model actually receives; decodability there does not mean the language model knows what these directions mean, nor that it knows how to use them. Appendix A details the experiment.

Representation	Accuracy	Macro-F1
Input spectrogram	0.449	0.459
Speech encoder output	0.875	0.876
After downsampling	0.855	0.856
As given to the LM	**0.855**	**0.856**

"I just opened the email, I got the offer"

Instructed also gets: "…pay attention not only to what the speaker says but also to the speaker's paralinguistic information…" | **Told** also gets: "... the user speaks in a flat, downcast voice"

Bare: That's wonderful, congratulations! Shall we look through the offer together?

Instructed: That's wonderful, congratulations! Take a breath, look through the details slowly, then decide on your next step. Shall I go over the terms with you?

Told: It sounds like this leaves you conflicted. Congratulations on the offer, and yet it seems to weigh on you. Do you want to tell me what feels heavy?

Figure 1: One pair from the listening set. All three replies come from the same frozen model and differ only in what its input adds to the audio. Read against the words alone, **Bare** and **Instructed** answer alike; **Told** is the only one that answers the delivery too.

twice with opposite delivery over an identical transcript, the bare view gave essentially the same reply in every pair; the told view separated every pair and answered correctly (shown in Figure 1). So we can tell that what the bare view lacks is the affect, not the words.

If the model hears the affect and can act on it once told, what is left to blame is the training itself, and the gap might exist in the objectives themselves: no term of a supervised or preference loss takes its value from the voice alone, so on a single utterance a reply informed by the voice and one that merely suits the words are nearly indistinguishable in loss (Kim et al., 2026), so listening is never rewarded over transcribing. Existing remedies compensate for this missing requirement instead of writing it into the loss: collecting empathetic dialogue data with paired emotion annotations, which is costly and whose labels do not always match what a listener hears in the audio; bolting an emotion recognizer onto the model, which reduces the audio to a predicted label and lets the reply depend on that label rather than on the voice itself, or optimizing the score of an audio judge with reinforcement learning, which compresses the reply into one scalar and stakes training on an external model. None of these puts the requirement where the diagnosis locates the deficit: inside the loss, attached to the voice. An objective that requires it has to be built on purpose: Section 3 does exactly that, and Section 5 returns to what makes Π_0 uniformly suboptimal rather than merely permitted.

As mentioned above, if stating the affect works, perhaps merely *instructing* the model to attend to it works too, as proposed for eliciting latent paralinguistic competence (Wang et al., 2026c), where it can indeed be enough. However, on the backbones we test it is not: substituted verbatim for the affect description, the instruction changes how a reply is worded but not what it responds to, and the pairs stay where the bare view left them (shown in Figure 1). An instruction asks the model to locate the cue on its own while stating the cue outright hands over the one thing the reply has to depend on. We therefore stop asking and start telling: state the cue, and make the told model the teacher. We adopt an on-policy distillation (OPD) approach, prompts the same frozen backbone with the user's perceived affect and the transcript in plain text as a training-time privilege, and trains a student that hears only raw audio, exactly as at deployment, to match it. The student samples its own replies, and the privileged teacher supervises every position of those replies, text tokens and speech tokens alike, with a full per-position distribution rather than a single scalar; nothing beyond low-rank adapters is trained, and inference needs no labels, no transcripts and no extra module.

Overall, our contributions are multifold:

- **We identify why S2S models fail to reply empathetically.** We fit linear probes along the frozen audio path and compare a bare and a told view of the same frozen backbone on

identical audio, and find that the user's affect is still recoverable where the audio is handed to the language model, and that the model answers different deliveries of one sentence apart as soon as that affect is stated to it in words. We find that failure therefore lies in how the model uses the affect it receives, not in whether the paralinguistic information reaches it.

- **We turn a pretrained S2S model into a reliably empathetic one without new supervision.** We propose a method using OPD to teach S2S models to be empathetic, which makes the told backbone its own teacher, so training needs no corpus of large amounts of paired user–assistant empathetic dialogues, no emotion recognizer at inference and no external judge or reward model. The student is supervised on the speech it produces as well as on the text, so it learns to produce empathetic replies in both content and vocal delivery, rather than only in textual content.

- **We explain why on-policy distillation works.** We compare OPD against directly supervising replies, by privileged-target fine-tuning and by preference optimization, and find that neither brings the gains we are after. We then explain this theoretically: reply-level objectives can isolate affect only through a manually injected affect feature, which the model learns to read instead of the audio, whereas OPD's two views differ in the affect alone, and we prove its loss reaches zero only if the audio representation itself encodes affect. We further ablate the reply's own speech tokens from the training sequence and show, empirically and theoretically, that their removal buys text scores at the cost of stable speech.

- **Consistent performance improvements across backbones, benchmarks, and human evaluations.** We evaluate on three public benchmarks including Chinese and English. On SpeechParaling-Bench (Liu et al., 2026b) our method improves the Chinese score from 43.6 to 56.3 and the English score from 60.5 to 71.8, outperforming all evaluated systems, and we observe consistent gains on EchoMind (Zhou et al., 2026a) and VocalBench-zh (Liu et al., 2026a) and on different backbones we train. Our human evaluation confirms the same ordering, with our model rated best by listeners.

2 Related Work

End-to-end speech-to-speech models now converse fluently (Zeng et al., 2024; Xu et al., 2025b; KimiTeam et al., 2025), and some already train explicitly for paralinguistic ability: Fun-Audio-Chat (Tongyi Fun Team et al., 2026) includes a voice-empathy dimension in its multi-task preference optimization, and Step-Audio 2 (Wu et al., 2025) combines paralinguistic captioning, emotion-instructed dialogue scripts, and reinforcement learning. Yet these recipes depend on costly curated data and preference annotation, and the resulting behavior remains fragile, evaluations agree: such models "transcribe rather than listen" (Chen et al., 2026a), collapse under acoustic–semantic conflict (Wang et al., 2026b), and are dominated by lexical semantics (Zhao et al., 2026); even end-to-end S2S replies are rated no better than a pipeline that ignores the speaking style (Yang et al., 2026). Section 1 localizes the failure: affect is already decodable from frozen representations, so the bottleneck is the training objective, not perception.

Many methods for empathetic response take speech in but answer in text (Wang et al., 2024; Chen et al., 2025; Wang et al., 2026a). Among models that answer in speech, most train on purpose-built empathetic dialogue corpora (Wang et al., 2025a; Geng et al., 2025; Wang et al., 2025b; Gao et al., 2026), which are costly to collect; others attach an affect encoder or an explicit condition (Hong et al., 2026; Dindar et al., 2026), or optimize a judge's reward with GRPO (Yang et al., 2026). Two recent methods instead supervise the model with a privileged view of its own input. ParaBridge (Wang et al., 2026c) prompts one view to attend to the user's voice and lets it supervise the model's own replies, but a prompt only reminds the model to look, and that view keeps changing as the student trains. X-OPD (Cao et al., 2026) treats text itself as the privilege: a separate, stronger language model scores text-only replies with a scalar per token, for capability alignment rather than empathy. Our method operates in a similar yet distinct manner: it provides the perceived emotion and transcript directly in plain text, remains frozen, and supervises both the textual and speech components of the reply.

3 METHODOLOGY

3.1 PROBLEM SETUP

We study empathetic speech-to-speech dialogue: given a spoken utterance x, the reply should follow the user's vocal state, in what it says and in how it sounds. The supervision our setting needs does not rely on explicit answer or reward, what it does supply is input-side information that inference never sees, the perceived affect e and the transcript c of each training utterance. This pair is *privileged information*, and the task is to convert it into behavior that persists when the model is given raw audio alone. Section 1 has shown that S2S models already answer empathetically once the affect is stated to it in text: *the told backbone is itself the teacher we are looking for*, and its next-token distributions turn the privilege we already hold into supervision that is dense and per-position.

3.2 TRAINING CORPUS

We construct a training set, a corpus of 34,305 spoken user utterances, roughly balanced between Chinese and English, with texts written by Qwen3-30B-A3B (Yang et al., 2025) and speech synthesized by IndexTTS2 (Zhou et al., 2026b). Lexical–vocal conflict is deliberately designed, so that how a user sounds cannot be read off what they say. Within each language, the utterances cover a wide range of affective states and are spoken in a variety of voices. Each utterance is annotated with the affect a listener perceives in it, fixed by human audit and stratified listening checks. Appendix B describes the corpus in full, and the training set will be released upon publication.

3.3 ON-POLICY DISTILLATION FROM A PRIVILEGED TEACHER

We adopt on-policy distillation (OPD) (Agarwal et al., 2024; Gu et al., 2024), in which a student learns from a teacher's distributions evaluated along the student's own sampled trajectories. Our instantiation introduces no model beyond the backbone itself. The *student* π_θ is the backbone equipped with trainable adapters and receives the raw user audio x, exactly as at inference; the *teacher* is the same backbone with adapters disabled, and receives in addition the privilege (e, c) stated in plain text within its context. The student samples its own response trajectory y, and both views are evaluated along the same prefixes:

$$p_t = \pi_\theta(\cdot \mid x, y_{<t}), \qquad q_t = \pi(\cdot \mid x, e, c, y_{<t}). \tag{2}$$

Because supervision lives on trajectories the student itself produces, the training signal matches the test-time distribution by construction.

The response comprises two disjoint sets of positions, text positions $\mathcal{T}_{\text{txt}}$ and speech positions $\mathcal{T}_{\text{sp}}$, and the objective is a per-position Jensen–Shannon divergence (JSD) over both,

$$\mathcal{L}(\theta) = \sum_{t \in \mathcal{T}_{\text{txt}}} \text{JSD}\left(p_t \| q_t\right) + \lambda_{\text{sp}} \sum_{t \in \mathcal{T}_{\text{sp}}} \text{JSD}\left(p_t \| q_t\right), \tag{3}$$

where

$$\text{JSD}\left(p_t \| q_t\right) = \tfrac{1}{2}\,\text{KL}\left(p_t \| m_t\right) + \tfrac{1}{2}\,\text{KL}\left(q_t \| m_t\right) \in [0, \ln 2], \qquad m_t = \tfrac{1}{2}\left(p_t + q_t\right), \tag{4}$$

and $\lambda_{\text{sp}} = 0.6$. Since the two position sets are disjoint and every position carries its own distributional target, credit assignment across modalities is given by construction and requires no routing (Lin et al., 2026). The speech term is averaged over individual codec tokens rather than over generation steps, and positions beyond the first end-of-speech token are masked out.

Why JSD, and why not reinforcement learning. On-policy sampling makes the divergence consequential: the student inevitably places mass on tokens the teacher assigns near-zero probability. The forward KL, weighted by teacher probabilities, is almost blind to this event, which is precisely what we intend to correct; the reverse KL reacts but is unbounded, so one token in the teacher's null region can dominate the gradient of a batch. The JSD compares both sides against the mixture $m_t \geq \frac{1}{2} p_t$, whose support contains both, and is bounded by $\ln 2$; both terms of Eq. 3 then share the range $[0, \ln 2]$, so λ_{sp} reads as the weight of one against the other. A natural alternative is reinforcement learning with the teacher as a reward model, scoring the student's sampled reply by the

log-probability the teacher assigns it. This optimizes the sequence-level reverse KL rejected above, and beyond that it minimizes the student's entropy, a pressure toward mode collapse (Appendix E), because

$$\max_{\theta} \mathbb{E}_{y\sim\pi_\theta}\left[\log \pi(y \mid x, e, c)\right] = \min_{\theta} \mathrm{KL}\left(\pi_\theta \| \pi(\cdot \mid x, e, c)\right) + \mathcal{H}(\pi_\theta). \tag{5}$$

Why the objective forces listening. The two views share every backbone weight, hear the same audio, and score the same response tokens; the only asymmetry is the privileged text in the teacher's context. Writing $\hat{e}_\theta(x)$ for whatever the student infers about the vocal state from the audio alone, a notational device for an implicit dependence, not an explicit input, the objective of Eq. 3 is a comparison of one function against itself under two conditionings,

$$\mathcal{L}(\theta) = \mathbb{E}\left[\, D\Big(\pi_\theta\big(\cdot \mid x, \hat{e}_\theta(x)\big) \;\Big\|\; \pi\big(\cdot \mid x, e, c\big)\Big)\right], \tag{6}$$

so that everything the two views have in common, the content of the reply, its style, its fluency, cancels in the comparison. What remains is precisely the contribution of the stated affect, and therefore

$$\mathcal{L}(\theta) = 0 \quad\Longleftrightarrow\quad \hat{e}_\theta(x) \text{ is sufficient for } e \text{ as seen through } \pi. \tag{7}$$

The objective is thus, read literally, a sufficiency condition on the student's audio-derived representation: no rewording, no stylistic polish, no fluency gain can drive the loss to zero, only a representation that substitutes for being told. Ignoring the vocal state is not merely discouraged; within this objective it is the only remaining source of loss, in contrast to likelihood-based training, where content and fluency dominate and attending to affect is permitted but never required.

4 Experiments

4.1 Evaluation Setup

Benchmarks. We use three public benchmarks that score how a spoken dialogue system responds to the feeling carried in a user's voice: the English Level 3 items of EchoMind that carry an intended emotional target; the Chinese emotional empathy split of VocalBench-zh; and the `sit_ada` split of SpeechParaling-Bench, in both Chinese and English. Every number in this paper comes from our own experiments and is scored by one pipeline; no system receives in-context examples or a prompt supplied by the benchmark.

Metrics. EchoMind scores a reply on four dimensions with a text judge; we report C4, which asks whether the reply fits how the user sounded, and the average over all four, the other three being context fit, naturalness and colloquialism. Two further scores read the reply audio and not its transcript: EmoAlign, the probability a speech emotion recognizer (Ma et al., 2024), assigns to the emotion the item targets on a 0–100 scale, and VES, a 1–5 rating of how expressively the reply is spoken. On VocalBench-zh, Acoustic is the rate at which an acoustic judge scores the reply at least 3 out of 5, and Overall the rate at which a semantic judge and the acoustic judge both do. On SpeechParaling-Bench we report a win score from 0 to 100 against one fixed opponent per language whose replies ship with the benchmark, Doubao Realtime (ByteDance Seed, 2025) for Chinese and Gemini Audio (Google Cloud, 2025) for English. Judge versions, scoring criteria and the remaining metrics are in Appendix D.

Systems compared. We apply the same training to two backbones that differ in how they emit speech: Fun-Audio-Chat and GLM-4-Voice (Zeng et al., 2024). Each backbone therefore contributes two rows, the released model and the same model after our training, and we mark the latter *+ OPD (Ours)*; no other system in this paper is trained by us. Seven released systems that return speech complete the comparison: Qwen2.5-Omni-7B (Xu et al., 2025a), Qwen3-Omni-30B-A3B-Instruct (Xu et al., 2025b), Kimi-Audio-7B-Instruct (KimiTeam et al., 2025), Step-Audio-2-mini, OpenS2S (Wang et al., 2025a), OSUM-EChat (Geng et al., 2025) and GPT-Audio-1.5 (OpenAI, 2026). We additionally report two systems that take speech in but answer in text, ParaBridge and BLSP-Emo (Wang et al., 2024).

Implementation. Both backbones stay frozen and every parameter we train is part of a low-rank adapter (Hu et al., 2022): 45.9M on Fun-Audio-Chat and 10.8M on GLM-4-Voice. Each arm trains for three epochs, 1608 steps, on eight H200 GPUs, taking 42 and 8 hours respectively. The teacher is

Table 2: Results on three benchmarks. All figures come from our own experiments under one scoring pipeline; higher is better everywhere. **Bold** marks the best value in a column and underline the second best. Dashes mark metrics that require speech output.

	EchoMind Level 3				SpeechParaling		VocalBench-zh	
Model Name	C4	C1–C4	EmoAlign	VES	zh	en	Acoustic	Overall
Systems with speech output								
Qwen2.5-Omni-7B	3.2336	3.9876	27.11	2.5648	8.39	12.14	44.6	23.6
Qwen3-Omni-30B-A3B-Instruct	3.4640	3.8268	28.69	2.5744	21.07	29.64	41.4	40.6
Kimi-Audio-7B-Instruct	3.6288	3.6540	22.03	3.0176	36.43	22.14	43.4	33.2
Step-Audio-2-mini	2.8928	3.3728	26.92	2.3088	12.86	21.79	45.4	23.0
OpenS2S	3.4016	3.6844	28.94	1.9376	30.89	26.61	58.4	44.8
OSUM-EChat	3.7184	3.9624	26.23	2.6549	18.93	8.57	42.4	18.8
GPT-Audio-1.5	3.8576	4.2888	31.68	3.1872	24.64	63.93	55.6	**54.2**
Systems that answer in text								
ParaBridge	**4.7356**	**4.7248**	–	–	–	–	–	–
BLSP-Emo	3.2816	3.2832	–	–	–	–	–	–
GLM-4-Voice backbone								
GLM-4-Voice	3.0880	3.7660	26.82	2.5641	26.96	35.54	59.4	46.6
+ OPD (Ours)	3.6304	4.1064	32.02	3.1200	28.04	45.36	**62.2**	51.6
Fun-Audio-Chat backbone								
Fun-Audio-Chat	3.8832	4.3552	29.91	2.9712	43.57	60.54	54.4	43.2
+ OPD (Ours)	4.3248	4.5916	**34.62**	**3.8176**	**56.25**	**71.79**	60.2	**54.2**

the same backbone reading the privileged view, so nothing beyond the adapters is trained or stored. Adapter placement, optimizer settings and the full configuration are in Appendix C.

4.2 Main Results

The gains hold across architectures and across languages. Fun-Audio-Chat predicts a text token and a speech token at every generation step, while GLM-4-Voice alternates between text and speech segments in one stream, and the same recipe with the same data lifts both: on every one of the eight columns in Table 2, both trained arms improve over their backbones. The largest movements sit on the criteria that listen rather than read: vocal expressiveness rises by 22–28% on the two backbones, and the win scores by 29% in Chinese on Fun-Audio-Chat and 28% in English on GLM-4-Voice. Three judges that see different things, the text, the emotion in the waveform and the delivery, agree on the direction for both backbones.

The gain is not cleaner audio and not longer replies. Two things could raise a score without any empathy, and neither is at work. The three signal-quality scores we record move by less than 0.1 between each backbone and its trained arm, so the trained replies sound the same and score higher anyway. And although text judges are known to favor longer answers (Zheng et al., 2023), here reply length correlates negatively with the EchoMind score within every system that answers in English, so no system can buy these numbers by talking more (both analyses in Appendix F).

Reading a reply and hearing it are different abilities. ParaBridge answers in text from a much larger backbone and leads both EchoMind text columns, a fair reading of its replies as text; on the five columns that require speech it has no entry at all. The text columns bound how well a reply reads, not how it is delivered, and Section 5.2 shows the two can come apart entirely.

5 Analysis

5.1 Why On-Policy Distillation

Ruling out affect-blind replies requires a counterfactual. We call a policy *blind* when it lies in the set Π_0 of Equation 1: it answers the words and ignores the voice. Section 1 located the failure not in perception but in identification: no term of a supervised or preference loss over replies takes a value that depends on the affect alone, so a blind policy fits the data about as well as a listening one

(Kim et al., 2026). Driving blindness out of the low-loss set requires a counterfactual, two signals that differ in the affect and in nothing else, and there are exactly two places to build one.

Input side: make the affect a condition, then swap it. Before adopting OPD we tried to supervise the reply directly. The backbone's own replies under the privileged view of Section 3 do respond to the affect, so we took one such reply, written y^+, as a target, first for fine-tuning and then as the preferred side of a preference pair. Neither route produced the gains we were after: *the trained models still answered the words and let the delivery pass unnoticed*. The failure is built into the objectives, y^+ carries the affect, but no term of either objective reads it, and reproducing the words satisfies both. To put the affect where the loss can vary it, we attached it to the context as its own explicit block: a small frozen module, trained to recognize the perceived emotion of the input audio as people hear it, reads out an affect feature $e(a)$. Training then scores y^+ twice: once under the affect feature of the actual input and once under a feature from a different utterance, written $\tilde{e}$, with the transcript, the audio representation and the reply all held fixed. The preference loss asks the model to prefer the matched condition,

$$L_{\text{pref}} = -\log \sigma(\beta\Delta), \qquad \Delta = \log \pi_\theta\big(y^+ \mid h, e(a)\big) - \log \pi_\theta\big(y^+ \mid h, \tilde{e}\big)\,, \tag{8}$$

where h is everything in the context except the affect condition and β is the usual preference temperature (Rafailov et al., 2023).

Proposition 1 (Ignoring the affect is uniformly suboptimal)**.** *If the policy's output does not depend on the affect block, then* $L_{\text{pref}} = \ln 2$ *whatever the data and whatever* β*, while* $\inf_\theta L_{\text{pref}} = 0$*; and at every such policy the gradient of* L_{pref} *has no component along directions that preserve this independence.*

Proof. Suppose the policy's output is independent of the affect block, $\pi_\theta(\cdot \mid h, e) = \pi_\theta(\cdot \mid h)$ for every e. Both branches of Equation 8 are then the same distribution:

$$\begin{aligned} \Delta &= \log \pi_\theta(y^+ \mid h, e(a)) - \log \pi_\theta(y^+ \mid h, \tilde{e}) = \log \pi_\theta(y^+ \mid h) - \log \pi_\theta(y^+ \mid h) = 0, \\ L_{\text{pref}} &= -\log \sigma(\beta \cdot 0) = -\log \tfrac{1}{2} = \ln 2, \end{aligned} \tag{9}$$

with no dependence on the data or on β, while $\sigma(\beta\Delta) \to 1$ as $\Delta \to \infty$ gives $\inf_\theta L_{\text{pref}} = 0$. Differentiating,

$$\nabla_\theta L_{\text{pref}} = -\beta\, \sigma(-\beta\Delta)\, \nabla_\theta \Delta \;\overset{\Delta \equiv 0}{=}\; -\tfrac{\beta}{2}\, \nabla_\theta \Delta, \tag{10}$$

and along any direction v that preserves the independence, $\Delta \equiv 0$ by Equation 9, so $\langle \nabla_\theta \Delta, v \rangle = 0$ and $\langle \nabla_\theta L_{\text{pref}}, v \rangle = 0$. □

The two branches differ in nothing but the affect block, so only a dependence on it can lower Δ: this is the term the reply-only objectives lacked. It buys an exit and not an arrival, since the model is forced to read the feature $e(a)$ rather than the waveform.

Label side: the privileged teacher is already the counterfactual. OPD needs none of this machinery. Teacher and student are one network with one set of weights, hearing the same audio and scoring the reply the student just produced; the two evaluations differ only in the privileged text the teacher is given (Section 3). The divergence therefore measures only what that text adds, and by Equation 7 it is zero exactly when the audio representation is sufficient as seen through the backbone. That is an arrival and not only an exit: what has to carry the affect is the audio representation itself, not a feature read off it. Both routes can move a model off affect-blind replies; OPD does it without a readout to train in advance or a feature to swap, using only the network's own two views of the same input.

One loss, two modalities, no credit to assign. The other cost OPD does not pay is cross-modal credit assignment. A sequence-level reward is a single number for an output that mixes text and speech tokens, and the policy gradient scales every one of them by it, so a reward earned by the words also reinforces the speech tokens around them, methods built on such a reward have to decide by hand which share of it belongs to which modality (Chen et al., 2026b; Lin et al., 2026). OPD has no such number. The teacher supplies a full distribution at every position, so position t is scored by its own divergence D_t. The two modalities occupy disjoint index sets, and the supervision arrives already addressed:

$$L = \sum_{t \in \mathcal{T}_{\text{txt}}} D_t \;+\; \lambda \sum_{t \in \mathcal{T}_{\text{sp}}} D_t. \tag{11}$$

Table 3: Results from three runs of Fun-Audio-Chat under the scoring pipeline of Table 2. Higher values indicate better performance for all columns except *Invalid*; **bold** denotes the best value within each column. The *Invalid* column counts replies consisting of silence or non-speech audio, aggregated across the three benchmarks.

	EchoMind Level 3				SpeechParaling		VocalBench-zh		
Training Fun-Audio-Chat	C4	C1–C4	EmoAlign	VES	zh	en	Acoustic	Overall	Invalid
+ SFT+PO	4.1280	4.3556	35.42	3.6760	42.14	58.21	56.2	41.0	0
+ SFT+PO-t	**4.5328**	**4.6288**	**37.15**	**3.9488**	46.25	65.18	59.4	39.6	12
+ OPD (Ours)	4.3248	4.5916	34.62	3.8176	**56.25**	**71.79**	**60.2**	**54.2**	0

5.2 Why Speech Tokens Must Stay in the Training Sequence

A speaking model must train on its own speech. Before settling on OPD we also trained the Fun-Audio-Chat backbone by supervised fine-tuning followed by preference optimization over full replies, written *SFT+PO* below. Following the practice of placing the training loss on the text of a reply alone (Wang et al., 2026c), one variant, written *SFT+PO-t*, additionally removed the assistant's speech tokens from the training sequence, so the model never conditioned on its own earlier speech when predicting later speech. Table 3 reports both runs, together with OPD. Though *SFT+PO-t* appears favorable on certain metrics, it is not stable in generating speech tokens, leading to 12 invalid replies like silence or gibberish. This underscores the critical role of exposing S2S models to speech tokens during training for reliable speech production.

Write $s_{1:T}$ for the speech tokens of a reply and u for everything the model conditions on before them, the user audio and the reply's text. Trained with its own speech absent from the sequence, SFT+PO-t can express only distributions of the form $q(s_t \mid u)$: every speech token is predicted from u, none from the speech before it. Proposition 2 measures what such a model can and cannot learn.

Proposition 2 (The marginals are learnable, the joint structure is not)**.** *For every q of the form $q(s_t \mid u)$,*

$$\mathrm{KL}\Big(p(s_{1:T} \mid u) \,\Big\|\, \prod_t q(s_t \mid u)\Big) = \sum_t I\big(s_t;\, s_{<t} \mid u\big) + \sum_t \mathrm{KL}\big(p(s_t \mid u) \,\big\|\, q(s_t \mid u)\big). \tag{12}$$

The second sum vanishes at $q = p(\cdot \mid u)$, itself of this form; the first contains no q at all. The best such model therefore matches every per-token marginal exactly and still falls short of the joint distribution by $\sum_t I(s_t; s_{<t} \mid u)$, the information the speech tokens carry about one another.

Proof. Expand $p(s_{1:T} \mid u)$ by the chain rule and insert the true marginal $p(s_t \mid u)$ into each ratio:

$$\mathbb{E}_p \sum_t \log \frac{p(s_t \mid s_{<t}, u)}{q(s_t \mid u)} = \mathbb{E}_p \sum_t \log \frac{p(s_t \mid s_{<t}, u)}{p(s_t \mid u)} + \mathbb{E}_p \sum_t \log \frac{p(s_t \mid u)}{q(s_t \mid u)}, \tag{13}$$

whose first term is by definition $\sum_t I(s_t; s_{<t} \mid u)$ and whose second is the sum of marginal divergences in Eq. 12. □

Proposition 2 and Table 3 are one fact seen twice. What the scores reward is largely carried by the per-token marginals, which *SFT+PO-t* matches, hence the best EchoMind columns in the table; what holds a stretch of speech together is the conditional information $\sum_t I(s_t; s_{<t} \mid u)$, the one term no model of its form can learn, hence the twelve invalid replies. How well a reply reads and whether it can be spoken at all are settled by different terms of the same decomposition, and keeping the reply's own speech in the training sequence is what makes the second term reachable at all.

Conditioning protects the joint, the loss decides the rest. Once the reply's speech stays in the sequence, the failure above is gone, whatever loss is placed on those tokens. What the loss still decides is how well the model learns from them, and there OPD comes out ahead of preference training: our reading is that a full teacher distribution at every position is a gentler signal for a vocabulary this fine-grained than the likelihood of a single sampled token.

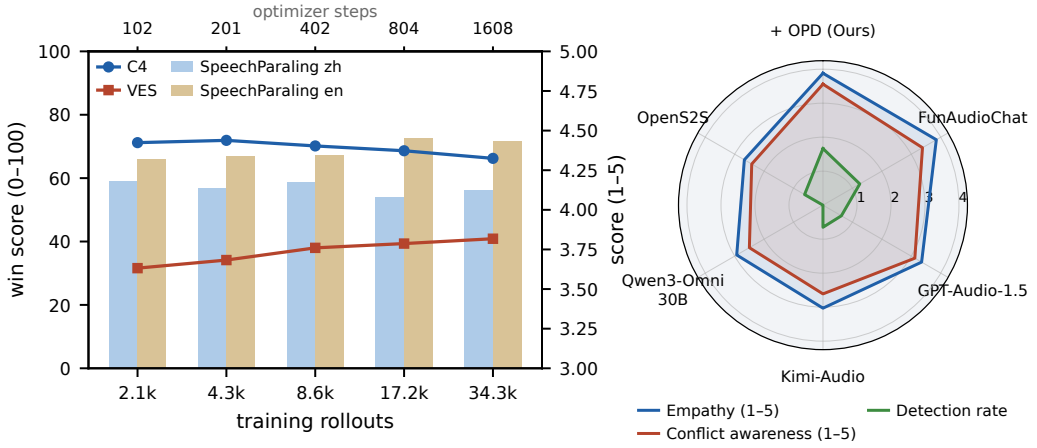

Figure 2: Left: Fun-Audio-Chat model retrained on subsets of the training data, one sixteenth to full; bars are SpeechParaling win scores (left axis), lines are C4 and vocal expressiveness (right axis), and the top axis gives the optimizer steps each subset amounts to. Right: human evaluation of six systems; the rings mark scores 1–4, and the detection rate shares the same radial axis, where the rings stand for 20%, 40%, 60% and 80%.

5.3 How Much Data Works, and What Human Ears Say

The supervision is attached to positions, not samples. We retrain the Fun-Audio-Chat model on nested subsets of the training data, from the full 34,305 rollouts down to one sixteenth. Across a sixteen-fold range of data the scores stay close to flat (shown in Figure 2, left; per-point numbers in Appendix I). This is the expected shape for on-policy distillation, which is known to carry far on very little data (Fu et al., 2026): the teacher hands the student a full distribution at every position of every rollout, so a few thousand rollouts already cover the states worth supervising, and what limits the student is how fast it absorbs that supervision, a matter of optimizer steps rather than of rows.

The gains are audible to human listeners. Every score so far is assigned by a model, so we had people listen. The test set is twenty user recordings built around the case this paper is about: the same words spoken once with a matching voice and once with the opposite emotion. Three listeners rated every reply of six systems without knowing which system spoke, for how empathetic it sounds and, on the conflicting recordings, for whether it takes up both the words and the voice. Our policy rates highest on both and catches the conflict most often, 33.3% of the time against 25.0% for its released backbone and at most 13.0% for any external system (shown in Figure 2, right). The judges' verdict survives human listening: on-policy distillation moves an S2S model toward replies people experience as empathetic. Details are in Appendix H.

6 Conclusion

S2S models fail at empathy not because the voice is lost: the user's affect is still decodable where audio enters the language model, and the model answers empathetically the moment that affect is stated in text. What is missing is an objective that requires the voice to change the reply. Supervised fine-tuning and preference optimization can be made to require it, but at the cost of extra machinery: the affect has to be injected as an explicit condition and swapped to build the contrast, so the empathy runs through that added module rather than through the model's own hearing. We instead adopt an OPD approach to fine-tune the model: the teacher is the frozen model itself, which additionally receives the user's affect and transcript of each training utterance in plain text, and it supervises every position of the replies the student samples from raw audio alone. This training consistently improves performance on the three benchmarks, across two backbones and two languages, and human listeners confirm the improvement. However, this work focuses on single-turn dialogue; carrying a user's emotional state across turns of a conversation, and following it as it shifts within a single utterance, are open challenges we leave to future work.

AI USE STATEMENT

In this work, we used generative AI tools for the following tasks. *To aid or polish writing*: drafting and reorganizing the structure, and consolidating material into the appendix. *For retrieval and discovery*: locating related work and checking how prior systems report their results; every cited paper was read by us before we relied on it. *For research ideation or execution*: writing and debugging the inference scripts that run each compared system on each benchmark. *For generating synthetic datasets*: synthesizing the training corpus of Appendix B; every clip was listened to and checked for intelligibility and label agreement before entering the corpus. *For proving mathematical claims*: discussing how to present the propositions of Section 5.1, 5.2 and how far each statement should reach. The propositions and every step of their derivations were carried out and checked by us; the assistance was in framing and exposition.

We have reviewed all AI-assisted work, and take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024.

ByteDance Seed. Doubao Realtime Voice Model. `https://seed.bytedance.com/en/realtime_voice/`, 2025. Accessed September 2026.

Di Cao, Dongjie Fu, Hai Yu, Siqi Zheng, Xu Tan, and Tao Jin. X-OPD: Cross-modal on-policy distillation for capability alignment in speech LLMs. In *Proc. Interspeech*, 2026.

Jing-Han Chen, Bo-Hao Su, Ya-Tse Wu, and Chi-Chun Lee. RE-LLM: Refining empathetic speech-LLM responses by integrating emotion nuance. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2025.

Jingyi Chen, Zhimeng Guo, Jiyun Chun, Pichao Wang, Andrew Perrault, and Micha Elsner. Do audio llms really listen, or just transcribe? measuring lexical vs. acoustic emotion cues reliance. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5848–5877, 2026a.

Yifu Chen, Shengpeng Ji, Qian Chen, Tianle Liang, Yangzhuo Li, Ziqing Wang, Wen Wang, Jingyu Lu, Haoxiao Wang, Xueyi Pu, Fan Zhuo, and Zhou Zhao. WavAlign: Enhancing intelligence and expressiveness in spoken dialogue models via adaptive hybrid post-training. *arXiv preprint arXiv:2604.14932*, 2026b.

Sukru Samet Dindar, Riki Shimizu, Xilin Jiang, and Nima Mesgarani. Sympatheia: Emotionally adaptive voice assistant with continuous affect conditioning. *arXiv preprint arXiv:2606.00851*, 2026.

Zixuan Fu, Bingxiang He, Yuxin Zuo, Haohuan Huang, Jinqian Zhang, Ruhang Xiao, Cheng Qian, Qinyu Luo, Huan-ang Gao, Yudong Wang, et al. Rethinking on-policy distillation of large language models ii: One training example. *arXiv preprint arXiv:2609.04172*, 2026.

Zhuoyue Gao, Xiaohui Wang, Xiaocui Yang, Wen Zhang, Daling Wang, Shi Feng, and Yifei Zhang. ES4R: Speech encoding based on prepositive affective modeling for empathetic response generation. *arXiv preprint arXiv:2601.16225*, 2026.

Xuelong Geng, Qijie Shao, Hongfei Xue, Shuiyuan Wang, Hanke Xie, Zhao Guo, Yi Zhao, Guojian Li, Wenjie Tian, Chengyou Wang, Zhixian Zhao, Kangxiang Xia, Ziyu Zhang, Zhennan Lin, Tianlun Zuo, Mingchen Shao, Yuang Cao, Guobin Ma, Longhao Li, Yuhang Dai, Dehui Gao, Dake Guo, and Lei Xie. OSUM-EChat: Enhancing end-to-end empathetic spoken chatbot via understanding-driven spoken dialogue. *arXiv preprint arXiv:2508.09600*, 2025.

Google Cloud. Gemini Live API now GA on Vertex AI. `https://cloud.google.com/blog/products/ai-machine-learning/gemini-live-api-available-on-vertex-ai`, 2025. Accessed September 2026.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Yun Hong, Yan Zhou, and Yang Feng. FreezeEmpath: Efficient training for empathetic spoken chatbots with frozen LLMs. *arXiv preprint arXiv:2604.18159*, 2026.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Hangrui Hu, Xinfa Zhu, Ting He, Dake Guo, Bin Zhang, Xiong Wang, Zhifang Guo, Ziyue Jiang, Hongkun Hao, Zishan Guo, Xinyu Zhang, Pei Zhang, Baosong Yang, Jin Xu, Jingren Zhou, and Junyang Lin. Qwen3-tts technical report, 2026. URL https://arxiv.org/abs/2601.15621.

Minseok Kim, Jingxiang Chen, Seong-Gyun Leem, Yin Huang, Rashi Rungta, Zhicheng Ouyang, Haibin Wu, Surya Teja Appini, Ankur Bansal, Yang Bai, Yue Liu, Florian Metze, Ahmed A Aly, Anuj Kumar, Ariya Rastrow, and Zhaojiang Lin. Aligning paralinguistic understanding and generation in speech LLMs via multi-task reinforcement learning. In Yevgen Matusevych, Gülşen Eryiğit, and Nikolaos Aletras (eds.), *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 5: Industry Track)*, pp. 636–648, Rabat, Morocco, March 2026. Association for Computational Linguistics. ISBN 979-8-89176-384-5. doi: 10.18653/v1/2026.eacl-industry.49. URL https://aclanthology.org/2026.eacl-industry.49/.

KimiTeam, Ding Ding, Zeqian Ju, Yichong Leng, Songxiang Liu, Tong Liu, Zeyu Shang, Kai Shen, Wei Song, Xu Tan, Heyi Tang, Zhengtao Wang, Chu Wei, Yifei Xin, Xinran Xu, Jianwei Yu, Yutao Zhang, Xinyu Zhou, Y. Charles, Jun Chen, Yanru Chen, Yulun Du, Weiran He, Zhenxing Hu, Guokun Lai, Qingcheng Li, Yangyang Liu, Weidong Sun, Jianzhou Wang, Yuzhi Wang, Yuefeng Wu, Yuxin Wu, Dongchao Yang, Hao Yang, Ying Yang, Zhilin Yang, Aoxiong Yin, Ruibin Yuan, Yutong Zhang, and Zaida Zhou. Kimi-Audio technical report. *arXiv preprint arXiv:2504.18425*, 2025.

Xiang Lin, Tian-Hao Zhang, Chunfeng Wang, Zhou Pan, Kun Zhan, and Liang Li. Towards more expressive spoken LLMs: Fine-grained intent benchmarking and acoustic-lexical decoupled policy optimization. *arXiv preprint arXiv:2608.03054*, 2026.

Heyang Liu, Yuhao Wang, Ziyang Cheng, Hongcheng Liu, Yiqi Li, Yixuan Hou, Ronghua Wu, Qunshan Gu, Yanfeng Wang, and Yu Wang. Vocalbench: Benchmarking the vocal conversational abilities for speech interaction models, 2026a. URL https://arxiv.org/abs/2505.15727.

Ruohan Liu, Shukang Yin, Tao Wang, Dong Zhang, Weiji Zhuang, Shuhuai Ren, Ran He, Caifeng Shan, and Chaoyou Fu. Speechparaling-bench: A comprehensive benchmark for paralinguistic-aware speech generation. *arXiv preprint arXiv:2604.20842*, 2026b.

Ziyang Ma, Zhisheng Zheng, Jiaxin Ye, Jinchao Li, Zhifu Gao, ShiLiang Zhang, and Xie Chen. emotion2vec: Self-supervised pre-training for speech emotion representation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 15747–15760, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.931. URL https://aclanthology.org/2024.findings-acl.931/.

OpenAI. GPT-Audio-1.5 model documentation. https://developers.openai.com/api/docs/models/gpt-audio-1.5, 2026. Accessed September 2026.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Tongyi Fun Team, Qian Chen, Luyao Cheng, Chong Deng, Xiangang Li, Jiaqing Liu, Chao-Hong Tan, Wen Wang, Junhao Xu, Jieping Ye, Qinglin Zhang, Qiquan Zhang, and Jingren Zhou. Fun-Audio-Chat technical report. *arXiv preprint arXiv:2512.20156*, 2026.

Chen Wang, Minpeng Liao, Zhongqiang Huang, Junhong Wu, Chengqing Zong, and Jiajun Zhang. BLSP-emo: Towards empathetic large speech-language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 19186–19199, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.1070. URL https://aclanthology.org/2024.emnlp-main.1070/.

Chen Wang, Tianyu Peng, Wen Yang, YiNan Bai, Guangfu Wang, Jun Lin, Lanpeng Jia, Lingxiang Wu, Jinqiao Wang, Chengqing Zong, and Jiajun Zhang. OpenS2S: Advancing fully open-source end-to-end empathetic large speech language model. In Ivan Habernal, Peter Schulam, and Jörg Tiedemann (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 906–917, Suzhou, China, November 2025a. Association for Computational Linguistics. ISBN 979-8-89176-334-0. doi: 10.18653/v1/2025.emnlp-demos.70. URL https://aclanthology.org/2025.emnlp-demos.70/.

Dingdong Wang, Shujie Liu, Tianhua Zhang, Youjun Chen, Jinyu Li, and Helen Meng. EmotionThinker: Prosody-aware reinforcement learning for explainable speech emotion reasoning. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=wbttgzp7MT.

Haoyu Wang, Guangyan Zhang, Jiale Chen, Jingyu Li, Yuehai Wang, and Yiwen Guo. Empathy Omni: Enabling empathetic speech response generation through large language models. *arXiv preprint arXiv:2508.18655*, 2025b.

Shuiyuan Wang, Zhixian Zhao, Hongfei Xue, Chengyou Wang, Shuai Wang, Hui Bu, Xin Xu, and Lei Xie. HumDial-EIBench: A human-recorded multi-turn emotional intelligence benchmark for audio language models. *arXiv preprint arXiv:2604.11594*, 2026b.

Yuxiang Wang, Qinke Ni, Shengbo Cai, Wan Lin, Liqiang Zhang, and Zhizheng Wu. Parabridge: Bridging paralinguistic perception and dialogue behavior in speech language models, 2026c. URL https://arxiv.org/abs/2606.10581.

Boyong Wu, Chao Yan, Chen Hu, Cheng Yi, Chengli Feng, Fei Tian, Feiyu Shen, Gang Yu, Haoyang Zhang, Jingbei Li, Mingrui Chen, Peng Liu, Wang You, Xiangyu Tony Zhang, Xingyuan Li, Xuerui Yang, Yayue Deng, Yechang Huang, Yuxin Li, Yuxin Zhang, Zhao You, Brian Li, Changyi Wan, Hanpeng Hu, Jiangjie Zhen, Siyu Chen, Song Yuan, Xuelin Zhang, Yimin Jiang, Yu Zhou, Yuxiang Yang, Bingxin Li, Buyun Ma, Changhe Song, Dongqing Pang, Guoqiang Hu, Haiyang Sun, Kang An, Na Wang, Shuli Gao, Wei Ji, Wen Li, Wen Sun, Xuan Wen, Yong Ren, Yuankai Ma, Yufan Lu, Bin Wang, Bo Li, Changxin Miao, Che Liu, Chen Xu, Dapeng Shi, Dingyuan Hu, Donghang Wu, Enle Liu, Guanzhe Huang, Gulin Yan, Han Zhang, Hao Nie, Haonan Jia, Hongyu Zhou, Jianjian Sun, Jiaoren Wu, Jie Wu, Jie Yang, Jin Yang, Junzhe Lin, Kaixiang Li, Lei Yang, Liying Shi, Li Zhou, Longlong Gu, Ming Li, Mingliang Li, Mingxiao Li, Nan Wu, Qi Han, Qinyuan Tan, Shaoliang Pang, Shengjie Fan, Siqi Liu, Tiancheng Cao, Wanying Lu, Wenqing He, Wuxun Xie, Xu Zhao, Xueqi Li, Yanbo Yu, Yang Yang, Yi Liu, Yifan Lu, Yilei Wang, Yuanhao Ding, Yuanwei Liang, Yuanwei Lu, Yuchu Luo, Yuhe Yin, Yumeng Zhan, Yuxiang Zhang, Zidong Yang, Zixin Zhang, Binxing Jiao, Daxin Jiang, Heung-Yeung Shum, Jiansheng Chen, Jing Li, Xiangyu Zhang, and Yibo Zhu. Step-audio 2 technical report, 2025. URL https://arxiv.org/abs/2507.16632.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-omni technical report, 2025a. URL https://arxiv.org/abs/2503.20215.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, Yuanjun Lv, Yongqi Wang, Dake Guo, He Wang, Linhan Ma, Pei Zhang, Xinyu Zhang, Hongkun Hao, Zishan Guo, Baosong Yang, Bin Zhang, Ziyang Ma, Xipin Wei, Shuai Bai, Keqin Chen, Xuejing Liu, Peng Wang, Mingkun Yang, Dayiheng Liu, Xingzhang Ren, Bo Zheng, Rui Men, Fan Zhou, Bowen Yu, Jianxin Yang, Le Yu, Jingren Zhou, and Junyang Lin. Qwen3-Omni technical report. *arXiv preprint arXiv:2509.17765*, 2025b.

An Yang, Anfeng Li, Baosong Yang, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Shu-wen Yang, Ming Tu, Andy T. Liu, Xinghua Qu, Hung-yi Lee, Lu Lu, Yuxuan Wang, and Yonghui Wu. ParaS2S: Benchmarking and aligning spoken language models for paralinguistic-aware speech-to-speech interaction. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=CcmDDh070o.

Aohan Zeng, Zhengxiao Du, Mingdao Liu, Kedong Wang, Shengmin Jiang, Lei Zhao, Yuxiao Dong, and Jie Tang. GLM-4-Voice: Towards intelligent and human-like end-to-end spoken chatbot. *arXiv preprint arXiv:2412.02612*, 2024.

Zhixian Zhao, Shuiyuan Wang, Wenjie Tian, Jingbin Hu, Ziyu Zhang, and Lei Xie. Beyond semantic dominance: Cognitive affective reasoning and empathetic response alignment in audio language models. In *Proc. Interspeech*, 2026.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Li Zhou, Lutong Yu, You Lyu, Yihang Lin, Zefeng Zhao, Junyi Ao, Yuhao Zhang, Benyou Wang, and Haizhou Li. Echomind: An interrelated multi-level benchmark for evaluating empathetic speech language models, 2026a. URL https://arxiv.org/abs/2510.22758.

Siyi Zhou, Yiquan Zhou, Yi He, Xun Zhou, Jinchao Wang, Wei Deng, and Jingchen Shu. IndexTTS2: A breakthrough in emotionally expressive and duration-controlled auto-regressive zero-shot text-to-speech. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(41): 35139–35148, 2026b. doi: 10.1609/aaai.v40i.40820.

A Linear Probing of the Audio Path

Stimuli. We use 256 synthesized utterances, the cross product of four speakers, four affective states and sixteen sentences. Every sentence is spoken in every state and by every speaker, so neither the words nor the identity of the voice reveals the label, and the task is four-way throughout, with chance accuracy 0.25.

Representations. We read four points along the audio path of the frozen backbone and never update its weights: the log-Mel spectrogram at the front end (100 Hz, 256 dimensions), the output of the speech encoder (25 Hz, 8192), the sequence after downsampling (5 Hz, 8192), and the representation at the position where audio enters the language model (5 Hz, 8192).

Classifier and splits. Each representation is read by its own classifier, a standard scaler followed by a multinomial logistic regression with $C = 1.0$. The evaluation is leave-one-speaker-out over the four speakers, giving four folds of 192 training and 64 test utterances each and sixteen fits in total. The classifier is the only thing trained.

Table 4: Accuracy of the four-way linear probe on held-out speakers. Folds test on a voice the classifier has not heard; the last row is the representation the language model actually receives. Reference voices are the Qwen3-TTS presets of Appendix B.

Held-out speaker	Log-Mel	Speech encoder	After downsampling	As given to the LM
Ryan	0.328	0.672	0.672	0.672
Serena	0.328	0.906	0.891	0.891
Uncle_Fu	0.484	0.969	0.969	0.969
Vivian	0.656	0.953	0.891	0.891
All (macro-F1)	0.459	0.876	0.856	0.856

Reading. Accuracy rises sharply past the front end and is essentially unchanged by the downsampling step that shortens the sequence for the language model, so coarse affect is not merely present but linearly readable at the exact position where audio enters the language model. The hardest unseen voice is Ryan, where the final representation still reaches 0.672 against a chance level of 0.25; the spread across voices is why every fold in the paper, here and in Section 4.1, holds speakers out rather than splitting at random.

Table 5: Composition of AffectPriv. The upper block splits the corpus by whether the words and the delivery agree, the lower block by the coarse category recorded in the manifest; both partition the same 34,305 items. The teacher is told the fine-grained label, not the coarse one.

	Chinese	English	Total
Positive words, downcast voice	7,896	7,974	15,870
Negative words, cheerful voice	7,726	7,703	15,429
Words and voice in agreement	1,827	1,179	3,006
Happy	4,689	4,650	9,339
Melancholic	4,790	4,794	9,584
Calm	2,263	2,273	4,536
Sad	2,255	2,189	4,444
Surprised	1,273	1,332	2,605
Angry	1,325	937	2,262
Anxious	690	578	1,268
Afraid	164	103	267
Total	17,449	16,856	34,305

What the probe does and does not show. It shows that the affect information survives the audio path, which rules out perception as the source of the gap. The classifier that recovers the label is one we trained on the representation, not a part of the model, and nothing here says the backbone's own forward pass consults those directions. Whether it does is answered by the replies of Figure 1, not by this probe.

B THE TRAINING CORPUS: AFFECTPRIV

What it is. We construct a training corpus of 34,305 spoken user utterances, 17,449 Chinese and 16,856 English, which we call **AffectPriv**. Each item is a single user turn with its audio, its transcript, and a label for the affect a listener perceives in the voice, and this is the only data used to train the systems of Section 4.2. The user turns were written by Qwen3-30B-A3B (Yang et al., 2025) under prompts that keep the valence of the words independent of the valence of the delivery, so that a sentence and the way it is spoken can agree or disagree. Most items are contradictory, a positive sentence delivered downcast or a negative one delivered cheerfully: 31,299 of the 34,305, 91.2%. On such an item no reply can be right by reading the words alone, which is what the corpus is built to demand. The texts were themselves checked by 6 people who sampled 6,000 of them, 1,000 each, and read for meaning and for whether the intended conflict was legible.

Speech. Every utterance is synthesized by IndexTTS2 (Zhou et al., 2026b), and the vocal state is not described to the synthesizer in words but set as an explicit 8-dimensional emotion vector with an intensity scalar, drawn from a bank of recipes that human listeners had already approved rather than sampled at synthesis time, so the intended state can be varied independently of the text. Reference voices are the 8 presets of the Qwen3-TTS voice library (Hu et al., 2026), each state routed to voices that carry it well, and the audio is 22,050 Hz mono.

Why a vector is not a label. What the synthesizer is asked for and what a listener hears are different quantities, and the corpus records both separately, because the two diverge systematically on precisely the items that matter most. A positive sentence synthesized with the sad vector is heard as *happy tears*; a positive sentence with the angry vector is heard as indignation rather than rage. A corpus that took the vector as its label would mislabel both, and would do so exactly where a reply has to follow the voice rather than the words. Every utterance therefore carries a separate *perceived affect*: the emotion a listener actually hears, established by listening, and where vector and ear disagree the label follows the ear.

Labels. Perceived affect is recorded at 33 fine-grained values, 17 for Chinese and 16 for English, and it is this label, not the synthesis vector, that is stated to the teacher. Mixed states such as happy tears are recorded under their own names, 2,050 English and 2,029 Chinese items, rather than folded into happy or sad, so a reply is never asked to answer a valence the voice does not carry. Each of these values is a distinct label the teacher may be given. They also carry a second, coarser tag recorded in the same manifest for grouping: every value belongs to one of 8 categories (happy, surprised, calm,

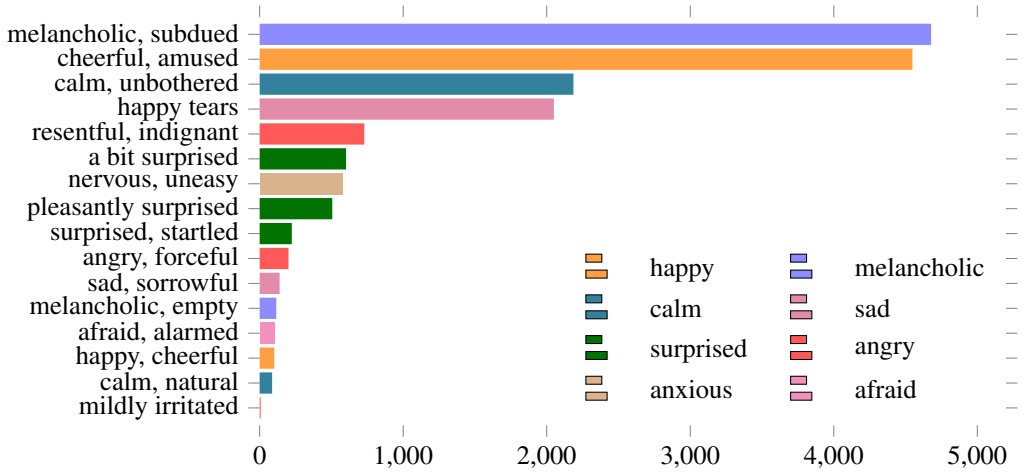

Figure 3: The 16 English perceived-affect labels in AffectPriv, sorted by frequency, each bar coloured by the coarse category the label belongs to. The 17 Chinese labels are a separate inventory with separate values and are listed in the manifest released with the corpus. The distribution is uneven by construction rather than by accident: under our synthesis rules IndexTTS2 renders some vocal states more faithfully than others, and each state was used at the frequency at which the synthesizer could produce it convincingly, so the counts reflect synthesis fidelity rather than how often these states occur in real conversation.

sad, melancholic, angry, anxious, afraid), and the two levels are not interchangeable. Happy tears sits with sad at the coarse level. That is right acoustically and wrong in meaning; this is why the teacher is told the fine-grained value and the coarse tag is never used as supervision. The label set is long-tailed and uneven (shown in Figure 3), spanning a 600-fold range in English and a 50-fold range in Chinese. The corpus is bilingual by design rather than by translation: Chinese and English items were written, synthesized and labelled independently, each language carrying its own label set and its own voices, and the two are close to equal both overall and within every category. Table 3 gives the composition of the corpus.

How the labels were established. The labels were produced by 8 people, the same 8 who fixed the synthesis recipes. A recipe, a voice and an intensity for a given state were proposed and listened to; when 1 listener judged the intensity right, the remaining 7 listened independently, and the recipe was adopted only if at least 4 of the 8 agreed. Recipes that never cleared that bar were not used. Sampling was stratified rather than uniform: the audit followed the joint space of synthesis recipe, language and voice, so that a weakness confined to 1 voice rendering 1 state could not hide inside a global average. Earlier in the construction of this corpus, 3 rounds of 20 randomly drawn utterances had all passed, and a stratified re-listen then found widespread label misalignment; what random listening had certified was not what the corpus needed certified. The claim this appendix supports is that every utterance was produced by a recipe that human listeners had confirmed, and that stratified listening over the corpus found no systematic mislabeling. It is not a claim that every one of the 34,305 utterances was heard. Labels are the only thing ever revised: vectors, voices, identifiers, order and audio files were left untouched, so a revision never alters a waveform. The corpus and the configuration that synthesizes it will be released on publication.

C Training Configuration

Both backbones stay frozen, and every parameter trained is a low-rank adapter (Hu et al., 2022). No full-rank module is trained in any run of this paper. The teacher adds no parameters at all: it is the same backbone reading the privileged view, so a run stores adapters and nothing else.

The teacher's privileged view. The privilege is text, and it is the only thing the teacher sees that the student does not. Figure 4 shows where it sits. On Fun-Audio-Chat the teacher's user turn is the student's audio span followed by two tagged fields, the utterance's transcript and its perceived affect; on GLM-4-Voice the same two fields are introduced by a sentence and placed before the audio span

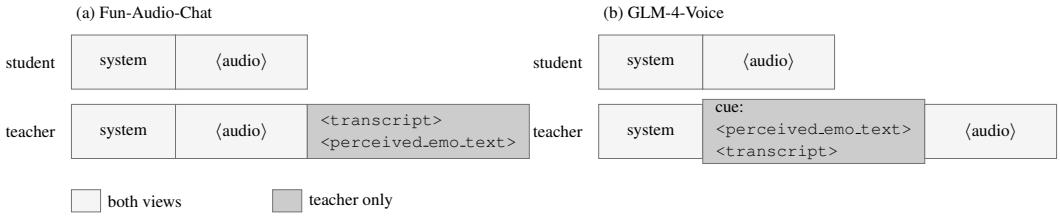

Figure 4: What the teacher sees that the student does not, on each backbone. The student is given the audio span alone. The teacher reads the same sequence with two extra text fields: the transcript of the utterance and its perceived affect, the latter a short natural-language phrase such as *melancholic and subdued*. The two backbones place these fields differently relative to the audio: after it on Fun-Audio-Chat, before it on GLM-4-Voice, where a sentence introduces them. Both fields are present in the teacher's context only during training.

rather than after it. Both fields are written in the utterance's own language. Neither is part of any prompt the student receives, and neither is needed at inference, where the student is given raw audio and nothing else.

The two user turns. Both are given verbatim below and both are released with the code. On Fun-Audio-Chat the fields are bare tags, with no sentence introducing them; on GLM-4-Voice they are prefaced by one, since that backbone's chat template has no equivalent of the tagged format.

```
<|audio_bos|><|AUDIO|><|audio_eos|>
<transcript>
{user_text}
<perceived_emo_text>
{perceived_emo_text}
```

On GLM-4-Voice they are read before the audio instead:

```
Below are the emotion in this user's voice and the actual
words they said.  The two may not point the same way; do
not base your reply on only one of them, and take both into
account when you reply.

<perceived_emo_text>
{perceived_emo_text}
<transcript>
{user_text}
```

The system turn. Both views on a backbone share one system turn, so whatever it says is read by the student too and none of it is privileged. Each backbone keeps its own official system prompt and appends the same instruction to it: reply in the user's language, and respond to the emotion in the voice rather than to the words alone. On Fun-Audio-Chat that instruction is a fixed string in Chinese, rendered identically for every row including the English ones, and it says no more than that. On GLM-4-Voice it is rendered per row in the row's own language, and the English version, which is the one English rows receive, adds a clause that resolves the two signals:

```
Reply in the same language the user speaks, and respond
naturally to the emotion carried in the user's voice; that
emotion is something you hear, not a fact the user stated
or their exact words.  When the literal meaning of the words
conflicts with the emotion in the voice, follow the voice.
```

GLM-4-Voice's system turn also carries a short persona describing how to listen and how long to reply.

The two backbones spend their speech-side capacity differently. Fun-Audio-Chat renders speech through a module that sits upstream of the speech-token logits, so the speech-side adapter is attached there and the two sets of adapters live on different modules. GLM-4-Voice has no such module: its

Table 6: Training configuration of the runs reported in Table 2. Both backbones are frozen, and the counts below are the only trainable parameters. The configurations carrying the affect condition add three tensors and 2,098,176 parameters for the bridge between the readout and the language model. The two backbones distribute the speech-side capacity differently and use different schedules, as described in the text.

	Fun-Audio-Chat	GLM-4-Voice
Trainable parameters	45,940,736 (0.56%)	10,813,440 (0.115%)
Trainable tensors	728	320
Language-side LoRA	rank 16, α 32, dropout 0.05	rank 8, α 16, dropout 0
Attached to	q, k, v, o, gate, up, down	`query_key_value`, `dense`
Layers	36	40
Tensors	504 (43,646,976)	160 (5,406,720)
Speech-side LoRA	rank 8, α 16, dropout 0	rank 8, α 16, dropout 0
Attached to	`audio_invert_tower` attention	same modules as above
Applied where	the audio rendering path	positions predicting a speech token
Tensors	224 (2,293,760)	160 (5,406,720)
MLP projections	trained (gate, up, down)	not attached
Epochs / steps	3 / 1,608 = 536 $\times$ 3	3 / 1,608 = 536 $\times$ 3
Global batch	64 rows	64 rows
Loss temperature	1.0	1.0
Rollout cap	512 tokens	512 tokens
Optimizer	AdamW, weight decay 0, clip 1.0	AdamW, weight decay 0, clip 1.0
Learning rate	5e-6	5e-6, both sets
Schedule	cosine, warmup 0.03	constant, warmup 0
Hardware / wall clock	8 $\times$ H200 / 42 h	8 $\times$ H200 / 8 h

audio decoder is a flow-matching model and a neural vocoder downstream of discrete unit sampling, where no gradient from a distributional objective arrives (see below). Its second set of adapters is therefore placed on the same linear projections as the first and distinguished by position instead, applied only where the token being predicted is a generated speech token, so the text positions are trained by the text policy and the speech positions by the rendering policy. The two configurations are the same idea carried out with the parts each architecture provides, which is why the parameter counts above should not be read as one backbone having been given more capacity than the other.

The second backbone was given no tuning budget. The learning rates and optimizer settings on GLM-4-Voice were carried over from the Fun-Audio-Chat run and never adjusted; the training script declares the transfer explicitly rather than leaving it implicit, and the step count is fixed at the same 1,608 for both. The schedule was not transferred: it is constant on GLM-4-Voice against cosine with warmup on Fun-Audio-Chat, which is the choice that asks for no tuning of its own.

Rollouts. The student samples its own replies during training, and the teacher scores exactly those tokens, so the supervision sits on the student's own trajectory rather than on a corpus. A rollout is capped at 512 tokens, and positions past the first end-of-speech token are masked from the speech term, which keeps the speech divergence averaged over codec tokens actually emitted rather than over padding.

Why the vocoder is not trained. At a speech position the model emits a distribution over codec tokens, and the divergence of Eq. 3 is defined on that distribution; sampling is the last differentiable step. The vocoder sits entirely downstream of that sample, so no gradient from the objective reaches it and its weights cannot affect what the objective measures. The same argument applies to the audio decoder of GLM-4-Voice, which is why its speech-side capacity is spent inside the language model rather than at the renderer.

Four configurations, three differences. Table 7 lists the configurations of our method. Read as a ladder, each row changes one thing against the row above it: the first gives the speech term no weight in the gradient, the second gives it weight, the third lets that gradient reach the speech-side adapters, and the fourth withholds the affect condition. The teacher, the data, the item order and the schedule are the same throughout. What does not change across the four rows is the one thing Section 5.2 is about: the reply's own speech tokens stay in the training sequence in all four. Every configuration

Table 7: The configurations of our method trained in this paper. The first three share the teacher, the data, the schedule and the trainable parameters.

	Speech term	Speech-side LoRA	Affect condition
Text positions only	$\lambda_{\mathrm{sp}} = 0$	frozen	present
Text and speech positions	$\lambda_{\mathrm{sp}} = 0.6$	frozen	present
Text and speech positions	$\lambda_{\mathrm{sp}} = 0.6$	trained	present
+ OPD (Ours)	$\lambda_{\mathrm{sp}} = 0.6$	trained	absent

Table 8: The sweep that fixed λ_{sp}. Both columns are means over the pilot, in the units the training objective uses; lower is better in both. The value kept is the only one best on both.

λ_{sp}	Text divergence	Speech divergence
0	0.04010	0.00882
0.3	0.03594	0.00614
0.6	**0.03502**	**0.00564**
1.0	0.04060	0.00688

produces usable speech on every item of every benchmark, including the first, in which the term that scores those tokens contributes nothing to the update.

Where the weight on the speech term comes from. $\lambda_{\mathrm{sp}} = 0.6$ was not inherited from prior work and not tuned on any result reported in this paper. It was fixed by a sweep over four values on a 128-row pilot of 32 optimizer steps with byte-identical initial weights, judged on the two training divergences alone.

Both divergences are lowest at 0.6 and rise again at either extreme, so the choice is a minimum rather than an endpoint of the range searched. No benchmark had been scored at that point and none was used to select the value.

D Benchmarks, Judges and the Remaining Metrics

Every number in this paper comes from our own runs: the released systems were generated and scored by us on the same items, under the same scorers, as our own arms, and no system receives in-context examples or a benchmark-supplied prompt.

What each benchmark scores. EchoMind's Level 3 conversation task scores a single-turn spoken reply against an intended emotional target, and we use its *strict emotion target* subset: 9 vocal-cue types (happy, sad, surprise, angry, fear, disgust, sobbing, sigh, laughter), 445 items whose user audio is synthesized and 180 whose user audio is a human recording, 625 in total. SpeechParaling-Bench's `sit_ada` split scores a reply against one fixed opponent per language whose audio ships with the benchmark, Doubao Realtime for Chinese and Gemini Audio for English, over 190 items per language; the reported win score is the raw margin rescaled to 0–100. VocalBench-zh's emotional empathy split scores 500 Chinese items with one judge that reads the reply as text and one that listens to the reply audio, each returning the rate at which it gives at least 3 of 5.

The text judge. The four EchoMind criteria, context fit, response naturalness, colloquialism degree and speech information relevance, are scored one reply at a time on a 1–5 scale by GPT-4o at temperature 0, using the benchmark's own prompt taken verbatim from the released evaluation code. The judge sees the reply transcript, the topic, the speaker's self-description and the user's spoken content, and not the audio, the reference reply, or any label identifying the system. Table 2 reports the fourth criterion, which asks whether the reply fits the way the user sounded, and the mean of all four criterion.

The two audio judges. EmoAlign runs emotion2vec on the reply audio and reports the probability the model assigns to the item's target emotion, averaged over items on a 0–100 scale. It is a nine-way classifier, and the target of a given item is one of seven categories that occur in the subset. VES asks a judge to rate, on a 1–5 scale, how well the reply's vocal delivery fits a one-line description of the speaker's state; the rubric is the benchmark's, and the judge is Gemini 3.1 Pro in place of the

Table 9: What scored each column.

Reported in	Scored by
C1–C4	`gpt-4o-2024-11-20`
EmoAlign	`iic/emotion2vec_plus_large`
VES	`Gemini-3.1-Pro-Preview`
NISQA, DNSMOS	the released NISQA and DNSMOS predictors
SpeechParaling win	`Gemini-3.1-Pro-Preview`
VocalBench semantic	`qwen3.8-max-0902`
VocalBench acoustic	`emotion2vec_plus_large`

Table 10: The columns of Table 2 that are not in it. **Bold** marks the best value in a column and underline the second best. *VB sem.* is VocalBench-zh semantic, the rate at which the text judge scores a reply at least 3 out of 5. The four *reference text* metrics compare the reply transcript with the references EchoMind ships: BLEU, ROUGE-L (*R-L*), METEOR (*MET*) and BERTScore (*BERT*), each the raw score multiplied by 100. The three *audio quality* metrics, NISQA, DNSMOS-OVRL and DNSMOS-P808, predict how a reply sounds without a reference. Dashes mark systems that answer in text and therefore have no reply audio to score.

	VB	Reference text				Audio quality		
Model	sem.	BLEU	R-L	MET	BERT	NISQA	OVRL	P808
Qwen2.5-Omni-7B	43.4	2.072	16.235	13.700	23.531	4.5113	3.4705	**4.1617**
Qwen3-Omni-30B-A3B-Inst.	97.6	2.976	17.286	**17.023**	**26.138**	**4.9035**	3.4750	4.0921
Kimi-Audio-7B-Instruct	64.6	0.630	5.132	2.801	−4.336	4.4005	3.3666	3.6159
Step-Audio-2-mini	49.2	1.369	14.050	7.066	17.257	4.7821	3.3224	3.7666
OpenS2S	74.6	1.480	13.046	7.311	16.941	4.8543	3.3591	3.7981
OSUM-EChat	51.6	0.508	4.270	3.115	−7.979	4.7778	**3.5431**	4.0294
GPT-Audio-1.5	97.4	**3.013**	18.165	15.507	25.760	4.6018	3.4941	4.1052
ParaBridge	**98.8**	0.310	4.667	10.937	11.859	–	–	–
BLSP-Emo	73.8	2.276	16.627	16.980	25.165	–	–	–
GLM-4-Voice	75.4	1.965	13.545	10.034	16.741	4.7901	3.4471	3.8609
+ OPD (Ours)	82.2	2.557	17.072	11.800	22.416	4.8019	3.4609	3.9187
Fun-Audio-Chat	76.6	2.155	17.653	9.943	23.206	4.3921	3.3716	3.8618
+ OPD (Ours)	87.8	2.019	**18.360**	10.721	23.280	4.4375	3.4348	3.9430

Gemini 2.5 Pro the benchmark used. Both judges rate the reply audio alone, and neither hears the user's voice.

E The Objective: Boundedness and Why Not Reinforcement Learning

This appendix supplies the two derivations the objective of Section 3 rests on: that the Jensen–Shannon divergence is bounded by $\ln 2$, which is what lets a single λ_{sp} be read as a weight, and that training on the teacher's likelihood with a policy gradient optimizes the sequence-level reverse KL of Equation 5, which is what keeps this paper from adding a reinforcement learning arm.

The divergence is bounded. Write p for the student distribution at a position, q for the teacher's, and $m = \frac{1}{2}(p + q)$ for their mixture. Then

$$\mathrm{JSD}(p \,\|\, q) = \tfrac{1}{2}\,\mathrm{KL}(p \,\|\, m) + \tfrac{1}{2}\,\mathrm{KL}(q \,\|\, m), \tag{14}$$

and since $m(v) \geq \frac{1}{2}p(v)$ at every token v,

$$\mathrm{KL}(p \,\|\, m) = \sum_v p(v) \log \frac{p(v)}{m(v)} \leq \sum_v p(v) \log 2 = \ln 2, \tag{15}$$

with the same bound for $\mathrm{KL}(q \,\|\, m)$ by symmetry. Averaging the two gives $\mathrm{JSD} \in [0, \ln 2]$, so both terms of Equation 3 share one range and λ_{sp} is a weight rather than a scale factor.

Table 11: The three candidate divergences at a single token, computed by hand. The student puts all of its mass on a token the teacher rules out. This is an illustration of the definitions, not a measurement from training: it is the event on-policy sampling makes common, and the numbers below are its limiting behaviour.

Divergence	Contribution from this token	Behaviour
$\mathrm{KL}(q \,\|\, p)$ (forward)	$q(v) \log \frac{q(v)}{p(v)} = 0$	blind to it
$\mathrm{KL}(p \,\|\, q)$ (reverse)	$p(v) \log \frac{p(v)}{q(v)} = +\infty$	unbounded
$\mathrm{JSD}(p \,\|\, q)$	$\frac{1}{2} p(v) \log \frac{p(v)}{m(v)} = \frac{1}{2} \ln 2$	bounded

Table 11 makes the choice concrete. With $p(v) = 1$ and $q(v) = 0$ at one token, the forward divergence, weighted by teacher probabilities, is exactly zero there: it cannot see the event we most need to correct. The reverse divergence sees it and diverges. The Jensen–Shannon divergence compares each side against a mixture whose support contains both, and returns $\frac{1}{2} \ln 2$.

Training on teacher likelihood is sequence-level reverse KL. A reward model built from the teacher scores a whole reply with a single number, its log-probability under the privileged view, $r(y) = \log \pi(y \mid x, e, c)$. Optimizing the policy against that reward has a closed form.

Proposition 3 (Likelihood-shaped rewards optimize reverse KL and shrink entropy)**.** *For any policy π_θ and any fixed distribution π over replies,*

$$\mathbb{E}_{y \sim \pi_\theta}[\log \pi(y)] = -\,\mathrm{KL}(\pi_\theta \,\|\, \pi) - \mathcal{H}(\pi_\theta). \tag{16}$$

Maximizing the expected teacher log-likelihood is therefore minimizing the sequence-level reverse KL to the teacher together with the entropy of the policy.

Proof. Adding and subtracting $\log \pi_\theta(y)$ inside the expectation and splitting the logarithm,

$$\begin{aligned} \mathbb{E}_{y \sim \pi_\theta}[\log \pi(y)] &= \sum_y \pi_\theta(y) \log \frac{\pi(y)}{\pi_\theta(y)} + \sum_y \pi_\theta(y) \log \pi_\theta(y) \\ &= -\,\mathrm{KL}(\pi_\theta \,\|\, \pi) - \mathcal{H}(\pi_\theta), \end{aligned} \tag{17}$$

where the first sum is the definition of the divergence and the second of the entropy, both nonnegative. □

Two consequences follow, and both are visible in Equation 16. The objective contains the very divergence rejected above, now at the sequence level, where it is mode-seeking: it rewards a policy for putting all of its mass on the replies the teacher already favours rather than for covering the replies the teacher allows. And the second term is not incidental: maximizing teacher likelihood actively minimizes the policy's entropy, which is a pressure toward collapse on top of whatever the divergence does. Equation 5 in the main text states this identity; the point of deriving it here is that neither term is a modelling choice we could have avoided by being careful, since both follow from using a scalar reward at all.

What a scalar reward discards. The cost is not the teacher's forward pass, which both approaches run for every rollout either way. It is what each approach takes from that forward pass. At each position the teacher produces a full distribution over the vocabulary, and $r(y) = \sum_t \log \pi_t(y_t)$ reduces it to the single entry at the token the student happened to sample, then sums those entries into one number per reply. Per-position supervision of the kind in Equation 3 keeps the distributions. A reply of a few hundred positions therefore arrives as one scalar rather than as a few hundred targets, and the credit for a good reply has to be spread across every position in it by a baseline, including the speech tokens, which carry no words to attribute credit to. Sampling K rollouts per prompt to build a group-relative baseline multiplies the generation cost by K while the information per rollout goes down, not up. A reinforcement learning arm would be the right instrument for a signal the teacher cannot supply, such as a preference collected from listeners or a metric defined on the waveform. For a signal the teacher already supplies in full at every position, it is a strictly more expensive way of using less of it.

Table 12: Signal-quality scores for every system in Table 2. **Bold** marks the best value in a column and underline the second best. All three are no-reference audio quality predictors and none of them listens for content or affect. Dashes mark systems that answer in text and produce no reply audio.

Model	NISQA	DNSMOS-OVRL	DNSMOS-P808
Qwen2.5-Omni-7B	4.5113	3.4705	**4.1617**
Qwen3-Omni-30B-A3B-Inst.	**4.9035**	3.4750	4.0921
Kimi-Audio-7B-Instruct	4.4005	3.3666	3.6159
Step-Audio-2-mini	4.7821	3.3224	3.7666
OpenS2S	4.8543	3.3591	3.7981
OSUM-EChat	4.7778	**3.5431**	4.0294
GPT-Audio-1.5	4.6018	3.4941	4.1052
ParaBridge	–	–	–
BLSP-Emo	–	–	–
GLM-4-Voice	4.7901	3.4471	3.8609
+ OPD (Ours)	4.8019	3.4609	3.9187
Fun-Audio-Chat	4.3921	3.3716	3.8618
+ OPD (Ours)	4.4375	3.4348	3.9430

F Audio Quality and Reply Length

The replies do not sound worse after training. Table 12 gives the three signal-quality scores for every system in Table 2. Between a backbone and the same backbone after our training, all three move by less than 0.1 in every case: on Fun-Audio-Chat they rise by 0.045, 0.063 and 0.081, and on GLM-4-Voice by 0.012, 0.014 and 0.058. The emotion and expressiveness columns move by far more over the same pairs, so what our training changes is how the reply is delivered, not how clean it is. The scores also separate the systems in Table 2 differently from the judge-based columns: OpenS2S has the second-highest NISQA and the lowest vocal expressiveness of any system in the table. These predictors measure the recording and the timbre, which is all we use them for here.

No system buys its scores by talking more. Text judges are known to favour longer answers (Zheng et al., 2023), so we checked whether the EchoMind columns could be a length effect. Inside each system we correlate the word count of a reply with its C1–C4 mean over the same 625 items, and in every system that answers the English items in English the correlation is negative: -0.43 for Qwen2.5-Omni, -0.29 for GPT-Audio-1.5, -0.24 for ParaBridge, -0.20 for GLM-4-Voice, and no system above zero. Splitting each system's replies into thirds by length gives the same picture, with the shortest third scoring highest in nearly every case. Across systems, the correlation between median reply length and mean score is only $+0.32$ and is carried by a single point, ParaBridge, whose replies are the longest in the table and whose C1–C4 mean is the highest; removing it leaves no relationship. Length therefore does not explain the ordering, and in particular does not explain the gap between ParaBridge and the backbone it comes from.

One qualifier attaches to the within-system numbers: two systems, Kimi-Audio and OSUM-EChat, answer the English items largely in Chinese, so counting words by whitespace returns a median of one per reply for them and their correlations are meaningless. They are omitted from the analysis rather than reported as 0, which is why the claim above is stated for the systems that answer in English.

G The Affect-Condition Route and the Preference-Training Runs

The affect condition. The route of Section 1 puts the user's vocal state into the context as an explicit, replaceable block, and the block has to be produced by something. We take a small module and train it once on human labels: for every training utterance we have the affect a listener perceives in it (Appendix B), so a readout trained to predict that label from the frozen audio representation learns to report what a person would hear rather than what the synthesis was asked for. The readout is trained on 48,346 rows with a held-out 512, on 33 language-specific labels, 17 Chinese and 16 English, under two heads.

Table 13: The affect readout. The validation set withholds transcripts but not voices, so the numbers describe generalization to unseen wording, not to unseen speakers. The two columns are the two training objectives tried; the one used in the route is the right-hand column.

	Without adversary	With adversary
Fine-grained labels, all	0.9277	0.9037
Fine-grained, Chinese	0.9375	0.8906
Fine-grained, English	0.3672	0.5313

Table 14: The two preference-training runs of Table 3. The row named *SFT+PO-t* places the loss on the text of each reply alone, following the practice discussed in Section 1.

	SFT+PO	SFT+PO-t
Preference items	34,305	34,305
Epochs / optimizer steps	3 / 1,611	3 / 1,611
Global batch	64	64
Trainable parameters	45.7M	45.7M
What is scored	text and speech tokens	text tokens only
The reply's speech in the sequence	present	removed

Two things about Table 13 matter for the argument that follows. The first is that the readout is accurate enough to be a usable condition and not accurate enough to be the whole story: on unseen wording its macro-F1 is about 0.90 over 33 classes, while the underlying representation is not the limitation (Appendix A reads four coarse classes out of the same audio at 0.855). The second is the English column, which is where the labeling is hardest: the errors are adjacent values inside one affect family rather than failures across families, which is why the four-family reading of the same predictions stays above 0.98.

How the condition enters. The readout emits a vector per affect dimension, which a shared zero-initialized projection turns into four tokens. Those tokens are inserted into the audio span of the context, after the semantic audio tokens and before the end-of-audio marker, so that they are positioned as part of what the model hears rather than as instructions it is given. Zero initialization is what makes the starting point meaningful: before training, the projection contributes nothing.

Why the slot has to be a slot. Proposition 1 rests on the two branches of Equation 8 being identical except for the affect condition. Re-recording the same line in a different tone cannot supply that contrast: the change in delivery alters the audio representation everywhere at once, and an effect of the affect becomes indistinguishable from an effect of everything else that moved with it. A block that can be lifted out and swapped whole is what makes the two branches differ in one thing and nothing else. This is the reason the condition is an explicit module rather than an implicit dependence, and it is the sense in which the module is an intervention handle and not a channel for information the model could not otherwise reach.

What the route buys, and what it does not. Proposition 1 says that a policy ignoring the affect condition cannot be optimal for Equation 8, and that the gradient at any such policy has no component along directions which preserve the blindness: the descent direction must leave Π_0. That is an exit. What it does not say is where the descent leads. The condition is an estimate produced by a frozen module reading a frozen representation, so a policy that lowers the loss by reading the condition more sharply is not thereby reading the waveform more sharply; the two can come apart, and in our runs they did, which is why this route was not the one we kept. Equation 7 is the arrival statement the later objective does support, and the comparison is the same in both cases: the input-side route can remove the worst solution, while the label-side route gives the objective a zero that only hearing the audio can reach.

The preference-training runs. The training configuration of the two runs in Table 3 is given in Table 14. Both start from the released backbone with fresh adapters, and both are trained on the same 34,305 preference items for the same three epochs, so the step count and the data are identical and only the forward path differs.

Table 15: Human ratings. H-Emp averages all twenty items over three listeners per arm; H-Conf is over the conflict set of eight items that at least two of three listeners flagged. The two rates partition a different quantity: among the replies that reached the third question, the fraction scoring at least 4, and the fraction scoring exactly 3. Kimi-Audio has one fewer rated item.

Arm	H-Emp (0–5)	H-Conf (1–5)	Aware $P(Q3 \geq 4)$	Text-only $P(Q3 = 3)$
+ OPD (Ours)	**3.88**	**3.56**	**33.3%**	**41.7%**
Fun-Audio-Chat	3.85	3.38	25.0%	70.8%
GPT-Audio-1.5	3.35	3.12	12.5%	79.2%
Kimi-Audio	3.03	2.61	13.0%	56.5%
Qwen3-Omni-30B	2.93	2.50	0.0%	70.8%
OpenS2S	2.67	2.42	12.5%	45.8%

Both runs are DPO on the preference pairs of Appendix B, with a chosen-reply cross-entropy term, the affect-condition preference term of Equation 8 and a grounding instruction. What separates them is one line: in the second run the preference pairs are scored through the text-only forward path, so the assistant's speech tokens never enter the sequence the loss is computed along. Speech feedback is not merely unweighted there, it is structurally impossible. Nothing else moves: same items, same order, same adapters, same schedule.

Two facts about this axis discipline what it can support. First, seven of the 34,305 preference pairs have chosen and rejected texts that are byte-identical, so both runs receive exactly $\sigma(0)$ on those pairs and they contribute nothing to either. Second, the axis varies the loss, not the data, so a difference between the two runs is a difference made by what the objective reads and not by what it was shown. That is what allows the contrast to be read as evidence about the objective rather than about the corpus, and it is also what keeps this axis separate from the speech-weight sweep: there the sequence is the same in both arms and only the weight on the speech term changes.

H Human Evaluation

What the listeners heard. Every score in this paper up to here was assigned by a model, so we also asked people. The test set is twenty recorded utterances built around the conflict this paper is about: ten sentences, each spoken twice over an identical transcript, once with a delivery that matches the words and once with a delivery that contradicts them, such as a sentence about winning the lottery read in a dejected voice. The recordings are human except for five that were read by a commercial speech agent; none of them comes from the corpus of Appendix B, and none of the sentences overlaps the other evaluation sets, so a listener is never hearing the training distribution.

How it was scored. Each reply was rated by three listeners, drawn from six raters who were split into two groups of three, under three questions. The first asks how empathetic the reply is, on a 0–5 scale, judging content and delivery together, and is answered for every reply. The second is about the stimulus rather than the system: does the user's tone of voice conflict with the words? A pair was kept in the conflict analysis set when at least two of its three listeners said yes. The third question was answered only on the items so kept, and asks how well the reply handles the conflict, on a 1–5 scale, from 1 for a reply that follows a template and ignores both signals, through 2 for one that follows only the voice and 3 for one that follows only the text, to 4 and 5 for replies that carry both. Replies were labelled A–F with the assignment shuffled per question, and the listeners did not know which system produced which. Replies were labelled A–F with the assignment shuffled per question, and the listeners did not know which system produced which. The scale of this study is deliberately small: twenty recordings, six systems and three listeners per reply, and eight items in the conflict set. It is sized to check whether the ordering the automatic judges produce survives a human ear, and not to establish an effect size; the statistical claims in this paper rest on the paired tests over the full item sets (Appendix J).

The two rates in Table 15 are reported together because the average of the third question rewards a particular failure. A reply that answers every item from the words alone, mentioning the voice only when the transcript already implies an emotion, collects a steady mid-scale score and never draws a low one; it is never penalized for missing a conflict it never looked for. Splitting the same

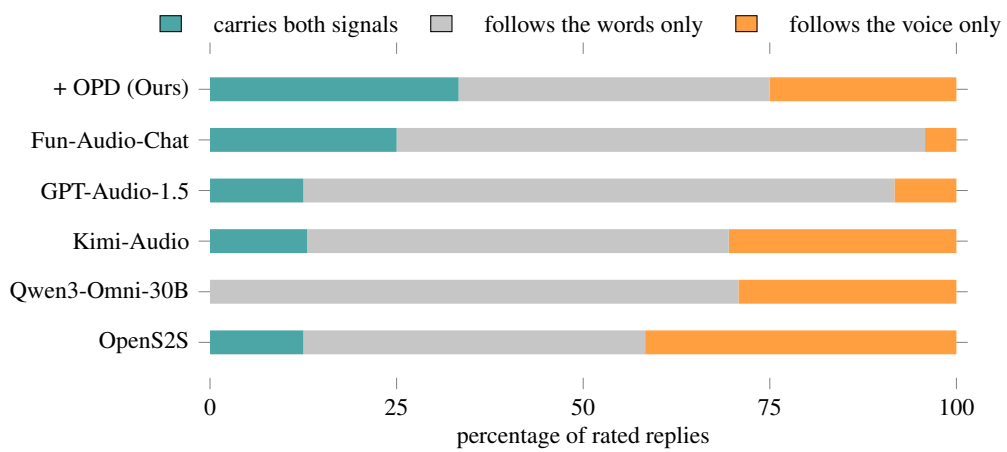

Figure 5: How the six systems handled the conflict items, as judged by listeners. Each bar is the three-listener verdict on the eight items at least two of three listeners flagged as conflicting. The middle segment is the failure the average of the third question hides: a reply that answers from the transcript and never looks at the voice. Our arm is the only one whose aware and text-only segments are of comparable size, and the two systems with the largest voice-only segment are also the two the average rates lowest.

question into the fraction of replies rated as having carried both signals and the fraction rated as having carried only the text separates the two behaviours that the average blends. The distinction is visible in Figure 5 and is what the ordering in Section 5 rests on.

Two exclusions, reported. The conflict set was formed by the listeners themselves rather than fixed in advance, and two pairs that were recorded as conflicting did not clear the bar: on both, one listener called them conflicting and the other two did not. They are therefore absent from the third question and from Table 15. None of the ten pairs recorded as agreeing was flagged, which is the outcome the design expects.

I Data Volume

How the subsets were built. Each level is a nested subset of the same 34,305 rows. They are drawn by stratified allocation rather than uniform sampling: rows are partitioned into (language, affect label) strata, each stratum receives a quota proportional to its share of the full set rounded up, and the quotas are then trimmed one row at a time from whichever stratum the rounding lifted furthest above its exact share. Every stratum keeps at least one row, so no label disappears at the smallest level. Rows are emitted in their original file order, which keeps each subset's shard composition across the eight GPUs structurally like the full run's. Eight specific rows are forced into every level including the smallest, because they are the rows the training step reaches first; at one sixteenth that is 8 of 2,177 rows. A level's size is set by the number of optimizer steps it runs, at $64 \times$ steps per epoch $+ 1$ rows, the same relation as the full run's $34{,}305 = 64 \times 536 + 1$.

The scores barely move. Across a sixteen-fold range of data the scores stay within a fraction of a point of the full-data setting: cut to one sixteenth, the model lands within the spread of the full run on all three benchmarks. What movement there is splits by judge, C4 mildly favoring the smaller subsets and the $p(\geq 4)$ rate mildly favoring the full set; the paired tests behind both directions are in Appendix J.

Why the axis behaves this way? On-policy distillation supervises positions rather than samples. At every position of every rollout the teacher produces a full distribution and the student is asked to match it, and the states that are worth matching recur across the corpus long before the corpus runs out. A few thousand rollouts therefore already cover what there is to learn, and what remains is how quickly the student absorbs it, which is a matter of optimizer steps rather than of rows. This is the shape the method is known to have, and it is why the cost of this approach is the teacher forward passes rather than the data.

Table 16: Five nested subsets of the training data, from the full set down to one sixteenth, all trained with the recipe of the OPD row in Table 2. **Bold** marks the best value in a row; higher is better everywhere. The first two rows are the axis itself, not scores.

	1/16	1/8	1/4	1/2	full
Training rows	2,177	4,289	8,577	17,153	34,305
Optimizer steps	102	201	402	804	1,608
EchoMind C4	**4.4240**	4.4384	4.4032	4.3728	4.3248
EchoMind C1–C4	4.6484	**4.6500**	4.6156	4.6020	4.5916
Vocal expressiveness	3.632	3.683	3.760	3.787	**3.818**
Emotion alignment	35.00	**35.95**	35.78	34.29	34.62
SpeechParaling Chinese	**58.93**	56.96	58.75	54.11	56.25
SpeechParaling English	66.07	66.79	67.14	**72.50**	71.79
VocalBench-zh overall	56.0	54.6	**56.8**	55.2	54.2

Table 17: Paired comparisons on EchoMind. The first block sets the method against the released backbone, the second against the preference-training run that keeps the reply's speech in the training sequence, and the third compares the two preference-training runs with one another. Δ is the mean per-item difference in favour of the first system; the interval is a 95% paired bootstrap over items. p is Holm-corrected within its family, except in the three cells marked †, which are the pre-declared primary comparison of their run and are reported uncorrected.

Comparison	Metric	Δ	95% CI	p
vs. Fun-Audio-Chat	C4	$+0.4416$	$[+0.3520, +0.5296]$	1.1×10^{-20} †
	EmoAlign	$+4.7079$	$[+1.5033, +8.0350]$	5.3×10^{-3}
vs. SFT+PO	C4	$+0.0768$	$[-0.0032, +0.1552]$	0.234
	EmoAlign	$+0.6540$	$[-2.2933, +3.5881]$	1.00
	VES* $p(\geq 4)$	$+0.0416$	$[+0.0016, +0.0816]$	0.044 †
SFT+PO-t vs. SFT+PO	VES* $p(\geq 4)$	$+0.1056$	$[+0.0720, +0.1408]$	4.2×10^{-9} †

J PAIRED TESTS

What is tested and how. Each comparison pairs a reply with the reply another system gave to the same item, so the item is its own control and item difficulty cancels out. Continuous quantities are compared with a Wilcoxon signed-rank test on the per-item differences, and a paired percentile bootstrap over 10,000 resamples of the item pairs at a fixed seed gives the interval. Binary quantities, here the rate of replies scoring at least 4 on vocal expressiveness, use an exact McNemar test on the discordant pairs. Multiplicity is corrected by Holm within each metric family; the one comparison that a run declared in advance as its primary is excluded from the correction and is marked below. The tests read only the per-item scores the scorers had already written, so no model and no judge was re-run to produce them.

The four subsets do not move the two families in the same direction. C4 rises as data is removed, and the intervals for quarter, eighth and sixteenth all clear zero in that direction; the $p(\geq 4)$ rate falls, and the intervals for eighth and sixteenth clear zero the other way. Both effects are small in absolute terms, which is what the flat span of the main text refers to, and neither is large enough to make a smaller subset the better setting.