Diagnose, Then Refine: A Closed-Loop TTS System with AudioLLM-Guided Correction

Zeyang Song[1,2], Tianchi Liu[3,*], Tianrui Wang[4]
Chenglin Xu[3], Yiwen Guo[5], Haizhou Li[6,7,*]

[1]National University of Singapore [2]Tencent [3]LIGHTSPEED
[4]Nanyang Technological University [5]Independent Researcher
[6]The Chinese University of Hong Kong, Shenzhen [7]Shenzhen Loop Area Institute

Abstract

Current TTS systems typically rely on open-loop, single-pass generation and can produce sporadic local prosodic defects, such as misplaced stress, unnatural pauses, or flattened intonation, that utterance-level metrics often fail to expose. We present **LOOPTTS**, a judge-guided Filter–Judge–Refiner framework for recovering low-quality TTS outputs diagnosed by an AudioLLM. Given an initial utterance from a base TTS model, an AudioLLM Judge identifies salient prosodic issues and generates structured refine instructions; a Refiner, our fine-grained instruction-following TTS model, then performs guided expressive re-synthesis conditioned on the initial utterance, target text, and instruction. To train the Refiner, we construct REFINER-DB, a ∼42K-example AudioLLM-annotated dataset with word-level prosodic weak supervision. Human evaluation on diagnosed low-quality utterances shows that LOOPTTS can detect perceptually salient errors and correct them with the Refiner, outperforming raw generated audio and practical open-loop re-generation baselines in recovery quality. The Refiner also demonstrates stronger instruction-following ability for stress and pause control in targeted prosody modification. LOOPTTS is available online.[1]

1 Introduction

The emergence of neural audio codecs has enabled a new generation of large-language-model-based text-to-speech (LLM-based TTS) systems (Wang et al., 2023; Chen et al., 2024; Du et al., 2024a; Anastassiou et al., 2024; Chen et al., 2025b; Wang et al., 2024b) that achieve near-human naturalness. Building on this foundation, controllable TTS has progressed rapidly, spanning emotion control (Yang et al., 2025a; Cho et al., 2024; Liu et al., 2026; Wang et al., 2026b) and instruction-following (Guo et al., 2023; Yang et al., 2024; Zhou et al., 2024; Du et al., 2024b), broadening the scope of speech generation.

Despite this progress, achieving consistently high-quality generation at scale remains an unsolved challenge. Most deployed TTS pipelines remain open-loop: after one generation pass, they either accept the audio or regenerate it from scratch. Due to the stochasticity of autoregressive and flow-based decoding, even strong systems can produce sporadic local prosodic defects, such as misplaced emphasis, contextually inappropriate pauses, and flattened intonation, particularly in complex linguistic contexts. These defects are subtle yet consequential; for instance, stressing a preposition ("the book IS on the table" instead of "the BOOK is on the table") can make an otherwise intelligible utterance sound unnatural. Conventional objective metrics such as Word Error Rate (WER), UTMOS, and speaker similarity are useful for content, audio quality, and identity, but they operate at the utterance level and only indirectly reflect such fine-grained prosodic failures. The central question we address is therefore: *how can we move beyond single-pass, open-loop TTS toward a system that integrates automatic prosodic diagnosis with bounded corrective generation?*

Existing attempts toward closing this loop remain limited. On the *generation side*, the default solution for a prosodically defective utterance is full re-synthesis with a different random seed, an untargeted strategy that may fix the problem but may also reproduce it or change other attributes. Speech editing methods (Peng et al., 2024; Wang et al., 2024a) can modify existing audio, but primarily address *content*-level replacement or insertion and do not expose controls such as "add emphasis to word X." On the *evaluation and alignment side*, recent work has introduced AudioLLM-based judges for preference alignment (Zhang et al., 2024;

*Corresponding authors.
[1]https://github.com/Pooookeman/LoopTTS

Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing, pages 28041–28059
October 24-29, 2026 ©2026 Association for Computational Linguistics

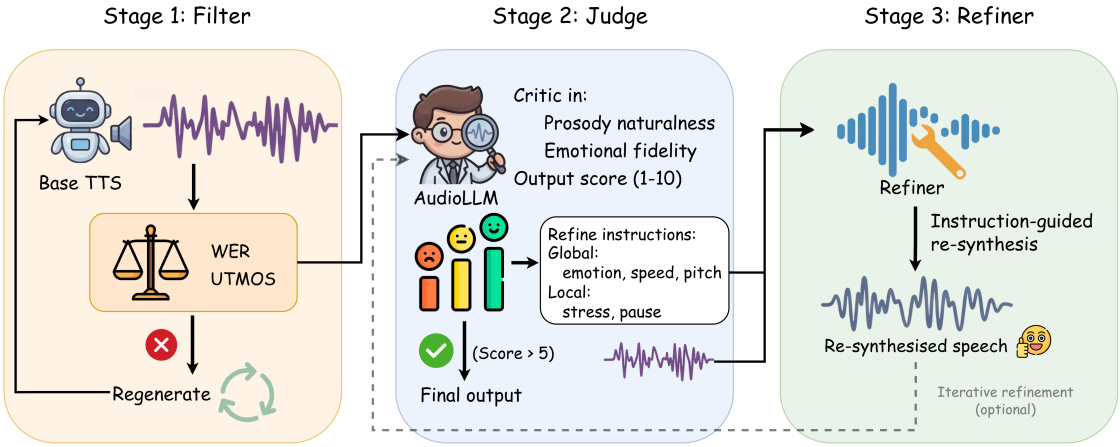

Figure 1: Overview of the LOOPTTS pipeline. **Stage 1 (Filter):** coarse-grained metrics (WER, UTMOS) discard catastrophically degraded outputs and trigger full re-generation. **Stage 2 (Judge):** an AudioLLM evaluates prosody naturalness and emotional fidelity, producing an overall score (1–10) along with structured refine instructions that specify both global attributes (emotion, speed, pitch) and local operations (stress, pause). Utterances meeting the threshold are accepted as final output. **Stage 3 (Refiner):** the Refiner takes the initial utterance together with the refine instructions and performs guided expressive re-synthesis, producing corrected speech with optional iterative refinement for residual defects.

Hussain et al., 2025a; Xia et al., 2025); however, these approaches usually collapse diagnostic detail into scalar preference signals, discarding information needed for corrective generation, such as which word should carry stress or where a pause is missing. We therefore study whether the judge can participate in the correction loop by producing executable instructions for a separate re-synthesis model.

To address this, we present **LOOPTTS**, a TTS system that places an AudioLLM Judge at its core and orchestrates two functionally complementary TTS models for judge-guided corrective generation (Figure 1). Specifically, we construct a three-stage **Filter–Judge–Refiner** architecture: a base TTS model first produces an initial utterance; the Judge identifies salient word-level prosodic issues and emits structured refine instructions; finally, the Refiner, our proposed fine-grained instruction-following TTS model, takes the initial utterance together with the refine instructions and performs guided expressive re-synthesis. This design is not waveform-local editing: unrequested regions and speaker embeddings can change, and we explicitly evaluate this trade-off. Training such a model requires word-level prosodic annotations that no existing dataset provides. We therefore leverage AudioLLM-based contrastive annotation to construct REFINER-DB, a ∼42K-utterance dataset with fine-grained but weak prosodic supervision. Through this judge-centric, multi-model mechanism, LOOPTTS studies how bounded test-time diagnosis and refinement can improve selected low-quality TTS outputs. Our contributions are as follows:

- We propose LOOPTTS, a judge-centric, multi-model collaborative TTS system that instantiates a closed-loop alternative through a three-stage Filter–Judge–Refiner architecture.
- We validate that AudioLLMs can provide scalable high-salience prosodic weak labels, and leverage this capability to construct REFINER-DB, a ∼42K-example dataset with word-level prosodic supervision.
- We introduce the Refiner, a fine-grained instruction-following TTS model trained with a position-aware objective that focuses supervision on instruction-specified prosodic positions, enabling guided expressive re-synthesis of flagged prosodic defects in our LoopTTS.

2 Related Work

Controllable and expressive TTS. Recent LLM-based TTS systems (Chen et al., 2024, 2025b; Wang et al., 2024b; Hussain et al., 2025b; Chen et al., 2025a; Cui et al., 2025; Ye et al., 2025; Zhou et al., 2025) have established a dominant backbone for high-quality speech synthesis. Built

on these backbones, controllable TTS (Xie et al., 2025) has progressed along several axes: emotion control (Cho et al., 2024; Wu et al., 2024; Yang et al., 2025a; Li et al., 2025a), natural-language description (Guo et al., 2023; Yang et al., 2024), and instruction-guided synthesis (Zhou et al., 2024; Du et al., 2024b). In parallel, speech editing methods (Peng et al., 2024; Wang et al., 2024a) enable post-hoc modification of existing audio by replacing or inserting words at the content level. However, none of these methods can accept prosodic-level refine instructions to refine an already-generated utterance.

LLM-as-Judge for TTS quality optimization. AudioLLMs are increasingly used to improve TTS quality. On the evaluation side, recent LLM-as-a-Judge models (Zheng et al., 2023) such as AudioJudge (Manakul et al., 2025) and SpeechJudge (Zhang et al., 2025) can perceive fine-grained acoustic details and prosody to assess speech naturalness, approaching human-level judgment. Because human preference data is expensive to collect and inter-annotator agreement is often low, a growing body of work adopts these LLM judges to provide preference signals for TTS post-training: RLAIF-SPA (Yang et al., 2025b) uses LLM judgments of semantic accuracy and prosody-emotion alignment as RLAIF rewards, and Step-Audio-EditX (Yan et al., 2025) applies reinforcement learning with LLM-derived rewards for expressive speech editing. Agent-based frameworks such as DialogueAgents (Li et al., 2025b) modify text or sentence-level emotional expression through multi-agent coordination. However, these methods do not focus on detecting prosodic defects or performing fine-grained refinement of an already-generated utterance.

3 The LOOPTTS Framework

This section presents the LOOPTTS framework: a three-stage inference pipeline for iterative refinement (Section 3.1) and a contrastive data construction paradigm for training the Refiner (Section 3.2). The AudioLLM capabilities underlying both are validated in Section 5.

3.1 Three-Stage Quality Assurance Pipeline

The pipeline is designed as a cascade that operates on utterances produced by a base TTS model, where each stage narrows the set of utterances requiring expensive processing (Figure 1): coarse filtering removes catastrophic failures at negligible cost, AudioLLM diagnosis evaluates the remainder, and the Refiner re-synthesizes only the flagged subset. This makes LOOPTTS an offline quality-assurance pipeline rather than a streaming TTS model.

Stage 1: Coarse-grained filtering. Given target text T, the base TTS model generates y_{init}. Stage 1 removes severe content or quality failures using WER and UTMOS (Saeki et al., 2022): an utterance passes only when Whisper-large-v3-turbo yields WER$\leq 3.0\%$ against T and UTMOS>3.0. Otherwise, Stage 1 performs up to two additional re-generation attempts with different random seeds. Utterances that still fail are excluded from later stages; the rest proceed to Stage 2.

Stage 2: AudioLLM prosodic diagnosis. For each retained utterance, an AudioLLM Judge scores prosodic naturalness and emotional fidelity on a 1–10 scale. Utterances scoring at least $\tau_{\text{judge}}=5$ are accepted as-is; lower-scoring utterances are flagged and assigned a refine instruction I covering global style (emotion, speed, pitch) and local prosody (stress, pause). The concrete Judge choices are validated in Section 5 and specified in the experimental setup. Full prompts are provided in Appendix I.

Stage 3: Guided expressive re-synthesis. The Refiner re-synthesizes flagged utterances conditioned on the initial audio, refine instruction, and target text: $y_R = \text{Refiner}(y_{\text{init}}, I, T)$. This is guided expressive re-synthesis rather than waveform-level local editing, so unrequested regions and speaker embeddings can change; we therefore evaluate speaker similarity and the resulting expressiveness–similarity trade-off. The output can be fed back to Stage 2 for bounded iterative refinement.

3.2 Contrastive Instruction Data Construction

Training the Refiner requires word-level prosodic supervision, which is expensive to label manually. We therefore construct REFINER-DB through contrastive AudioLLM annotation (Figure 2): each expressive recording y_{tgt} is paired with a neutral synthetic counterpart y_{neu} of the same text and speaker, and the AudioLLM extracts salient high-contrast prosodic differences between them. Global style metadata and local AudioLLM annotations

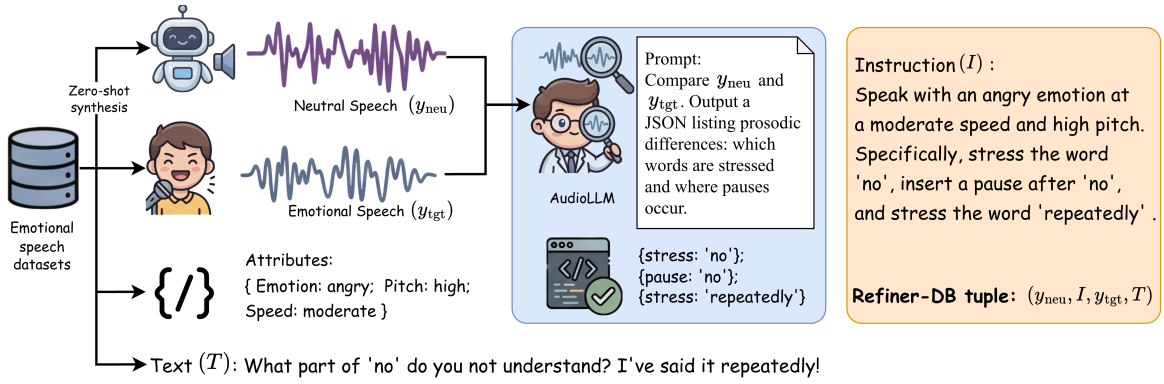

Figure 2: Contrastive data construction for REFINER-DB. Each expressive recording y_{tgt} is paired with a zero-shot neutral clone y_{neu} synthesized from the same text and speaker with neutral emotion. Global style comes from dataset metadata; an AudioLLM compares the two utterances and outputs word-level prosodic differences (stress, pause) as structured JSON. Both sources are merged into a natural-language refine instruction I, yielding tuples $(y_{\text{neu}}, I, y_{\text{tgt}}, T)$. Full prompts are in Appendix I.

are merged into a refine instruction I, yielding training tuples $\mathcal{D} = \{(y_{\text{neu}}, I, y_{\text{tgt}}, T)\}$.

This procedure yields ∼42K weakly annotated contrastive tuples. At inference time, y_{init} plays the same role as an utterance to be improved. Dataset composition and filtering details are provided in Appendix B, and full AudioLLM prompts are provided in Appendix I.

4 Refiner: Architecture and Training

The Refiner must execute refine instructions from the AudioLLM Judge, yet standard training objectives lack the inductive bias to focus on the sparse, prosodically critical positions that carry the refine signal. In a typical utterance, only 10–30 out of 200+ codec tokens correspond to instruction-specified stress or pause positions, making it difficult for uniform objectives to supervise these regions. We address this with a position-aware training objective (Figure 3): token-level loss reweighting that amplifies gradients at instruction-specified positions (Section 4), and span-level relational alignment that encourages internal coherence within stress spans (Section 4.3).

4.1 Architecture

Our Refiner is built on EmoVoice (Yang et al., 2025a), a codec language model for emotion-controllable TTS that autoregressively predicts 50 Hz CosyVoice (Du et al., 2024a) semantic tokens, decoded into waveforms via flow matching and HiFi-GAN (Kong et al., 2020). We initialize from the pretrained checkpoint and fine-tune on REFINER-DB. The input sequence is the concatenation $(I,\ y_{\text{init}},\ T)$: placing the refine instruction I first ensures that it conditions all subsequent processing under causal attention; the initial utterance y_{init} precedes the target text T to provide a concrete acoustic reference.

4.2 Position-Weighted Cross-Entropy Loss

Under standard cross-entropy (CE) loss, all tokens contribute equally to the gradient, leaving the learning signal at prosodically critical positions diluted by the majority of unchanged tokens. Our Position-Weighted CE loss assigns higher weights to instruction-specified stress and pause positions, forcing the model to focus on these sparse but decisive regions.

Prosody span identification. We use Whisper word-level timestamps[2] to locate the codec token spans corresponding to each stressed word and each pause position specified in the refine instruction. These prosodic spans, i.e., the regions where the Refiner must deviate from the input, serve both the Position-Weighted CE (all spans) and the Structural Alignment Loss (stress spans only; Section 4.3). We encode them as a per-token position label $m_t \in \{0, 1, 2\}$:

$$m_t = \begin{cases} 1 & \text{if position } t \text{ falls within a stress span,} \\ 2 & \text{if position } t \text{ falls within a pause span,} \\ 0 & \text{otherwise.} \end{cases}$$

[2] https://github.com/linto-ai/whisper-timestamped

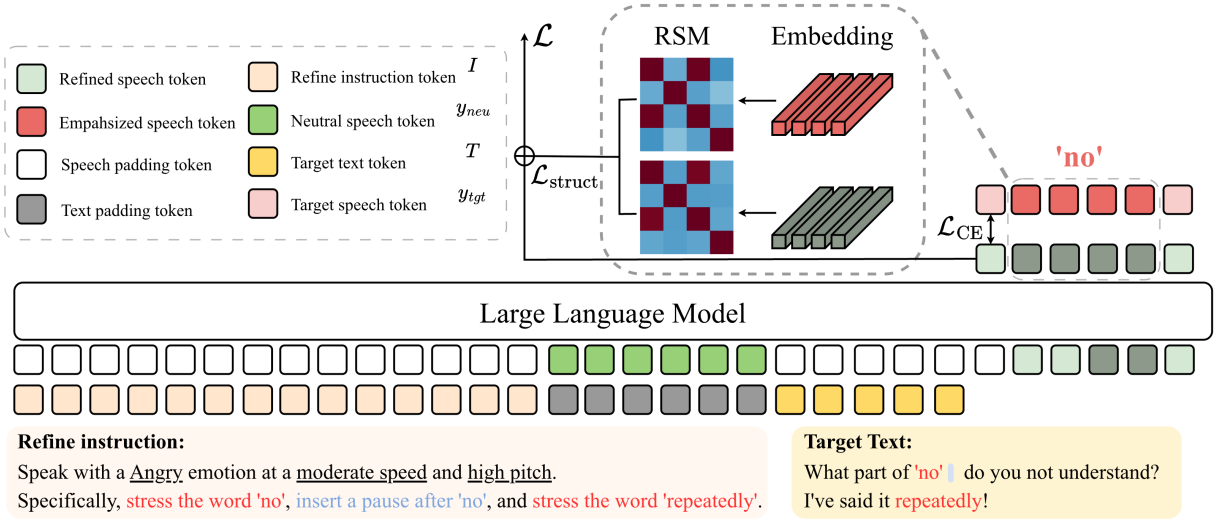

Figure 3: Refiner architecture and position-aware training objective. The Refiner takes a refine instruction I, the initial utterance y_{init}, and the target text T as input. Two losses are applied in training: a position-weighted CE loss $\mathcal{L}^{\text{CE}}$ that amplifies gradients at stressed and pause positions, and a structural alignment loss $\mathcal{L}_{\text{struct}}$ that aligns span-level consistency.

Position weighting. Each token receives a weight determined by its prosodic role:

$$w_t = 1 + \alpha \cdot \mathbb{1}[m_t{=}1] + \beta \cdot \mathbb{1}[m_t{=}2] \quad (1)$$

Stress and pause tokens thus receive mild upweighting relative to the unit baseline, encouraging the model to attend to the annotated local edits without overwhelming global speech modeling. The Position-Weighted CE loss is:

$$\mathcal{L}^{\text{CE}} = \frac{\sum_t \text{CE}(\hat{y}_t,\ y_t) \cdot w_t \cdot v_t}{\sum_t w_t \cdot v_t} \quad (2)$$

where $v_t = \mathbb{1}[y_t \neq \text{pad}]$ masks padding positions. In practice we set $\alpha{=}0.2$ and $\beta{=}0.3$, a gentle boost that prioritizes prosodic positions without destabilizing the generation of other attributes (emotion, pitch, speed).

4.3 Position-Aware Structural Alignment Loss

The Position-Weighted CE optimizes each token independently, but stress realization is inherently span-level, spanning 5–20 contiguous tokens. Token-wise optimization may predict each position correctly in isolation yet fail to capture the internal structure of a stress span. To encourage span-level consistency, we introduce a structural alignment loss based on Representational Similarity Matrices (RSMs) (Park et al., 2019; Kornblith et al., 2019), aligning pairwise token relations within each stress span between prediction and target. (Pause spans are excluded because inserted pauses consist almost entirely of silence tokens, whose trivial internal structure does not benefit from relational alignment.)

RSM construction and alignment. Using the prosodic spans identified in Section 4, let $\mathcal{P} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots\}$ denote the set of all stress spans (maximal contiguous regions where $m_t = 1$). For each span $\mathcal{S} = [s, e)$, let $\mathbf{H}_{\mathcal{S}} \in \mathbb{R}^{|\mathcal{S}| \times D}$ denote the acoustic-code embeddings within the span. After row-wise ℓ_2 normalization ($\bar{\mathbf{H}}_{\mathcal{S}}$), the RSM is:

$$\mathbf{R}_{\mathcal{S}} = \bar{\mathbf{H}}_{\mathcal{S}} \bar{\mathbf{H}}_{\mathcal{S}}^{\top} \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{S}|} \quad (3)$$

where each entry is the cosine similarity between two positions in the span. During teacher-forced training, we construct $\mathbf{R}_{\mathcal{S}}^{\text{pred}}$ from the model hidden representations at acoustic-code positions and $\mathbf{R}_{\mathcal{S}}^{\text{tgt}}$ from the corresponding target acoustic-code embeddings, making the loss an embedding-level alignment objective. We then minimize the squared Frobenius distance averaged over all spans:

$$\mathcal{L}_{\text{struct}} = \frac{1}{|\mathcal{P}|} \sum_{\mathcal{S} \in \mathcal{P}} \frac{1}{|\mathcal{S}|^2} \left\| \mathbf{R}_{\mathcal{S}}^{\text{pred}} - \mathbf{R}_{\mathcal{S}}^{\text{tgt}} \right\|_F^2 \quad (4)$$

Dividing by $|\mathcal{S}|^2$, the number of entries in the RSM, ensures scale invariance across spans of different lengths. When $\mathcal{P} = \emptyset$, the loss is set to zero.

The overall training objective combines both terms:

$$\mathcal{L} = \mathcal{L}^{\text{CE}} + \lambda \cdot \mathcal{L}_{\text{struct}} \quad (5)$$

AudioLLM	Regular	Emotional
Gemini-3-flash (Team et al., 2023)	3.1	3.2
Gemini-3-pro (Team et al., 2023)	**3.3**	**3.4**
Kimi-Audio (KimiTeam et al., 2025)	2.2	2.5
Qwen3-Omni (Xu et al., 2025)	2.6	2.7
Qwen3-Omni Thinking (Xu et al., 2025)	2.9	3.0

Table 1: Mean human–AudioLLM alignment scores (1–4). Higher is better.

AudioLLM	Stress		Pause	
	Rec.	F1	Rec.	F1
Gemini-3-flash	0.861	0.535	0.573	0.432
Gemini-3-pro	**0.884**	**0.673**	**0.697**	**0.532**
Kimi-Audio	0.588	0.438	0.440	0.316
Qwen3-Omni	0.613	0.441	0.487	0.330
Qwen3-Omni Thinking	0.667	0.465	0.478	0.325

Table 2: AudioLLM annotation quality (micro Recall and F1) against professional human labels. The labels mark salient stress and pause differences rather than exhaustive prosodic structure.

We set λ=0.1 in all experiments.

5 AudioLLM Evaluation

Before deploying AudioLLMs in our pipeline, we validate two capabilities: (1) *diagnosing*, whether their inference-time diagnoses align with human perception (Section 5.1), and (2) *annotation*, whether they can extract salient prosodic labels for training data construction (Section 5.2). We benchmark five AudioLLMs on evaluation data held out from the training set, sourced from EmoVoice (Emotional) and LibriTTS (Regular). Detailed evaluation setups are provided in Appendix G.

5.1 Diagnostic Capability

Each AudioLLM diagnoses prosodic defects in 100 TTS-generated utterances (50 Regular, 50 Emotional) and generates refine instructions in the same format as Stage 2 (prompt in Appendix I). Six non-technical evaluators with fluent English listening proficiency then rate whether each diagnosis aligns with their own judgment on a 1–4 scale (4: fully aligned, 3: mostly aligned, 2: partially aligned, 1: not aligned).

Results (Table 1) show that Gemini-3-pro achieves the highest alignment among the evaluated AudioLLMs on both subsets (3.3 Regular, 3.4 Emotional). We therefore use it as the main Judge, while treating its diagnoses as fallible signals rather than ground truth.

5.2 Annotation Capability

We assess whether AudioLLMs can identify salient prosodic differences between neutral and expressive speech, the core capability required for constructing REFINER-DB. Professional annotators label prominent stress and pause positions on 200 neutral–expressive utterance pairs; each AudioLLM independently annotates the same pairs (prompt in Appendix I) and is evaluated against these labels using micro recall and F1. This is a weak-labeling evaluation: the goal is to obtain scalable high-salience supervision, not exhaustive word-level prosody annotation.

Results (Table 2) show that Gemini-3-pro achieves the highest recall and F1 on both stress and pause, making it the best annotator among the tested AudioLLMs for the large-margin contrasts used in our data construction pipeline (Section 3.2). The absolute F1 values remain moderate, especially for pause (0.532), so we do not interpret the labels as precise or exhaustive word-level ground truth. We adopt Gemini-3-pro for REFINER-DB because it provides useful scalable weak supervision, and we later evaluate final outputs with human listeners rather than using these labels alone.

6 Experiments

6.1 Metrics

We evaluate the content consistency of synthesized speech using WER, with transcriptions obtained from Whisper-large-v3-turbo (Radford et al., 2023). Speaker identity preservation is measured by speaker similarity (SIM), the cosine similarity between WavLM-based speaker embeddings of the output and the reference audio[3]. We also report UTMOS (Saeki et al., 2022) as an automatic speech quality estimator. For MOS-style subjective evaluation, ten professional paid evaluators with fluent English listening proficiency independently score each sample in a blind shuffled test. MOS rates overall naturalness and audio quality (1–5), while MOS-I rates instruction-following fidelity across five dimensions (emotion, pitch, speed, stress, pause), each on a 1–5 scale. When per-evaluator means are available, subjective scores are reported with 95% confidence intervals (subscript $\pm$) computed from the t-distribution. For paired evaluation, the same evaluator pool performs blind A/B comparisons over 30 flagged samples

[3] https://github.com/BytedanceSpeech/seed-tts-eval

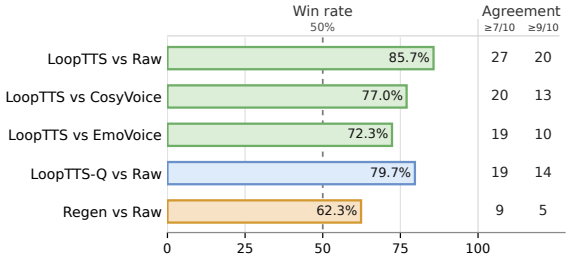

Figure 4: Blind paired preference evaluation on flagged utterances. Each paired comparison uses 30 samples.

per comparison; we report the preferred system, preference rate, and the number of samples with at least 7/10 and 9/10 evaluator agreement. Scoring rubrics and reliability statistics are provided in Appendices H and D.4.

6.2 LoopTTS Pipeline Evaluation

We evaluate the full LOOPTTS pipeline on CosyVoice2-generated speech. We sample 100 Stage 2 flagged utterances (score < 5) per condition (Neutral and Emotional, 200 total) for human evaluation. We choose this selected failure subset for two reasons: only a small fraction of the full set is routed to Stage 3 refinement, so full-distribution MOS would be dominated by already accepted utterances and make recovery gains harder to distinguish; and full-distribution MOS over all systems would require substantially more human ratings. Appendix A reports full-pipeline stage accounting and objective metrics over 3,000 utterances.

Table 3 compares raw flagged utterances, budget-aligned no-target re-generation, and closed-loop refinement. Re-generation uses the same initial output and Stage 2 Judge trigger as LOOPTTS, but spends the one-correction budget on untargeted full re-synthesis rather than the instruction-conditioned Refiner. The Global-only control retains emotion, speed, and pitch instructions while removing word-level stress and pause cues. LOOPTTS-Q keeps the Gemini-annotated Refiner unchanged and replaces the entire inference-time Judge with Qwen3-Omni Thinking (Xu et al., 2025), including both diagnosis and instruction generation. Figure 4 reports win rate for the first-listed system and agreement at 7/10 and 9/10 listeners.

Table 3 shows that a single LOOPTTS refinement round improves MOS over both raw flagged audio and the strongest re-generation baseline under Neutral and Emotional conditions. Removing local stress and pause cues reduces MOS from 4.17/4.01 to 3.96/3.72 while WER and SIM remain comparable, supporting the contribution of position-specific instructions beyond global style control. Figure 4 supports the recovery result: listeners prefer LOOPTTS over raw flagged audio (85.67%), CosyVoice2 re-generation (77.00%), and EmoVoice re-generation (72.3%). For the latter comparison, 19/30 samples receive at least 7/10 listener agreement and 10/30 receive at least 9/10 agreement. Because the re-generation baselines use the same judge-triggered one-correction budget, these results support targeted guided expressive re-synthesis over a no-target retry strategy.

The held-out Judge result further shows that the framework is not tied to Gemini at inference time. LOOPTTS-Q replaces Gemini-3-pro with Qwen3-Omni Thinking for both diagnosis and instruction generation, yet obtains MOS scores of 4.09/3.98 under Neutral/Emotional conditions and remains preferred over raw flagged audio in paired evaluation (79.67%). Because our Refiner performs guided expressive re-synthesis rather than local waveform editing, SIM reduction is an expected trade-off when prosody or emotion is strengthened. Similar trade-offs have also been observed in expressive TTS studies (Wang et al., 2026a; Cho et al., 2025).

6.3 Instruction-Following Comparison

This experiment isolates the Refiner's instruction-following capability outside the full cascade. Human evaluators use MOS-I to judge whether each requested control–emotion, pitch, speed, stress, or pause–is executed naturally without introducing harmful unintended changes. We compare against CosyVoice2 (Du et al., 2024b), CosyVoice2-marker, and EmoVoice (Yang et al., 2025a). All systems receive the same instruction I and text T; only the Refiner additionally conditions on the initial utterance y_{init}. Appendix E provides the baseline setup details and limitations.

Results are shown in Table 4. The Refiner achieves the highest MOS-I on Pitch, Speed, Stress, and Pause under this protocol, indicating that fine-tuning on REFINER-DB improves execution of the annotated prosodic controls. The slight MOS gap relative to EmoVoice is largely due to occasional less vivid emotional delivery, suggesting that the Refiner is better suited as a Stage 3 corrective model than as a standalone emotional TTS generator.

Condition	Neutral			Emotional		
	WER↓	SIM↑	MOS↑	WER↓	SIM↑	MOS↑
Baseline						
Before refinement	2.04	**0.73**	$3.21_{\pm.15}$	2.11	0.69	$3.01_{\pm.16}$
Re-generation (open-loop)						
CosyVoice2 (Du et al., 2024b)	1.95	0.71	$3.45_{\pm.14}$	2.03	**0.70**	$3.15_{\pm.15}$
EmoVoice (Yang et al., 2025a)	1.83	0.68	$3.90_{\pm.13}$	**1.72**	0.65	$3.85_{\pm.12}$
LOOPTTS *(closed-loop)*						
LOOPTTS (Global-only)	1.87	0.68	$3.96_{\pm.10}$	1.82	0.66	$3.72_{\pm.16}$
LOOPTTS (1-round)	1.86	0.68	$4.17_{\pm.12}$	1.79	0.67	$4.01_{\pm.14}$
LOOPTTS-Q (1-round)	1.90	0.67	$4.09_{\pm.16}$	1.84	0.66	$3.98_{\pm.15}$
LOOPTTS (2-round)	1.85	0.66	$4.16_{\pm.13}$	1.80	0.64	$4.17_{\pm.13}$
Human-instruction	**1.79**	0.67	$\mathbf{4.20}_{\pm.14}$	1.78	0.67	$\mathbf{4.26}_{\pm.12}$

Table 3: Pipeline recovery evaluation on Stage 2 flagged utterances under Neutral and Emotional conditions. LOOPTTS uses Gemini-3-pro as the inference-time Judge, while LOOPTTS-Q uses the same Refiner with the inference-time Judge changed to Qwen3-Omni Thinking.

Model	Objective			Subjective					
	WER↓	SIM↑	UTMOS↑	MOS↑	MOS-I↑				
					Emo.	Pitch	Speed	Stress	Pause
CosyVoice2†	–	**0.72**	3.40	$2.21_{\pm.11}$	$2.25_{\pm.12}$	$2.31_{\pm.10}$	$2.54_{\pm.09}$	$2.15_{\pm.10}$	$2.12_{\pm.09}$
CosyVoice2-marker	4.74	0.69	3.27	$2.95_{\pm.16}$	$3.02_{\pm.11}$	$3.78_{\pm.12}$	$3.46_{\pm.09}$	$4.32_{\pm.07}$	$3.88_{\pm.08}$
EmoVoice	**2.81**	0.67	**3.86**	$\mathbf{4.32}_{\pm.13}$	$\mathbf{4.13}_{\pm.11}$	$3.45_{\pm.09}$	$3.76_{\pm.08}$	$3.83_{\pm.08}$	$2.92_{\pm.09}$
Refiner	2.90	0.68	3.60	$4.21_{\pm.14}$	$4.04_{\pm.10}$	$\mathbf{4.01}_{\pm.10}$	$\mathbf{4.13}_{\pm.08}$	$\mathbf{4.57}_{\pm.07}$	$\mathbf{4.28}_{\pm.08}$

Table 4: Instruction-following comparison. †CosyVoice2 frequently reads the instruction text verbatim, making its WER not comparable.

Variant	WER↓	MOS↑	Avg. MOS-I↑
Refiner	2.90	$\mathbf{4.21}_{\pm.13}$	$\mathbf{4.21}_{\pm.09}$
w/o $\mathcal{L}_{\text{struct}}$	**2.78**	$4.06_{\pm.15}$	$4.01_{\pm.09}$
w/o Position-Weighting	2.92	$4.17_{\pm.14}$	$4.13_{\pm.08}$
w/o Both	2.87	$4.03_{\pm.14}$	$3.75_{\pm.10}$

Table 5: Ablation: effect of loss components.

6.4 Ablation Studies

We ablate Position-Weighting and Structural Alignment to isolate their individual contributions.

Results (Table 5) suggest that the two loss components help instruction-following, but the evidence should be interpreted cautiously. Removing $\mathcal{L}_{\text{struct}}$ causes the largest Avg. MOS-I drop, consistent with stress realization benefiting from span-level relational alignment. Removing position weighting produces a smaller but consistent MOS/MOS-I decrease, suggesting that token-level emphasis on annotated spans is useful. However, WER is slightly better for some ablated variants and the absolute MOS gaps are modest, so these ablations support the losses as helpful design choices rather than uniquely isolating the mechanism behind all gains. The separate audio-conditioning ablation in Appendix D.1 shows that removing y_{init} only slightly affects MOS but reduces SIM and MOS-I, indicating that the initial utterance helps preserve speaker/prosodic identity and avoid unnecessary edits.

Notably, the full Refiner exhibits a minor WER increase compared to ablated variants. This reflects an inherent trade-off: more aggressive prosodic modification, particularly emphatic stress and inserted pauses, occasionally causes higher WER (e.g., Whisper misrecognizes elongated stressed syllables or transcribes deliberate pauses as punctuation).

7 Conclusion

We have presented LOOPTTS, an offline Filter–Judge–Refiner framework that integrates prosodic diagnosis with guided expressive re-synthesis. The key idea is to leverage AudioLLMs as scalable but imperfect sources of diagnostic weak supervision and pair them with a Refiner trained on contrastive prosody data and a position-aware objective. Experiments show that the Refiner improves fine-grained instruction following under our

protocol, and that bounded refinement improves human-rated recovery quality on Stage 2 flagged utterances compared with raw audio and practical open-loop re-generation baselines. The modular architecture allows the inference-time Judge to be replaced, as shown by LOOPTTS-Q, although the training labels remain Gemini-derived. Extending the framework to full-distribution human evaluation, stronger deployment baselines, tonal languages, and lower-cost open judges are important directions for future work.

Limitations

The main human pipeline evaluation is conducted on Stage 2 flagged utterances, directly testing recovery from diagnosed defects. This subset is where refinement can change the output; on the full distribution, the observed effect would be diluted by many utterances that already pass without refinement. Due to the cost of blind human listening tests, we do not run full-distribution human evaluation over every initial-generation utterance. Appendix A instead reports larger-scale full-pipeline accounting and objective metrics over 3,000 utterances.

Our experiments focus on English read and emotional speech with a limited range of speakers and corpora. The framework is modular in that the initial TTS system and inference-time Judge can be replaced, as illustrated by the re-generation baselines and LOOPTTS-Q, but broader speaker, language, and speaking-style coverage remains future work. Extending the framework to larger multilingual, multi-speaker emotional corpora and incorporating prompt-based speaker conditioning are natural next steps.

Current annotations and controls represent stress and pause decisions as binary labels. This supports word-level diagnosis and correction, but does not model degrees of stress intensity or calibrated pause duration. Future extensions could use more fine-grained prosodic labels to support continuous or multi-level expressive control.

The Refiner performs guided expressive resynthesis conditioned on the original audio rather than waveform-level local editing. This design supports broad prosodic correction, but expressive or prosodic changes may also shift speaker embeddings. Such SIM degradation is a common trade-off in expressive TTS, where emotional/prosodic changes can affect speaker similarity (Wang et al., 2026a; Cho et al., 2025); Appendix D.5 discusses this trade-off in more detail.

Ethics Statement

Human annotation and fair compensation. Human participation in this work includes ten professional paid blind listening test evaluators with fluent English listening proficiency, six diagnostic alignment raters, and professional prosodic annotators who provided annotation ground-truth labels and human-instruction upper-bound references. All participants were compensated in accordance with local labor regulations and institutional guidelines, consistent with ACL requirements regarding fair treatment and remuneration of human participants. All human tasks were limited to listening to AI-synthesized speech and providing subjective quality ratings; no personally identifiable information was collected from any participant. Our institution classifies such perception-only studies as minimal-risk research, exempt from formal ethics board review.

Data privacy and consent. All training data in REFINER-DB are constructed from publicly available speech corpora released for research purposes (ESD, RAVDESS, SAVEE, MESS, EmoVoice-DB, and LibriTTS). The contrastive instruction data are generated entirely from these public datasets via our AudioLLM-annotated data construction pipeline; no private, user-uploaded, or personally identifiable data are used at any stage. For any dataset release, we will follow the license and redistribution terms of each source corpus: derived annotations and construction scripts will be released where permitted, while source audio will be redistributed only when the original license allows it. The released dataset and model checkpoints do not contain any private user data.

Licensing and responsible use. The complete code, data construction scripts, prompts, model checkpoints, generated annotations, raw AudioLLM JSON outputs where license-compatible, and evaluation sample lists will be released upon acceptance for non-commercial academic research. We acknowledge that controllable prosodic speech synthesis carries potential risks of misuse, including generating deceptive or manipulative audio content. We emphasize that LOOPTTS is intended as a research contribution to advance fine-grained prosodic control in TTS, and we encourage responsible use with appropriate human oversight in any

downstream application.

Usage of AI assistants. AI language models were used in this work in three capacities: (1) language polishing during paper writing, (2) prosodic annotation during dataset construction (Appendix I.1), and (3) prosodic diagnosis in the Judge stage of the LOOPTTS pipeline (Appendix I.2). All experimental design, analysis, and scientific conclusions were made by the authors.

Acknowledgments

This work was supported by the Program for Guangdong Introducing Innovative and Entrepreneurial Teams (Grant No. 2023ZT10X044).

References

Philip Anastassiou, Jiawei Chen, Jitong Chen, Yuanzhe Chen, Zhuo Chen, Ziyi Chen, Jian Cong, Lelai Deng, Chuang Ding, Lu Gao, et al. 2024. Seed-tts: A family of high-quality versatile speech generation models. *arXiv preprint arXiv:2406.02430*.

Jie Chen, Jinhao Jiang, Yingqian Min, Zican Dong, Shijie Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2025a. Sticker-TTS: Learn to utilize historical experience with a sticker-driven test-time scaling framework. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 12328–12338.

Sanyuan Chen, Shujie Liu, Long Zhou, Yanqing Liu, Xu Tan, Jinyu Li, Sheng Zhao, Yao Qian, and Furu Wei. 2024. Vall-e 2: Neural codec language models are human parity zero-shot text to speech synthesizers. *arXiv preprint arXiv:2406.05370*.

Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, JianZhao JianZhao, Kai Yu, and Xie Chen. 2025b. F5-tts: A fairytaler that fakes fluent and faithful speech with flow matching. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6255–6271.

Deok-Hyeon Cho, Hyung-Seok Oh, Seung-Bin Kim, Sang-Hoon Lee, and Seong-Whan Lee. 2024. Emosphere-tts: Emotional style and intensity modeling via spherical emotion vector for controllable emotional text-to-speech. *arXiv preprint arXiv:2406.07803*.

Deok-Hyeon Cho, Hyung-Seok Oh, Seung-Bin Kim, and Seong-Whan Lee. 2025. Diemo-tts: Disentangled emotion representations via self-supervised distillation for cross-speaker emotion transfer in text-to-speech.

Domenic V Cicchetti. 1994. Guidelines, criteria, and rules of thumb for evaluating normed and standardized assessment instruments in psychology. *Psychological assessment*, 6(4):284.

Jiayan Cui, Zhihan Yang, Naihan Li, Jiankun Tian, Xingyu Ma, Yi Zhang, Guangyu Chen, Runxuan Yang, Yuqing Cheng, Yizhi Zhou, et al. 2025. Glm-tts technical report. *arXiv preprint arXiv:2512.14291*.

Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, et al. 2024a. Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens. *arXiv preprint arXiv:2407.05407*.

Zhihao Du, Yuxuan Wang, Qian Chen, Xian Shi, Xiang Lv, Tianyu Zhao, Zhifu Gao, Yexin Yang, Changfeng Gao, Hui Wang, et al. 2024b. Cosyvoice 2: Scalable streaming speech synthesis with large language models. *arXiv preprint arXiv:2412.10117*.

Zhifang Guo, Yichong Leng, Yihan Wu, Sheng Zhao, and Xu Tan. 2023. Promptttts: Controllable text-to-speech with text descriptions. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

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

Shehzeen Hussain, Paarth Neekhara, Xuesong Yang, Edresson Casanova, Subhankar Ghosh, Roy Fejgin, Ryan Langman, Mikyas Desta, Leili Tavabi, and Jason Li. 2025a. Align2speak: Improving tts for low resource languages via asr-guided online preference optimization. *arXiv preprint arXiv:2509.21718*.

Shehzeen Samarah Hussain, Paarth Neekhara, Xuesong Yang, Edresson Casanova, Subhankar Ghosh, Roy Fejgin, Mikyas T. Desta, Rafael Valle, and Jason Li. 2025b. Koel-TTS: Enhancing LLM based speech generation with preference alignment and classifier free guidance. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 21219–21234.

Shengpeng Ji, Qian Chen, Wen Wang, Jialong Zuo, Minghui Fang, Ziyue Jiang, Hai Huang, Zehan Wang, Xize Cheng, Siqi Zheng, et al. 2025. Controlspeech: Towards simultaneous and independent zero-shot speaker cloning and zero-shot language style control. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6966–6981.

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, and 21 others. 2025. Kimi-audio technical report. *Preprint*, arXiv:2504.18425.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in neural information processing systems*, 33:17022–17033.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMlR.

Haoxun Li, Yu Liu, Yuqing Sun, Hanlei Shi, Leyuan Qu, and Taihao Li. 2025a. Emorl-tts: Reinforcement learning for fine-grained emotion control in llm-based tts. *arXiv preprint arXiv:2510.05758*.

Xiang Li, Duyi Pan, Hongru Xiao, Jiale Han, Jing Tang, Jiabao Ma, Wei Wang, and Bo Cheng. 2025b. Dialogueagents: A hybrid agent-based speech synthesis framework for multi-party dialogue. pages 1–6.

Tianchi Liu, Zeyang Song, Tianrui Wang, Zhipeng Li, Chenglin Xu, and Yiwen Guo. 2026. EmoTra-TTS: Smooth intra-utterance emotion transitions for speech synthesis. *arXiv preprint arXiv:2608.23791*.

Steven R Livingstone and Frank A Russo. 2018. The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english. *PloS one*, 13(5):e0196391.

Potsawee Manakul, Woody Haosheng Gan, Michael J Ryan, Ali Sartaz Khan, Warit Sirichotedumrong, Kunat Pipatanakul, William Held, and Diyi Yang. 2025. Audiojudge: Understanding what works in large audio model based speech evaluation. *arXiv preprint arXiv:2507.12705*.

Shae D Morgan. 2019. Categorical and dimensional ratings of emotional speech: Behavioral findings from the morgan emotional speech set. *Journal of Speech, Language, and Hearing Research*, 62(11):4015–4029.

MOSS-TTS Team. 2026. MOSS-TTS Family: An open-source speech and sound generation model family. https://github.com/OpenMOSS/MOSS-TTS.

Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. 2019. Relational knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3967–3976.

Puyuan Peng, Po-Yao Huang, Shang-Wen Li, Abdelrahman Mohamed, and David Harwath. 2024. Voicecraft: Zero-shot speech editing and text-to-speech in the wild. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12442–12462.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.

Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. 2022. Utmos: Utokyo-sarulab system for voicemos challenge 2022. *arXiv preprint arXiv:2204.02152*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, et al. 2023. Neural codec language models are zero-shot text to speech synthesizers. *arXiv preprint arXiv:2301.02111*.

Siyi Wang, Shihong Tan, Siyi Liu, Hong Jia, Gongping Huang, James Bailey, and Ting Dang. 2026a. Cocoemo: Composable and controllable human-like emotional tts via activation steering. *arXiv preprint arXiv:2602.03420*.

Tianrui Wang, Ziyang Ma, Yizhou Peng, Haoyu Wang, Zhikang Niu, Zikang Huang, Yihao Wu, Yi-Wen Chao, Yu Jiang, Yuheng Lu, et al. 2026b. Evaluating the expressive appropriateness of speech in rich contexts. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9088–9106.

Xiaofei Wang, Manthan Thakker, Zhuo Chen, Naoyuki Kanda, Sefik Emre Eskimez, Sanyuan Chen, Min Tang, Shujie Liu, Jinyu Li, and Takuya Yoshioka. 2024a. Speechx: Neural codec language model as a versatile speech transformer. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:3355–3364.

Yuancheng Wang, Haoyue Zhan, Liwei Liu, Ruihong Zeng, Haotian Guo, Jiachen Zheng, Qiang Zhang, Xueyao Zhang, Shunsi Zhang, and Zhizheng Wu. 2024b. Maskgct: Zero-shot text-to-speech with masked generative codec transformer. *arXiv preprint arXiv:2409.00750*.

Haibin Wu, Xiaofei Wang, Sefik Emre Eskimez, Manthan Thakker, Daniel Tompkins, Chung-Hsien Tsai, Canrun Li, Zhen Xiao, Sheng Zhao, Jinyu Li, and Naoyuki Kanda. 2024. Laugh now cry later: Controlling time-varying emotional states of flow-matching-based zero-shot text-to-speech. In *IEEE Spoken Language Technology Workshop (SLT)*, pages 690–697.

Kangxiang Xia, Xinfa Zhu, Jixun Yao, and Lei Xie. 2025. Mpo: Multidimensional preference optimization for language model-based text-to-speech. *arXiv preprint arXiv:2509.00685*.

Tianxin Xie, Yan Rong, Pengfei Zhang, Wenwu Wang, and Li Liu. 2025. Towards controllable speech synthesis in the era of large language models: A systematic survey. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 764–791.

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, and 19 others. 2025. Qwen3-omni technical report. *arXiv preprint arXiv:2509.17765*.

Chao Yan, Boyong Wu, Peng Yang, Pengfei Tan, Guoqiang Hu, Li Xie, Yuxin Zhang, Fei Tian, Xuerui Yang, Xiangyu Zhang, et al. 2025. Step-audio-editx technical report. *arXiv preprint arXiv:2511.03601*.

Dongchao Yang, Songxiang Liu, Rongjie Huang, Chao Weng, and Helen Meng. 2024. Instructtts: Modelling expressive tts in discrete latent space with natural language style prompt. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:2913–2925.

Guanrou Yang, Chen Yang, Qian Chen, Ziyang Ma, Wenxi Chen, Wen Wang, Tianrui Wang, Yifan Yang, Zhikang Niu, Wenrui Liu, et al. 2025a. Emovoice: Llm-based emotional text-to-speech model with freestyle text prompting. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 10748–10757.

Qing Yang, Zhenghao Liu, Junxin Wang, Yangfan Du, Pengcheng Huang, and Tong Xiao. 2025b. Rlaif-spa: Optimizing llm-based emotional speech synthesis via rlaif. *arXiv preprint arXiv:2510.14628*.

Zhen Ye, Xinfa Zhu, Chi-Min Chan, Xinsheng Wang, Xu Tan, Jiahe Lei, Yi Peng, Haohe Liu, Yizhu Jin, Zheqi Dai, et al. 2025. Llasa: Scaling train-time and inference-time compute for llama-based speech synthesis. *arXiv preprint arXiv:2502.04128*.

Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. 2019. Libritts: A corpus derived from librispeech for text-to-speech. *arXiv preprint arXiv:1904.02882*.

Dong Zhang, Zhaowei Li, Shimin Li, Xin Zhang, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2024. Speechalign: Aligning speech generation to human preferences. *Advances in Neural Information Processing Systems*, 37:50343–50360.

Xueyao Zhang, Chaoren Wang, Huan Liao, Ziniu Li, Yuancheng Wang, Li Wang, Dongya Jia, Yuanzhe Chen, Xiulin Li, Zhuo Chen, et al. 2025. Speechjudge: Towards human-level judgment for speech naturalness. *arXiv preprint arXiv:2511.07931*.

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

Kun Zhou, Berrak Sisman, Rui Liu, and Haizhou Li. 2021. Seen and unseen emotional style transfer for voice conversion with a new emotional speech dataset. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 920–924. IEEE.

Siyi Zhou, Yiquan Zhou, Yi He, Xun Zhou, Jinchao Wang, Wei Deng, and Jingchen Shu. 2025. Indextts2: A breakthrough in emotionally expressive and duration-controlled auto-regressive zero-shot text-to-speech. *arXiv preprint arXiv:2506.21619*.

Yixuan Zhou, Xiaoyu Qin, Zeyu Jin, Shuoyi Zhou, Shun Lei, Songtao Zhou, Zhiyong Wu, and Jia Jia. 2024. Voxinstruct: Expressive human instruction-to-speech generation with unified multilingual codec language modelling. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 554–563.

A LoopTTS on scaled evaluation set

Beyond the small-scale human evaluation in Section 6.2, we conduct a larger-scale objective comparison on 3,000 emotional synthesis utterances (Table 6). Each sample provides a neutral reference audio and a target emotion label; the system must synthesize emotionally expressive speech for the given text. We report WER (↓), SIM (↑), and UTMOS (↑). The high error rate of CosyVoice2 and EmoVoice is largely due to generation failures (e.g., garbled audio, missing words); Stage 1 performs up to two additional re-generation attempts after the initial generation and removes most such failures before Judge-based diagnosis. Samples that still fail the Stage 1 filters after these attempts are not passed to Stage 2, while the remaining samples enter optional refinement.

To contextualize full-distribution performance, we additionally apply untargeted CosyVoice2 and EmoVoice re-generation directly to the same 3,000 synthesis requests. We also evaluate a best-of-5 alternative: for each request, CosyVoice2 generates five candidates, each candidate is independently scored by the AudioLLM for target-emotion fidelity and overall quality, and the highest-scoring candidate is retained.

System	WER ↓	SIM ↑	UTMOS ↑
CosyVoice2	3.69	0.70	3.33
EmoVoice	3.65	0.65	3.31
LoopTTS (Stage 1)	2.94	0.74	3.76
CosyVoice2 re-generation	2.90	0.74	3.81
EmoVoice re-generation	2.81	0.72	3.79
Judge-reranked best-of-5	2.90	0.74	3.82
LoopTTS (Full)	2.73	0.73	3.93

Table 6: Objective evaluation on 3,000 emotional synthesis utterances. WER is computed via Whisper-large-v3-turbo; SIM is measured against the neutral speaker reference; UTMOS estimates perceptual quality.

All rows use the same 3,000 requests and evaluation metrics. The full LoopTTS pipeline obtains

the lowest WER and highest UTMOS while retaining comparable SIM. Judge-reranked best-of-5 provides little improvement over a single CosyVoice2 retry and remains below LOOPTTS in WER and UTMOS, while requiring five generations and five AudioLLM scoring calls per request. In the refinement stage of LOOPTTS, each flagged request instead requires one AudioLLM call to produce its refine instruction and one Refiner generation.

To make the scaled-set cascade more transparent, we also report stage-level accounting in Table 7. This separates utterances that pass the initial coarse filter, those recovered by Stage 1 re-generation, those flagged by Stage 2, and the remaining flagged subset after one refinement round.

Stage	# Utt.	Rate
Total	3,000	100.00%
Initial pass	2,367	78.90%
After regen.	2,753	91.76%
Stage 2 flagged	237	7.90%
After 1st refine	103	3.43%

Table 7: Stage-level accounting for the large-scale LOOPTTS cascade.

B Data Statistics

Table 8 summarizes the composition of REFINER-DB. The dataset contains ∼42K contrastive tuples sourced from six publicly available corpora, all used for training. In addition, we construct three non-overlapping evaluation splits from the same source corpora: 1,796 tuples for validation and 500 tuples for testing. There is no overlap in target recordings or text utterance instances among the training, validation, and test sets. We do not claim an unseen-speaker split for all corpora; same-speaker neutral prompts are part of the zero-shot cloning setup.

ESD (Zhou et al., 2021), RAVDESS (Livingstone and Russo, 2018), SAVEE[4], MESS (Morgan, 2019), and EmoVoice-DB (Yang et al., 2025a) provide emotionally expressive recordings, while LibriTTS (Zen et al., 2019) contributes read speech with natural prosodic variation. Where available, utterance-level emotion, speed, and pitch metadata are sourced from VccmDataset (Ji et al., 2025). For each target recording y_{tgt}, we synthesize a neutral counterpart y_{neu} with CosyVoice2 (Du et al., 2024b) in zero-shot mode using a neutral prompt from the same speaker, and retain pairs only when the neutral synthesis satisfies WER=0 and UTMOS>3. The paired utterances therefore share text and speaker prompt, making prosody the dominant contrast while not eliminating all acoustic differences.

Corpus	# Tuples	Type
ESD	14,000	Emotional
RAVDESS	1,440	Emotional
SAVEE	480	Emotional
MESS	6,800	Emotional
EmoVoice-DB	17,280	Emotional
LibriTTS	2,000	Regular
Total	∼42,000	—

Table 8: REFINER-DB composition by source corpus.

The 500 test tuples are partitioned into three non-overlapping subsets: 200 neutral–emotional pairs for AudioLLM annotation capability assessment (Section 5.2), 100 TTS-generated utterances for AudioLLM diagnostic capability assessment (Section 5.1), and 200 tuples for instruction-following comparison (Section 6.3).

C Pipeline Evaluation Setup

Data preparation. Using speakers from the evaluation set, we pair their neutral reference audio with text transcripts as speaker prompts for CosyVoice2 zero-shot generation, synthesizing 2,000 utterances per condition (*Neutral* and *Emotional*). For the Emotional condition, the target emotion label is provided by the dataset condition and used for all systems. Stage 1 performs up to two additional re-generation attempts after the initial generation, stopping early once WER$\leq$3.0% and UTMOS>3.0 are both satisfied; samples still failing after these attempts do not enter Stage 2. After Stage 1 filtering and Stage 2 AudioLLM diagnosis (score 1–10), utterances scoring < 5 are flagged. We sample 100 flagged utterances per condition (200 total) for MOS-style human evaluation, directly testing each method's ability to recover from diagnosed defects. For LOOPTTS-Q, the same evaluation protocol and Refiner are used, but Qwen3-Omni Thinking replaces Gemini-3-pro for the complete inference-time Judge step, including diagnosis and instruction generation.

Condition details. Table 3 reports eight conditions in three groups:

- **Before refinement** (Baseline): original flagged utterances, serving as the lower

[4] http://kahlan.eps.surrey.ac.uk/savee

bound.

- **CosyVoice2 re-generation** (Re-generation, open-loop): re-synthesized from scratch by CosyVoice2 with a different random seed but the same text T, speaker prompt, and emotion instruction. This is the default untargeted retry strategy, using the same Stage 2 Judge trigger and one additional synthesis call as one-round LOOPTTS.

- **EmoVoice re-generation** (Re-generation, open-loop): same setup as above but using EmoVoice as the generator, testing cross-model recovery under the same one-correction budget.

- **LOOPTTS, Global-only** (Closed-loop): the Refiner retains the Judge-derived global emotion, speed, and pitch controls but receives no word-level stress or pause cues. All other inputs and evaluation settings match one-round LOOPTTS.

- **LOOPTTS, 1-round refinement** (Closed-loop): the flagged y_{init} is diagnosed by Gemini-3-pro to produce instruction I; the Refiner takes (y_{init}, I, T) and outputs corrected speech y_R.

- **LOOPTTS-Q, 1-round refinement** (Closed-loop): same as LOOPTTS, except that Qwen3-Omni Thinking performs the complete Stage 2 Judge step, including scoring, diagnosis, and instruction generation. The Refiner is unchanged and is not retrained.

- **LOOPTTS, 2-round refinement** (Closed-loop): after the first round produces $y_R^{(1)}$, it is re-diagnosed; if still < 5, a second instruction $I^{(2)}$ is generated and $y_R^{(2)} = \text{Refiner}(y_R^{(1)}, I^{(2)}, T)$. No second round if the first already scores ≥ 5.

- **Human-instruction** (Closed-loop): a native English speaker with phonetics training first writes the refine instruction in the same structured format (global style + word-level stress/-pause), replacing the AudioLLM output. Two additional professional annotators then review and revise the stress/pause choices until consensus. This isolates instruction quality and serves as a soft upper bound.

Evaluation protocol. SIM is computed against the neutral speaker reference. Ten professional paid evaluators score each sample on MOS (1–5) in a blind test with all conditions shuffled and system identities hidden. Conditions are compared on matched texts, speakers, and target styles; evaluators may therefore hear the same text across systems, but presentation order is randomized. For paired preference evaluation, each comparison uses 30 flagged samples and the same 10 evaluators in a blind A/B setting. The order of systems is randomized per sample, and we report the aggregate preference rate as well as the number of samples for which at least 7/10 or 9/10 evaluators agree on the preferred system.

D Additional Robustness and Reliability Analyses

D.1 Audio Conditioning Ablation

The Refiner conditions on the initial utterance y_{init}, which provides an acoustic reference for preserving speaker identity and unmodified prosodic attributes. Removing this input reduces the model to instruction-conditioned synthesis from the text and speaker prompt, testing whether targeted correction benefits from explicitly seeing the utterance to be refined. This ablation is also a check on whether the system is genuinely using the defective input or mainly acting as a standalone expressive TTS model.

Variant	WER↓	SIM↑	MOS↑	Avg. MOS-I↑
Refiner	2.90	0.68	$4.21_{\pm.14}$	$4.21_{\pm.09}$
w/o y_{init}	2.91	0.62	$4.17_{\pm.12}$	$3.94_{\pm.11}$

Table 9: Ablation of conditioning on the initial utterance.

Removing y_{init} only slightly decreases MOS from 4.21 to 4.17, showing that the model can still generate broadly natural speech from the text, speaker prompt, and instruction. However, SIM drops from 0.68 to 0.62 and Avg. MOS-I decreases from 4.21 to 3.94. This indicates that the initial utterance is important for targeted refinement: without this acoustic reference, the model is more likely to drift from the intended speaker/prosodic identity and, in some instruction-following cases, change prosodic regions that were not requested. The full Refiner therefore better balances naturalness, preservation, and instruction execution.

D.2 Counterfactual Instruction Robustness

To test whether the Refiner blindly executes erroneous local edits, we construct a counterfactual set of 40 utterances containing linguistically inappropriate stress placements on function words or pause placements between tightly coupled constituents. For each utterance, one correct stress/pause position is replaced by a counterfactual one, allowing us to check both wrong-position execution and correct-position preservation.

Setting	WER↓	SIM↑
Before refinement	1.92	0.68
With GT instruction	1.72	0.67
With counterfactual instruction	1.87	0.68

Table 10: Objective results under counterfactual local stress/pause instructions.

The counterfactual instruction affects refinement, but WER still improves over the pre-refinement audio. This suggests that the Refiner does not simply sacrifice content fidelity to execute a noisy instruction. One reason is the training setup: during Refiner training, instructions are paired with correct target audio, so the model mainly learns acoustically plausible corrections that move the input toward a better target realization, rather than arbitrary local command execution. This test is limited to a small counterfactual set and does not fully measure false-edit rates on natural unrequested positions.

Type	Wrong Exe. Rate	Corr. Pres. Rate
Stress	10%	34%
Pause	28%	52%

Table 11: Professional annotation of counterfactual instruction execution. Exe. and Pres. denote execution and preservation.

These results suggest partial robustness to over-diagnosed local edits: erroneous local edits are usually not introduced. The non-zero preservation rates should be interpreted more cautiously; they show that some emotionally salient stress/pause cues can remain natural even when omitted from the instruction, but they do not eliminate the risk of under-diagnosis.

D.3 Iteration Budget

We further check a 3-round setting to evaluate whether additional iterations continue to help.

The objective gains saturate after one to two rounds, while SIM gradually decreases as later rounds refine previously generated audio rather than the original audio. AudioLLM over- or under-diagnosis may also contribute to this accumulation through unnecessary edits or residual defects. We therefore view one or two rounds as the practical operating point.

Setting	Neutral		Emotional	
	WER↓	SIM↑	WER↓	SIM↑
1-round	1.86	0.66	1.79	0.65
2-round	1.85	0.64	1.80	0.61
3-round	1.85	0.60	1.80	0.58

Table 12: Objective results with additional refinement rounds.

D.4 Human Evaluation Reliability

For MOS-style ratings, we compute Intraclass Correlation Coefficient ICC(2,k), which measures the reliability of averaged continuous scores from multiple raters. Following Cicchetti's guideline (Cicchetti, 1994), values from 0.60 to 0.74 are considered good and 0.75 to 1.00 excellent.

Most MOS/MOS-I reliability values are in the good range or above, and the Refiner shows good consistency on the prosody-related dimensions. Some dimensions remain below the good range, reflecting the inherent difficulty of fine-grained subjective prosody evaluation.

D.5 Speaker Similarity Trade-off

SIM in our main experiments is computed against a neutral speaker reference. This metric can decrease when emotional prosody moves away from the neutral reference and therefore conflates intended emotional variation with speaker-identity drift. To examine speaker preservation under emotional generation, we additionally compute same-speaker expressive-reference similarity (SIM-EXP). For each generated output x, we average its similarity to five different-utterance references r_i from the same speaker and target emotion:

$$\text{SIM-EXP}(x) = \frac{1}{5}\sum_{i=1}^{5}\cos(e(x), e(r_i))\,, \quad (6)$$

where $e(\cdot)$ denotes the normalized WavLM speaker embedding.

CosyVoice2 has the highest neutral-reference SIM but the lowest SIM-EXP, consistent with its

System	MOS	Emotion	Pause	Stress	Pitch	Speed
CosyVoice2	0.784	0.562	0.915	0.496	0.728	0.848
CosyVoice2-marker	0.549	0.588	0.943	0.598	0.729	0.865
EmoVoice	0.496	0.706	0.899	0.694	0.683	0.823
Refiner	0.791	0.679	0.886	0.677	0.744	0.835

Table 13: ICC(2,k) reliability for MOS and MOS-I ratings.

System	Neutral SIM↑	SIM-EXP↑
CosyVoice2	**0.72**	0.65
EmoVoice	0.67	0.75
Refiner	0.68	**0.77**

Table 14: Speaker similarity against a neutral reference and five same-speaker, same-emotion expressive references.

weaker target-emotion realization in Table 4 (Emotion MOS-I: 2.25). The Refiner retains neutral-reference SIM comparable to EmoVoice while achieving the highest SIM-EXP. These results suggest that the SIM reduction under emotional generation primarily reflects intended emotional variation rather than substantial speaker drift. Similar patterns have been reported in expressive TTS: CoCoEmo observes that speaker similarity can decrease as emotional similarity increases (Wang et al., 2026a), and DiEmo-TTS shows that speaker embeddings can form emotion-dependent sub-clusters (Cho et al., 2025).

D.6 Construction Cost Estimate

For REFINER-DB construction, each tuple requires one AudioLLM call over two audio clips. Including audio, transcript, and prompt, each tuple uses about 2.7K input tokens and 0.6K output tokens. For 42K tuples, this corresponds to about 113.4M input tokens and 25.2M output tokens. Using Gemini-3-pro pricing of $4 per 1M input tokens and $18 per 1M output tokens, the estimated construction cost is about $907.2. At inference time, each retained utterance requires one Judge call, and each flagged utterance additionally requires one Refiner generation per refinement round; this is why our intended deployment setting is offline batch synthesis.

D.7 Prompt-only TTS Baselines

We also conducted small-sample listening trials with Qwen3-TTS (Hu et al., 2026) and MOSS-TTS (MOSS-TTS Team, 2026). In these trials, the models did not reliably execute fine-grained word-level stress or pause edits through prompts alone, making them difficult to use as comparable refiners for this task. This observation is preliminary rather than a full MOS-style baseline, and we do not claim that all future prompt-only or SSML-like systems would fail. The key distinction is that prompt-only optimization is limited by the controllability exposed by the original TTS model, whereas LOOPTTS performs correction through a separate post-generation Refiner.

E Instruction-Following Baselines

CosyVoice2 (Du et al., 2024b) is a streaming TTS that accepts natural-language instructions alongside target text and generates speech from scratch. It must interpret open-ended instructions, which often leads to verbatim reading of the instruction rather than executing it as a prosodic directive.

CosyVoice2-marker augments CosyVoice2 with explicit prosodic markup tags (e.g., `<strong>`, `<breath>`) instead of free-form language. This provides unambiguous structural directives and avoids verbatim reading, but can cause mispronunciation at modified positions. We tune prompts on the validation set and use the best-performing markup variant for the comparison.

EmoVoice (Yang et al., 2025a) is the pretrained 1.5B backbone *without* refinement fine-tuning, isolating the contribution of our training procedure from the backbone's inherent capabilities.

Speech editing methods (VoiceCraft (Peng et al., 2024), SpeechX (Wang et al., 2024a)) are excluded as they address content-level word replacement and cannot accept prosodic instructions.

F Implementation Details

Model architecture. The Refiner is built on EmoVoice-1.5B, which uses Qwen2.5-1.5B (hidden dimension 896) as the language model backbone. Audio is represented as 3-layer grouped semantic codes from the CosyVoice codec (audio vocabulary size 4,160 per layer). We initialize from the pretrained EmoVoice checkpoint and fine-tune all LLM parameters (encoder frozen).

Training setup. We fine-tune with AdamW (learning rate 5×10^{-6}, no weight decay) using a warmup schedule (1,000 warmup steps, 300K total steps) and a batch size of 8.

Hyperparameters. The position-weighting hyperparameters are α=0.2 (stress) and β=0.3 (pause) in Equation 1. The structural alignment loss weight is λ=0.1 in Equation 5. These values were selected based on validation-set MOS-I and WER trends without extensive search.

AudioLLM versions. We accessed Gemini-3-pro through Google's official Gemini API using the model code `gemini-3-pro-preview` in January 2026. For Qwen3-Omni Thinking, we use the official `Qwen/Qwen3-Omni-30B-A3B-Thinking` checkpoint.

Inference. The Judge score threshold is τ_{judge}=5. Stage 1 filtering uses Whisper-large-v3-turbo for WER computation and UTMOS for quality estimation, with WER$\leq$3.0% and UTMOS>3.0 as the pass condition. The Refiner uses deterministic greedy decoding (argmax, without sampling) with a maximum of 256 new tokens per utterance. AudioLLM responses are parsed as strict JSON according to the schemas in Appendix I; malformed outputs are retried once, and unresolved failures are excluded during data construction or treated as no-refinement fallbacks during inference. Because proprietary AudioLLM endpoints can change, we will release the prompts, raw parsed outputs, and per-call metadata where permitted.

G AudioLLM Evaluation Setup

Data source. Evaluation data is drawn from the 500-tuple test set held out from REFINER-DB. We use 200 neutral–expressive pairs for annotation evaluation and 100 TTS-generated utterances for diagnostic evaluation. For each emotional utterance, a neutral counterpart is synthesized via CosyVoice2 zero-shot generation, yielding 200 neutral–emotional pairs.

Diagnostic Capability (Section 5.1). We select 50 Regular and 50 Emotional TTS-generated utterances with noticeable prosodic defects. Each utterance is presented to five AudioLLMs along with its text transcript; the AudioLLM diagnoses prosodic defects and generates refine instructions. Six non-technical evaluators with fluent English listening and reading proficiency independently listen to each utterance, review the AudioLLM's diagnosis, and rate its alignment with their own judgment on a 1–4 scale (4: fully aligned, 3: mostly aligned, 2: partially aligned, 1: not aligned). Scores are averaged across evaluators.

Annotation Capability (Section 5.2). All 200 neutral–expressive pairs are used. Professional annotators with expertise in prosodic analysis listen to each pair and label the most prominent stressed words and pause positions that distinguish the expressive utterance from its neutral counterpart (see the annotation interface in Appendix J). Each AudioLLM is independently prompted to annotate the same pairs, and its outputs are evaluated against the professional labels using micro recall and F1.

H Scoring Rubrics

Evaluators receive the following rubrics before the blind test. Each sample is rated independently on MOS and five MOS-I dimensions. In Table 3, MOS measures whether the recovered utterance sounds natural after correction. In Table 4, MOS-I separately measures whether the specified instruction is executed on the intended control dimension, rather than overall naturalness. For local controls, MOS-I is our human subjective proxy for requested stress/-pause execution and unintended changes that make the instruction less correctly or naturally realized.

MOS: Naturalness & Audio Quality. Rate how natural and human-like the speech sounds, regardless of whether instructions are followed. **5 (Excellent):** Completely natural, human-like, no artifacts. **4 (Good):** Natural, but with very minor artifacts. **3 (Fair):** Understandable, but somewhat robotic or buzzy. **2 (Poor):** Unnatural, distinct metallic or robotic sounds. **1 (Bad):** Completely unnatural, unintelligible artifacts.

MOS-I: Instruction Adherence. Each of the five sub-dimensions (emotion, pitch, speed, stress, pause) is scored independently. Focus only on whether the specific instruction for that dimension is followed. **5 (Natural adherence):** The instruction is fully followed, and the result sounds natural and effortless. **4 (Followed, slightly unnatural):** The instruction is followed, but the execution feels slightly exaggerated or mechanical. **3 (Partially followed):** The instruction is partially followed; some aspects are correct, others are missing or weak. **2 (Barely followed):** The instruction is barely perceptible; very weak or mostly missing.

1 (Not followed): The instruction is completely ignored or contradicted.

I AudioLLM Prompts

This section provides the full prompts used for AudioLLM-based prosodic annotation (Section 3.2) and inference-time diagnosis (Section 3.1).

I.1 Annotation Prompt (Training-Time)

The following prompt is used during REFINER-DB construction (Section 3.2, Step 2). The AudioLLM receives Audio A (y_{neu}, the neutral synthesis) and Audio B (y_{tgt}, the expressive human recording) along with the shared transcript, and outputs a structured JSON specifying word-level prosodic differences. Global style attributes (emotion, speed, pitch) are extracted separately from dataset metadata and merged with the JSON output in post-processing to form the final refine instruction I.

Annotation Prompt

```
SYSTEM INSTRUCTIONS
You are a Prosody Comparison Expert.  Your goal is to
compare two audio recordings of the same text and identify
word-level prosodic differences.

TASK
1. Listen to both audios: Audio A is a neutral, prosodically
flat synthesis.  Audio B is an expressive human recording
of the same text.
2.  Compare A vs B: Identify where A fails to match B's
prosody, focusing on two dimensions:
   - stress: Words that are emphasized/stressed in B but
not in A.
   - pause: Positions where B has a noticeable pause but A
does not (report the word before the pause).
3. Select top changes: Pick the 3-5 most salient prosodic
differences.
4. Generate JSON: Output the result in strict JSON format.

OUTPUT FORMAT
{
    "local_edits": [
       { "word": "<word>", "type": "stress" | "pause" }
    ]
}

CONSTRAINTS
1. Type values: MUST be one of "stress" or "pause".
2. Word matching: The "word" field MUST be an exact word
from the provided transcript.
3. Evidence-based: Only report differences you can clearly
hear between A and B. Do not guess.
4.   Output:  RAW  JSON  ONLY.  No  markdown  blocks,  no
explanation.
```

I.2 Diagnostic Prompt (Inference-Time)

The following prompt is used during Stage 2 inference-time diagnosis (Section 3.1). Unlike the annotation prompt, which compares two audio recordings, the diagnostic prompt receives a single TTS-generated utterance y_{init} along with its transcript and, when available, the target emotion label from the synthesis condition. The AudioLLM first scores prosodic naturalness and emotional fidelity (1–10). If the score is 5 or above, only the score is returned and the utterance is accepted. Otherwise, the AudioLLM produces a full diagnosis covering global style (emotion, speed, pitch) and local prosody (word-level stress and pause positions), along with a natural-language refine instruction forwarded to Stage 3.

Diagnostic Prompt (Inference-Time)

```
SYSTEM INSTRUCTIONS
You are a Prosodic Quality Judge.  Your goal is to
diagnose prosodic defects in a TTS-generated utterance and,
if needed, produce a structured refine instruction for
correction.

TASK
1. Listen to the audio and read the transcript carefully.
2. Score: Rate overall prosodic naturalness and emotional
fidelity on a 1-10 scale (10:  perfect, 1:  completely
unnatural).
3.  If score >= 5:  Output only the score.  No further
diagnosis is needed.
4. If score < 5: Produce a full diagnosis:
   a. Diagnose global style: Specify the target emotion,
speed, and pitch using the strict enums below.
   b. Diagnose local prosody: Identify the 3-5 most salient
word-level prosodic defects, focusing on two dimensions:
      - stress: Words that should be emphasized but are
not.
      - pause: Positions where a pause is missing (report
the word before the pause position).
   c.  Generate a natural-language refine instruction
summarizing all changes.
5. Generate JSON: Output the result in strict JSON format.

GLOBAL STYLE ENUMS (STRICT)
emotion:  ["Angry", "Contempt", "Disgusted", "Fear",
          "Happy", "Sad", "Surprised", "Neutral"]
speed:    ["Slow", "Moderate", "Fast"]
pitch:    ["Low", "Moderate", "High"]

OUTPUT FORMAT
If score >= 5:
{ "score": <1-10> }

If score < 5:
{
    "score": <1-10>,
    "global_style": {
       "emotion": "<target>",
       "speed": "<target>",
       "pitch": "<target>"
    },
    "local_edits": [
       { "word": "<word>", "type": "stress" | "pause" }
    ],
    "instruction": "<natural-language refine instruction>"
}

CONSTRAINTS
1. Strict enums: Global style values MUST be chosen from
the provided lists.
2. Word matching: The "word" field MUST be an exact word
from the provided transcript.
3. Type values: MUST be one of "stress" or "pause".
4. Instruction: Summarize all suggested changes (global and
local) in one concise natural-language sentence.
5.   Output:  RAW  JSON  ONLY.  No  markdown  blocks,  no
explanation.
```

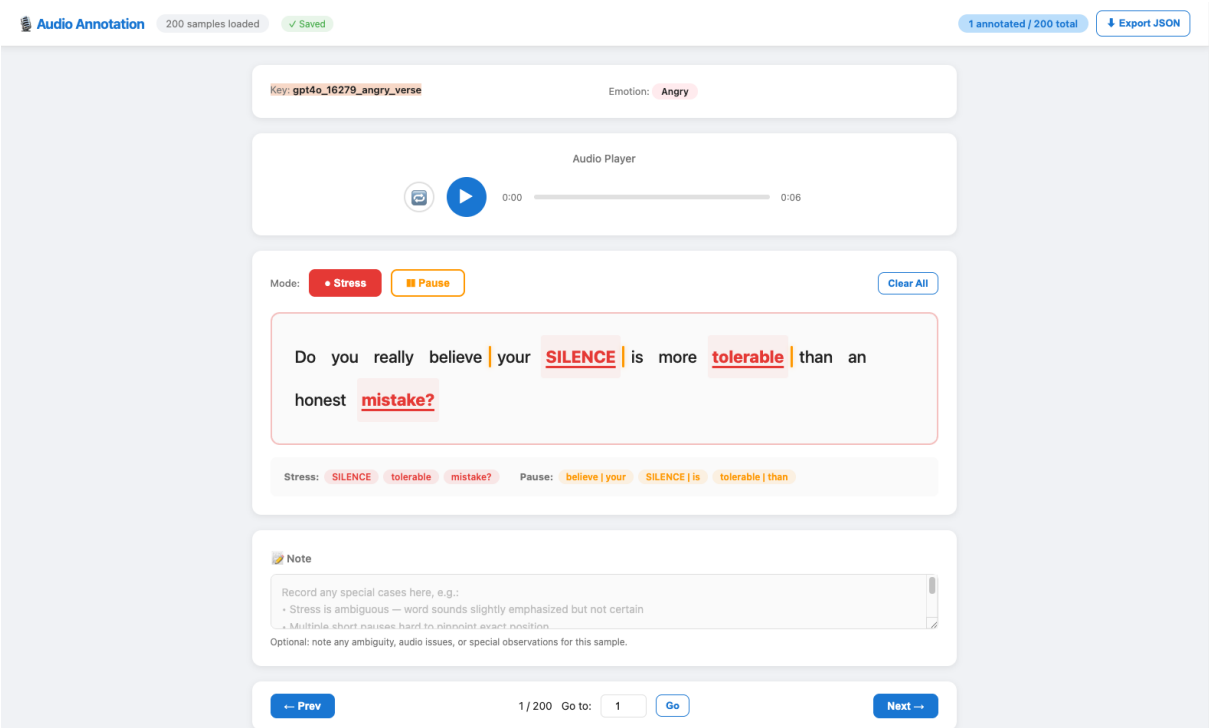

Figure 5: Screenshot of the web-based human annotation interface used for collecting prosodic ground-truth labels (Section 5.2). Annotators toggle between *Stress* and *Pause* modes, clicking words or inter-word gaps to mark stressed words (red) and pause positions (vertical bars). A live preview and free-text note field are provided below.

J Human Annotation Interface

To collect professional human labels for evaluating AudioLLM annotation quality (Section 5.2), we build a web-based annotation tool (Figure 5). For each of the 200 neutral–expressive utterance pairs, annotators listen to the expressive recording and interact with the displayed transcript in two modes:

- **Stress mode**: clicking a word toggles it as a stressed word (highlighted in red).
- **Pause mode**: clicking the gap between two words toggles a pause marker at that position (indicated by a vertical bar).

Selected stress words and pause positions are displayed in a live preview below the text. Annotators can also leave free-text notes for ambiguous cases. Annotations are auto-saved and exportable as structured JSON, with each entry recording the list of stressed word indices and pause slot indices for downstream evaluation.