HUMAN-IN-THE-LOOP DATA CURATION AND PREFERENCE OPTIMIZATION FOR VIETNAMESE NON-VERBAL TEXT-TO-SPEECH

Tri-Nhan Do Minh-Khang Le Hoang-Minh Tran
Thanh-Hau Luong Long-Hai Tran Viet-Lam Phung*

Independent Researchers, Vietnam

{dotrinhan99, minhkhangle2k4, tranhminh8464, luonghau2909, longhai7803, lam.phungviet33}@gmail.com

*Corresponding author

ABSTRACT

Few open text-to-speech (TTS) models can render non-verbal vocalizations (NVs) such as laughter, breathing, sniffing and throat clearing, and fewer still support Vietnamese. We describe our system for the VLSP 2026 ViNV-TTS shared task, which requires each tagged NV to be rendered at its intended position while the voice of the target speaker is preserved. Starting from a multilingual masked generative TTS model, we build a human-in-the-loop pipeline in which listener judgments decide all training signal. Human listening turns a small, noisily tagged corpus into clean single-NV examples, selects the best of the model's own synthesized speech as additional training data, and supplies preference pairs for Direct Preference Optimization, which we adapt to masked token prediction by estimating sequence likelihoods from masked-token log-probabilities. The pipeline teaches the model all four NV classes from a few dozen clean examples of the rarest ones, and on the public test, preference optimization yields the largest improvement in NV placement at a small cost in speaker similarity.

***Index Terms*—** text-to-speech, non-verbal vocalization, Vietnamese, preference optimization, masked generative model

1. INTRODUCTION

Spontaneous conversation is full of sounds that are not words. Speakers laugh, breathe audibly, sniff and clear their throats, and synthetic speech without these non-verbal vocalizations (NVs) sounds read rather than spoken. Recent TTS systems have begun to expose NVs as inline tags, although the supported tag sets are small and differ from one model to the next. Vietnamese has received little attention so far, and none of the open-weight models in our survey covers all of the NV classes considered here (Section 2).

The VLSP 2026 ViNV-TTS shared task [1] addresses this gap. Given Vietnamese text with inline tags for laughter, breathing, sniffing and throat clearing, a system must produce speech in which every tagged NV is present, of the correct type and at the intended position, while the speech remains intelligible and keeps the identity of the target speaker. Track A covers speakers seen in training, whereas Track B requires zero-shot cloning of unseen speakers from a short reference clip. The organizers provide the first public Vietnamese conversational corpus with inline NV tags, collected from podcasts and talk shows, and rank systems by a weighted combination of human and automatic judgments (Section 3).

Two properties of this setting shape our approach. The corpus tags are noisy, since many marked NVs are inaudible or belong to another class, and after careful listening the rarest classes keep only a handful of usable examples. In this regime the choice of examples matters more than their number. We therefore build a three-stage human-in-the-loop pipeline around a multilingual masked generative TTS model, in which listener judgments select every training example and every preference pair (Fig. 1). Our contributions are threefold.

- A listening-based curation procedure that turns a noisy corpus into single-NV training segments and teaches a multilingual model four Vietnamese NVs from a few dozen clean examples of the rarest classes.
- A self-synthesis loop in which blind raters grade the model's own outputs and only accepted clips extend the training data.
- Direct Preference Optimization (DPO) [2] for a masked multi-codebook TTS model, trained on the same ratings, with sequence likelihoods estimated from masked-token log-probabilities under masks shared by the policy, the reference and both clips of a pair.

2. RELATED WORK

Non-verbal vocalizations in TTS. Among codec-language-model systems, CosyVoice 2 [3] supports laughter and breath tags, Dia [4] has 21 tags that include all four of our classes, and Orpheus [5], Chatterbox-Turbo [6], Fish Audio S2 Pro [7] and Higgs TTS 3 [8] provide overlapping subsets. On the data side, NonverbalTTS [9] releases 17 h of English speech with ten NV types and shows that fine-tuning open models on it approaches CosyVoice 2, and Ye et al. [10] build NV-annotated speech at scale with an automatic pipeline. All of these rely on large, mostly English data. In our survey of twelve open-weight models (Table 1), the models that speak Vietnamese cover at most three of the four classes, and VieNeu-TTS [11], the only Vietnamese-specific model, offers only an experimental throat-clearing tag. We instead start from a backbone chosen for its voice cloning and audio quality rather than its NV coverage, and adapt it with a small, noisy and unbalanced corpus by focusing on which examples it learns from.

Masked generative TTS. Following MaskGIT [15], SoundStorm [16] and MaskGCT [17] generate codec tokens in parallel by iterative unmasking. Like flow-matching TTS [18], such models define no tractable sequence likelihood, which complicates likelihood-based preference optimization.

Preference alignment for speech. DPO [2] has been applied to autoregressive codec-LM TTS in SpeechAlign [19], UNO [20], Tian et al. [21] and Koel-TTS [22], and Seed-TTS [23] uses preference-based RL. These methods use exact autoregressive likelihoods, and most build pairs from automatic rewards such as WER and speaker similarity. Diffusion-DPO [24] replaces the likelihood with an ELBO,

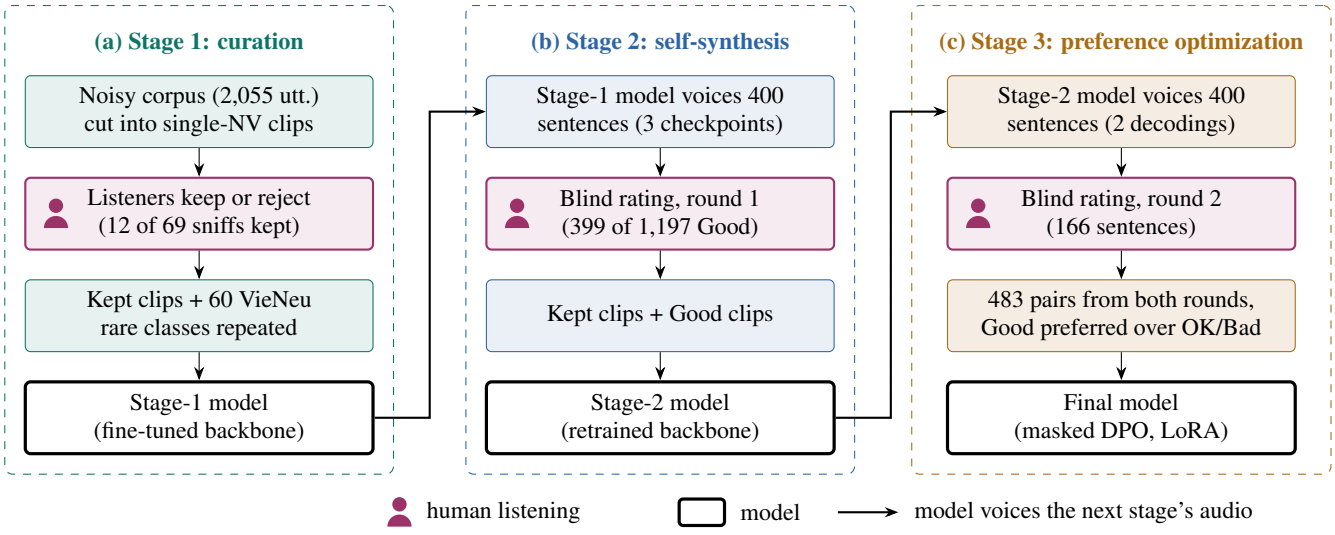


Fig. 1. Overview. Every stage follows the same pattern, in which audio is collected or synthesized, humans listen to it, and only what they accept is used for training. The model produced by one stage voices the audio that is rated in the next.

Model	Vi	Laugh	Breath	Sniff	Throat
OmniVoice [12]	✓	✓	×	×	×
Higgs TTS 3 [8]	✓	✓	≈	✓	≈
Fish S2 Pro [7]	✓	✓	✓	?	✓
VoxCPM2 [13]	✓	×	×	×	×
VieNeu-TTS v3 [11]	✓	≈	≈	×	✓
Dia [4]	×	✓	✓	✓	✓
Orpheus [5]	×	✓	≈	✓	≈
Breeze-TTS-2 [14]	×	✓	≈	×	✓
Chatterbox-T. [6]	×	✓	×	×	≈

Table 1. Survey of open-weight TTS models, compiled from model cards and papers. Vi marks Vietnamese support. A check mark denotes a tag for the class, ≈ a tag for a related sound only (sigh or cough), and ? free-form tags. Qwen3-TTS, Higgs Audio v2 and OpenAudio S1-mini cover at most one class and do not speak Vietnamese.

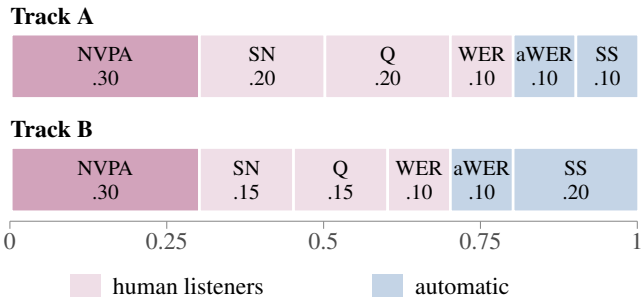


Fig. 2. Weights of the official score. Each bar sums to one. Magenta terms are judged by human listeners, with NVPA, the largest term, in a darker shade, and blue terms are computed automatically. Every term is normalized to $[0, 1]$, and error rates enter as one minus the rate.

and VRPO [25] estimates it for masked diffusion language models from masked-token log-probabilities, sharing masks between policy and reference to reduce variance. We adapt this estimator to multi-codebook masked TTS and build all pairs from human ratings, so the evaluation proxy never serves as the reward.

3. TASK, DATA AND BACKBONE

Scoring. The official score combines four terms judged by human listeners, namely NV placement accuracy (NVPA), speech naturalness (SN), audio quality (Q) and a word error rate measured in a listening test (WER), with two automatic terms, an ASR word error rate (aWER) from a Zipformer model [26] and speaker similarity (SS) from ECAPA-TDNN [27]. Fig. 2 shows the weights. NVPA carries the largest weight in both tracks, and Track B moves weight from naturalness and quality to speaker similarity, in line with its zero-shot setting.

Data. The train and dev splits contain 2,055 utterances at 24 kHz with inline tags and speaker IDs. The public test consists of 30 sentences read by 7 Track A and 4 Track B speakers, which gives 330 utterances with 88 occurrences of each tag. We follow the organizers' class definitions. Breathing is an ordinary audible intake or release of breath rather than an expressive sigh, and in the corpus it is often light and barely audible. Throat clearing is intentional, unlike a reflexive cough, and a sniff is a brief, audible nasal inhalation. Besides the pretrained backbone, the only external data in the submitted model are 60 throat-clearing utterances synthesized with VieNeu-TTS and accepted by listeners. The remaining models in this paper serve only for data preparation and evaluation, and neither test set is used for training. Both the backbone and the VieNeu-TTS data were declared to the organizers, as the task rules require.

Backbone selection. Since NVs can be added by fine-tuning but voice identity and audio quality are hard to repair, we looked for a backbone that clones voices well and generates clean audio. The survey (Table 1) shortlisted OmniVoice and Breeze-TTS-2, and we listened to every tag from these and other candidates against the corpus (Table 2). No candidate renders all four classes reliably.

Tags fire only intermittently, laughter comes with a spoken "haha", breathing becomes a sigh, and sniff turns into sobbing. Breeze-TTS-2 often hallucinates, whereas OmniVoice [12] gave the cleanest speech and the best cloning, so we use it as the backbone despite its weak NV support. Prompting it with corpus references that contain the target NV did not help either, since many of those NVs are mislabelled, for instance coughs tagged as throat clearing.

Backbone. The backbone has 0.6B parameters, which allows full fine-tuning (Fig. 3). The Higgs Audio v2 [28] tokenizer encodes 24 kHz audio into 8 residual codebooks at 25 frames/s. The 8 code embeddings of a frame are summed into one position of a bidirectional Qwen3 [29] transformer, whose input is a style prefix, the reference and target transcripts, the reference audio tokens and the masked target tokens. Eight heads predict all codebooks of masked positions in parallel, and inference unmasks them over 32 steps with classifier-free guidance [30]. Training minimizes masked-token cross-entropy with codebook weights $w = (8, 8, 6, 6, 4, 4, 2, 2)$. NV tags are plain text, so the vocabulary is unchanged.

4. METHOD

4.1. Stage 1: segmentation and listening

Single-NV segments. Corpus utterances are long and often contain several NVs, so one NV cannot be judged or kept without the others. We therefore cut each utterance in four steps. First, the transcript is split into words and tags, each tag is assigned to the gap between two words, and adjacent tags in the same gap form one NV group. Second, MMS_FA [31] aligns the words after Vietnamese diacritics are removed, and the time window of a tag runs from the end of the preceding word to the start of the following one. Third, each NV group yields a segment that runs from 50 ms after the previous group's window to 50 ms before the next one, or to the utterance boundary, with the words on both sides and the tags as its transcript. Each segment therefore holds exactly one NV group with its surrounding speech, and an utterance with several groups yields one overlapping segment per group. Finally, we drop segments shorter than 1 s or longer than 15 s and the 82 groups that mix two NV types. This gives 744 laughter, 4,267 breathing, 69 sniff and 117 throat-clearing segments over train and dev. Untagged spans give 4,592 bare segments, which serve as clean voice prompts with a 30–60% prompt ratio.

Listening. Annotators label segments Keep, Reject (wrong or inaudible NV, broken audio) or Unusable in a web tool that plays ± 1.2 s around the tag, optionally at $0.75\times$ speed. Beyond the official definitions, they separate throat clearing, produced with the mouth closed, from coughing, which opens it. We listened to all sniff and throat-clearing segments and to a sample of breathing (Table 3); laughter was taken as is. Only 17% of sniffs and 16% of throat clearings pass, so most corpus tags in the rare classes are unusable as training targets. To compensate, we mapped the native throat-clearing tag of 4,556 VieNeu-TTS [11] utterances to [throatclearing], listened to 409 and kept 60. Rare classes are repeated 4–8 times per epoch.

4.2. Stage 2: self-synthesis with blind rating

Three Stage-1 checkpoints (full fine-tuning at step 500; LoRA at steps 250 and 500) synthesize 400 training sentences with one tag each (100 per tag), using another utterance of the same speaker as reference and sampling with class temperature 1.0 and guidance 2.0 for diversity. Four raters graded the three candidates of each sentence as Good (correct NV, natural prosody, voice preserved), OK, or Bad (wrong or missing NV, voice drift, artefacts, long silences, wrong words). The page hides which checkpoint produced each clip, shuffles their order per sentence, and reveals the source only after all three clips are graded. One clip in three is Good (399 of 1,197), so three candidates per sentence give about one usable clip each. Good clips come from all three checkpoints (112–152 each), so pooling them adds more variety than any single checkpoint would. All Good clips join the Stage-1 data with four-fold repetition, and both variants are retrained from the original pretrained checkpoint. A preliminary run that also added the OK clips performed worse, so from then on we use only Good audio, both as training data and as preferred samples in Stage 3.

4.3. Stage 3: DPO for masked TTS

Why preference optimization. The synthesized candidates are inconsistent in a way that more supervised data does not fix. For the same tag, a model sometimes renders the NV, sometimes skips it and sometimes coughs instead of clearing its throat. Listener ratings capture exactly these failures, so we turn them into preference pairs. Raters also checked the waveform, where throat clearing shows two short bursts, a sniff a brief high peak and breathing a sustained, even envelope, and they counted hallucinated silences against a clip. Preferring clips that pass these checks should make the NVs both more reliable and closer to the class definitions.

Preference pairs. Within a sentence, a Good clip is preferred over every OK or Bad clip; we drop pairs between OK and Bad clips so that the preferred sample is always acceptable. Stage-2 ratings give 436 pairs. A second round on the best Stage-2 model (two decodings per sentence, 166 of 400 sentences fully rated by three raters) adds 47, for 483 pairs (291 against Bad, 192 against OK), split by speaker into 427 training and 56 validation pairs.

Objective. DPO [2] compares a trainable policy π_θ with a frozen reference π_{ref} on pairs of clips that share the same conditioning x, namely the text and the reference audio. With y_w the preferred and y_l the rejected clip, it minimizes

$$\mathcal{L} = -\log \sigma\big(\beta\,[\Delta(y_w) - \Delta(y_l)]\big), \qquad (1)$$
$$\Delta(y) = \log \pi_\theta(y \mid x) - \log \pi_{\text{ref}}(y \mid x).$$

The loss falls when the policy raises the likelihood of the preferred clip more than that of the rejected one, measured relative to the reference, and β limits how far the policy may drift from the reference.

Scoring a clip with a masked model. An autoregressive model obtains $\log \pi(y \mid x)$ by summing its next-token log-probabilities. A masked model has no such factorization, since it only predicts hidden tokens from visible ones. We therefore score a clip the way the model is trained, by hiding part of its tokens and measuring how well the model recovers them. A clip is a grid y of 8 codebooks by T frames. For each of $K = 4$ draws, we hide a random set M_k of positions, with the hidden fraction spread evenly over $[0.1, 1]$ across the draws, and compute

$$s_\theta(y) = \frac{N}{K} \sum_{k=1}^{K} \sum_{c=1}^{8} \frac{\hat{w}_c}{|M_{k,c}|} \sum_{j \in M_{k,c}} \log p_\theta\big(y_{c,j} \mid x, y_{\setminus M_k}\big), \qquad (2)$$

where $M_{k,c}$ is the hidden part of codebook c, $\hat{w}_c$ weights the codebooks as in training, and $N = 8T$ is the number of tokens in the clip. The inner sum is the average log-probability of the hidden tokens of one codebook, and $s_\theta(y)$ replaces $\log \pi_\theta(y \mid x)$ in Eq. (1), with $s_{\text{ref}}(y)$ computed in the same way.

Two choices matter in practice. First, multiplying by N gives s_θ the scale of a whole-clip log-likelihood. With a plain per-token average, differences in Δ are only about 10^{-2}, β has to be raised to about

	Laughter	Breathing	Sniff	Throat clearing
Corpus (target)	crisp, loud burst	quick, quiet intake	sharp, audible intake	clear, but often a cough
OmniVoice	light and short	sigh instead	not produced	not produced
Higgs	with a spoken “haha”	correct, but intermittent	not produced	clear, on target
VieNeu-TTS	not produced	exhaled sigh	sobbing instead	hallucinated or missing
Breeze-TTS-2	light, sounds like sniff	correct, but intermittent	present, but too weak	mixed with coughs

Table 2. Listening check of the corpus NVs and of candidate models before fine-tuning.

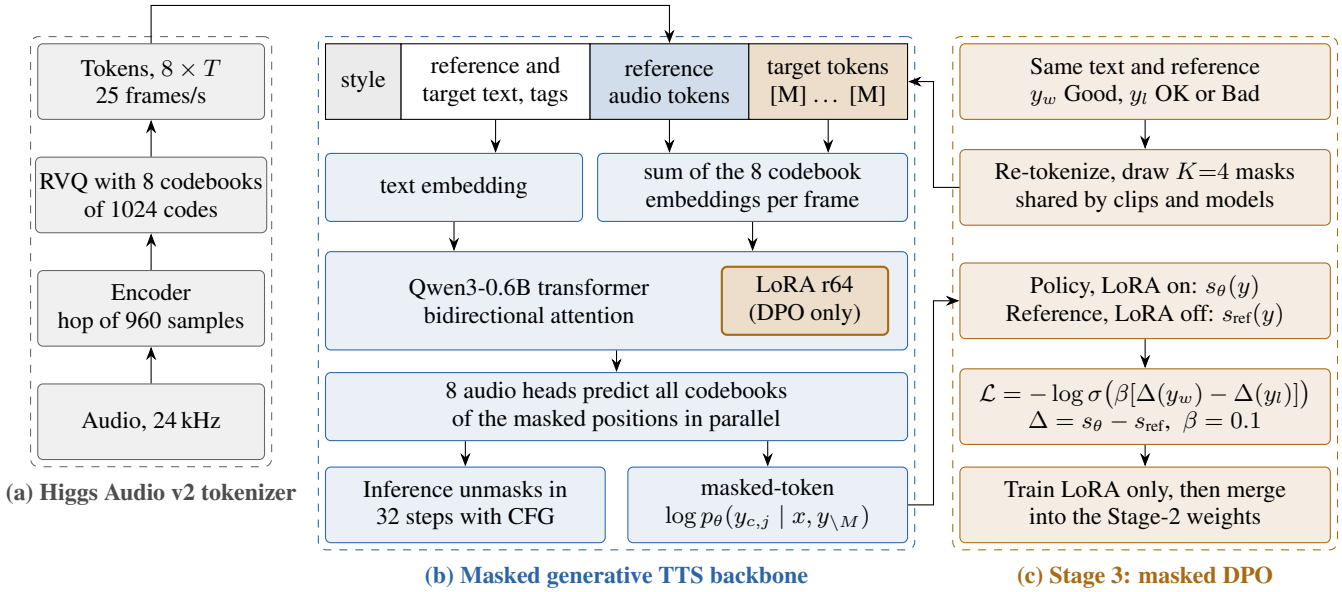


Fig. 3. Model and DPO attachment. (a) The frozen audio tokenizer. (b) The backbone, which predicts all codebooks of the masked target positions in parallel. (c) DPO, where both clips of a pair are scored with Eq. (2) under shared masks, and the Stage-2 model with and without a LoRA adapter serves as policy and reference.

	Stage 1 curation				Stage 2 rating		
Class	Segs	Heard	Kept	Rep	Good	OK	Bad
laughter	647	–	647	×1	92	101	107
breathing	4,267	212	89	×4	92	60	148
sniff	69	69	12	×8	119	88	93
throat.	117	117	19	×4	96	64	137
+ VieNeu	4,556	409	60	×4			
bare	3,887	–	500	×1	–	–	–
total					399	313	485

Table 3. Training data. Laughter and bare segments come from the train split, with dev held out, and the other classes use train and dev. An epoch has 1,915 samples in Stage 1 and 3,511 in Stage 2, which adds the 399 Good clips four times.

20, and a pilot run gained less NVPA. Second, as in VRPO [25], the policy and the reference, and both clips of a pair, use the same mask draws, so the randomness of masking largely cancels in $\Delta(y_w) - \Delta(y_l)$.

Policy and reference. The policy is the Stage-2 model with a rank-64 LoRA adapter, with the audio embeddings and heads also trainable. Switching the adapter off recovers the Stage-2 model, which serves as the reference, so only one network is held in memory (Fig. 3). We use $\beta = 0.1$. Because the clips are re-tokenized from their final waveforms, y only approximates the tokens that the model sampled.

5. EXPERIMENTS

Training. Stages 1 and 2 train two variants for 500 steps on 2×H100 with bf16, 16,384-token batches, 2% warm-up and seed 42. Variant (a) is full fine-tuning with learning rate 10^{-5}. Variant (b) is LoRA [32] with rank 64 and $\alpha = 128$ on all attention and MLP projections plus the audio embeddings and heads (57M of 670M parameters), with learning rate 10^{-4}, merged after training. DPO runs 300 steps of 8 pairs on one H100, learning rate $5 \cdot 10^{-5}$ with cosine decay; adapters at steps 100/200/300 are merged for evaluation. All systems decode identically, with 32 steps, guidance 2.0, greedy class sampling, the denoising token and a fixed seed.

Evaluation. The human-judged terms cannot be reproduced during development, so we select checkpoints with a proxy score that keeps the automatic terms of the official formula, substitutes our own estimate for the human NVPA and renormalizes the weights per track. aWER is computed with the organizers’ Zipformer model after tags and punctuation are removed, and SS is the ECAPA cosine similarity to the reference clip. To estimate NVPA, each tag defines a slot

	System	NVPA↑	Lau	Bre	Sni	Thr	AudioSet↑	aWER↓	SS↑	Sil.↓	Extra↓	Proxy A↑	Proxy B↑
	Base model	.136	.07	.34	.07	.07	13.9%	5.87%	**.756**	–	3.40	.443	.451
S1	full, ck500	.656	.24	.65	.84	.90	–	2.86%	.747	16.1%	–	.761	.708
	LoRA, ck500	.685	.28	.74	.77	.94	48.6%	3.01%	.743	11.5%	3.19	.774	.726
S2	full, ck500	.742	.40	.69	.93	.94	56.5%	3.02%	.748	12.1%	**3.18**	.804	.764
	LoRA, ck500	.710	.28	.64	.92	**1.00**	54.0%	3.12%	.743	9.1%	–	.774	.760
S3	DPO ck100	.847	.67	.82	**.97**	.93	77.0%	2.82%	.719	**3.6%**	4.11	.849	.829
	DPO ck200	**.875**	**.84**	.85	.84	.97	**85.8%**	2.77%	.682	3.9%	4.38	**.865**	.824
	DPO ck300†	**.875**	.81	**.86**	.84	.99	83.8%	**2.71%**	.678	4.2%	4.27	.853	**.838**

Table 4. Results on the public test (330 utterances, Tracks A and B). NVPA is our probe-based estimate, overall and per tag with 88 tags each. AudioSet is the share of tags detected by the original AudioSet classifier, which uses no probe. Sil. is the share of utterances with an internal pause longer than 0.6 s, and Extra counts NVs detected at untagged slots per utterance. Proxy A and B apply the official weights to NVPA, aWER and SS, computed per track. S3 starts from S2 full. †Submitted.

between two force-aligned words. AST [33] embeddings over 0.5 s windows with a 0.1 s hop are max- and mean-pooled over the slot widened by 0.1 s on each side, the log slot length is appended, and a class-balanced, L2-regularized logistic regression assigns the slot to one of the four NVs or to none. A tag counts as correct if its own class receives the highest probability. The probe is trained only on real recordings from 400 train utterances and the dev set, with slot width and regularization chosen by speaker-grouped 5-fold cross-validation, and moving every tag to the adjacent slot or relabelling it must make the hit rate collapse. Because we train the probe ourselves on corpus audio, and because it is weakest on laughter, we also report AudioSet detection, which involves no trained probe. A tag counts as detected if the original AudioSet [34] head of AST assigns a probability above 0.1, anywhere in the widened slot, to a label from the tag's sound family. These families are Laughter, Giggle and Chuckle for laughter, Breathing, Sigh, Gasp and Pant for breathing, Sniff and Snort for sniff, and Throat clearing and Cough for throat clearing. The two measures answer different questions. NVPA asks whether the tagged class wins against the other three NVs and silence at that slot, so it penalizes a wrong type as well as a missing NV. AudioSet detection only asks whether a sound of the right family is present, so it accepts a sigh for breathing or a cough for throat clearing. It shares the AST encoder with the probe but neither the probe's training data nor its decision rule, so agreement between the two rules out gains that come from the probe alone.

5.1. Results

All systems in Table 4 use the same references, seeds and decoding.

Curation matters more than quantity. After Stage 1 the model produces all four classes, and the classes we filtered most strictly learn best. Sniff, trained on only 12 kept clips, reaches 0.77–0.84 and is almost never mistaken for breathing (3 of 88). Laughter, the largest class but the only one we did not listen to, stays at 0.24–0.28. Breathing, the quietest class, remains the hardest to detect. Fine-tuning also halves aWER, from 5.87% to 3.01%.

Self-synthesis helps where real data is scarce. The 399 Good clips raise NVPA by 0.06 at no cost in aWER or SS, and the gain concentrates on the classes with the fewest or noisiest real examples, with sniff gaining 0.16 and laughter 0.12. Breathing loses 0.05, the only class that gets worse. With the larger data set, full fine-tuning overtakes LoRA (0.742 vs. 0.710), the reverse of Stage 1.

DPO makes NVs reliable. DPO gives the largest gain of the three stages, raising NVPA from 0.742 to 0.875 by step 200, and it targets the failures the raters penalized. Laughter, the weakest class so far, rises from 0.40 to 0.81–0.84 after 200 steps, and utterances with long hallucinated pauses become about three times rarer (3.6–4.2% vs. 12.1%). AudioSet detection rises as well, from 56.5% to 77–86%. Human labels matter here. In a pipeline test, 582 pairs labelled by the probe itself pushed NVPA to 0.889, which shows how easily the proxy can be optimized directly and why we never let it pick training data. Although the adapter memorizes its pairs, with training accuracy reaching 1.0 by step 70 while validation accuracy stays at 0.68 from step 50, the public-test gains show that what it learns transfers.

DPO trades voice identity for NV reliability. SS falls with every DPO step, from 0.748 before DPO to 0.719, 0.682 and 0.678 at steps 100, 200 and 300. The loss is concentrated on seen speakers, where SS falls from 0.732 to 0.645 on Track A at step 300, against 0.777 to 0.737 for the unseen speakers of Track B. DPO also adds NVs at untagged slots (about one more per utterance), though part of this count is probe false positives, as the base model already scores 3.4.

Submitted checkpoint. The official score weights NVPA three times as much as SS in Track A and 1.5 times as much in Track B, so the NV gain outweighs the SS loss, and every DPO checkpoint beats the Stage-2 model on both proxies. We submitted ck300. It ties ck200 for the best NVPA, has the lowest aWER and the best Track B proxy, and places NVs best for unseen speakers (Track B NVPA 0.859 against 0.828 for ck200). Compared with ck200, it gives up 0.012 on the Track A proxy for 0.014 on Track B, where zero-shot cloning makes NV placement hardest.

Limitations. All results are on the public test, which we also used to select checkpoints, so they are likely optimistic. Private-test results had not been released when this report was written. Each Stage-2 sentence was rated by one team member, without measured inter-rater agreement. The agreement of our NVPA estimate with the human NVPA is unknown, and the other human-judged terms, which carry half of the Track A score and 40% of the Track B score, are outside our selection criterion. LoRA runs are not bit-exact across reruns (Stage-1 NVPA 0.685 vs. 0.705).

6. CONCLUSION AND FUTURE WORK

With scarce and noisy NV data, human listening proved the most useful training signal, first as a filter on corpus and synthesized clips and then, through preference pairs, as the source of the largest

improvement. Several directions remain untested. (i) Keep NV gains without losing speaker identity, e.g. by adding an SS term to the preference objective or to the rating criteria. (ii) Denoise the reference audio with a speech-enhancement model at inference, since background noise in podcast references is cloned into the output. (iii) Choose the reference clip per NV tag at inference, e.g. a listener-approved clip of the same speaker that contains the target NV, to prime its realization. (iv) Filter the unlistened laughter class, e.g. by removing quiet, weak laughs automatically. (v) Mix English NV data such as NonverbalTTS [9] into training, or start from pretrained models with native NV tags such as Dia/Dia2 [4] and Chatterbox-Turbo [6], to transfer NVs across languages. Code, listener labels and evaluation scripts will be shared privately with the organizers for verification.

7. COMPLIANCE WITH ETHICAL STANDARDS

All listening and rating was done by volunteering team members, and no personal data were collected. Training audio comes from the ViNV-TTS corpus under the organizers' usage agreement and from synthetic VieNeu-TTS utterances. Because the system clones voices and could be misused for impersonation, neither the model nor the code is released publicly.

8. REFERENCES

[1] VLSP, "VLSP 2026 shared task: Vietnamese non-verbal text-to-speech for conversational synthesis," https://vlsp.org.vn/vlsp2026/eval/vinvtts, 2026.

[2] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn, "Direct preference optimization: Your language model is secretly a reward model," in *Proc. NeurIPS*, 2023.

[3] Zhihao Du et al., "CosyVoice 2: Scalable streaming speech synthesis with large language models," *arXiv preprint arXiv:2412.10117*, 2024.

[4] Nari Labs, "Dia: A TTS model for dialogue generation," https://github.com/nari-labs/dia, 2025.

[5] Canopy Labs, "Orpheus TTS," https://github.com/canopyai/Orpheus-TTS, 2025.

[6] Resemble AI, "Chatterbox TTS," https://github.com/resemble-ai/chatterbox, 2025.

[7] Fish Audio, "Fish Audio S2 Pro," https://huggingface.co/fishaudio/s2-pro, 2026.

[8] Boson AI, "Higgs TTS 3," https://huggingface.co/bosonai/higgs-audio-v3-tts-4b, 2026.

[9] Maksim Borisov, Egor Spirin, and Daria Diatlova, "NonverbalTTS: A public English corpus of text-aligned nonverbal vocalizations with emotion annotations for text-to-speech," in *Proc. 13th ISCA Speech Synthesis Workshop (SSW)*, 2025, pp. 104–109.

[10] R. Ye et al., "A scalable pipeline for enabling non-verbal speech generation and understanding," *arXiv preprint arXiv:2508.05385*, 2025.

[11] pnnbao97, "VieNeu-TTS," https://github.com/pnnbao97/VieNeu-TTS, 2025.

[12] k2-fsa, "OmniVoice: Towards omnilingual zero-shot text-to-speech," *arXiv preprint arXiv:2604.00688*, 2026.

[13] OpenBMB, "VoxCPM2," https://huggingface.co/openbmb/VoxCPM2, 2026.

[14] BreezeBlue, "Breeze-TTS-2," https://huggingface.co/BreezeBlue/Breeze-TTS-2, 2026.

[15] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman, "MaskGIT: Masked generative image transformer," in *Proc. CVPR*, 2022.

[16] Zalán Borsos et al., "SoundStorm: Efficient parallel audio generation," *arXiv preprint arXiv:2305.09636*, 2023.

[17] Yuancheng Wang et al., "MaskGCT: Zero-shot text-to-speech with masked generative codec transformer," in *Proc. ICLR*, 2025.

[18] Yushen Chen et al., "F5-TTS: A fairytaler that fakes fluent and faithful speech with flow matching," *arXiv preprint arXiv:2410.06885*, 2024.

[19] Dong Zhang et al., "SpeechAlign: Aligning speech generation to human preferences," in *Proc. NeurIPS*, 2024.

[20] Chen Chen et al., "Enhancing zero-shot text-to-speech synthesis with human feedback," *arXiv preprint arXiv:2406.00654*, 2024.

[21] Jinchuan Tian et al., "Preference alignment improves language model-based TTS," in *Proc. ICASSP*, 2025.

[22] Shehzeen Hussain et al., "Koel-TTS: Enhancing LLM based speech generation with preference alignment and classifier free guidance," *arXiv preprint arXiv:2502.05236*, 2025.

[23] Philip Anastassiou et al., "Seed-TTS: A family of high-quality versatile speech generation models," *arXiv preprint arXiv:2406.02430*, 2024.

[24] Bram Wallace et al., "Diffusion model alignment using direct preference optimization," in *Proc. CVPR*, 2024.

[25] Fengqi Zhu, Rongzhen Wang, et al., "LLaDA 1.5: Variance-reduced preference optimization for large language diffusion models," in *Proc. ACL*, 2026.

[26] Zengwei Yao et al., "Zipformer: A faster and better encoder for automatic speech recognition," in *Proc. ICLR*, 2024.

[27] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," in *Proc. Interspeech*, 2020.

[28] Boson AI, "Higgs Audio v2," https://github.com/boson-ai/higgs-audio, 2025.

[29] An Yang et al., "Qwen3 technical report," *arXiv preprint arXiv:2505.09388*, 2025.

[30] Jonathan Ho and Tim Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598*, 2022.

[31] Vineel Pratap et al., "Scaling speech technology to 1,000+ languages," *Journal of Machine Learning Research*, vol. 25, 2024.

[32] 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 *Proc. ICLR*, 2022.

[33] Yuan Gong, Yu-An Chung, and James Glass, "AST: Audio spectrogram transformer," in *Proc. Interspeech*, 2021.

[34] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," in *Proc. ICASSP*, 2017.