

# Instrumental Text-to-Music Generation with Auxiliary Conditioning Branches

Junyoung Koh

*Department of Artificial Intelligence, Yonsei University*

*MAAP*

*KRAFTON*

Seoul, Republic of Korea

solbon1212@yonsei.ac.kr

**Abstract**—Text-to-music generation has advanced rapidly, with modern autoregressive and diffusion-based models producing convincing music from natural-language prompts. However, much of this progress relies on large-scale training data and external pretraining, making it difficult to isolate which design choices remain effective when data and pretraining are controlled. We study this setting using a Diffusion Transformer backbone with lyric and timbre conditioning, adapted to an instrumental-only text-to-music task in which the auxiliary lyric and timbre branches receive only degenerate conditioning signals. Through controlled ablations, we find that models retrained without these branches score lower across AudioBox aesthetics, LLM-as-judge, and human MOS, and that reinvesting the saved parameters as additional DiT depth recovers only marginally. This suggests the auxiliary branches may act as training-time architectural anchors whose contribution goes beyond their explicit conditioning content. We validate the same model through comparisons with external instrumental baselines and through our submission to the ICME 2026 Academic Text-to-Music (ATTM) Grand Challenge, where our Performance submission ranked first under both the objective metrics and the subsequent organizer-administered MOS over 35 raters, attaining the highest overall MOS across all challenge submissions, while our Efficiency submission was a finalist that tied for second under the objective metrics.

**Index Terms**—text-to-music generation, diffusion transformer, flow matching, training strategies, classifier-free guidance

## I. INTRODUCTION

Generative modeling of music conditioned on text has advanced substantially, with autoregressive models such as MusicGen [1] and MusicLM [2] and diffusion-based models such as AudioLDM2 [3], Stable Audio [4], and MeLoDy [5] now able to synthesize convincing instrumental audio from natural-language prompts. This progress is driven by stronger backbones such as the Diffusion Transformer [6] with flow matching [7], and by richer LLM-generated captions [8], [9]. However, much of this progress relies on large-scale training data and external pretraining, making it difficult to isolate which design choices remain effective once data and pretraining are controlled.

We study this setting using a Diffusion Transformer backbone with lyric and timbre conditioning, specifically the ACE-STEP 1.5 architecture [10], adapted to an instrumental-only text-to-music task. ACE-STEP 1.5 was developed for vocal song generation, where its lyric and timbre encoders

consume real lyrics and a reference vocal or instrumental timbre. Whether such a vocal-centric backbone transfers cleanly to instrumental-only generation, and whether its auxiliary branches still play a useful role under degenerate inputs, is not obvious a priori.

Through controlled architectural ablations, we find that retraining the model without these branches consistently degrades perceptual quality measured by AudioBox aesthetics [11], an LLM-as-judge protocol, and single-rater human MOS, while validation MSE and CLAP-style scores are much less sensitive to the change. Inference-time zeroing of either branch is much milder, suggesting that the encoder contribution is realized primarily during training. Even when we reallocate the parameters saved by removing the branches into additional DiT depth, so that the resulting model matches the original parameter count, the gap only partially closes, suggesting that the auxiliary branches provide an architectural benefit beyond their explicit conditioning content.

We validate these findings through comparisons with external instrumental text-to-music baselines, namely Stable Audio Open and three sizes of MusicGen. We additionally submit our model to the ICME 2026 Academic Text-to-Music (ATTM) Grand Challenge [12], where two tracks, an Efficiency Track capped at 500M parameters and an unrestricted Performance Track, share the same training dataset, a 457-hour subset of MTG-Jamendo [13], and the same set of allowed pretrained components. Building on the ACE-STEP backbone, we combine Min-SNR- $\gamma$  loss weighting [14], adaptive timestep sampling, random segment cropping, and post-hoc EMA averaging [15] on the training side, and a guidance interval [16] with a tuned EMA window on the inference side; complementary training-time and inference-time ablations quantify each component’s contribution. We apply this recipe per track, training a 499M DiT for the Efficiency Track and a 2.4B DiT for the Performance Track. Our Performance model ranked first under both the objective metrics and the subsequent organizer-administered MOS, and our Efficiency model was a finalist that tied for second under the objective metrics.

## II. RELATED WORK

**Text-to-Music Generation.** Existing approaches fall into two categories. Autoregressive models such as MusicGen [1] and MusicLM [2] generate discrete audio tokens, while diffusion-based models such as AudioLDM2 [3], Stable Audio [4], and MeLoDy [5] operate in continuous latent spaces. Recently, ACE-STEP [10] has combined a Diffusion Transformer [6] with flow matching [7] in a compressed audio latent space, achieving strong results with a modular architecture.

**Audio Captioning for Generation.** Caption quality and style significantly impact text-conditioned generation. LLM-generated captions [17], [18] have emerged as an effective way to augment music datasets, first shown by LP-MusicCaps [8] and later scaled by JamendoMaxCaps [9], which provides 362K Jamendo captions produced with Qwen2-Audio. The ATTM challenge follows the same trend by releasing its own official Qwen2-Audio captions, which ensures style consistency between training captions and test prompts.

**Diffusion Training Strategies.** Several techniques have been proposed to improve diffusion model training. Min-SNR- $\gamma$  weighting [14] addresses timestep imbalance by clamping loss weights based on signal-to-noise ratio. EDM2 [15] introduces post-hoc EMA reconstruction for improved checkpoint selection without retraining. Guidance interval [16] restricts classifier-free guidance to mid-range timesteps, improving both sample quality and distribution fidelity.

## III. METHOD

We adopt the ACE-STEP 1.5 architecture [10] as our backbone, a strong open-source music DiT model whose modular design consists of an audio VAE, a text encoder, a small condition encoder, and a DiT decoder. We train two instances at different scales, one per track: a 499M Efficiency model that uses 99.8% of the Efficiency Track’s 500M parameter cap, and a 2.4B Performance model. Both models use the same training and inference strategies; full configurations are in Table I.

**Frozen pretrained components.** We use a frozen AutoencoderOobleck [4] as the audio VAE; it maps 48 kHz stereo audio to 25 Hz, 64-channel latents at 1920:1 temporal compression. A frozen Qwen3-Embedding-0.6B [19] serves as the text encoder, producing 1024-dimensional embeddings.

**Trained components.** The ATTM challenge is instrumental-only, so there are no real lyric or reference-audio inputs available to ACE-STEP’s lyric and timbre branches. We therefore feed the lyric branch a single placeholder token “[Instrumental]” and the timbre branch a zero tensor. The condition encoder is a small transformer stack that further processes three conditioning streams, namely text, lyric, and timbre, and packs them into a single sequence for cross-attention. The lyric branch uses 6 layers in the Efficiency model and 8 in the Performance model, and the timbre branch uses 3 and 4 layers respectively. The lyric and timbre sub-modules therefore see constant inputs at every step. The DiT decoder is the main generative model, with alternating full and sliding-window attention layers, Grouped Query Attention [20], Adaptive Layer Normalization [6]

for timestep conditioning, and patch-based input and output projections with stride-2 convolutions.

TABLE I  
MODEL CONFIGURATIONS FOR BOTH TRACKS

	Efficiency	Performance
Hidden / Intermediate	1152 / 3072	2048 / 6144
DiT layers	16	24
Attention heads (KV heads)	9 (3)	16 (8)
Head dimension	128	128
Lyric encoder layers	6	8
Timbre encoder layers	3	4
Sliding window size	128	128
<b>Total parameters</b>	<b>499.0M</b>	<b>2,419.1M</b>
Batch size $\times$ accumulation	16 $\times$ 4	8 $\times$ 8
Effective batch size	64	64
Training epochs	120	120
Hardware	1 $\times$ A100-80GB	1 $\times$ A100-80GB
Approx. GPU-hours	$\sim$ 20h	$\sim$ 60h

## IV. TRAINING STRATEGIES

**Min-SNR- $\gamma$  loss weighting.** Standard flow matching [7] weights every timestep equally, which over-emphasizes easy high-SNR steps and under-uses informative low-SNR ones [14]. This imbalance is especially costly in our limited-data regime, where every informative timestep matters. We therefore apply Min-SNR- $\gamma$  weighting with  $\gamma$  of 5.0,  $w(t) = \min(\text{SNR}(t), \gamma) / \text{SNR}(t)$  where  $\text{SNR}(t) = ((1-t)/t)^2$ , which clamps easy-step weights while keeping full weight on the hard ones.

**Adaptive timestep sampling.** Min-SNR- $\gamma$  reweights each *drawn* timestep, but leaves the sampling distribution itself fixed. We additionally adapt the distribution to the model’s current loss profile: we discretize  $[0, 1]$  into 100 bins, track an EMA with  $\beta = 0.99$  of per-bin losses, and sample bins proportional to their temperature-scaled loss EMA. Before enough statistics accumulate, we fall back to a logit-normal default with  $\mu = -0.4$  and  $\sigma = 1.0$ . The two techniques are complementary: one rebalances each draw’s weight, the other rebalances how often each timestep is drawn at all.

**Random segment cropping.** The training set is 30-second clips, but generation targets 10 seconds. We randomly crop a 10-second window of 250 latent frames at 25 Hz at every step, which provides about  $3\times$  augmentation and exposes the model to diverse temporal content per track across epochs.

**Post-hoc EMA checkpoint averaging.** We save model snapshots every 1,000 steps throughout training. After training, we identify a late-training window in which validation loss has stabilized and uniformly average all snapshots inside it, following EDM2 [15]. For the Efficiency model we use steps 70K through 96K, which corresponds to roughly the last 30 epochs, since averaging only converged snapshots avoids dragging the model toward early under-trained weights.

**Stability and optimization.** We clamp each per-sample MSE at 20.0 before averaging, so a single outlier cannot dominate a batch update; outliers are still trained on but their

gradient contribution is bounded. Optimization uses AdamW with learning rate  $3 \times 10^{-4}$ , weight decay 0.01, and  $\beta = (0.9, 0.95)$ , applied with a 200-step linear warmup followed by cosine annealing. Both models are trained for 120 epochs with gradient checkpointing, and we use a 15% classifier-free guidance dropout rate.

## V. INFERENCE STRATEGIES

**ODE sampling with a guidance interval.** We generate with 100-step Euler ODE sampling and apply classifier-free guidance, abbreviated CFG [21], only inside a *guidance interval* [16]  $t \in [0.1, 0.9]$ .

**CFG scale strategy.** The CFG scale produces a well-known trade-off between FAD and CLAP: higher scales sharpen the conditional distribution and improve prompt adherence but can over-constrain the output. Instead of committing to a single CFG scale, we sweep CFG on a held-out development set and select two operating points per track that bracket the trade-off curve.

## VI. DATA AND CAPTION ANALYSIS

### A. Training Data

We use exclusively the MTG-Jamendo *raw\_30s* subset with organizer-provided vocal separation, which contains approximately 55,700 tracks for a total of 457 hours, as released for the challenge in [12]. Audio is preprocessed into VAE latents and text embeddings are cached as .pt files. A 2% random split is held out for validation.

### B. Caption Strategy Comparison

Caption quality proved to be the most impactful factor in our experiments. We evaluated three caption strategies, summarized in Table II.

TABLE II

IMPACT OF CAPTION STRATEGY ON VAL LOSS (EFFICIENCY 414M, 60 EPOCHS).  $\Delta$  IS VAL LOSS RELATIVE TO THE TEMPLATE BASELINE.

Caption Source	Val Loss	$\Delta$
Template (tag-to-text rules)	0.968	–
JamendoMaxCaps (42% match)	0.943	–0.025
Official Qwen2-Audio (100%)	0.943	–0.025

Template captions are tag-derived sentences obtained by applying simple rules to the MTG-Jamendo tag set, for example “an atmospheric rock track with guitar,” whereas Qwen2-Audio captions are detailed paragraphs describing genre, tempo, mood, and instrumentation. Crucially, the challenge’s test prompts are also generated by Qwen2-Audio, so using Qwen2-Audio training captions aligns the train and test text distributions. JamendoMaxCaps, also produced by Qwen2-Audio, reaches the same val loss but covers only 42% of our training tracks, so we use the official organizer-provided captions for full coverage. The  $-0.025$  val-loss improvement from switching to LLM-generated captions exceeds the gain from any single training technique we tested, underscoring the importance of aligning the training and test text distributions.

## VII. EXPERIMENTAL RESULTS AND DISCUSSION

### A. Training Convergence

Both models converge smoothly over 120 epochs, with a stable training and validation generalization gap of 0.29 to 0.31 that indicates no severe overfitting, as shown in Fig. 1.

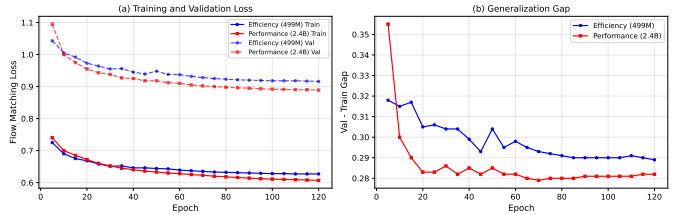


Fig. 1. (a) Training and validation loss for both models over 120 epochs. (b) The generalization gap (val – train) remains stable, indicating controlled overfitting.

### B. Challenge Results

Table III reports the organizer’s final-test evaluation on the 100 official prompts, with our two submitted entries (bold) listed alongside the top Efficiency-Track entries by MOS, the runner-up Performance entry, and the organizer-released MusicGen-small baseline.

TABLE III

FINAL-TEST RESULTS ON THE 100 OFFICIAL PROMPTS.  $MOS_{all}$  AND  $MOS_{exp}$  AGGREGATE OVER ALL 35 RATERS AND THE 20-EXPERT SUBSET; RANK IS BY  $MOS_{all}$  WITHIN EACH TRACK. **BOLD ROWS ARE OUR ENTRIES.**

Entry	FAD↓	CLAP↑	CCS↑	$MOS_{all}$ ↑	$MOS_{exp}$ ↑	Rank
1st in Efficiency Track by MOS (e07)	0.417	0.261	0.867	3.250	3.186	1
2nd in Efficiency Track by MOS (e01)	0.577	0.338	0.863	3.225	3.177	2
3rd in Efficiency Track by MOS (e08)	–	–	–	3.119	3.044	3
<b>Ours, Efficiency 499M (e05, CFG 8.0)</b>	<b>0.487</b>	<b>0.305</b>	<b>0.800</b>	<b>2.969</b>	<b>2.929</b>	<b>4</b>
<b>Ours, Performance 2.4B (p05, CFG 7.0)</b>	<b>0.514</b>	<b>0.306</b>	<b>0.800</b>	<b>3.344</b>	<b>3.327</b>	<b>1</b>
2nd in Performance Track by objective (p00)	0.557	0.311	0.796	–	–	–
MusicGen-small (organizer baseline)	–	–	–	3.538	3.425	–

**Dry-run.** On the organizer’s dry-run evaluation, our 414M Efficiency variants scored FAD 0.51–0.56 / CLAP 0.31–0.32 at CFG 5.0/7.0 and our 2.4B Performance scored FAD 0.518 / CLAP 0.310, against the best dry-run team at FAD 0.404 / CLAP 0.323; we then scaled the Efficiency model from 414M to 499M for the final submission to fully use the parameter budget.

**Final test.** The organizers selected CFG of 8.0 for our Efficiency entry and CFG of 7.0 for our Performance entry; the Efficiency choice sits comfortably on the CLAP plateau identified for that model in Fig. 2. Our 2.4B Performance entry achieves the top CCS in the Performance Track and the lowest FAD among Performance entries while CLAP stays within 0.005 of the next model. Our 499M Efficiency entry ties for 2nd by objective metrics.

**Post-challenge MOS.** Under the organizer-administered MOS (Table III, last two columns), our Performance submission attained the highest Overall score of any challenge entry in either track and won the Performance Track, while our

Efficiency submission trailed the three Efficiency winners. The organizer-released MusicGen-small baseline outscored every submission, mirroring the training-data-scale gap also visible in our single-rater MOS (Sec. VII-H).

**Self-eval.** Self-evaluation on the same 100 prompts with `msclap` CLAP and MERT-based FAD against an FMA-Medium reference we sample ourselves (absolute values not directly comparable to the organizer scale) gives Efficiency 499M FAD 6.63/6.90 and CLAP 0.495/0.495 at CFG 7.0/8.0, and Performance 2.4B FAD 7.59/8.61 and CLAP 0.475/0.477 at CFG 7.0/10.0.

### C. Incremental Strategy Ablation

Table IV shows the cumulative impact of each strategy on the Efficiency track model.

TABLE IV  
INCREMENTAL STRATEGY ABLATION (EFFICIENCY TRACK).  $\Delta$  IS VAL LOSS RELATIVE TO THE BASELINE. BOLD MARKS THE LOWEST VAL LOSS REACHED.

Configuration	Val Loss	$\Delta$
Baseline (template captions, self-processed audio, 60ep)	0.968	–
+ Official vocal-separated audio	0.960	–0.008
+ Official Qwen2-Audio captions	0.943	–0.025
+ Post-hoc EMA averaging	0.938	–0.030
+ Scale 414M $\rightarrow$ 499M	<b>0.916</b>	<b>–0.052</b>
+ Extended to 120 epochs	<b>0.916</b>	<b>–0.052</b>

Measured incrementally on top of the previous row, the largest single step is model scaling at  $-0.022$ , followed by caption quality at  $-0.017$  incrementally or  $-0.025$  versus the template baseline in Table II, official audio at  $-0.008$ , and EMA at  $-0.005$  with zero additional training cost.

### D. Component-Wise Training Ablation

To isolate the contribution of each training-strategy component, we additionally train one variant per component on the Efficiency 499M model with exactly that component disabled while the others are held fixed, for up to 30 epochs.

TABLE V  
COMPONENT-WISE TRAINING ABLATION, STEP-ALIGNED (EFFICIENCY 499M).  $\Delta$  IS VAL LOSS RELATIVE TO THE BASELINE AT THE MATCHED STEP; POSITIVE MEANS THE DISABLED COMPONENT WAS HELPING.

Disabled component	Val Loss	$\Delta$ (vs. base)
CFG dropout 0.15 $\rightarrow$ 0.05	0.962	+0.032
CFG dropout 0.15 $\rightarrow$ 0.30	0.961	+0.029
Min-SNR- $\gamma$ ( $\gamma=5 \rightarrow 0$ )	0.960	+0.026
Adaptive timestep sampling off	0.951	+0.020
Random crop off (fixed center)	0.892	–0.041

All four cleanly comparable components produce a strictly worse model when disabled, confirming that each contributes positively. The chosen CFG dropout of 15% is a sweet spot: both halving the rate to 5%, with  $\Delta = +0.032$ , and doubling it to 30%, with  $\Delta = +0.029$ , hurt by similar amounts. The

random-crop row is for completeness only: val uses the same crop policy, so disabling it makes val deterministic (fixed first 250 frames) and intrinsically easier.

### E. Inference Ablations

We re-evaluate our Efficiency 499M model under three controlled inference-time changes, summarized in Table VI, using the same checkpoint and 100 final-test prompts each time. Removing the guidance interval and applying CFG over the full  $[0, 1]$  drops CLAP by  $-0.031$ , which is the largest single inference-time degradation we see and confirms that CFG at extreme noise levels actively hurts text alignment [16]. Halving the ODE step count from 100 to 50 actually yields a marginal  $+0.003$  improvement, and 200 steps is statistically tied with 100, so 50 steps is a strict efficiency win. Finally, the post-hoc EMA window matters: the last 10 or 30 epochs are interchangeable, but widening to all 96 snapshots from the full run drops CLAP by  $-0.051$ , because early under-converged snapshots pull the averaged model away from its operating point.

TABLE VI  
INFERENCE ABLATIONS (EFFICIENCY 499M, CFG 7.0, `MSCLAP`,  $n = 100$ ).  $\Delta$  IS CLAP RELATIVE TO THE BASELINE.

Configuration	CLAP $\uparrow$	$\Delta$
Baseline (steps=100, GI=[0.1,0.9])	0.4958	–
<i>Guidance Interval</i>		
CFG over full $[0,1]$ (GI off)	0.4652	–0.031
<i>ODE Sampling Steps</i>		
50 steps	0.4986	+0.003
200 steps	0.4951	–0.001
<i>Post-hoc EMA Window</i>		
Last 10 epochs (16 snapshots)	0.4961	+0.000
Last 60 epochs (all 96 snap.)	0.4455	–0.051

### F. CFG Sweep

To probe the relationship between CFG and CLAP at finer resolution, we sweep CFG in unit increments from 3 to 15 on the Efficiency 499M model and re-evaluate each with `msclap`, as plotted in Fig. 2. The curve shows a clear under-guidance regime up to CFG of about 5, where CLAP rises from 0.482 to 0.494, followed by a plateau above CFG 6 within  $\pm 0.003$ .

### G. Perceptual Evaluation and Architectural Conditioning Ablation

We evaluate our two submitted models alongside four instrumental-only baselines, namely Stable Audio Open [4] and three sizes of MusicGen [1], on the same 100 final-test prompts. The evaluation uses three automatic measures: `msclap` CLAP, AudioBox aesthetics [11], and an LLM-as-judge protocol; we also report human MOS in Sec. VII-H. The AudioBox axes report content enjoyment, content usefulness, production complexity, and production quality, abbreviated CE, CU, PC, and PQ. The LLM-as-judge protocol prompts two multimodal audio models, GPT-4o-audio

TABLE VII

COMBINED EVALUATION ON 100 OFFICIAL FINAL-TEST PROMPTS (MSCLAP CLAP, AUDIOBOX, LLM-AS-JUDGE, SINGLE-RATER HUMAN MOS; AXES AND PROTOCOL DESCRIBED IN SEC. VII-G–VII-H). **BOLD** MARKS THE BEST PER COLUMN OVERALL; **BLUE BOLD** MARKS THE BEST WITHIN THE EFFICIENCY BLOCK; **RED BOLD** MARKS THE BEST WITHIN THE PERFORMANCE BLOCK.

Setup	Params	Data	msclap $\uparrow$	AudioBox $\uparrow$				GPT-4o-audio $\uparrow$			Gemini 3 Pro $\uparrow$			Human $\uparrow$		
				CE	CU	PC	PQ	CE	PQ	PA	CE	PQ	PA	Q	F	P
<i>Instrumental baselines</i>																
Stable Audio Open [4]	1.2B	$\sim$ 7.3kh	0.507	6.45	7.54	4.33	7.63	7.34	8.31	8.89	4.24	4.96	4.88	3.60	<b>3.85</b>	3.35
MusicGen-small [1]	300M	$\sim$ 20kh	<b>0.531</b>	6.86	7.47	5.22	7.40	7.40	8.36	8.93	5.58	5.44	7.05	3.60	3.55	3.55
MusicGen-medium [1]	1.5B	$\sim$ 20kh	0.528	6.85	7.53	5.25	7.44	<b>7.42</b>	<b>8.42</b>	8.92	5.85	<b>5.72</b>	7.20	<b>3.85</b>	3.75	<b>3.80</b>
MusicGen-large [1]	3.3B	$\sim$ 20kh	<b>0.531</b>	<b>7.12</b>	<b>7.69</b>	<b>5.42</b>	<b>7.58</b>	7.38	8.40	<b>8.95</b>	<b>5.95</b>	<b>5.72</b>	<b>7.43</b>	<b>3.85</b>	3.75	3.55
<i>Ours: Efficiency and conditioning ablations</i>																
Efficiency Original, 16-layer DiT	499M	457h	0.4993	<b>6.92</b>	7.40	<b>5.17</b>	<b>7.33</b>	<b>7.41</b>	<b>8.40</b>	8.89	<b>3.08</b>	2.90	<b>4.00</b>	2.95	3.10	2.75
without lyric encoder (inference)	499M	457h	<b>0.5005</b>	6.66	7.39	4.81	7.26	7.35	8.38	8.88	3.07	<b>3.12</b>	3.80	<b>3.15</b>	<b>3.30</b>	<b>2.90</b>
without timbre encoder (inference)	499M	457h	0.4996	6.81	<b>7.42</b>	4.96	7.33	7.39	8.37	<b>8.95</b>	3.07	3.01	3.86	2.95	2.90	2.75
retrained without lyric & timbre encoders, 16-layer DiT	371M	457h	0.4718	6.45	7.03	5.15	7.02	7.38	8.33	8.91	1.94	1.97	2.51	2.45	2.50	2.15
retrained without lyric & timbre encoders, 23-layer DiT	489M	457h	0.4844	6.50	7.11	5.02	7.05	7.35	8.30	8.92	2.60	2.60	3.28	2.75	2.90	2.65
<i>Ours: Performance and conditioning ablations</i>																
Performance Original, 24-layer DiT	2.4B	457h	0.4737	<b>6.65</b>	<b>7.37</b>	4.89	<b>7.29</b>	7.37	8.35	8.87	3.34	3.43	4.25	2.95	3.20	2.85
without lyric encoder (inference)	2.4B	457h	0.4739	6.38	7.24	4.60	7.15	7.35	8.35	8.87	3.44	3.63	<b>4.36</b>	3.10	<b>3.40</b>	3.10
without timbre encoder (inference)	2.4B	457h	<b>0.4826</b>	6.53	7.31	4.70	7.25	7.37	8.37	8.90	<b>3.55</b>	<b>3.72</b>	4.32	<b>3.15</b>	3.25	<b>3.25</b>
retrained without lyric & timbre encoders, 24-layer DiT	1.8B	457h	0.4776	6.47	7.06	<b>5.20</b>	7.01	<b>7.40</b>	<b>8.39</b>	<b>8.95</b>	2.03	2.00	2.81	2.25	2.55	2.25

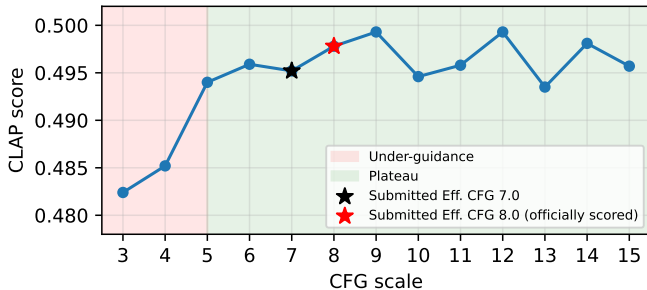


Fig. 2. Sweep of CLAP against CFG on the Efficiency 499M model, evaluated with msclap on  $n = 100$  final-test prompts. CLAP rises monotonically up to CFG of about 5 and then plateaus within  $\pm 0.003$ . Stars indicate the two Efficiency-track variants we submitted, CFG 7.0 and 8.0; the organizers selected CFG 8.0 for the official scoring. Both sit on the CLAP plateau.

at version `gpt-4o-audio-preview-2025-06-03` and Gemini 3 Pro at version `gemini-3-pro-preview`, to rate each 10-second clip on a 1 to 10 scale for content enjoyment, production quality, and prompt adherence, abbreviated CE, PQ, and PA.

We additionally probe the lyric and timbre encoders, which receive constant inputs (lyric “[Instrumental]”, timbre zero tensor) yet still emit non-zero features into cross-attention. To distinguish a meaningful learned representation from a structural anchor the DiT co-adapts to, we run three controlled tests: (i) inference-time zeroing of one encoder, (ii) a from-scratch retrain with both encoders removed ( $\sim 26\%$  fewer parameters), and (iii) a capacity-matched retrain reallocating that budget into a 23-layer DiT (489M, still under the Efficiency cap).

The LLM-as-judge results in Table VII highlight that the choice of judge model matters as much as the choice of metric. GPT-4o-audio produces a much narrower score range across models, suggesting limited discriminative power under this prompt. Gemini 3 Pro is far more discriminative: MusicGen

leads on every axis, followed by SAO, with our submitted models and the inference-time conditioning ablations clustering below. Within each of our model groups, Gemini cleanly singles out the text-only architectures as the clearest losers. The AudioBox metrics also reflect this signal, but more weakly.

**Discussion.** Taken together with the mild inference-time zeroing effect, this pattern suggests the encoder contribution is realized primarily during training: the lyric and timbre branches play a structural role—either by reshaping the cross-attention pattern or by providing stable conditioning anchors—that more DiT depth alone cannot replicate, since the capacity-matched retrain recovers only marginally despite matching val MSE.

#### H. Human Evaluation

To complement the automatic metrics and LLM judges, a single author rates 20 of the 100 final-test prompts across all 13 setups in Table VII on a 1 to 5 scale for quality (Q), prompt fidelity (F), and overall preference (P). The Q/F/P columns of Table VII report the per-setup means.

Three patterns emerge. The MusicGen family leads at preference 3.55–3.80, followed by Stable Audio Open at 3.35 and our models at 2.7–2.9, mirroring the training-data-scale gap. The text-only retrains are the worst among our models at 2.15–2.65, agreeing with the AudioBox CE drop and with Gemini 3 Pro singling out the same architectures, and supporting the architectural-ablation finding that the lyric and timbre branches contribute as training-time anchors. Inference-time zeroing of either branch does not hurt human ratings and slightly improves them in several rows, consistent with the training-time effect interpretation.

**Discussion.** Across the conditioning ablation, the gap is visible on AudioBox aesthetics, Gemini 3 Pro, and human MOS but essentially invisible on validation MSE and msclap, and GPT-4o-audio would have masked it entirely. We therefore

recommend triangulating across both metric families and judge models when comparing architecturally similar variants.

### I. Failure Case Analysis

Three systematic failure modes show up under qualitative inspection. The first is *instrument fidelity*, where distinctive instruments such as saxophone, harmonica, or vibraphone often render as generic timbres, reflecting their limited representation in the 457-hour set. The second is *tempo precision*, where the model captures coarse energy levels but does not reliably hit specific BPM targets, since tempo is conveyed only implicitly through captions. The third is *complex arrangements*, where prompts asking for four or more simultaneous instruments produce texturally muddy output and individual instruments are not separable. These primarily reflect the constrained data scale rather than architectural deficiencies.

## VIII. LIMITATIONS

Our model is trained from scratch on only 457 hours of audio under strict challenge constraints, our analyses use the official 100-prompt test set, and the human MOS used inside Table VII for the conditioning ablation is single-rater on a 20-prompt subset; the organizer-released multi-rater MOS reported in Sec. VII-H covers only the two submitted models and the organizer baseline, not the conditioning ablations. The structural role we attribute to the lyric and timbre sub-modules is observational; the precise mechanism is left to future work.

## IX. CONCLUSION

We studied a Diffusion Transformer text-to-music model with lyric and timbre conditioning, adapted to an instrumental-only task in which both auxiliary branches receive only degenerate inputs. Across AudioBox aesthetics, LLM-as-judge, and human MOS, retraining the model without these branches consistently degraded perceptual quality, while a capacity-matched deeper DiT recovered only marginally despite matching validation MSE. This suggests the auxiliary branches may act as training-time conditioning anchors that more DiT depth alone cannot easily replicate. We validated the same model through comparisons with external instrumental baselines and through our submission to the ICME 2026 ATTM Grand Challenge, where our Performance submission ranked **first under both the objective metrics and the subsequent organizer-administered MOS**, attaining the highest overall MOS across all challenge submissions, while our Efficiency submission was a finalist that tied for second under the objective metrics. These observations motivate further study of the architectural role of auxiliary conditioning branches in text-to-audio generation, including whether the same training-time anchor effect generalizes to other multi-modal architectures.

## REFERENCES

[1] Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez, “Simple and controllable music generation,” in *Proc. NeurIPS*, 2024.

[2] Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Tagliasacchi, Neil Zeghidour, and Christian Frank, “Musiclm: Generating music from text,” in *Proc. ICML*, 2023.

[3] Haohe Liu, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Qiao Tian, Yuping Wang, Mark D Plumbley, and Wenwu Wang, “Audioldm 2: Learning holistic audio generation with self-supervised pretraining,” *IEEE/ACM Trans. Audio, Speech, and Language Processing*, 2024.

[4] Zach Evans, Julian Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons, “Stable audio open,” *arXiv preprint arXiv:2407.14358*, 2024.

[5] Max WY Lam, Qiao Tian, Li Tang, Zongyu Wang, Ruibin Liu, Jiawei Yin, Jingyi Huang, Yi Wu, et al., “Efficient neural music generation,” *Proc. NeurIPS*, 2024.

[6] William Peebles and Saining Xie, “Scalable diffusion models with transformers,” in *Proc. ICCV*, 2023, pp. 4195–4205.

[7] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le, “Flow matching for generative modeling,” in *Proc. ICLR*, 2023.

[8] SeungHeon Doh, Keunwoo Choi, Jongpil Lee, and Juhan Nam, “Lp-musiccaps: Llm-based pseudo music captioning,” in *Proc. ISMIR*, 2023.

[9] Ilaria Manco et al., “Jamendomaxcaps: A large scale music-caption dataset with imputed metadata,” *arXiv preprint arXiv:2502.07461*, 2025.

[10] ACE-Step Team, “Ace-step: A step towards music generation foundation model,” *arXiv preprint arXiv:2506.00045*, 2025.

[11] Andros Tjandra, Yi-Chiao Wu, Baishan Guo, John Hoffman, Brian Ellis, Apoorv Vyas, Bowen Shi, Sanyuan Chen, Matthew Le, Nick Zacharov, Zach Wood-Doughty, Bapi Genzel, Kaustubh Schwenk, Joseph Yungster, Wei-Ning Hsu, Mary Williamson, et al., “Meta Audiobox aesthetics: Unified automatic quality assessment for speech, music, and sound,” *arXiv preprint arXiv:2502.05139*, 2025.

[12] Fang-Chih Hsieh, Wei-Jaw Lee, Chun-Ping Wang, Hung-yi Lee, Hao-Wen Dong, and Yi-Hsuan Yang, “Academic text-to-music grand challenge: Datasets, baselines, and evaluation methods,” in *International Conference on Multimedia and Expo, Grand Challenge Paper*, 2026.

[13] Dmitry Bogdanov, Minz Won, Philip Tovstogan, Alastair Porter, and Xavier Serra, “The mtg-jamendo dataset for automatic music tagging,” in *Proc. ICML Workshop on Machine Learning for Music Discovery*, 2019.

[14] Tiankai Hang, Shuyang Gu, Chen Li, Jianmin Bao, Dong Chen, Han Hu, Xin Geng, and Baining Guo, “Efficient diffusion training via min-snr weighting strategy,” in *Proc. ICCV*, 2023.

[15] Tero Karras, Miika Aittala, Jaakko Lehtinen, Janne Hellsten, Timo Aila, and Samuli Laine, “Analyzing and improving the training dynamics of diffusion models,” *arXiv preprint arXiv:2312.02696*, 2024.

[16] Tuomas Kynkäänniemi, Miika Aittala, Timo Aila, and Jaakko Lehtinen, “Applying guidance in a limited interval improves sample and distribution quality in diffusion models,” in *Proc. NeurIPS*, 2024.

[17] Junyoung Koh, Soo Yong Kim, Yongwon Choi, and Gyu Hyeong Choi, “Jamendo-qa: A large-scale music question answering dataset,” 2025.

[18] Junyoung Koh, Jaeyun Lee, Soo Yong Kim, Gyu Hyeong Choi, Jung In Koh, Jordan Phillips, Yeonjin Lee, and Min Song, “Jamendo-mt-qa: A benchmark for multi-track comparative music question answering,” 2026.

[19] Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou, “Qwen3 embedding: Advancing text embedding and reranking through foundation models,” *arXiv preprint arXiv:2506.05176*, 2025.

[20] Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai, “GQA: Training generalized multi-query transformer models from multi-head checkpoints,” in *Proc. EMNLP*, 2023.

[21] Jonathan Ho and Tim Salimans, “Classifier-free diffusion guidance,” in *NeurIPS Workshop on Deep Generative Models and Downstream Applications*, 2021.