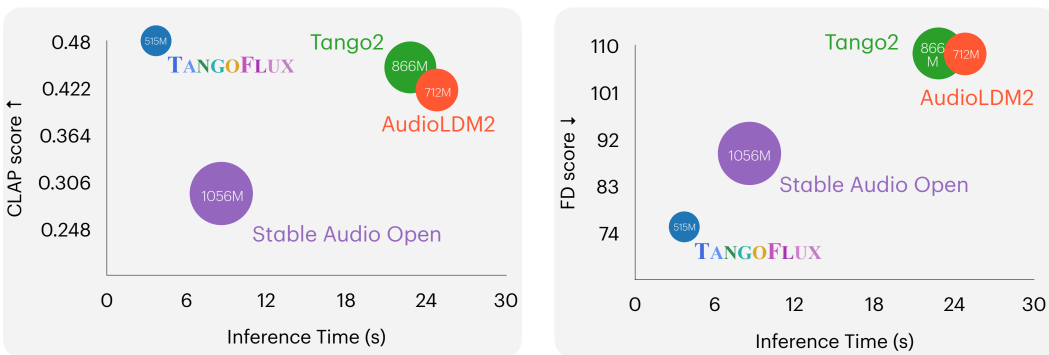
# TANGOFLUX: SUPER FAST AND FAITHFUL TEXT TO AUDIO GENERATION WITH FLOW MATCHING AND CLAP-RANKED PREFERENCE OPTIMIZATION

**Chia-Yu Hung**[1] **Navonil Majumder**[1] **Zhifeng Kong**[2] **Ambuj Mehrish**[1]
**Rafael Valle**[2] **Bryan Catanzaro**[2] **Soujanya Poria**[1]

[1]Singapore University of Technology and Design (SUTD)
[2]NVIDIA

{chiayu_hung, navonil_majumder, ambuj_mehrish, sporia}@sutd.edu.sg
{zkong, rafaelvalle, bcatanzaro}@nvidia.com




🚀 Comparing our model, TANGOFLUX, with other state-of-the-art text-to-audio generation models:🔥 TangoFlux achieves better quality audio (measured by CLAP and FD scores) while being approximately 2x faster ⏩ than the next fastest model, all with fewer trainable parameters!

## TANGOFLUX *Resources*

*Website* → **https://tangoflux.github.io**

*Code Repository* → **https://github.com/declare-lab/TangoFlux**

*Pretrained Model* → **https://huggingface.co/declare-lab/TangoFlux**

*Dataset Fork* → **https://huggingface.co/datasets/declare-lab/CRPO**

*Interactive Demo* → **https://huggingface.co/spaces/declare-lab/TangoFlux**

## ABSTRACT

We introduce TANGOFLUX, an efficient Text-to-Audio (TTA) generative model with 515M parameters, capable of generating up to 30 seconds of 44.1kHz audio in just 3.7 seconds on a single A40 GPU. A key challenge in aligning TTA models lies in the difficulty of creating preference pairs, as TTA lacks structured mechanisms like verifiable rewards or gold-standard answers available for Large Language Models (LLMs). To address this, we propose CLAP-Ranked Preference Optimization (CRPO), a novel framework that iteratively generates and optimizes preference data to enhance TTA alignment. We demonstrate that the audio preference dataset generated using CRPO outperforms existing alternatives. With this framework, TangoFlux achieves state-of-the-art performance across both objective and subjective benchmarks. We open source all code and models to support further research in TTA generation.

# 1 Introduction

Audio plays a vital role in daily life and creative industries, from enhancing communication and storytelling to enriching experiences in music, sound effects, and podcasts. However, creating high-quality audio, such as foley effects or music compositions, demands significant effort, expertise, and time. Recent advancements in text-to-audio (TTA) generation (Majumder et al., 2024; Ghosal et al., 2023; Liu et al., 2023; 2024b; Xue et al., 2024; Vyas et al., 2023; Huang et al., 2023b;a) and offer a transformative approach, enabling the automatic creation of diverse and expressive audio content directly from textual descriptions. This technology holds immense potential to streamline audio production workflows and unlock new possibilities in multimedia content creation. However, many existing models face challenges with controllability, occasionally struggling to fully capture the details in the input prompts, especially when the prompts are complex. This can sometimes result in generated audio that omits certain events or diverges from the user intent. At times, the generated audio may even contain input-adjacent, but unmentioned and unintended, events, that could be characterized as hallucinations.

In contrast, the recent advancements in Large Language Models (LLMs) (Ouyang et al., 2022) have been significantly driven by the alignment stage after pre-training and supervised fine-tuning. This alignment stage, often leveraging reinforcement learning from human feedback (RLHF) or other reward-based optimization methods, endows the generated outputs with human preferences, ethical considerations, and task-specific requirements (Ouyang et al., 2022). Despite the rapid progress in TTA models, until recently (Majumder et al., 2024) alignment, that could mitigate the aforementioned issues with audio outputs, has not been a mainstay in TTA model training.

One critical challenge in implementing alignment for TTA models lies in the creation of preference pairs. Unlike LLM alignment, where off-the-shelf reward models (Lambert et al., 2024a;b) and human feedback data or verifiable gold answers are available, TTA domain as yet lacks such tooling. For instance, in general, LLM alignment settings, such as safety or instruction following, tools exist for categorizing specific safety risks (Inan et al., 2023). Frontier LLMs like GPT-4 (OpenAI et al., 2024) are often used directly to judge the candidate outputs (Zheng et al., 2023).

While audio language models (Chu et al., 2024; 2023; Tang et al., 2024) can process audio inputs and generate textual outputs, they often produce noisy feedback, unfit for preference pair creation for audio. BATON (Liao et al., 2024) employs human annotators to assign a binary score of 0 or 1 to each audio sample based on its alignment with a given prompt. However, such labor-intensive manual approach is often economically not viable at a large scale.

To address these issues, we propose CLAP-Ranked Preference Optimization (CRPO), a simple yet effective approach to generate audio preference data and perform preference optimization on rectified flows. As shown in Fig. 1, CRPO consists of iterative cycles of data sampling, generating preference pairs, and performing preference optimization, resembling a self-improvement algorithm. We first demonstrate that the CLAP model (Wu* et al., 2023) can serve as a proxy reward model for ranking generated audios by alignment with the text description. Using this ranking, we construct an audio preference dataset that yields superior performance after preference optimization, as compared to other audio preference datasets, such as, BATON and Audio-Alpaca (Majumder et al., 2024). Finally, we demonstrate the effectiveness of this iterative optimization, emphasizing the importance of each component, including the modified loss function compared to conventional preference optimization loss.

Additionally, many TTA models are trained on proprietary data (Evans et al., 2024b;a; Copet et al., 2024), with their weights often unavailable to the public or accessible only through private APIs, posing challenges for public use and foundational research. Moreover, the diffusion-based TTA models (Ghosal et al., 2023; Majumder et al., 2024; Liu et al., 2024b) are known to require too many denoising steps to generate a decent output, consuming much GPU compute and time.

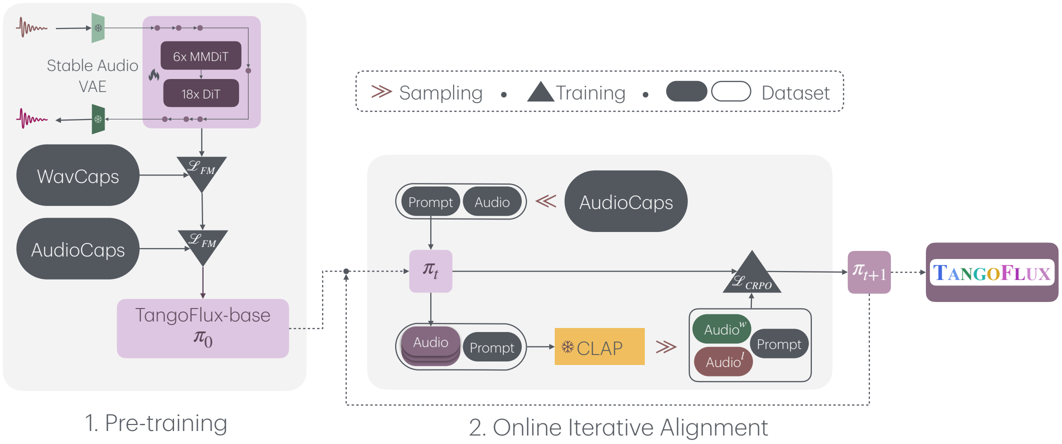



Figure 1: A depiction of the overall training pipeline of TANGOFLUX.

To address this, we introduce TANGOFLUX, trained on a completely non-proprietary dataset, achieving *state-of-the-art* performance on benchmarks and out-of-distribution prompts, despite its smaller size. Unlike conventional TTA models, TANGOFLUX supports variable-duration audio generation up to 30 seconds with a blazing inference speed of 3.7 seconds on a single A40 GPU. This is achieved by the use of transformer (Vaswani et al., 2023) backbone that undergoes pretraining, fine-tuning, and preference optimization with rectified flow matching as training objective—enabling good quality audio output guided by much fewer sampling steps tracing the almost straight path between the noise and output audio.

**Our contributions**:

(i) We introduce TANGOFLUX, a small and fast TTA model based on rectified flow that achieves *state-of-the-art* performance for fully non-proprietary training data.
(ii) We propose CRPO, a simple and effective strategy to generate audio preference data and align rectified flow, demonstrating its superior performance over other audio preference datasets.
(iii) We conduct extensive experiments and highlight the importance of each component of CRPO in aligning rectified flows for improving scores on benchmarks.
(iv) We publicly release the code and model weights to foster research on text-to-audio generation.

## 2 METHOD

TANGOFLUX consists of FluxTransformer blocks which are Diffusion Transformer (DiT) (Peebles & Xie, 2023) and Multimodal Diffusion Transformer (MMDiT) (Esser et al., 2024), conditioned on textual prompt and duration embedding to generate audio at 44.1kHz up to 30 seconds. TANGOFLUX learns a rectified flow trajectory from audio latent representation encoded by a variational autoencoder (VAE) (Kingma & Welling, 2022). TANGOFLUX training pipeline consists of three stages: pre-training, fine-tuning then preference optimization. TANGOFLUX is aligned via CRPO which iteratively generates new synthetic data and constructs preference pairs to perform preference optimization. The overall pipeline is depicted in Fig. 1.

### 2.1 AUDIO ENCODING

We use the VAE from Stable Audio Open (Evans et al., 2024c), which is capable of encoding stereo audio waveforms at 44.1kHz into audio latent representations. Given a stereo audio $X \in \mathbb{R}^{2\times d\times sr}$ with $d$ as the duration and $sr$ as the sampling rate, the VAE encodes $X$ into a latent representation $Z \in \mathbb{R}^{L\times C}$, with $L$, $C$ being the latent sequence length and channel size, respectively. The VAE decodes the latent representation $Z$ back into the original stereo audio $X$. The entire VAE is kept frozen during TANGOFLUX training.

### 2.2 MODEL CONDITIONING

To enable the controllable generation of audio of varying lengths, we employ textual conditioning and duration conditioning. Textual conditioning controls the event present of the generated audio based on a provided description, while duration conditioning specifies the desired audio length, up to a maximum of 30 seconds.

**Textual Conditioning.** Given the textual description of an audio, we obtain the text encoding $c_{text}$ from a pretrained text-encoder. Given the strong performance of FLAN-T5 (Chung et al., 2022; Raffel et al., 2023) as conditioning in text-to-audio generation (Majumder et al., 2024; Ghosal et al., 2023), we select FLAN-T5 as our text encoder.

**Duration Encoding.** Inspired by the recent works (Evans et al., 2024c;a;b), to generate audios with variable length, we firstly use a small neural network to encode the audio duration into a duration embedding $c_{dur}$. This is concatenated with the text encoding $c_{text}$ and fed into TANGOFLUX to control the duration of audio output.

### 2.3 MODEL ARCHITECTURE

Following the recent success of FLUX models in image generation [1], we adopt a hybrid MMDiT and DiT architecture as the backbone for TANGOFLUX. While MMDiT blocks demonstrated a strong performance, simplifying some of them into single DiT block improved scalability and parameter efficiency [2]. These lead us to select a model architecture consisting of 6 blocks of MMDiT, followed by 18 blocks of DiT. Each block uses 8 attention heads, with each attention head dimension of 128, resulting in a width of 1024. This configuration results in a model with 515M parameters.

### 2.4 FLOW MATCHING

Several generative models have been successfully trained under the diffusion framework (Ho et al., 2020; Song et al., 2022; Liu et al., 2022). However, this approach is known to be sensitive to the choice of noise scheduler, which may significantly affect performance. In contrast, the flow matching (FM) framework (Lipman et al., 2023; Albergo & Vanden-Eijnden, 2023) has been shown to be more robust to the choice of noise scheduler, making it a preferred choice in many applications, including text-to-audio (TTA) and text-to-speech (TTS) tasks (Liu et al., 2024a; Le et al., 2023; Vyas et al., 2023).

Flow matching builds upon the continuous normalizing flows framework (Onken et al., 2021). It generates samples from a target distribution by learning a time-dependent vector field that maps samples from a simple prior distribution (e.g., Gaussian) to a complex target distribution. Prior work in TTA, such as AudioBox (Vyas et al., 2023) and Voicebox (Le et al., 2023), has predominantly adopted the Optimal Transport conditional path proposed by (Lipman et al., 2023). However, in our approach, we utilize rectified flows (Liu et al., 2022) instead, which is a straight line path from noise to distribution, corresponding to the shortest path.

**Rectified Flows.** Given a latent representation of an audio sample $x_1$, a noise sample $x_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, time-step $t \in [0, 1]$, we can construct a training sample $x_t$ where the model learns to predict a velocity $v_t = \frac{dx_t}{dt}$ that guides $x_t$ to $x_1$. While there exist several methods of constructing transport path $x_t$ , we used rectified flows (RFs) (Liu et al., 2022), in which the forward process are straight paths between target distribution and noise distribution, defined in Eq. (1). It was empirically demonstrated that rectified flows are sample efficient and degrade less compared to other formulations when reducing lesser number of sampling steps (Esser et al., 2024). We use $\theta$ to denote the model $u$'s parameter. The model directly regresses the predicted velocity $u(\mathbf{x}_t, t; \theta)$ against the ground truth velocity $\mathbf{v}_t$ where the loss is shown in Eq. (2).

$$x_t = (1-t)x_1 + t\tilde{x}_0, v_t = \frac{dx_t}{dt} = \tilde{x}_0 - x_1, \tag{1}$$

[1] https://blackforestlabs.ai/

[2] https://blog.fal.ai/auraflow/

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{x_1,x_0,t} \left\| u(x_t, t; \theta) - v_t \right\|^2 . \tag{2}$$

**Inference.** During inference, we sample a noise from prior distribution $\tilde{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and use an ordinary differential equation solver to compute $x_1$, based on the model-predicted velocity $v_t$ at each time step $t$. We use the Euler solver for this process.

### 2.5 CLAP-Ranked Preference Optimization (CRPO)

CLAP-Ranked Preference Optimization (CRPO) leverages a text-audio joint-embedding model like CLAP (Wu* et al., 2023) as a proxy reward model to rank the generated audios by similarity with the input description and subsequently construct the preference pairs.

We firstly set a pre-trained checkpoint of TangoFlux architecture as the base model to align, denoted by $\pi_0$. Thereafter, CRPO iteratively aligns checkpoint $\pi_k := u(\cdot; \theta_k)$ into checkpoint $\pi_{k+1}$, starting from $k = 0$. Each of such alignment iterations consists of three steps: (i) batched online data generation, (ii) reward estimation and preference dataset creation, and (iii) fine-tuning $\pi_k$ into $\pi_{k+1}$ via direct preference optimization.

This approach to aligning rectified flow is inspired by a few LLM alignment approaches (Zelikman et al., 2022; Kim et al., 2024a; Yuan et al., 2024; Pang et al., 2024). However, there are key distinctions to our work: (i) we focus on aligning rectified flows for audio generation, rather than autoregressive language models; (ii) while LLM alignment benefits from numerous off-the-shelf reward models (Lambert et al., 2024b), which facilitate the construction of preference datasets based on reward scores, LLM judged outputs, or programmatically verifiable answers, the audio domain lacks such models or method for evaluating audio. We demonstrate that the CLAP model can serve as an effective proxy audio reward model, enabling the creation of preference datasets (see Section 4.3). Finally, we highlight the necessity of generating online data at every iteration, as iterative optimization on offline data leads to quicker performance saturation and subsequent degradation.

#### 2.5.1 CLAP as a Reward Model

CLAP reward score is calculated as the cosine similarity between textual and audio embeddings encoded by the model. Thus, we assume that CLAP can serve as a reasonable proxy reward model for evaluating audio outputs against the textual description. In Section 4.3, we demonstrate that using CLAP as a judge to choose the best-of-N inferred policies improves performance in terms of objective metrics.

#### 2.5.2 Batched Online Data Generation

To construct a preference dataset at iteration $k$, we first sample a set of prompts $M_k$ from a larger pool $B$. Subsequently, we generate $N$ audios for each prompt $y_i \in M_k$ using $\pi_k$ and use CLAP[3] (Wu* et al., 2023) to rank those audios by similarity with $y_i$. For each prompt $y_i$, we select the highest-rewarded or -ranking audio $x_i^w$ as the winner and the lowest-rewarded audio $x_i^l$ as the loser, yielding a preference dataset $\mathcal{D}_k = \{(x_i^w, x_i^l, y_i) \mid y_i \in M_k\}$.

#### 2.5.3 Preference Optimization

Direct preference optimization (DPO) (Rafailov et al., 2024c) is shown to be effective at instilling human preferences in LLMs (Ouyang et al., 2022). Consequently, DPO is successfully translated into DPO-Diffusion (Wallace et al., 2023) for alignment of diffusion models. The DPO-diffusion loss is defined as

$$L_{\text{DPO-Diff}} = -\mathbb{E}_{(x_0^w, x_0^l) \sim \mathcal{D}} \log \sigma \left( \beta \mathbb{E}_{x_{1:T}^w \sim p_\theta(x_{1:T}^w | x_0^w), x_{1:T}^l \sim p_\theta(x_{1:T}^l | x_0^l)} \left[ \log \frac{p_\theta(x_{0:T}^w)}{p_{\text{ref}}(x_{0:T}^w)} - \log \frac{p_\theta(x_{0:T}^l)}{p_{\text{ref}}(x_{0:T}^l)} \right] \right) \tag{3}$$

[3] https://huggingface.co/lukewys/laion_clap/blob/main/630k-audioset-best.pt

$$= -\mathbb{E}_{n,\epsilon^w,\epsilon^l} \log \sigma\big(-\beta(\|\epsilon_n^w - \epsilon_\theta(x_n^w)\|_2^2 - \|\epsilon_n^w - \epsilon_{\text{ref}}(x_n^w)\|_2^2 - (\|\epsilon_n^l - \epsilon_\theta(x_n^l)\|_2^2 - \|\epsilon_n^l - \epsilon_{\text{ref}}(x_n^l)\|_2^2))\big) \tag{4}$$

After some algebraic simplification of Eq. (3), as shown by Wallace et al. (2023), $L_{\text{DPO-Diff}}$ reduces to a tractable term shown in Eq. (4). Here, $T$ denotes the diffusion timestep $n \sim U(0, T)$, $x_n^l$ and $x_n^w$ the losing and winning audio respectively, and $\epsilon \sim \mathcal{N}(0, \mathbf{I})$.

Following Esser et al. (2024), DPO-Diffusion loss is applicable to rectified flow through the equivalence (Lipman et al., 2023) between $\epsilon_\theta$ and $u(\cdot;\theta)$, thereby the noise matching loss terms can be substituted with flow matching terms:

$$L_{\text{DPO-FM}} = -\mathbb{E}_{t\sim\mathcal{U}(0,1),x^w,x^l} \log \sigma\Bigg( -\beta\Bigg( \underbrace{\|u(x_t^w, t; \theta) - v_t^w\|_2^2}_{\text{Winning loss}} - \underbrace{\|u(x_t^l, t; \theta) - v_t^l\|_2^2}_{\text{Losing loss}}$$
$$- \Big( \underbrace{\|u(x_t^w, t; \theta_{\text{ref}}) - v_t^w\|_2^2}_{\text{Winning reference loss}} - \underbrace{\|u(x_t^l, t; \theta_{\text{ref}}) - v_t^l\|_2^2}_{\text{Losing reference loss}} \Big)\Bigg)\Bigg), \tag{5}$$

where $t$ is the flow matching timestep and $x_t^l$ and $x_t^w$ represent losing and winning audio, respectively.

The DPO loss for LLMs models the relative likelihood of the winner and loser responses, allowing minimization of the loss by increasing their margin, even if both log-likelihoods decrease (Pal et al., 2024). As DPO optimizes the relative likelihood of the winning responses over the losing ones, not their absolute values, convergence actually requires both likelihoods to decrease despite being counterintuitive (Rafailov et al., 2024b). The decrease in likelihood does not necessarily decrease performance, but required for improvement (Rafailov et al., 2024a). However, in the context of rectified flows, this behavior is less clear due to the challenges in estimating the likelihood of generating samples with classifier-free guidance (CFG). A closer look at $\mathcal{L}_{\text{DPO-FM}}$ (Eq. (5)) reveals that it can similarly be minimized by increasing the margin between the winning and losing losses, even if both losses increase. In Section 4.5, we demonstrate that preference optimization of rectified flows via $\mathcal{L}_{\text{DPO-FM}}$ suffer from this phenomenon as well.

To remedy this, we incorporate the winning loss directly into the optimization objective to prevent *winning loss* from increasing. Our loss is denoted as

$$\mathcal{L}_{\text{CRPO}} := \mathcal{L}_{\text{DPO-FM}} + \mathcal{L}_{\text{FM}},$$

where $\mathcal{L}_{\text{FM}}$ is the flow matching loss computed on the winning audio as shown in Eq. (2). While the DPO loss is effective at improving preference rankings between chosen and rejected audio, relying on it alone can lead to overoptimization. This can distort the semantic and structural fidelity of the winning audio, causing the model's outputs to drift from the desired distribution. Adding the $\mathcal{L}_{\text{FM}}$ component mitigates this risk by anchoring the model to the high-quality attributes of the chosen data. This regularization stabilizes training and preserves the essential properties of the winning examples, ensuring a balanced and robust optimization process. Our empirical results demonstrates $\mathcal{L}_{\text{CRPO}}$ outperform $\mathcal{L}_{\text{DPO-FM}}$ as shown in Section 4.5.

## 3 Experiments

### 3.1 Model Training

We pretrained TangoFlux on Wavcaps (Mei et al., 2024) for 80 epochs with the AdamW (Loshchilov & Hutter, 2019), $\beta_1 = 0.9, \beta_2 = 0.95$, a max learning rate of $5 \times 10^{-4}$. We used a linear learning rate scheduler for 2000 steps. We used five A40 GPUs with a batch size of 16 on each device, resulting in an overall batch size of 80. After pretraining, TangoFlux was finetuned on the *AudioCaps* training set for 65 additional epochs. Several works find that sampling timesteps $t$ from the middle of its range $[0, 1]$ leads to superior results (Hang et al., 2024; Kim et al., 2024b; Karras et al., 2022), thus, we sampled $t$ from a logit-normal distribution with a mean of 0 and variance of 1, following the approach in (Esser et al., 2024). We name this version as TANGOFLUX-base.

During the alignment phase, we used the same optimizer, but an overall batch size of 48, a maximum learning rate of $10^{-5}$, and a linear warmup of 100 steps. For each iteration of CRPO, we train for 8 epochs and select the last epoch checkpoint to perform batched online data generation. We performed 5 iterations of CRPO due to the manifestation of performance saturation.

### 3.2 DATASETS

**Training dataset**. We use complete open source data which consists of approximately 400k audios from *Wavcaps* (Mei et al., 2024) and 45k audios from the training set of *AudioCaps*. (Kim et al., 2019). For audios shorter than 30 seconds, we pad the remaining audio with silence. For audios longer than 30 seconds, we perform center cropping of 30 seconds. Since the audio files are all mono, we duplicated the channel to create "stereo" audio for compatibility with our model.

**CRPO dataset**. We initialize the prompt bank as the prompts of *AudioCaps* training set, with a total of 45k prompts. At the start of each iteration of CRPO, we randomly sample 20k prompts from the prompt bank and generate 5 audios per prompt, and use the CLAP model to construct 20k preference pairs.

**Evaluation dataset**. For the main results, we evaluated TANGOFLUX on the *AudioCaps* test set, using the same 886-sample split as (Majumder et al., 2024). Objective metrics are reported on this subset. Additionally, we categorized *AudioCaps* prompts using GPT-4 to identify those with multiple distinct events, such as "Birds chirping and thunder strikes," which includes "sound of birds chirping" and "sound of thunder." Metrics for these multi-event captions are reported separately. Subjective evaluation was conducted on an out-of-distribution dataset with 50 challenging prompts.

### 3.3 OBJECTIVE EVALUATION

**Baselines**. We compare TANGOFLUX to four existing strong baselines for text-to-audio generation: `Tango 2`, `AudioLDM 2`, and `Stable Audio Open`, including the previous SOTA models. For all of our baseline evaluations, we use the default recommended classifier free guidance (CFG) scale (Ho & Salimans, 2022) and number of steps. For TANGOFLUX, we use a CFG scale of 4.5 and 50 steps for inference. Since TANGOFLUX and Stable Audio Open allow variable audio generation length, we set the duration conditioning to 10 seconds and use the first 10 seconds of generated audio to perform the evaluation. We also report the effect of CFG scale in the appendix A.1.

**Evaluation metrics**. We evaluate TANGOFLUX using both objective and subjective metrics. For objective metrics, we report the 4 metrics: Fréchet Distance ($\text{FD}_{\text{openl3}}$) (Cramer et al., 2019), Kullback–Leibler divergence ($\text{KL}_{\text{passt}}$) , $\text{CLAP}_{\text{score}}$ and Inception Score (IS) (Salimans et al., 2016). We chose this set of metrics proposed by (Evans et al., 2024a) due to the capabilities to perform high-quality audio evaluation up to 48kHz. $\text{FD}_{\text{openl3}}$ evaluates the similarity between the statistics of a generated audio set and another reference audio set in the feature level space. A low $\text{FD}_{\text{openl3}}$ indicates that the generated audio is realistic and closely resembles the reference audio. $\text{KL}_{\text{passt}}$ computes the KL divergence over the probabilities of the labels between the generated and the reference audio given the state-of-the-art audio tagger **PaSST**. A low $\text{KL}_{\text{passt}}$ signifies the generated and reference audio share similar semantics tags. $\text{CLAP}_{\text{score}}$ is a reference-free metric that measures the cosine similarity between the audio as well as the text prompt. High $\text{CLAP}_{\text{score}}$ score denotes the generated audio aligns with the textual prompt. IS measures the specificity and coverage for a set of samples. A high IS score represents the diversity of the generated audio. We use stable-audio-metrics (Evans et al., 2024a) to compute $\text{FD}_{\text{openl3}}$, $\text{KL}_{\text{passt}}$, $\text{CLAP}_{\text{score}}$ and AudioLDM evaluation toolkit (Liu et al., 2023) to compute **IS**. Note that we use different CLAP checkpoints to create our preference dataset (*630k-audioset-best*) and to perform the evaluation (*630k-audioset-fusion-best*). [4]

[4] `https://huggingface.co/lukewys/laion_clap/blob/main/630k-audioset-fusion-best`

### 3.4 Human Evaluation

To evaluate the instruction-following capabilities and robustness of TTA models, we created 50 out-of-distribution complex captions, such as "A pile of coins spills onto a wooden table with a metallic clatter, followed by the hushed murmur of a tavern crowd and the creak of a swinging door." These captions describe multiple events (ranging from 3 to 6 per caption) and go beyond conventional or overused sounds, such as simple animal noises, footsteps, or city ambiance. Events were identified using GPT4o to evaluate the captions generated. Each of the generated prompts contains multiple events including several where the temporal order of the events must be maintained. Details of our caption generation template and samples of generated captions can be found in the Appendix A.2.

Following prior studies (Ghosal et al., 2023; Majumder et al., 2024), our subjective evaluation focuses on two primary attributes of the generated audio: overall audio quality (OVL) and relevance to the text input (REL). The OVL metric evaluates the general sound quality, including clarity and naturalness, irrespective of the alignment with the input prompt. In contrast, the REL metric specifically measures the alignment of the generated audio with the provided text input. Annotators rated each audio sample on a scale from 0 (worst) to 100 (best) for both OVL and REL. This evaluation was performed on 50 GPT4o-generated prompts, with each sample independently assessed by at least four annotators.

Additional details on the evaluation instructions and annotators can be found in Appendix A.2.

#### 3.4.1 Metrics

We report three key metrics for subjective evaluation:

**Scores:** The average of the scores assigned by individual annotators. Due to the subjective nature of these scores and the significant variance observed in the annotator scoring patterns, the ratings were normalized to z-scores at the annotator level: $z_{ij} = (s_{ij} - \mu_i)/\sigma_i$. $z_{ij}$: The z-score for annotator $i$'s score of model $M_j$. This is the score after applying z-score normalization. $s_{ij}$: The raw score assigned by annotator $i$ to model $j$. This is the original score before normalization. $\mu_i$: The mean score assigned by annotator $i$ across all models. It represents the central tendency of the annotator's scoring pattern. $\sigma_i$: The standard deviation of annotator $i$'s scores across all models. This measures the variability or spread in the annotator's ratings.

This normalization procedure adjusts the raw scores, centering them around the annotator's mean score and scaling by the annotator's score spread (standard deviation). This ensures that scores from different annotators are comparable, helping to mitigate individual scoring biases.

**Ranking:** Despite z-score normalization, the variability in annotator scoring can still introduce noise into the evaluation process. To address this, models are also ranked based on their absolute scores. We utilize the mean (average rank of a model), and mode (the most common rank of a model) as metrics for evaluating these rankings.

**Elo:** Elo-based evaluation, a widely adopted method in language model assessment, involves pairwise model comparisons. We first normalized the absolute scores of the models using z-score normalization and then derived Elo scores from these pairwise comparisons. Elo score mitigates the noise and inconsistencies observed in scoring and ranking techniques. Specifically, Elo considers the relative performance between models rather than relying solely on absolute or averaged scores, providing a more robust measure of model quality under subjective evaluation. While ranking-based evaluation provides an ordinal comparison of models, determining the order of performance (e.g., Model A ranks first, Model B ranks second), it does not capture the magnitude of differences between ranks. For instance, if the difference between the first and second rankers is minimal, this is not evident from ranks alone. Elo scoring addresses this limitation by integrating both ranking and pairwise performance data. In ranking-based systems, the rank $R_i$ of a model $M_i$ is determined purely by its position relative to others: $R_i =$ position of $M_i$ in the sorted list of models based on performance.. However, this approach fails to quantify: 1) The gap in performance between consecutive ranks. 2) The consistency of relative performance across different pairwise comparisons. Elo scoring provides a probabilistic measure of model performance based on pairwise comparisons. By leveraging annotator scores, Elo assigns a continuous score $E_i$ to each model $M_i$, capturing its relative strength.

| Model | #Params. | Duration | Steps | $FD_{openl3} \downarrow$ | $KL_{passt} \downarrow$ | $CLAP_{score} \uparrow$ | IS$\uparrow$ | Inference Time (s) |
|---|---|---|---|---|---|---|---|---|
| `AudioLDM 2-large` | 712M | 10 sec | 200 | 108.3 | 1.81 | 0.419 | 7.9 | 24.8 |
| `Stable Audio Open` | 1056M | 47 sec | 100 | 89.2 | 2.58 | 0.291 | 9.9 | 8.6 |
| `Tango 2` | 866M | 10 sec | 200 | 108.4 | **1.11** | 0.447 | 9.0 | 22.8 |
| TANGOFLUX-base | **515M** | 30 sec | 50 | 80.2 | 1.22 | 0.431 | 11.7 | **3.7** |
| TANGOFLUX | **515M** | 30 sec | 50 | **75.1** | 1.15 | **0.480** | **12.2** | **3.7** |

Table 1: Comparison of audio generation models across various metrics. Output length represents the duration of the generated audio. Metrics include $FD_{openl3}$ for Frechet Distance, $KL_{passt}$ for KL divergence, and $CLAP_{score}$ for alignment. All inference time is computed on the same A40 GPU. We report the trainable parameters in the #Params column.

| Model | #Params. | Duration | $FD_{openl3} \downarrow$ | $KL_{passt} \downarrow$ | $CLAP_{score} \uparrow$ | IS$\uparrow$ |
|---|---|---|---|---|---|---|
| `AudioLDM 2-large` | 712M | 10 sec | 107.9 | 1.83 | 0.415 | 7.3 |
| `Stable Audio Open` | 1056M | 47 sec | 88.5 | 2.67 | 0.286 | 9.3 |
| `Tango 2` | 866M | 10 sec | 108.3 | **1.14** | 0.452 | 8.4 |
| TANGOFLUX-base | **515M** | 30 sec | 79.7 | 1.23 | 0.438 | 10.7 |
| TANGOFLUX | **515M** | 30 sec | **75.2** | 1.20 | **0.488** | **11.1** |

Table 2: Comparison of text-to-audio models on multi-event inputs.

# 4 Results

## 4.1 Main Results

Table 1 compares TANGOFLUX with prior text-to-audio generation models on *AudioCaps* in terms of the objective metrics. Model performance on the prompts with more than one event, namely *multi-event* prompts, are reported in Table 2.

The results suggest that TANGOFLUX consistently outperforms the prior works on all objective metrics, except `Tango 2` on $KL_{passt}$. Interestingly, the margin on $CLAP_{score}$ between TANGOFLUX and baselines is higher when evaluated on *multi-event* prompts. This suggests that TANGOFLUX excels at understanding and generating audio for complex instructions involving multiple events, effectively capturing nuanced details and relationships within the text compared to the baselines.

## 4.2 Batched Online Data Generation is Necessary

To show the impact of generating new samples at each iteration, in Fig. 2 we present the results of 5 training iterations of CRPO, both with and without generating new data at each iteration. Our findings suggest that training on the same dataset over multiple iterations leads to quick performance saturation and eventual degradation. Specifically, for offline CRPO, the CLAP score decreases after the second iteration, while the KL increases significantly. By the final iteration, performance degradation is evident, with both the CLAP score and KL worse than the first iteration, emphasizing the limitations of using offline data. In contrast, the online CRPO with data generation at the beginning of each iteration consistently outperforms the offline CRPO in terms of both CLAP score and KL.

A possible explanation of this performance degradation could be reward over-optimization (Rafailov et al., 2024a; Gao et al., 2022). Previous work by (Kim et al., 2024a) demonstrated that the reference model serves as a lower bound in DPO training for language models. Several iterations of updating the reference model (lower bound) with the same dataset cause the current model to excessively minimize the loss in unexpected ways. In Section 4.5, we identify unexpected phenomena in loss minimization that could explain over-optimization. This over-optimization ultimately leads to performance degradation as shown by the spike in KL and drop in CLAP score.

## 4.3 CLAP as reward model

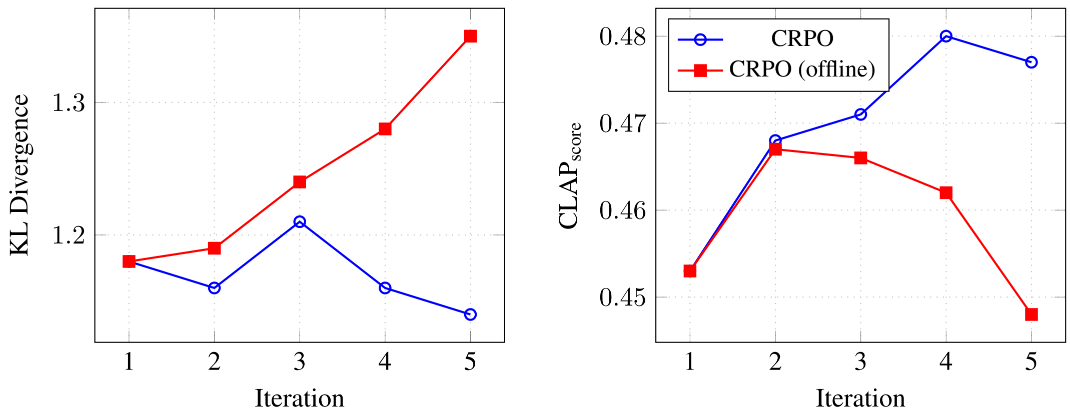



Figure 2: The trajectory of CLAP score and KL divergence across the training iterations. This plot shows the stark difference between online and offline training. Offline training clearly peaks early, by the second iteration, indicated by the peaking CLAP score and increasing KL. In contrast, the CLAP score of online training continues to increase until iteration 4, while the KL divergence has a clear downward trend throughout.

We experiment to validate that the CLAP model can serve as a proxy reward model for evaluating audio output. We experiment with evaluating TangoFlux with a Best-of-N policy, with $N \in \{1, 5, 10, 15\}$. We use the *630k-audioset-best.pt* checkpoint to rank the generated audio. We report the result in Table 4. Results suggest that increasing $N$ yield better CLAP$_{\text{score}}$ and KL$_{\text{passt}}$ while FD$_{\text{openl3}}$ remains about the same. This indicates that the CLAP model effectively identifies and ranks higher-quality audio outputs that better align with the textual descriptions, without sacrificing output diversity or quality as shown by the lower KL$_{\text{passt}}$ and similar FD$_{\text{openl3}}$.

| Dataset | FD$_{\text{openl3}}$ ↓ | KL$_{\text{passt}}$ ↓ | CLAP$_{\text{score}}$ ↑ |
|---|---|---|---|
| BATON | 80.5 | 1.20 | 0.437 |
| Audio Alpaca | 80.0 | 1.20 | 0.448 |
| CRPO | **79.1** | **1.18** | **0.453** |

Table 3: Comparison of difference preference dataset used for preference tuning. Metrics include FD$_{\text{openl3}}$ for Frechet Distance, KL$_{\text{passt}}$ for KL divergence, and CLAP$_{\text{score}}$ for alignment.

### 4.4 CRPO Dataset is Better than Other Audio Preference Datasets

To validate the effectiveness of CRPO in constructing preference datasets, we compared the performance of CRPO with two other audio preference datasets: Audio-Alpaca (Majumder et al., 2024) and BATON (Liao et al., 2024).

**BATON**: BATON collects human-annotated data by asking labelers to assign a binary score of 0 or 1 to each audio sample based on its alignment with a given prompt. A score of 1 indicates alignment, while 0 indicates misalignment. From this data, we construct a preference dataset by pairing audio samples scored as 1 (winners) with those scored as 0 (losers) for the same prompt, creating a set of winner-loser pairs.

**Audio-Alpaca**: Audio-Alpaca, in contrast, is already structured as a preference dataset, requiring no further processing.

We use the base model TangoFlux-`base` for preference optimization, conducting only one iteration since Audio-Alpaca and BATON are fixed datasets. Table 3 reports objective metrics FDopenl3, KLpasst, and CLAPscore, demonstrating that preference optimization with the CRPO dataset outperforms the other two audio preference datasets across all metrics. Despite its simplicity, CRPO proves highly effective for constructing audio preference datasets for optimization.

### 4.5 $\mathcal{L}_{\text{CRPO}}$ vs $\mathcal{L}_{\text{DPO-FM}}$

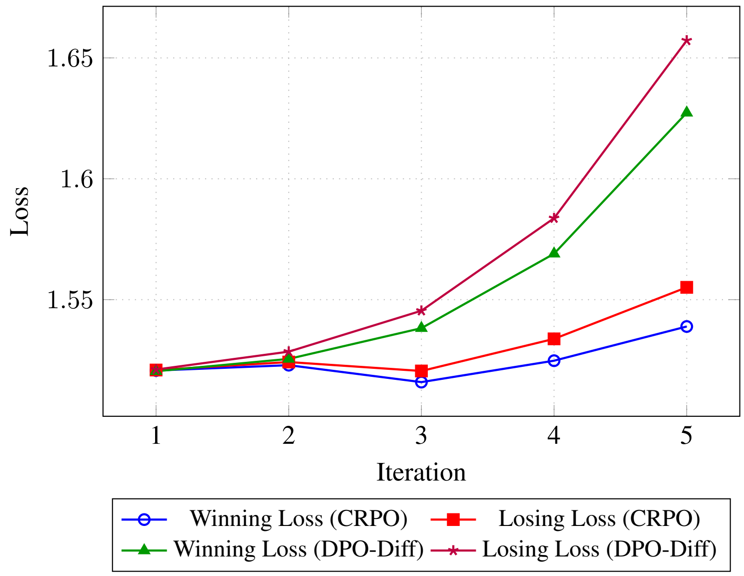
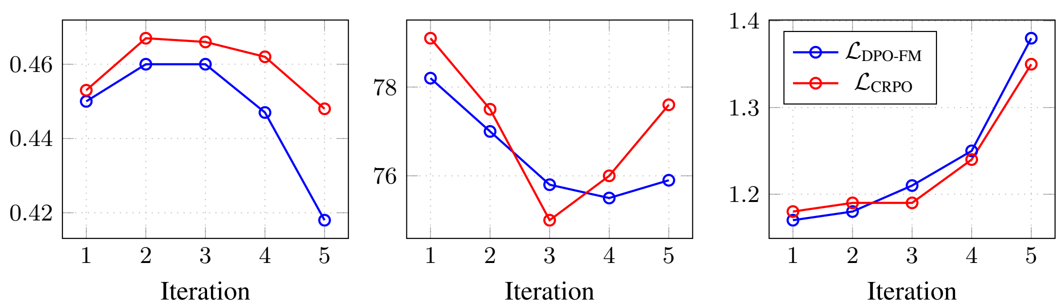



Figure 3: Winning and Losing losses of $\mathcal{L}_{\text{DPO-FM}}$ and $\mathcal{L}_{\text{CRPO}}$ at each iteration. Winning and Losing losses increase each iteration, as well as their margin.

We investigate whether using DPO to align rectified flow increases both winning and losing losses of Eq. (5) while increasing the margin of them simultaneously. To do so, we calculate the average winning and losing losses on the training set using the final checkpoint (epoch 8) from each iteration. We present the result in Fig. 3. We also present benchmark performance on *AudioCaps* training with $\mathcal{L}_{\text{CRPO}}$ and $\mathcal{L}_{\text{DPO-FM}}$ in Fig. 4. Here, we investigate only with offline data such that we use the fixed dataset generated by TangoFlux-`base`.

| Model | N | $\text{FD}_{\text{openl3}} \downarrow$ | $\text{KL}_{\text{passt}} \downarrow$ | $\text{CLAP}_{\text{score}} \uparrow$ |
|---|---|---|---|---|
| TangoFlux | 1 | 75.0 | 1.15 | 0.480 |
| | 5 | 74.3 | 1.14 | 0.494 |
| | 10 | 75.8 | 1.08 | 0.499 |
| | 15 | 75.1 | 1.11 | 0.502 |
| `Tango 2` | 1 | 108.4 | 1.11 | 0.447 |
| | 5 | 108.8 | 1.05 | 0.467 |
| | 10 | 108.4 | 1.08 | 0.474 |
| | 15 | 108.7 | 1.06 | 0.473 |

Table 4: Comparison of different preference datasets used for preference tuning. Metrics include $\text{FD}_{\text{openl3}}$ for Fréchet Distance, $\text{KL}_{\text{passt}}$ for KL divergence, and $\text{CLAP}_{\text{score}}$ for alignment.




(a) $\text{CLAP}_{\text{score}}$ (b) $\text{FD}_{\text{openl3}}$ (c) $\text{KL}_{\text{passt}}$

Figure 4: Comparison of metrics across iterations for $\mathcal{L}_{\text{DPO-FM}}$ and $\mathcal{L}_{\text{CRPO}}$.

As shown in Fig. 3, the winning and losing losses for both $\mathcal{L}_{\text{CRPO}}$ and $\mathcal{L}_{\text{DPO-FM}}$ increase with each iteration, along with their margin. Despite the increase in losses, benchmark performance improves, with $\mathcal{L}_{\text{CRPO}}$ achieving superior results in $\text{CLAP}_{\text{score}}$ while maintaining similar $\text{KL}_{\text{passt}}$ and $\text{FD}_{\text{openl3}}$ across all iterations. We observe a notable acceleration in loss growth after iteration 3, which may indicate performance saturation or degradation. In contrast, $\mathcal{L}_{\text{CRPO}}$ exhibits a more gradual and stable increase in loss, maintaining a smaller margin and more controlled growth, leading to less performance degradation compared to $\mathcal{L}_{\text{DPO-FM}}$. This highlights the role of the *winning loss* as a regularizer in controlling the overall optimization dynamics. Specifically, adding *winning loss* helps to stabilize the training process by preventing the model from excessively focusing on increasing the margin through increasing both *winning loss* and *losing loss*.

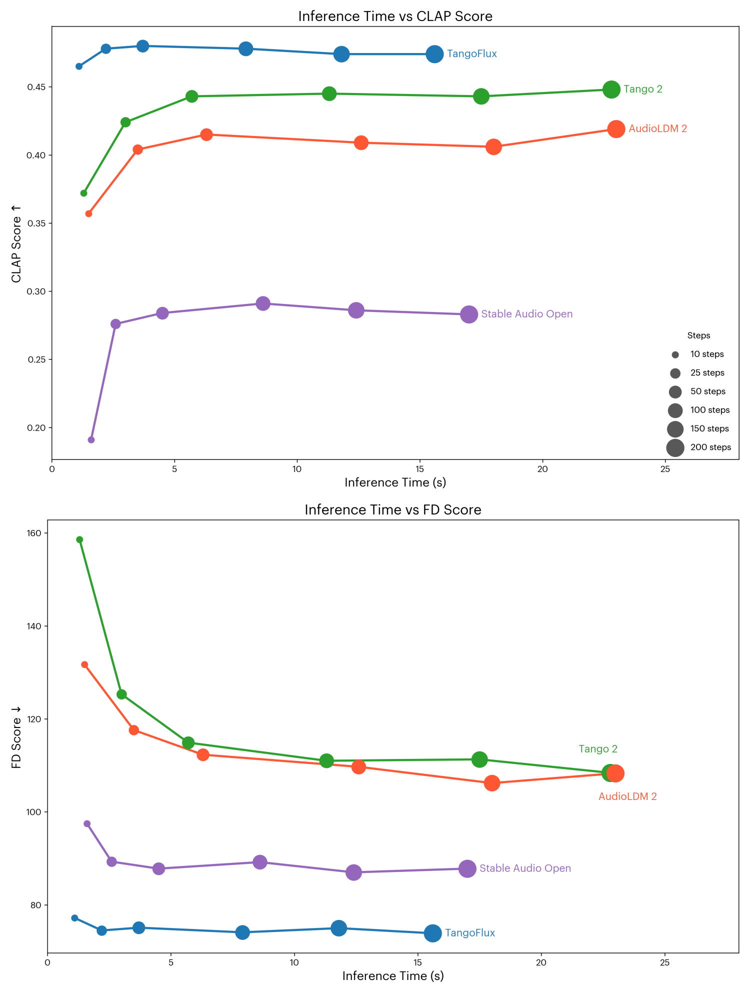



Figure 5: CLAP and FD Scores vs Inference Time for each model. We plot this for steps count of 10, 25, 50, 100, 150 and 200.

Interestingly, our findings show that in aligning rectified flow with DPO, both winning and losing losses increase, while the margin between them widens—similar to the behavior observed when aligning LLMs with DPO (Rafailov et al., 2024b). This behavior is consistent across both $\mathcal{L}_{\text{CRPO}}$ and $\mathcal{L}_{\text{DPO-FM}}$, where performance improves despite the seemingly counterintuitive nature of this phenomenon, as also noted by (Rafailov et al., 2024a) in the context of LLMs.

### 4.6 Inference Time and Performance Comparison

We compare inference times, CLAP scores, and FD scores across models for steps 10, 25, 50, 100, 150, and 200, as shown in Figure 5. TangoFlux demonstrates a remarkable balance between efficiency and performance, consistently achieving higher CLAP scores and lower FD scores while requiring significantly less inference time compared to other models. For example, at 50 steps, TangoFlux achieves a CLAP score of 0.480 and an FD score of 75.1 in just 3.7 seconds. In comparison, `Stable Audio Open` requires 4.5 seconds for the same step count but only achieves a

CLAP score of 0.284 (41% lower than TangoFlux) and an FD score of 87.8 (17% worse than TangoFlux). This demonstrates that TangoFlux achieves superior performance metrics in less time. Additionally, at a lower step count of 10, TangoFlux maintains strong performance with a CLAP score of 0.465 and an FD score of 77.2 in just 1.1 seconds. In contrast, `Audioldm2` at the same step count achieves a lower CLAP score of 0.357 (23% lower) and a significantly worse FD score of 131.7 (70% higher), while requiring 1.5 seconds (36% more time). We also observe that reducing the step count from 200 to 10 has a minimal impact on TangoFlux's performance, highlighting its robustness. Specifically, TangoFlux's CLAP score decreases by only 3.2% (from 0.480 to 0.465), and its FD score increases by only 4.5% (from 73.9 to 77.2). In contrast, `Tango 2` shows a larger degradation, with its CLAP score decreasing by 16.0% (from 0.443 to 0.372) and its FD score increasing by 37.8% (from 108.4 to 158.6).

These results highlight TangoFlux's effectiveness in delivering high-quality outputs with lower computational requirements, making it a highly efficient choice for scenarios where inference time is critical.

**TL;DR**

1. **Model Comparison**:
   - TangoFlux outperforms prior works in almost all objective metrics on *AudioCaps*, especially for prompts with multiple events.
   - It achieves superior performance in $\text{FD}_{\text{openl3}}$, $\text{CLAP}_{\text{score}}$, and Inception Score (IS), with notable efficiency gains (lowest inference time).
   - Only `Tango 2` marginally surpasses TangoFlux in $\text{KL}_{\text{passt}}$.
2. **Multi-Event Prompts**:
   - The margin in $\text{CLAP}_{\text{score}}$ between TangoFlux and baselines is larger for multi-event inputs, demonstrating its capability to handle complex and nuanced scenarios.
3. **Training Strategies:**
   - Online batched data generation significantly outperforms offline strategies, preventing performance degradation caused by over-optimization.
   - Online training maintains consistent improvement across $\text{CLAP}_{\text{score}}$ and $\text{KL}_{\text{passt}}$ over iterations.
4. **Preference Optimization:**
   - CRPO dataset leads to better results than other preference datasets like BATON and Audio-Alpaca across all metrics.
   - Larger $N$ in the Best-of-N policy enhances $\text{CLAP}_{\text{score}}$ and $\text{KL}_{\text{passt}}$, validating CLAP as an effective reward model.
5. **Optimization Techniques**:
   - $\mathcal{L}_{\text{CRPO}}$ demonstrates more stable and effective optimization than $\mathcal{L}_{\text{DPO-FM}}$, with reduced performance saturation and better benchmark results.
   - The controlled growth in optimization metrics with $\mathcal{L}_{\text{CRPO}}$ highlights its robustness for rectified training processes.
6. **Inference Time**:
   - While delivering superior performance, TangoFlux also boasts a much lower inference time, resulting in greater efficiency compared to other models.
   - TangoFlux shows less performance decline compared to other models when sampling at fewer steps.

### 4.7 Human Evaluation Results

The results of the human evaluation are presented in Table 5, with detailed comparisons of the models across the evaluated metrics: z-scores, rankings, and Elo scores for both overall audio quality (OVL) and relevance to the text input (REL). Below, we provide an analysis of the findings.

**Z-scores:** The z-scores offer a normalized perspective on annotator evaluations, mitigating individual biases by transforming the absolute variable into a standard normal variable with zero mean and one standard deviation. Among the models, TANGOFLUX demonstrated the highest performance across both metrics, with z-scores of 0.2486 for OVL and 0.6919 for REL. This indicates its superior quality and strong alignment with the input prompts. Conversely, `AudioLDM 2` scored the lowest with z-scores of -0.3020 (OVL) and -0.4936 (REL), suggesting both lower sound quality and weaker adherence to textual inputs compared to the other models.

**Ranking:** Rankings provide an alternative ordinal measure of performance, capturing both the mean and mode of model rankings. Consistent with the z-score findings, TANGOFLUX achieved the best rankings with a mean rank of 1.7 for OVL and 1.1 for REL, and a mode rank of 2 (OVL) and 1 (REL). This reinforces the model's superior position in the subjective evaluations. `AudioLDM 2` consistently ranked the lowest, with mean rankings of 3.5 (OVL) and 3.7 (REL), and mode rankings of 4 for both metrics. Interestingly, `StableAudio` and `Tango 2` showed competitive mean ranks for OVL, both at 2.4, but diverged on REL where `Tango 2` outperformed `StableAudio` (1.9 vs. 3.3 in mean rank). However, curiously, `StableAudio` has two modes 1 and 3 for OVL, indicating a polarized perception by the annotators. The lower mode could be explained by a possible bias induced by the stark misalignment between the prompt and audio output, indicated by the mean and mode of 3.3 and 3, respectively, in terms of ranking by REL.

**Elo Scores:** The Elo scores provide a probabilistic and continuous measure of model performance, offering insights into the magnitude of differences in relative performance. Here, TANGOFLUX again excelled, achieving the highest Elo scores for both OVL (1,501) and REL (1,628). The Elo results highlight the robustness of TANGOFLUX, as it consistently outperformed other models in pairwise comparisons. `Tango 2` emerged as the second-best performer, with Elo scores of 1,419 (OVL) and 1,507 (REL). `StableAudio` followed, showing competitive performance in OVL (1,444) but a weaker REL score (1,268). As observed in other metrics, `AudioLDM 2` ranked last with the lowest Elo scores (1,236 for OVL and 1,196 for REL).

**TL;DR**

**1.** TANGOFLUX consistently demonstrated superior performance across all metrics, highlighting its strength in generating high-quality, text-relevant audio. This is particularly evident in its significant lead in the REL metrics, showcasing its robust capability to align with complex, multi-event prompts.

**2.** `Tango 2` performed strongly in REL, reflecting its alignment capability. However, it slightly lagged behind `TangoFlux` in OVL, indicating potential room for improvement in audio clarity and naturalness.

**3.** `Stable Audio Open` displayed competitive performance in OVL, but its REL scores suggest limitations in accurately and faithfully representing complex text inputs.

**4.** `AudioLDM2` consistently underperformed across all metrics, reflecting challenges in both audio quality and relevance to complex prompts. This positions it as the least preferred model in this evaluation.

| Model | z-scores | | Ranking | | | | Elo | |
|---|---|---|---|---|---|---|---|---|
| | OVL | REL | OVL | | REL | | OVL | REL |
| | | | Mean | Mode | Mean | Mode | | |
| `AudioLDM 2` | -0.3020 | -0.4936 | 3.5 | 4 | 3.7 | 4 | 1,236 | 1,196 |
| `Stable Audio Open` | 0.0723 | -0.3584 | 2.4 | 1, 3 | 3.3 | 3 | 1,444 | 1,268 |
| `Tango 2` | -0.019 | 0.1602 | 2.4 | 2 | 1.9 | 2 | 1,419 | 1,507 |
| TANGOFLUX | **0.2486** | **0.6919** | **1.7** | **2** | **1.1** | **1** | **1,501** | **1,628** |

Table 5: Human evaluation results on two attributes: OVL (overall quality) and REL (relevance). We report the z-scores, ranking, and Elo scores to mitigate individual annotator biases and present a relative performance comparison.

## 5 Related Works

**Text-To-Audio Generation.** TTA Generation has garnered attention lately due to models such as AudioLDM series model (Liu et al., 2024b; 2023), Tango series model (Majumder et al., 2024; Ghosal et al., 2023; Kong et al., 2024) and Stable Audio series model (Evans et al., 2024a;c;b). These models commonly adopt the diffusion framework (Song & Ermon, 2020; Rombach et al., 2022; Song et al., 2022; Ho et al., 2020), which trains a latent diffusion model conditioned on either T5 embedding or CLAP embedding. However, another common framework for TTA generation is the flow matching framework which was employed in models such as VoiceBox (Le et al., 2023), AudioBox (Vyas et al., 2023), FlashAudio (Liu et al., 2024c).

**Alignment Method.** Preference optimization is the standard approach for aligning LLMs, achieved either by training a reward model to capture human preferences (Ouyang et al., 2022) or by using the LLM itself as the reward model (Rafailov et al., 2024c). Recent advances improve this process through iterative alignment, leveraging human annotators to construct preference pairs or utilizing pre-trained reward models. (Kim et al., 2024a; Chen et al., 2024; Gulcehre et al., 2023; Yuan et al., 2024). Verifiable answers can enhance the construction of preference pairs. For diffusion and flow-based models, Diffusion-DPO shows that these models can be aligned similarly (Wallace et al., 2023). However, constructing preference pairs for TTA remains challenging due to the lack of ”gold” audio for given text prompts and the subjective nature of audio. Tango2 addresses this by using prompt perturbation, while BATON (Liao et al., 2024) relies on human annotation to construct preference pairs which is not a scalable solution.

## 6 Conclusion

We introduce **TangoFlux**, a fast flow-based text-to-audio model aligned using synthetic preference data generated online during training. Objective and human evaluations show that **TangoFlux** produces audio more representative of user prompts than existing diffusion-based models, achieving state-of-the-art performance with significantly fewer parameters. Additionally, **TangoFlux** demonstrates greater robustness, maintaining performance even when sampling with fewer time steps. These advancements make **TangoFlux** a practical and scalable solution for widespread adoption.

## References


Michael S. Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants, 2023. URL https://arxiv.org/abs/2209.15571.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models, 2024. URL https://arxiv.org/abs/2401.01335.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models, 2023. URL https://arxiv.org/abs/2311.07919.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen2-audio technical report, 2024. URL https://arxiv.org/abs/2407.10759.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models, 2022. URL https://arxiv.org/abs/2210.11416.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation, 2024. URL https://arxiv.org/abs/2306.05284.

Aurora Linh Cramer, Ho-Hsiang Wu, Justin Salamon, and Juan Pablo Bello. Look, listen, and learn more: Design choices for deep audio embeddings. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3852–3856, 2019. doi: 10.1109/ICASSP.2019.8682475.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis, 2024. URL https://arxiv.org/abs/2403.03206.

Zach Evans, CJ Carr, Josiah Taylor, Scott H. Hawley, and Jordi Pons. Fast timing-conditioned latent audio diffusion, 2024a. URL https://arxiv.org/abs/2402.04825.

Zach Evans, Julian D. Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. Long-form music generation with latent diffusion, 2024b. URL https://arxiv.org/abs/2404.10301.

Zach Evans, Julian D. Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. Stable audio open, 2024c. URL https://arxiv.org/abs/2407.14358.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization, 2022. URL https://arxiv.org/abs/2210.10760.

Deepanway Ghosal, Navonil Majumder, Ambuj Mehrish, and Soujanya Poria. Text-to-audio generation using instruction-tuned llm and latent diffusion model, 2023. URL https://arxiv.org/abs/2304.13731.

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, Wolfgang Macherey, Arnaud Doucet, Orhan Firat, and Nando de Freitas. Reinforced self-training (rest) for language modeling, 2023. URL https://arxiv.org/abs/2308.08998.

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, 2024. URL https://arxiv.org/abs/2303.09556.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. URL https://arxiv.org/abs/2207.12598.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. URL https://arxiv.org/abs/2006.11239.

Jiawei Huang, Yi Ren, Rongjie Huang, Dongchao Yang, Zhenhui Ye, Chen Zhang, Jinglin Liu, Xiang Yin, Zejun Ma, and Zhou Zhao. Make-an-audio 2: Temporal-enhanced text-to-audio generation, 2023a. URL https://arxiv.org/abs/2305.18474.

Rongjie Huang, Jiawei Huang, Dongchao Yang, Yi Ren, Luping Liu, Mingze Li, Zhenhui Ye, Jinglin Liu, Xiang Yin, and Zhou Zhao. Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models, 2023b. URL https://arxiv.org/abs/2301.12661.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and Madian Khabsa. Llama guard: Llm-based input-output safeguard for human-ai conversations, 2023. URL https://arxiv.org/abs/2312.06674.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models, 2022. URL https://arxiv.org/abs/2206.00364.

Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. AudioCaps: Generating captions for audios in the wild. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 119–132, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1011. URL https://aclanthology.org/N19-1011.

Dahyun Kim, Yungi Kim, Wonho Song, Hyeonwoo Kim, Yunsu Kim, Sanghoon Kim, and Chanjun Park. sdpo: Don't use your data all at once, 2024a. URL https://arxiv.org/abs/2403.19270.

Myunsoo Kim, Donghyeon Ki, Seong-Woong Shim, and Byung-Jun Lee. Adaptive non-uniform timestep sampling for diffusion model training, 2024b. URL https://arxiv.org/abs/2411.09998.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022. URL https://arxiv.org/abs/1312.6114.

Zhifeng Kong, Sang gil Lee, Deepanway Ghosal, Navonil Majumder, Ambuj Mehrish, Rafael Valle, Soujanya Poria, and Bryan Catanzaro. Improving text-to-audio models with synthetic captions, 2024. URL https://arxiv.org/abs/2406.15487.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, Noah A. Smith, Yizhong Wang, Pradeep Dasigi, and Hannaneh Hajishirzi. Tulu 3: Pushing frontiers in open language model post-training, 2024a. URL https://arxiv.org/abs/2411.15124.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. Rewardbench: Evaluating reward models for language modeling, 2024b. URL https://arxiv.org/abs/2403.13787.

Matthew Le, Apoorv Vyas, Bowen Shi, Brian Karrer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, and Wei-Ning Hsu. Voicebox: Text-guided multilingual universal speech generation at scale, 2023. URL https://arxiv.org/abs/2306.15687.

Huan Liao, Haonan Han, Kai Yang, Tianjiao Du, Rui Yang, Zunnan Xu, Qinmei Xu, Jingquan Liu, Jiasheng Lu, and Xiu Li. Baton: Aligning text-to-audio model with human preference feedback, 2024. URL https://arxiv.org/abs/2402.00744.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling, 2023. URL https://arxiv.org/abs/2210.02747.

Alexander H. Liu, Matt Le, Apoorv Vyas, Bowen Shi, Andros Tjandra, and Wei-Ning Hsu. Generative pre-training for speech with flow matching, 2024a. URL https://arxiv.org/abs/2310.16338.

Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D. Plumbley. Audioldm: Text-to-audio generation with latent diffusion models, 2023. URL https://arxiv.org/abs/2301.12503.

Haohe Liu, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Qiao Tian, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D. Plumbley. Audioldm 2: Learning holistic audio generation with self-supervised pretraining, 2024b. URL https://arxiv.org/abs/2308.05734.

Huadai Liu, Jialei Wang, Rongjie Huang, Yang Liu, Heng Lu, Wei Xue, and Zhou Zhao. Flashaudio: Rectified flows for fast and high-fidelity text-to-audio generation, 2024c. URL https://arxiv.org/abs/2410.12266.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow, 2022. URL https://arxiv.org/abs/2209.03003.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. URL https://arxiv.org/abs/1711.05101.

Navonil Majumder, Chia-Yu Hung, Deepanway Ghosal, Wei-Ning Hsu, Rada Mihalcea, and Soujanya Poria. Tango 2: Aligning diffusion-based text-to-audio generations through direct preference optimization, 2024. URL https://arxiv.org/abs/2404.09956.

Xinhao Mei, Chutong Meng, Haohe Liu, Qiuqiang Kong, Tom Ko, Chengqi Zhao, Mark D. Plumbley, Yuexian Zou, and Wenwu Wang. WavCaps: A ChatGPT-assisted weakly-labelled audio captioning dataset for audio-language multimodal research. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pp. 1–15, 2024.

Derek Onken, Samy Wu Fung, Xingjian Li, and Lars Ruthotto. Ot-flow: Fast and accurate continuous normalizing flows via optimal transport, 2021. URL https://arxiv.org/abs/2006.00104.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022. URL https://arxiv.org/abs/2203.02155.

Arka Pal, Deep Karkhanis, Samuel Dooley, Manley Roberts, Siddartha Naidu, and Colin White. Smaug: Fixing failure modes of preference optimisation with dpo-positive, 2024. URL https://arxiv.org/abs/2402.13228.

Richard Yuanzhe Pang, Weizhe Yuan, Kyunghyun Cho, He He, Sainbayar Sukhbaatar, and Jason Weston. Iterative reasoning preference optimization, 2024. URL https://arxiv.org/abs/2404.19733.

William Peebles and Saining Xie. Scalable diffusion models with transformers, 2023. URL https://arxiv.org/abs/2212.09748.

Rafael Rafailov, Yaswanth Chittepu, Ryan Park, Harshit Sikchi, Joey Hejna, Bradley Knox, Chelsea Finn, and Scott Niekum. Scaling laws for reward model overoptimization in direct alignment algorithms, 2024a. URL https://arxiv.org/abs/2406.02900.

Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From $r$ to $q^*$: Your language model is secretly a q-function, 2024b. URL https://arxiv.org/abs/2404.12358.

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, 2024c. URL https://arxiv.org/abs/2305.18290.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023. URL https://arxiv.org/abs/1910.10683.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. URL https://arxiv.org/abs/2112.10752.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans, 2016. URL https://arxiv.org/abs/1606.03498.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. URL https://arxiv.org/abs/2010.02502.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution, 2020. URL https://arxiv.org/abs/1907.05600.

Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. Salmonn: Towards generic hearing abilities for large language models, 2024. URL https://arxiv.org/abs/2310.13289.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. URL https://arxiv.org/abs/1706.03762.

Apoorv Vyas, Bowen Shi, Matthew Le, Andros Tjandra, Yi-Chiao Wu, Baishan Guo, Jiemin Zhang, Xinyue Zhang, Robert Adkins, William Ngan, Jeff Wang, Ivan Cruz, Bapi Akula, Akinniyi Akinyemi, Brian Ellis, Rashel Moritz, Yael Yungster, Alice Rakotoarison, Liang Tan, Chris Summers, Carleigh Wood, Joshua Lane, Mary Williamson, and Wei-Ning Hsu. Audiobox: Unified audio generation with natural language prompts, 2023. URL https://arxiv.org/abs/2312.15821.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization, 2023. URL https://arxiv.org/abs/2311.12908.

Yusong Wu*, Ke Chen*, Tianyu Zhang*, Yuchen Hui*, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, 2023.

Jinlong Xue, Yayue Deng, Yingming Gao, and Ya Li. Auffusion: Leveraging the power of diffusion and large language models for text-to-audio generation, 2024. URL https://arxiv.org/abs/2401.01044.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models, 2024. URL https://arxiv.org/abs/2401.10020.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. Star: Bootstrapping reasoning with reasoning, 2022. URL https://arxiv.org/abs/2203.14465.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. URL https://arxiv.org/abs/2306.05685.

| Model | Steps | CFG | $\text{FD}_{\text{openl3}} \downarrow$ | $\text{KL}_{\text{passt}} \downarrow$ | $\text{CLAP}_{\text{score}} \uparrow$ |
|---|---|---|---|---|---|
| TangoFlux | 50 | 3.0 | 77.7 | **1.14** | 0.479 |
| | 50 | 3.5 | 76.1 | **1.14** | **0.481** |
| | 50 | 4.0 | 74.9 | 1.15 | 0.476 |
| | 50 | 4.5 | 75.1 | 1.15 | 0.480 |
| | 50 | 5.0 | **74.6** | 1.15 | 0.472 |

Table 6: TangoFlux with different classifier free guidance (CFG) values.

# A Appendix

## A.1 Effect of CFG scale

We conduct an ablation of the effect of CFG scale for TangoFlux and show the result in Table 6. It reveals a trade-off: higher CFG values improve FD score (lower FD) but slightly reduce semantic alignment (CLAP score), which peaks at CFG=3.5. The results emphasize CFG=3.5 as the optimal balance between fidelity and semantic relevance.

## A.2 Human Evaluation

The human evaluation was performed using a web-based Gradio[5] app. Each annotator was presented with 20 prompts, each having four audio samples generated by four distinct text-to-audio models, shuffled randomly, as shown in Fig. 6. Before the annotation process, the annotators were instructed with the following directive:

Welcome *username*

# Instructions for evaluating audio clips

**Please carefully read the instructions below**.

## Task

You are to evaluate four 10-second-long audio outputs to each of the 20 prompts below. These four outputs are from four different models. You are to judge each output with respect to two qualities:

- Overall Quality (OVL): The overall quality of the audio is to be judged on a scale from 0 to 100: 0 being absolute noise with no discernible feature. Whereas, 100 is perfect. **Overall fidelity, clarity, and noisiness of the audio are important here.**
- Relevance (REL): The extent of audio alignment with the prompt is to be judged on a scale from 0 to 100: with 0 being absolute irrelevance to the input description. Whereas, 100 is a perfect representation of the input description. **You are to judge if the concepts from the input prompt appear in the audio in the described temporal order.**

**You may want to compare the audios of the same prompt with each other during the evaluation.**

## Listening guide

1. Please use a head/earphone to listen to minimize exposure to the external noise.
2. Please move to a quiet place as well, if possible.

## UI guide

[5] https://www.gradio.app

1. Each audio clip has two attributes OVL and REL below. You may select the appropriate option from the dropdown list.
2. To save your judgments, please click on any of the *save* buttons. All the *save* buttons function identically. They are placed everywhere to avoid the need to scroll to save.

Hope the instructions were clear. Please feel free to reach out to us for any queries.

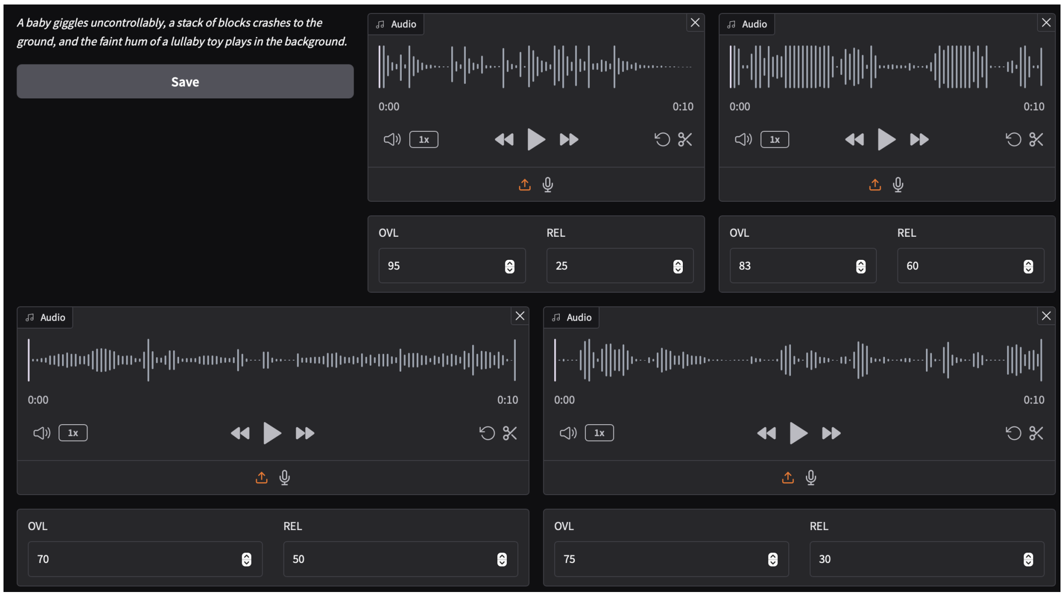



Figure 6: The Gradio-based human evaluation form created for the annotators to score the model generated audios with respect to the input prompts.

### A.2.1 Prompts Used in the Evaluation

| **Prompts** | **Multiple Events** | **Temporal Events** |
|---|---|---|
| A robotic arm whirs frantically while an electric plasma arc crackles and a metallic voice counts down ominously, interspersed with glass vials clinking to the floor. | ✓ | ✓ |
| Unfamiliar chirps overlap with a low, throbbing hum as bioluminescent plants audibly crackle and squelch with movement. | ✓ | ✗ |
| Dripping water echoes sharply, a distant growl reverberates through the cavern, and soft scraping metal suggests something lurking unseen. | ✓ | ✗ |
| Alarms blare with rising urgency as fragments clatter against a metallic hull, interrupted by a faint hiss of escaping air. | ✓ | ✓ |
| Hundreds of tiny wings buzz with a chaotic pitch shift, joined by the faint clattering of mandibles and an organic squish as they collide. | ✓ | ✗ |
| Jagged rocks crumble underfoot while distant ocean waves crash below, punctuated by the sudden snap of a rope. | ✓ | ✓ |

| | | |
|---|---|---|
| Digital beeps and chirps meld with overlapping chatter in multiple languages, as automated drones whiz past, scanning barcodes audibly. | ✓ | ✗ |
| Rusted swings creak in rhythmic disarray, a faint mechanical laugh stutters from a distant speaker, and the sound of gravel crunches under unseen footsteps. | ✓ | ✗ |
| Bubbling lava gurgles ominously, instruments beep irregularly, and faint crackling signals static from a failing radio. | ✓ | ✓ |
| Tiny pops and hisses of chemical reactions intermingle with the rhythmic pumping of a centrifuge and the soft whirr of air filtration. | ✓ | ✗ |
| The faint hiss of a gas leak grows louder as metal chains rattle and a single marble rolls across the floor. | ✓ | ✓ |
| A hand slaps a table sharply, followed by the shuffle of playing cards and the hum of an overhead fan. | ✓ | ✓ |
| A train horn blares in the distance as a bicycle bell chimes and a soda can pops open with a fizzy hiss. | ✓ | ✗ |
| A drawer creaks open, papers rustle wildly, and the sharp click of a lock snapping shut echoes. | ✓ | ✗ |
| A burst of static interrupts soft typing sounds, followed by the distant chirp of a pager and a cough. | ✓ | ✓ |
| A heavy book thuds onto a desk, accompanied by the faint buzz of a fluorescent light and a muffled sneeze. | ✓ | ✗ |
| The sharp squeak of sneakers on a gym floor blends with the rhythmic bounce of a basketball and the screech of a metal door. | ✓ | ✗ |
| An elevator dings, its doors sliding open, as muffled voices overlap with the shuffle of heavy bags. | ✓ | ✗ |
| A clock ticks steadily, a light switch clicks on, and the crackle of a fire igniting briefly fills the silence. | ✓ | ✓ |
| A fork scrapes a plate, water drips slowly into a sink, and the faint hum of a refrigerator lingers in the background. | ✓ | ✗ |
| A cat hisses sharply as glass shatters nearby, followed by hurried footsteps and the slam of a closing door. | ✓ | ✓ |
| A parade marches through a town square, with drumbeats pounding, children clapping, and a horse neighing amidst the commotion. | ✓ | ✓ |
| A basketball bounces rhythmically on a court, shoes squeak against the floor, and a referee's whistle cuts through the air. | ✓ | ✗ |
| A baby giggles uncontrollably, a stack of blocks crashes to the ground, and the faint hum of a lullaby toy plays in the background. | ✓ | ✗ |
| The rumble of a subway train grows louder, followed by the screech of brakes and muffled announcements over a crackling speaker. | ✓ | ✓ |

| | | |
|---|---|---|
| A beekeeper moves carefully as bees buzz intensely, a smoker puffs softly, and wooden frames creak as they're lifted. | ✓ | ✗ |
| A dog shakes off water with a noisy splatter, a bicycle bell rings, and a distant lawnmower hums faintly in the background. | ✓ | ✗ |
| Books fall off a shelf with a heavy thud, a chair scrapes loudly across a wooden floor, and a surprised gasp echoes. | ✓ | ✗ |
| A soccer ball hits a goalpost with a metallic clang, followed by cheers, clapping, and the distant hum of a commentator's voice. | ✓ | ✓ |
| A hiker's pole taps against rocks, a mountain goat bleats sharply, and loose gravel tumbles noisily down a steep slope. | ✓ | ✓ |
| A rooster crows loudly at dawn, joined by the rustle of feathers and the crunch of chicken feed scattered on the ground. | ✓ | ✗ |
| A carpenter saws through wood with steady strokes, a hammer strikes nails rhythmically, and a measuring tape snaps back with a metallic zing. | ✓ | ✗ |
| A frog splashes into a pond as dragonflies buzz nearby, accompanied by the distant croak of toads echoing through the marsh. | ✓ | ✗ |
| The crack of a whip startles a herd of cattle, their hooves clatter against a dirt path as a rancher shouts commands. | ✓ | ✗ |
| A paper shredder whirs noisily, the rustle of documents being fed in grows louder, and a stapler clicks shut in rapid succession. | ✓ | ✗ |
| An elephant trumpets in the savanna as a herd stomps through dry grass, accompanied by the buzz of flies and the distant roar of a lion. | ✓ | ✗ |
| A mime claps silently as a juggling act clinks glass balls together, and a crowd bursts into laughter at the clatter of a dropped prop. | ✓ | ✗ |
| A train conductor blows a sharp whistle, metal wheels screech on the rails, and passengers murmur while settling into their seats. | ✓ | ✓ |
| A squirrel chitters nervously as acorns drop from a tree, landing with dull thuds, while leaves rustle above in quick bursts of movement. | ✓ | ✗ |
| A blacksmith hammers molten iron with rhythmic clangs, a bellows pumps air with a whoosh, and sparks sizzle on a stone floor. | ✓ | ✗ |
| A skateboard grinds loudly against a metal rail, followed by the sharp slap of wheels hitting pavement and a triumphant cheer from the rider. | ✓ | ✗ |

| | | |
|---|---|---|
| An old typewriter clacks rapidly as paper rustles with each keystroke, interrupted by the sharp ding of the carriage return. | ✓ | ✗ |
| A pack of wolves howls in unison as dry leaves crunch underfoot, and the faint trickle of a nearby stream echoes through the forest. | ✓ | ✗ |

Table 7: Prompts used in human evaluation and their characteristics.