AUDIO-JOURNEY: OPEN DOMAIN LATENT DIFFUSION BASED TEXT-TO-AUDIO GENERATION

Jackson Michaels[†Δ], Juncheng B Li[⋆Δ], Laura Yao[⋆], Lijun Yu[⋆], Zach Wood-Doughty[†], Florian Metze[⋆]

ABSTRACT

Despite recent progress, machine learning for open domain audio generation is lagging behind models for image, text, speech, and music. In this paper, we leverage state-of-the-art (SOTA) Large Language Models (LLMs) to augment the existing weak labels of the audio dataset to enrich captions; we adopt SOTA video-captioning model to automatically generate video caption, and we again use LLMs to merge the audio-visual captions to form a rich dataset of large-scale. In our experiment, we first verified that our Audio+Visual Caption is of high quality against baselines and ground truth (12.5% gain in semantic score against baselines). Using this dataset we constructed a Latent Diffusion Model to generate in the encodec encoding latent space. Our model is novel in the current SOTA audio generation landscape due to our generation space, text encoder, noise schedule, and attention mechanism functioning together to provide competitive open domain audio generation.

The samples, models, and implementation will be at `https://audiojourney.github.io`.

Index Terms— Deep Learning, Open Domain Audio Generation, Audio-Visual Training, Large Language Models

1. INTRODUCTION

Audio Generation has shown high quality performance in limited domains, primarily focused on speech and music. This performance has not been shown in the open domain generation, whereas image generation has long been able to generate arbitrary images. Our primary goal, therefore, is to design and train a model capable of open-domain audio generation and contribute to closing this gap in performance.

To achieve this goal, we harness the power of generative models, Large Language Models, and SOTA audio compression to generate any sound class present in Audioset [1], the worlds largest audio dataset. Recent work has demonstrated the ability of Large Language Models (LLMs) to extract an enormous amount of knowledge from billions of text inputs [2, 3]. The in-context generation capabilities of these models allowed automatic conversion from the weak labeling in audioset to human-like captions. By using LLMs in conjunction with Video-to-Text (VTT) models such as BLIP2 [4] and few shot engineered prompts we created synthetic prompts combining context from the weak labels with important visual queues for a more complete description of the sound scene.

Having constructed a substantially enriched audio-text dataset, we can train a powerful generative model for audio. We encode each audio clip into a post-quantization encodec embedding space [6], and train a score-based latent diffusion model to reconstruct the audio, conditional on a T5[5] encoding of our generated captions. We delve deeper into our modeling choices and motivations in Section§ 4. Our entire system is illustrated in Figure 1. Our experimental results reveal that our diffusion model outperforms baseline models such as AudioLDM [7, 8] in generating high quality outputs.

Our work, motivated by the need for high quality open domain audio generation, makes the following contributions:

1. We showcase our methods ability for converting from weak video captions to information rich human-like captions
2. We successfully train a diffusion model using a pretrained latent encoder-decoder, bypassing the need to train a VAE and vocoder (e.g., HiFiGAN [9]), which demonstrates excellent generation quality.
3. We provide valuable intuitions on attention mechanisms for textually guided generation.

2. BACKGROUND & RELATED WORKS

Diffusion Models for Audio: Denoising Diffusion Probabilistic Models (Diffusion Models) [10] are a class of score-based generative models to predict how a data point diffuses over set time steps. The motivation for these models is as follows, given an image and a known forward diffusion process defined as:

$$\begin{aligned} q(\mathbf{x}_t|\mathbf{x}_{t-1}) &= \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \\ q(\mathbf{x}_{1:T}|\mathbf{x}_0) &= \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}) \end{aligned} \tag{1}$$

model and predict the reverse diffusion process defined as:

$$\begin{aligned} p_\theta(\mathbf{x}_{0:T}) &= p(\mathbf{x}_T)\prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) \\ p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) &= \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t)) \end{aligned} \tag{2}$$

Where $\mathbf{x}$ is the noised latent, t is the timestep, p_θ is an unconditional denoising model to approximate conditional proba-

† Northwestern University, Chicago, IL
⋆ Carnegie Mellon University, Pittsburgh, PA
Δ These authors contributed equally to this work.

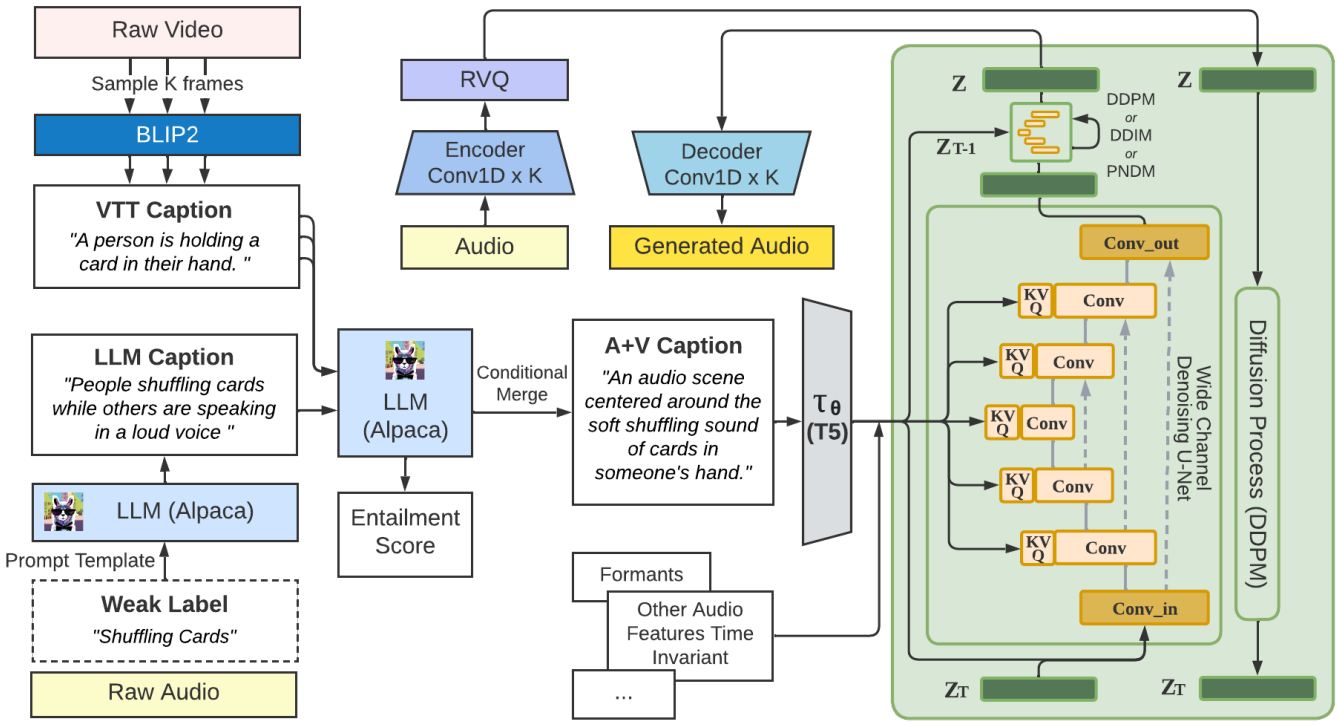

Fig. 1. Our overall system diagram. BLIP2:[4], T5:[5], Conv_in and Conv_out layers are modified to 128 channels. Audio Encoder, decoder and residual vector quantized (RVQ) layers are pretrained by Encodec [6].

bilities, and β_t is a value from 0-1 retrieved from the noise scheduler. This model is parameterized through a score estimator $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$, which is equivalent to $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, y = \varnothing)$. It's noteworthy that both p_θ and $\boldsymbol{\epsilon}_\theta$ can be effectively learned using a single neural network, in this work, we implement it with a high-channel UNet.

Once trained, the reverse diffusion process can map random noise into new samples from the training data's distribution. While accurately modeling the proper probability density function (PDF) of a sufficiently complex dataset $P(X)$ is intractable, diffusion models instead model the gradient or stein score of the PDF: $\nabla_x \log P(X)$. Through integration, this score function conserves the information stored within the PDF without being intractable to compute [11], allowing for superior data coverage compared to other generative models.

Diffusion models have excelled at tasks including image synthesis [12] and audio generation ([7, 13, 8]). In contrast to other generative models, diffusion models suffer from a significant drawback: the extended duration required for sampling. This happens because the iterative denoising process requires multiple steps instead of a single forward pass employed by GANs and VAEs for generation. Many modern diffusion models address this limitation by operating in the latent space of an autoencoder, significantly reducing the dimensionality required for generation [14]. This approach improves generation quality while simultaneously lowering sampling and training time.

Several recent works have used latent diffusion models for audio generation. AudioLDM and DiffSound [7, 13] generate audio by applying diffusion to spectrogram representations of sound. However, in addition to the denoising network, these approaches require training both a new VAE and an entirely separate vocoder (e.g., HiFi-GAN [9]) to convert from the generated spectrograms back into waveforms.

Other Representation Learning Models: Directly modeling the score-based measure of the PDF allows diffusion models to significantly improve the diversity of their generation compared to other methods. For instance, GANs [15] and VAEs are both powerful generative models but suffer from poor coverage of the underlying data distribution [12] and often generate lower-quality samples [11] respectively. Diffusion models generate samples that are of comparable or superior performance to GANs while simultaneously producing better probability coverage of the underlying distribution [11].

Caption Generation: Another common issue for audio datasets is the lack of high quality captions. Other efforts, such as WavCaps [16] and AudioCaps [17], have taken various approaches to this challenge including human captioners and ChatGPT. Both methods fail to scale effectively due to the often prohibitive cost of human captioners and premium closed-source APIs.

3. HARNESSING LLMS TO GENERATE AUDIO+VISUAL CAPTIONS: PROMPT ENGINEERING

Prompt Context	Similarity Score	Vocabulary Size
Zero-Shot	0.474	-
One-Shot	0.605	-
Few-Shot	0.686	-
Audio-Visual Merge	**0.750**	**1480**
WavCaps[16]	0.667	509

Table 1. Average similarity scores ranging from 0.0 - 1.0 between captions generated with Alpaca prompts and ground truth captions (AudioCaps) where zero-shot means no examples given to Alpaca, one-shot is one example, and few-shot is several examples, automatic evaluation metrics compared to the ground truth, and vocabulary size for the sample captions generated.

We leverage the power of LLMs to increase the descriptiveness of the audio captions on datasets such as AudioSet[1], which only contains weak labels without descriptive captions. We use Alpaca [3] (INT8-quantized) and engineered prompts to generate a richer caption for every sample in AudioSet balanced and unbalanced sets, unifying the list of audio classes and introducing the relevant concepts. Alpaca is an open-source instruction-following model fine-tuned on the Llama-7b model [3]. To generate text captions from class label lists, we used the following prompt with added examples: *"For each of these, summarize the sounds into a single sentence: \n describe a situation with all of these sounds together: [LIST OF LABELS]"*. A limitation of the Alpaca model is its tendency to add unnecessary details or ignore relevant labels when generating captions. By adding examples to the prompt, we leveraged the in-context learning ability of Alpaca to enrich our captions and better utilize the "hallucination" of LLMs. This helped add missing features not expressed in the original labels such as in Figure 1 where the caption "shuffling cards" is correctly extended to include a human in the scene. The appendix covers more details on these prompts and provides examples.

Building upon the potential of LLMs, this study significantly improved the descriptiveness of captions in AudioSet using Alpaca, an open-source instruction-following model. Notably, our strategy also involved a novel integration of video-based captions generated from the state-of-the-art BLIP2 model with our enriched audio captions. We utilized Alpaca again to merge these disparate data sources, effectively consolidating audio and visual context while reducing inaccuracies. This approach yielded more nuanced and rich captions, demonstrating the value of merging LLMs, Alpaca, and video-to-text models to elevate data representation and quality.

To assess the performance of our captions and the improvements additional context provided in their generation we analyzed a subset of AudioCaps [17] captions (human generated captions) and their corresponding audio clips against our generated captions, scoring each on a scale from 0 to 1 on the similarity to AudioCaps[17] while referencing the actual audio clip as shown in Table 1. Since most automatic metrics are based on n-gram similarity or longest common subsequence we decided to use a human metric because of the large vocabulary variation as shown in the subset vocabulary size in Table 1. Additionally, the WavCaps [16] model is fine-tuned on AudioCaps which helps boost their automatic metric scores in comparison to our approach.

4. TEXT-GUIDED DIFFUSION IN QUANTIZED LATENT SPACE

Text Encoder τ_θ: We experimented with several text encoders for the prompt conditioned generation including CLIP [18], CLAP [19], and T5 [5]. While we initially used CLAP for its textual-audio joint embedding, we found it performed worse than T5. T5 has a larger embedding space than CLAP or CLIP, requiring an additional linear projection to connect it to the U-Net. We found this detail crucial to changing the text encoder while preserving pretraining knowledge due to the projection layer functioning similarly to an adapter layer [20]. The final consideration for text encoding is using an attention mask on the text embedding. While the mask experimentally had varying effects, shown in Table 2 and also discussed in [14] as "unmasked" expert model, we selected T5-unmasked as our final model.

Noise schedule: The denoising process is trained across a fixed number T of DDPM [10] steps. A naive approach would be to simply compute noise percent as a linear (scaled) interpolation from 0 - 1 across the timesteps to control the β_t in Eq. 1, as done in AudioLDM [7]. With recent work [21] showing the superior performance of non-linear schedule vs. linear schedule, we adopt cosine noise scheduling without the need to tune the $\beta_{start}, \beta_{end}$ from $t = 0$ to $t = T$: $\beta_t = \text{Clip}(1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}}, 0.999) \quad \bar{\alpha}_t = \frac{f(t)}{f(0)}$ where $f(t) = \cos\left(\frac{\frac{t}{T}+\delta}{1+\delta} \cdot \frac{\pi}{2}\right)^2$, where δ is a small offset. During the sampling steps, we explored the usages of DDPM [10], and accelerated approximating schedulers: DDIM[22], and PNDM [23]. Our empirical results show the best speed-quality tradeoff for PNDM [23], thus the results reported are PNDM results.

U-Net ϵ_θ Design: We refer to our U-Net as a Wide Channel U-Net due to our choice to train and generate in a 128-channel latent space instead of the typical one or three channels used in SOTA audio generation. We had two main observations that informed this decision: first, the receptive field of the U-Net convolutional blocks could not fully explore the 128×504 latent space representations from the Encodec encoder; second, the latent encoding showed little variance within the 128 dimensions. We were able to leverage the second observation to correct the first by reshaping the latent vectors from a one-channel 128×512 image to a 128-channel 21×24 image. We then normalized each channel to a mean of zero and std

of one representation to assist the U-Net in learning the noise: $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. With this new representation, the convolutional blocks are able to contain the entire image in their receptive field without losing resolution and result in higher fidelity audio. After generation, these transformations can be fully inverted to allow for decoding back into a waveform.

Another difference between our diffusion approach and that of past work is our use of cross-attention instead of embedding adding. This change enables us to conserve text embedding features. In self-attention, the text embedding is first concatenated to the image embedding, subjecting it to modifications at each layer of the U-Net. For cross-attention, we instead use the unmodified text for attention at each layer of the U-Net, maintaining the text embeddings fidelity throughout generation and improving class guidance. In our cross-attention, the text conditioning y remains unchanged between diffusion steps; in self-attention, y would first be incorporated into the noised latent z_i before diffusion begins. In the self-attention setting, as the noisy latent passes through the U-Net the text embedding become increasingly interwoven into the noised latent and loses its reliability.

Generation Latent Space: In this work, we dramatically reduce the engineering effort and GPU hours needed to train an audio diffusion model. Rather than training our own VAE and vocoder, we use Encodec [6], an off-the-shelf VQ-GAN model which has demonstrated competitive MUSHRA [24] in high-fidelity audio generation. This allows us to focus all our training resources on the denoising U-Net. The Encodec model [6] we selected consists of an encoder, vector quantizer, and decoder stages.

Despite not being used in training, we pre-computed the entirety of AudioSet 2M into discrete code vectors and saved these new compressed versions to disk for training. We trained our model on the decoder embedding, a continuous representation with a shape of 128×504. This slight change from reading raw audio files to reading compact discrete codes from disk substantially accelerated training for two reasons. First, the I/O read times became significantly shorter as the files consist of 8×504 features instead of $160,000 \times 1$ for a $> 95\%$ reduction. Second, without needing to store these large waveforms in memory, we increased our batch sizes significantly, greatly improving training time.

5. EXPERIMENT AND RESULTS

Generation Quality: Table 2 presents interesting quantitative comparisons between our models and previous SOTA models. We performed our evaluation similarly to AudioLDM [7]; first, we extracted all captions from the AudioCaps [17] test set and generated samples based on each of these captions. We then compare FD scores against the ground truth audio from the AudioCaps [17] test set for each model, IS and KL scores are similarly measured.

This shows two noteworthy trends: first, our generative model holds up to current SOTA models despite training exclusively on AudioSet with Alpaca-generated captions, whereas previous SOTA works include multiple other datasets. Second, the inverse trends of our FD vs. KL scores imply a trade-off between quality and diversity. This intuition is reflected in the models with said scores. CLAP model's superior KL scores are a reflection of the similarity between CLAP and CLIP, which our models were pretrained on. T5-based model's superior FD scores imply T5 assists in generation more than CLAP despite lower variance.

Model	Datasets	FD	IS	KL
DiffSound [13]	AS+AC	47.68	4.01	7.76
AudioGen [25]	AS+AC+8	-	-	2.09
AudioLDM [7]	AS+AC+2	23.32	8.13	1.59
Ours-CLAP	AS	67.6	1.63	**0.127**
Ours-CLAP-M	AS	55.5	1.64	0.134
Ours-T5-M	AS	13.14	1.64	0.209
Ours-T5	AS	**12.09**	**1.64**	0.259

Table 2. comparison between our models and current SOTA models. -M indicates an attention mask was used. These models are scored on frechet distance (FD), inception score (IS), and kullback–leibler divergence (KL). Our scores are computed by comparison against AudioCaps [17] test set explained further in § 5. Scores for DiffSound, AudioGen, and AudioLDM are from [7].

Surprisingly, the masked T5 model performed worse than the unmasked model. This is counter-intuitive, given that attention masking is a common mechanism used to handle variable length inputs. Because the max sequence length is significantly larger than the number of audio caption tokens in our captions, the attention score A and the V tensor result in a low-rank dot-product which result in a low-ranked Z. In an extreme case of when token length is 1, this is reduced to a rank-1 vector dot product of column by row. We believe this low-ranked representation of Z is suboptimal to the full-ranked version when unmasked. Therefore, to compensate for the above issue, we reduce our max token length to 50, and use the unmasked versions in our best-performing recipe.

6. REFERENCES

[1] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[2] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learn-

ers," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[3] R. Taori, I. Gulrajani, T. Zhang, Y. Dubois, X. Li, C. Guestrin, P. Liang, and T. B. Hashimoto, "Stanford alpaca: An instruction-following llama model," https://github.com/tatsu-lab/stanford_alpaca, 2023.

[4] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," *arXiv preprint arXiv:2301.12597*, 2023.

[5] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.

[6] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," *arXiv preprint arXiv:2210.13438*, 2022.

[7] H. Liu, Z. Chen, Y. Yuan, X. Mei, X. Liu, D. Mandic, W. Wang, and M. D. Plumbley, "Audioldm: Text-to-audio generation with latent diffusion models," 2023.

[8] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, "Diffwave: A versatile diffusion model for audio synthesis," 2021.

[9] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17 022–17 033, 2020.

[10] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[11] Y. Song and S. Ermon, "Generative modeling by estimating gradients of the data distribution," 2020.

[12] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021.

[13] D. Yang, J. Yu, H. Wang, W. Wang, C. Weng, Y. Zou, and D. Yu, "Diffsound: Discrete diffusion model for text-to-sound generation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.

[14] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 684–10 695.

[15] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 28, pp. 2880–2894, 2020. [Online]. Available: https://doi.org/10.1109/TASLP.2020.3030497

[16] X. Mei, C. Meng, H. Liu, Q. Kong, T. Ko, C. Zhao, M. D. Plumbley, Y. Zou, and W. Wang, "Wavcaps: A chatgpt-assisted weakly-labelled audio captioning dataset for audio-language multimodal research," *arXiv preprint arXiv:2303.17395*, 2023.

[17] C. D. Kim, B. Kim, H. Lee, and G. Kim, "Audiocaps: Generating captions for audios in the wild," in *NAACL-HLT*, 2019.

[18] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8748–8763.

[19] Y. Wu*, K. Chen*, T. Zhang*, Y. Hui*, T. Berg-Kirkpatrick, and S. 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.

[20] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. de Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-efficient transfer learning for NLP," *CoRR*, vol. abs/1902.00751, 2019. [Online]. Available: http://arxiv.org/abs/1902.00751

[21] T. Chen, "On the importance of noise scheduling for diffusion models," *arXiv preprint arXiv:2301.10972*, 2023.

[22] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[23] L. Liu, Y. Ren, Z. Lin, and Z. Zhao, "Pseudo numerical methods for diffusion models on manifolds," *arXiv preprint arXiv:2202.09778*, 2022.

[24] B. Series, "Method for the subjective assessment of intermediate quality level of audio systems," *International Telecommunication Union Radiocommunication Assembly*, 2014.

[25] F. Kreuk, G. Synnaeve, A. Polyak, U. Singer, A. Défossez, J. Copet, D. Parikh, Y. Taigman, and Y. Adi, "Audiogen: Textually guided audio generation," *arXiv preprint arXiv:2209.15352*, 2022.