Audio-Journey: Efficient Visual+LLM-aided Audio Encodec Diffusion

Juncheng B Li *[1] Jackson Michaels *[2] Laura Yao [1] Lijun Yu [1] Zach Wood-Doughty [2] Florian Metze [1]

Abstract

Despite recent progress, machine learning for the audio domain is limited by the availability of high-quality data. Visual information already presented in a video should complement the information in audio. In this paper, we leverage a state-of-the-art (SOTA) LLM to perform data augmentation, bypassing the need for costly human annotation; We also leverage audio encodec model to allow for extremely efficient adaptation of a pre-trained text-to-image generation latent diffusion model to perform text-to-audio generation. Our approach exemplifies a promising method for augmenting low-resource audio datasets. The samples, models, and implementation will be at `https://audiojourney.github.io`.

1. Introduction

The field of machine learning for the audio domain, despite making significant strides, is currently constrained by the scarcity of high-quality data. The largest datasets available, including AudioSet (Gemmeke et al., 2017), CLAP (Wu* et al., 2023), and VGGSound (Chen et al., 2020), comprise in total less than 3 million examples, falling orders of magnitude short of datasets in other domains, such as the Laion 5B Image-Text dataset (Schuhmann et al., 2022). Even for these datasets, the majority of annotations are collected through weak labeling, which may introduce noise. Our primary goal, therefore, is to efficiently augment the training resources based on the limited existing resources.

To achieve this goal, we harness the power of generative models to efficiently create large amounts of diverse, high-quality audio captions. Recent work has demonstrated the ability of Large Language Models (LLMs) to extract an enormous amount of knowledge from billions of text inputs (Brown et al., 2020; Taori et al., 2023). The in-context generation capabilities of these models are so convincing as to raise a concern about their ability to "hallucinate" false responses that mislead human users (Bender et al., 2021). As part of a careful data augmentation strategy, however, these hallucinations can be used effectively to generate captions that can significantly enrich the existing weak labels that annotate audio datasets. To make full use of existing datasets that were sourced from videos, we believe the visual information in the videos can complement the audio. Video-to-Text (VTT) models can efficiently extract and transcribe visual cues into text representation, and state-of-the-art (SOTA) models such as BLIP2 (Li et al., 2023) have demonstrated robust performance at this task. We apply BLIP2 to frames sampled from the video to generate a set of possible captions that capture visual information.

We generated audio captions by prompting an LLM (quantized) with existing weak label annotations. Independently, we have also generated video captions using a VTT model. We once again leverage LLMs to generate a merged audio-visual caption that provides an enriched annotation for the audio clip. Our methodology so far is visualized in the left half of Figure 1, starting with the Raw Audio and Raw Video and leading up to the Audio+Visual (A+V) Caption. Altogether, applying our methodology to AudioSet (Gemmeke et al., 2017) produces a dataset of over 2 million audio clips with significantly-enriched captions, allowing us to avoid the exorbitant expenses associated with hiring human annotators. Our human evaluation suggests that our captions are quantitatively better than previously-generated captions (Mei et al., 2023) and qualitatively comparable to human annotations released by AudioCAPs (Kim et al., 2019).

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 embedding space (Défossez et al., 2022) for efficiency purposes, and train a score-based latent diffusion model to reconstruct the audio, conditional on a T5(Raffel et al., 2020) 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 (Liu et al., 2023; Kong et al., 2021) in generating higher-quality outputs. Additionally, we prove that this model can replicate all

[*] Equal contribution [1] Carnegie Mellon University, Pittsburgh, PA [2] Northwestern University, Chicago, IL. Correspondence to: Juncheng B Li <junchenl@cs.cmu.edu>, Jackson Michaels <jacksonmichaels2021@u.northwestern.edu>.

Work presented at the ES-FoMo Workshop at ICML 2023

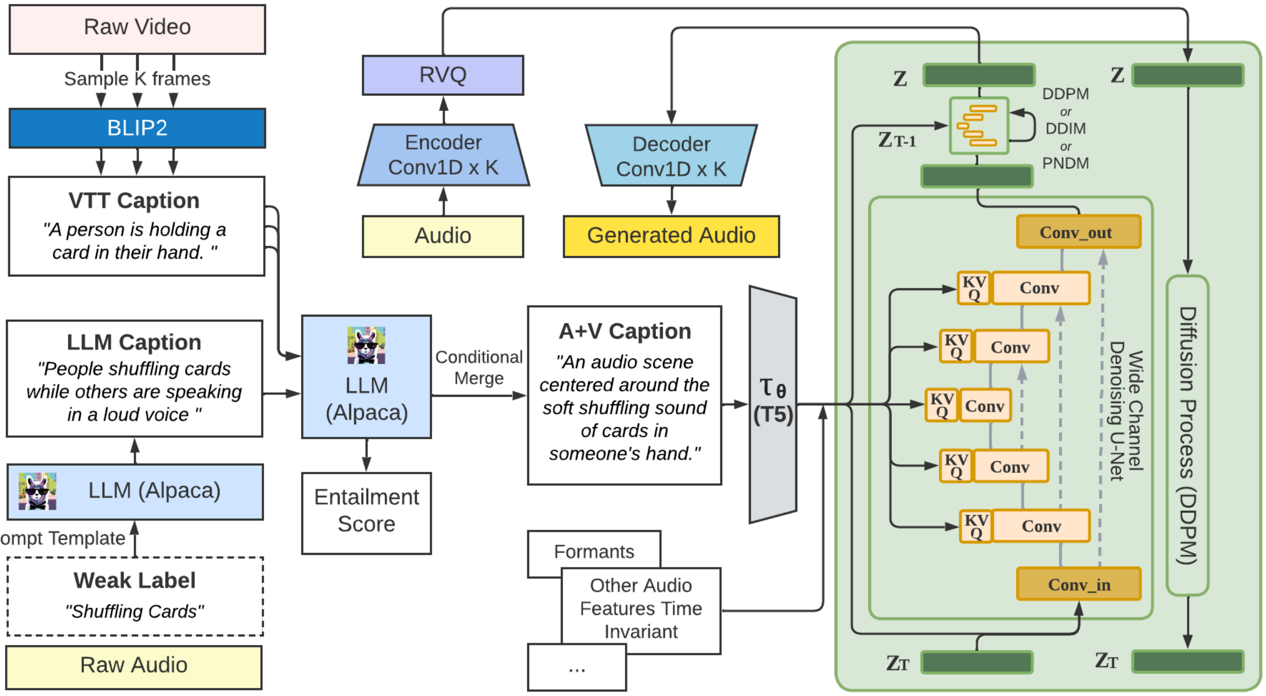


Figure 1. Our overall system diagram. BLIP2:(Li et al., 2023), T5:(Raffel et al., 2020), Conv_in and Conv_out layers are modified to 128 channels. Audio Encoder, decoder and residual vector quantized (RVQ) layers are pretrained by Encodec (Défossez et al., 2022).

the supplemental capabilities of Stable Diffusion (Rombach et al., 2022), including efficient fine-tuning like ControlNet (Zhang and Agrawala, 2023) and Dreambooth (Ruiz et al., 2022), among others[1].

Our work, motivated by the need to augment low-resource audio datasets, makes the following contributions:

1. We release a new large-scale audio dataset with efficiently generated captions.
2. We efficiently train a diffusion model using a pre-trained latent encoder-decoder, bypassing the need to train a separate VAE and vocoder (e.g., HiFiGAN (Kong et al., 2020a)), and still achieves competitive inference speed.
3. We showcase our diffusion model's ability to generate *useful* audio data.

2. Background & Related Works

Diffusion Models: Denoising Diffusion Probabilistic Models (Diffusion Models) (Ho et al., 2020) 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, model and predict the reverse diffusion process. 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 (Song and Ermon, 2020), allowing for superior data coverage compared to other generative models.

Diffusion models have excelled at tasks including image synthesis (Dhariwal and Nichol, 2021) and audio generation ((Liu et al., 2023; Yang et al., 2023; Kong et al., 2021)). 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 (Rombach et al., 2022). This approach improves image quality while simultaneously lowering sampling and training time.

Latent Diffusion: Several recent works have used latent diffusion models for audio generation. AudioLDM and DiffSound (Liu et al., 2023; Yang et al., 2023) 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 (Kong et al., 2020a)) to

[1] Refer to the Appendix for more details.

Model	Datasets	FD	IS	KL
DiffSound	AS+AC	47.68	4.01	7.76
AudioGen	AS+AC+8	-	-	2.09
AudioLDM-L-Full	AS+AC+2	23.32	8.13	1.59
Ours-CLAP	AS	67.6	1.63	**0.127**
Ours-CLAP-masked	AS	55.5	1.64	0.134
Ours-T5-masked	AS	13.14	1.64	0.209
Ours-T5	AS	**12.09**	**1.64**	0.259

Table 1. comparison between our models and current SOTA models. These models are scored on frechet distance (FD), inception score (IS), and kullback–leibler divergence (KL). Our scores are computed by comparison against AudioCaps (Kim et al., 2019) test set. Scores for DiffSound (Yang et al., 2023), AudioGen (Kreuk et al., 2022), and AudioLDM are from (Liu et al., 2023).

convert from the generated spectrograms back into waveforms. This requires significant engineering effort and may be difficult to reproduce or generalize to new domains (e.g., if FFT parameters for spectrograms are hard-coded).

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 (Défossez et al., 2022), an off-the-shelf VQ-GAN model which has demonstrated competitive MUSHRA (Series, 2014) in high-fidelity audio generation. This allows us to focus all our training resources on the denoising U-Net. While using the pretrained VQ-GAN prevents us from jointly learning the latent space and the diffusion model, our model is still able to adapt to the Encodec model's latent space. Our use of the Encodec model is similar to that of AudioGen (Kreuk et al., 2022), except they instead train an auto-regressive model. AudioGen also does not have public training code, making it a blackbox model and difficult to replicate.

Automatic Caption Generation: Another common issue for audio datasets is the lack of high quality captions. Other efforts, such as WavCaps (Mei et al., 2023) and AudioCaps (Kim et al., 2019), have taken various approaches to this challenge. AudioCaps employed human judges to create audio-text pairs for over 46 thousand samples taken from AudioSet, whereas WavCaps used ChatGPT to generate captions based on the weak labels resulting in a new dataset of approximately 400k samples. Both methods fail to scale effectively due to the often prohibitive cost of human judges and premium closed-source APIs.

3. Harnessing LLMs to generate Audio+Visual Captions: Prompt Engineering

We leverage the power of LLMs to increment the descriptiveness of the audio captions on datasets such as AudioSet(Gemmeke et al., 2017), which only contains weak labels without descriptive captions. We use Alpaca (Taori et al., 2023) (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 (Taori et al., 2023). To generate text captions from class label lists, we used the following prompt: *"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. 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.

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 (Radford et al., 2021), CLAP (Wu* et al., 2023), and T5 (Raffel et al., 2020). The model originally used CLIP but, CLIP is trained on image-text pairs. Next we tested CLAP for its textual-audio joint embedding, though 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 pre-training knowledge.

The final consideration for text encoding is using an attention mask on the text embedding. We experimented with and without attention masks with varying results. Experimentally, as shown in Table 1, masking had different effects on CLAP and T5-based models. Intuition would say that masking the T5 embedding would yield a more significant improvement as its fixed length is larger than CLAP; however, the addition of the linear projection layer between the text embedding and the U-Net functions as a type of masking resulted in inferior performance when combined with an attention mask, as was also discussed in (Rombach et al., 2022) as "unmasked" expert model. The second major benefit to adding a projection layer is it allowed us to completely skip fine-tuning the text encoder model as this layer functions as an adapter between T5 and our U-Net.

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: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. 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 (e.g., AudioLDM (Liu et al., 2023)) is our use of cross-attention instead of embedding adding. 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 embedding's fidelity throughout generation and improving class guidance. The appendix covers the details of our attention mechanism.

Format	Batch Size	Disk	Z-Shape	Train Steps	Inference Time
Waveform	8	1.4 TB	1×160k	-	-
Spectrogram	12	1.2 TB	128×1024	60k	>30s
Encodec	192	**63 GB**	8×504	**60k**	**14.7s**
AudioLDM	8	2.3 TB	$8 \times 16 \times 250$	1.75M	25.8s
AudioGen	256	2.0 TB	-	200k	-
DiffSound	16	-	80×860	8days	49.6s

Table 2. Efficiency comparison. Our pipelins (top): Waveform is prohibitively large, thus not successful; We regenerate wave from spectrogram using Griffin-Lim Algo. Our training is done on AudioSet 2M using 8 A100 GPUs, inference is 1 A100 GPU. SOTA works (bottom) AudioLDM(Liu et al., 2023) DiffSound(Yang et al., 2023) (Kreuk et al., 2022) respective papers or computed. (more details clarified in the appendix)

Generation Latent Space: The Encodec model (Défossez et al., 2022) we selected consists of an encoder, vector quantizer, and decoder stages. (Figure 2) Initially, we attempted to directly learn the discrete "codebook" of RVQ as this has the highest degree of compression, at only 8×504, and could leverage the generative benefits of the Encodec codebook and decoder stage. We pre-computed the entirety of AudioSet 2M into discrete vectors and saved these new compressed versions to disk for training. However, during experimentation, we observed nearly 0 decreases in train loss over time, as diffusion is only suitable for continuous vector space (Song and Ermon, 2020), suggesting an AutoRegressive model might better suit this. We trained our next model on the decoder embedding, which is of larger size, at 128×504, but is continuous. This slight change improved training substantially, only requiring us to perform one forward pass of the pre-trained dequantizer while having 2 huge advantages: First, the I/O read times became significantly shorter as the files consist of 8×504 features instead of $160,000 \times 1$, resulting in a complete copy of AudioSet 2M that only takes 63 GB compared to 1.4 TB before for a $> 95\%$ reduction. Second, without needing to store these large waveforms in memory, we increased our batch sizes significantly, greatly improving training time.

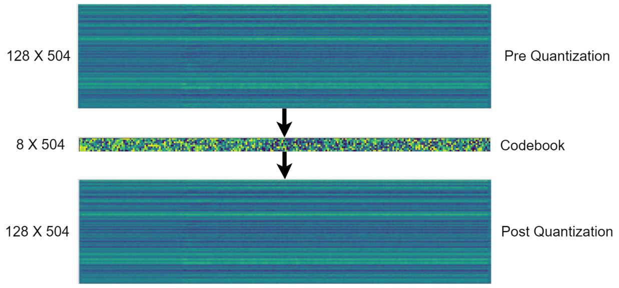


Figure 2. Encodec (Défossez et al., 2022) quantization process for encoder latent representations. Encoder-decoder pair not included in figure. Codebook stage is cropped in the figure to improve visibility.

5. Experiment and Results

Table 1 presents quantitative comparisons between our models and previous SOTA models. We performed our evaluation similarly to AudioLDM (Liu et al., 2023); first, we extracted all captions from the AudioCaps (Kim et al., 2019) test set and generated samples based on each of these captions. We then compare FD scores against the ground truth audio from the AudioCaps (Kim et al., 2019) test set for each model, IS and KL scores are similarly measured. This shows two noteworthy trends: 1) 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. 2) 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 these models were pretrained on. T5-based model's superior FD scores imply T5 assists in generation more than CLAP despite lower variance. Table 2 shows our approaches efficiency advantage.

References

Awad, G., Curtis, K., Butt, A. A., Fiscus, J., Godil, A., Lee, Y., Delgado, A., Zhang, J., Godard, E., Chocot, B., Diduch, L., Liu, J., Graham, Y., , and Quénot, G. (2022). An overview on the evaluated video retrieval tasks at trecvid 2022. In *Proceedings of TRECVID 2022*. NIST, USA.

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Chen, H., Xie, W., Vedaldi, A., and Zisserman, A. (2020). Vggsound: A large-scale audio-visual dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 721–725. IEEE.

Défossez, A., Copet, J., Synnaeve, G., and Adi, Y. (2022). High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dhariwal, P. and Nichol, A. (2021). Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794.

Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A. H., Chechik, G., and Cohen-Or, D. (2022). An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*.

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

Gong, Y., Chung, Y.-A., and Glass, J. (2021). Ast: Audio spectrogram transformer. *arXiv preprint arXiv:2104.01778*.

Gong, Y., Rouditchenko, A., Liu, A. H., Harwath, D., Karlinsky, L., Kuehne, H., and Glass, J. (2022). Contrastive audio-visual masked autoencoder. *arXiv preprint arXiv:2210.07839*.

Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851.

Kim, C. D., Kim, B., Lee, H., and Kim, G. (2019). Audiocaps: Generating captions for audios in the wild. In *NAACL-HLT*.

Kong, J., Kim, J., and Bae, J. (2020a). Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in Neural Information Processing Systems*, 33:17022–17033.

Kong, Q., Cao, Y., Iqbal, T., Wang, Y., Wang, W., and Plumbley, M. D. (2020b). Panns: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE ACM Trans. Audio Speech Lang. Process.*, 28:2880–2894.

Kong, Z., Ping, W., Huang, J., Zhao, K., and Catanzaro, B. (2021). Diffwave: A versatile diffusion model for audio synthesis.

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

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

Li, J. B., Qu, S., Li, X., Huang, P.-Y. B., and Metze, F. (2022a). On adversarial robustness of large-scale audio visual learning. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 231–235. IEEE.

Li, J. B., Qu, S., Metze, F., et al. (2022b). Audiotagging done right: 2nd comparison of deep learning methods for environmental sound classification. *arXiv preprint arXiv:2203.13448*.

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

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

Mei, X., Meng, C., Liu, H., Kong, Q., Ko, T., Zhao, C., Plumbley, M. D., Zou, Y., and Wang, W. (2023).

Wavcaps: A chatgpt-assisted weakly-labelled audio captioning dataset for audio-language multimodal research. *arXiv preprint arXiv:2303.17395*.

Nagrani, A., Yang, S., Arnab, A., Jansen, A., Schmid, C., and Sun, C. (2021). Attention bottlenecks for multimodal fusion. *Advances in Neural Information Processing Systems*, 34:14200–14213.

Park, D. S., Chan, W., Zhang, Y., Chiu, C.-C., Zoph, B., Cubuk, E. D., and Le, Q. V. (2019). Specaugment: A simple augmentation method for automatic speech recognition. In *INTERSPEECH*.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695.

Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., and Aberman, K. (2022). Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*.

Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al. (2022). Laion-5b: An open large-scale dataset for training next generation image-text models. *arXiv preprint arXiv:2210.08402*.

Series, B. (2014). Method for the subjective assessment of intermediate quality level of audio systems. *International Telecommunication Union Radiocommunication Assembly*.

Song, J., Meng, C., and Ermon, S. (2020). Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Song, Y. and Ermon, S. (2020). Generative modeling by estimating gradients of the data distribution.

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

Trabucco, B., Doherty, K., Gurinas, M., and Salakhutdinov, R. (2023). Effective data augmentation with diffusion models. *arXiv preprint arXiv:2302.07944*.

Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., and Paluri, M. (2018). A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459.

Wu*, Y., Chen*, K., Zhang*, T., Hui*, Y., Berg-Kirkpatrick, T., and Dubnov, S. (2023). 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*.

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

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. (2017). mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

Zhang, L. and Agrawala, M. (2023). Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*.

A. Appendix

Notes on Table 1 Table 2: AudioGen (Kreuk et al., 2022) employed separate data-cleaning and augmentation, we reported our estimate. All disk spaces are calculated based on the number of samples they reported in the paper. Each work reported batch-size in a different way, DiffSound(Yang et al., 2023) reported batchsize on each GPU, AudioGen did not specify, using 64 GPUs. Train steps was not reported in DiffSound, 200 epochs was reported (8 days). Ours 60k steps took 2 days. DiffSound's is also spectrogram regeneration, their spectrogram hyperparams are missing, their VQVAE was on spectrogram with latent size 5 × 53.

Appendix Outline:

1. Section A will cover more details about prompt engineering, and the usage of entailment scoring.
2. Section B will cover more details about our model training and hyperparameters
3. Section C offers a further in-depth understanding of our **cross-attention mechanism** as described in the main paper, and why it produced the results in Table 1 in the main paper.
4. Section D includes more generation results from our diffusion model.
5. Section E showcases our diffusion model could be further used as a data-augmentation method. Classifiers trained with diffusion-generated data improve over those trained only on the original data.

A. Harnessing LLMs to generate Audio+Visual Captions: Prompt Engineering

Prompting given audio-only weak labels: We leverage the power of LLMs to increment the descriptiveness of the audio captions on datasets such as AudioSet(Gemmeke et al., 2017), which only contains weak labels without descriptive captions. We use Alpaca (Taori et al., 2023) 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. Table A1 shows specific examples of the class list to caption transformation.

Alpaca is an open-source instruction-following model fine-tuned on the Llama-7b model (Taori et al., 2023). Utilizing this model generates more grammatical captions that remain faithful to the original labels.

To generate text captions from class label lists, we used the following prompt: *"For each of these, summarize the sounds into a single sentence: \n describe a situation with all of these sounds together:"* followed by the clip's 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. Table A1 covers more details.

Filter Hallucination and Obtain Visual Captions: Despite initial success with our approach, some captions contain Alpaca hallucinations, particularly in the cases of a single class caption. For example, in Table A1 line 4, *"swamp"* is a hallucination, however plausible. To address this, we filter captions to replace single-class captions with a simpler non-LLM derived caption *"The sound of [CLASS]"*. This second pass does not invalidate the Alpaca-generated captions, which are far superior at capturing the complexity of audio samples with multiple classes. Recognizing the lack of detail in this single class captions, we utilized a SOTA Video-to-text model BLIP2 (Li et al., 2023) to generate video-based captions for each video. These captions were derived from 3 sampled frames within the video at the $\frac{1}{3}$, $\frac{1}{2}$, and $\frac{3}{4}$ points of the 10 second clips. We again used Alpaca to combine these three captions into one with the following merging prompt: *"Create one sentence that summarizes these three simply:"*, allowing us to more effectively summarize the information of each video from the frames sampled.

Merging Audio and Visual Captions: These video captions were then combined with the audio captions from single-class labels with Alpaca to provide the needed visual context within each caption and combat the hallucination generated with single-label Alpaca audio captions. In our prompt, we specifically focused on the audio-label while using the visual caption as auxiliary information: *"Summarize these two captions conditioned on the second caption, the second caption describes an audio class and is the main concept:"*. For all the prompts, we provided examples for Alpaca to better utilize the strength of in-context learning. Table A3,A4 show more examples.

Audio-visual False Positive and Entailment Scoring: By observing the audio-visual captions generated while being aware of the audio-visual false-positive issue, we noticed that some of the captions in which the audio and video were not aligned resulted in noisy captions with details unrelated to audio in the actual clip. To measure the alignment between the visual caption and audio labels and test the capabilities of a decoder-only language model, we created an entailment score Alpaca prompt: *"on a scale from 0 to 1, output the probability that these two captions happen together in float format:"* In our in-context examples, we emphasized similar concepts even if the exact audio label was not referred to, such as an "ukelele" in audio and

Classes	**LLM Generated Captions**
‘Singing’, ‘Yodeling’, ‘Speech’	‘A person singing and yodeling while talking.’
‘Pump (liquid)’, ‘Water’	‘The sound of a pump dispensing liquid and running water.’
‘Dog’, ‘Growling’, ‘Animal’	‘A dog growling and making animal sounds.’
’Frog’	’A frog is croaking in a dark, musty swamp.’

Table A1. Examples of conversions between class list and free text captions made to resemble image captions generated with an engineered LLM prompt.

a “mandolin” in video receiving a high entailment score. Nonetheless, we maintain the integrity of the video caption by ensuring it did not introduce sounds not present in the audio label, such as the implication of speech through the depiction of a person, when the audio label did not include speech.

We sought to investigate the fidelity of our AV entailment score and overall usage of decoder-only LMs to calculate textual similarity and other metrics. In parallel, we use T5-sentence encoder (Raffel et al., 2020) and the BERT sentence encoder and computed the embedding similarities between the audio and video captions. The Pearson correlation was then computed between the Alpaca entailment score and the scores from both T5 and BERT encoders to get 0.13 for T5 and 0.14 for BERT. We also calculated the (min, median, max) score for each metric with Alpaca having $(0, 0.55, 1)$, T5 having $(0.60, 0.75, 0.98)$, and BERT having $(-0.20, 0.34, 0.98)$.

These scores are not correlated which implies that a decoder LM may be less reliable than an encoder LM when determining textual similarity and calculating metrics. Despite this, using a caption similarity metric could ensure that future audio-visual training data is less noisy and has clearer relationships between different modalities and be used to guide the merging of audio-visual captions to determine when the visual context is useful to include in our caption.

Evaluation of Caption Generation: AudioSet’s label coarseness and class imbalance are mitigated by our application of multiple Alpaca LLM caption generations, yielding 2.2M detailed audio clip captions. [2] Previous efforts on creating audio captions focused on automatic audio captioning(Mei et al., 2023), but with the small training set size and imbalanced classes within the dataset § B.1, the performance of their model is also limited. These are also typically end-to-end with WavCaps(Mei et al., 2023) using a HTSAT-BART model (Mei et al., 2023) which lacks the explainability and scalability compared to our human language-focused approach with weak labeling. Our approach better considers different modalities and introduces more flexibility in label generation due to the controlled hallucination that Alpaca has.

To assess the performance of our captions and the improvements additional context provided in their generation we analyzed a subset of AudioCaps (Kim et al., 2019) 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(Kim et al., 2019) while referencing the actual audio clip as shown in Table A2. Since most automatic metrics are based on n-gram similarity or LCS [3] and generally do not perform as well on individual sentence comparisons (Awad et al., 2022), we decided to use a human metric because of the large vocabulary variation as shown in the subset vocabulary size shown in Table A2. Additionally, the WavCaps (Mei et al., 2023) model is fine-tuned on AudioCaps which helps boost their automatic metric scores in comparison to our approach.

These results clearly display the qualitative improvements gained from in-context prompting for LLMs with clear examples to guide text generation as well as the addition of visual elements into the captions. One limitation is the absence of labels in the original AudioSet which the human judges mentioned in the AudioCaps captions. Typically this occurred with speech and wind sounds being present in AudioCaps but not AudioSet labels. However, these cases show the benefits of using LLMs for generation which was able to use context clues and natural language understanding to add these missing features. An example of this context-aware enrichment can be seen in Figure 1 where the caption “shuffling cards” is correctly extended to include a human in the scene. Similarly, we observed that Alpaca would hallucinate “at the park” or similar setting-specific details for audio samples weakly labeled with ducks, water, or speech. Such hallucinations are often correct, but even when false they encode relevant domain-knowledge that helps improve the quality of our captions.

A.1. Prompting given audio-only weak labels:

Table A3 shows the few-shot examples we feed Alpaca model with to generate audio captions based on the labels given.

[2] We will provide more details in the appendix and will release all captions as open-source resources.

[3] Longest common subsequence (LCS) and comparison of n-gram overlaps perform poorly with vastly different vocabularies and description styles as it is not able to capture semantic similarity well or match with completely different caption structures and scenarios

Prompt Context	Similarity Score	BLEU$_1$	BLEU$_4$	METEOR	ROUGE$_l$	CIDEr	Vocabulary Size
Zero-Shot	0.474	0.128	0.006	0.079	0.128	0.094	-
One-Shot	0.605	0.178	0.014	0.100	0.182	0.193	-
Few-Shot	0.686	0.188	0.016	0.168	0.229	0.242	-
Audio-Visual Merge	**0.750**	0.165	0.013	0.109	0.178	0.183	**1480**
WavCaps(Mei et al., 2023)	0.667	**0.231**	**0.056**	**0.136**	**0.277**	**0.682**	509
AudioCaps(Kim et al., 2019)	N/A	N/A	N/A	N/A	N/A	N/A	871

Table A2. 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.

Prompt	Response
alarm, burp, inside, small room.	burping while an alarm plays inside a small room.
dog, bark, howl, speech.	a dog barking and howling with a person speaking as well.
Music, jazz, piano, singing, speaking.	a person plays jazz piano with a singer while people talk.
engine, vehicle, wind, music, speech.	people talking inside a car while driving and listening to music.
water, gargle, inside, small room.	air is passing through the water in their mouth in a small room with water.
scratch, hammer, metal.	hammer striking a metal surface and scratching sounds can be heard.
thunder, wind, bark, small room.	a dog is barking in a small room during a thunderstorm with audible wind.
gunshot, vehicle engine, siren, crash.	a car chase with gunfire and sirens where a vehicle crashes.
waterfall, wind, sizzle, crackle.	a fire is cracking with something sizzling near a waterfall with wind.
stream, cough, cat, Purr.	a cat purrs near a coughing person while a stream can be heard.

Table A3. Few-shot examples of Audio-only labels to captions. The list of audio labels is preceded by the prompt: *"For each of these, summarize the sounds into a single sentence: \n describe a situation with all of these sounds together:"*

A.2. Filter Hallucination and Obtain Visual Captions:

We noticed that many of the single-label audio captions often had hallucinations where there would be extra details added (usually from one of the examples given). One example of this would be *"A person is sprinting while a dog is barking and howling."* when only the *"run"* label was given.

To address this, we created visual captions to help enrich our audio data. Table A4 shows the specific examples we used to join together the three visual captions that were generated using BLIP2 (Li et al., 2023).

A.3. Merging Audio and Visual Captions:

Table A5 shows the specific examples that were used in the prompt to join our single-label audio class and summarized video captions. These examples help us filter the hallucination generated when creating captions from audio-only weak labels by utilizing visual information.

A.4. Audio-visual Entailment Score:

These scores can be used to address the audio-visual false positive issue in which the audio of the video does not match the visual frames presented. These scores also increase human understanding of image-audio-text correlations. We can further utilize these scores to help perform caption filtering, gating based on the entailment score between the audio and visual concepts demonstrated within each video.

A.4.1. Entailment scoring with Alpaca, T5, and BERT:

We conducted entailment scoring using Alpaca(Taori et al., 2023), T5(Raffel et al., 2020), and BERT(Devlin et al., 2018) on the balanced set of AudioSet(Gemmeke et al., 2017). This allowed us to compare the results of the three methods and how they would consider entailment differently. Since Alpaca(Taori et al., 2023) is a decoder-only model, we noticed that the entailment scoring was not always reliable, scoring differently on additional runs. This is due to its autoregressive nature, which is highly dependent on its context. To stabilize the scoring, we employ few-shot examples (in

Prompt	Response
a video game with a dragon in the water final fantasy xv - the end of an era final fantasy x-2 - the end of an era.	The end of an era awaits in Final Fantasy XV and X-2.
a person is using a printer to print out a document a white cash register with a keyboard and a keypad a person is holding a small piece of plastic on a plane.	a person using a piece of plastic to make a purchase at a cash register.
a man with a beard sitting on a bed a man in a cowboy hat is playing an acoustic guitar a man standing in the desert holding an acoustic guitar.	A man in a cowboy hat is playing a guitar and strumming away in the desert.

Table A4. Few-shot merging visual caption examples. Each set of captions is preceded by this prompt: *"Create one sentence that summarizes these three simply:"*

Table A6, Table A7) to better utilize the in-context learning ability of Alpaca. When conducting entailment scoring for audios that only had one label, we utilized the AudioSet ontology description (Gemmeke et al., 2017) and label due to the hallucination in the default audio captions.

Despite using these examples, we still noticed that Alpaca would sometimes output “incorrect” scores, e.g. given the following pairing: (*“A background of traditional Indian music with lyrics from a bhangra song playing in the foreground.”*, *“A bright star is twinkling in the night sky, shining amidst the dark velvety backdrop.”*) a score of 0.9 or (*“A person dribbling a basketball, slamming it on the ground, and speaking.”*, *“A man in a purple shirt is playing basketball, a man in a red shirt is playing soccer.”*) a score of 0.1. These cases are very counter-intuitive for human to explain.

On the other hand, we noticed that encoder-decoder models (we only use the encoder) such as T5 (Raffel et al., 2020) and BERT(Devlin et al., 2018) scoring tended to score some single-label (ontology-based) captions lower than an alpaca-generated caption that had less which audio-visual correspondence. For example, when the caption pairing was (*“An ice cream truck outside a small room, playing music.”*, *“An old digital audio box sits proudly on a table, displaying its unique blue and white design.”*), T5 gave it a score of 0.81 whereas when the pairing was (*“Fire : Sounds resulting from the rapid oxidation of a material in the exothermic chemical process of combustion, releasing heat, light, and various reaction products.”*, *“Firefighters are putting out a blazing fire in a building.”*), T5 only gave it a score of 0.80 despite the two captions being much closer in meaning.

Although there were some discrepancies when analyzing these scores, we still found that overall, T5 scoring seemed to be more consistent with the actual content of the captions. The different score distributions for the three metrics utilized on the balanced set can be found here (Figure 3(a), Figure 3(b), Figure 3(c)).

These distributions reinforce how T5 had a generally higher score range whereas BERT had scores in a lower range (even below 0) as compared to Alpaca. It also shows how T5 and BERT had similarly shaped distributions whereas Alpaca tended to spread a bit more evenly across the spectrum (with peaks at different points in the score distribution). The result of the 3 different types of scoring on the balanced set can be found on our open-source page[4]. We have also utilized T5 scoring on the unbalanced set for AudioSet which will also be made available.

A.4.2. Effect of Audio-visual Entailment Scores

To test the effectiveness of our entailment score, we employ a similar audio-visual classification pipeline as described in (Li et al., 2022a), which is illustrated in Figure 4. Our audio encdoer is the same AST/DeiT model(Gong et al., 2021) used in the main paper. The video is encoded by a pre-trained R2+1D(Tran et al., 2018) model. The naive fusion is pure concatenation. We plug in our entailment scores in the attention mechanism by discounting the attention score for the video portion. Specifically, $\mathbf{c}_t = \sum_{i=1}^{T} \alpha_{t,i}\mathbf{h}_i$ the corresponding embedding indexes' attention score gets discounted by the entailment score. The result is shown in Table A8. Although our best score still lags the best SoTA model on AudioSet, the point is our improvement over naive fusion shows all three types of entailment scores outperform the naive version of fusion.

[4] `https://audiojourney.github.io/`

Prompt	Response
man in a cowboy hat is throwing a rope while standing on a green field. Whip	An audio scene emphasizing the sharp sound of a whip, set against a visual backdrop of a man in a cowboy hat throwing a whip on a green field.
a man eating a cupcake at a table Tick	An audio scene centered around the subtle ticking sound, with a backdrop of a man enjoying a cupcake at a table.
a person using a piece of plastic to make a purchase. Cash register	An audio scene capturing the distinctive sound of a cash register, accompanying the moment when a person uses a piece of plastic to make a purchase.
a man brushing his teeth Toothbrush	a man cleans his teeth with a toothbrush's audible scrubbing.
man standing in the desert holding an acoustic guitar Country	An audio scene focused on the Country genre, featuring a man standing in the desert holding an acoustic guitar.

Table A5. Few-shot audio and visual caption merge examples. Caption-label pair is preceded by the following prompt: *"Summarize these two captions conditioned on the second caption, the second caption describes an audio class and is the main concept:"*

A.5. More Details about Caption Generation:

Similarity Score (Table A2) We recruited 5 human subjects to evaluate 500 samples and report their average ratings for zero-shot, one-shot, and few-shot and A-V Merge captions and WavCaps (Mei et al., 2023) generated results in Table 2 of the main paper. When evaluating our captions, we ask the human subject to provide a similarity score between the generated captions and the AudioCaps(Kim et al., 2019) ground truth, ranging from 0-1. In the cases of ambiguous samples or when AudioCaps(Kim et al., 2019) labels are not reliable, we also provide the original youtube links to the evaluators and ask them to use the *audio content* as the ground truth.

Key Statistics Comparison: To further compare the different caption sets, we analyzed the vocabularies of AudioCaps (Kim et al., 2019) versus the WavCaps (Mei et al., 2023) and captions our system generated from the same clips. By observing the top 20 words (Figure 7, Figure 5, Figure 6), we can see that AudioCaps and WavCaps have many similarities with 14 of the top 20 words being identical. These similarities help support the closeness of vocabulary between the two sets (resulting in high automatic metrics), meanwhile reducing the diversity and generalizability of the captions. We also noticed that in the top 20 words of our audio-video merged captions contain the term 'audio scene' which reflects the examples we used when prompting Alpaca. This shows a clear path that we could inject our own inductive bias into any future LLM-based dataset augmentations. In addition to the vocabulary size difference, we also noticed a key difference in the length of captions generated through our system versus those of AudioCaps or WavCaps. The minimum caption length for all three main types of captions were all 3. The maximum caption differed a lot, with the AudioCaps (Kim et al., 2019) samples having a maximum of 31, WavCaps (Mei et al., 2023) samples having a maximum of 25, and our audio-video merged captions having a maximum 45. Additionally, the average length of AudioCaps is 10.49, for WavCaps is 6.89, and for our audio-video merge was 17.07. This demonstrates a drastic increase in length and thus richness of the captions generated by our LLM-based audio-video merging methods. We see that the WavCaps (Mei et al., 2023) captions are generally the shortest which suggests a lack of variation in structure when generating captions.

Some key examples of where utilizing audio-video merged captions benefited our model over WavCaps(Mei et al., 2023) are included in Table A9. We see that in these examples, our captions utilized details such as the flag or the caption on the video to determine what exactly the sound was, adding key details that weren't apparent in the WavCaps captions. These also demonstrate how simple WavCaps captions are overall.

Overall, we believe this is a very promising paradigm to augment the existing audio dataset. In the future, if we could leverage high-performance classifiers to auto-label audio from the wild or to fine-tune the LLM to plug into Automatic Audio Captioning system, it would be a clear pathway to scale up audio training datasets.

Prompt	Response
a man in a cowboy hat is throwing a rope while standing on a green field. Whip : The sound of whipping, i.e., the greatly accelerated motion of the tip of a flexible structure, as the result of concentrated angular momentum.	0.85
a man eating a cupcake at a table Tick : A metallic tapping sound.: A metallic tapping sound.	0.00
a person using a piece of plastic to make a purchase. Cash register : Sounds of a mechanical or electronic device for registering and calculating transactions, usually attached to a drawer for storing cash.	0.45
a man brushing his teeth Toothbrush : Sound of an instrument used to clean the teeth and gums consisting of a head of tightly clustered bristles mounted on a handle.	1.00
man standing in the desert holding an acoustic guitar Country : A genre of United States popular music with origins in folk, Blues and Western music, often consisting of ballads and dance tunes with generally simple forms and harmonies accompanied by mostly string instruments such as banjos, electric and acoustic guitars, dobros, and fiddles as well as harmonicas.	0.90

Table A6. Alpaca audio-visual entailment score examples for single-label. The caption-label pair is preceded by the following prompt: *"on a scale from 0 to 1, output the probability that the first caption describes a scenario with the second caption's sound description:"*

B. Training Details

B.1. Datasets

Datasets: AudioSet contains 2 million 10-second YouTube clips, each weakly annotated for 527 types of audio events. Multiple events can occur in the same clip; a video of water boiling might be labeled with both "Liquid" and "Boiling." The data contains three splits: a class-balanced training subset (22K clips), an unbalanced training subset (2M clips), and an evaluation set (20k clips). The size disparity between training subsets highlights the underlying imbalance: there are over one million clips each labeled with "Music" or "Speech," but the rarest class ("Toothbrush") has only 127 clips. AudioSet uses a hierarchical ontology[5] to categorize sounds; for example, "Toothbrush" is fully categorized as ("Sounds of things" → "Domestic sounds, home sounds" → "Toothbrush"). Despite the complexity of this hierarchy, many clips have only a single label that fails to capture the full complexity of the video's context. For example, a video of a toothbrush might be labeled simply with "Toothbrush" while containing the sounds of running water or speech.

We downloaded around 1.97M unbalanced training, 20K balanced training, and 19K evaluation clips. Some samples have been deleted from YouTube and could not be downloaded. For the AS-2M experiments in Table A11, we use the union of unbalanced and balanced sets for pretraining and fine-tuning. For the AS-20K experiments, we use AS-2M for pretraining and the 20K balanced set for fine-tuning. We report the testing mAP on the 19K eval set, and the same recipe as (Li et al., 2022b).

B.2. Implementation

We fine-tuned our models from models available on HuggingFace Diffuser Library, specifically their v1.4 model (Rombach et al., 2022). While we initialized our model from the Stable Diffusion checkpoint, the model, training code, and larger pipeline have been heavily modified to fit our purposes. Most notably, we had to make changes to work with non-HuggingFace models, such as the Encodec model(Défossez et al., 2022), along with changes to the U-Net to adapt to wide-channel inputs. All data for models other than ours in Table B.2 was copied from their respective papers training details as they do not provide *training* code as of the time of writing, only AudioLDM(Liu et al., 2023) has public code for loading checkpoints.

For our training, we exclusively trained on individual machines with eight A100 GPUs. We chose most hyperparameters following (Rombach et al., 2022), with variations to fit our hardware and model. We could use significantly larger batch sizes due to the extremely high compression from the Encodec (Défossez et al., 2022) model's discrete codebook; we lowered our memory overhead by 56.5% per sample. Pre-computing the codebook codes made these gains possible by allowing larger batches and faster training. As Table 4 of the main paper described, we trained multiple models, varying the text encoder with all other parameters and stages remaining the same. However, as will be described in Section C, the T5 models required an additional linear projection layer from the length 1028 T5 encoding to

[5] https://research.google.com/audioset/ontology/index.html

Prompt	Response
two young men wearing red and white shirts A person is speaking while there is a loud gush of air.	0.10
goat grazing on grass in the mountains A goat making music and someone speaking aswell.	0.75
a young boy is riding a skateboard down a sidewalk A male is singing and a child is singing along to the same music.	0.05
a person is holding a gun with a glove on it Gunfire and cap gun.	0.80
a screenshot of a game with three people in red and blue outfits Gunfire and cap gun.	0.00
a person playing a ukulele with their hands There is background music with a mandolin being played.	0.70
a man in a green shirt is talking to the camera A cat meowing and a person speaking.	0.50
a person holding a snake in front of a door A snake hissing.	0.60

Table A7. Alpaca audio-visual entailment score examples for multi-label. The caption pair is preceded by the following prompt: *"on a scale from 0 to 1, output the probability that the first caption describes a scenario with the second caption's sound description:"*

Model	Backbone	PT	AS-20k (mAP)			AS-2M (mAP)		
			A	V	A+V	A	V	A+V
Naive Fusion	DeiT-B/R2+1D	IN+KI-SL	34.6±.20	18.1±.09	37.4±.18	45.4±.70	23.9±.12	46.5±.29
Alpaca score Fusion	DeiT-B/R2+1D	IN+KI-SL	34.6±.20	18.1±.09	38.4±.14	45.4±.70	23.9±.12	47.6±.33
T5 score Fusion	DeiT-B/R2+1D	IN+KI-SL	34.6±.20	18.1±.09	**39.1**±.10	45.4±.70	23.9±.12	**49.5**±.42
BERT score Fusion	DeiT-B/R2+1D	IN+KI-SL	34.6±.20	18.1±.09	38.7±.15	45.4±.70	23.9±.12	49.1±.37
AST (Gong et al., 2021)	DeiT-B	IN	34.6	-	-	45.4	-	-
MBT (Nagrani et al., 2021)	ViT-B	IN-SL	31.3	27.7	43.9	41.5	31.3	49.6
CAV-MAE (Gong et al., 2022)	ViT-B	SSL	37.7	19.8	42.0	46.6	26.2	51.2

Table A8. **Comparison with other state-of-the-art models** on audio-visual classification evaluated on AudioSet(Gemmeke et al., 2017) test set, using both audio and visual features. Metrics are mAP for AS. For pre-training (PT) dataset, AS:AudioSet, KI:Kinetics (for R2+1D(Tran et al., 2018)), and IN:ImageNet. SSL: self-supervised learning, SL: supervised learning; We gray-out baselines. Best single models in AS-2M are compared (no ensembles).

the expected 786 input dimension for cross-attention. This extra layer adds parameters to the model and would affect training, but the added parameters are negligible compared to the size of the U-Net. Due to our A100 GPUs are the 40GB version, without model parallel, we could not afford to unfreeze and fine-tune the text encoder together with training the UNet.

B.3. Encodec Latent Embedding Space

Figure 8 displays the Encodec (Défossez et al., 2022) latents at each denoising stage. Perceptually the reconstruction is excellent with a subjective evaluation (MUSHRA score) of 88.0[6]. As we could see, Diffusion is able to learn the pattern of the latent space despite it being very subtle if not imperceptible by human's standard.

B.4. Training Instability from Frozen Blocks

One warning from the (Rombach et al., 2022) paper for fine-tuning their models is catastrophic forgetting. While we faced this issue when training our CLAP models, it did not noticeably affect the training of our T5 models. Our U-Net consists of a few major components: a conv_in block, down blocks, a mid-block, up blocks, and a conv_out block. During experimentation, we attempted to accelerate training

[6] `https://en.wikipedia.org/wiki/MUSHRA`

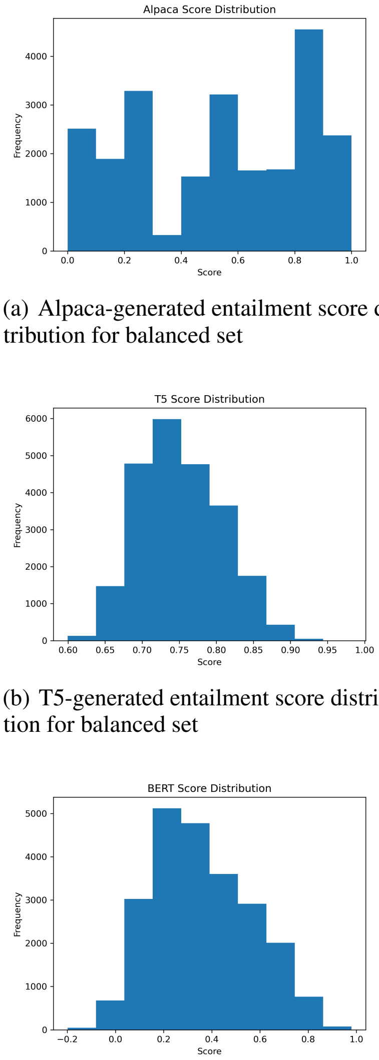
Youtube ID	Audio Label	Video Caption	Merged Caption	WavCaps
BjWf0keANT8	Sine Wave	10,000 hertz sine wave audio frequency.	10,000 hertz audio frequency, represented by a sine wave.	a continuous sine wave sound
83IJft_3Z4E	Throbbing	An ultrasound image of a baby in the womb, proof of new life and potential new beginnings.	An audio scene depicting the throbbing sound of a heartbeat, accompanying the visual of an ultrasound image of a baby in the womb, proof of new life and potential new beginnings.	a heartbeat is being recorded
6hj2F5xvGYE	Male singing	A man is singing into a microphone in front of an American flag.	A male singing into a microphone in front of an American flag, capturing the emotion of the patriotic song.	someone is singing a song

Table A9. Examples of our Audio-Video merged captions versus WavCaps(Mei et al., 2023)

	Diffusion (Denoising Network)				Classification	
Configuration	DiffSound	AudioGen	AudioLDM	Ours	AS-20K	AS-2M
Optimizer	AdamW	Adam	-	AdamW	AdamW	AdamW
Optimizer β_1 - β_2	0.9 - 0.94	-	-	0.9 - 0.999	0.9 - 0.999	
Base learning rate	3.0e-6	5.0e-4	1.0e-4	2.56e-4	0.001	2e-4
LR schedule	Constant	Inv Sqrt	Constant	CosDecay	CosDecay	CosDecay
Noise schedule	Sc-Linear	-	Sc-Linear	Cosine	-	-
ChannelMultiplier	-	1,2,4,8	1,2,3,5	1,2,4,4	-	-
Diffusion Steps	-	-	1K	1K	-	-
Warm-up epochs	-	-	-	-	1	4
Training Epochs	600	-	-	-	60	10
Warm-up steps	-	3K	-	1K	1K	1K
Training Steps	-	200K	1.5M	40K	-	-
Batch size	16	256	8	192	256	32
GPUs	32	128	1	8	4	4
GPU Type	V100	A100	A100	A100	V100	V100
SpecAug	-	-	-	-	192/48	192/48
Mixup	-	-	-	-	0.5	0.5
Loss Function	-	ℓ_1,ℓ_2,CE	MSE	MSE	BCE	BCE
Sampler	DDIM	-	DDIM	PNDM	-	-
Sample Steps	100	-	200	45	-	-
Guidance Scale	-	1 - 5	4.5	3.5 - 7.5	-	-
Normalization	-	-	-	Channel	(-4.27, 4.57)	(-4.27, 4.57)

Table A10. Table comparing training hyperparameters between SOTA audio generation models, our model, and our classification models. All "-" values are either unknown or not applicable to the given model. All values are from respective papers and appendices sections on training. Inv Sqrt = Inversed Square root; Sc-Linear = Scaled Linear; CosDecay = Cosine with Decay. For normalization we include a "-" if the values are unknown, channel for our per-channel normalization, or (μ, σ) for the dataset. Citations: DiffSound (Yang et al., 2023), AudioGen (Kreuk et al., 2022), AudioLDM (Liu et al., 2023), SpecAug (Park et al., 2019), Mixup (Zhang et al., 2017), DDIM (Song et al., 2020), PNDM (Liu et al., 2022)

and conserve the pre-trained weights of *Stable Diffusion-1-4* by freezing the weights of all blocks other than conv_in and conv_out for 5,000 training steps. This method proved inferior to simply allowing the entire pipeline to learn together in final loss value and training stability. Figure 10 shows an example training loss graph for one of the aforementioned experiments, clearly displaying the high degree of instability that emerged after the blocks were unfrozen. This instability alone would not be a reason to abandon this technique; however, the more troubling trend was the loss values plateauing around 0.4 compared to 0.19 in our best models.



(a) Alpaca-generated entailment score distribution for balanced set

(b) T5-generated entailment score distribution for balanced set

(c) BERT-generated entailment score distribution for balanced set

Figure 3. Comparison of entailment score distributions for different models on AudioSet(Gemmeke et al., 2017) balanced set.

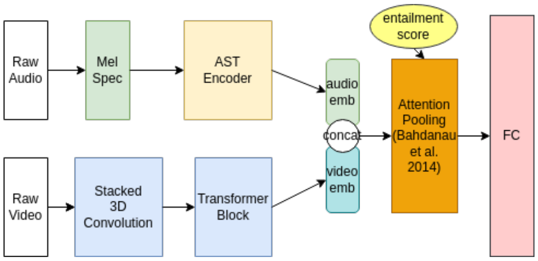


Figure 4. Entailment score penalizing attention score at the attention pooling layer

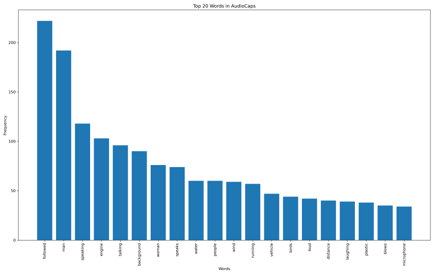


Figure 5. Top 20 words from the vocabulary of the AudioCaps samples (not including stop words)

C. Cross Attention Mechanism

In Table 1 of our main paper, we observed our *AudioJourney-CLAP* models generally under perform *AudioJourney-T5* models, we believe there could be 2 main reasons:
1) CLAP(Wu* et al., 2023) was trained on 660k samples, which is way smaller dataset than what T5(Raffel et al., 2020) was trained on. Although the CLAP text encoder demonstrated good performance on audio embedding (Wu* et al., 2023), T5 may be superior in a general setting.
2) Since CLAP (Wu* et al., 2023) output matches with d, we did not adapt its last layer. Using a frozen encoder would solely depend on the W_q, W_k, W_v to learn the mapping, which might be suboptimal.

$$
\begin{aligned}
&\text{Cross-Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\right) \cdot \mathbf{V} \\
&\text{where } \mathbf{Q} = \mathbf{W}_Q^{(i)} \cdot \varphi_i(\mathbf{z}_i),\ \mathbf{K} = \mathbf{W}_K^{(i)} \cdot \tau_\theta(y), \\
&\mathbf{V} = \mathbf{W}_V^{(i)} \cdot \tau_\theta(y) \\
&\text{and } \mathbf{W}_Q^{(i)} \in \mathbb{R}^{d_q \times d_\epsilon^i},\ \mathbf{W}_K^{(i)} \in \mathbb{R}^{d_k \times d_{\tau'}}, \mathbf{W}_V^{(i)} \in \mathbb{R}^{d_v \times d_{\tau'}}, \\
&\varphi_i(\mathbf{z}_i) \in \mathbb{R}^{n \times d_\epsilon^i},\ \tau_\theta(y) \in \mathbb{R}^{m \times d_\tau} \\
&\text{in our case: } d_\tau = 1024, d_{\tau'} = 768, d_k = d_v = d_q = d = 768 \\
&\text{(initialized from Stable Diffusion (Rombach et al., 2022) weights),} \\
&d_\epsilon^i \text{ is the } i^{th} \text{layer of Unet } \varphi_i\text{'s output size}
\end{aligned}
\tag{1}
$$

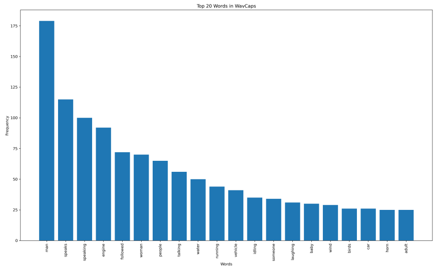


Figure 6. Top 20 words from the vocabulary of the WavCaps samples (not including stop words)

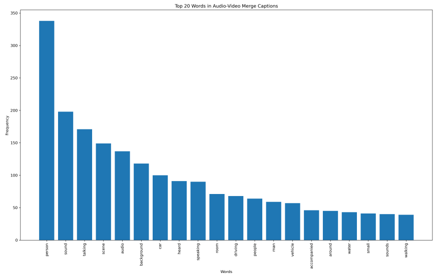


Figure 7. Top 20 words from the vocabulary of the our audio-video merged samples (not including stop words)

A surprising result in Table 4 of our main paper is that the masked 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. Figure 11 explains how masking negatively affects cross-attention in the U-Net. As is illustrated, the white part of the text embedding $\tau_\theta(y)$ indicates the masked out content because the max sequence length is larger than the number of audio caption tokens. Towards the end, you can see if we pass this mask to the U-Net, this would result in a low-rank dot-product of the attention score A and the V tensor, 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.

D. Additional Samples and Demos

For overall listening experience, we put our listening samples and spectrogram visualizations to our website: `https://audiojourney.github.io/` and the code and implementations are at: `https://github.com/audiojourney/audiojourney.github.io`

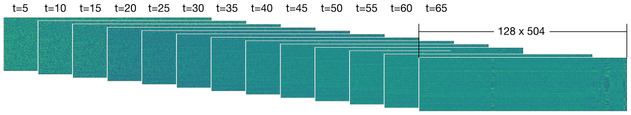


Figure 8. Latent space representations created throughout the denoising process. These images notably display the lack of interpretability in our generation space.

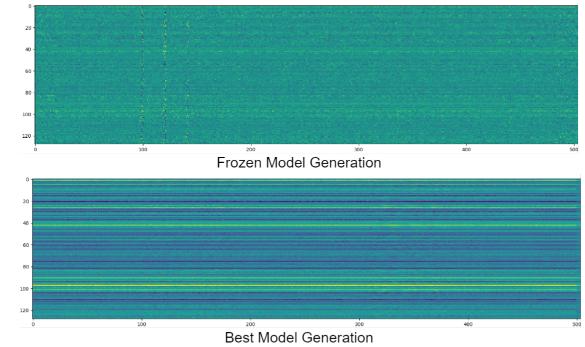


Figure 9. Generation comparison between model trained with frozen blocks (top) and model trained without (bottom). Comparing this to other examples, such as Figure 2 clearly shows the lack of quality from frozen models.

Our Audio-journey models could serve as the base model similar to Stabel Diffusion (Rombach et al., 2022) in vision, and it would allow a separate smaller network to learn new concepts from new dataset (ControlNet) (Zhang and Agrawala, 2023), and would also allow finetuning through low-rank approximation like Dreambooth (Ruiz et al., 2022).

E. Diffusion as Augmentation

To further validate the quality of both our generated captions and the trained diffusion model, we generate a large dataset of new audio samples and use it to train a classifier from scratch. Furthermore, we can also use our generated data to *supplement*, rather than replace, the existing AudioSet training data; we show that the combination of real and generated data results in improved SOTA classification accuracy.

Using a pretrained text-to-audio diffusion model, we generated over 80,000 new audio samples randomly divided among the 527 audio classes in the AudioSet-20K balanced set. A random value $N \in [1, 2, 3]$ was selected and mapped to N random classes from the AudioSet-20K class list for each sample. With these samples generated, we combined them into the datasets shown in Table A11.

Due to the unbalanced distribution of AudioSet labels, we adopted the balanced sampling strategy as mentioned in Chapter (Li et al., 2022b). Note here, the weights for the

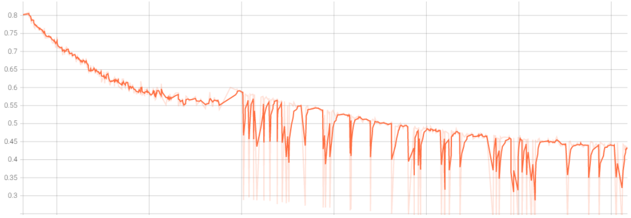


Figure 10. Training loss graph for model. This model started with all layers other than conv_in and conv_out frozen, then unfroze these blocks after 5,000 training steps.

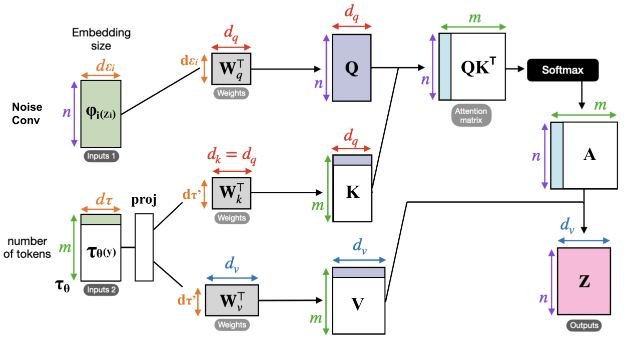


Figure 11. Illustration of the cross-attention mechanism under masking scenario, the white-colored portions indicate masking.

lowest performing 100 classes are amplified as a control set for the experiment. When mixing real and synthetic data, we adopt the same strategy as (Trabucco et al., 2023):

$$i \sim \mathcal{U}(\{1, \dots, N\}), \quad j \sim \mathcal{U}(\{1, \dots, M\})$$

$$B_{l+1} \leftarrow B_l \cup \begin{cases} X_i & \text{with probability } (1-\alpha) \\ \tilde{X}_{ij} & \text{otherwise} \end{cases} \tag{1}$$

Note here $X \in \mathbb{R}^{N \times H \times W \times 1}$ denotes a dataset of N real audio latents, and $i \in \mathbb{Z}$ specifies the index of a particular latent X_i. For each latent, we generate M augmentations, resulting in a synthetic dataset $\tilde{X} \in \mathbb{R}^{N \times M \times H \times W \times 1}$ with $N \times M$ augmentations, where $\tilde{X}_{ij} \in \mathbb{R}^{H \times W \times 3}$ enumerates the jth augmentation for the ith latent in the dataset. Indices i and j are sampled uniformly from the available N real audio latents and their M augmented versions respectively. Given indices ij, with probability $(1-\alpha)$ a real audio latent X_i is added to the batch B, otherwise its augmented latent $\tilde{X}_{ij}$ is added.

To closely match the original AS-20K dataset, these new samples do not have LLM-generated captions, the naive approach is to generate them with a simple prompt formatted as "The sound of [LIST OF AUDIO CLASSES]." This configuration was used in (Liu et al., 2023) to convert labels to captions and provide prompts. We believe it is suboptimal for data augmentation, since everything needs to start generate from random Gaussian noise, and it is trading class accuracy off for diversity. If the class is not accurate, diversity in this case will be in vain. Therefore, we use Textual Inversion (Gal et al., 2022; Trabucco et al., 2023) to update the token of the class name of each of the AudioSet classes, and fine-tune their embeddings. At the generation step, we use the same prompted "The sound of [LIST OF AUDIO CLASSES]." but perform image-to-image diffusion, so that our denoising U-Net does not have to denoise all the way from pure Gaussian noise, but from a noisy version of ground-truth latent.

As shown in Table A11, we can see the classification models trained on augmented audio datasets improve with the growing size of the dataset. These results display the value of additional diffusion-generated samples as a form of data augmentation.

Classification Accuracy: Table A11 lists classifier performance when trained on our diffusion-augmented datasets showing a clear image that additional samples generated with diffusion measurably improve classifier accuracy. Due to the large number of parameters of AST, it struggles to train from scratch, diffusion augmentation visibly alleviated this training difficulty and complemented pretraining. The most significant improvement comes from augmenting AudioSet-20K with an extra 20K generated samples, and the benefits slowly attenuate with more examples. However, it is important to note that diffusion cannot entirely replace ground truth data, as demonstrated by the inferior scores for AS-20kG. Nonetheless, it does yield measurable improvements when used as augmentation, especially when used in conjunction with other augmentation methods.

	Model	PT	Aug	AS-20kG	AS-20k	+20kG	+40kG	+60kG	+80kG	AS-2M
PANNs (Kong et al., 2020b)	CNN	-	-	-	22.1	-	-	-	-	37.5
PANNs (Kong et al., 2020b)	CNN	-	mx+sp	-	27.8	-	-	-	-	43.1
TALtrans (Li et al., 2022b)	CNN+T	-	-	-	22.4	-	-	-	-	38.3
TALtrans (Li et al., 2022b)	CNN+T	-	mx+sp	-	28.0	-	-	-	-	43.7
Our TALtrans	CNN+T	-	-	10.1±.50	22.4±.16	25.8±.11	27.0±.13	28.1±.06	30.1±.03	38.3±.15
Our TALtrans	CNN+T	-	mx+sp	11.2±.40	28.0±.20	29.4±.31	29.5±.13	30.7±.21	32.3±.14	43.7±.25
AST (Gong et al., 2021)	DeiT	-	-	-	14.8	-	-	-	-	36.6
AST (Gong et al., 2021)	DeiT	IM	mx+sp	-	34.7	-	-	-	-	45.9
Our AST	DeiT	-	-	3.2±.20	14.8±.17	15.4±.22	16.7±.14	18.1±.01	20.2±.18	34.6±.20
Our AST	DeiT	-	mx+sp	8.2±.61	16.9±.12	18.4±.32	19.5±.12	20.7±.22	22.4±.31	37.6±.10
Our AST	DeiT	IM	mx+sp	13.5±.50	34.7±.77	35.1±.18	36.1±.42	36.9±.03	37.5±.12	45.4±.70

Table A11. **Comparison with other state-of-the-art models** on audio and speech classification tasks. All datasets, other than AS-20k and AS-2M, are based from AS-20k with diffusion augmentation to add samples, details in section § 4. Metric is mean average precision (mAP). For pretraining (PT) dataset, AS:AudioSet, and IM:ImageNet. For augmentation (aug), mx+sp:mixup(Zhang et al., 2017) and SpecAug(Park et al., 2019). Generation model and classification accuracy for each augmented dataset showing the improvements measured from diffusion as augmentation. Dataset AS-20kG consists exclusively of generated samples with 0 real samples from AudioSet. Our TALtrans Model (CNN+T: CNN+Transformer) has 12.1M params, and our AST model (DeiT/ViT-B) has 88M params.