MusicSem: A Semantically Rich Language-Audio Dataset of Organic Musical Discourse

Rebecca Salganik[1], **Teng Tu**[2], **Fei-Yueh Chen**[1], **Xiaohao Liu**[2], **Kaifeng Lu**[1], **Ethan Luvisia**[1]
Zhiyao Duan[1], **Guillaume Salha-Galvan**[3], **Anson Kahng**[1], **Yunshan Ma**[4], **Jian Kang**[1]
[1]University of Rochester, [2]National University of Singapore
[3]Kibo Ryoku, [4]Singapore Management University
rsalgani@ur.rochester.edu

Abstract

Music understanding underpins a wide range of downstream tasks in music information cross modal retrieval and cross modal generation. While recent advances in multimodal learning have enabled the alignment of language and audio, progress remains limited by the lack of datasets that reflect the rich, human-centered semantics through which listeners describe music. In this work, we formalize the concept of musical semantics—encompassing emotion, context, and personal meaning—and propose a taxonomy that distinguishes between five types of music captions. We identify critical gaps in existing datasets and argue for the need to capture more authentic, nuanced musical discourse. To address this, we present a novel dataset MusicSem, with over 35K human-annotated language-audio pairs derived from organic music discussions. Our dataset emphasizes subjective semantics, including emotional resonance, contextual use, and co-listening patterns. We further conduct comprehensive evaluation of state-of-the-art retrieval and generation models, highlighting the importance of semantic sensitivity and our dataset in advancing multimodal music understanding.

1 Introduction

Music understanding, or music representation learning [1, 2], underpins most (if not all) downstream music tasks, including categorization [3, 4, 5], generation [6, 7], and recommendation [8, 9] of musical content. While past research efforts mainly focused on audio-centric approaches [10, 11, 4, 5] recent advances in multi-modal learning, particularly the alignment of textual annotations and audio, have opened new avenues in tasks of cross modal retrieval [9, 12, 13, 14] and cross modal generation, such as music-to-text generation [15, 16, 17] and text-to-music generation [18, 19, 20, 21].

At the heart of these developments there lies a critical need for high-quality semantically-rich language-audio datasets. Recently several canonical datasets [18, 22] have been used to advance generative and retrieval music tasks, enabling greater control over generation [18, 23, 19] and richer contextualization of audio representations [12, 9] via textual conditioning. Despite the increasing reliance on language-audio data, there has been little rigorous examination of what this language should entail or what kinds of information it should meaningfully convey. In particular, there is often an interpretation gap, in which generative models are unable to interpret the a user's expressed intent in their generated output [24, 25]. Furthermore, while professional musicians use descriptive language when engaging with music, laypeople often rely on abstract semantic content when engaging with music [26, 27]. Thus, the commonly used musician-annotated datasets for music understanding [18] could be *too well-curated*, and there is an urgent need for textual annotations that capture that user-centered nuances of music semantics and encompass a broader form of musical discourse.

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

More concretely, musical semantics encompass several nuanced expressions of a musical work. We categorize these expressions into five categories: (1) descriptive, relating to concrete musical attributes, (2) atmospheric, relating to the emotions or aesthetic vibe elicited by a song, (3) situational, relating to the situation in which a song is listened to, (4) contextual, relating to collections of songs which contextualize a user's listening intent, and (5) metadata-based, relating to information found in tags or background research (e.g. chart performance, artist background, release dates). Figure 1 presents an illustrative example of these categories. However, most existing datasets are lacking an adequate representation of such attributes, particularly those related to atmosphere, context, or situational use. This homogenization is further exacerbated by the emergence of large language model (LLM)-generated datasets. Due to limited public data, many recent efforts have relied on LLMs to enrich existing music descriptions or augment metadata [28, 29, 30, 15]. However, these datasets often replicate the limitations of their source data (e.g., musician annotations, tags being augmented), failing to incorporate the rich semantics that define authentic human-music dialogue.

Additionally, the lack of standardized evaluation in generative and retrieval tasks hinders the ability to assess the extent to which a model understands the nuances in musical discourse [24, 31]. For instance, in text-to-music generation [18, 19, 21], a user's prompt provides critical context to his/her generation intent and should be used when evaluating a model's effectiveness. Thus, without a principled framework for capturing and evaluating these expressions, it is impossible to assess a model's sensitivity to the dimensions of meaning that matter most to listeners.

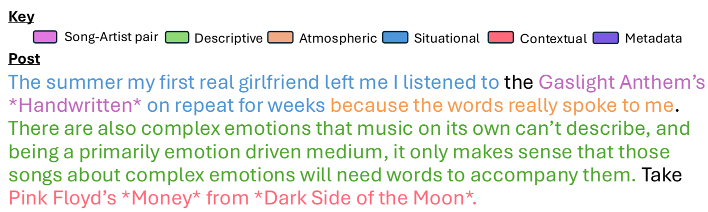

Figure 1: Semantic content in musical descriptions. We show an example of Reddit-based musical post from MusicSem, highlighting the five categories of annotation elements.

In this work, we present a semantically rich language-audio dataset, named MusicSem, to address the scarcity of personalized, humanistic, semantic data. Our empirical analysis shows that existing models are not sensitive to several categories of music semantics due to lack of semantic awareness in the training data. Following this, we construct MusicSem that captures the idiosyncratic, colloquial, and semantically diverse language commonly found in organic musical discourse. Drawn from a large corpus of musical discussions on Reddit, our dataset consists of over 35,977 language-audio pairs. Crucially, the language in our dataset explicitly encompasses not just descriptive attributes of the music itself, but also the emotions it evokes, situational and contextual use cases, and co-listening patterns (i.e., songs frequently listened to together). Third, we perform extensive evaluation of state-of-the-art (SOTA) cross modal retrieval and generation models on both our MusicSem and recently widely-used datasets. We highlight a series of key insights, including performance inconsistency, open challenges to SOTA methods, more importantly, the gap in semantics sensitivity, where this study makes the initial endeavor and our MusicSem dataset demonstrates great potential towards addressing them. In summary, the contributions of this paper are as follows:

1. **Music semantics.** We categorize human expressions of musical works into five major categories and show that existing models lack the awareness of such music semantics.
2. **Data curation.** We construct a semantically rich language-audio dataset that includes over 35K language-audio pairs and captures the music semantics. We further release an automated pipeline for extracting semantic captions associated with music samples for dataset extension.
3. **Comprehensive evaluation and insights**: We conduct a thorough evaluation of prominent SOTA models on both cross modal retrieval and generation tasks. Our analysis yields key insights that can guide and inspire future research in this area.

2 Related Work

We briefly review datasets that are complementary to our work. A more comprehensive review of related works is available in Appendix B and in a survey by Christodoulou, Lartillot, and Jensenius [32].

Depending on the source of textual data, current language-audio music datasets can be categorized as human-annotated datasets or LLM-augmented datasets. For human-annotated language-audio music datasets, *MusicCaps* [18] is one of the most commonly used dataset. It consists of approximately 5,521 language-audio samples annotated by professional musicians. These annotations contain descriptive language that often involves attributes such as instrumentation, genre, and stylistic analysis. Similarly, *YouTube8M-MusicTextClips* [33] contains approximately 4,169 language-audio pairs, but the associated captions are written by text-for-hire annotators. More recently, Manco et al. [22] presented the *Song Describer* [22] extended 1,100 of audio samples in *Jamendo* [11] with crowd-sourced annotations. Meanwhile, there are also datasets with LLM-augmented annotations [29, 28, 17, 16, 30], which, though although they have a larger scale, lack precise description on how music is experienced in the real world [34, 35, 36]. Different from these datasets that primarily capture the acoustic elements of a song, our work seeks to understand how a song makes a user feel and the contexts in which users listen to it.

There also exist other music datasets based on Reddit threads [37, 38]. However, they are intended for different settings from ours. For example, *Tip-Of-My-Tongue* [37] is based on `r/TipOfMyTongue` for text-to-music querying. Alternatively, Veselovsky, Waller, and Anderson [38] scrape Reddit for $536,860$ unique song-artist pairs to analyze the music sharing behaviors in Reddit communities.

3 Music Semantics

One of the goals in language-audio music understanding tasks is the design of models which are able to capture the nuances that contextualize a listening experience. We organize these contextual elements into five major categories, which we term *music semantics* [35, 39, 34]. Then, we highlight the importance of *music semantics* in language-audio datasets by quantifying the semantic sensitivity in a wide range of generative and retrieval models.

Table 1: Categorization of different caption elements.

Category	Description	Example
Descriptive	concrete musical attributes	"I like the high pass filter on the vocals in the chorus, really makes harmonies pop"
Contextual	other songs	"Sabrina Carpenter's *Espresso* is just a mix of old Ariana Grande and 2018 Dua Lipa"
Situational	an activity or environment	"I listened to this song on the way to quitting my sh**ty corporate job"
Atmospheric	emotions and expressive adjectives	"This song makes me feel like a manic pixie dream girl in a bougie coffeeshop"
Metadata	technical & background information	"This deluxe edition of this song was released in 2013 and it has three bonus hiphop tracks"

Categorization of music semantics. Consider the following two prompts: *"This song is a ballad. It contains guitar, male vocals, and a piano. It sounds like something I would listen to at church"* or *"This song is a ballad. It contains guitar, male vocals, and a piano. It sounds like something I would listen to while tripping on acid"*. While their descriptions of musical attributes (e.g., ballad, guitar, male vocals, piano) remain the same, the change in the situational context (listen to at church vs. while tripping on acid) should drastically change our expectations for the associated audio in generative and retrieval settings. To this end, we present a comprehensive formal categorization of music semantics, including (1) descriptive elements to describe the musical attributes of a song, (2) contextual elements that highlight other songs that are similar to a song or might be co-listened together, (3) situational elements to describe an activity or environment in which a song is listened to, (4) atmospheric that express the emotions a song evokes or other expressive adjective of a song, and (5) metadata that provides technical and background information of a song and/or its corresponding artist. An example for each category is presented in Table 1.

Insensitivity to varying semantic context. Here we quantify the sensitivity of multimodal music understanding models to such varying contexts. Given any i-th language-audio pair (t_i, a_i) in a language-audio dataset, we construct a counterfactual annotation $\tilde{t}_i^c$ by changing descriptions with respect to a semantic category c, e.g., while at church vs. while tripping on acid in the aforementioned example. We randomly sampled 50 language-audio pairs in *MusicCaps* and create a counterfactual example with respect to each semantic category present in each language-audio pairs.[1] We release the full set of counterfactual examples created from the MusicCaps [18] at `https://tinyurl.com/bddrn8pr`. Then, for generative models, we quantify its sensitivity as

$$G^c = \frac{1}{n}\left[\sum_{i=1}^{n} 1 - \text{cosine}(f_i, \tilde{f}_i^c)\right], \tag{1}$$

[1] Note that a textual annotation includes at least one semantic category, but may not include all five categories.

Table 2: Semantic sensitivity analysis in text-to-music generative models. Best performance is highlighted in **bold**, second best in underline. The superscripts ^{d}, ^{a}, ^{s}, ^{m}, ^{c} refer to descriptive, atmospheric, situational, metadata, and contextual, respectively.

Model	G^d	G^a	G^s	G^m	G^c
AudioLDM2	0.68	0.37	0.35	0.40	0.34
MusicLM	0.50	0.36	0.42	0.39	0.35
Mustango	0.62	0.27	0.25	0.26	0.32
MusicGen	0.57	0.47	0.39	0.47	0.52
Stable Audio	**0.72**	**0.67**	**0.68**	**0.70**	**0.74**
Mureka[2]	-	-	-	-	-

Table 3: Semantic sensitivity analysis on cross modal retrieval models. Best performance is highlighted in **bold**, second best in underline. The superscripts ^{d}, ^{a}, ^{s}, ^{m}, ^{c} refer to descriptive, atmospheric, situational, metadata, and contextual, respectively. We set K=10.

Model	R^d	R^a	R^s	R^m	R^c
LARP	**0.98**	0.17	0.06	0.0	**0.56**
CLAP	0.95	0.52	0.35	0.42	0.52
ImageBind	0.84	0.39	0.35	0.38	0.41
CLaMP3	0.92	**0.58**	**0.49**	**0.62**	0.55

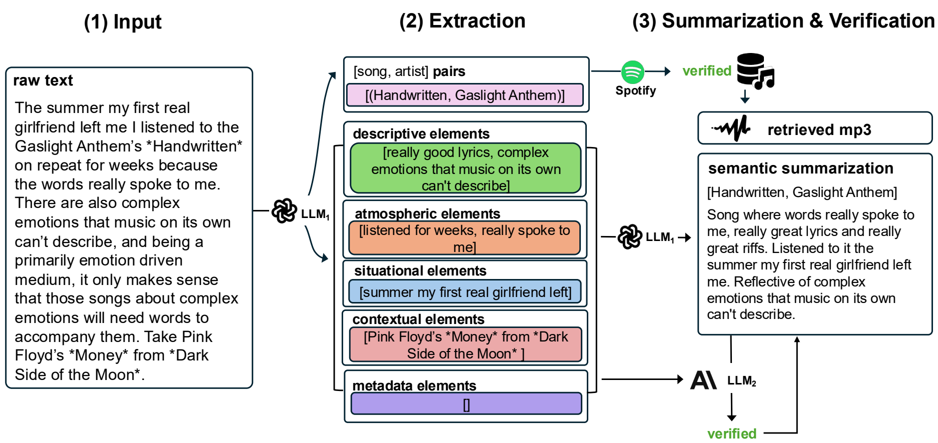

Figure 2: Visualization of the extraction and verification pipeline for dataset construction.

where n is the number of language-audio pairs, $f_i = \mathcal{M}(t_i)$ and $\tilde{f}_i^c = \mathcal{M}(\tilde{t}_i)$ are the outputs of a text-to-music generative model $\mathcal{M}$. We denote $f_i = a_i = \mathcal{M}(t_i)$ and $\tilde{f}_i = \tilde{a}_i = \mathcal{M}(\tilde{t}_i)$ in text-to-music generation. However, f_i and $\tilde{f}_i$ could also be the any music representation depending on the downstream tasks. Similarly, for retrieval models, we quantify its sensitivity with respect to top-k retrieved audio candidates as

$$R@k = \frac{1}{n}\left[\sum_{i=1}^{n} 1 - \frac{|A_i \cap \tilde{A}_i|}{|A_i|}\right], \tag{2}$$

where $A_i = \mathcal{M}(t_i)$ and $\tilde{A}_i = \mathcal{M}(\tilde{t}_i)$ are the retrieved top-k audio candidates.

Table 2 and 3 show the sensitivity of a wide range of SOTA text-to-music generative and retrieval models. From the tables, we observe that these models maintain a substantially higher sensitivity to changes in descriptive elements compared to atmospheric, situational, contextual, or metadata change. These results highlight the lack of semantic awareness in the textual conditioning of a music understanding model, which manifests a misalignment between the audio candidates expected by a user and the model output.

4 MusicSem: A Language-Audio Dataset Embracing Music Semantics

To address the lack of representation towards nuanced music semantics in existing datasets for training and evaluating music understanding models, we present MusicSem, a semantically rich dataset with

[2]We are unable to finish all experiments for Mureka due to a change in the Mureka API on May 9th, 2025, which made it no longer possible to automatically load from their website.

language-audio pairs from online musical discourse. Our data sources include five large Reddit threads that cover different genres with rich user discussions about music, including `r/electronicmusic`, `r/popheads`, `r/progrockmusic`, `r/musicsuggestions`, and `r/LetsTalkMusic`. These threads are selected using the PushShift API.[3] In addition, we also release an entire collection of 68 threads by searching the keyword "music" using the PushShift API. After selecting the source Reddit threads, the curation processing is followed by an (S1) extraction step to extract semantic contents from the textual elements in the selected Reddit threads and a (S2) summarization and verification step to format extracted tags into sentence-like semantic annotations, verify the association between songs and artists, as well as the truthfulness of the extracted semantic information from the extraction step. An illustrative example of the dataset construction process is shown in Figure 2. For more details about dataset curation, we defer to Appendix C due to space limitation.

S1: Extraction. This phase aims to transform a raw textual post into a dictionary containing the (song, artist) pairs and different categories of music semantics associated with song(s) mentioned in the post. Specifically, we concatenate the title and body of a post into one single prompt and pass it into an LLM. Here we use GPT-4o (2024-08-06) [40] as the extraction model. For the prompt to extract semantics, inspired by [41], we craft the prompt with examples manually and iteratively refine it through conversation with an LLM. Prompts we used can be found in Appendix D.

Table 4: Semantic diversity in MusicSem and canonical language-audio music datasets.

Category	MusicCaps	Song Describer	MusicSem (Ours)
Descriptive	100%	94%	100%
Contextual	6%	8%	77%
Situational	41%	16%	48%
Atmospheric	57%	33%	64%
Metadata	28%	6%	64%

S2: Summarization and verification. This step aims to format the extracted semantics into sentences using LLMs similar to existing datasets [18, 22, 33] and verify the correctness of (song, artist) pairs as well as truthfulness of LLM-generated semantic annotations. To avoid wrong association between song and artist in a (song, artist) pair, we first check the overlap between the extracted information (i.e., song, artist name) and the original text body (all in lowercase). We remove a (song, artist) pair if the character-wise overlap is less than 75%. Then we query Spotify to obtain a unique ID for each pair. In cases when the query returns multiple entries, we apply the same filtration strategy in the first step. Finally, we query the audio clips using the Spotdl library and download the entire mp3 file from YouTube. If an mp3 file is unable to be found, we remove the (song, artist) pair. After obtaining the (song, artist) pairs, the associated audio files, and the semantic extractions, we use GPT-4o to rephrase the extracted semantic tags into sentence-like semantic annotations. Then we verify the truthfulness with an alternate verification LLM (Claude Sonnet 3.7 [43] specifically) to compare the extracted semantic tags in the extraction step and the LLM-rephrased semantic annotations. The verification model is prompted to generate a binary decision and remove entries listed as hallucinated. Please note that, in respecting the copyright of these songs, we will only release the unique identifiers for each song and the extraction pipeline, but not the audio files.

Table 5: Vocabulary statistics of MusicSem and other canonical datasets. Average number of tokens (# Tokens) of all text annotations in a dataset are calculated using BERT [42].

Statistics	MusicCaps	Song Describer	MusicSem (Ours)
# Entries	5,521	1,100	35,977
# Vocab. Words	6,245	2,824	23,208
# Tokens	59.36	23.88	80.54
# Genres	267	152	493

Training/test splits. After extraction and verification of the Reddit sources, our entire dataset consists of 35,977 language-audio pairs. To further facilitate meaningful evaluation, we select a human-validated test set of 480 entries.[4] The test set will remain unpublished for developing a leaderboard in the future. The remaining entries are collected as the training set which can be accessed at `https://huggingface.co/datasets/Rsalga/MusicSem`.

[3] `https://github.com/pushshift/api`

[4] The test set is made available to the reviewers at `https://tinyurl.com/3n8je74z`. We will remove its access upon publication.

Table 6: Evaluation results on the text-to-music retrieval task. R represents Recall, and N represents NDCG. Best performance for each metric within a dataset is in **bold** and second best in underline.

Dataset	Model	R@1 ↑	R@5 ↑	R@10 ↑	N@5 ↑	N@10 ↑	MRR ↑
MusicCaps	Random	0.04	0.18	0.36	0.10	0.16	0.31
	LARP	0.14	0.49	0.98	0.30	0.45	0.62
	CLAP	**5.84**	**15.57**	**22.60**	**10.73**	**12.99**	**11.60**
	ImageBind	3.15	10.18	14.91	6.72	8.25	7.23
	CLaMP3	2.73	8.82	13.65	5.81	7.32	9.07
Song Describer	Random	0.14	0.71	1.41	0.41	0.64	1.01
	LARP	0.36	1.72	2.62	1.05	1.29	1.61
	CLAP	4.61	17.3	27.67	11.20	14.54	12.41
	ImageBind	4.43	13.02	20.71	8.72	11.16	9.84
	CLaMP3	**10.49**	**27.31**	**38.61**	**19.21**	**22.84**	**19.83**
MusicSem (Ours)	Random	0.21	1.05	2.11	0.62	0.96	1.42
	LARP	0.22	1.02	3.07	0.54	1.22	1.47
	CLAP	0.82	5.74	9.84	3.54	4.74	4.65
	ImageBind	2.05	5.94	11.07	3.83	5.48	5.24
	CLaMP3	**7.79**	**18.85**	**26.84**	**13.65**	**16.21**	**14.68**

Semantic diversity. We analyze the semantic diversity in MusicSem. Table 4 shows the proportion of data points that contain each semantic category in MusicSem and other canonical datasets. It is clear that MusicSem is semantically rich with higher proportion for all categories. MusicSem is also semantically rich in terms of vocabulary size, with 2x more unique words and genres compared to MusicCaps and Song Describer as shown in Table 5.

Implications of MusicSem. In addition to semantic diversity, we highlight two other key elements of MusicSem: (1) the presence of personalization and (2) contextualization of songs. First, for personalization, each song is discussed in 2.98 posts on average, which could yield varying opinions on the same song. Such broader scope of perspectives in MusicSem could offer the opportunity to develop models with personalized understanding of each musical piece. Besides, each post in MusicSem contains an average of 10.51 songs mentioned in tandem. These songs could be semantically aligned to each other in a unified theme (e.g., positivity). This form of contextualization shows a need to create association between songs in music understanding.

5 Evaluation on Cross Modal Retrieval

To demonstrate the utility and superiority of our dataset, we evaluate representative multimodal music understanding models on cross modal retrieval [13, 12], which is one of the major tasks where multimodal learning plays a pivotal role. In particular, we focus on evaluating text-to-audio retrieval in text inputs are treated as queries to retrieve corresponding musical works. We test four models, including LARP [9], CLAP [12], ImageBind [14], and ClaMP3 [13], as well as a native baseline Random. More details of the implementation of these methods can refer to Appendix H.2. In addition to our dataset, we also evaluate these models in two recent widely-used datasets, i.e., MusicCap [16] and Song Describer [22]. We employ the standard evaluation metrics for retrieval, i.e., Recall@K, NDCG@K, and MRR (Mean Reciprocal Rank), where $K \in \{1, 5, 10\}$. Table 6 details the evaluation results, based on which we derive the following insights.

Insight 1.1: Different datasets yield different rankings over the best model. We can see that for the MusicCaps dataset CLAP shows the best performance. Meanwhile, for Song Describer and MusicSem CLaMP3 is the highest performing model. We believe that this this can be attributed to the underlying datasets which were used to train each model. While CLAP was originally trained on a mixture of music and ambient audio, CLaMP3 is designed specifically for music. This is aligned with the audio samples provided in each of the datasets. While MusicCaps has audio which is taken from Youtube and is overlapping with AudioSet [44] (which contains generalized ambient audio), Song Describer and MusicSem use exclusively studio-quality audio which is devoid of ambient noises. The inconsistency of the testing performance across different datasets implies that existing multimodal music understanding models have a large space to improve in terms of generalization capability.

Table 8: Results of music-to-text generation. Best performance within each dataset is in **bold**.

Dataset	Model	$B_1 \uparrow$	$B_2 \uparrow$	$B_3 \uparrow$	M $\uparrow$	R $\uparrow$	CIDEr $\uparrow$	Bert-S $\uparrow$
MusicCaps	LP-MusicCaps	**53.21**	**47.28**	**44.60**	**51.90**	3.35	**384.72**	**90.47**
	MU-LLaMA	1.35	0.55	0.22	40.22	11.27	0.09	80.47
	FUTGA	8.80	3.07	1.19	44.77	**11.90**	2.63e-17	81.67
Song Describer	LP-MusicCaps	9.51	3.07	0.94	**8.90**	10.45	1.03	**84.40**
	MU-LLaMA	**12.03**	**4.73**	**1.73**	8.72	**13.00**	**3.59**	83.51
	FUTGA	3.39	1.28	0.43	8.72	6.30	3.58e-30	82.55
MusicSem (Ours)	LP-MusicCaps	**11.57**	**3.05**	**0.72**	**20.59**	9.54	0.77	**82.13**
	MU-LLaMA	4.11	1.41	0.51	22.33	**10.57**	**0.92**	81.63
	FUTGA	4.82	1.50	0.44	22.23	7.48	0.01	80.93

Insight 1.2: MusicSem is more challenging than existing datasets. Comparing MusicSem and Song Describer, almost all of the models perform worse on MusicSem than those on Song Describer, especially considering that the candidate set of MusicSem (480) is much smaller than that of Song Describer (1K) [5]. This demonstrates that MusicSem is much harder than Song Describer for the existing multimodal music understanding models. Analogously, MusicCaps has an even larger candidate set (5K), while its performance only slightly lower than that of MusicCap, implying that MusicSem is more challenging to existing retrieval models. Furthermore, this insight probably hints that current multimodal music understanding models are still limited in semantics understanding, which could be extensively studied or even addressed using our datasets in the future.

6 Evaluation on Cross Modal Generation

In addition to cross modal retrieval, MusicSem is also well suited for cross modal generation, including music-to-text generation [22, 15, 17] and text-to-music generation [19, 20, 21].

6.1 Music-to-Text Generation

Music-to-text generation, also known as music captioning, focuses on generating natural language descriptions of a musical work. We consider three SOTA models, including LP-MusicCaps [7], MU-LLaMA [15], and FUTGA [17], and evaluate them on three datasets, i.e., MusicCaps [18], Song Describer [22] and the test set of our proposed datset, MusicSem. We employ objective evaluation metrics borrowed from natural language processing such as BLEU (B) [45], METEOR (M) [46], ROUGE (R) [47], CIDEr [48], and BERT-score (Bert-S) [42] which are commonly used in evaluation for this task. For a more in-depth discussion of the evaluation metrics and intuitions behind them, please see Appendix H.6. The results are presented in Table 8 with the following insights.

Table 7: Semantics analysis of the music-to-text generation results on MusicSem. 'G.T.' refers to 'Ground Truth'.

Model	LP-MusicCaps	MU-LLaMA	FUTGA	G.T.
Descriptive	100%	99%	100%	83%
Contextual	2%	1%	0%	17%
Situational	42%	0%	1%	38%
Atmospheric	78%	3%	91%	62%
Metadata	32%	2%	34%	15%

Insight 2.1: Model performance differs between datasets and metrics. When looking at the results for MusicCaps and MusicSem datasets we can see that LP-MusicCaps [16] has strong performance on this dataset. Meanwhile, on the Song Describer dataset, MU-LLaMA outperforms both models. This observation coincides with the performance inconsistency observed in cross modal retrieval, further justifying that existing music-to-text generation models have generalization issues. Developing highly generalizable models would be one of the key research questions for text-to-music generation.

Insight 2.2: The performance inconsistency is attributed to the diverse semantics among datasets. To further demystify the performance inconsistencies, we analyze the presence of each type of semantics both in the ground truth of the MusicSem test set and the text generated by each model in

[5] In retrieval tasks, a larger candidate set often results in lower performance.

Table 9: Overall evaluation results on text-to-music generation. Best performance for each metric within a dataset is in **bold** and second best in underline.

Dataset	Model	$\text{FAD}_{\text{V}}^{\text{MC}}$ ↓	$\text{FAD}_{\text{V}}^{\text{FMA}}$ ↓	$\text{FAD}_{\text{M}}^{\text{FMA}}$ ↓	$\text{FAD}_{\text{E}}^{\text{FMA}}$ ↓	KLD ↓	Vendi ↑	CS ↑
MusicCaps	MusicLM	5.70	21.57	87.39	249.72	1.79	1.55	0.28
	Stable Audio	6.97	**15.60**	82.21	377.02	1.90	1.31	0.31
	MusicGen	7.03	16.29	73.22	354.07	0.90	**1.57**	0.29
	AudioLDM2	3.29	19.31	60.02	202.11	**0.61**	**1.57**	**0.36**
	Mustango	**1.27**	22.96	**55.84**	**161.47**	1.51	1.48	0.27
	Mureka[7]	-	-	-	-	-	-	-
Song Describer	MusicLM	7.20	20.59	87.12	241.95	0.89	1.49	0.28
	Stable Audio	4.42	14.90	79.16	341.92	1.07	1.29	0.31
	MusicGen	2.64	14.60	65.74	354.07	0.66	**1.50**	**0.35**
	AudioLDM2	2.74	17.19	57.88	184.03	**0.62**	1.48	0.34
	Mustango	2.58	18.50	56.69	170.27	1.48	1.46	0.29
	Mureka	**2.42**	**9.85**	**35.58**	**47.84**	1.38	1.38	0.23
MusicSem (Ours)	MusicLM	7.25	22.57	86.97	248.42	1.00	1.46	0.27
	Stable Audio	5.50	14.96	79.35	342.53	1.15	1.28	**0.31**
	MusicGen	3.75	14.67	68.11	229.29	0.74	**1.50**	0.30
	AudioLDM2	3.47	17.66	57.71	181.11	**0.55**	1.46	0.28
	Mustango	5.06	19.15	55.11	157.32	1.46	1.41	0.20
	Mureka	**2.70**	**9.69**	**34.75**	**44.75**	1.40	1.33	0.18

Table 7. From this statistics, we can see that LP-MusicCaps's high performance positively correlates with its higher percentage of atmospheric, situational, and contextual annotations in our dataset. LP-MusicCaps is the model with the highest percentage of these semantic categories represented in its output. Furthermore, we can clearly see that all of the models are skewed towards presenting descriptive captions and very few are able to capture the contextual, situational, and atmospheric elements of the Reddit-based annotations. This highlights the challenge of generating accurate and meaningful semantic information using the existing SOTA models, and MusicSem can be instrumental in bridging this gap.

6.2 Text-to-Music Generation

Text-to-music generation aims to generate a musical audio given a textual input. In this work we consider one of the most challenging settings, i.e., multi-track generation, or generations containing multiple instruments [18, 29, 23, 30, 49, 19, 21] with one-shot textual prompting. [6]. We specifically select six cutting edge models, including MusicLM [18], Stable Audio [20], MusicGen [19], AudioLDM2 [21], Mustango [30], and Mureka. For evaluation, we consider multiple widely-used objective metrics that can be grouped along three dimensions: 1) Quality of generated audio, i.e., Frechet Audio Distance (FAD) [52, 53], 2) Diversity of generated audio, i.e., Kullback–Leibler Divergence (KLD) [54] and Vendi Score (VS) [55]; and 3) Fidelity of generated audio with textual input, i.e., CLAP score (CS) [12]. More details of the baselines and the evaluation metrics are in see Appendix H.3 and H.6. We present the results in Table 9 with the following insights.

Insight 3.1: Each metric tells its own story. First, different variations of FAD demonstrate different results. Given a reference model (indicated by the subscript, where V, M, E corresponds to VGG [56], MERT [57], and Encodec [58], respectively) and reference dataset (indicated by the superscript, where MC and FMA refers to MusicCaps[18] and Free Music Audio Dataset (FMA) [59], respectively), FAD measures the distance of the mean and covariance of embeddings between the real and generated audio, extracted using the reference model. Similar to the findings presented by Gui et al. [53], we see that the values calculated using VGG, MERT, and Encodec demonstrate significant differences between competing models (often by a factor of x100). Although the proprietary model Mureka has the best performance, the second best model varies significantly based on the selection of reference model and dataset. This indicates that non-proprietary models still have a large gap in producing high quality music. Second, the models with high FAD do not necessarily have high Vendi Score, implying that achieving both high quality and high diversity is still a very challenging problem in text-to-music generation. Lastly, there is also noticeable performance inconsistency between the canonical metrics used in Table 9 and the semantic sensitivity results in Table 2. For example, Stable Audio achieves high performance in semantic sensitivity test while performs poor according to the

[6]We leave multi-turn interactive generation [25, 50, 51] for future work.

canonical metrics. Meanwhile, Mustango shows the opposite trend. This highlights the complexity of maintaining semantic consistency while also satisfying the existing canonical criteria of text-to-music generation. The inclusion of semantic sensitivity as an evaluation metric poses new challenges to current methods, which are highlighted by MusicSem and its motivational materials.

Insight 3.2: Limitations of the CLAP score. The CLAP score is one of the key metrics used to objectively evaluate alignment between a textual prompt and its associated generated audio output. Surprisingly, we are unable to see any performance differences between various models on the canonical benchmark datasets and MusicSem. This is unusual because MusicSem contains significantly less descriptive annotations which should, intuitively, be reflected in the CLAP score performance. To demystify this, we leverage the semantic sensitivity metric in Eq. 1 and calculate the cosine similarity of text embeddings generated by the CLAP model, in order to assess its ability to adequately capture semantic differences in a textual prompt. However, the results in Table 10 show that CLAP, too, has a similar lack of semantic sensitivity. This strongly indicates that the CLAP score is highly limited in capturing the rich semantics of music.

Table 10: The sensitivity of CLAP score. The superscripts ^{d}, ^{a}, ^{s}, ^{c}, and ^{m} refer to descriptive, atmospheric, situational, contextual, and metadata, respectively.

Category	Metric	Score
Descriptive	G^d	0.55
Atmospheric	G^a	0.36
Situational	G^s	0.32
Contextual	G^c	0.29
Metadata	G^m	0.36

7 Limitations

MusicSem has two limitations. First, MusicSem consists of one-to-many mapping of textual annotation to audio files. In our test set, we purposefully exclude this one-to-many mapping to be comparable in existing datasets such as *MusicCaps* and *Song Describer* in evaluating music understanding models. However, we believe this contextualization of many songs within one post is more of a feature rather than a bug to enhanced personalized music understanding. Second, MusicSem is affected by selection bias because our data sources are English-oriented subreddits and the users who actively discuss music on Reddit are often more opinionated or aligned with niche communities [60, 61]. It might lack a more comprehensive cultural representation of music from non-Western cultures and general population.

Additionally, the nascent stage of proprietary music generation companies creates significant roadblocks for evaluating models at scale. For example, Mureka was the only proprietary model which was compatible with API calls (not a manual interface). But it is still challenging to set up for comprehensive evaluation. Moreover, due to the nature of our dataset, we exclude tasks like music QA [1, 6] and controllable music generation [51] and leave them for future work.

8 Conclusion

In this work, we introduce MusicSem, a semantically rich language-audio dataset that captures the diverse language in organic musical discourse. We categorize these textual annotations into five categories of music semantics and show the importance of music semantics. We evaluate a suite of music understanding models in multimodal generation and cross modal retrieval tasks on MusicSem and other canonical datasets, which reveal critical insights about pitfalls in existing evaluations of music understanding and the importance of capturing nuances in musical annotations.

MusicSem paves the way for many future opportunities. First, we plan to further expand the scale and scope of MusicSem using more threads about music. Additionally, MusicSem currently does not include any conversations about lyrics or symbolic representations of music, which could also be beneficial in music representation learning. We also consider the expansion of benchmarking evaluations for controllable music generation [51, 50] and text-guided recommendation [62, 63] using MusicSem. Finally, the insights from benchmarking existing models highlight the need for a more comprehensive collection of objective metrics for evaluating the alignment between language-audio pairs. We hope MusicSem will shed light on developing models that understand the nuanced language people commonly use when engaging with music.

References

[1] Meinard Müller. *Fundamentals of Music Processing*. 01 2015. ISBN 978-3-319-21944-8. doi: 10.1007/978-3-319-21945-5.

[2] Markus Schedl, Emilia Gómez, and Julián Urbano. 2014. doi: 10.1561/1500000042.

[3] Matthew C. McCallum, Filip Korzeniowski, Sergio Oramas, Fabien Gouyon, and Andreas F. Ehmann. Supervised and unsupervised learning of audio representations for music understanding. In *ISMIR*, pages 256–263, 2022.

[4] Sergio Oramas, Oriol Nieto, Francesco Barbieri, and Xavier Serra. Multi-label music genre classification from audio, text and images using deep features. In *ISMIR*, pages 23–30, 2017.

[5] Ruibin Yuan, Yinghao Ma, Yizhi Li, Ge Zhang, Xingran Chen, Hanzhi Yin, Le Zhuo, Yiqi Liu, Jiawen Huang, Zeyue Tian, Binyue Deng, Ningzhi Wang, Chenghua Lin, Emmanouil Benetos, Anton Ragni, Norbert Gyenge, Roger B. Dannenberg, Wenhu Chen, Gus Xia, Wei Xue, Si Liu, Shi Wang, Ruibo Liu, Yike Guo, and Jie Fu. MARBLE: music audio representation benchmark for universal evaluation. In *NeurIPS*, 2023.

[6] Joshua Patrick Gardner, Simon Durand, Daniel Stoller, and Rachel M. Bittner. Llark: A multimodal instruction-following language model for music. In *ICML*. OpenReview.net, 2024.

[7] Shansong Liu, Atin Sakkeer Hussain, Chenshuo Sun, and Ying Shan. Music understanding llama: Advancing text-to-music generation with question answering and captioning. In *ICASSP*, pages 286–290. IEEE, 2024.

[8] Aaron Van den Oord, Sander Dieleman, and Benjamin Schrauwen. Deep content-based music recommendation. *Advances in Neural Information Processing Systems*, 26, 2013.

[9] Rebecca Salganik, Xiaohao Liu, Yunshan Ma, Jian Kang, and Tat-Seng Chua. LARP: language audio relational pre-training for cold-start playlist continuation. In *KDD*, pages 2524–2535. ACM, 2024.

[10] Yu-Ching Lin, Yi-Hsuan Yang, and Homer H. Chen. Exploiting online music tags for music emotion classification. *ACM Trans. Multim. Comput. Commun. Appl.*, 7(Supplement):26, 2011.

[11] Dmitry Bogdanov, Minz Won, Philip Tovstogan, Alastair Porter, and Xavier Serra. The mtg-jamendo dataset for automatic music tagging. In *Machine Learning for Music Discovery Workshop, International Conference on Machine Learning (ICML 2019)*, 2019.

[12] Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP*, pages 1–5. IEEE, 2023.

[13] Shangda Wu, Zhancheng Guo, Ruibin Yuan, Junyan Jiang, Seungheon Doh, Gus Xia, Juhan Nam, Xiaobing Li, Feng Yu, and Maosong Sun. Clamp 3: Universal music information retrieval across unaligned modalities and unseen languages. 2025. URL `https://arxiv.org/abs/2502.10362`.

[14] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *CVPR*, 2023.

[15] Shansong Liu, Atin Sakkeer Hussain, Chenshuo Sun, and Ying Shan. Music understanding llama: Advancing text-to-music generation with question answering and captioning. 2023. URL `https://arxiv.org/abs/2308.11276`.

[16] SeungHeon Doh, Keunwoo Choi, Jongpil Lee, and Juhan Nam. Lp-musiccaps: Llm-based pseudo music captioning. 2023. URL `https://arxiv.org/abs/2307.16372`.

[17] Junda Wu, Zachary Novack, Amit Namburi, Jiaheng Dai, Hao-Wen Dong, Zhouhang Xie, Carol Chen, and Julian McAuley. FUTGA: Towards fine-grained music understanding through temporally-enhanced generative augmentation. In Anna Kruspe, Sergio Oramas, Elena V. Epure, Mohamed Sordo, Benno Weck, SeungHeon Doh, Minz Won, Ilaria Manco, and Gabriel Meseguer-Brocal, editors, *Proceedings of the 3rd Workshop on NLP for Music and Audio (NLP4MusA)*, pages 107–111, Oakland, USA, November 2024. Association for Computational Lingustics. URL `https://aclanthology.org/2024.nlp4musa-1.17/`.

[18] Andrea Agostinelli, Timo I. Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, Matt Sharifi, Neil Zeghidour, and Christian Frank. Musiclm: Generating music from text. 2023. URL `https://arxiv.org/abs/2301.11325`.

[19] Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[20] Zach Evans, CJ Carr, Josiah Taylor, Scott H. Hawley, and Jordi Pons. Fast timing-conditioned latent audio diffusion. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

[21] Haohe Liu, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Qiao Tian, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D. Plumbley. Audioldm 2: Learning holistic audio generation with self-supervised pretraining. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 32:2871–2883, May 2024. ISSN 2329-9290. doi: 10.1109/TASLP.2024.3399607. URL `https://doi.org/10.1109/TASLP.2024.3399607`.

[22] Ilaria Manco, Benno Weck, Seungheon Doh, Minz Won, Yixiao Zhang, Dmitry Bogdanov, Yusong Wu, Ke Chen, Philip Tovstogan, Emmanouil Benetos, Elio Quinton, György Fazekas, and Juhan Nam. The song describer dataset: a corpus of audio captions for music-and-language evaluation. *CoRR*, abs/2311.10057, 2023.

[23] Flavio Schneider, Ojasv Kamal, Zhijing Jin, and Bernhard Schölkopf. Moûsai: Efficient text-to-music diffusion models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8050–8068, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.437. URL `https://aclanthology.org/2024.acl-long.437/`.

[24] Yongyi Zang and Yixiao Zhang. The interpretation gap in text-to-music generation models. 2024. URL `https://arxiv.org/abs/2407.10328`.

[25] Francesca Ronchini, Luca Comanducci, Gabriele Perego, and Fabio Antonacci. Paguri: a user experience study of creative interaction with text-to-music models. 2024. URL `https://arxiv.org/abs/2407.04333`.

[26] Joyce Eastlund Gromko. Perceptual differences between expert and novice music listeners: A multidimensional scaling analysis. *Psychology of Music*, 21(1):34–47, 1993. doi: 10.1177/030573569302100103.

[27] David Bainbridge, Sally Cunningham, and J. Downie. How people describe their music information needs: A grounded theory analysis of music queries. 01 2003.

[28] Seungheon Doh, Keunwoo Choi, Jongpil Lee, and Juhan Nam. Lp-musiccaps: Llm-based pseudo music captioning. In *ISMIR*, pages 409–416, 2023.

[29] Qingqing Huang, Daniel S. Park, Tao Wang, Timo I. Denk, Andy Ly, Nanxin Chen, Zhengdong Zhang, Zhishuai Zhang, Jiahui Yu, Christian Frank, Jesse Engel, Quoc V. Le, William Chan, Zhifeng Chen, and Wei Han. Noise2music: Text-conditioned music generation with diffusion models. 2023. URL `https://arxiv.org/abs/2302.03917`.

[30] Jan Melechovsky, Zixun Guo, Deepanway Ghosal, Navonil Majumder, Dorien Herremans, and Soujanya Poria. Mustango: Toward controllable text-to-music generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8286–8309, 2024.

[31] Yannis Vasilakis, Rachel Bittner, and Johan Pauwels. Evaluation of pretrained language models on music understanding. In Anna Kruspe, Sergio Oramas, Elena V. Epure, Mohamed Sordo, Benno Weck, SeungHeon Doh, Minz Won, Ilaria Manco, and Gabriel Meseguer-Brocal, editors, *Proceedings of the 3rd Workshop on NLP for Music and Audio (NLP4MusA)*, pages 98–106, Oakland, USA, November 2024. Association for Computational Lingustics. URL `https://aclanthology.org/2024.nlp4musa-1.16/`.

[32] Anna-Maria Christodoulou, Olivier Lartillot, and Alexander Refsum Jensenius. Multimodal music datasets? challenges and future goals in music processing. *Int. J. Multim. Inf. Retr.*, 13 (3):37, 2024.

[33] Daniel McKee, Justin Salamon, Josef Sivic, and Bryan Russell. Language-guided music recommendation for video via prompt analogies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[34] Jeong Choi, Anis Khlif, and Elena Epure. Prediction of user listening contexts for music playlists. In *Proceedings of the 1st Workshop on NLP for Music and Audio (NLP4MusA)*. Association for Computational Linguistics, 2020.

[35] Mark Levy and Mark Sandler. Learning latent semantic models for music from social tags. *Journal of New Music Research*, 2008.

[36] Darius Afchar, Alessandro B. Melchiorre, Markus Schedl, Romain Hennequin, Elena V. Epure, and Manuel Moussallam. Explainability in music recommender systems. *CoRR*, abs/2201.10528, 2022.

[37] Samarth Bhargav, Anne Schuth, and Claudia Hauff. When the music stops: Tip-of-the-tongue retrieval for music. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 2506–2510, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450394086. doi: 10.1145/3539618.3592086. URL `https://doi.org/10.1145/3539618.3592086`.

[38] Veniamin Veselovsky, Isaac Waller, and Ashton Anderson. Imagine all the people: Characterizing social music sharing on reddit. In *ICWSM*, pages 739–750. AAAI Press, 2021.

[39] Juhan Nam, Keunwoo Choi, Jongpil Lee, Szu-Yu Chou, and Yi-Hsuan Yang. Deep learning for audio-based music classification and tagging: Teaching computers to distinguish rock from bach. *IEEE Signal Process. Mag.*, 36(1):41–51, 2019.

[40] OpenAI. Hello GPT-4o. `https://openai.com/index/hello-gpt-4o/`, 2024.

[41] Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438, 2020. doi: 10.1162/tacl_a_00324. URL `https://aclanthology.org/2020.tacl-1.28/`.

[42] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL `https://aclanthology.org/N19-1423/`.

[43] Anthropic. Claude 3.7 Sonnet and Claude Code. `https://www.anthropic.com/news/claude-3-7-sonnet`, 2025.

[44] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017.

[45] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL, 2002. doi: 10.3115/1073083.1073135. URL `https://aclanthology.org/P02-1040/`.

[46] Satanjeev Banerjee and Alon Lavie. METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In Jade Goldstein, Alon Lavie, Chin-Yew Lin, and Clare R. Voss, editors, *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization@ACL 2005, Ann Arbor, Michigan, USA, June 29, 2005*, pages 65–72. Association for Computational Linguistics, 2005. URL `https://aclanthology.org/W05-0909/`.

[47] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL `https://aclanthology.org/W04-1013/`.

[48] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. 2015. URL `https://arxiv.org/abs/1411.5726`.

[49] Max W. Y. Lam, Qiao Tian, Tang Li, Zongyu Yin, Siyuan Feng, Ming Tu, Yuliang Ji, Rui Xia, Mingbo Ma, Xuchen Song, Jitong Chen, Yuping Wang, and Yuxuan Wang. Efficient neural music generation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[50] Liwei Lin, Gus Xia, Yixiao Zhang, and Junyan Jiang. Arrange, inpaint, and refine: Steerable long-term music audio generation and editing via content-based controls. In Kate Larson, editor, *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 7690–7698. International Joint Conferences on Artificial Intelligence Organization, 8 2024. doi: 10.24963/ijcai.2024/851. URL `https://doi.org/10.24963/ijcai.2024/851`. AI, Arts & Creativity.

[51] Yixiao Zhang, Yukara Ikemiya, Woosung Choi, Naoki Murata, Marco A. Martínez-Ramírez, Liwei Lin, Gus Xia, Wei-Hsiang Liao, Yuki Mitsufuji, and Simon Dixon. Instruct-musicgen: Unlocking text-to-music editing for music language models via instruction tuning. 2024. URL `https://arxiv.org/abs/2405.18386`.

[52] Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi. Fréchet audio distance: A reference-free metric for evaluating music enhancement algorithms. In Gernot Kubin and Zdravko Kacic, editors, *20th Annual Conference of the International Speech Communication Association, Interspeech 2019, Graz, Austria, September 15-19, 2019*, pages 2350–2354. ISCA, 2019. doi: 10.21437/INTERSPEECH.2019-2219. URL `https://doi.org/10.21437/Interspeech.2019-2219`.

[53] Azalea Gui, Hannes Gamper, Sebastian Braun, and Dimitra Emmanouilidou. Adapting frechet audio distance for generative music evaluation. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1331–1335, 2024. doi: 10.1109/ICASSP48485.2024.10446663.

[54] Jonathon Shlens. Notes on kullback-leibler divergence and likelihood. 2014. URL `https://arxiv.org/abs/1404.2000`.

[55] Dan Friedman and Adji Bousso Dieng. The vendi score: A diversity evaluation metric for machine learning. 2023. URL `https://arxiv.org/abs/2210.02410`.

[56] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL `http://arxiv.org/abs/1409.1556`.

[57] Yizhi Li, Ruibin Yuan, Ge Zhang, Yinghao Ma, Xingran Chen, Hanzhi Yin, Chenghao Xiao, Chenghua Lin, Anton Ragni, Emmanouil Benetos, Norbert Gyenge, Roger B. Dannenberg, Ruibo Liu, Wenhu Chen, Gus Xia, Yemin Shi, Wenhao Huang, Zili Wang, Yike Guo, and Jie Fu. MERT: acoustic music understanding model with large-scale self-supervised training.

In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=w3YZ9MSlBu.

[58] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. 2022. URL https://arxiv.org/abs/2210.13438.

[59] Michaël Defferrard, Kirell Benzi, Pierre Vandergheynst, and Xavier Bresson. Fma: A dataset for music analysis. 2017. URL https://arxiv.org/abs/1612.01840.

[60] Elena Epure and Romain Hennequin. A human subject study of named entity recognition in conversational music recommendation queries. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1281–1296, 2023.

[61] Alexey N Medvedev, Renaud Lambiotte, and Jean-Charles Delvenne. The anatomy of reddit: An overview of academic research. *Dynamics on and of Complex Networks*, pages 183–204, 2017.

[62] Enrico Palumbo, Gustavo Penha, Andreas Damianou, José Luis Redondo García, Timothy Christopher Heath, Alice Wang, Hugues Bouchard, and Mounia Lalmas. Text2tracks: Prompt-based music recommendation via generative retrieval. 2025. URL https://arxiv.org/abs/2503.24193.

[63] Mathieu Delcluze, Antoine Khoury, Clémence Vast, Valerio Arnaudo, Léa Briand, Walid Bendada, and Thomas Bouabça. Text2playlist: Generating personalized playlists from text on deezer. 2025. URL https://arxiv.org/abs/2501.05894.

[64] Qingqing Huang, Daniel S. Park, Tao Wang, Timo Denk, Andy Ly, Nanxin Chen, Zhengdong Zhang, Zhishuai Zhang, Jiahui Yu, Christian Frank, Jesse Engel, Quoc V. Le, William Chan, Zhifeng Chen, and Wei Han. *Noise2Music: Text-conditioned Music Generation with Diffusion Models*, 2023. URL https://arxiv.org/abs/2302.03917.

[65] Edith Law, Kris West, Michael Mandel, Mert Bay, and J. Downie. Evaluation of algorithms using games: The case of music tagging. pages 387–392, 01 2009.

[66] Brian McFee, Thierry Bertin-Mahieux, Daniel P.W. Ellis, and Gert R.G. Lanckriet. The million song dataset challenge. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12 Companion, page 909–916, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450312301. doi: 10.1145/2187980.2188222. URL https://doi.org/10.1145/2187980.2188222.

[67] Abhinaba Roy, Renhang Liu, Tongyu Lu, and Dorien Herremans. Jamendomaxcaps: A large scale music-caption dataset with imputed metadata. 2025. URL https://arxiv.org/abs/2502.07461.

[68] John Thickstun, Zaid Harchaoui, Dean P. Foster, and Sham M. Kakade. Invariances and data augmentation for supervised music transcription. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2018.

[69] John Thickstun, Zaid Harchaoui, and Sham M. Kakade. Learning features of music from scratch. In *International Conference on Learning Representations (ICLR)*, 2017.

[70] S Sakshi, Utkarsh Tyagi, Sonal Kumar, Ashish Seth, Ramaneswaran Selvakumar, Oriol Nieto, Ramani Duraiswami, Sreyan Ghosh, and Dinesh Manocha. Mmau: A massive multi-task audio understanding and reasoning benchmark. 2024. URL https://arxiv.org/abs/2410.19168.

[71] David Hauger, Andrej Kosir, Marko Tkalčič, and Markus Schedl. The million musical tweets dataset: What we can learn from microblogs. 11 2013.

[72] Rebecca Salganik, Fernando Diaz, and Golnoosh Farnadi. Fairness through domain awareness: Mitigating popularity bias for music discovery. In *Advances in Information Retrieval: 46th European Conference on Information Retrieval, ECIR 2024, Glasgow, UK, March 24–28, 2024,*

Proceedings, Part IV, page 351–368, Berlin, Heidelberg, 2024. Springer-Verlag. ISBN 978-3-031-56065-1. doi: 10.1007/978-3-031-56066-8_27. URL https://doi.org/10.1007/978-3-031-56066-8_27.

[73] Yu-An Chung, Yu Zhang, Wei Han, Chung-Cheng Chiu, James Qin, Ruoming Pang, and Yonghui Wu. w2v-bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 244–250, 2021. doi: 10.1109/ASRU51503.2021.9688253.

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

[75] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.703. URL https://aclanthology.org/2020.acl-main.703/.

[76] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. 2022. URL https://arxiv.org/abs/2212.04356.

[77] Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley. Panns: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894, 2020.

[78] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[79] Ke Chen, Xingjian Du, Bilei Zhu, Zejun Ma, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. HTS-AT: A hierarchical token-semantic audio transformer for sound classification and detection. In *ICASSP*, pages 646–650. IEEE, 2022.

[80] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *CVPR*, pages 2818–2829. IEEE, 2023.

[81] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *ACL*, pages 8440–8451. Association for Computational Linguistics, 2020.

[82] Renrui Zhang, Jiaming Han, Chris Liu, Peng Gao, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. 2024. URL https://arxiv.org/abs/2303.16199.

[83] Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. SALMONN: Towards Generic Hearing Abilities for Large Language Models. April 2024. doi: 10.48550/arXiv.2310.13289. URL http://arxiv.org/abs/2310.13289.

[84] Qingqing Huang, Aren Jansen, Joonseok Lee, Ravi Ganti, Judith Yue Li, and Daniel P. W. Ellis. Mulan: A joint embedding of music audio and natural language. 2022. URL https://arxiv.org/abs/2208.12415.

[85] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. Soundstream: An end-to-end neural audio codec. 2021. URL `https://arxiv.org/abs/2107.03312`.

[86] Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-fidelity audio compression with improved rvqgan. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

[87] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019. URL `https://arxiv.org/abs/1907.11692`.

[88] Pengcheng He, Jianfeng Gao, and Weizhu Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, 2023. URL `https://arxiv.org/abs/2111.09543`.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In this work we state three clear contributions: (1) Formalizing a semantic taxonomy, (2) Curating a semantically rich language-audio dataset which is reflective of organic musical discourse on Reddit, and (3) a comprehensive empirical evaluation of SOTA music retrieval and generation models. In Section 3 we present our taxonomy and motivate the need for understanding the categories of musical semantics by defining a novel sensitivity metric for assessing a model's ability to reflect the nuances of musical discourse. In Section 4 we present our dataset construction pipeline and the unique attributes of our dataset which we believe contribute to its high quality information. In Sections 6 and 7 we perform extensive evaluations across two canonical datasets and our own. To the best of our ability we select models with are current, SOTA, and have publicly released their architectures and checkpoints.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.

- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Yes, we allocate an independent section for a discussion of limitations in which we concretely present 4 limitations of our methodology (2 for dataset and 2 for evaluation). Additionally we present a series of directions for future work that can improve the current dataset and evaluations.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: No theoretical results.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.

- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: First, we release the entire dataset onto Huggingface. Second, we provide extensive details listing the dataset construction pipeline (in Section 4 and Appendix). In addition, we put links to the Github repository including: (1) model implementation (including individual conda environments for easier reproducibility), (2) evaluation scripts, and (3) data construction pipeline. Finally, we also list the hyperparameters of each model in the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code base can be accessed at: `https://github.com/Rsalganik1123/MusicSem`, all hyperparameter settings are listed in the Appendix, and dataset is available at `https://huggingface.co/datasets/Rsalga/MusicSem/tree/main`.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.

- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: All information is listed in the Appendix. We provide information on the settings of the datasets and hyperparameters for the evaluated models.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Our claims are not related to the relative improvements of one particular model so we do not test for statistical significance.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.

- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer:[Yes]

   Justification: Yes, we include this information in our Appendix including the latency of running each model and a comprehensive detail of our computational resources.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We are aligned with the code of conduct.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Yes, we address this in the Appendix of our work. On the positive impact of this work, we believe that broadening the scope of musical discourse provides a deeper, more nuanced perspective on music discourse and paves the way for future innovations in generative and retrieval music models. From a negative perspective, we consider the controversial nature of generative art and the tension with artist communities. While our work does not personally exacerbate the issues of memorization, we understand that any contribution to this domain should be treated with sensitivity.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: Yes, we address this in our Appendix. Indeed, our data is scraped from Reddit. But, given the anonymous nature of this platform and the fact that we do not release any information that is not publicly available on the web, we believe that we mitigate these risks to the best of our ability.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: While our paper indeed uses audio that is published by musicians who are not credited in this work, we do not release this as part of our dataset, instead providing unique ids on Spotify which we have gathered and leave it up to other users to scrape or work exclusively with the language in our dataset.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Yes, we publish our dataset and the entire data construction pipeline used to construct it. We provide clear instructions for reproducibility and extension. Furthermore, we present the opportunity for other researchers to expand the scope of our dataset by released data which was extracted but not integrated into our final version.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: We do not do any crowdsourcing.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA] .

    Justification: We do not do any crowdsourcing.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Yes we provide clear description of our use of LLMs for data cleaning and define hallucination protocols to limit the potential for misinformation.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Leaderboard and Demos

We create a website for the publication of a future leaderboard and demonstrations. This website can be accessed at `https://music-sem-web.vercel.app/`. The home page is visualized in Figure 3. The webpage also includes a selection of the key results from this paper, visualizations of the dataset

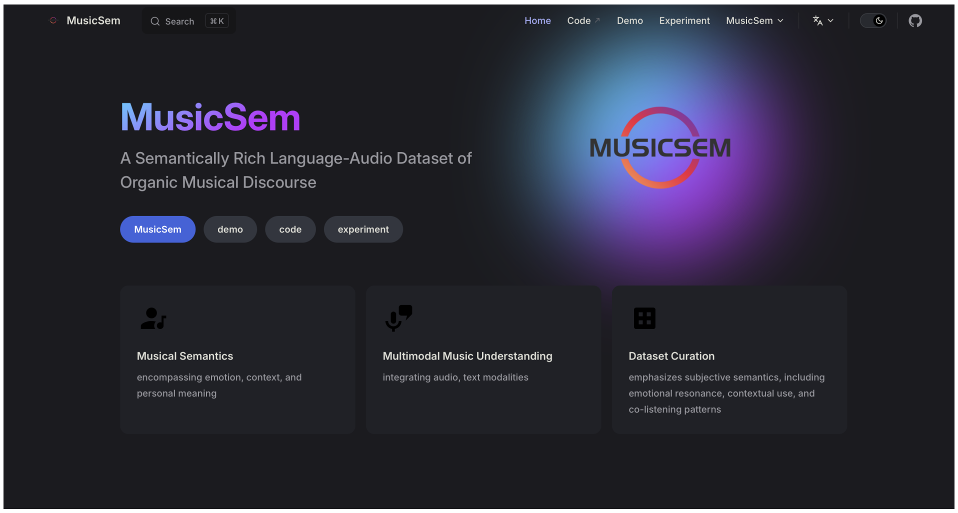

Figure 3: Website Homepage

construction protocol and a placeholder for future demos.

B More Related Works

Given the complexity of achieving true music understanding, there has been a large body of work which attempts to address different facets of this challenging task via various approaches to language-audio representation learning, each needing their own format of data. In the background section of this work we have already presented *MusicCaps* [18], *Song Describer* [22], and *Youtube-8M* [33]. Together, these form the canonical datasets which are most frequently applied to the tasks of cross modal generation and retrieval.

Recently, there has been a push to expand the scope of these datasets via LLM-based augmentation. For example in their work [64] build a combined dataset of 6.8M pairs by fusing *MusicCaps* [18] with LLM-based annotations of 150K popular songs. *LP-MusicCaps* [16] combines several sources including *MusicCaps* [18], *MagnaTagATune* [65], and *Million Song Dataset* [66] to construct 2.2M captions paired with 0.5M audio samples by generating sentence-like captions generated by an LLM. Similarly, *MusicBench* [30] is a dataset with 52K paired language-audio samples constructed by applying automatic algorithms for extracting downbeats, chords, keys, and tempo from the audio included in *MusicCaps* and augmenting its original captions to include this information. The dataset used to train FUTGA [17] follows a similar augmentation strategy in which an LLM is prompted to augment the annotations in *MusicCaps* [18] and *Song Describer* [22] to include structural elements in the music. *JamendoMaxCaps* [67] is generated by collecting 200,000 audio samples from the *Jamendo* [11] dataset and applying a music captioning model to generate automatic textual annotations. *Text2Music* [23] is another such dataset which contains 50K language-audio pairs which are compiled by scraping Spotify for the top 10 most popular playlists and using an LLM to rephrase their metadata into sentence-like structures. Notably, despite the prevalent use of LLMs in the construction of these datasets, to our knowledge, there is extremely limited discussion within this body of work on the hallucination protocols used to ensure data quality. While we acknowledge that it is impossible to fully mitigate hallucination when engaging on a large scale with LLMs, it is important to consider the effectiveness of various mitigation techniques in order to ensure data quality.

In another strain of music understanding tasks, several works have begun to consider music understanding through the lens of generative retrieval or musical question-answering [6, 15]. To serve the

Table 11: Language-Audio Music Dataset Statistics. Note, for brevity, we present the datasets that are most comparable with our setting. Here, we use L-A Pairs to mean Language-Audio Pairs and Annotation Source to indicate the source of the textual annotations.

Dataset Name	Year	# L-A Pairs	Annotation Source	Base Dataset
MusicNet 68	2018	330	Human	-
Song Describer 22	2023	1,106	Human	-
YouTube8M-MusicTextClips 33	2023	4,169	Human	
MusicCaps 18	2023	5,521	Human	-
MusicSem (Ours)	2025	35,977	Human or LLM	-
MuLaMCap 29	2023	6,800,000	LLM	AudioSet
LP-MusicCaps 28	2023	2,000,000	LLM	MusicCaps, Magnatagtune, & Million Song Dataset
Text2Music [23]	2024	50,000	LLM	Spotify
FUTGA 17	2024	51, 800	LLM	MusicCaps & Song Describer
MusicBench 30	2024	53,168	LLM	MusicCaps
JamendoMaxCaps 67	2025	200,000	LLM	Jamendo

needs of this novel task, several works have proposed datasets that reformat the textual information described above as question-answer pairs. For example, *MusicQA* [15] uses a LLM to reformulate the captions in *MusicCaps* [18] and *Magnatagtune* [65] into 4,500 question-answer pairs. Alternatively *LLaRK* [6] propose a dataset with over 1.2M language-audio pairs by combining *MusicCaps* [18], *YouTube8M* [33], *MusicNet* [69], *FMA* [59], Jamendo [11], and *MagnaTagATune* [65]. Finally Sakshi et al. [70] curate 10K a set of generalized audio and music question-answer pairs which assess a variety of music understanding tasks.

Finally, in a complementary body of musical datasets, several works have analyzed music understanding through the lens of online discourse. In addition to the datasets mentioned in the main body of our work [37, 38] which contained discourse from Reddit, the *Million Tweet Dataset* [71] analyzed over 1M tweets associated with music to understand the trends in popularity among songs and artists.

C Further Details of Dataset Construction Pipeline

We present the pseudocode for the complete extraction pipeline in Algorithm 1.

Algorithm 1 Collection Framework

Input: thread name T, language models $\mathcal{M}_1$, $\mathcal{M}_2$
Output caption set C

1: **procedure** DATASET GENERATION($T, \mathcal{M}$)
2: posts = Load_Entire_Thread(T)
3: filtered = Length_and_Mod_Filter(posts)
4: sa_pairs, caption_extracts = $\mathcal{M}_1$(filtered)
5: descriptive, atmospheric, situational, contextual, metadata = caption_extracts
6: song_ids = Spotify_Metadata(sa_pairs)
7: sa_pairs = Hallucination_Check1(sa_pairs,fltrd)
8: mp3s = Spotify_Audio(song_ids)
9: final_summaries = Summarize(sa_pairs,caption_extracts, mp3s)
10: filtered_captions = Hallucination_Check2(caption_extracts, final_captions, $\mathcal{M}_2$)

In Lines 2-3 we filter the posts within the thread itself, removing any posts that were written by moderators and any posts that had less than 20 characters. In Line 4 we perform the extraction, using an LLM to extract semantic information from the text using a prompt (see Appendix D.1 for the full prompt). In Line 6 we query the Spotify API to find a unique identifier associated with each song mentioned in a thread. In Line 7 we perform the first hallucination check, ensuring that the audio is aligned with the extracted song-artist pair. In Line 8 we extract the mp3 files associated with the audio of each song. In Line 9 we generate summaries from the extraction caption categories that mimic those of *MusicCaps* [18] or Song Describer [22]. Finally, in Line 10 we perform one more hallucination check using a different model to ensure that the summary did not deviate from the extracted caption categories (see Appendix D.2 for the full prompt). In total, this process yields a dataset of approximately 35K language-audio pairs. For a visualization of the entire pipeline, please see Figure 2 within the main body of the paper.

D Prompts

D.1 Extraction Prompt

Below we present the prompt which is used to extract semantic content from raw text posts on Reddit. Following the formulation of caption categories in Table 1, we break down the elements which are contained in each of the five categories. We also provide an example extraction for guidance.

```
% Feature Extraction

Task Description
You are tasked with analyzing Reddit posts about music and extracting structured information into specific categories. When given a Reddit post discussing music, identify and extract the following:
Categories to Extract
Song/Artist pairs
(using the names of artists and their songs with unfixed form) some examples:

'Shake it Off by Taylor Swift'
'Radiohead's Weird Fishes'
'Genesis - Yes'
'Maroon5 [She Will Be Loved]'

Descriptive (using musical attributes)
This includes detailed observations about:

Instrumentation: 'I love the high pass filter on the vocals in the chorus and the soft piano in the bridge'
Production techniques: 'The way they layered those harmonies in the second verse is incredible'
Song structure: 'That unexpected key change before the final chorus gives me goosebumps'
Sound qualities: 'The fuzzy lo-fi beats with that vinyl crackle in the background'
Technical elements: 'The 6/8 time signature makes it feel like its swaying'

Contextual (using other songs/artists)
This includes meaningful comparisons such as:

Direct comparisons: 'Sabrina Carpenter's Espresso is just a mix of old Ariana Grande and 2018 Dua Lipa'
Influences: 'You can tell they've been listening to a lot of Talking Heads'
Genre evolution: 'It's like 90s trip-hop got updated with modern trap elements'
Sound-alikes: 'If you like this, you should check out similar artists like...'
Musical lineage: 'They're carrying the torch that Prince lit in the 80s'

Situational (using an activity, setting, or environment)
This includes relatable scenarios like:

Life events: 'I listened to this song on the way to quitting my sh**ty corporate job'
Regular activities: 'This is my go-to album for late night coding sessions'
Specific locations: 'Hits different when you're driving through the mountains at sunset'
```

```
Social contexts: 'We always play this at our weekend gatherings and everyone vibes to it'
Seasonal connections: 'This has been my summer anthem for three years running'

Atmospheric (using emotions and descriptive adjectives)
This includes evocative descriptions such as:

Emotional impacts: 'This song makes me feel like a manic pixie dream girl in a bougie coffeeshop'
Visual imagery: 'Makes me picture myself in a coming-of-age indie movie, running in slow motion'
Mood descriptions: 'It has this melancholic yet hopeful quality that hits my soul'
Sensory experiences: 'The song feels like a warm embrace on a cold day'
Abstract feelings: 'Gives me this feeling of floating just above my problems'

Lyrical (focusing on words and meaning)
This includes thoughtful commentary on:

Storytelling: 'The lyrics tell such a vivid story of lost love that I feel like I've lived it'
Wordplay: 'The clever double entendres in the chorus make me appreciate it more each listen'
Messaging: 'The subtle political commentary woven throughout the verses really resonates'
Personal connection: 'These lyrics seem like they were written about my own life experiences'
Quotable lines: 'That line 'we're all just stardust waiting to return' lives rent-free in my head'

Metadata (using information found in labels or research)
This includes interesting facts like:

Technical info: 'The song is hip-hop from the year 2012 with a bpm of 100'
Creation context: 'They recorded this album in a cabin with no electricity using only acoustic instruments'
Chart performance: 'It's wild how this underplayed track has over 500 million streams'
Artist background: 'Knowing the guitarist was only 17 when they recorded this makes it more impressive'
Release details: 'This deluxe edition has three bonus tracks that are better than the singles'

Sentiment (whether the person feels good or bad about the song)
Output Format
Return your analysis as a structured JSON with these categories:
Copy{
  'pairs': [(song_1, artist_1), (song_2, artist_2), ...],
  'Descriptive': [],
  'Contextual': [],
  'Situational': [],
  'Atmospheric': [],
  'Lyrical': [],
  'Metadata': [],
  'Sentiment': []
}
Example
Input:
'I like Plastic Love by Mariya Takeuchi because of the funky, jazzy, retro vibes. I listen to this music at 3am when Im lonely because it romanticizes my loneliness and makes it meaningful. It helps me
```

```
    to enjoy my own loneliness. It has very distinctive synthesizer
    sounds in the chorus and leading bass lines in the bridge. The
    vocals are chill and blended. Another song that sounds very
    similar is Once Upon a Night by Billyrrom or Warm on a Cold Night
    by Honne. The genre is like City Pop which describes an idealized
    version of a city.'
Output:
Copy{
  'pairs': [('Plastic Love', 'Mariya Takeuchi'), ('Once Upon a Night',
     'Billyrrom'), ('Warm on a Cold Night', 'HONNE')],
  'Situational': ['3am when Im lonely'],
  'Descriptive': ['funky', 'jazzy', 'retro vibes', 'distinctive
    synthesizer in chorus', 'leading bass lines in bridge', 'chill and
     blended vocals', 'genre of City Pop'],
  'Atmospheric': ['romantic loneliness', 'vulnerability', 'kind of sad
     in a good way', 'acting heartbroken', 'idealized version of a
    city'],
  'Contextual': ['Plastic Love sounds similar to Once Upon a Night', '
    Plastic Love sounds similar to Warm on a Cold Night'],
  'Metadata': ['funky', 'jazzy', 'retro vibes', 'genre of City Pop']
}
```

D.2 Hallucination Check Prompt

Below we present the prompt which is used to validate the results of an extraction and summarization. Here, we use a secondary model to check for hallucination between an extraction of semantic tags and their sentence-like summarization. Please note that we present the LLM with two examples: one negative (i.e. containing no hallucinations) and one positive (i.e. containing hallucinations) as we found in our ablation experiments that this significantly improved the model's ability to identify hallucinations.

```


% Getting summarizations
# Summarization task

Write a sentence which combines the associated sentence fragments.
Please do not add anything other than the information given to you.

Your description should:
- Be maximum 4 sentences in length

Your description shouldn't:
- Add any additional information that is not present in the tags
- Include any information that is based on your own knowledge or
    assumptions

Example:
  'Situational':  ['3am when Im lonely'],\
  'Descriptive':['funky', 'jazzy', 'retro vibes', 'distinctive
    synthesizer in chorus', 'leading bass lines in bridge', 'chill and
     blended vocals', 'genre of City Pop'],\
  'Atmospheric'': ['romantic loneliness', 'vulnerability', 'kind of
    sad in a good way', 'acting heartbroken', 'idealized version of a
    city'],\
  'Contextual': ['Plastic Love sounds similar to Once Upon a Night', '
    Plastic Love sounds similar to Warm on a Cold Night'],\
  'Metadata': ['funky', 'jazzy', 'retro vibes', 'genre of City Pop']\

  Desired output: This song has funky, jazzy, retro vibes. I listen to
     this music at 3am when Im lonely because it romanticizes my
    loneliness and makes it meaningful. \
```

```
 It helps me to enjoy your own loneliness. It has very distinctive synthesizer sounds in the chorus and leading bass lines in the bridge. \
 The vocals are chill and blended.  The genre is like City Pop which describes an idealized version of a city.'  \

Tags:

{input_tags}

% Hallucination
# Hallucination Check Prompt for Generated Summary

## Instructions
Evaluate whether the generated summary contains hallucinations based on the provided features/tags from the original source.
A hallucination is defined as information in the summary that is not present in or contradicts the features from the source material.

## Input Format
- **Original Features/Tags**: [List of key features/tags from the source material]
- **Generated Summary**: [The summary to be evaluated]

## Task
1. Compare each claim or statement in the summary against the original features/tags
2. Identify any information in the summary that:
   - Is not supported by the original features/tags
   - Contradicts the original features/tags
   - Represents an embellishment beyond what can be reasonably inferred
3. **The output should be in JSON format.**

## Output Format
```
{{
"hallucination_detected": [True/False],
}}
```

## Example 1
**Input Data**:
{{
  "original_features": {{
    'situational':  ['3am when Im lonely'],
    'descriptive':['funky', 'jazzy', 'retro vibes', 'distinctive synthesizer in chorus', 'leading bass lines in bridge', 'chill and blended vocals', 'genre of City Pop'],
    'atmospheric'': ['romantic loneliness', 'vulnerability', 'kind of sad in a good way', 'acting heartbroken', 'idealized version of a city'],
    'contextual': ['Plastic Love sounds similar to Once Upon a Night', 'Plastic Love sounds similar to Warm on a Cold Night'],
  }},
  "generated_summary": 'funky, jazzy, retro vibes. I listen to this music at 3am when Im lonely because it romanticizes my loneliness and makes it meaningful.
    It helps me to enjoy your own loneliness. It has very distinctive synthesizer sounds in the chorus and leading bass lines in the bridge.
    The vocals are chill and blended.  The genre is like City Pop which describes an idealized version of a city.'
}}
```

~~~

Expected Output:
```
{{
"hallucination_detected": False,
}}

## Example 2
**Input Data**:
{{
  "original_features": {{
    'situational':  ['when I'm quitting my corporate job'],
    'descriptive':['angry punk guitar', 'killer drums', 'harcore vocal processing', 'distortion'],
    'atmospheric'': ['pumped up vibes', 'makes me want to take charge of my life'],
    'contextual': [''],
  }},
  "generated_summary": 'This song makes me happy. It has a soft and exciting vibe with killer drums. I listen to this song at parties or festivals when I feel positive.'
}}

**Expected Output**:
```
{{
"hallucination_detected": True,
}}
```

**Input Data**:
```
{{
  "original_features": {features},
  "generated_summary": {summary}
}}
```
**Expected Output**:
```
~~~

E Properties of the Dataset

We present additional insights into several unique aspects of MusicSem. As mentioned in earlier sections, MusicSem contains two key attributes: personalization and contextualization.

Personalization As we show in Table 12, for each song in our dataset there are approximately 3 different posts which discuss it. This yields a variety of annotations containing differing opinions on the same song. For example, in Figure 4 we showcase the semantic associations of two different users for the same song. We can see that this broadens the scope of perspectives that are represented by a dataset, presenting the opportunity for a more nuanced understanding of each musical piece.

Contextualization of Songs In Table 12 we can see that many songs are presented in tandem, where each post contains approximately 10 songs. For an intuitive example of this, we present a case study in Figure 4. In this case study the user describes a set of songs which are aligned under a unified theme (e.g. positivity). This form of contextualization provides an explicit definition of the underlying latent need that creates association between songs.

Table 12: Properties of the dataset.

Total Size	# Unique Songs	# Unique Artists	# Posts per Song	# Songs per Post	# Genres per Song
35,977	11,842	4,430	2.98	10.51	2.71

Example of Personalization

Speaking about **David Bowie's Wishful Beginnings**
User1: 'hackneyed drum beat', 'horrible synthy bassline', 'flat and uninteresting music', 'dated sound';
User 2: 'awesome voice', 'perfectly used vocal effects', 'amazing layering', 'worth listening with good headphone'

Example of Contextualization

suggestions for songs that emanate optimism (but not toxic positivity)?

Hi! I'm working on a playlist for my girlfriend. Her depression has been bad lately and so I wanted to make her a playlist of songs that feel like a warm hug and can comfort her. It's important that these songs promote genuine optimism as opposed to toxic positivity. I already have about 90 minutes worth of songs but I was wondering what others could help. Some examples of what I already have: (1) Your Song- Elton John -- our song :) // (2) Wildflowers- Tom Petty // (3) Rainbow- Kacey Musgraves // (4) Dusty Trails- Lucius //(5) Grow As We Go- Ben Platt

Figure 4: An example of personalization and contextualization on Reddit.

F Dataset Visualizations

We present a series of figures for visualizing key aspects of our dataset. First, we showcase the unique genres associated with our dataset in Figure 5. As we can see from the cloud, our selection of threads has created a very high representation of rock, electronica, and pop music in our dataset. This selection bias is broadly addressed in the limitations portion of our dataset. In this version of our dataset we focus on finding semantically rich musical discourse on Reddit without specifically considering genre coverage. In the future we hope to continue expanding the scope of the dataset to include a broader variety of genres.

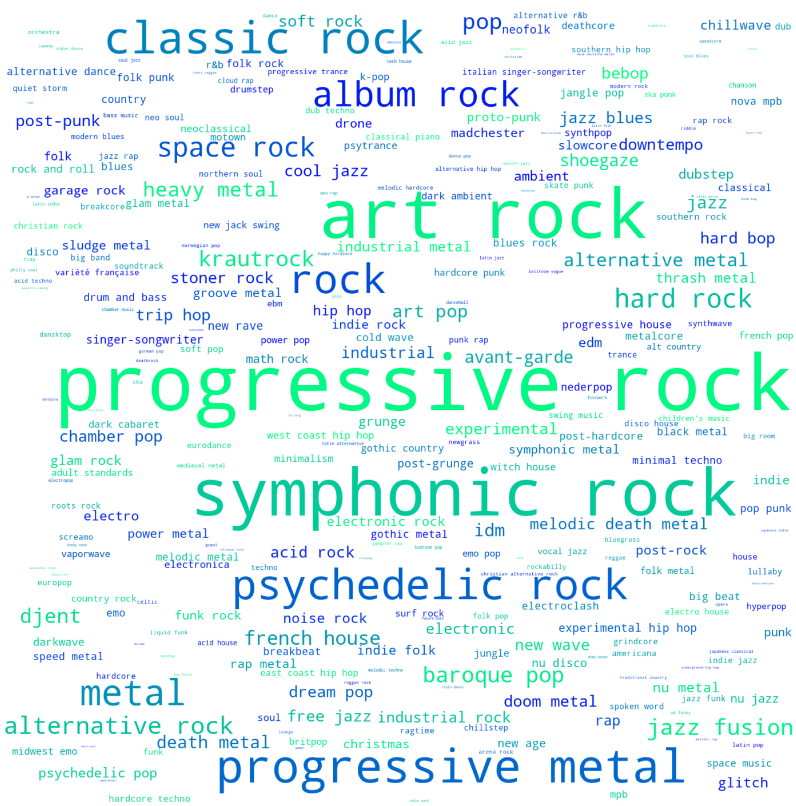

Figure 5: Word cloud of the genres in MusicSem, where genres with larger font size correspond to higher popularity in the dataset.

We visualize the distribution of song appearances in our dataset in Figure 6. Here we can see that the dataset follows a power-law distribution where some songs are mentions a large number of times and others are rarely discussed. This is aligned with common trends in music datasets where popularity bias often creates significant disparities between representation of mainstream and niche musics [72].

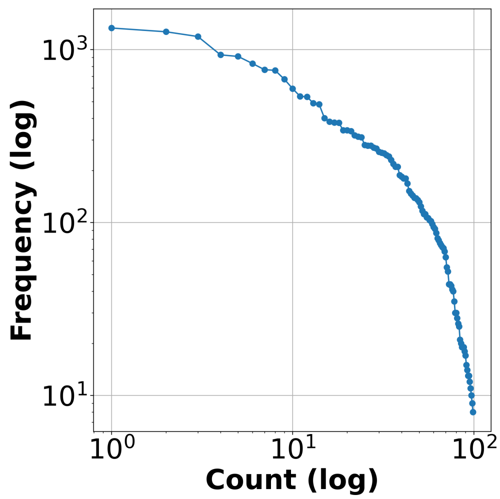

Figure 6: Popularity distribution in MusicSem.

Finally, we consider the word count of the raw posts in Figure 7. As we can see, this dataset skews towards longer discussions with more than 360 characters in each one. This contributes to the rich vocabulary of our dataset and the abundance of semantic content found within it.

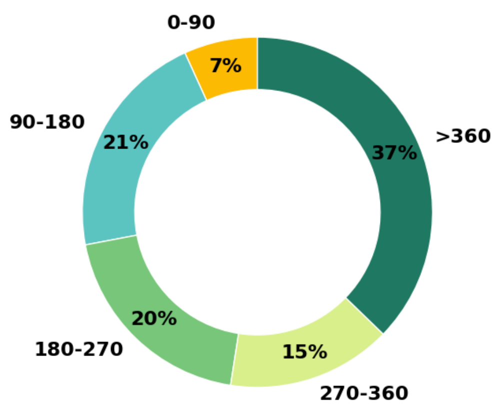

Figure 7: Distribution of number of words in each data sample.

G Safeguards

Here, we specifically address the sensitive nature of releasing data that is scraped from the internet. In our work we release a large collection of Reddit threads which were scraped from the internet. While we understand that releasing data which is scraped from Reddit can have lasting impacts, we do our best to mitigate these. First, since the domain of our dataset is music, we are not dealing with a safety-critical setting. Second, although the original raw posts contain the user ids, we do not release these in the final version of our dataset. Finally, given the already anonymous nature of Reddit, we hope that our scraped posts will cannot be used to identify specific users.

H Experimental Settings

H.1 Hyperpameter Settings

We present the baseline models and the specific details of their implementations. The evaluation involves both retrieval and generation tasks, where the tested models are summarized in Table 13.

Table 13: An overview of all the models we evaluate in this work. 'Hier.', 'Trans.', 'Diff.', and 'Co-List.' are short of Hierarchical, Transformer, Diffusion, and Co-Listing, respectively.

Task	Name	Date	Architecture	Text Conditioner	Length	Sample Rate	Proprietary
Text-to-Music	MusicLM [18]	2023	Hier. Trans. + SoundStream	w2v-BERT [73]	variable	24kHz	
	AudioLDM 2 [21]	2023	VAE + 2D U-Net	CLAP [12]	variable	16kHz	
	Stable Audio [20]	2023	VAE + 2D U-Net	CLAP [12]	up to 95s	48kHz	
	MusicGen [19]	2024	AE + 1D U-Net	FLAN-T5 [74]	10s	48kHz	
	Mustango [30]	2024	VAE + 2D U-Net	FLAN-T5 [74]	10s	16kHz	
	Mureka	2024	-	-	-	-	✓
Task	**Name**	**Year**	**Architecture**	**Audio Conditioner**	**Length**	**Sample Rate**	**Proprietary**
Music-to-Text	MU-LLaMA [15]	2024	Diff. Trans.	MERT [57]	60s	16kHz	
	LP-MusicCaps [7]	2023	Trans.	BART [75]	10s	16kHz	
	FUTGA [17]	2024	Hier. Trans. + VAE	Whisper[76]	240s	16kHz	
Task	**Name**	**Year**	**Architecture**	**Modalities**	**Length**	**Sample Rate**	**Proprietary**
Retrieval	CLAP [12]	2023	Contrastive Learning	Text + Waveform	-	48kHz	
	LARP [9]	2024	Contrastive Learning	Text + Waveform + Co-List. Graph		48kHz	
	ImageBind [14]	2023	Contrastive Learning	Text + Image	-	16kHz	
	CLaMP3 [13]	2024	Contrastive Learning	Text + Image + Waveform	-	24kHz	

H.2 Cross Modal Retrieval Models

CLAP [12] learns joint embeddings between audio clips and text descriptions through Contrastive Language-Image Pretraining `https://arxiv.org/abs/2103.00020`, on 630K audio-text pairs. For audio data, it first represents signals using log Mel spectrograms at a sampling rate of 44.1kHz, then employs CNN14 [77] (80.8M parameters) pretrained on AudioSet with 2M audio clips. For text data, it uses BERT [78] (110M parameters) to encode text descriptions, taking the [CLS] token embedding as text representation. Both modality encodings are projected into a multimodal space using two learnable projection matrices, resulting in an output dimension of 1024. We employ its music variant from the official repository `https://github.com/LAION-AI/CLAP`.

LARP [9] addresses the cold-start problem in playlist continuation through a three-stage contrastive learning framework. Built upon the BLIP framework, it consists of two uni-modal encoders: HTS-AT [79] for audio encoding and BERT for text processing (using [CLS] token embeddings), with their original 768-dimensional encodings being projected into a unified 256-dimensional space. The framework then performs within-track contrastive learning, track-track contrastive learning, and track-playlist contrastive learning to optimize representations from both semantic and intra-playlist music relevance perspectives. We use the official implementation from `https://github.com/Rsalganik1123/LARP`.

ImageBind [38] unifies six modalities (image, audio, text, etc.) in a single embedding space through multimodal contrastive learning. While not music-specific, its general-purpose audio-text alignment capability provides a strong baseline for cross-domain retrieval. ImageBind employs Transformer architectures for all modality encoders. For audio input, it converts 2-second 16kHz samples into spectrograms using 128 mel-spectrogram bins. Treating spectrograms as 2D signals similar to images, it processes them using a ViT with patch size 16 and stride 10. For text input, it utilizes pretrained text encoders (302M parameters) from OpenCLIP [80]. After projection, different modalities are encoded into a 768-dimensional shared space. We extract audio embeddings from the ViT-B/16 variant available at `https://github.com/facebookresearch/imagebind`.

CLaMP3 [13] establishes a unified multilingual music-text embedding space through cross-modal alignment of sheet music, audio recordings, and text in 12 languages. The audio processing pipeline adopts pre-trained acoustic features from MERT-v1-95M [57]. Each 5-second clip is represented by a single embedding obtained through averaging across all MERT layers and time steps. For textual content processing, the model employs XLM-R-base [81], a multilingual transformer, which features a 12-layer architecture with 768-dimensional hidden states. The framework implements contrastive learning to align multimodal representations, incorporating novel components such as a retrieval-augmented training mechanism that enhances cross-modal association. We use the checkpoints and architecture from the original authors' implementation at `https://sanderwood.github.io/clamp3`, specifically the SaaS version optimized for audio.

H.3 Cross Modal Generation Models

Music-to-Text Generation Models:

MU-LLaMA[15] is a music-specific adaptation of the LLaMA-2-7B architecture, integrating MERT [57] acoustic features through LLaMA-Adapter [82] tuning. We use the official implementation from `https://github.com/shansongliu/MU-LLaMA`, with the same hyperparameter settings: the input audio is split into 60-second audio signal at 16 kHz and the temperature for LLaMA-2-7B is set to 0.6, top_p is set to 0.8, and the maximum sequence length is 1024 tokens.

LP-MusicCaps [16] employs a BART-based encoder-decoder architecture [75] with 768 widths and six transformer blocks for both the encoder and the decoder, and the encoder takes a log-mel spectrogram with convolution layers similar to whisper [76]. We use the official implementation from `https://github.com/seungheondoh/lp-music-caps` and their pretrained checkpoint, splitting our test audio to 10-second audio signal at 16 kHz and choose the longest caption among all the clips as the inference result. In addition, the num_beams is set as five and the maximum sequence length is 128 tokens.

FUTGA[17] enables time-located music captioning by automatically detecting functional segment boundaries. Built upon SALMONN-7B [83] with LoRA-based instruction tuning, it integrates a music feature extractor for full-length music captioning. For our evaluation of this model we use the checkpoints and architecture presented by the original authors on `https://huggingface.co/JoshuaW1997/FUTGA`. In the implementation, Vicuna-7B `https://huggingface.co/lmsys/vicuna-7b-v1.5` is used as the backbone. For the hyperparameter settings, the repetition_penalty is set to 1.5, num_beams is set to 5, top_p is set to 0.95, top_k is set to 50, and an audio file is processed as 240-second 16kHz audio signal.

Text-to-Music Generation Models:

MusicLM [18] is a generative model that produces high-quality music from text prompts by using a hierarchical sequence-to-sequence approach. It leverages audio embeddings from a self-supervised model and autoregressively generates semantic and acoustic tokens. Unfortunately this model does not have any publicly available architecture or checkpoints. However, we use a crowd-sourced implementation available at `https://github.com/zhvng/open-musiclm`. Notably, this implementation deviates from the originally proposed text conditioning model by using the open-sourced version of CLAP [12] instead of Mulan [84] and Encodec [58] instead of SoundStream [85]. The purpose of including this implementation is to showcase the performance of a large collection of publicly available models.

Stable Audio [20] is a diffusion-based music generation model that creates audio from text and optional melody input, using a latent audio representation. The Stable Audio model is based on a combination of a latent diffusion model consisting of a variational autoencoder, a conditioning signal, and a diffusion model. The VAE consists of a Descript Audio Codec [86] encoder and decoder. The textual conditioning signal comes from a pre-trained CLAP model [12], specifically the HT-SAT [79] and RoBERTa-based [87] iteration. Finally, the diffusion model is based on a U-Net [23] which consists of 4 levels down-sampling encoder blocks and up-sampling decoder blocks, with skip connections between them. encoder and decoder blocks providing a residual For our evaluation of this model we use the checkpoints and architecture presented by the original authors on `https://github.com/Stability-AI/stable-audio-tools`.

MusicGen [19] is a transformer-based model that generates music from text descriptions. In our implementation with use the 300M parameter model. This model uses a five layer EnCodec model for 32 kHz monophonic audio with a stride of 640, resulting in a frame rate of 50 Hz, an initial hidden size of 64 and a final embedding size of 640. The embeddings are quantized with using an RVQ with four quantizers, each with a codebook size of 2048. Finally, for sampling, the model employs top-k sampling, keeping the top 250 tokens and a temperature of 1.0. For our evaluation of this model we use the checkpoints and architecture presented by the original authors in `https://github.com/facebookresearch/audiocraft`.

AudioLDM2 [21] is a diffusion model for text-to-audio generation, trained on large-scale data and designed to handle diverse audio types, including music and sound effects. It improves over its predecessor by using high-quality representations and efficient training strategies. For our evaluation we use the checkpoints and architecture presented by the original authors in `https://github.com/haoheliu/AudioLDM2`. For the specific hyperparameters of the checkpoint architecture, we use the version with a 2-layer latent diffusion model. As their audio encoder the model uses a AudioMAE with a patch size of 16 ×16 and no overlapping, resulting in a 768-dimension feature sequence with length 512 for every ten seconds of mel spectrogram. For the text encoder there is a GPT-2 model that has an embedding dimension of 768 with 12 layers of transformers.

Mustango [30] is a multi-stage latent diffusion model that generates music from text prompts, focusing on both coherence and audio quality. It introduces a time-aware transformer to model long audio sequences and supports multi-track generation. For our evaluation we use the checkpoints and architecture presented by the original authors in `https://github.com/AMAAI-Lab/mustango`. During inference, the model uses two transformer-based text-to-music-feature generators which predict the beat and chord features. For the beats prediction, this model uses DeBERTa Large model [88] which predicts both the meter and the sequence of interval duration between the beats. Simultaneously, the chord predictions are made by a FLAN-T5 Large model [74].

Mureka is a proprietary music generation model available at `https://www.mureka.ai`. We build our own pipeline for making calls to their API which will be available on our Github repository once the API issues are resolved.

H.4 Computational Resources

For generative tasks, all experiments were conducted on a system equipped with NVIDIA L40 GPUs with 48GB VRAM per card, utilizing 12.6. Each experiment was executed on a single GPU instance.

For retrival tasks, all experiments were conducted on a system equipped with NVIDIA A40 GPUs with 46GB VRAM per card, utilizing CUDA 12.4. Each experiment was executed on a single GPU instance.

H.5 Runtime Analysis

Text-to-Music Generation

Table 14: The inference time of text-to-music generation models on MusicSem. Note: Tradeoff = Inference Time/Generation Size.

Model	Inference Time (sec)	Generation Size (sec)	Tradeoff ↓
MusicLM	102	5	20.40
AudioLDM2	13	10	1.30
Mustango	50	10	5.00
MusicGen	40	20	2.00
Stable Audio	18	45	0.40
Mureka	120	150	0.80

We evaluate the relationship between inference time and duration of its generated music in Table 14. Since the duration of generation varies significantly between models and generating longer stretches of cohesive music is a critical challenge in the task of text-to-music generation 19, we evaluate each model using the duration settings specified in its original formulation and code base. We present a tradeoff metric which is calculated as a ratio of Inference Time divided by Generation Size. From the results in Table 14 we can see that of the publicly available models, Stable Audio has the best latency during inference time. Furthermore, we can see that the proprietary model, Mureka, is able to generate longer stretches of cohesive audio than all the publicly available models, signaling a clear gap between the publicly available generation models and those which require payment.

Music-to-Text We evaluate the relationship between inference time and the length of the annotation produced by a model in Table 15. From the results we can see that LP-MusicCaps has the highest tradeoff (i.e. one second of inference time generates the highest number of characters).

Table 15: The inference time of music-to-text generation models on MusicSem. Note: Tradeoff = Inference Time/Generation Size.

Model	**Inference Time (sec)**	**Generation Size (in characters)**	**Tradeoff ↓**
LP-MusicCaps	8	2000	0.004
MU-LLaMA	4	70	0.057
FUTGA	15	1138	0.013

Text-to-Music Retrieval

We evaluate the inference time of the cross modal retrieval models in Table 16. AS we can see from these results, there is little variability across the latencies of the models during inference however ImageBind [14] is slightly faster.

Table 16: The inference time of cross modal retrieval models on MusicSem.

Model	**Inference Time (sec)**
LARP	0.26
CLAP	0.23
ImageBind	0.21
CLAMP3	0.28

H.6 Evaluation Metrics

H.6.1 Intuition for Interpreting Music-to-Text Metrics

In this section we present a brief overview for the metrics used for evaluating music-to-text models. Following the canonical works in music-to-text generation [15, 16] we begin by presenting three n-gram based metrics borrowed from machine translation tasks called BLEU [45], ROUGE [47] and METEOR [46]. BLEU (B) uses precision to compare the overlap in n-grams (sequences of 1, 2, or 3 words - (B_1, B_3, B_3) between the original annotation and the generated musical caption. Alternatively, ROUGE (R) uses recall to compare the overlap in n-grams between the original annotation and the generated musical caption. Finally, METEOR (M) is designed to be better aligned with human judgments by extending the comparison to include synonym and paraphrasing-based matches in addition to the exact matches covered by BLEU/ROUGE. Meanwhile, we also include the CIDEr [48] metric which was originally proposed for image captioning. This metric measures how well the generated text matches the consensus of a set of original annotation, using a weighted n-gram similarity. Finally, we present the Bert Score [42] which uses the Bert model to compare the embeddings between the generated and original musical annotations.

The purpose of using each of these evaluation metrics is to present increasing levels of abstraction in considering the alignment between the original annotations and their generated counterparts. As we can see the Bert Score remains the most stable across all three datasets while the range of the n-gram based metrics maintains high variability between both datasets and models.

H.6.2 CLAP Score

Contrastive Language-Audio Pretraining [12] Score (CLAP Score) is a simple but effective and reference-free metric that quantifies how closely audio signal matches a text description. This metric is commonly used in text-to-music generation to evaluate how well a generative model is able to express the information provided in a textual input which forms the basis for its generation. Thus, given a set of associated language-audio pairs, $(T, \tilde{A})$ where the audio $\tilde{A} = \mathcal{M}(T)$ is generated by providing the associated textual inputs T to a music generation model (e.g. MusicGen [19]). We can generate embeddings for each modality using the CLAP model such that

$$Z_{\tilde{A}} = \text{CLAP}_{\text{audio}}(\tilde{A}), \quad Z_T = \text{CLAP}_{\text{text}}(T),$$

where $Z_{\tilde{A}}, Z_T$ are the output from the audio encoder and text encoder for the CLAP model, respectively. Given these sets of audio and text embeddings we can measure the cosine similarity of the audio and the text embeddings in their joint representation space. We slightly abuse the notation for indexing borrowing from the syntax used for coding matrices such that $Z_{\tilde{A}}[i]$ reflects the i-th embedding. Thus, we can formalize the CLAP Score as:

$$CS(T, \tilde{A}) = \frac{1}{n} \sum_{i=1}^{n} \frac{< Z_{\tilde{A}}[i], Z_T[i] >}{||Z_{\tilde{A}}[i]|| \cdot ||Z_T[i]||}.$$

As we can see, the more alignment there is between the language and audio representational spaces, the higher this score will be.

H.6.3 Vendi Score

It is non-trivial to confirm that the Vendi Score [55], which is normally used for images is compatible with spectrograms (which are the images of audio when mapped to the frequency domain). Thus, we conduct a small ablation study to identify whether the Vendi Score is sensitive to changes in music. We construct a small ablation set of 15 seed tracks. For each seed track we select three *positive* and three *negative* examples. In this case, the positive examples consist of cover songs in which another musician sings the same song as the initial seed track. Meanwhile, negative examples consist of songs from completely different genres and artists. We hypothesize that if the Vendi Score can, indeed, be used to measure diversity in collections of audio, then it will clearly distinguish between the positive and negative groups when applied with respect to a seed track. As we can see from Figure 8, the Vendi score is clearly able to distinguish between an original seed track when compared with its "synthetic" example (e.g. covers) or opposite "negative" examples. For almost each song in our 15 seed tracks (represented across the x-axis) we can see that the score is noticeably higher among the negative examples (orange) than the cover songs (blue). The songs selected for our ablation study can be found at `https://tinyurl.com/2ff3d4f6`.

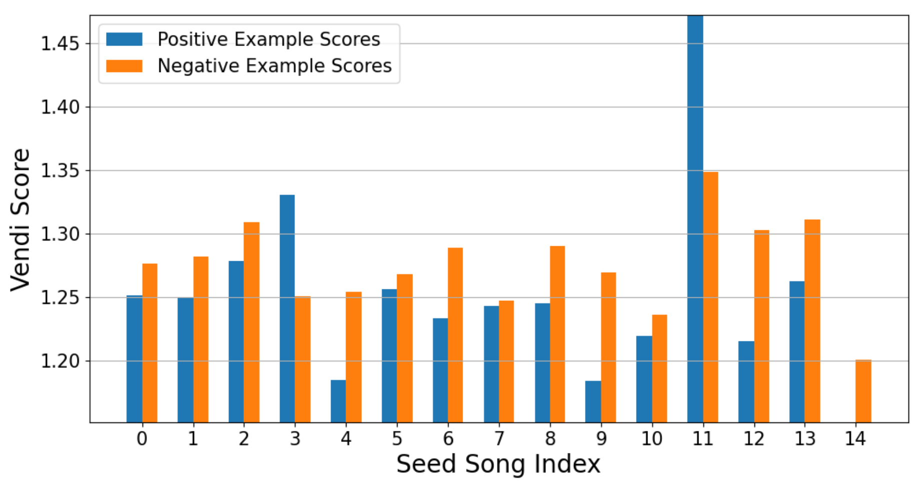

Figure 8: Comparison between Vendi Scores for cover songs and random collection.

H.7 Dataset Splits

For each of our evaluations on MusicCaps [18] and Song Describer [22], we evaluate over the entire dataset that is currently publicly. This choice is justified by the fact that neither dataset has openly published concrete train-test splits which can be used to standardize over models. For example, although in the original MusicCaps paper they address the existence of a test set, on the publicly available version of their dataset released on Kaggle there is only a training split. Thus, in many of the works which evaluate on MusicCaps, they simply create a synthetic test set by implementing their own train-test split over the available data [30, 17, 16]. And, without testing over the entire set, we cannot ascertain their performance. Unfortunately, the possibility of overfitting cannot be accounted for without leaderboard access to the held-out test set. The same holds for Song Describer. Loading from the storage site does not yield any clear demarcation of the dataset meaning that each paper that evaluates on this dataset selects its own split. Since in our work we do not engage in any fine-tuning, we felt it was best to evaluate over the entire set and see the final performance. Meanwhile, for our dataset, which has a clearly demarcated evaluation set, we use only this portion of the data for evaluation and publish the rest for training.

I More Evaluation Results

I.1 Case study of music-to-text generation

When looking at the performance of the various music-to-text models reported in Table 8 within the main body of this work, it seems that LP-MusicCaps is the best performing model but a deeper analysis of its output challenges this. In Figure 9 we showcase a case study of the comparative outputs between the original annotation and the captions produced by each model. As we can see in the case study, FUTGA generates a much more detailed and accurate description of the audio however, it receives a lower overall performance score because, in generating more content, it has the potential for a lower n-gram overlap. Meanwhile, MU-LLaMA, though completely incorrect retains scores which are close to that of FUTGA potentially due to the shortened length of the model's output. Furthermore, despite the seemingly high performance of each model on the objective metrics, each caption output contains at least one factually incorrect description of the input music. This indicates that there is still a significant information gap that SOTA models are unaware of.

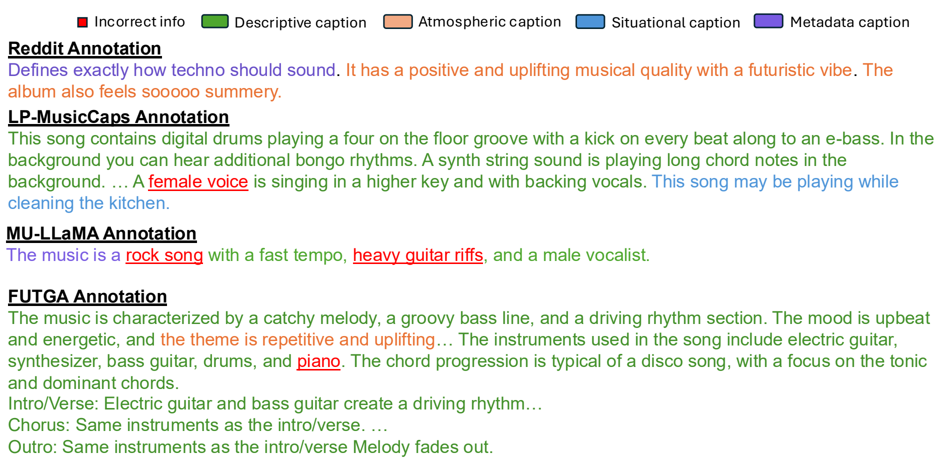

Figure 9: Case study of music-to-text generation evaluation. We can see that all models make objective factual errors and focus primarily on descriptive annotations. For reference, please listen to the song on Youtube – While Others Cry by The Future Sound of London.

J Broader Impact

From a positive perspective we consider the broader impacts of this work on several key levels. First, we can consider the contributions of this work to the broader domain of general AI in which there remains a large gap between performance in specialized domains like music (or other art forms) and general purpose domains. MusicSem takes another step toward addressing this gap by providing a more nuanced understanding of musical discourse. Furthermore, our work will have concrete impacts on the music domain. First, the nature of our principled study will create a foundation through which to compare model performance. Second, our newly introduced sensitivity metrics will improve our ability to audit existing models for their semantic awareness. Third, given the flexibility and diversity of tasks served by our dataset, MusicSem will continue to be relevant as the field of music understanding continues to expand.

From a negative perspective, we grapple with the controversial nature of generative art and its associated with the dis-empowerment of artists and other creators. We concede that there are many issues associated with the generation of music such as memorization which are not addressed by our work. Although this work does not explicitly exacerbate these issues, we understand that any contribution to the generative domain should be treated with sensitivity and care.