MuseChat: A Conversational Music Recommendation System for Videos

Zhikang Dong [*†]
Stony Brook University
zhikang.dong.1@stonybrook.edu

Bin Chen
TikTok
chen.bin@bytedance.com

Xiulong Liu [†]
University of Washington
xl1995@uw.edu

Paweł Polak
Stony Brook University
pawel.polak@stonybrook.edu

Peng Zhang
TikTok
zhang.peng@bytedance.com

Abstract

We introduce MuseChat, an innovative dialog-based music recommendation system. This unique platform not only offers interactive user engagement but also suggests music tailored for input videos, so that users can refine and personalize their music selections. In contrast, previous systems predominantly emphasized content compatibility, often overlooking the nuances of users' individual preferences. For example, all the datasets only provide basic music-video pairings or such pairings with textual music descriptions. To address this gap, our research offers three contributions. First, we devise a conversation-synthesis method that simulates a two-turn interaction between a user and a recommendation system, which leverages pre-trained music tags and artist information. In this interaction, users submit a video to the system, which then suggests a suitable music piece with a rationale. Afterwards, users communicate their musical preferences, and the system presents a refined music recommendation with reasoning. Second, we introduce a multi-modal recommendation engine that matches music either by aligning it with visual cues from the video or by harmonizing visual information, feedback from previously recommended music, and the user's textual input. Third, we bridge music representations and textual data with a Large Language Model(Vicuna-7B). This alignment equips MuseChat to deliver music recommendations and their underlying reasoning in a manner resembling human communication. Our evaluations show that MuseChat surpasses existing state-of-the-art models in music retrieval tasks and pioneers the integration of the recommendation process within a natural language framework.

1. Introduction

Music serves as a complementary modality within videos, enriching the viewer's experience and aiding in content comprehension. Thus, choosing the right music for a video is crucial. Current music recommendation systems effectively curate lists of tracks that harmonize with a video's content. For instance, they might select scary music for a horror movie or high-energy tracks for a dance video. While this focus on content compatibility is important, we argue that user preferences are equally essential. For example, individuals born in the '80s may prefer synth-pop for a nostalgia-themed video, whereas teenagers might lean toward contemporary pop. Both genres fall under the 'pop' category, but the choice between them can significantly impact user engagement.

The challenge of personalized recommendation remains relevant, as many systems leverage user profiles and activity data to generate recommendations. However, we identify two key limitations: (1) the inability to consistently meet user preferences, and (2) the cold-start problem for new users without prior data. Current music recommendation systems aim to provide lists of songs based on user history, but these may not always align with user needs for specific videos. This not only hampers user experience but also underscores the complexity of predicting preferences, which may deviate due to factors like user's recent trends. To mitigate these limitations, we propose incorporating a feedback mechanism. This would allow the system to adjust its recommendations according to user feedback, making them more aligned with changing preferences. For new users without historical data, a cold-start scenario arises, leading to recommendations to be content-driven once again.

In this study, we introduce MuseChat, a comprehensive conversational music recommendation system. As Figure 1 shows, it composes of music recommendation module and sentence generation module, so that a user can upload a video and receive a recommended music. More impor-

[*]Corresponding author.
[†]Work done during the internship at TikTok.

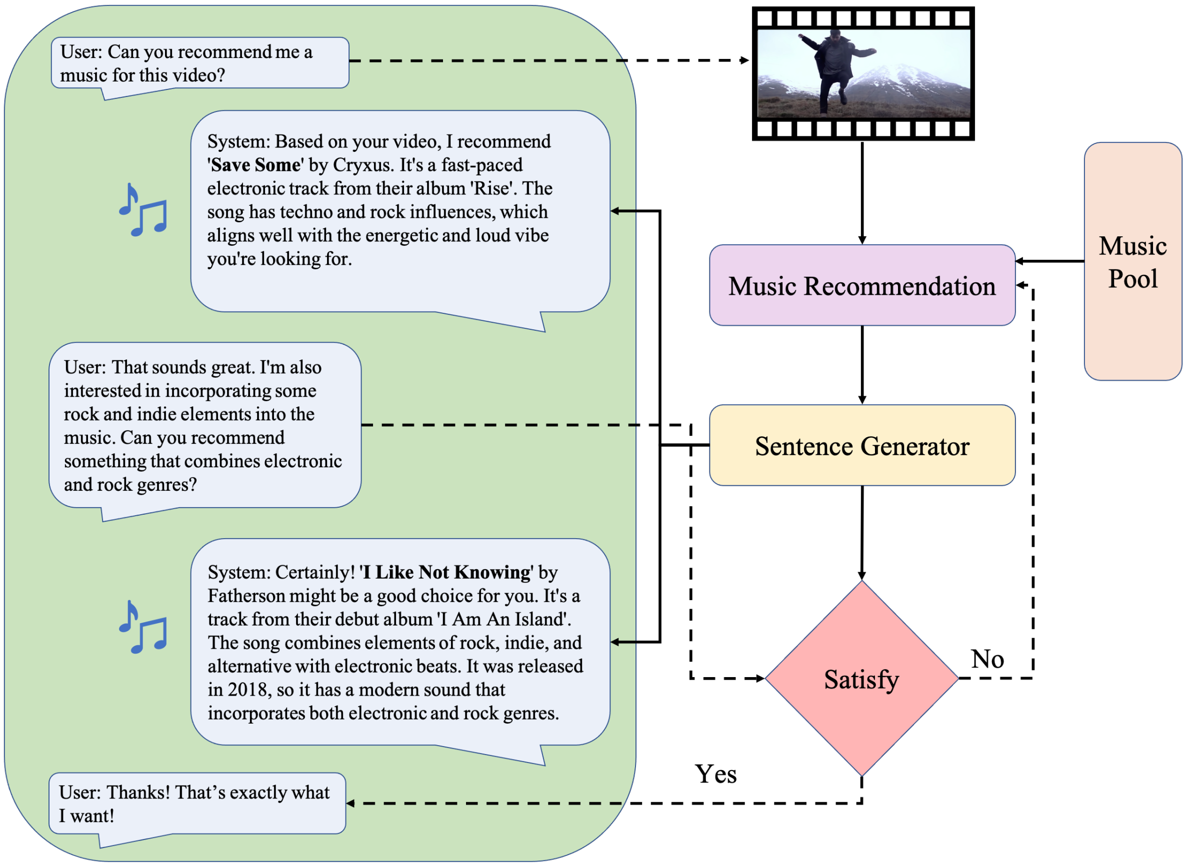

Figure 1. The MuseChat model consists of two main components: the Music Recommendation Module and the Sentence Generator Module. The Music Recommendation Module can operate in two modes: it can either process just the video input, as illustrated by the first turn in the left-hand chatbox, or use a composite of the video, the user's prompt, and previously suggested music, as shown by the second turn. The Sentence Generator Module takes the music embeddings and titles identified by the Music Recommendation Module as its inputs. Whenever a new music recommendation is generated, this module incorporates the corresponding music title, enabling the creation of natural language recommendations.

tantly, MuseChat enables users to interact with the system in a conversational manner. At each dialog turn, users can refine these recommendations by specifying criteria such as mood, genre, instruments, theme, and artist details in natural language, until they identify their desired track. Another standout feature of MuseChat is its explainability. Machine learning models usually work in a black-box fashion, lacking of interpretability on how the inner systems work and thus lower the confidence on the predictions. To tackle this, we adapt sentence generator module to MuseChat, which provides reasons for its recommendations. It not only clarifies the selection to users but also assists them in crafting their personal narratives with the music.

Constructing a conversational music recommendation system presents three core challenges. (1) **Dataset:** existing datasets such as [1, 2, 4, 27, 35, 37] primarily comprise music-video pairs, music-text pairs, or music-text-video triplets. These do not align well with MuseChat's intended use-case for two main reasons: First, current datasets only include single-turn interactions, lacking the multi-turn dialogues that are crucial for more interactive and dynamic recommendation systems. Second, these datasets omit explanations for the recommendations, a key feature for enhancing user understanding and trust. (2) **Joint Multi-modality Learning:** The task of creating a joint embedding space for video, music, and text is complex. Each of these modalities has its unique sequential features, making it a challenge to combine them into a unified representation. In this paper, we introduce a new methodology that effectively integrates spatiotemporal information from these diverse modalities, leading to a more holistic representation. (3) **Prediction Reasoning:** While current research on multi-modality Large Language Models (MLLMs) exhibit capabilities to process and understand diverse modalities, such as video and audio, a significant gap exists. Specifically, none of these models are purpose-built for the nuanced task of music interpretation and recommendation.

In order to address these challenges, we make the following contributions: (1) We introduce a novel dataset tailored for dialogue-driven music recommendations and reasoning within the context of videos. The data contains 98,206 quartets: a video, original music, candidate music and a two-turn conversation. This setup mimics the user's interaction with recommendation systems. It starts with uploading a video, receiving an initial music recommendation, and then accommodating a user's textual prompt to finalize the music selection; (2) We present a cutting-edge tri-modal architecture designed for music-video matching, enhanced with textual input. This model not only processes the previously recommended music and video content but also integrates user-provided textual prompts to fine-tune its music recommendations; (3) We equip our model with a unique feature: the ability to articulate the reasoning behind its music recommendations. By harnessing the capabilities of LLMs, we craft a sentence generator module. Drawing on music representation from an upstream module, this generator deeply understands musical features and subsequently produces coherent reasoning outputs, guaranteeing a harmonious alignment between music and textual descriptors.

2. Related Work

Automatic music tagging. Music tags efficiently summarize songs by providing descriptive keywords that cover various elements such as emotion, genre, and theme. Numerous studies have ventured into the domain of automatic music tagging, as evidenced by works such as [7, 8, 29, 30, 38, 50]. Specifically, [49] employed a model that uses shallow convolutional layers to extract acoustic features, which are then processed by stacked self-attention layers in a semi-supervised setting. Similarly, [58] introduced S3T, a self-supervised pre-training method based on the Swin Transformer [32] architecture, further optimized by a music-specific data augmentation process.

Music description in free-form natural language. Describing music in free-form natural language has also gained research attention [24, 25, 33, 50]. For instance, [13] proposed a universal retrieval system, benchmarked to handle both tag- and sentence-level inputs. This system demonstrated adaptability across nine different music classification tasks. Moreover, [34] launched "Song Describer", an open-source tool aimed at crowdsourcing text descriptions of music tracks. This initiative resulted in the creation of a public audio-caption dataset for the music domain.

Music recommendation for video. The task of music recommendation based on video attributes has received attention in previous studies [39,42,53,54]. While some work has focused on creating joint embeddings of music and free-form natural language [25, 33], other studies have examined the relationship between video, everyday audio sounds (excluding music), and language [20, 51]. [35] introduced a method that enables users to guide music recommendations using a single text description summarizing both music and video attributes. However, their approach neither incorporates user feedback nor adapts its recommendations based on such feedback or prior recommendation results. Our work with MuseChat aims to address these limitations.

Conversational recommendation system. Conversational Recommender Systems (CRS) have gained research attention for their ability to support task-oriented, multi-turn dialogues with users [10, 15, 26, 52]. These systems can capture the user's detailed and current preferences, provide explanations for suggested items, and process user feedback on recommendations. The emergence of LLMs has substantially enhanced the capabilities of CRS, particularly in understanding and generating natural language. [48] proposed an interactive evaluation approach that balances the focus between matching ground truth and maintaining interactivity. [14] developed an LLM-driven user simulator to generate synthetic dialogues, addressing the lack of conversational data. Their RecLLM, built on LaMDA [43], demonstrates versatility and fluency in recommending YouTube videos. [17] introduced Chat-Rec, a paradigm that integrates LLMs into recommender systems by transforming user profiles and historical data into prompts. This enhances both the interactivity and explainability of the recommendations and also offers solutions for cold-start scenarios. [22] redefined the recommendation challenge as a conditional ranking task. They found that, with specific prompting approaches, LLMs can achieve competitive zero-shot ranking abilities compared to traditional models. However, they also noted challenges related to LLMs' understanding of sequential interactions and their susceptibility to biases, which can be mitigated through tailored prompting and bootstrapping techniques.

Multi-modalities and Large language models. The rapid evolution of LLMs has been a game-changer in the landscape of artificial intelligence, becoming a focal point in contemporary research. Originating from transformer architectures [46], these models are trained on extensive corpora, containing billions of words [12, 41]. Noteworthy models like OpenAI's GPT-3 [5], Meta's LLaMA [45], and Google's LaMDA [43] have set benchmarks for textual generation that closely resembles human articulation. The recent advances in LLMs extend beyond text to multi-modal inputs. These models are proficient at synthesizing and interpreting information across different data types. For instance, [62] introduced MiniGPT-4, which incorporates a visual encoder into a large language model. This has led to the model's ability to generate narratives inspired by images. Other notable works include [55], which can interpret video content to generate informed textual responses, and [31], which demonstrated how to encode answer candidates into GPT-3 prompts, enabling external knowledge integration. Similarly, [19] developed a question-generation module that, when paired with a vision-language model, produced synthetic question-answer sets. The healthcare sector has also benefited from these advancements. [47, 59] used pre-trained models to interpret medical images and texts, generating concise medical reports. [61] introduced SkinGPT-4, a specialized model trained on a large dataset of skin conditions, serving as a conversational diagnostic tool. While the computational cost of training large models posed challenges for smaller research groups, recent innovations like fine-tuning adapters [11, 16, 23, 56] have democratized access. These adaptations enable smaller research teams to customize LLMs for niche applications.

3. Dataset

We are simulating a two-turn dialog to create one data sample: in the first turn only video is provided and a candidate music will be recommended by an underlying music recommendation system; In the second turn, based on the recommended music, user prompts changes in natural language to the target music, along with video and the recommended music, the system will output another one that matches the video most. The overall process of data generation is illustrated in Figure 2.

YouTube-8M dataset. We construct our conversational music recommendation dataset based on the YouTube8M dataset [1]. It is a large-scale video collection containing millions of YouTube video IDs and associated labels spread across thousands of classes, including genres like music, sports, and documentaries etc. It serves as an invaluable dataset for video understanding and especially for fields like music recognition and categorization within the broader spectrum of video research. We begin by filtering out videos tagged with "music video" and removing any unavailable videos. This process results in a dataset comprising 98,206 music videos. From each video, we extract a 120-second clip, focusing on the central segment. We then randomly allocate 88,000 of these music videos to our training set and the remaining 10,206 videos to our testing set. Each video and its corresponding music are set as the ground truth, namely video and target music as above.

Music-Video Pretrained (MVP) model. We employ a Music-Video Pretrained (MVP) model to be the recommendation system, which share the similar model strucutre as MuLan [25] model. Except the MVP model utilizes the pretrained CLIP Image encoder [40] for video feature extraction and the pretrained Audio Spectrogram Transformer (AST) [18] for music feature extraction.[1] This model is trained on our proprietary dataset consisting of millions of music-video pairs. To recommend a music from a given music pool with the input video, MVP simply takes both candidate music and video as input and outputs a similarity score. Music from the pool will then be ranked in the order of decreased similarity. Specifically, we restrict the candidate pool to be 2000 for training and 500 for testing (original music is excluded in both setting), instead of using the whole music pool, for the purpose of not being affected by low quality music. It should be noted that the MVP model here is not intended to identify the track that most similar to the original; rather, we are interested in a track that represents a noticeable divergence, serving as a prior recommendation with which the user may not be fully satisfied.

Prompt Constructor. Given a triplet consisting of a video, its original music track, and a recommended candidate music track, our aim is to construct a two-turn conversation. Specifically, during each user turn, a prompt should be provided to bridge the original music and the current recommended candidate music. During the bot's turn, descriptions about the returned music (e.g., the recommended candidate music in the first turn, and the original music in the second turn) are essential. To achieve this, we utilize the method introduced by [38] to assign tags to each music track. This tagging leverages two separate systems: one from the MagnaTagATune dataset [21] and another from the Million Song dataset [3]. Both systems have a 50-tag vocabulary, and using dual sources enhances the tagging robustness. Alongside music tags, we also collect metadata for every music video. Metadata encompasses title and video description from the YouTube website. However, it's important to note that not every music video in our dataset is labeled as an official music video by YouTube. Consequently, while every video possesses a title and description, supplementary details like official artist names, album specifics, and release dates are available for only around 30,000 tracks. With the music tags and metadata of the original and candidate music in hand, we manually craft a two-turn conversation between a user and a music recommendation system. To expand our dataset, we use above human-curated conversations as a template but provide different original-candidate music pair information along with respective metadata to guide GPT-3.5 to generate conversations case by case but in a similar manner. The full template prompt can be found in supplementary materials. While many music tracks may lack comprehensive metadata like official artist name, release date, or album title, such information could potentially be obtained from the video title or the YouTube description. GPT-3.5, with its capability to parse these sources, can extract this valuable information, which significantly elevates the quality of the simulated conversations. Each entry in our dataset, therefore,

[1]The pretrained weights used are clip-vit-large-patch14 for the CLIP Image encoder and MIT/ast-finetuned-audioset-10-10-0.4593 for the AST.

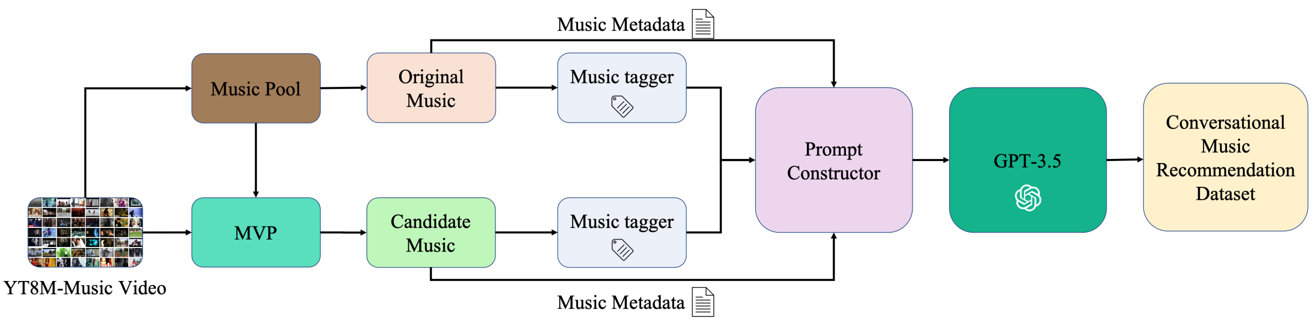

Figure 2. The generation pipeline for Conversational Music Recommendation Dataset.

consists of a video v, original target music track m_t, candidate music track m_c and simulated conversation text t. More specifically, we denote t_i as the sequence order of each conversational turn: t_1 and t_3 are from users, while t_2 and t_4 are from recommendation system.

4. Approach

We propose an approach to address this task on this dataset, setting as a new baseline performance. As showed in Figure 1, there are two main modules involved in the system: music recommendation and sentence generator. We illustrate them below.

4.1. Music Recommendation

The objective of music recommendation module is to select the most relevant music from a music pool, using a variety of inputs such as video, music and text. Each training sample is defined as a quartet (v, m_c, m_t, t_3), where v is the video, m_c denotes the candidate music track, m_t is the target original music track and t_3 is the text of user's preferences. As illustrated in Figure 3, our focus is to enhance the model's ability to transition the recommendation from a previous track m_c to target music track m_t. To accomplish this, each training sample is transformed into base features: $\mathbf{x}^v = g^v(v)$ for visual inputs, $\mathbf{x}^{t_3} = g^t(t_3)$ for text inputs, $\mathbf{x}^{m_t} = g^{m_t}(m_t)$ and $\mathbf{x}^{m_c} = g^{m_c}(m_c)$ for audio inputs. It's crucial to note that g^v and g^t are frozen during training, while g^{m_t} and $g^{m_c} are$ subject to fine-tuning. Since these features come from different backbone models, we use a trainable linear projection layer to map them into a common embedding space. This results in $\mathbf{x}^{t_3} \in \mathcal{R}^{n_t \times d}$, and $\mathbf{x}^{m_c} \in \mathcal{R}^{n_m \times d}$. Given the aim to align the target music track m_t with the overall video content, we average the sequence dimension of each $\mathbf{x}^v$ to yield $\mathbf{x}^{\bar{v}} \in \mathcal{R}^{1\times d}$. To summarize the target music, we use the first cls token from $\mathbf{x}^{m_t}$, resulting in $\mathbf{x}^{m_t^{cls}} \in \mathcal{R}^{1\times d}$.

To better capture the information from audio and text, we apply Transformer [46] encoder to both of them, denoted $f^T_{m_c}$ and f^T_t respectively, to capture long-range dependencies and complex relationships in the sequence data. This yields:

$$\begin{aligned} \widetilde{\mathbf{x}}^{t_3} &= f^T_t\left(\mathbf{x}^{t_3}\right), \\ \widetilde{\mathbf{x}}^{m_c} &= f^T_{m_c}\left(\mathbf{x}^{m_c}\right), \end{aligned} \tag{1}$$

We further develop a fusion method for the audio and text modalities, where we represent the transformed features $\widetilde{\mathbf{x}}^{t_3}$ and $\widetilde{\mathbf{x}}^{m_c}$ as sequences:

$$\begin{aligned} \widetilde{\mathbf{x}}^{t_3} &= \left[\widetilde{x}^{t_3}_{\text{cls}}, \widetilde{x}^{t_3}_1, \ldots, \widetilde{x}^{t_3}_{(n_t)}\right], \\ \widetilde{\mathbf{x}}^{m_c} &= \left[\widetilde{x}^{m_c}_{\text{cls}}, \widetilde{x}^{m_c}_1, \ldots, \widetilde{x}^{m_c}_{(n_m)}\right], \end{aligned} \tag{2}$$

where cls serves as a summary of the respective sequence, along with the other elements capturing detailed features.

Then we define our multi-head cross-modality attention layer as

$$\text{Attn}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V, \tag{3}$$

where d_k is the dimensionality of key vectors. Q and K, V are from two different modalities.

After fusing $\widetilde{\mathbf{x}}^{t_3}$ and $\widetilde{\mathbf{x}}^{m_c}$, we have final fusion embeddings:

$$\mathbf{x}^f = \mathbf{x}^{\bar{v}} + \text{Attn}(\widetilde{x}^{t_3}_{\text{cls}}, \widetilde{\mathbf{x}}^{m_c}, \widetilde{\mathbf{x}}^{m_c}) + \text{Attn}(\widetilde{x}^{m_c}_{\text{cls}}, \widetilde{\mathbf{x}}^{t_3}, \widetilde{\mathbf{x}}^{m_c}) \tag{4}$$

During training, we use the Contrastive Multiview Coding loss function [44], which is a cross-modal variant of InfoNCE [36] and NT-Xent [6]. For each batch B, we have ranking loss:

$$\mathcal{L}_{\mathcal{R}} = -\sum_{i=1}^{B}\left[\log\frac{h\left(\mathbf{x}^f_{(i)}, \mathbf{x}^{m_t^{cls}}_{(i)}\right)}{\sum_{j\neq i} h\left(\mathbf{x}^f_{(i)}, \mathbf{x}^{m_t^{cls}}_{(j)}\right) + h\left(\mathbf{x}^f_{(j)}, \mathbf{x}^{m_t^{cls}}_{(i)}\right)}\right], \tag{5}$$

where $\mathbf{x}^f_{(i)}$ and $\mathbf{x}^{m_t^{cls}}_{(i)}$ are i-th fusion vectors and target music representations in the batch respectively. $h(\mathbf{x}, \mathbf{y}) = \exp^{\mathbf{x}^\top \mathbf{y}/\tau}$ is a discriminating function, and τ the temperature hyperparameter, and τ is a trainable hyperparameter. It is essential to note that larger batch sizes have been found to be beneficial in contrastive learning [6, 28].

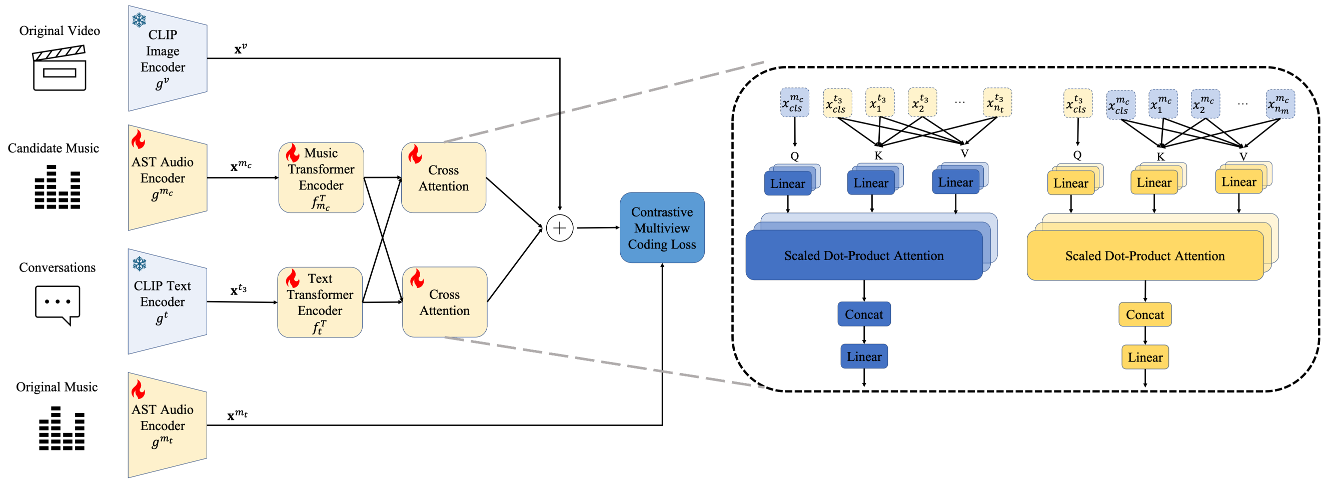

Figure 3. The music recommendation module incorporates three types of inputs: video, music, and text. For extracting base embeddings from video and text, we utilize CLIP [40]. Representations from music are extracted using the Audio Spectrogram Transformer (AST) [18]. To align these embeddings in the same dimensional space, we employ a linear projection layer. Subsequently, latent features from candidate music and text are encoded using Transformer layers. These two modalities are then fused using a multi-head cross-attention layer. The fused features, rich in contextual information, are combined with video embeddings, resulting in significant improvements.

4.2. Sentence Generator

In the second stage, we build a multi-modal LLM based on Vicuna-7B [60] by finetuning Llama2-7B [45] weights. Each training instance consists of a music representation $\mathbf{x}^{m_t}$ comes from music recommendation module music encoder and the corresponding recommendation reasoning statement t_4 from simulated recommendation system. To align music representation $\mathbf{x}^{m_t}$ with text embedding space, we train linear projection f_l to connect representation to Vicuna. To reduce the number of trainable parameters, we leverage LoRA [23] to finetune the Vicuna's attention structures. The structure is illustrated in Figure 4.

$$\mathcal{L}_{\mathcal{G}}\left(\mathbf{y};\theta\right)=\prod_{i=1}^{n}p_{\theta}\left(y_i \mid \left[f_l\left(\mathbf{x}^{m_t}\right):\mathbf{x}^{t_4}\right];\theta\right), \qquad (6)$$

where y_i is the i-th token in the response $\mathbf{y}$, and θ is the trainable parameters in linear projection layers and LoRA weights. We only compute the loss from the part of system responses during the training.

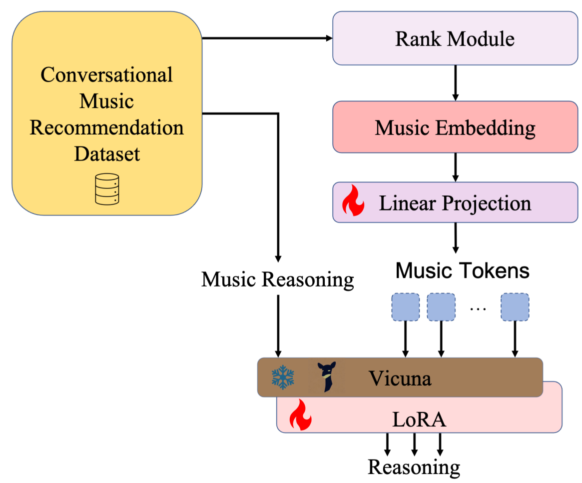

Figure 4. Illustration of sentence generator. During training, we only train the linear projection layers and the additional LoRA weights while keeping the parameters of Vicuna-7B frozen.

5. Experimental Results

5.1. Implementation Details

For each 120-second music video clip, we divide it into twelve 10-second segments and capture 5 frames per second from each segment. In our training process for the music recommendation module, each training sample includes a 10-second video clip, a corresponding 10-second original music clip, a 10-second candidate music clip, and a user prompt. To extract video and text features, we use OpenAI's CLIP model, and for audio features, we employ the AST model.[2] We convert all these basic features into 256-sized embeddings using a linear projection for each input type. Following this, we apply four Transformer encoder layers and a multi-head cross-attention layer, each with 16 heads, to process these embeddings. In the sentence generator module, we limit the maximum sequence length to 128 and set the temperature hyperparameter at 0.1. The entire

[2]The pretrained weights used are clip-vit-base-patch32 for the CLIP Image encoder and MIT/ast-finetuned-audioset-10-10-0.4593 for the AST.

Model	Train Modality	Input Modality	MR ↓	R@1 ↑	R@5 ↑	R@10 ↑	SR@10 ↑
MVP baseline	Video	Video	7	20.71	48.89	63.14	63.14
MuseChat (1st turn)	Video, Audio&Text	Video	5	20.74	48.83	63.10	63.10
MuseChat (2nd turn)	Video, Audio&Text	Video, Audio&Text	**2**	**32.79**	**63.92**	**76.53**	**82.98**
Chance	-	-	250	0.20	1.00	2.00	2.00

Table 1. Music retrieval results for baseline, multi-turn MuseChat.

model is trained using 16 Nvidia V100 32G GPUs.

5.2. Ranking Evaluation

In our test set, we have a total of 10,206 music tracks. We randomly divide these into 20 different music pools, with each pool containing over 500 music tracks. Importantly, each pool has only one correct music track for each video. For the track-level testing, we start by calculating embeddings for all 12 segments of each 120-second video and music track. We then take the average of these 12 embeddings to create a single representative embedding for each video and each music track. Using these averaged embeddings, we evaluate the performance of the music recommendation module. In the first turn, music is suggested based solely on video features, as we assume that the user hasn't provided any specific requirements at this point. In the second turn, we include the user's text prompts and candidate music along with the video features. This setup allows us to evaluate the system's ability to modify its initial recommendations based on the new information. For both turns, we rank music tracks by calculating the cosine similarity between the features of the music in the pool and the input features. We then compute various metrics such as Recall@K for K = 1, 5, 10, and Median Rank. We also measure the "success rate at 10," abbreviated as SR@10. This gauges the percentage of videos for which the correct music track appears in the top 10 recommended list within two turns. Finally, we report the average performance for each of these metrics across all test music pools. To assess the effectiveness of our conversational recommendation system, MuseChat, we develop a strong baseline model with a two-tower structure. This baseline model shares the same encoder model as MuseChat for handling video and the original music track, but it lacks the ability to handle text data. We train both the baseline and MuseChat using the same dataset and loss function. The results are summarized in Table 1. We evaluate the performance of different models under various input conditions. Interestingly, MuseChat, trained on fused features from three different modalities, performs comparably to the baseline when only visual information is given in the first turn. However, when additional modalities are introduced in the second turn, we observe an improvement exceeding 10% across metrics. We do not directly compare our results with those in [35, 42]. The reasons for not directly comparing our results with those in [42] and [35] include lack of access to their models and data splits, differences in data preprocessing, and incompatibility in task requirements—specifically, the model in [35] cannot accommodate candidate music as input.

5.3. Reasoning Evaluation

To underscore the importance of training our sentence generator module with both music embeddings and music titles as inputs, we introduce two baseline models for comparison. The first baseline employs the frozen Vicuna-7B [60] model, which is based on the Llama2-7B [45] architecture. As this model can't process music embeddings, we only present it with the recommended music title. The second baseline utilizes the same architecture as our sentence generator module but takes only music embeddings as input. We employ various common metrics to evaluate the performance of these baseline models and our sentence generator module on simulated conversations. As shown in Table 2, the Vicuna-7B model performs the worst. This is largely because it fails to extract the music name and artist name from the given music video title, thus lacking a comprehensive understanding of the recommended track. Even when this information is explicitly provided, the model struggles to grasp the musicality of the given track, as it was solely trained on text modality. As for the second baseline, while it successfully captures the musical essence of the recommended track due to its training on both music and text modalities, it still falls short. The model can't accurately identify the correct music name and artist name based solely on audio information. In contrast, our sentence generator module, which uses both audio information and music title inputs, outperforms the baselines, demonstrating the efficacy of our approach.

5.4. Qualitative Evaluation

To highlight the versatility and efficacy of our MuseChat system in the context of conversational music recommendations, we conduct a qualitative evaluation. This evaluation aims to demonstrate the system's proficiency in comprehending, recommending, and generating music that is relevant to visual content, user preferences, and audio at-

Trainable	Input Modality	BertScore (f1) ↑	AB Divergence ↓	$\mathcal{L}_2$ Distance ↓	Fisher-Rao Distance ↓
No	Music Title	0.9453	3.93	0.382	2.11
Yes	Music Embeddings	0.9526	2.68	0.279	2.02
Yes	Music Title + Embeddings	**0.9676**	**1.51**	**0.208**	**1.47**

Table 2. Comparison of semantic similarity between output and simulated conversations using various metrics. BERTScore [57] assesses token-level similarity, while AB Divergence, $\mathcal{L}_2$ Distance, and Fisher-Rao Distance are derived based on InfoLM [9].

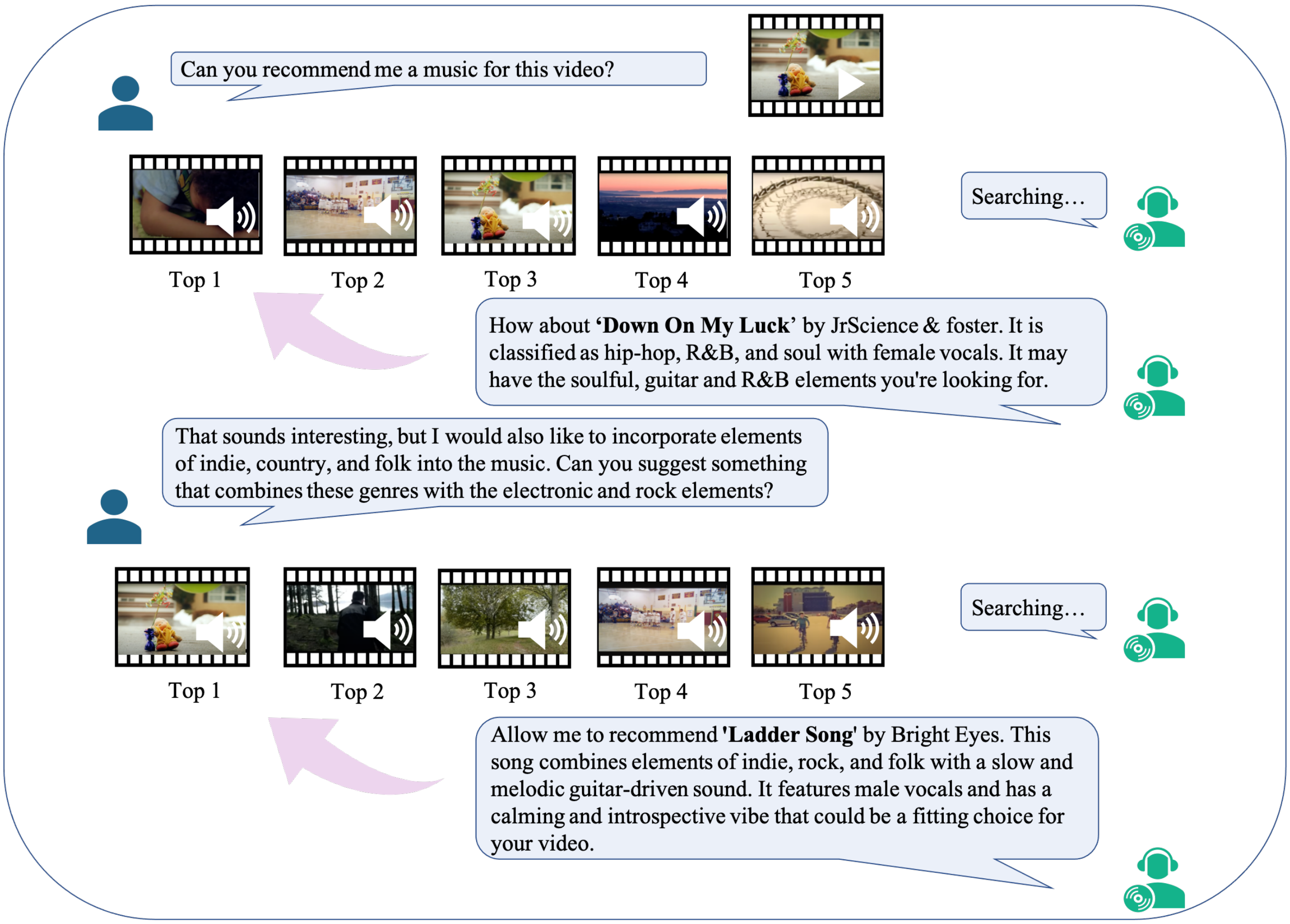

Figure 5. An example when MuseChat retrieves target music in the two turns.

tributes. Figure 5 illustrates how MuseChat seamlessly interacts with users, dynamically adjusting its recommendations based on the video content, user preferences, and contextual information about the music. For a more detailed exploration, additional examples are provided in the supplementary materials.

6. Conclusions

Conventional music recommendation systems primarily focus on delivering personalized suggestions through implicit methodologies, which may not always capture the true preferences of users. In this paper, we take one step toward bridging humans and music recommendation system with more interactions. This paves the way for more accurately tailored recommendation outputs. To achieve this, we (1) create a new dataset based on the public YouTube-8M dataset by simulating two-turn dialogues and (2) propose a baseline model designed to process multi-modal inputs and reason its outputs using free-form natural language. We consider this research a foundational step that opens up new avenues for more holistic, intelligent and responsive recommendation systems. For future work, LLMs could be fine-tuned to handle multi-modal inputs, serving as a more generalized conversational recommendation system that goes beyond the specific domain of music recommendations.

7. Appendix

7.1. Template Prompt

Figure 6 illustrates the template prompt sent to GPT-3.5 for simulating conversations between a user and a recommendation system. Initially, we establish constrained rules to guide the content GPT-3.5 generates. We then supply titles and top 5 music tags from each of two referenced datasets: the MagnaTagATune (MTT) dataset [21] and the Million Song Dataset (MSD) [3]. These tags apply to both example original music and example candidate music. If metadata like official track names, album names, or artist names are available, they are also included in the prompt. Finally, we provide human-written conversation templates featuring example original and candidate music. During generation, we input different pairs of original and candidate music, guiding GPT-3.5 to create new conversations based on the provided human-written examples. Figure 7 shows two examples of simulated conversations based on above prompt.

7.2. Qualitative Results

To demonstrate our model's efficacy, we offer additional qualitative results. Figures 8 and 9 showcase MuseChat's performance in two-turn conversational music recommendation. In each turn, the music recommendation module identifies a list of the top 5 most relevant music tracks based on similarity metrics. To minimize user confusion, the sentence generator module selects only the most relevant track from this list to generate a recommendation text. We also present second-turn recommendation examples in Figures 10 to 13. These examples highlight features that set our model apart from existing video-music recommendation systems.

You are simulating a conversation between a user and a music recommendation system based on the original music and candidate music. You must follow the following rule:

- In the simulated conversation, user must not mention original music name or title.
- User can only mention metadata of the original music.
- Recommender must not use irrelevant descriptions in the recommendation.
- Recommender can use background knowledge to improve the quality of simulated conversation.
- Following is an example

Original music: Title: FREESTYLE "DOMINIQUAIN" 2013. AVEC LA PHAZZ ET REENO. Track name: DOMINIQUAIN. Album name: None. Artists: None. MSD tags: loud, fast, male, rock, male vocal. MTT tags: Hip-Hop, electronic, indie, dance, experimental.

Candidate music: Title: FIFA LB - Xeque-Mate (Feat: MagBoy Rsk) (VideoClip). Track name: None. Album name: None, Artists: None. MSD tags: Hip-Hop, electronic, chillout, ambient, electronica. MTT tags: techno, electronic, beat, synth, vocal.

User: Can you recommend a music for my dance video?

System: Based on your video, I recommend 'Xeque-Mate' by FIFA LB featuring MagBoy Rsk. It's a chillout and ambient track with electronic and techno elements. The song has a relaxing and atmospheric vibe, which could complement the visuals of your video. The vocals and synth in the track add a unique touch to the overall sound.

User: That sounds great, I like the electronic and chillout genre in your music recommendations. However, I would like to incorporate a bit more of a hip-hop style, especially with experimental and indie influences. I'm also interested in fast-paced beats and male vocals.

System: Allow me to recommend 'DOMINIQUAIN' by La Phazz and Reeno. It's a freestyle track that combines elements of hip-hop, electronic, and indie genres. The song has experimental and dance influences, which could add a unique touch to your video. It was released in 2013, and the fast-paced beats and male vocals align with your preferences.

Given the following information, please simulate conversation.

Original music: Title: {title}. Track name: {track name}. Album name: {album name}. Artists: {artist}. MSD tags: {tags}. MTT tags: {tags}.
Candidate music: Title: {title}. Track name: {track name}. Album name: {album name}. Artists: {artist}. MSD tags: {tags}. MTT tags: {tags}.

Figure 6. Template prompt

Example 1

Original music: Title: New TV Wonders - Shine. Track name: Shine. Album name: None. Artists: New TV Wonders. MSD tags: rock, indie, alternative, indie rock, pop. MTT tags: rock, guitar, pop, male, male vocal.

Candidate music: Title: Dengaz - Obrigado (feat. Dino) (Directed By DNG). Track name: Obrigado. Album name: AHYA. Artists: Dengaz. MSD tags: Hip-Hop, rock, pop, electronic, soul. MTT tags: techno, electronic, pop, vocal, fast.

User: Can you recommend a music for my video?

System: Based on your video, I recommend 'Obrigado' by Dengaz. It's a track from the album 'AHYA' and features Dino. This song combines elements of hip-hop, rock, pop, and electronic music with soulful vocals. It has a techno and fast-paced sound too.

User: That sounds great, but I also want to incorporate some rock and indie elements into my video. Do you have any recommendations that align with those genres?

System: Certainly! How about 'Shine' by New TV Wonders? It's a rock and indie track from their album 'The Ostrich Maneuver'. The song has alternative and pop influences as well. It was released in spring 2012 and directed by Hans Frederik Munch.

Example 2

Original music: Title: Caroline & Damon "i knew you were trouble...". Track name: I Knew You Were Trouble. Album name: None. Artists: Madilyn Bailey. MSD tags: female vocalists, pop, folk, rock, indie. MTT tags: female, woman, vocal, guitar, female vocal.

Candidate music: Title: Crown The Empire / Voices (Official Music Video). Track name: None. Album name: None. Artists: None. MSD tags: metal, electronic, rock, experimental, alternative. MTT tags: loud, rock, fast, techno, electronic.

User: Can you recommend a music for my video?

System: Based on your video, I recommend 'Voices' by Crown The Empire. It's a loud and fast-paced track with a mix of electronic and rock elements. The song has an experimental and alternative vibe, which can add intensity and energy to your video.

User: That sounds great! I'm also looking for a song with a female vocal and a pop/rock sound. Do you have any recommendations?

System: Certainly! I suggest 'I Knew You Were Trouble' by Madilyn Bailey. It's a cover of the original song by Taylor Swift and has a pop/rock sound with a female vocal. This track can add a catchy and energetic element to your video while maintaining a similar vibe as the original song you mentioned.

Figure 7. Two examples of simulated conversations

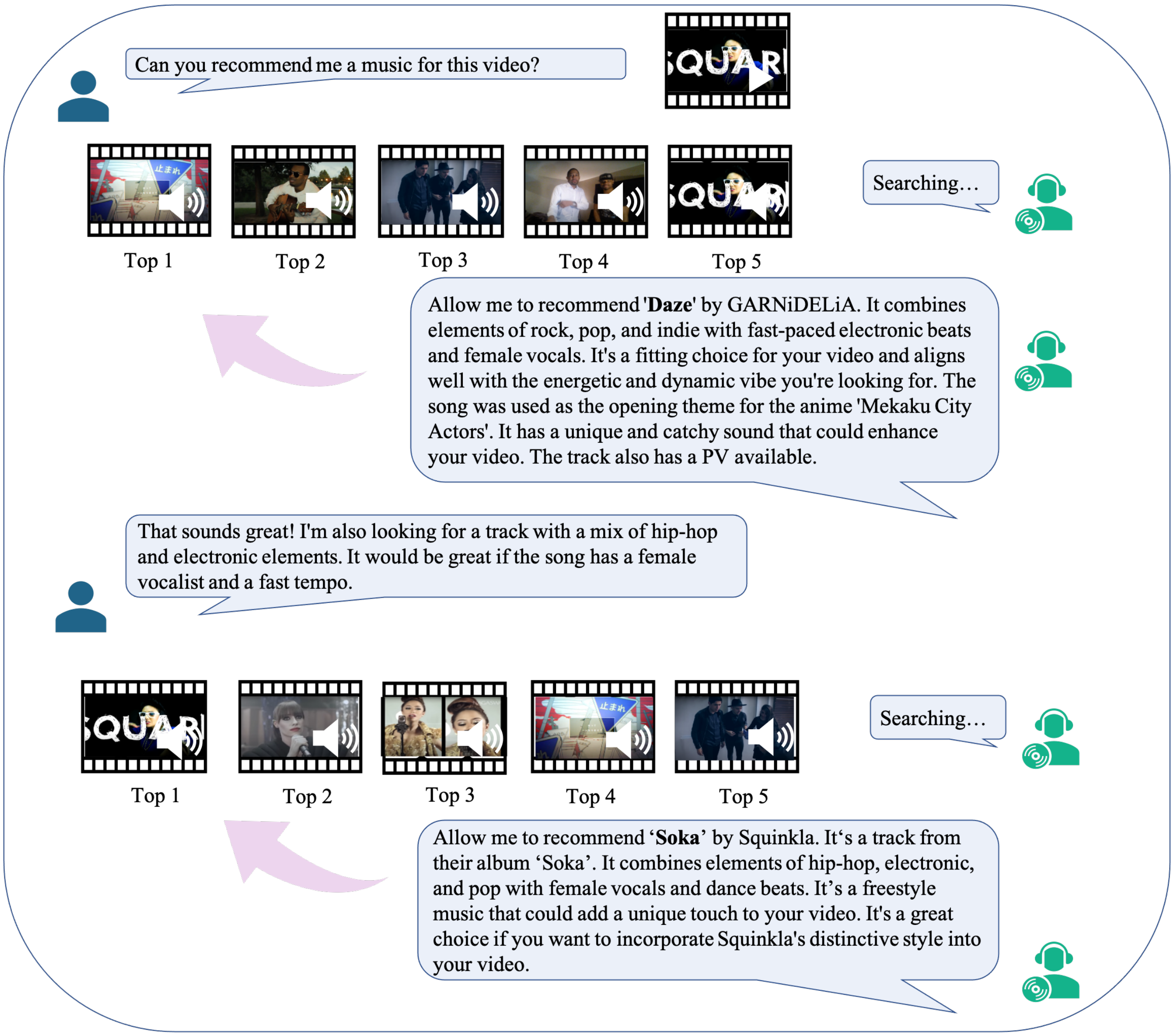

Figure 8. Example showing MuseChat's capability in conversational video-music recommendation.

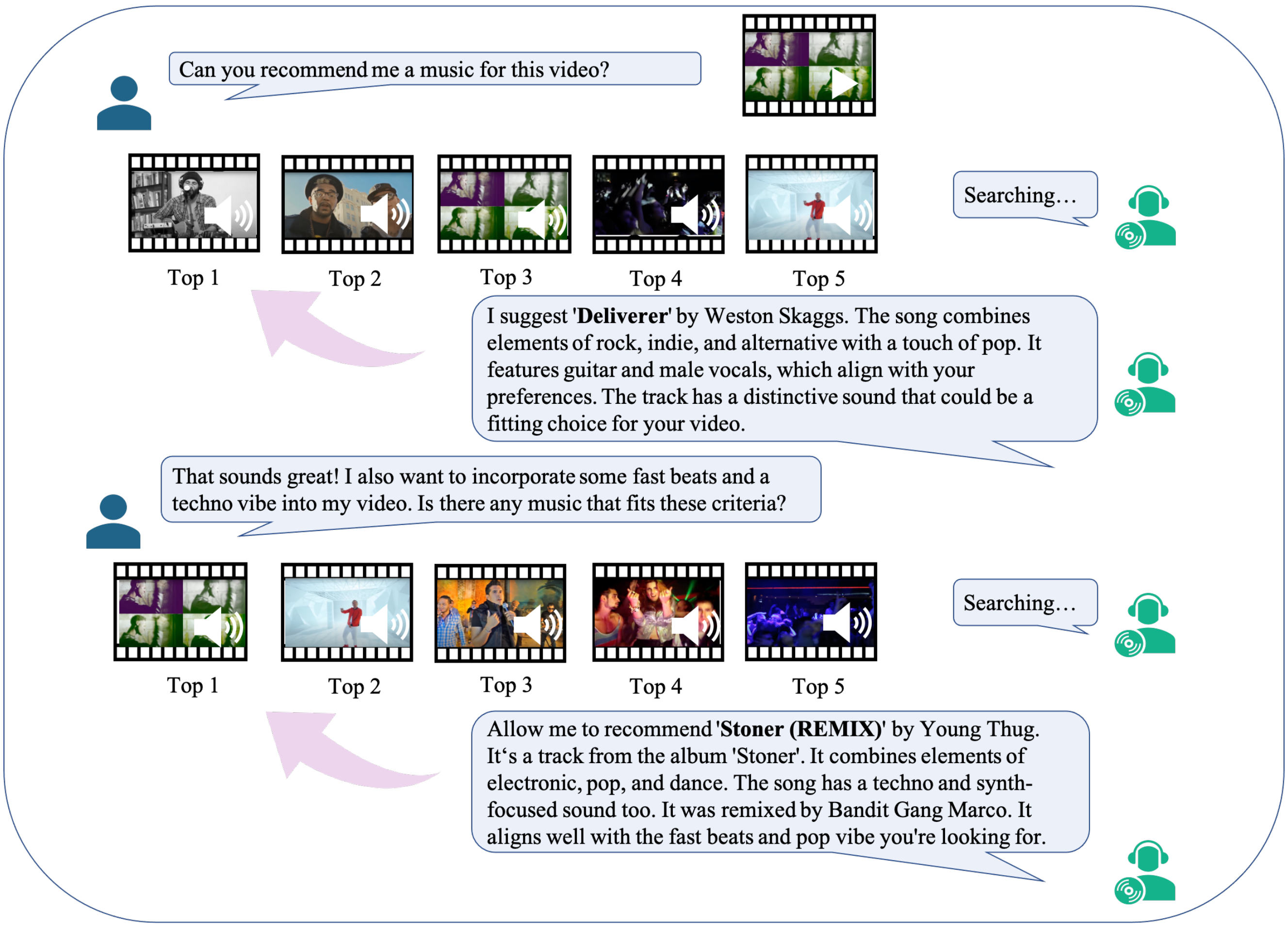

Figure 9. Example showing MuseChat's capability in conversational video-music recommendation.

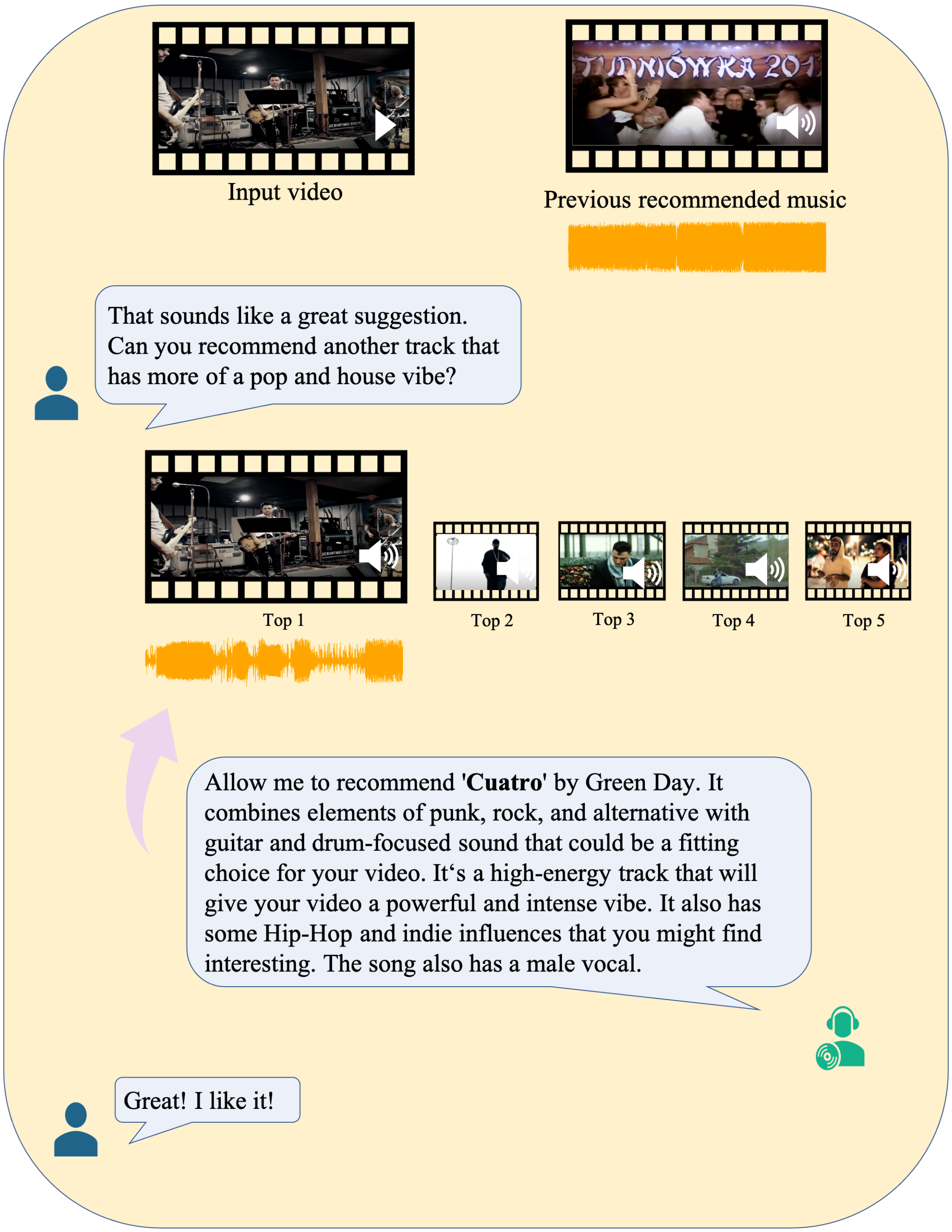

Figure 10. Example showing MuseChat's capability in improving recommendation results by learning user's preferences, contextual music and video content.

Figure 11. Example showing MuseChat's capability in improving recommendation results by learning user's preferences, contextual music and video content.

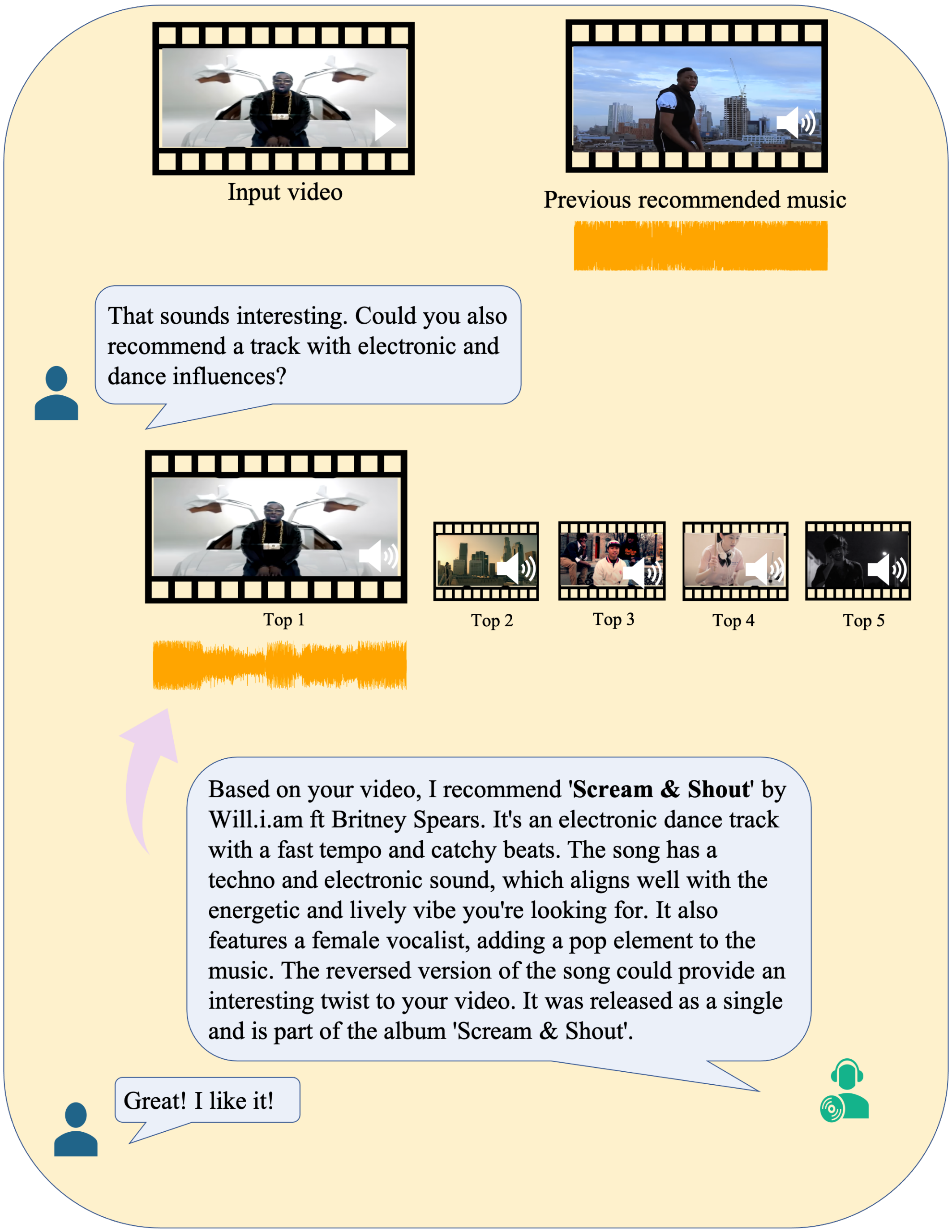

Figure 12. Example showing MuseChat's capability in improving recommendation results by learning user's preferences, contextual music and video content.

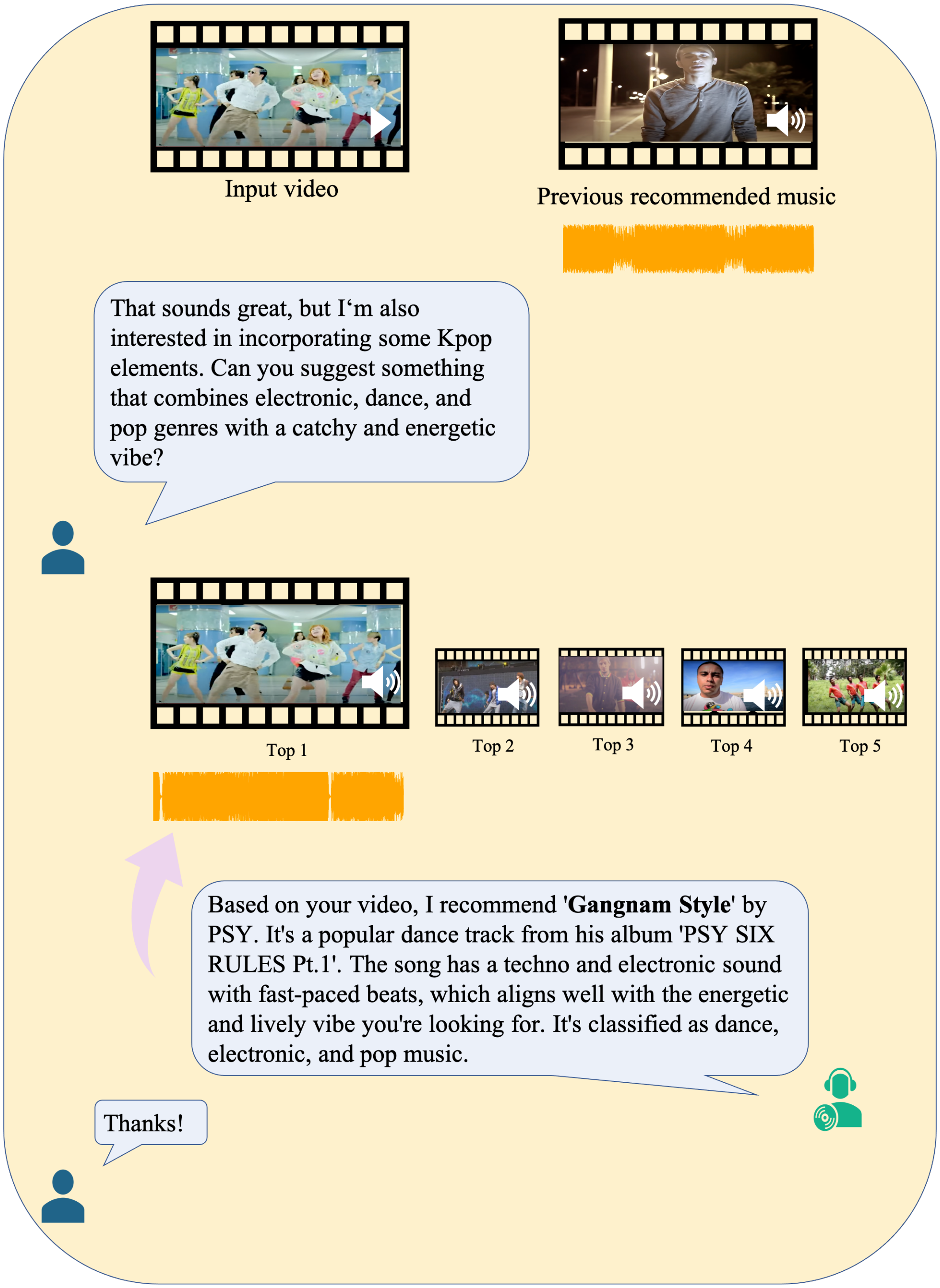

Figure 13. Example showing MuseChat's capability in improving recommendation results by learning user's preferences, contextual music and video content.

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016. 2, 4

[2] Thierry Bertin-Mahieux, Daniel PW Ellis, Brian Whitman, and Paul Lamere. The million song dataset. 2011. 2

[3] Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The million song dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*, 2011. 4, 9

[4] Dmitry Bogdanov, Minz Won, Philip Tovstogan, Alastair Porter, and Xavier Serra. The mtg-jamendo dataset for automatic music tagging. ICML, 2019. 2

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 3

[6] T Chen, S Kornblith, M Norouzi, and GE Hinton. A simple framework for contrastive learning of visual representations. bt-proceedings of the 37th international conference on machine learning, icml 2020, 13-18 july 2020, virtual event.(pp. 1597–1607), 2020. 5

[7] Jeong Choi, Jongpil Lee, Jiyoung Park, and Juhan Nam. Zero-shot learning for audio-based music classification and tagging. *arXiv preprint arXiv:1907.02670*, 2019. 3

[8] Keunwoo Choi, George Fazekas, and Mark Sandler. Automatic tagging using deep convolutional neural networks. *arXiv preprint arXiv:1606.00298*, 2016. 3

[9] Pierre Colombo, Chloe Clavel, and Pablo Piantanida. Infolm: A new metric to evaluate summarization & data2text generation, 2022. 8

[10] Yang Deng, Wenxuan Zhang, Weiwen Xu, Wenqiang Lei, Tat-Seng Chua, and Wai Lam. A unified multi-task learning framework for multi-goal conversational recommender systems. *ACM Transactions on Information Systems*, 41(3):1–25, 2023. 3

[11] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms, 2023. 4

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[13] SeungHeon Doh, Minz Won, Keunwoo Choi, and Juhan Nam. Toward universal text-to-music retrieval. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023. 3

[14] Luke Friedman, Sameer Ahuja, David Allen, Terry Tan, Hakim Sidahmed, Changbo Long, Jun Xie, Gabriel Schubiner, Ajay Patel, Harsh Lara, et al. Leveraging large language models in conversational recommender systems. *arXiv preprint arXiv:2305.07961*, 2023. 3

[15] Chongming Gao, Wenqiang Lei, Xiangnan He, Maarten de Rijke, and Tat-Seng Chua. Advances and challenges in conversational recommender systems: A survey. *AI Open*, 2:100–126, 2021. 3

[16] Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, Hongsheng Li, and Yu Qiao. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*, 2023. 4

[17] Yunfan Gao, Tao Sheng, Youlin Xiang, Yun Xiong, Haofen Wang, and Jiawei Zhang. Chat-rec: Towards interactive and explainable llms-augmented recommender system. *arXiv preprint arXiv:2303.14524*, 2023. 3

[18] Yuan Gong, Yu-An Chung, and James Glass. Ast: Audio spectrogram transformer. *arXiv preprint arXiv:2104.01778*, 2021. 4, 6

[19] Jiaxian Guo, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Boyang Li, Dacheng Tao, and Steven Hoi. From images to textual prompts: Zero-shot visual question answering with frozen large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10867–10877, 2023. 4

[20] Andrey Guzhov, Federico Raue, Jörn Hees, and Andreas Dengel. Audioclip: Extending clip to image, text and audio. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 976–980. IEEE, 2022. 3

[21] Umut Güçlü, Jordy Thielen, Michael Hanke, and Marcel A. J. van Gerven. Brains on beats, 2016. 4, 9

[22] Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. Large language models are zero-shot rankers for recommender systems. *arXiv preprint arXiv:2305.08845*, 2023. 3

[23] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. 4, 6

[24] Tao Hu, Xuyu Xiang, Jiaohua Qin, and Yun Tan. Audio–text retrieval based on contrastive learning and collaborative attention mechanism. *Multimedia Systems*, pages 1–14, 2023. 3

[25] Qingqing Huang, Aren Jansen, Joonseok Lee, Ravi Ganti, Judith Yue Li, and Daniel PW Ellis. Mulan: A joint embedding of music audio and natural language. *arXiv preprint arXiv:2208.12415*, 2022. 3, 4

[26] Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, and Li Chen. A survey on conversational recommender systems. *ACM Computing Surveys (CSUR)*, 54(5):1–36, 2021. 3

[27] Edith Law, Kris West, Michael I Mandel, Mert Bay, and J Stephen Downie. Evaluation of algorithms using games: The case of music tagging. In *ISMIR*, pages 387–392. Citeseer, 2009. 2

[28] Hyodong Lee, Joonseok Lee, Joe Yue-Hei Ng, and Paul Natsev. Large scale video representation learning via relational graph clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6807–6816, 2020. 5

[29] Jongpil Lee, Nicholas J Bryan, Justin Salamon, Zeyu Jin, and Juhan Nam. Disentangled multidimensional metric learning for music similarity. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6–10. IEEE, 2020. 3

[30] Jongpil Lee, Jiyoung Park, Keunhyoung Luke Kim, and Juhan Nam. Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms. *arXiv preprint arXiv:1703.01789*, 2017. 3

[31] Yan-Bo Lin, Yi-Lin Sung, Jie Lei, Mohit Bansal, and Gedas Bertasius. Vision transformers are parameter-efficient audio-visual learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2299–2309, 2023. 4

[32] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 3

[33] Ilaria Manco, Emmanouil Benetos, Elio Quinton, and György Fazekas. Contrastive audio-language learning for music. *arXiv preprint arXiv:2208.12208*, 2022. 3

[34] Ilaria Manco, Benno Weck, Philip Tovstogan, Minz Won, and Dmitry Bogdanov. Song describer: a platform for collecting textual descriptions of music recordings. In *Ismir 2022 Hybrid Conference*, 2022. 3

[35] 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*, pages 14784–14793, 2023. 2, 3, 7

[36] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 5

[37] Sergio Oramas, Oriol Nieto, Francesco Barbieri, and Xavier Serra. Multi-label music genre classification from audio, text, and images using deep features. *arXiv preprint arXiv:1707.04916*, 2017. 2

[38] Jordi Pons and Xavier Serra. musicnn: Pre-trained convolutional neural networks for music audio tagging. *arXiv preprint arXiv:1909.06654*, 2019. 3, 4

[39] Laure Prétet, Gael Richard, and Geoffroy Peeters. Cross-modal music-video recommendation: A study of design choices. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–9. IEEE, 2021. 3

[40] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 4, 6

[41] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018. 3

[42] Dídac Surís, Carl Vondrick, Bryan Russell, and Justin Salamon. It's time for artistic correspondence in music and video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10564–10574, 2022. 3, 7

[43] Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*, 2022. 3

[44] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 776–794. Springer, 2020. 5

[45] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023. 3, 6, 7

[46] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3, 5

[47] Sheng Wang, Zihao Zhao, Xi Ouyang, Qian Wang, and Dinggang Shen. Chatcad: Interactive computer-aided diagnosis on medical image using large language models. *arXiv preprint arXiv:2302.07257*, 2023. 4

[48] Xiaolei Wang, Xinyu Tang, Wayne Xin Zhao, Jingyuan Wang, and Ji-Rong Wen. Rethinking the evaluation for conversational recommendation in the era of large language models. *arXiv preprint arXiv:2305.13112*, 2023. 3

[49] Minz Won, Keunwoo Choi, and Xavier Serra. Semi-supervised music tagging transformer. *arXiv preprint arXiv:2111.13457*, 2021. 3

[50] Minz Won, Andres Ferraro, Dmitry Bogdanov, and Xavier Serra. Evaluation of cnn-based automatic music tagging models. *arXiv preprint arXiv:2006.00751*, 2020. 3

[51] Ho-Hsiang Wu, Prem Seetharaman, Kundan Kumar, and Juan Pablo Bello. Wav2clip: Learning robust audio representations from clip. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4563–4567. IEEE, 2022. 3

[52] Bowen Yang, Cong Han, Yu Li, Lei Zuo, and Zhou Yu. Improving conversational recommendation systems' quality with context-aware item meta information. *arXiv preprint arXiv:2112.08140*, 2021. 3

[53] Jing Yi, Yaochen Zhu, Jiayi Xie, and Zhenzhong Chen. Cross-modal variational auto-encoder for content-based micro-video background music recommendation. *IEEE Transactions on Multimedia*, 2021. 3

[54] Donghuo Zeng, Yi Yu, and Keizo Oyama. Audio-visual embedding for cross-modal music video retrieval through supervised deep cca. In *2018 IEEE International Symposium on Multimedia (ISM)*, pages 143–150. IEEE, 2018. 3

[55] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023. 3

[56] 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. *arXiv preprint arXiv:2303.16199*, 2023. 4

[57] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert, 2020. 8

[58] Hang Zhao, Chen Zhang, Bilei Zhu, Zejun Ma, and Kejun Zhang. S3t: Self-supervised pre-training with swin transformer for music classification. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 606–610. IEEE, 2022. 3

[59] Zihao Zhao, Sheng Wang, Jinchen Gu, Yitao Zhu, Lanzhuju Mei, Zixu Zhuang, Zhiming Cui, Qian Wang, and Dinggang Shen. Chatcad+: Towards a universal and reliable interactive cad using llms. *arXiv preprint arXiv:2305.15964*, 2023. 4

[60] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. 6, 7

[61] Juexiao Zhou, Xiaonan He, Liyuan Sun, Jianan Xu, Xiuying Chen, Yuetan Chu, Longxi Zhou, Xingyu Liao, Bin Zhang, and Xin Gao. Skingpt-4: An interactive dermatology diagnostic system with visual large language model. *medRxiv*, pages 2023–06, 2023. 4

[62] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 3