Language-Guided Music Recommendation for Video via Prompt Analogies

Daniel McKee[1*] Justin Salamon[2] Josef Sivic[2,3] Bryan Russell[2]
[1]University of Illinois at Urbana-Champaign [2]Adobe Research
[3]Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University
dbmckee2@illinois.edu salamon@adobe.com josef.sivic@cvut.cz brussell@adobe.com
https://www.danielbmckee.com/language-guided-music-for-video

Abstract

We propose a method to recommend music for an input video while allowing a user to guide music selection with free-form natural language. A key challenge of this problem setting is that existing music video datasets provide the needed (video, music) training pairs, but lack text descriptions of the music. This work addresses this challenge with the following three contributions. First, we propose a text-synthesis approach that relies on an analogy-based prompting procedure to generate natural language music descriptions from a large-scale language model (BLOOM-176B) given pre-trained music tagger outputs and a small number of human text descriptions. Second, we use these synthesized music descriptions to train a new trimodal model, which fuses text and video input representations to query music samples. For training, we introduce a text dropout regularization mechanism which we show is critical to model performance. Our model design allows for the retrieved music audio to agree with the two input modalities by matching visual style depicted in the video and musical genre, mood, or instrumentation described in the natural language query. Third, to evaluate our approach, we collect a testing dataset for our problem by annotating a subset of 4k clips from the YT8M-MusicVideo dataset with natural language music descriptions which we make publicly available. We show that our approach can match or exceed the performance of prior methods on video-to-music retrieval while significantly improving retrieval accuracy when using text guidance.

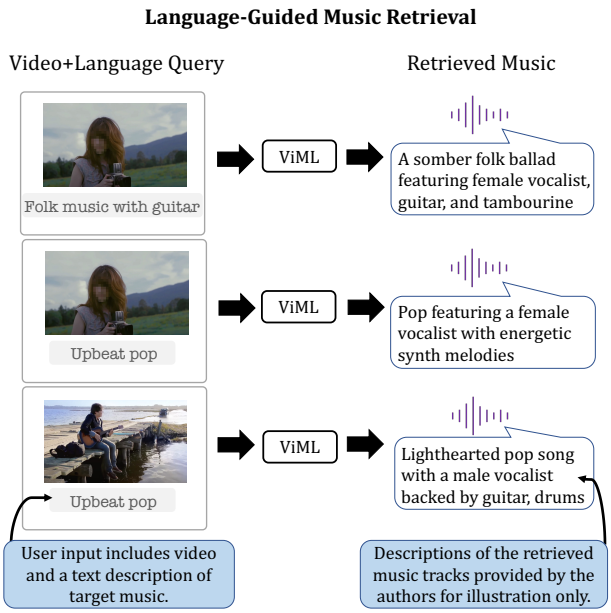

Figure 1. **Language-guided music retrieval.** Our ViML model takes a video and text prompt as input to retrieve a suitable music track from a database. The model learns to fuse video and language representations in order to guide retrieval. Notice how our approach retrieves audio matching both the video and language content. For the same video query (top two rows), we can change the music style to match the language query, and for the same `Upbeat pop` query (bottom two rows) we can change the vocalist to match the video content. **To fully appreciate our results, please view and listen to the companion video on our website**.

1. Introduction

A key part of the video editing process for creators is choosing a musical soundtrack. Especially given the rise of short-form videos on social media platforms, automated music recommendation systems have become an increasingly common and important part of video editing applications. While these systems can be helpful for finding relevant music, they often provide limited capability for user control over the types of music recommended. In previous work, music is retrieved based solely on the visual content and style from a video [32, 37]. However, music itself can convey critical information about how a video should

*Work done as an intern with Adobe Research

be perceived. Music selection alone can transform a visual scene into one that is perceived as happy, scary, or sad[1]. As a result, the lack of user input capability to describe a target music for an inputted video is a key limitation on the utility of current music recommendation methods.

In this work, we propose a more flexible music-for-video recommendation approach that allows a user to guide recommendations towards specific musical attributes including mood, genre, or instrumentation, illustrated in Figure 1. To maximize flexibility and user convenience, we propose to take user musical attribute descriptions in the form of *free-form natural language* (*e.g.*, "Folk music with guitar" in Figure 1). There are two key challenges in learning a model for language-guided music recommendation for video. First, while there are datasets which include music+text [5, 7, 19, 30] or music+video [1], there are no available datasets which include music, video, and text together. Further, the existing datasets that do include text and music focus on a limited vocabulary of tags rather than free-form text. Second, previous works have explored jointly learning visual, audio, and text embeddings [2–4, 34, 47], and without careful regularization, a network can overfit and possibly learn to ignore one of the input modalities. We seek to train a model that keeps the information flow through the network and does not ignore one of the modalities.

In order to meet the challenges outlined above, our work makes the following contributions:

(1) We propose a new approach to automatically generate natural language descriptions for a music video dataset. This approach combines a pre-trained music tagger with a large-scale language model to output natural language descriptions for any music clip, illustrated in Figure 2 (left). First, the tagger predicts tags from a pre-defined vocabulary describing musical genre, mood, or instrumentation. Second, these predicted tags, together with their probabilities, are converted into a rich natural language description for the music video using a carefully designed large-scale language model prompting procedure based on analogies with a small number of human-provided text descriptions (*i.e.*, A (`tags`) : A' (`description`) :: B (`tags`) : B' (`description`)), where A and B are music tags automatically provided by the tagger, A' is a human-provided text description, and B' is the natural language description output by the large-scale language model.

(2) We propose a Transformer-based model architecture with a video-text fusion module. Our model, which we call **Vi**deo to **M**usic with **L**anguage (ViML), is able to retrieve music that matches both the visual content/style of the input video and described musical genre, mood, and instrumentation in the natural language query. Similar to prior work [16, 28, 39], we find that training with text dropout as a regularization mechanism is critical to achieve music retrieval performance improvements from added text inputs.

(3) We release a dataset of 4000 high quality text annotations for clips from a subset of the YT8M-MusicVideo dataset [1] to evaluate language-guided music recommendation. We show that our method can achieve substantial improvements over prior works on music retrieval when incorporating text inputs. Moreover, our model can match or even exceed performance of baseline music-for-video recommendation models when the text input is ignored.

[1] https://www.youtube.com/watch?v=iSkJFs7myn0

2. Related Work

Music and language. There are numerous music tagging datasets which contain tags specifying attributes like mood, genre, or instrumentation [5, 7, 19, 30], and several works have studied training automated music taggers from such datasets [10, 20, 22, 31, 43, 44]. Beyond these methods constrained to limited tag vocabularies, some works also have studied jointly embedding music and free-form natural language [9, 14, 27, 44]. However, none of these approaches incorporate the video modality.

Music recommendation for video. Others have investigated automatic recommendation of music based on style and content of an input video [13, 24, 32, 37, 49]. Prétet et al. [32] build on previous self-supervised methods [13] by incorporating learned audio features instead of handcrafted features. More recently, Surís et al. [37] propose the MVPt model which employs a self-supervised contrastive loss and Transformer [38] architecture to greatly improve the long-range temporal context modeling in order to retrieve suitable music for a given input video. However, none of these approaches incorporate the natural language modality which we focus on in this work.

Video, audio, & language. While a wide variety of works have explored audio-visual or vision-language topics, a smaller number focus on jointly embedding video, audio, and language [2, 4, 12, 34, 45, 47]. Specifically, Alayrac et al. [3] investigate how best to combine audio and video with text representations. The VATT model [2] is a fully end-to-end tri-modal model capable of using a single shared Transformer backbone across modalities. Lastly, two recent methods [12, 45] extend CLIP [33] to jointly embed audio. While relevant, all of these approaches share a common focus on "environmental" or "everyday" sounds rather than music, and they lack the long-range temporal context modeling critical for music recommendation as a result. In addition, none of these works address a downstream problem of using two modalities in combination (video, text) to query results from another (music).

Few-shot language model prompting. Recent large language models have shown significant success at a wide variety of few-shot or zero-shot tasks from those related to read-

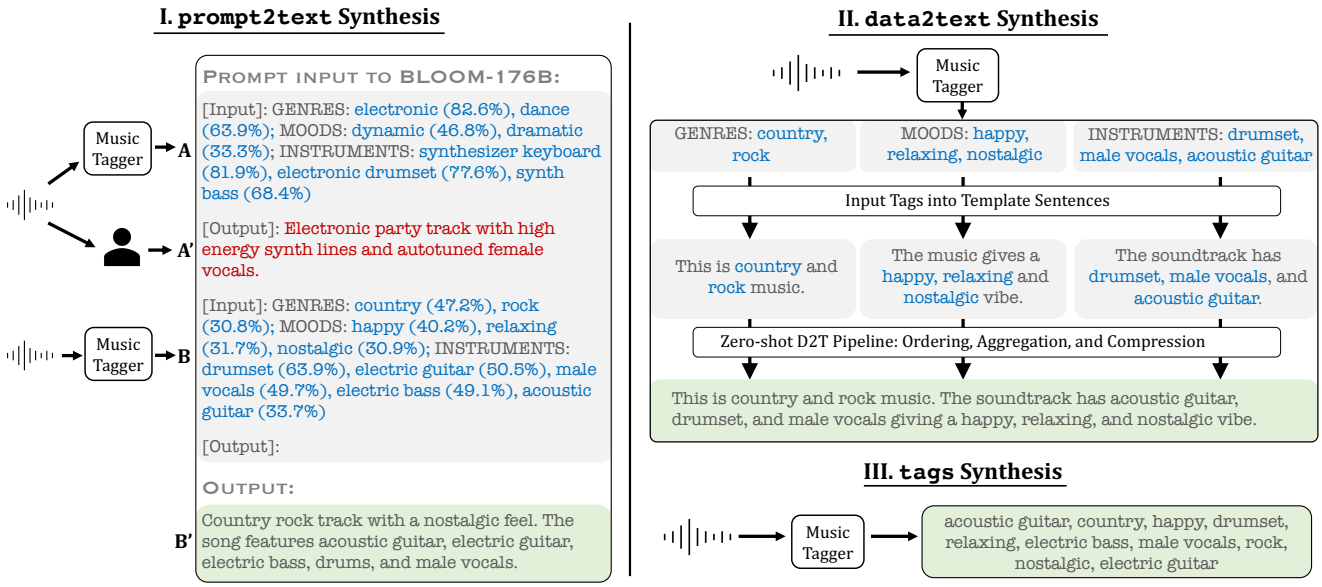

Figure 2. **Overview of three text synthesis approaches explored in our work.** All rely on tag predictions from a pretrained music tagger model. We highlight output text from each method in green, inputs from the tagger in blue font, and inputs from a human annotator in red font. **Left:** We introduce the `prompt2tags` approach for generating natural language descriptions given automatically predicted music tags and a small set of human descriptions. We ask a large language model (BLOOM-176B) to complete an analogy task ($A : A' :: B : B'$) between music tags (A, B) and descriptions (A', B'). **Top right:** The `data2text` pipeline inserts sampled tags into randomly selected template sentences corresponding to each tag category. The Zero-shot D2T model [17] then orders, aggregates, and compresses these templates into a final output description. **Bottom right:** The `tags` approach involves direct concatenation of high confidence tags to form the text description of the music.

ing comprehension and QA [8, 41], to reasoning [18, 42], or even data augmentation [11, 40, 46]. A few works have extended this success to multimodal applications. For example, Zeng et al. [48] show that language models can solve video understanding or image captioning tasks by reformulating these problems as reading comprehension or QA tasks with inputs from large visual or audio models. Other works have used language models to help with generating or retrieving text annotations for multimodal tasks [25] or in robotic planning [15, 35]. In this work, we propose a completely new application of few-shot language query modeling: generating free-form musical text descriptions from music tags.

3. Approach

Our goal is to train a pair of feature encoders f^{vt} and f^{m} which are capable of predicting the similarity $s(f^{vt}, f^{m})$ between an input pair of video and musical text description (v, t) and a music clip m, as illustrated in Figure 3. To train such a model in a supervised manner, it is necessary to have a dataset of corresponding triplets (v, m, t). While large-scale datasets of videos with paired music are available, it is difficult to find datasets which also contain high-quality natural language descriptions of the paired music tracks. As a result, we investigate a synthesis approach based on a model G which generates text descriptions from available structured data in the form of music tags for each music track. In the following sections, we first discuss the musical description synthesis approach G before describing an approach to train a language-guided video-to-music recommendation model.

3.1. Synthesizing Text Descriptions for Music

Suppose that we are given a set of video and music audio pairs (v_i, m_i) and that we also have access to structured data $d_i \in \mathcal{T}^D$ which describe the music m_i. In our case, this structured data consists of musical tags with confidences. Each music track m_i may be described by a free-form human text description $t_i \in \mathcal{T}^T$. However, it can be prohibitively expensive to obtain high-quality human descriptions on a large scale. Instead, we propose to synthesize such text descriptions using a generator $G : \mathcal{T}^D \rightarrow \mathcal{T}^T$ which maps structured data describing an audio track to the space of natural human descriptions.

The goals of such a mapping function G are that: (i) a predicted output $\tilde{t}_i = G(d_i)$ should preserve the semantic meaning contained within the structured data d_i corresponding to a specific musical track and (ii) the distribu-

tion of predicted outputs $\tilde{t}_i$ should follow the distribution of ground truth human text annotations $t_i \in \mathcal{T}^T$. Training a fully supervised model to be the generator function G would require a large quantity of human text descriptions. Instead, we explore zero-shot or few-shot approaches to obtain a generator function G. In particular, we describe three approaches that all use the automatically predicted music tags: a `prompt2text` approach which relies entirely on careful few-shot prompting of a pretrained language model, a zero-shot `data2text` approach which rephrases templated sentences using pretrained language models, and a zero-shot `tags` baseline that represents the music track description directly via the set of automatically obtained tags. Details are given next.

I. Few-shot `prompt2text` approach. We first explore whether the full mapping function G can be encompassed by a single large language model through careful few-shot prompting. This approach relies on a small set of example human-provided descriptions $t_0, ..., t_N$ where $t_i \sim \mathcal{T}^T$. We assume that for each example t_i, we also have a paired structured data output d_i, provided by the automatic music tagger, which describes the same audio track. Unlike prior prompt-based data augmentation works [11, 40, 46] which aim for an unconditional generator G, we aim to generate text data $\tilde{t}_i \sim G(d_i)$ conditioned on structured data d_i such that it follows the distribution of human sentences $\tilde{t}_i \in \mathcal{T}^T$.

As shown in Figure 2 (left), the structured data output d_i is converted to text form via a template, and a set of pairs $(d_0, t_0), ..., (d_k, t_k)$ are used to form k input/output components in the prompt. The final segment of the prompt is the structured data d_i corresponding to a new music track. Given d_i, the model will attempt to output a description t_i following the mapping $\mathcal{T}^D \rightarrow \mathcal{T}^T$ suggested by the example inputs. For text generation in this setting, we use the BLOOM-176B [6] model which is trained on a highly diverse 1.5TB text corpus.

Given that the `prompt2text` allows for the greatest freedom in generation, the model can more easily generate a diverse set of text resembling the target distribution $\mathcal{T}^T$. The `prompt2text` approach is also very flexible as large language models like BLOOM can handle a variety of different structured data inputs such as both tags and their confidence predictions. However, the model may also be less likely to preserve semantic meaning from structured data.

II. Zero-shot `data2text` approach. The second setting that we propose involves a data-to-text generation process which is illustrated in Figure 2 (top right). At a high level, the goal of this method is to insert structured tag data into predefined template sentences and rephrase these template sentences using a language model while preserving original semantic meaning. We begin with the tags predicted for each music track and grouped into genre, mood, and instrument categories. We define a set of category-specific templates in the form of short sentences with placeholders for tags. We randomly sample a template sentence for each category, and fill the template with the high-confidence predicted tags for that category. To form these sentences into more natural free-form descriptions, we make use of pretrained large language models. Specifically, we follow the Zero-shot D2T approach [17], which consists of ordering, aggregation, and compression modules built on pretrained RoBERTa [26] and BART [23] language models. The pipeline components first set the order of the individual filled template sentences and assign which sentences should be combined into a single sentence. Next, the compression module uses a generative text model to rewrite the input sentences based on the ordering and aggregation specifications. The module aims to rephrase the information while preserving semantic meaning. Because this D2T pipeline makes use of models that are pre-trained on large, general text corpuses, we find these modules to perform well at generating music descriptions in a zero-shot manner.

III. `tags` approach. The final setting we use involves a simple concatenation of predicted tags. We take the set of top filtered predicted tags for each music track (this set typically numbers around 10-15 tags total). We then randomly shuffle these tags to prevent model dependence on ordering and concatenate all of the tags into a comma-separated list of musical descriptions (*e.g.*, "synthesizer keyboard, electronic drumset, pop, dance, synth bass, electronic, happy, electric guitar, frantic, dynamic"). While this approach strongly preserves the semantic meaning, it fails to generate text with diverse vocabulary and form which would well represent the human annotation distribution $\mathcal{T}^T$.

3.2. Text Dropout for Music Retrieval Training

The objective here is to retrieve music track m matching a query video v and a natural language query t describing the target music track. This is a challenging task as the model has to fuse together information from both the input video and the input language query to then find a semantically appropriate music track. Moreover, the difference in granularity between audio/video and text can significantly hinder training. We design a tri-modal approach, dubbed ViML, for this task and introduce text dropout to address the granularity issue. In a similar manner to the way dropout prevents overfitting by reducing co-adaptation between individual neurons [36], text dropout serves to avoid overfitting to the text inputs and prevent co-adaptations between the video and text encoders. The approach is illustrated in Figure 3 and details of model architecture, loss, and text dropout are given next.

Model architecture. Our model is trained on a set of (video, music, text) pairings, (v, m, t), corresponding to a

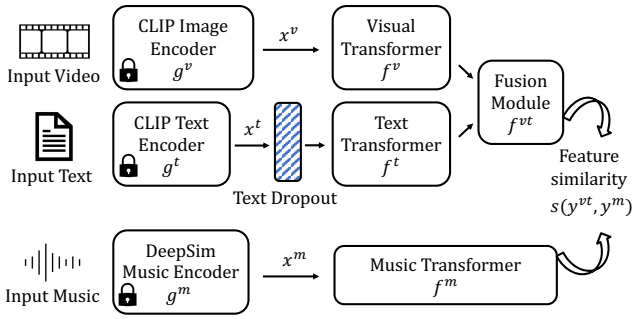

Figure 3. Our proposed ViML model embeds inputs from three modalities (video, text, and audio) into an embedding space. We extract base features using DeepSim [20] for music input and from CLIP [33] for video frames and text descriptions. These base features are inputted to Transformer encoders for each modality. The video and text features are combined with a fusion module to enable querying of music in a shared embedding space. Finally, we employ text dropout to address the difference in granularity between the three modalities. Since video is a more complex input modality, text dropout forces an improved video representation by preventing co-adaptation of the video and text representations.

music video clip v, which has been labeled with a generated text description t of its music track m, as outlined in Section 3.1. We transform these inputs into base features $x^v = g^v(v)$ for visual video features, $x^m = g^m(m)$ for music features, and $x^t = g^t(t)$ for text features using pretrained large-scale encoders g^v, g^m, and g^t which are frozen during training.

Each base feature representation x is of dimension $n \times d$, where n is the length of the temporal sequence of base features representing a video clip and d is the dimension of the base feature. We note that while our model is capable of handling a sequence of temporal text descriptions similar to music or video, we obtain only a track-level text description in practice meaning that $n = 1$ for text.

Our tri-modal model consists of three separate modules corresponding to each modality f^v, f^m, f^t, and a fourth fusion module f^{vt} to combine video and text representations. The modules take respective base features and output embeddings $y^v = f^v(x^v)$, $y^m = f^m(x^m)$, $y^t = f^t(x^t)$. The fusion model outputs a fused embedding from the video and text embeddings $y^{vt} = f^{vt}(y^v, y^t)$.

Fusion loss. For training, we use an InfoNCE loss [29] between music and fused video-text embeddings:

$$\mathcal{L}_{vt\to m} = -\frac{1}{|\mathcal{D}|}\sum_{i\in\mathcal{D}}\frac{\exp(s(y_i^{vt}, y_i^m)/\tau)}{\sum_{j\in\mathcal{D}}\exp\left(s(y_i^{vt}, y_j^m)/\tau\right)} \quad (1)$$

where s is a similarity function, $\mathcal{D}$ is a batch of data, and τ is a temperature hyperparameter we set as $\tau = 0.03$. For our similarity metric, we use the cosine similarity defined as $s(x, y) = x^T y/(\|x\| \cdot \|y\|)$. We note that the loss $\mathcal{L}_{vt\to m}$ is not symmetric as negatives are sampled from music embeddings only. So that our loss is symmetric, we instead train with the summed loss $\mathcal{L}_{m,vt} = \mathcal{L}_{vt\to m} + \mathcal{L}_{m\to vt}$.

Text dropout. To address difficulties posed by the difference in granularity between audio/video and text, we introduce text dropout as a regularization mechanism. With probability p, we set the input text embedding x^t to a specific value x^{NULL}. In practice, we assign this x^{NULL} input as the embedding produced by the pretrained g^t model for an empty string. However, using a zero vector as x^{NULL} works similarly. In addition to improving the performance of music retrieval from text and video together, training with text dropout yields a model which also performs well at retrieval from video alone by removing dependence on text inputs.

3.3. Implementation Details

Music tag generation. The key first step to our text generation process is obtaining the structured data describing each musical track. To do this, we use a music tagger trained on a dataset of music tracks manually annotated with a fixed predefined vocabulary of tags [20, 21]. Specifically, the tagger predicts confidences for 41 instrument tags, 20 genre tags, and 28 mood tags. We aggregate these predictions at the clip or track level, and we filter the subsequent set based on confidence, keeping only those above a particular threshold (0.3 in our experiments).

ViML Model. Following MVPt [37], we employ the Transformer architecture [38] for our music and video encoders f^m and f^v. Transformers play a key role in improving model performance by encoding long-term context from video and music clips. We also use a similar two-layer Transformer architecture for our text encoder f^t and the video-text fusion layer f^{vt}. However, we find that other fusion module architectures such as a single linear layer yield similar results. Please see our supplemental for study of fusion module architectures.

For base features, we use CLIP [33] to encode representations for video frames and text inputs, and DeepSim [20] to encode music. Following communication with the authors of [37], we split the video into 10-second segments and compute a feature for each segment by averaging CLIP embedding features computed at 6 frames per second. We compute 512 dimensional CLIP embedding features using OpenAI's CLIP ViT-B/32 model. We encode all input base features into embeddings of size $d = 256$ using a linear projection layer for each modality. We also select $d = 256$ as the output dimension for encoded video, text, music, and fused video-text representations from our model.

A faint, simple acoustic piece of singing by a female vocalist with an acoustic guitar with a fast-paced strumming pattern in a closed room recorded live. great for singing along.

Hip-hop track with a dark synth pad with male aggressive rapping along with a chipmunk voice.

Instrumental track featuring an ambient pad and bell-like sounds. Seems to be a film score.

A cover version of Elvis Presley's pop song featuring male and female vocals and piano layers. Sounds like a confession in love.

Punjabi track with multiple vocals and other instruments that's great for traditional marriage celebrations.

Figure 4. **Example annotations from our collected YouTube8M-MusicTextClips dataset.** Each example shows a frame from the 10sec source video clip from which audio was extracted for annotation. Note that annotators were only provided *audio* from the music video, so the annotation describes the music, but not the corresponding video. Each example in the figure contains a hyperlink to the corresponding YT8M source video with timestamp at the start of the 10sec target clip. Hover over the video frame image and click to follow the link.

4. Experiments

In this section, we report our experimental settings and results. First, we describe our datasets and the evaluation protocol in Sec. 4.1. Next, we investigate tag-based video-to-music retrieval, comparing against state-of-the-art video-to-music retrieval methods in Sec. 4.2. In Sec. 4.3, we evaluate performance of video-to-music retrieval guided by free-form text annotations. Finally, we perform ablation studies to measure the influence of text dropout in Sec. 4.4.

4.1. Datasets and Evaluation Protocol

YT8M-MusicVideo. In all of our experiments, we train models using the YT8M-MusicVideo dataset which includes around 100k videos with the "music video" tag from the much larger YouTube8M dataset [1]. We synthesize tags and a natural language text describing the music track of each video for the full dataset using the approaches described in Sec. 3.1. We also use the test split of YT8M-MusicVideo to evaluate tag-based retrieval in Sec. 4.2.

YT8M-MusicTextClips. In addition to the full YT8M-MusicVideo dataset, we also annotate a 4,000 sample subset of clips from YT8M-MusicVideo with human-provided text descriptions of the music track accompanying each video. To create these annotations, we sample 10 second audio clips from the middle of each music video, and we ask human annotators to describe the music they hear after listening to the audio clip. Thus, an annotation describes only the *music* from a YT8M sample, and the annotators do not see the corresponding video. Example annotations are shown in Figure 4 with links to the starting timestamp of the 10sec clips in corresponding YouTube videos. This annotated set is meant mainly for evaluation. As a result, the annotations are split into a larger set of 3,000 samples from the test set of YT8M-MusicVideo and a smaller set of 1,000 samples from the train set of YT8M-MusicVideo which we use as examples in the few-shot `prompt2text` synthesis process. We make the annotated text descriptions publicly available at our companion website[2].

Evaluation Set-up and Metrics. We evaluate music retrieval performance consistently with previous works [32, 37]. However, in our case, a query can be either a video alone or a video and corresponding text annotation together. For each query, we compute feature similarity between the query and a pool of N music tracks (we set N=2000 in the track-level setting and N=500 for evaluation on clips). The pool contains a single ground truth music track corresponding to the input query (the positive example) with the remaining music tracks in the pool being non-matching (*i.e.*, negative examples). We rank the music tracks in a query's pool by feature similarity, and find the rank of the query's ground truth matching music track (the positive example). We then compute Recall@K (shortened to R@K) for K=1,5,10 and Median Rank, calculating the average of each of these metrics across the full set of test queries.

4.2. Tag-Based Retrieval

For our first set of experiments, we explore the setting of tag-based retrieval. Here the goal is to retrieve a music track given a query video together with a set of tags from a predefined vocabulary, such as "happy", "piano" and "jazz". This setting could be practically interesting in some applications, *e.g.*, tag-based search. To address this setting, we train our model on text synthesized with the `tags` approach. In these experiments, we train a track-level model and perform retrieval on a track-level in a manner consistent with prior work [32, 37]. To directly compare results with prior work, we perform retrieval on the full YT8M test set consisting of around 10K samples. As shown in Table 1, we include three baselines: the model proposed by Pretét et al. [32], the MVPt model [37], and an improved version of MVPt that we call MVPt+, where we tune the temperature

[2] https://www.danielbmckee.com/language-guided-music-for-video/index.html

Method	Train Text	Query Text Input	Median Rank ↓	R@1 ↑	R@5 ↑	R@10 ↑
a. Pretét et al. [32]	-	-	234	0.76	3.42	5.90
b. MVPt [37]	-	-	13	6.09	24.91	41.89
c. MVPt+ [37]	-	-	5	27.93	50.64	60.68
d. ViML (ours)	`tags`	-	3	29.43	62.49	75.40
e. ViML (ours)	`tags`	`tags`	**2**	**49.49**	**81.61**	**89.41**
f. Chance			1000	0.05	0.25	0.50

Table 1. **Tag-based music retrieval on full YouTube8M-MusicVideo test set.** We compare ViML against prior methods on video to music retrieval without tag queries (row d.). We also evaluate ViML on video+text to music retrieval using (synthetic) tags at test time (row e.). The text descriptions for both training and evaluation are generated with the `tags` approach for these experiments.

Method	Train Text	MR ↓	R@1 ↑	R@5 ↑	R@10 ↑
a. MVPt+	-	17	12.20	29.43	40.46
b. ViML	`tags`	15	11.95	30.34	42.62
c. ViML	`data2text`	13	13.61	33.94	46.24
d. ViML	`prompt2text`	**12**	**14.09**	**35.04**	**47.88**
Chance		250	0.20	1.00	2.00

Table 2. **Music retrieval with free-form natural language on YT8M-MusicTextClips test set.** All methods which take text input are evaluated on the human text annotations as queries. Since the MVPt+ model does not take text inputs, it is evaluated on music retrieval from video alone for the same set of 3k video clips. MR is median rank.

parameter τ in the InfoNCE loss to 0.03. This change leads to further significant improvement in performance.

Next, we introduce our model (ViML) trained on data generated from the `tags` approach. We evaluate our ViML model in two settings. First, we evaluate without input texts at test time (an empty text input is used instead). Second, we evaluate with text inputs at test time. As we do not have track-level human-provided music tag annotations for the full YT8M MusicVideo split, we evaluate the track-level model on synthetically generated tags using our `tags` approach. While a model trained on the `tags` synthesized data may not generalize to out-of-domain free-form text inputs, the tag-based prompting can be a convenient way to guide music retrieval with key desired attributes (for example "female vocalist, guitar, happy"). The tag-based retrieval we report can serve as an upper bound for this type of user tag-guided retrieval since the tag-based text for testing comes from the same music tagger model we used to synthesize training data.

Evaluating our model with synthetic tags leads to a very substantial performance increase over MVPt+ of 20-30 points in each recall metric. Interestingly, our ViML model evaluated without text at test time not only matches the video-to-music retrieval performance of MVT+ but substantially improves over MVPt+, especially in Recall@5 and Recall@10. This performance increase is not simply a result of added parameters in the fusion layer, as a fusion module consisting of only a single linear layer yields similar results (see our supplemental for further details). This result suggests that training jointly with the text domain can lead to improvements in the video and audio representations. We hypothesize that the joint training with language helps to disentangle the video-audio space into semantically meaningful dimensions corresponding to the provided tags as well as helps to suppress non-relevant dimensions, *e.g.*, corresponding to presence/absence of some non-relevant objects.

4.3. Free-Form Natural Language Retrieval

For the next experiments, we turn to retrieval with free-form natural language inputs. The goal is, given an input video and a query free-form natural language description, to retrieve a relevant music track. For this setting, we evaluate on testing videos from the YT8M-MusicTextClips dataset which contains free-form human text annotations describing the music corresponding to each video in the dataset.

In these experiments, we use a similar protocol to the "segment-level" setting reported by Surís et al. [37], but our input video includes only a 30sec clip surrounding the 10sec of audio labeled by a human annotator. In contrast, a model had access to a large context spanning the full source video in the previous segment-level setting reported by Surís et al. [37]. We note that retrieval in this setting is significantly more difficult than the segment-level setting in [37] or the track-level setting reported in 4.2 due to the limited context. However, such retrieval is of particular interest given the rise of short-form video in social media and entertainment.

Results are summarized in Table 2. Our baseline is an MVPt+ model which has been trained on 30sec segments (training MVPt+ on full videos and testing on 30sec clips causes a much more severe drop in performance). We next report music retrieval using video and free-form human text descriptions as input queries to our ViML model. In Table 2, we report three variants trained on YT8M music videos with text synthesized by each of the three approaches described in Sec. 3.1. The model trained with our first `tags` syn-

Method	Train Text	Dropout	Text Inputs	Median Rank ↓	R@1 ↑	R@5 ↑	R@10 ↑
a. MVPt+	-	-	-	17	12.20	29.43	40.46
b. MVT-Fuse	`prompt2text`	✗	-	20	9.94	26.42	37.01
c. MVT-Fuse	`prompt2text`	✗	`human`	15	11.45	30.45	42.77
d. MVT-Fuse	`prompt2text`	✓	-	16	12.27	30.34	41.51
e. MVT-Fuse	`prompt2text`	✓	`human`	**12**	**14.09**	**35.04**	**47.88**

Table 3. **Influence of training with text dropout on retrieval performance.** Evaluated on the YT8M-MusicTextClips test set.

thesis baseline (b.) provides substantial improvement over retrieval with MVPt+ using only video (a.). Next, we evaluate the `data2text` approach (c.) which generates more natural phrases while strictly preserving tag semantics. This approach provides a consistent improvement over the ViML `tags` variant (b.). Finally, our `prompt2text` approach (d.) leads to the best performance showing that large language models prove to be strong annotators on this task with careful few-shot prompting.

Qualitative results. In Figure 5, we provide qualitative retrieval results for examples in YouTube8M-MusicTextClips. In the first example, both models retrieve tracks that match the style and beat of the input video well. However, only the ViML can match the correct musical style by using the input text. In the second example, only the ViML result correctly matches the desired music genre and the mood of the video.

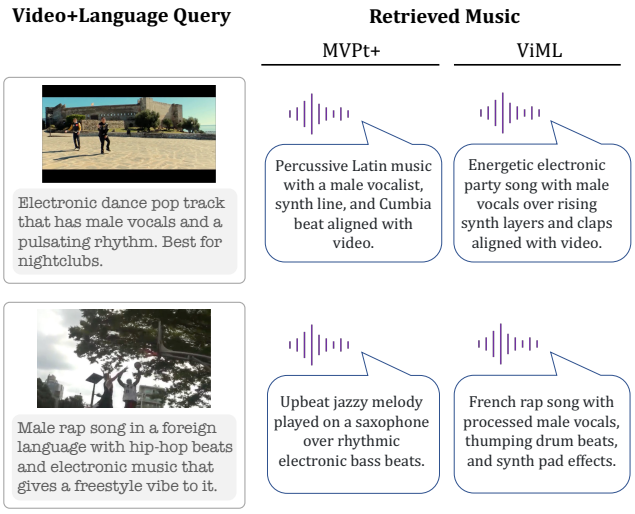

Figure 5. **Qualitative results on YouTube8M-MusicTextClips test set.** We compare music retrieval quality for two examples using the MVPt+ model and our ViML model. The column on the left includes a frame from the input video and the input text description describing the target music. The MVPt+ model takes only the video as an input while the ViML model takes both video and corresponding text. The two columns on the right contain retrieved music for MVPt+ and ViML respectively. **Please see results in the companion video on our website.**

4.4. Analysis of Text Dropout

In Table 3, we compare the performance of our ViML model trained on `prompt2text` descriptions with and without text dropout. We evaluate this model on music retrieval in two settings: (i) using only video (inputting empty text, rows b. and d.) as a query and (ii) using both video and human text descriptions together as a query (rows c. and e.). As expected, adding text dropout during training (d.) improves the performance of retrieval using only video (b.). However, interestingly, text dropout also substantially improves performance when the query includes natural language (e. vs. c.), suggesting that text dropout is a very useful regularization technique in the multimodal setting. We find that without text dropout, training begins to plateau early as the model starts overfitting to the training text inputs. Since video is a much richer and more complex modality, forcing more attention to this modality during training improves learning. We find that the dropout technique is most effective at high rates of dropout in the range 0.8-0.95, and we use a dropout rate of 0.8 in all of our experiments.

5. Conclusion

In this work, we introduced an approach to allow language-guided music recommendation for video. We proposed a model, ViML, which fuses text and video inputs to find music matching both domains and introduced the text dropout technique to improve training. To obtain data for training, we proposed a free-form music description synthesis approach using a large language model (BLOOM-176B) and outputs from a pretrained music tagger. Our results show that large language models provide a powerful tool for training data synthesis in domains where text data is limited but other structured data is available. To evaluate our method, we also introduced a new dataset, YouTube8M-MusicTextClips, which includes high quality free-form human descriptions of the music in YT8M videos. There are many exciting directions to build upon this work including allowing more fine-grained control over specific music attributes or language-guided audio-video generation.

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, 6

[2] Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. *Advances in Neural Information Processing Systems*, 34:24206–24221, 2021. 2

[3] Jean-Baptiste Alayrac, Adria Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. *Advances in Neural Information Processing Systems*, 33:25–37, 2020. 2

[4] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. See, hear, and read: Deep aligned representations. *arXiv preprint arXiv:1706.00932*, 2017. 2

[5] Thierry Bertin-Mahieux, Daniel PW Ellis, Brian Whitman, and Paul Lamere. The million song dataset. In *Proceedings of the 12th International Society for Music Information Retrieval Conference*, 2011. 2

[6] BigScience. Bigscience large open-science open-access multilingual language model, 2022. "https://huggingface.co/bigscience/bloom". 4

[7] 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)*, Long Beach, CA, United States, 2019. 2

[8] 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

[9] Jeong Choi, Jongpil Lee, Jiyoung Park, and Juhan Nam. Zero-shot learning for audio-based music classification and tagging. In *International Society for Music Information Retrieval Conference*, 2019. 2

[10] Keunwoo Choi, George Fazekas, and Mark Sandler. Automatic tagging using deep convolutional neural networks. In *Proceedings of the 12th International Society for Music Information Retrieval Conference*, 2016. 2

[11] Bosheng Ding, Linlin Liu, Lidong Bing, Canasai Kruengkrai, Thien Hai Nguyen, Shafiq Joty, Luo Si, and Chunyan Miao. DAGA: Data augmentation with a generation approach for low-resource tagging tasks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6045–6057, Online, Nov. 2020. Association for Computational Linguistics. 3, 4

[12] 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. 2

[13] Sungeun Hong, Woobin Im, and Hyun Seung Yang. Cbvmr: Content-based video-music retrieval using soft intra-modal structure constraint. *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, 2017. 2

[14] 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. 2

[15] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, et al. Inner monologue: Embodied reasoning through planning with language models. *arXiv preprint arXiv:2207.05608*, 2022. 3

[16] Ahmed Hussen Abdelaziz, Barry-John Theobald, Paul Dixon, Reinhard Knothe, Nicholas Apostoloff, and Sachin Kajareker. Modality dropout for improved performance-driven talking faces. In *Proceedings of the 2020 International Conference on Multimodal Interaction*, pages 378–386, 2020. 2

[17] Zdeněk Kasner and Ondřej Dušek. Neural pipeline for zero-shot data-to-text generation. *arXiv preprint arXiv:2203.16279*, 2022. 3, 4

[18] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*, 2022. 3

[19] 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, 2009. 2

[20] Jongpil Lee, Nicholas J. Bryan, Justin Salamon, Zeyu Jin, and Juhan Nam. Disentangled multidimensional metric learning for music similarity. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020. 2, 5

[21] Jongpil Lee, Nicholas J. Bryan, Justin Salamon, Zeyu Jin, and Juhan Nam. Disentangled multidimensional metric learning for music similarity. *arXiv preprint arXiv:2008.03720*, 2020. 5

[22] Jongpil Lee, Jiyoung Park, Keunhyoung Kim, and Juhan Nam. Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms. 2017. 2

[23] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019. 4

[24] Bochen Li and Aparna Kumar. Query by video: Cross-modal music retrieval. In *ISMIR*, pages 604–611, 2019. 2

[25] Xudong Lin, Fabio Petroni, Gedas Bertasius, Marcus Rohrbach, Shih-Fu Chang, and Lorenzo Torresani. Learning to recognize procedural activities with distant supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13853–13863, 2022. 3

[26] 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. *arXiv preprint arXiv:1907.11692*, 2019. 4

[27] Ilaria Manco, Emmanouil Benetos, Elio Quinton, and György Fazekas. Contrastive audio-language learning for music. *arXiv preprint arXiv:2208.12208*, 2022. 2

[28] Natalia Neverova, Christian Wolf, Graham Taylor, and Florian Nebout. Moddrop: adaptive multi-modal gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(8):1692–1706, 2015. 2

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

[30] Sergio Oramas, Oriol Nieto, Francesco Barbieri, and Xavier Serra. Multi-label music genre classification from audio, text and images using deep features. In *International Society for Music Information Retrieval Conference*, 2017. 2

[31] Jordi Pons and Xavier Serra. musicnn: Pre-trained convolutional neural networks for music audio tagging. *arXiv preprint arXiv:1909.06654*, 2019. 2

[32] 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. 1, 2, 6, 7

[33] 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. 2, 5

[34] Andrew Rouditchenko, Angie Boggust, David Harwath, Brian Chen, Dhiraj Joshi, Samuel Thomas, Kartik Audhkhasi, Hilde Kuehne, Rameswar Panda, Rogerio Feris, et al. Avlnet: Learning audio-visual language representations from instructional videos. *arXiv preprint arXiv:2006.09199*, 2020. 2

[35] Ishika Singh, Valts Blukis, Arsalan Mousavian, Ankit Goyal, Danfei Xu, Jonathan Tremblay, Dieter Fox, Jesse Thomason, and Animesh Garg. Progprompt: Generating situated robot task plans using large language models. *arXiv preprint arXiv:2209.11302*, 2022. 3

[36] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014. 4

[37] 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. 1, 2, 5, 6, 7

[38] 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. 2, 5

[39] Weiyao Wang, Du Tran, and Matt Feiszli. What makes training multi-modal classification networks hard? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12695–12705, 2020. 2

[40] Yufei Wang, Can Xu, Qingfeng Sun, Huang Hu, Chongyang Tao, Xiubo Geng, and Daxin Jiang. PromDA: Prompt-based data augmentation for low-resource NLU tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4242–4255, Dublin, Ireland, May 2022. Association for Computational Linguistics. 3, 4

[41] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021. 3

[42] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc., 2022. 3

[43] Minz Won, Keunwoo Choi, and Xavier Serra. Semi-supervised music tagging transformer. In *Proc. of International Society for Music Information Retrieval*, 2021. 2

[44] Minz Won, Andres Ferraro, Dmitry Bogdanov, and Xavier Serra. Evaluation of cnn-based automatic music tagging models. In *Proc. of 17th Sound and Music Computing*, 2020. 2

[45] 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. 2

[46] Yiben Yang, Chaitanya Malaviya, Jared Fernandez, Swabha Swayamdipta, Ronan Le Bras, Ji-Ping Wang, Chandra Bhagavatula, Yejin Choi, and Doug Downey. Generative data augmentation for commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1008–1025, Online, Nov. 2020. Association for Computational Linguistics. 3, 4

[47] Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi. Merlot reserve: Neural script knowledge through vision and language and sound. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16375–16387, 2022. 2

[48] Andy Zeng, Adrian Wong, Stefan Welker, Krzysztof Choromanski, Federico Tombari, Aveek Purohit, Michael Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, et al. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv preprint arXiv:2204.00598*, 2022. 3

[49] 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. 2