Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI-24)
Special Track on AI, the Arts and Creativity

Retrieval Guided Music Captioning via Multimodal Prefixes

Nikita Srivatsan[1], **Ke Chen**[2], **Shlomo Dubnov**[2] and **Taylor Berg-Kirkpatrick**[2]
[1]Carnegie Mellon University
[2]UC San Diego
nsrivats@cmu.edu, {kec204,sdubnov,tberg}@ucsd.edu

Abstract

In this paper we put forward a new approach to music captioning, the task of automatically generating natural language descriptions for songs. These descriptions are useful both for categorization and analysis, and also from an accessibility standpoint as they form an important component of closed captions for video content. Our method supplements an audio encoding with a retriever, allowing the decoder to condition on multimodal signal both from the audio of the song itself as well as a candidate caption identified by a nearest neighbor system. This lets us retain the advantages of a retrieval based approach while also allowing for the flexibility of a generative one. We evaluate this system on a dataset of 200k music-caption pairs scraped from Audiostock, a royalty-free music platform, and on MusicCaps, a dataset of 5.5k pairs. We demonstrate significant improvements over prior systems across both automatic metrics and human evaluation.

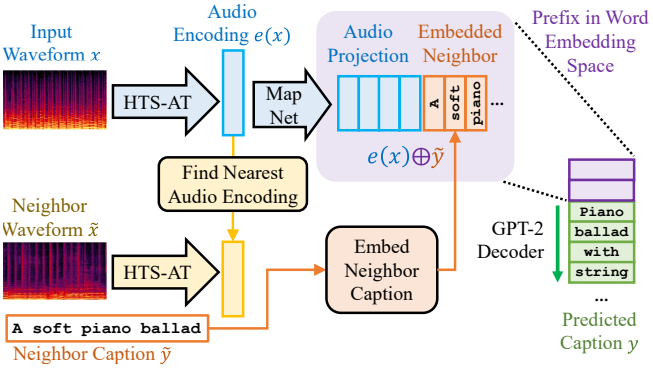

Figure 1: Visualization of RGMC, our music captioning model. We first encode the input song using an HTS-AT audio encoder, and project it through a mapping network to a sequence of word embedding sized vectors. We separately retrieve the song in train with the most similar audio encoding and postpend its caption to the audio prefix. This gets passed to GPT-2 which autoregressively outputs the predicted caption.

1 Introduction

Captioning is an important multimodal task that aims to generate natural language descriptions for data of other modalities. While this task has been well-studied for various domains including images, speech, and video, one domain that is less thoroughly researched by comparison is music. Thorough musical captions span several different aspects of the song – a good description needs to be able to identify tempo, instrumentation, genre, and/or emotional character, while rendering this information in concise natural language.

Being able to produce these captions automatically is important for a variety of reasons. Musical descriptions are helpful for curation and promotional purposes, and also play a meaningful role in *accessibility*. For example, much audiovisual content including streaming media, movies, and television is made accessible to deaf and hard-of-hearing users through the use of closed captions. Beyond simply transcribing speech, these captions must also convey other forms of audio such as background noises and music. Music of course has many attributes that would be apparent to a hearing user, all of which contribute to the emotional content of the scene. Yet even manually written captions frequently fall short in their descriptiveness [Kim, 2020; Davidson, 2022] despite their importance in conveying meaningful information to the viewer [Aleksandrowicz, 2020; Revuelta *et al.*, 2020]. Musical descriptions are also useful for categorization, search, and many other applications where the audio is not renderable or is computationally intractable to directly reason over. Further, there is also growing interest in using text representations in *creative musical applications* – for instance, in systems that can produce and edit music in collaboration with humans using natural language as an intermediary modality [Zhang *et al.*, 2023]. There is therefore reason to want systems that can at large scale produce higher quality captions for music.

Prior research on music captioning as a task has predominantly explored either retrieval or generative methodologies [He *et al.*, 2022a; Manco *et al.*, 2021]. Retrieval-based models are inherently tethered to the distribution in their train set. On the other hand, generative models require large training sets and cannot as easily take advantage of near matches to datapoints previously seen. They also require extensive pretraining and are less adaptable to frequently changing corpora. Therefore, an approach that combines these ideas to

mitigate their respective downsides is desirable.

Unfortunately, building such a system requires large scale and freely available captioning datasets for popular contemporary music, which are currently underutilized in prior work on this task. For example, **Mus**Caps [Manco *et al.*, 2021], a recent step forward in this domain, uses a private dataset of production music with just 6035 audio-caption pairs. In order to both evaluate our method robustly and also encourage future work in this direction, we instead use a public dataset of ∼250k song-caption pairs scraped from the royalty-free music website Audiostock, spanning a wide variety of genres and moods [1]. We also perform further evaluation on Agostinelli *et al.* [2023]'s smaller dataset of 5.5k pairs.

This larger scale enables the modeling strategy that we put forward here, which we call RGMC (**R**etrieval **G**uided **M**usic **C**aptioner). RGMC takes inspiration from prior work on image captioning and contrastive pretraining. Namely, ClipCap [Mokady *et al.*, 2021] previously introduced a strategy in which an image is embedded via a pretrained CLIP encoder network [Radford *et al.*, 2021] and then projected to a vector sequence in word embedding space in order to be conditioned on by a language model decoder, specifically GPT-2 [Radford *et al.*, 2019]. We similarly use a pretrained audio encoder from CLAP [Wu *et al.*, 2023] to build a music feature prefix for our songs. We then further augment this encoding with a retrieved candidate caption from our training set, allowing our decoder to condition on signal from both and either combine them or back off to one if the other is uninformative. By allowing the model to incorporate both audio features and text examples through a multimodal prefix, RGMC can generate captions that are both well-formed and stylistically similar to those in the dataset while also being adaptable to the specific details of each song.

We evaluate this model across several automatic captioning metrics, and demonstrate large improvements over a strong nearest neighbor baseline as well as prior neural work. We also conduct a round of human evaluation which we similarly lead in both measures of fluency and descriptiveness. Additionally, we conduct various ablations which test the brittleness of our approach and measure the relative contributions of the audio and text components.

In summary, this paper makes the following contributions: we (1) Put forward a novel method for captioning music clips based on a retrieval guided generative pipeline (2) Demonstrate improvements in generated caption quality across two datasets in terms of both automatic metrics and human eval.

2 Related Work

Audio captioning more broadly has been relatively well studied, with a variety of published datasets and models for that task [Mei *et al.*, 2022]. However, noticeably less work has gone into studying music captioning specifically, and there are relatively few open datasets and models built with music-to-text applications in mind. At the same time, music captioning as a task differs substantially from general audio captioning. The language of musical description is focused and rich, and therefore musical captions can contain complex detail and analytical insights about tempo, harmony, and timbre.

Music retrieval systems do exist both for captioning and other multimodal tasks. He *et al.* [2022b] and He *et al.* [2022a] take advantage of a retrieval strategy to augment a text decoder, although their method conditions on song lyrics, which many songs (especially those used as background music) do not have. McKee *et al.* [2023] developed a model that retrieves music potentially relevant to a provided video, although they do not tackle captioning. There is also much work on using retrieval-augmentation generation in the context of language models, although this is largely focused on unimodal settings [Lewis *et al.*, 2020; Borgeaud *et al.*, 2021; Khandelwal *et al.*, 2019].

One of the most directly relevant works to our own is **Mus**Caps [Manco *et al.*, 2021] a generative system for music captioning. Despite achieving strong performance and being one of the most prominent models for this task, they use a relatively small scale and private dataset, and do not incorporate retrieval into their pipeline. By scaling up to a larger corpus, we can by contrast take advantage of retrieval strategies. Other related work on generative music captioning has focused on classical music [Kuang *et al.*, 2022], although this limits dataset size and contemporary applicability. There is also recent work on using LLMs for data augmentation for this task by generating pseudo-captions from a taglist [Doh *et al.*, 2023]. While this method operates under an incomparable task setup as it assumes tags are available and does not have a way of conditioning on audio, we do compare to their supervised music-to-text baseline. There has been some similar work on building interactive QA systems capable of understanding audio, speech, and music as well [Deng *et al.*, 2023; Gong *et al.*, 2023b; Gong *et al.*, 2023a].

Prior work has also studied contrastive text-audio representation learning for pretraining purposes. Audio and text representations from CLAP [Wu *et al.*, 2023] have performed well on a variety of downstream tasks, such as audio classification, music genre classification, and sound event detection. MuLan [Huang *et al.*, 2022] is another recent work in this space, and has even been used for pseudo-labeling of music for training diffusion models [Huang *et al.*, 2023]. CALL [Manco *et al.*, 2022] has also seen similar success. This underscores the versatility of jointly learned embeddings and the potential they hold for music captioning.

Our architecture also takes inspiration from image captioning work that similarly takes advantage of contrastively pretrained multimodal encoders. CLIP [Radford *et al.*, 2021] has seen tremendous success across several vision tasks. Its image encoder was utilized by ClipCap [Mokady *et al.*, 2021] which put forward the idea of projecting its visual feature embeddings into word embedding space in order to be conditioned on by a text decoder. Srivatsan *et al.* [2024] further augmented this projection with relevant text in order to create prefixes representing multimodal features for the downstream task of alt-text generation.

There does also exist a recent body of work on the adjacent task of playlist captioning, which aims to assign a description to not just one song but an entire set of them [Choi *et al.*, 2016; Gabbolini *et al.*, 2022; Kim *et al.*, 2023; Doh *et al.*,

[1] https://github.com/LAION-AI/audio-dataset/blob/main/laion-audio-630k

2021]. This requires models to infer similarities across songs instead of prioritizing individual specificity.

Of course beyond the technical component, this task necessarily has meaningful considerations from an accessibility standpoint. Foley and Ferri [2012] write on the delicateness of assistive vs accessible technologies to either improve or diminish access for people with disabilities in a broad sense. Aleksandrowicz [2020] conducted a recent study on the way that deaf and hard-of-hearing people perceive the emotions of film music from captions, motivating the importance of this domain. Revuelta *et al.* [2020] highlighted the difficulties in transmitting sufficient emotional information through traditional captioning alone, with Lucía *et al.* [2020] proposing vibrotactile captioning as an alternative.

3 Model

At a high level, RGMC works by building a prefix in word embedding space and then feeding that as input to a language model decoder which autoregressively continues the sequence to generate the predicted caption. See Figure 1 for a visualization. This prefix consists of two halves: the candidate caption $\tilde{y}$ retrieved by nearest neighbor, and an audio embedding of the song x that has been projected to a sequence of vectors in word embedding space (but does not exactly map to any particular token sequence) which we refer to as $e(x)$. Our aim is that $e(x)$ will capture the specific details of the audio, and $\tilde{y}$ will provide a stylistic template or perhaps include reference to uncommon features not picked up on by the audio encoder, and that by combining these two sources of information our decoder can produce a more accurate caption.

More formally, our modeling goal is to formulate and optimize a distribution $p(y|x, \tilde{y})$ over the caption y conditioned on both the audio of that particular song x as well as a corresponding candidate caption $\tilde{y}$ with respect to our model parameters. We represent x as a 10 second waveform cropped from the middle of the full song with a sample rate of $48\,\text{kHz}$ such that $x \in \mathbb{R}^{480\text{k}}$. We will optimize our model using a dataset $D = \{(x_1, y_1), ..., (x_N, y_N)\}$ consisting of N songs along with corresponding text captions. In this case, our set of candidates is simply our training set excluding that example if it is contained, or $\tilde{D} = D - \{(x, y)\}$. This prevents our model from learning a degenerate solution where it simply copies the nearest candidate and ignores the audio features.

Over the rest of this section we will walk through the three basic steps of our pipeline, namely the encoding and projection of audio features into a conditionable prefix, the retrieval of a nearest neighbor candidate caption, and finally the decoder itself.

3.1 Audio Conditioning

In order for our decoder to be able to condition on audio information from the original waveform, we need to be able to map a feature representation of it to a manifold that our language model can reason over. This process is shown in blue in the upper left portion of Figure 1. Formally, we desire an encoder function $f : \mathbb{R}^{480\text{k}} \to \mathbb{R}^{512}$ which maps these waveforms to a music feature space.

For the purposes of captioning audio data, a cross-modal representation that aligns language and audio features is extremely desirable. Therefore, we specifically parameterize $f(x)$ via the audio encoder from CLAP [Wu *et al.*, 2023], which is itself an HTS-AT [Chen *et al.*, 2022] network followed by an MLP projection layer. While we only directly use the audio encoder, CLAP is explicitly trained to maximize the dot product of corresponding audio and text embeddings (while also incorporating erroneous pairs as negative samples) which makes it an inherently well-suited pretrained model for our purposes.

Having obtained the audio encoding $f(x)$, we use a mapping network to project it to word embedding space. This takes the form of a function $g : \mathbb{R}^{512} \to \mathbb{R}^{10\times d}$ where $d = 768$ is the token dimensionality that our language model expects. We parameterize g with a simple two layer MLP. Together then we can define our prefix $e(x) = g(f(x))$, a text-like sequence that contains audio features from the song.

3.2 Retrieve and Edit

Given however the relatively shallow diversity of song captions, it becomes useful to allow RGMC to condition on a candidate caption suggested by a retrieval system (depicted to the lower left in yellow in Figure 1). This allows the decoder to learn simple edits instead of having to construct the entire caption from scratch. It also biases it towards learning a copy mechanism which it can back off to. We can choose a neighbor $\tilde{x}$ for any song based on the dot product of the audio embeddings $f(x)$. Supposing that for every song in our candidate set we have a corresponding caption $\tilde{y}$, this can be concatenated together with the audio prefix we already have to obtain a multimodal prefix $e(x) \oplus \tilde{y}$.

Of course, the addition of a search system within our pipeline complicates training. Not only is the nearest neighbor retrieval non-differentiable as it involves the discrete selection of a single candidate, it is also computationally expensive to have to execute for each batch during training as the encoder (and therefore the embeddings being compared) will change after each gradient step. In order to circumvent this, we simply do not backprop through the candidate selection process, and instead assume that it is a fixed observation that the model conditions its caption on. That being said, in order to keep the candidate embeddings relatively up to date over the course of training, we recompute them at the beginning of each epoch based on the current parameters of the CLAP encoder. The encoder therefore does not receive explicit learning signal from retrieval, but our hope is that since it does receive loss from captioning it will nonetheless create feature representations that would place songs with similar captions next to each other in its embedding space.

3.3 Decoding

Having computed the multimodal prefix, our GPT-2 [Radford *et al.*, 2019] decoder simply conditions on it as a textual prompt and autoregressively outputs the predicted caption, as shown in green to the right of Figure 1. Putting together our notation from earlier, this yields our desired distribution over the caption:

$$\begin{aligned}\hat{y} &= \arg\max_y \log p(y|x, \tilde{y}) \\ &= \arg\max_y \log p(y|e(x) \oplus \tilde{y}) \\ &= \arg\max_y \log p(y|g(f(x)) \oplus \tilde{y})\end{aligned}$$

At train time, we perform teacher forcing and backprop the cross entropy loss between the logits and the gold reference tokens through the network to obtain gradients for GPT-2, the mapping network, and HTS-AT. At test time, we employ beam search with a beam size of 5, and also disallow any trigram from being generated twice in an output sequence.

4 Implementation Details

We rely on HTS-AT's official pretrained music checkpoint which is trained on a composite collection of music datasets contained within the LAION-Audio-Dataset[2]. For our audio encoder, we use the projected 512-dimensional contrastively trained embedding, which is itself the output of a 1024-dimensional CLAP encoding passed through an MLP layer.

For the GPT-2 [Radford *et al.*, 2019] decoder, we use the publicly available `'gpt2'` checkpoint from HuggingFace.

We use the public implementation of **Mus**Caps[3] which we train from scratch on the Audiostock data due to the unavailability of their dataset and checkpoints, performing preprocessing and tokenization as they describe.

We train our own models with a batch size of 32 and the Adam optimizer [Kingma and Ba, 2015] with a learning rate of $1e-5$. Our implementation runs on a single NVIDIA A6000 GPU in roughly 24-36 hours, and inherits some code from the ClipCap [Mokady *et al.*, 2021] repository. We perform early stopping based on our loss on the held out validation set. Demo and code are available at https://github.com/NikitaSrivatsan/RGMCPublicIJCAI.

5 Dataset

CLAP introduced a 250k dataset [Wu *et al.*, 2023; Chen* *et al.*, 2024] of music tracks and associated metadata scraped from Audiostock[4], which we use for our experiments. These tracks are accompanied by both a short text caption and a long text caption, as well as three tag lists, one describing purpose, one describing impression, and one general. For our purposes we only consider the short text caption, as the long text is inconsistent in format and quality, and the tag lists are not natural language. These tracks have an original sampling rate varying from $32-48$ kHz, which we resample to a fixed 48 kHz. The total duration of the raw dataset is $11\,305$ hours. Since the audio input is processed by CLAP which resamples all audio inputs to 48 kHz to extract the embeddings, our proposed music caption model can support audio inputs with arbitrary sample rates. Following **Mus**Caps [Manco *et al.*, 2021], we standardize the formatting of the captions by casting them to lowercase and removing all punctuation.

One issue with Audiostock as a source of music/caption pairs is the prevalence of duplicate and near-duplicate captions. We frequently found instances of songs with almost identical captions, for example with the only difference being a version number at the end. We therefore needed to remove these redundant examples, both to prevent train/test leakage and also to keep the model from overfitting to those captions. Following Lee *et al.* [2022] we computed the Jaccard similarity between the set of 5-grams within all pairs of captions and marked any that scored above 0.8 as duplicates. We agglomeratively clustered duplicates based on this threshold, and for each cluster retained only the datapoint with the smallest ID number. This ultimately left us with $200\,170$ total datapoints, split into $184\,242$ train, 7925 val, and 8003 test.

In addition to Audiostock, we also perform some experiments on **Music**Caps [Agostinelli *et al.*, 2023], a much smaller scale dataset of 5.5k music-text pairs (not to be confused with our baseline system **Mus**Caps). **Music**Caps specifies a roughly even train/eval split. We additionally split off 487 from the train set to use as dev for hyperparameter tuning and early stopping. It's worth noting that while the captions in **Music**Caps are lengthy and descriptive, they also reflect a high degree of annotator subjectivity [Lee *et al.*, 2023].

6 Experiments

Having described our model, we can now perform captioning experiments to evaluate our method. In this section we will go over the setup of these experiments including the baselines we will compare against, and the metrics we will use to evaluate our output against the gold references.

6.1 Baselines

We compare RGMC against a variety of baselines. These include naive retrieval methods, generative prior work, and ablations of the modalities represented in our prefix.

Random Caption. The first of these is a Random Caption system which just returns a randomly selected caption from the train set. This is effectively a lower bound for this task, and measures to some extent the uniformity of our data.

Nearest Neighbor. We also try a Nearest Neighbor system, identical to the one inside our model's pipeline. It encodes every song in train via a frozen HTS-AT, and when given a song at test time retrieves the one with the highest dot product against its embedding, and returns its corresponding caption verbatim.

MusCaps. Our primary neural baseline is **Mus**Caps. This model differs from ours in a few key ways, most notably in that it is not retrieval guided and uses an LSTM decoder as opposed to a transformer LM. The encoder and decoders of our systems are also pretrained on different data. We retrain **Mus**Caps on the Audiostock dataset using their publicly available implementation.

LP-MusicCaps Supervised. LP-MusicCaps developed a pseudo-captioning system as well as an audio-to-text music captioning model. They listed numbers for their supervised captioning system trained and evaluated exclusively on the **Music**Caps dataset, which we compare to here.

[2]https://github.com/LAION-AI/audio-dataset
[3]https://github.com/ilaria-manco/muscaps
[4]https://audiostock.net/

Song ID	Model	Music Caption
119693	Gold	sad and lonely violin and piano sound
	MusCaps	a magnificent and moving orchestra that feels the beginning of the story
	RGMC	a sad ballad of piano and strings
1702	Gold	cute and nimble fantasy pop
	MusCaps	a heartwarming and cute reggae song
	RGMC	a little comical dark fantasy song
128382	Gold	light and catch a kind of electric jingle
	MusCaps	a song that makes you feel the beginning
	RGMC	danceable beat electro pop
73107	Gold	japanese style hip hop with slow tempo
	MusCaps	a refreshing and bright song with a refreshing acoustic guitar
	RGMC	easy to use japanese style bgm with koto and shakuhachi
58045	Gold	strings dissonance jingle horror
	MusCaps	a magnificent and magnificent song that feels the universe
	RGMC	horror bgm with a sense of tension
1349464	Gold	slightly sad music box with ambient sound
	MusCaps	music box and environmental sounds that match the rain scene
	RGMC	a music box that fits the sad scene with ambient sound

Table 1: Comparison of RGMC's output captions against MusCaps and the original gold reference for various songs in the Audiostock dataset. We generally see that our predictions are more accurate to the genre and other details, and also less likely to repeat adjectives and produce other grammatical mistakes.

Model	Descriptiveness	Fluency
MusCaps	29.0	20.4
RGMC	**63.4**	**49.5**
Equal Quality	7.5	30.1

Table 2: Results from human evaluation, showing annotator preference for RGMC vs. MusCaps by fluency and descriptiveness.

LTU. Gong *et al.* [2023b] and Gong *et al.* [2023a] are QA systems that both have the ability to perform music captioning. Their performance on the MusicCaps dataset was recorded by Deng *et al.* [2023], which we show here.

RGMC (Audio Only). The first of our baselines forgoes the nearest neighbor candidate retrieval, and instead simply feeds the projected audio prefix $e(x)$ directly into the decoder. This lets us measure the relative predictive importance of the audio features in our prefix, and indicates the level of quality achievable in the absence of an on hand candidate set to perform retrieval over.

RGMC (Text Only). We also examine a version that masks out the audio encoding, and only conditions on the candidate. This measures the redundance between the information provided by the neighbor caption and the audio features themselves. Note that this model is still able to indirectly condition on the audio insofar as the retrieved candidate describes it (and it being chosen based on those audio features).

6.2 Metrics

Previous work on music captioning (as well as other forms of captioning) evaluate using standard string similarity metrics that compare the ngram overlap of the model's generated caption with a gold reference. Since we have the user written captions for songs from the uploader, we can treat these as gold and measure these same scores. Specifically, we measure performance on BLEU@4 [Papineni *et al.*, 2002], METEOR [Denkowski and Lavie, 2014], ROUGE-L [Lin and Och, 2004], and CIDEr [Vedantam *et al.*, 2015] using a public fork of the MS COCO [Lin *et al.*, 2014] evaluation repository[5]. We also use two neural evaluation metrics that measure the alignment of our generated captions under a pretrained contrastive model, specifically CLAP [Wu *et al.*, 2023] using the default checkpoint, following a similar approach to Chen* *et al.* [2024]. Specifically we measure the average cosine similarity between CLAP's embedding for the music and our predicted text (which we call CLAP A-T), as well as the similarity between its embedding of our predicted text and the gold caption (which we call CLAP T-T). The former is an especially insightful metric as unlike all the others, it is not measuring the similarity between our output and the subjective human written reference but rather directly measuring the similarity between that output and the original song. We also include the similarity between the gold caption and the audio as a rough proxy for human captioning performance on the former. We find that these metrics qualitatively line up with human judgement of quality for this dataset, and are mostly comparable in range to those reported for existing systems on other datasets for this task.

7 Results

We now discuss the results of our experiments, including automatic captioning metrics, an ablation of both our multimodal prefix and our retrieval method, human evaluation, and a qualitative inspection of our generated descriptions.

[5]https://github.com/LuoweiZhou/coco-caption

Model	CIDEr	BLEU@4	METEOR	ROUGE-L	CLAP A-T	CLAP T-T
Gold Caption	1000	100	100	100	22.01	100
Random Caption	2.81	0.11	1.39	3.26	10.86	29.50
Nearest Neighbor	57.41	**5.93**	8.69	16.79	21.62	44.22
MusCaps	26.98	2.05	6.62	14.22	24.52	42.47
RGMC (Audio Only)	41.63	3.08	8.56	16.78	**27.68**	45.61
RGMC (Text Only)	47.04	3.65	7.72	15.15	24.69	43.63
RGMC	**59.69**	4.96	**9.83**	**19.15**	27.00	**46.36**

Table 3: Results from RGMC on Audiostock as compared to Nearest Neighbor and MusCaps. We additionally provide an ablation of our system that ignores the candidate caption and only conditions on audio, and similarly one that only conditions on the candidate without the audio. Scores are also shown for returning the gold and a randomly selected caption from train. CLAP A-T indicates the CLAP score between audio and predicted text, and CLAP T-T indicates the score between the predicted and gold text.

Candidate	Copy %	CIDEr	BLEU@4	METEOR	ROUGE-L
Random	3.97	35.66	2.70	7.72	15.32
NN	12.66	59.69	4.96	9.83	19.15
Gold	9.76	231.28	22.04	20.25	36.42

Table 4: Comparison of various methods for selecting the candidate caption, including random, nearest neighbor search (default), and returning the gold caption itself. We show scores on the same metrics as well as the Copy %, or the frequency of our decoder returning an identical string to the candidate.

7.1 Human Evaluation

Despite the usefulness and scalability of automatic metrics, the true test of output quality is human judgement. We therefore perform a survey of human annotators comparing the quality of captions generated by RGMC against our neural baseline MusCaps. We randomly selected 31 songs from the Audiostock test set and presented them to a group of three hearing human annotators via an online survey. For each song the annotators were provided a 10 second clip from the middle of the track (i.e. the same input fed to the model's encoder) as well as two candidate captions for that song, one from each model. The order of the models' captions was randomly shuffled and they were not labeled. Annotators were asked which of the two captions was the more accurate description of the provided song, and also which of the two was more fluent in terms of grammar and phrasing. They could answer that the two were too similar to distinguish, but discouraged from doing so frequently.

We report the proportional votes for each model across all annotators in Table 2. We can see that the annotators had a strong preference for our system with respect to descriptiveness, and infrequently selected that the two models were indistinguishably similar. In terms of fluency our model was still preferred by plurality although by a much smaller margin, with annotators reporting that the two were similarly fluent almost a third of the time. This could be explained by the high variability in the fluency of the gold descriptions themselves. We find annotators are in unanimous agreement 35.5% of the time on descriptiveness and 38.7% of the time on fluency (chance would be 11.1%).

7.2 Qualitative Inspection

In Table 1 we show some examples of RGMC's predicted captions as compared to MusCaps and the gold reference. We generally observe that we do slightly better at identifying musical genre and repeat adjectives less frequently than the baseline. We can also see from looking through the gold examples that they tend to vary in which aspect of the music they primarily focus on describing (e.g. instrumentation, mood) and also in the fluency of their style, with some reading closer to a list of tags than a natural language description. This variation in captioning style does of course manifest in the output of the models themselves.

7.3 Automatic Metrics

Table 3 shows results from our system alongside our baselines and ablations for Audiostock. The full RGMC model gets the highest performance on most metrics. Nearest neighbor is a fairly competitive baseline, perhaps to some extent due to dense genre clusters of similar songs present in our data. It's also worth emphasizing that BLEU is a precision driven metric as opposed to for example ROUGE which is based on recall. The fact that nearest neighbor does better on the former but not latter could mean that while it can find similar songs with captions that do not contain incorrect information, it may be less able to produce idiosyncratic details that the gold may mention. This is also supported by its relatively poor performance on CLAP A-T. MusCaps does quite poorly by comparison, even underperforming the audio only ablation of our system. This could be due to its differences in architecture and pretraining. Our ablations also show that a meaningful amount of our system's performance comes from the retrieval component, although it is not sufficiently informative by itself. Some songs will naturally not have a close equivalent in the train set, and the retrieval method itself is imperfect and the caption it retrieves may not fully describe the audio content of the original piece. We also note that many of our systems achieve a higher CLAP A-T score than the gold; this is not entirely unexpected as the gold captions are not necessarily the only correct output for a given song, and there may easily be other captions that are more similar under CLAP's embedding space.

Table 5 shows results of our system and various baselines on the MusicCaps dataset. This dataset has been evaluated

Model	CIDEr	BLEU@4	METEOR	ROUGE-L	CLAP A-T	CLAP T-T
Gold Caption	1000	100	100	100	34.02	100
Random Caption	3.30	3.21	8.91	17.47	10.64	33.19
Nearest Neighbor	8.07	4.49	11.27	19.77	30.97	54.96
MusCaps	1.0	2.1	10.3	19.6	18.79	38.95
LP-MusicCaps Supervised	-	4.79	-	19.22	-	-
LTU	-	-	7.6	8.5	-	-
LTU-AS	-	-	6.0	6.3	-	-
RGMC	8.76	5.90	**11.85**	20.82	30.34	56.06
RGMC (Transfer)	**11.25**	**5.92**	11.83	**21.25**	**32.38**	**57.06**

Table 5: Results from RGMC on the MusicCaps dataset as compared to Nearest Neighbor and MusCaps, as well as reported results from the LP-MusicCaps supervised baseline, and LTU. Scores are also shown for returning the gold and a randomly selected caption from train. CLAP A-T indicates the CLAP score between audio and predicted text, and CLAP T-T indicates the score between the predicted and gold text. RGMC (Transfer) indicates our model pretrained on Audiostock and then finetuned on MusicCaps. All other models except LTU are only trained on MusicCaps, and we only include results from the literature that use the same test set and metric implementations.

on by some prior work which makes it a useful point of comparison. However at the same time, it is substantially smaller, which makes it far less ideal for assessing the performance of a retrieval guided approach such as ours, which benefits substantially from a large scale and diverse candidate pool. Nonetheless, we see that our system achieves the best performance on all metrics compared to the retrained MusCaps as well as Doh *et al.* [2023] and Gong *et al.* [2023b; 2023a]. It's likely that on a small dataset like this, our model is better able to learn to produce fluent output by way of its retrieval mechanism. We also investigate the effects of transfer learning. Specifically we take RGMC trained on Audiostock and then further finetune it on the MusCaps train set, and find that this yields a mild boost in performance.

7.4 Retrieval Method Ablation

In order to measure the downstream effect of the specific retrieved candidate we performed an ablation which replaces the nearest neighbor with a more or less informative alternative. We can use these as a way to artificially strengthen or weaken the retrieval aspect of our pipeline, and examine how much of an effect this has on downstream performance. This tells us how reliant the decoder is on the relevance of the candidate, how robust it is to retrieval mistakes, and how well it can take advantage of in-domain close matches at test time.

Random Candidate. First we replace the candidate with a randomly chosen one from the train set. This lets us see how the model responds when the retrieval system does not recover anything relevant to the particular example.

Gold Candidate. Second we try an oracle that replaces the nearest neighbor with the gold caption. This measures how well the system can take advantage of a "perfect" retrieval that returns the most possibly relevant caption.

We also list the frequency at which the model's generated caption exactly matches the retrieved candidate. This lets us measure to what extent the decoder is simply regurgitating the retrieved candidate as opposed to synthesizing it with the audio signal and producing something novel.

Results on Audiostock are shown in Table 4. We see that random does worst, followed by nearest neighbor, and then the gold oracle. Furthermore, we find that gold and especially random tend to copy the candidate exactly less frequently. This could indicate that when retrieval fails, the model is able to recognize that and back off to only conditioning on the audio, and avoid being distracted by the confounding signal from the "neighbor" caption. Similarly when it produces something desirable, the model is able to take advantage of that and enjoy a large boost in performance, despite not exactly duplicating it more frequently than it would otherwise. This shows that the quality of search has a large effect on downstream captioning, and there may be headroom left here.

8 Conclusion

In this paper we put forward RGMC, a novel method for music captioning that combines retrieval and generative strategies. We trained and evaluated this system on a large scale dataset of music-caption pairs scraped from Audiostock, and demonstrated considerable improvements over prior approaches both in terms of performance on quantitative automatic metrics, and also human evaluation. This model could be useful for curation purposes, accessibility, and has the potential to play a strong role in human-in-the-loop systems for music creation and editing. There is however still significant headroom left on this task, and many avenues for future work exist that may continue to close it. The results from our ablation indicate that we may see further gains from improving the retrieval method, or perhaps even allowing the decoder to condition on a top-k list as opposed to simply the closest individual caption. The datasets also contain other sources of information that we have not yet utilized. For example, jointly training a predictive head on our encoder over the tag lists, or even using the short captions to pretrain learning generation of the longer ones may be fruitful. It may also be worth explicitly biasing the model towards generating interpretable prefixes, as one of the advantages of our approach is that it implicitly learns pseudo-textual audio embeddings. Finally, in order to make strong claims about the usefulness of our system from an accessibility perspective we would need to conduct an evaluation using deaf or hard-of-hearing annotators, which was unfortunately beyond the scope of this work.

Ethical Statement

If this work were simply deployed as a substitute for human captioning, it may lead to a worse experience for those who rely on captions for accessibility. The model may be biased towards genres that are overrepresented within its train data, and may not generalize well to other styles. It is also likely to mirror problematic descriptions that users wrote; we do observe usage of Eurocentric phrases like "oriental" and "exotic" within the dataset. The model also has the potential to misgender vocalists. The copyright implications of training on these datasets are currently ambiguous.

References

[Agostinelli *et al.*, 2023] Andrea Agostinelli, Timo I. Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, Matthew Sharifi, Neil Zeghidour, and Christian Havnø Frank. Musiclm: Generating music from text. *ArXiv preprint 2301.11325*, 2023.

[Aleksandrowicz, 2020] Paweł Aleksandrowicz. Can subtitles for the deaf and hard-of-hearing convey the emotions of film music? a reception study. *Perspectives*, 28(1):58–72, 2020.

[Borgeaud *et al.*, 2021] Sebastian Borgeaud, Arthur Mensch, and Jordan Hoffmann et al. Improving language models by retrieving from trillions of tokens. In *Proceedings of International Conference on Machine Learning (ICML)*, 2021.

[Chen *et al.*, 2022] Ke Chen, Xingjian Du, Bilei Zhu, Zejun Ma, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Hts-at: A hierarchical token-semantic audio transformer for sound classification and detection. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 646–650, 2022.

[Chen* *et al.*, 2024] Ke Chen*, Yusong Wu*, Haohe Liu*, Marianna Nezhurina, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Musicldm: Enhancing novelty in text-to-music generation using beat-synchronous mixup strategies. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024.

[Choi *et al.*, 2016] Keunwoo Choi, György Fazekas, and Mark B. Sandler. Towards music captioning: Generating music playlist descriptions. *ArXiv preprint 1608.04868*, 2016.

[Davidson, 2022] Michael Davidson. 7. a captioned life. In *Distressing Language*, pages 157–182. New York University Press, 2022.

[Deng *et al.*, 2023] Zihao Deng, Yi Ma, Yudong Liu, Rongchen Guo, Ge Zhang, Wenhu Chen, Wenhao Huang, and Emmanouil Benetos. Musilingo: Bridging music and text with pre-trained language models for music captioning and query response. *ArXiv preprint 2309.08730*, 2023.

[Denkowski and Lavie, 2014] Michael Denkowski and Alon Lavie. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Workshop on Statistical Machine Translation*, pages 376–380, 2014.

[Doh *et al.*, 2021] Seungheon Doh, Junwon Lee, and Juhan Nam. Music playlist title generation: A machine-translation approach. In *Proceedings of the Workshop on NLP for Music and Spoken Audio (NLP4MusA)*, 2021.

[Doh *et al.*, 2023] Seungheon Doh, Keunwoo Choi, Jongpil Lee, and Juhan Nam. Lp-musiccaps: Llm-based pseudo music captioning. In *Proceedings of International Society for Music Information Retrieval Conference (ISMIR)*, 2023.

[Foley and Ferri, 2012] Alan Foley and Beth A Ferri. Technology for people, not disabilities: Ensuring access and inclusion. *Journal of Research in Special Educational Needs*, 12(4):192–200, 2012.

[Gabbolini *et al.*, 2022] Giovanni Gabbolini, Romain Hennequin, and Elena Epure. Data-efficient playlist captioning with musical and linguistic knowledge. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022.

[Gong *et al.*, 2023a] Yuan Gong, Alexander H Liu, Hongyin Luo, Leonid Karlinsky, and James Glass. Joint audio and speech understanding. In *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2023.

[Gong *et al.*, 2023b] Yuan Gong, Hongyin Luo, Alexander H Liu, Leonid Karlinsky, and James Glass. Listen, think, and understand. *ArXiv preprint 2305.10790*, 2023.

[He *et al.*, 2022a] Zihao He, Weituo Hao, Weiyi Lu, Changyou Chen, Kristina Lerman, and Xuchen Song. Alcap: Alignment-augmented music captioner. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022.

[He *et al.*, 2022b] Zihao He, Weituo Hao, and Xuchen Song. Recap: Retrieval augmented music captioner. *ArXiv preprint 2212.10901*, 2022.

[Huang *et al.*, 2022] Qingqing Huang, Aren Jansen, Joonseok Lee, Ravi Ganti, Judith Yue Li, and Daniel P. W. Ellis. Mulan: A joint embedding of music audio and natural language. In *Proceedings of International Society for Music Information Retrieval Conference (ISMIR)*, 2022.

[Huang *et al.*, 2023] Qingqing Huang, Daniel S. Park, Tao Wang, Timo I. Denk, Andy Ly, Nanxin Chen, Zhengdong Zhang, Zhishuai Zhang, Jiahui Yu, Christian Havnø Frank, Jesse Engel, Quoc V. Le, William Chan, and Weixiang Han. Noise2music: Text-conditioned music generation with diffusion models. *ArXiv preprint 2302.03917*, 2023.

[Khandelwal *et al.*, 2019] Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. *ArXiv preprint 1911.00172*, 2019.

[Kim *et al.*, 2023] Haven Kim, Seungheon Doh, Junwon Lee, and Juhan Nam. Music playlist title generation using artist information. *ArXiv preprint 2301.08145*, 2023.

[Kim, 2020] Christine Sun Kim. [closer captions]. https://www.youtube.com/watch?v=tfe479qL8hg, 2020.

[Kingma and Ba, 2015] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *Proceedings of International Conference on Learning Representations (ICLR)*, 2015.

[Kuang *et al.*, 2022] Zhihuan Kuang, Shi Zong, Jianbing Zhang, Jiajun Chen, and Hongfu Liu. Music-to-text synaesthesia: Generating descriptive text from music recordings. *ArXiv preprint 2210.00434*, 2022.

[Lee *et al.*, 2022] Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL))*, pages 8424–8445, 2022.

[Lee *et al.*, 2023] Minhee Lee, Seungheon Doh, and Dasaem Jeong. Annotator subjectivity in the musiccaps dataset. In *Proceedings if the HCMIR Workshop at International Society for Music Information Retrieval Conference (ISMIR)*, 2023.

[Lewis *et al.*, 2020] Patrick Lewis, Ethan Perez, Aleksandara Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. *ArXiv preprint 2005.11401*, 2020.

[Lin and Och, 2004] Chin-Yew Lin and Franz Josef Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 605–612, 2004.

[Lin *et al.*, 2014] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of European Conference on Computer Vision (ECCV)*, pages 740–755, 2014.

[Lucía *et al.*, 2020] María J Lucía, Pablo Revuelta, Álvaro García, Belén Ruiz, Ricardo Vergaz, Víctor Cerdán, and Tomás Ortiz. Vibrotactile captioning of musical effects in audio-visual media as an alternative for deaf and hard of hearing people: An eeg study. *IEEE Access*, 8:190873–190881, 2020.

[Manco *et al.*, 2021] Ilaria Manco, Emmanouil Benetos, Elio Quinton, and György Fazekas. Muscaps: Generating captions for music audio. *Proceedings of International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2021.

[Manco *et al.*, 2022] Ilaria Manco, Emmanouil Benetos, Elio Quinton, and György Fazekas. Contrastive audio-language learning for music. In *Proceedings of International Society for Music Information Retrieval Conference (ISMIR)*, 2022.

[McKee *et al.*, 2023] Daniel McKee, Justin Salamon, Josef Sivic, and Bryan Russell. Language-guided music recommendation for video via prompt analogies. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14784–14793, 2023.

[Mei *et al.*, 2022] Xinhao Mei, Xubo Liu, MarkD . Plumbley, and Wenwu Wang. Automated audio captioning: An overview of recent progress and new challenges. *EURASIP Journal on Audio, Speech, and Music Processing*, pages 1–18, 2022.

[Mokady *et al.*, 2021] Ron Mokady, Amir Hertz, and Amit H Bermano. Clipcap: Clip prefix for image captioning. *ArXiv preprint 2111.09734*, 2021.

[Papineni *et al.*, 2002] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318, 2002.

[Radford *et al.*, 2019] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[Radford *et al.*, 2021] 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 *Proceedings of International Conference on Machine Learning (ICML)*, pages 8748–8763, 2021.

[Revuelta *et al.*, 2020] Pablo Revuelta, Tomás Ortiz, María J Lucía, Belén Ruiz, and José Manuel Sánchez-Pena. Limitations of standard accessible captioning of sounds and music for deaf and hard of hearing people: An eeg study. *Frontiers in integrative neuroscience*, 14:1, 2020.

[Srivatsan *et al.*, 2024] Nikita Srivatsan, Sofia Samaniego, Omar U. Florez, and Taylor Berg-Kirkpatrick. Alt-text with context: Improving accessibility for images on twitter. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2024.

[Vedantam *et al.*, 2015] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4566–4575, 2015.

[Wu *et al.*, 2023] Yusong* Wu, Ke* Chen, Tianyu* Zhang, Yuchen* Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1–5, 2023.

[Zhang *et al.*, 2023] Yixiao Zhang, Akira Maezawa, Gus G. Xia, Kazuhiko Yamamoto, and Simon Dixon. Loop copilot: Conducting ai ensembles for music generation and iterative editing. *ArXiv preprint 2310.12404*, 2023.