Bootstrapping Language-Audio Pre-training for Music Captioning

Luca A. Lanzendörfer* *ETH Zurich* lanzendoerfer@ethz.ch
Constantin Pinkl* *ETH Zurich* cpinkl@ethz.ch
Nathanaël Perraudin *Swiss Data Science Center* nathanael.perraudin@sdsc.ethz.ch
Roger Wattenhofer *ETH Zurich* wattenhofer@ethz.ch

***Abstract*—We introduce BLAP, a model capable of generating high-quality captions for music. BLAP leverages a fine-tuned CLAP audio encoder and a pre-trained Flan-T5 large language model. To achieve effective cross-modal alignment between music and language, BLAP utilizes a Querying Transformer, allowing us to obtain state-of-the-art performance using 6x less data compared to previous models. This is a critical consideration given the scarcity of descriptive music data and the subjective nature of music interpretation. We provide qualitative examples demonstrating BLAP's ability to produce realistic captions for music, and perform a quantitative evaluation on three datasets. BLAP achieves a relative improvement on FENSE compared to previous models of 3.5%, 6.5%, and 7.5% on the MusicCaps, Song Describer, and YouTube8m-MTC datasets, respectively. The codebase is available at https://github.com/ETH-DISCO/blap.**

***Index Terms*—Music Captioning, Language Models, Contrastive Language-Audio Pre-training**

I. Introduction

The field of music captioning, a specialized subset of general audio captioning, presents unique challenges in generating natural language descriptions for music. Current generative models, which can generate various modalities from textual prompts, underscore the importance of having datasets that pair these modalities with corresponding textual annotations. As an example, the field of image generation has thrived, in part, due to the availability of images accompanied by descriptive captions. However, such captions are not commonly found in the music domain, and only a few such annotated datasets exist. Additionally, these text-music datasets contain only a fraction of data compared to their text-image dataset counterparts.

Despite the advancements of LLMs in processing textual and visual data as part of vision-language pre-training [1]–[3], their integration with the audio domain, particularly music, remains an open challenge [4]–[6]. This is in part due to the scarcity of audio data with descriptive captions, especially in music, where subjective interpretation plays a significant role alongside objective elements such as instruments or keys.

Addressing this gap, our work introduces BLAP (Bootstrapping Language-Audio Pre-training), a novel music captioning model utilizing a pre-trained CLAP [7] audio encoder and a pre-trained Flan-T5 [8] language model.

A significant challenge in audio-language pre-training, much like in vision-language pre-training, is achieving effective cross-modal alignment. In the case of BLAP, this involves aligning musical elements with appropriate linguistic descriptions. Given that LLMs are not exposed to raw audio data in their initial training, creating a coherent bridge between audio and language is essential. Our approach, inspired by the methodologies in vision-language pre-training [2], leverages a Querying Transformer (Q-Former) to facilitate this alignment, creating an intermediate representation suitable for music data. Utilizing a Q-Former as a knowledge transfer model between music and language modalities, reduces computational demands by bootstrapping a pre-trained audio encoder and an LLM.

Our contributions can be summarized as follows:

- We introduce BLAP, a new pre-trained language-audio model that bootstraps a pre-trained audio encoder and an LLM to generate high-quality captions for music using 6x less samples compared to previous state-of-the-art.
- We perform a qualitative and quantitative evaluation on several metrics and three different datasets, and demonstrate that BLAP outperforms previous state-of-the-art models.
- We open-source the code and model weights in order to contribute to the broader accessibility and advancement of the music captioning field.

II. Related Work

Audio and Music Captioning has received increased interest in recent years, where Pengi [4] was a notable contribution to the field. Pengi was trained on a composite of various audio datasets, consisting of approximately 3.4 million audio-text pairs. MusCaps [9] was one of the first approaches to focus on music captioning, using a model that combines convolutional and recurrent neural network architectures to process audio-text inputs. The music captioning model of LP-MusicCaps [5] follows a similar approach to Pengi, but fine-tunes the model on their own augmented MSD [10] dataset, consisting of 445k samples. Their model differs from our approach, as they do not employ intermediate representation but directly forward the music features to the LLM. Salmonn [6] is a multimodal LLM designed to process and understand general audio inputs, including speech, audio events, and music. Salmonn was trained on 2.3 million samples, of which 53k are music clips. Llark [11] integrates a music feature extractor with a pre-trained LlaMa model [12]. This approach is designed to handle

*Equal contribution.

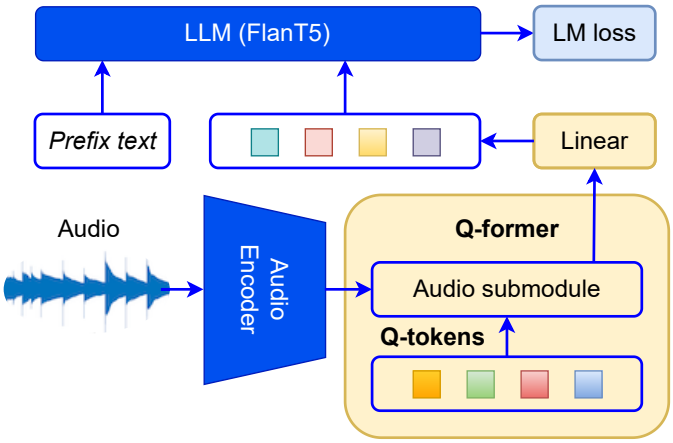

Fig. 1. Training Pipeline for the Generative Learning Stage. The pipeline first encodes the audio via an Audio Encoder. The result is then fed into the Q-former via cross-attention to produce output tokens that are further processed using a linear layer. The resulting tokens are then concatenated with an LLM instruction prompt. Finally, the LLM produces a loss function based on the reference caption. We freeze the audio encoder and the LLM during training. Yellow indicates the components that are trained, blue indicates frozen components.

various music-related tasks, such as music understanding, captioning, and reasoning.

Audio Datasets, especially in the music captioning domain generally lack data in terms of quantity or quality. MusicCaps [13] was one of the first music captioning datasets, containing 5k samples with expert-defined captions. MusicCaps is a subset of AudioSet [14], which contains captions for 10-second audio clips. SongDescriber [15] is a music captioning dataset that contains 1.1k crowd-sourced handwritten annotations for 706 full-length music pieces. YT8M-MusicTextClips [16] is a video-text dataset created for the task of retrieving suitable music for a video clip. The dataset contains handwritten audio captions for 4k video clips from the YouTube-8M dataset [17].

MusicBench [18] augmented MusicCaps by extracting and including music features of chords, beats, tempo, and key. The existing samples in MusicCaps were multiplied with musically meaningful augmentations, resulting in over 50k samples. The music captions were augmented using an LLM.

III. Model

We propose BLAP, a model capable of generating high-quality natural language captions for music. Our model architecture and training methodology are based on the BLIP-2 architecture [2], a recent successful image captioning model. BLAP bootstraps from a pre-trained audio encoder and a pre-trained LLM. This is enabled by a Q-Former that aligns the audio and text representations. Instead of fine-tuning an entire LLM, this strategy allows us to only learn the weights of the Q-former. Given that the Q-Former contains only a fraction of the parameters compared to the frozen LLM, this approach demands considerably less data than architectures in audio-language systems where the LLM is also trained.

The training process of BLAP consists of two stages. In the first stage, we focus on representation learning, training the Q-Former and computing the correct query tokens to extract relevant audio information (cf. Section III-A). In this stage, we also fine-tune the pre-trained CLAP audio encoder. The second stage aims to generate accurate captions by leveraging an LLM (cf. Section III-B). The LLM is kept frozen to reduce computing costs and helps avoid overfitting. Additionally, the audio encoder is frozen in the second stage to focus solely on optimizing the Q-Former and its query tokens. The authors of BLIP-2 [2] found that this two-stage training approach helps mitigate the problem of catastrophic forgetting.

A. Representation Learning Stage

In the first stage of training, we aim to learn a meaningful joint representation of music and text. We connect an audio encoder to the Q-Former. The goal is to train the Q-Former such that the query tokens learn to extract the most relevant information for text generation. In the first stage, we learn a representation that aligns both music and text and can be decoded into a descriptive caption. Therefore, we complete the audio-text contrastive loss (L_{ATC}) with two additional losses: the audio-text matching loss L_{ATM}, and an audio-based language generating loss L_{LM}. Each loss uses its own self-attention pattern to connect the queries and the text tokens. For general training all three losses are minimized jointly.

B. Generative Learning Stage

Figure 1 illustrates the generative learning stage. During this stage, prompt tuning is conducted by fine-tuning and transforming the query outputs of the Q-Former to generate appropriate inputs for the frozen LLM. More precisely, the Q-Former query outputs are transformed using a linear layer to acquire the prefix tokens, which are then prepended to a fixed prefix text of `"Generate an objective music description."` This processed input is then fed into the LLM. We use a pre-trained FLAN-T5-xl model [8] which is an encoder-decoder transformer architecture [19]. The encoder uses bi-directional attention to process the input into an embedding. Subsequently, the language modeling (LM) loss [20] is computed by inputting the reference caption into the decoder with causal attention. Both the Language Model and the audio encoder remain frozen throughout this training stage.

Freezing the LLM and the audio encoder leads to a significant reduction in the number of parameters that need to be updated, enabling the use of larger batch sizes. Additionally, this forces the model to refine the Q-Former's capability to extract audio information relevant to the caption from the audio encoding.

IV. Experiments

A. Setup

As described in Section III, the model consists of three main components: an audio encoder, a Q-Former, and an LLM. We list each component in more detail and denote the stage in which each component is active:

Audio Encoder (Stage 1 and 2): To encode the audio we use the HTS Audio Transformer (HTS-AT), introduced in CLAP [21]. We used their checkpoint to initialize the model. HTS-AT has been shown to outperform traditional audio encoders using CNNs [22]. We initialize HTS-AT using the LAION model checkpoint [7].

Q-Former (Stage 1 and 2): We used $\text{BERT}_{\text{base}}$ [23] as the base model in the Q-Former. We initialize the BERT model with pre-trained weights and initialize all cross-attention layers randomly since the original BERT model did not use cross-attention. The Q-Former contains 182 million trainable parameters. Since the hidden size of the BERT model is 768 we use 16 query tokens each of dimension 768.

Large Language Model (Stage 2): We use a pre-trained FLAN-T5-xl model [8]. The total number of parameters of BLAP sum up to 3 billion parameters, however, since most of these parameters belong to the frozen LLM, we need to update only a fraction of the total parameter count during training (182 million).

Dataset. For training we used a dataset consisting of 31k royalty-free music snippets from Shutterstock, totaling 700 hours of data. The music snippets also contain metadata tags and human-generated captions. The included metadata describes mood, genre, and the instruments used.

B. Evaluation Metrics

To evaluate models in music captioning, we use metrics commonly found in image captioning [2] and general audio captioning [24]. Historically, metrics such as BLEU [25], ROUGE [26], and METEOR [27] have been used to assess text quality and similarity [28]; however, these metrics have been superseded by more recent metrics [29], [30].

We evaluate on SPICE [31] and SPIDEr [32]. SPICE focuses on the semantic propositional content of captions rather than the n-gram overlap, aiming to better simulate human judgment [31]–[33]. SPIDEr combines SPICE and CIDEr [34]. The authors of SPIDEr introduce a policy gradient method to optimize this combined metric, improving both the semantic relevance and the syntactic fluency of generated captions. Both metrics have been shown to correlate more closely with human judgments compared to other metrics such as CIDEr, METEOR, ROUGE, and BLEU [31]–[33].

Furthermore, we evaluate models on FENSE [35], a learned metric specifically tailored for audio captioning, integrating the capabilities of Sentence-BERT [36] to assess similarity, along with an error detector. The error detector identifies and focuses on fluency errors within sentences. Unlike Sentence-Bert and BERT-Score [33], which do not sufficiently penalize or may even favor incorrectly phrased sentences, FENSE offers a more comprehensive evaluation of the quality of the generated captions, with a particular emphasis on linguistic fluency.

C. Training Details

During training of the first stage, in addition to training the Q-Former and Q-tokens we also fine-tuned the weights of the CLAP audio encoder. We found that this significantly helped the overall model performance; we assume this is because CLAP was mainly trained on general audio, and by fine-tuning CLAP the resulting embeddings better represent music. The small model size of the first stage allowed us to use a large batch size of 1120, which is important for contrastive learning. In comparison, BLIP-2 [2] used a batch size of 2320. We used a learning rate of $3 \cdot 10^{-5}$, a weight decay of $3 \cdot 10^{-4}$, and the Adam optimizer [37] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. During training of the second stage, we only update the Q-former weights and Q-tokens, keeping the audio encoder frozen. The second stage is initialized with the weights trained in the first stage. Due to the larger model size of the second stage, we reduced the batch size to 400, and trained with a learning rate of $3 \cdot 10^{-4}$.

BLAP was trained on eight A100 GPUs. The first stage (cf. Section III-A) was trained for 22.5k steps. The second stage (cf. Section III-B) was trained for 70k steps. During training, the model did not show signs of overfitting and maintained effective generalization on the validation set, and therefore, we assume that the model could be improved further with more training.

To compare BLAP with LP-MusicCaps and Salmonn, both trained on the MusicCaps dataset, we also fine-tuned BLAP on MusicCaps. We used the MusicCaps training split, containing 2.6k samples, and trained for 500 steps, using a batch size of 400. To avoid overfitting on MusicCaps, we adjusted the learning rate from $3 \cdot 10^{-4}$ to $3 \cdot 10^{-6}$.

V. Results

Quantitative Analysis. We compared BLAP with LP-MusicCaps [5], Qwen-Audio [38], Pengi [4], and Salmonn [6] on MusicCaps [13], Song Describer [15], and YouTube8M-MTC [16]. For the LP-MusicCaps model, we used the model checkpoint fine-tuned on MusicCaps. For Salmonn, we used the 13B model checkpoint.

The results of the models are shown in Table I. We observe that BLAP produces competitive results on all three music-text datasets. Since LP-MusicCaps, Salmonn, and Pengi have evaluated their performance on the BLEU metric we also add it to our evaluation, even though BLEU has been shown to correlate poorly with human judgment [31]–[33]. A primary factor contributing to BLAP's weaker performance in this metric is its tendency to produce more concise captions, which inherently result in lower n-gram scores. When considering SPIDEr, SPICE, and FENSE, which all correlate well with human judgement, BLAP outperforms the other models on the evaluation datasets. In the case of FENSE, which is the most relevant metric for audio and music captioning [35], BLAP achieves a relative improvement of 3.5% for MusicCaps, 6.5% for Song Describer, and 7.5% for YouTube8M-MTC compared to the previous state-of-the-art LP-MusicCaps.

Qualitative Analysis. To provide an analysis of the qualitative performance, we generate captions with BLAP, LP-MusicCaps, Salmonn, Qwen-Audio, and Pengi on the evaluation subset of MusicCaps samples (cf. Table II). We highlight model performance and comparisons on the MusicCaps

TABLE I

RESULTS ON MUSICCAPS, SONG DESCRIBER, AND YOUTUBE8M-MUSICTEXTCLIPS DATASETS. WE MEASURE THE PERFORMANCE OF PENGI, QWEN-AUDIO, SALMONN, LP-MUSICCAPS (LP-MC), AND BLAP ON BLEU@1 (B1), BLEU@4 (B4), FENSE (F), SPIDER (SP), AND SPICE (SC). FOR ALL METRICS, HIGHER IS BETTER. BLAP OUTPERFORMS PREVIOUS MODELS IN SPICE AND FENSE, TWO RELEVANT CAPTIONING METRICS.

	MusicCaps					Song Describer					YouTube8M-MTC				
Model	B1	B4	F	SP	SC	B1	B4	F	SP	SC	B1	B4	F	SP	SC
Pengi	12.5	0.6	45.8	5.3	7.1	10.8	0.3	38.6	4.0	4.5	8.1	0.3	35.1	4.6	5.9
Qwen-Audio	14.9	2.1	36.4	4.5	3.1	5.8	0.2	31.3	2.0	2.2	4.3	0.1	35.9	**5.5**	3.2
Salmonn	**29.2**	3.7	47.9	10.5	9.2	11.2	**0.7**	43.9	3.6	6.5	9.3	**0.6**	45.6	3.2	7.0
LP-MC	29.1	**5.3**	54.7	**10.4**	10.2	11.4	0.3	46.1	3.4	5.1	9.6	0.3	48.0	3.2	6.1
BLAP (ours)	26.2	3.3	**56.6**	8.8	**11.2**	**12.9**	0.4	**49.1**	**4.3**	**6.6**	**11.6**	0.3	**51.6**	3.9	**7.4**

TABLE II

EXAMPLES FROM THE MUSICCAPS DATASET WITH THE ORIGINAL EXPERT HUMAN-ANNOTATED CAPTION AND THE CAPTIONS GENERATED BY PENGI (9K HOURS OF TRAINING DATA), QWEN-AUDIO (137K HOURS OF TRAINING DATA), SALMONN (4.4K HOURS OF TRAINING DATA), LP-MUSICCAPS (4.4K HOURS OF TRAINING DATA), AND OUR PROPOSED MODEL BLAP (0.7K HOURS OF TRAINING DATA). ADDITIONAL EXAMPLES CAN BE FOUND ONLINE.

Models	**Caption**
Original Caption	The low quality recording features an emotional electric guitar melody played with two different reverb pedal effects, one after the other. It sounds like the player is comparing the two reverbs. The recording is a bit noisy.
BLAP (ours)	The low quality recording features an electric guitar playing a blues song. It sounds like it is being played by someone who has just learned how to play the instrument for the first time.
LP-MusicCaps	This audio contains someone playing a guitar on a clean e-guitar. In the background you can hear the crowd cheering. This song may be playing live during a concert.
Salmonn	This music is an instrumental. The main melody is played on a guitar and is accompanied by a keyboard playing a simple chord progression. The mood of the music is upbeat and cheerful. This music is suitable for use in the soundtrack of a romantic comedy.
Qwen-Audio	A man sings this song. The song is medium tempo with a piano accompaniment, guitar lead, bass guitar and drums. The song is emotional and sad. The song is a cover of a song originally by John Lennon.
Pengi	a guitar is being played on a electric guitar. a guitar is being played. this audio contains sound events: guitar, plucked string instrument, musical instrument, music.
Original Caption	This is a loud rock music piece played over two toy trains crashing. There is an even louder explosion sound effect whenever the toys crash. The distorted electric guitar plays a simple tune in the minor scale while the acoustic drums keep a basic rock beat as the rhythmic background.
BLAP (ours)	This is a heavy metal song. The sound quality is low, but the music is loud and energetic. It sounds like it's being played on an amplified guitar or synthesizer. There are no vocals in this track.
LP-MusicCaps	This audio contains a fully overdriven aggressive kick sound with a lot of digital noise sounds. This is an amateur recording. This song may be playing in a post apocalyptic video game.
Salmonn	This music is an instrumental. The tempo is fast with an energetic and upbeat rhythm. The main melody is played on the electric guitar with a distorted sound. The rhythm section consists of a punchy bass line and powerful drums. The overall mood of the music is energetic and upbeat. This music is suitable for use in action scenes in movies or video games.
Qwen-Audio	A low quality recording features a video game soundtrack playing in the background, followed by a loud explosion and a mechanical sound effect. It sounds like a video game level or a training video.
Pengi	a person is playing a music loop. a loop is being played. this audio contains sound events: the sounds of drums and bass, vocals, organic music and organic music.

dataset, as it is a dataset with captions written by expert musicians, ensuring consistently high-quality reference captions. We provide additional examples,[1] including from Song Describer and YouTube8M-MusicTextClips, together with their audio. We find that BLAP tends to generate more concise captions compared to other models, because the ShutterStock training dataset contains mostly brief and low-quality captions. However, BLAP manages to capture the essence of the music piece, while using significantly less training data. The generated captions provide a well-rounded mixture of high-level music descriptions and low-level music details, ensuring a comprehensive and informative representation of the musical content. Although BLAP compares well to LP-MusicCaps and Salmonn while using significantly less data and smaller model architectures, it is clear that BLAP also cannot yet reach the quality of the expert human-annotated captions found in MusicCaps, leaving room for further improvements in future work.

VI. CONCLUSION

We propose BLAP, a new music captioning model that bootstraps a pre-trained CLAP audio encoder and a frozen Flan-T5-xl LLM in order to lower data and compute requirements. BLAP relies on a Q-Former, and is trained in a two-stage approach. We demonstrate the effectiveness of BLAP on several relevant metrics and in qualitative evaluations. We believe BLAP represents a promising avenue for exploring the intersection of music and language, with potential future directions for research and applications in cross-modal learning.

[1]https://lucala.github.io/BLAP/

REFERENCES

[1] Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi, "BLIP: bootstrapping language-image pre-training for unified vision-language understanding and generation," in *ICML*, 2022, pp. 12888–12900.

[2] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," 2023.

[3] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, and Furu Wei, "Image as a foreign language: Beit pretraining for all vision and vision-language tasks," *arXiv preprint arXiv:2208.10442*, 2022.

[4] Soham Deshmukh, Benjamin Elizalde, Rita Singh, and Huaming Wang, "Pengi: An audio language model for audio tasks," 2023.

[5] SeungHeon Doh, Keunwoo Choi, Jongpil Lee, and Juhan Nam, "Lp-musiccaps: Llm-based pseudo music captioning," 2023.

[6] Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang, "Salmonn: Towards generic hearing abilities for large language models," *arXiv preprint arXiv:2310.13289*, 2023.

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

[8] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al., "Scaling instruction-finetuned language models," *arXiv preprint arXiv:2210.11416*, 2022.

[9] Ilaria Manco, Emmanouil Benetos, Elio Quinton, and György Fazekas, "Muscaps: Generating captions for music audio," in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2021, pp. 1–8.

[10] Thierry Bertin-Mahieux, Daniel Ellis, Brian Whitman, and Paul Lamere, "The million song dataset.," *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*, pp. 591–596, 01 2011.

[11] Josh Gardner, Simon Durand, Daniel Stoller, and Rachel M Bittner, "Llark: A multimodal foundation model for music," *arXiv preprint arXiv:2310.07160*, 2023.

[12] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al., "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[13] Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, et al., "Musiclm: Generating music from text," *arXiv preprint arXiv:2301.11325*, 2023.

[14] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[15] Ilaria Manco, Benno Weck, SeungHeon Doh, Minz Won, Yixiao Zhang, Dmitry Bodganov, Yusong Wu, Ke Chen, Philip Tovstogan, Emmanouil Benetos, et al., "The song describer dataset: a corpus of audio captions for music-and-language evaluation," *arXiv preprint arXiv:2311.10057*, 2023.

[16] Daniel McKee, Justin Salamon, Josef Sivic, and Bryan Russell, "Language-guided music recommendation for video via prompt analogies," 2023.

[17] 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.

[18] Jan Melechovsky, Zixun Guo, Deepanway Ghosal, Navonil Majumder, Dorien Herremans, and Soujanya Poria, "Mustango: Toward controllable text-to-music generation," 2023.

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," 2023.

[20] Yoshua Bengio, Réjean Ducharme, and Pascal Vincent, "A neural probabilistic language model," *Advances in neural information processing systems*, vol. 13, 2000.

[21] Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang, "Clap learning audio concepts from natural language supervision," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

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

[23] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019.

[24] Etienne Labbé, Thomas Pellegrini, and Julien Pinquier, "Irit-ups dcase 2023 audio captioning and retrieval system," in *Proc. Conf. Detection Classification Acoust. Scenes Events Challenge*, 2023, pp. 1–5.

[25] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[26] Chin-Yew Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.

[27] Satanjeev Banerjee and Alon Lavie, "Meteor: An automatic metric for mt evaluation with improved correlation with human judgments," in *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 2005, pp. 65–72.

[28] Mehmet Ali Dursun and Soydan Serttaş, "A multi-metric model for analyzing and comparing extractive text summarization approaches and algorithms on scientific papers," *Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi*, vol. 15, no. 1, pp. 31–48, 2024.

[29] Kathrin Blagec, Georg Dorffner, Milad Moradi, Simon Ott, and Matthias Samwald, "A global analysis of metrics used for measuring performance in natural language processing," *arXiv preprint arXiv:2204.11574*, 2022.

[30] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi, "Clipscore: A reference-free evaluation metric for image captioning," *arXiv preprint arXiv:2104.08718*, 2021.

[31] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould, "Spice: Semantic propositional image caption evaluation," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*. Springer, 2016, pp. 382–398.

[32] Siqi Liu, Zhenhai Zhu, Ning Ye, Sergio Guadarrama, and Kevin Murphy, "Improved image captioning via policy gradient optimization of spider," in *2017 IEEE International Conference on Computer Vision (ICCV)*. Oct. 2017, IEEE.

[33] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi, "Bertscore: Evaluating text generation with bert," *arXiv preprint arXiv:1904.09675*, 2019.

[34] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh, "Cider: Consensus-based image description evaluation," 2015.

[35] Zelin Zhou, Zhiling Zhang, Xuenan Xu, Zeyu Xie, Mengyue Wu, and Kenny Q Zhu, "Can audio captions be evaluated with image caption metrics?," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 981–985.

[36] Nils Reimers and Iryna Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *arXiv preprint arXiv:1908.10084*, 2019.

[37] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," 2017.

[38] Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou, "Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models," *arXiv preprint arXiv:2311.07919*, 2023.