arXiv:2309.09836v1 [eess.AS] 18 Sep 2023

RECAP: RETRIEVAL-AUGMENTED AUDIO CAPTIONING

Sreyan Ghosh, Sonal Kumar, Chandra Kiran Reddy Evuru, Ramani Duraiswami, Dinesh Manocha

University of Maryland, College Park, USA

ABSTRACT

We present **RECAP** (**RE**trieval-Augmented Audio **CAP**tioning), a novel and effective audio captioning system that generates captions conditioned on an input audio and other captions similar to the audio retrieved from a datastore. Additionally, our proposed method can transfer to any domain without the need for any additional fine-tuning. To generate a caption for an audio sample, we leverage an audio-text model CLAP [1] to retrieve captions similar to it from a replaceable datastore, which are then used to construct a prompt. Next, we feed this prompt to a GPT-2 decoder and introduce cross-attention layers between the CLAP encoder and GPT-2 to condition the audio for caption generation. Experiments on two benchmark datasets, Clotho and AudioCaps, show that RECAP achieves competitive performance in in-domain settings and significant improvements in out-of-domain settings. Additionally, due to its capability to exploit a large text-captions-only datastore in a *training-free* fashion, RECAP shows unique capabilities of captioning novel audio events never seen during training and compositional audios with multiple events. To promote research in this space, we also release 150,000+ new weakly labeled captions for AudioSet, AudioCaps, and Clotho[1].

***Index Terms*—** Automated audio captioning, multimodal learning, retrieval-augmented generation

1. INTRODUCTION

Audio captioning is the fundamental task of describing the contents of an audio sample using natural language. Compared to Automatic Speech Recognition (ASR), which transcribes human speech, audio captioning focuses on describing distinct environmental sounds in the input audio [2, 3]. By bridging the gap between text and audio modalities, audio captioning has found various applications in real-world use cases like environment monitoring, gaming, etc. [4].

In the past, most audio captioning models employed an encoder-decoder architecture using an off-the-shelf pre-trained audio encoder and a language decoder [5, 6]. The audio encoder generates an audio embedding sequence that is used to condition the language decoder for caption generation. However, most of these systems do not perform well on cross-domain settings (trained on one domain and tested on the other), and every use case might need separate training. We hypothesize that the primary reason behind this phenomenon is the shift of occurrence of unique audio events with a domain shift. For example, the AudioCaps benchmark dataset [2] has several audio concepts (e.g., the sound of jazz or an interview) that Clotho, another benchmark dataset, does not. This is also representative of real-world scenarios where not only do audio concepts change from one domain to another (e.g., environmental sounds in a city versus a forest), but new audio concepts also keep emerging within a domain (e.g., new versions of an online game).

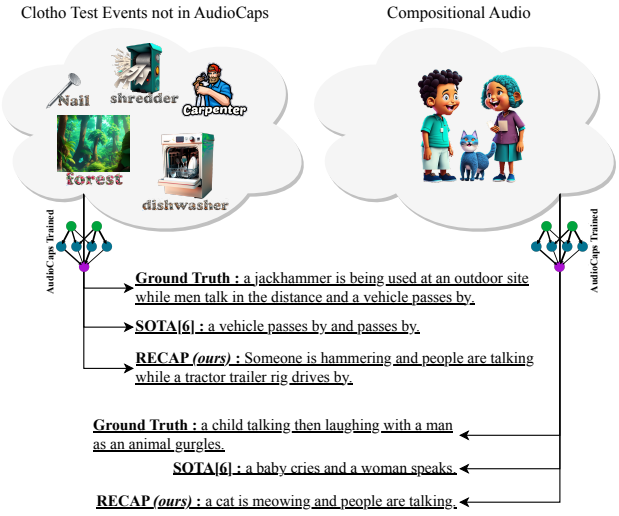

Fig. 1. We propose **RECAP**, a retrieval-augmented audio captioning model. RECAP can caption novel concepts never before seen in training and improves the captioning of audio with multiple events.

Main Contributions. We propose RECAP, **RE**trieval-Augmented Audio **CAP**tioning, a simple and scalable solution to the aforementioned problems of domain shifts. Similar to other audio captioning systems in literature [5, 6, 7], RECAP is built on an audio encoder and a language decoder (GPT-2 in our setting). However, we introduce three novel changes: (1) Instead of employing an audio encoder pre-trained only on audio, we use CLAP [1] as our audio encoder. CLAP is pre-trained on audio-text pairs to learn the correspondence between audio and text by projecting them into a

[1]We will release code and data on paper acceptance.

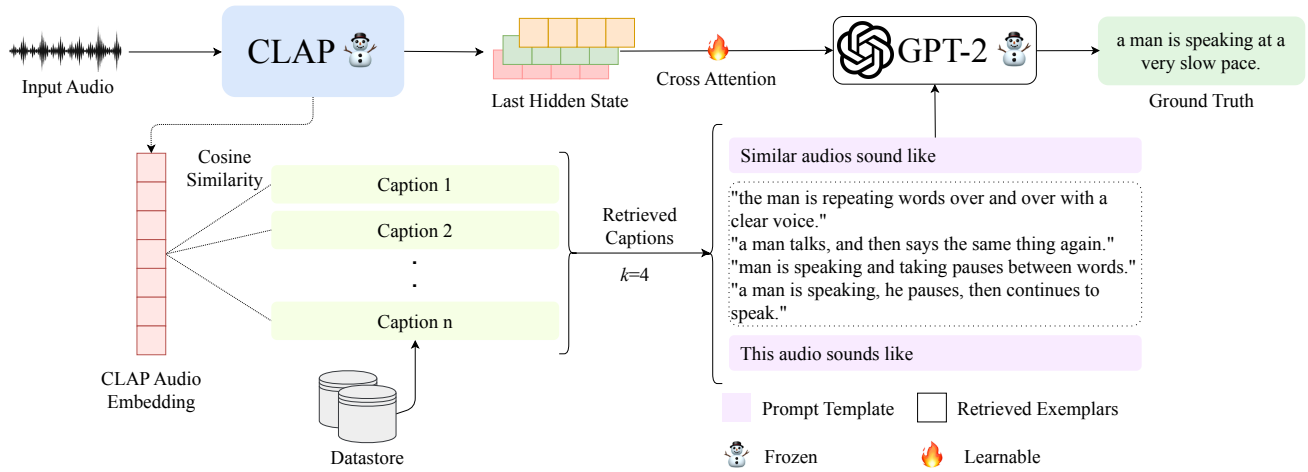

Fig. 2. Illustration of **RECAP**. RECAP fine-tunes a GPT-2 LM conditioned on audio representations from the last hidden state of CLAP [1] and a text prompt. The text prompt is constructed using captions most similar to the audio, retrieved from a datastore using CLAP.

shared latent space. Thus, CLAP hidden state representations are better suited for captioning due to their enhanced linguistic comprehension. (2) We condition the audio for caption generation by introducing new cross-attention layers between CLAP and the GPT-2. (3) Finally, beyond just conditioning audio, we also condition a custom-constructed prompt for training and inference. We construct the prompt using the top-k captions most similar to the audio from a datastore retrieved using CLAP. We provide more details in Section 3.1. RECAP builds on retrieval-augmented generation (RAG) [8], which offers multiple advantages discussed further in Section 3. RECAP is lightweight, fast to train (as we only optimize the cross-attention layers), and can exploit any large text-caption-only datastore in a *training-free* fashion. We evaluate RECAP on two benchmark datasets, Clotho [3] and AudioCaps [2], and show that while being competitive to the state-of-the-art in in-domain settings, RECAP outperforms all baselines in out-of-domain settings by a large margin. Additionally, RECAP can effectively caption novel audio events never seen during training and can better generate captions for compositional audios with multiple audio events.

2. RELATED WORK

Automated Audio Captioning. Current work in audio captioning primarily employs encoder-decoder models where a caption is generated by an autoregressive language decoder conditioned on representations obtained from an audio encoder [5, 6, 7]. The language decoder employed is either pretrained on web-scale data [5, 6, 7] or learned from scratch [9, 10] during fine-tuning. The work closest to ours is [7], where the authors condition a GPT-2 on prompts constructed using retrieved captions. However, the key difference between our work and theirs is that we require only a text-caption-only datastore for RECAP, whereas their system requires both audio and text pairs. We also introduce additional cross-attention layers for audio conditioning. *Kim et. al* [6], the current state-of-the-art system, proposed prefix tuning for audio captioning where the authors feed a prefix or a fixed-size embedding sequence to GPT-2 for audio captioning. Other works include synthetic data augmentation techniques [11, 12], and training tricks to improve learning on the source training data [13, 14].

Retrieval-augmented Generation. The core idea of retrieval-augmented generation (RAG) is to condition generation on additional data retrieved from an external datastore [8]. RAG has been shown to benefit knowledge-intensive NLP tasks like open-domain question-answering on datasets that require world knowledge and advanced reasoning capabilities [15, 16]. RAG has also proven to be extremely effective in various computer vision tasks, including image captioning [17, 18]. We argue that audio captioning, especially in out-of-domain scenarios, is a knowledge-intensive task as it requires the model to caption novel audio concepts never seen during training, and can benefit from RAG.

3. METHODOLOGY

Problem Formulation. Given a dataset $\mathcal{D}$ with audio-text pairs ($\mathcal{A}$,$\mathcal{T}$), where each text caption $t_i \in \mathcal{T}$ corresponding to an audio $a_i \in \mathcal{A}$ describes the content or events of the audio, we aim to train a model θ to generate t_i from a_i. Different from other audio captioning systems, we also assume that the model has access to a datastore $\mathcal{DS}$ with text captions during inference. These captions come from the training set of $\mathcal{D}$ or external sources but have no overlap with the validation or test sets of $\mathcal{D}$.

3.1. RECAP

Overall Architecture. The overall architecture of RECAP

Table 1. Evaluation on Clotho. Each method is trained on three different settings and tested on the AudioCaps dataset. For evaluation, we use a datastore that has captions from the training set ($\mathcal{DS}$), Clotho ($\mathcal{DS}_{caps}$), or a large external dataset ($\mathcal{DS}_{large}$).

Training set	Method	$BLEU_1$	$BLEU_2$	$BLEU_3$	$BLEU_4$	METEOR	$ROUGE_L$	CIDEr	SPICE	SPIDEr
(1) Clotho	Mei *et al*. [19]	0.527	0.327	0.211	0.131	0.158	0.356	0.320	0.105	0.213
	Gontier *et al*. [5]	0.506	0.318	0.210	0.134	0.148	0.338	0.278	0.092	0.185
	Chen *et al*. [20]	0.534	0.343	0.230	0.151	0.160	0.356	0.346	0.108	0.227
	Xu *et al*. [10]	0.556	0.363	0.242	0.159	0.169	0.368	0.377	0.115	0.246
	Koh *et al*. [21]	0.551	0.369	0.252	**0.168**	0.165	0.373	0.380	0.111	0.246
	Kim *et al*. [6]	0.560	0.376	0.253	0.160	0.170	0.378	0.392	0.118	**0.255**
	RECAP (w/ $\mathcal{DS}$)	0.563	0.381	**0.257**	0.165	**0.179**	0.383	0.398	0.122	0.214
	RECAP (w/ $\mathcal{DS}_{large}$)	**0.582**	**0.384**	**0.257**	0.166	0.177	**0.395**	**0.411**	**0.125**	0.224
(2) AudioCaps	Mei *et al*. [19]	0.294	0.146	0.080	0.043	0.096	0.239	0.117	0.050	0.084
	Gontier *et al*. [5]	0.309	0.146	0.071	0.034	0.098	0.233	0.112	0.046	0.079
	Chen *et al*. [20]	0.226	0.114	0.065	0.039	0.086	0.228	0.109	0.042	0.076
	Kim *et al*. [6]	0.342	0.195	0.115	0.065	0.112	0.276	0.192	0.074	0.133
	RECAP (w/ $\mathcal{DS}_{caps}$)	0.339	0.193	0.109	0.068	0.110	0.276	0.195	0.084	0.137
	RECAP (w/ $\mathcal{DS}$)	0.515	0.349	0.210	0.143	0.155	**0.328**	**0.332**	0.988	0.201
	RECAP (w/ $\mathcal{DS}_{large}$)	**0.519**	**0.355**	**0.216**	**0.149**	**0.157**	0.324	0.331	**1.004**	**0.209**
(3) Clotho & AudioCaps	Mei *et al*. [19]	0.516	0.318	0.204	0.127	0.157	0.351	0.313	0.105	0.209
	Gontier *et al*. [5]	0.461	0.282	0.182	0.117	0.136	0.318	0.251	0.083	0.167
	Chen *et al*. [20]	0.516	0.325	0.215	0.141	0.153	0.350	0.314	0.102	0.208
	Kim *et al*. [6]	0.539	0.346	0.227	0.142	0.159	0.366	0.319	0.111	0.215
	RECAP (w/ $\mathcal{DS}$)	0.547	**0.361**	**0.238**	0.149	**0.167**	0.379	0.322	**0.116**	**0.222**
	RECAP (w/ $\mathcal{DS}_{large}$)	**0.549**	0.360	**0.238**	**0.150**	0.166	**0.381**	**0.323**	**0.116**	0.221

is quite simple and lightweight. RECAP employs CLAP as the audio encoder and GPT-2 as the auto-regressive language decoder. To generate the caption, the language decoder conditions on the output of the audio encoder and an individually crafted prompt for each audio. We discuss how we construct the prompt in the next subsection.

For audio conditioning, we first pass the audio samples through the CLAP audio encoder and extract the last hidden state $A \in n \times d$, where n is the sequence length and d is the embedding dimension. This embedding is extracted from the penultimate layer of the CLAP audio encoder right before the final projection. As the audio embeddings and decoder operate on different vector spaces, we connect them through randomly initialized cross-attention modules as each decoder layer. To train the RECAP, we freeze both GPT-2 and the CLAP and only train the cross-attention layers, which reduces overall compute requirements and time for training and retains the expressivity and generalization capabilities of GPT-2. RECAP performs well even after training only 5.4% of total parameters because, like other retrieval-augmented models [8, 22, 23], RECAP does not need all information to be stored in its weights as it has access to external knowledge from a datastore of text. Additionally, CLAP generates an audio embedding that correlates well with its corresponding textual description, thus further lowering training time due to its superior understanding of the audio content.

Constructing prompts with Retrieved Captions. Instead of just conditioning audio features for captioning, RECAP is also conditioned on a prompt, individually crafted for each audio during training and inference. To construct this prompt, RECAP exploits CLAP text and audio encoders [1], to retrieve top-*k* captions similar to an audio from a datastore. CLAP encodes audio and text to a shared vector space and has outperformed all prior models on audio-to-text and text-to-audio retrieval, thus making it most suitable for our task. Specifically, for retrieval, we calculate the cosine similarity between the embeddings of the current audio a_i and all the text captions in the datastore $\mathcal{DS}$ and just choose the captions with the highest similarity. Once we have retrieved the top-*k* similar captions, we construct a prompt in the following manner: *"Audios similar to this audio sounds like: caption 1, caption 2, $\cdots$ caption k. This audio sounds like:"*. For retrieval, we naturally ignore the original caption t_i corresponding to a_i. RECAP is then trained using the generic cross-entropy loss between the tokens for the predicted caption $\hat{t_i}$ and the ground-truth caption t_i.

4. EXPERIMENTS AND RESULTS

Datasets. For training and evaluating RECAP, we use either Clotho [3], AudioCaps [2], or a combination of both. Clotho has 3839/1045/1045 unique audios in train/dev/test splits, respectively, with five captions for each audio. AudioCaps has 49,838/495/975 with five captions each only for the train set.

Baselines. We compare RECAP with six competitive baselines that are taken from literature. Eren *et al*. [9] and Xu *et al*. [10] train a Gated Recurrent Unit (GRU) for generating captions, conditioned on audio embeddings extracted from an audio encoder. Chen *et al*. [20] replaces the GRU with a transformer decoder, and Mei *et al*. [19] trains an entire encoder-decoder transformer architecture from scratch. Kim *et al*. [6] and Gontier *et al*. [5] use a pre-trained language model, where the former employs GPT-2, and the latter employs BART [24].

Experimental Setup. To compare the performance of RECAP, we conduct experiments in three distinct setups: (1) We

Table 2. Evaluation on AudioCaps. Each method is trained on three different settings and tested on the AudioCaps dataset. For evaluation, we use a datastore that has captions from the training set ($\mathcal{DS}$), Clotho ($\mathcal{DS}_{clotho}$), or a large external dataset ($\mathcal{DS}_{large}$).

Training set	Method	$BLEU_1$	$BLEU_2$	$BLEU_3$	$BLEU_4$	METEOR	$ROUGE_L$	CIDEr	SPICE	SPIDEr
(1) AudioCaps	Mei *et al*. [19]	0.647	0.488	0.356	0.252	0.222	0.468	0.679	0.160	0.420
	Gontier *et al*. [5]	0.699	0.523	0.380	0.266	0.241	0.493	**0.753**	0.176	0.465
	Chen *et al*. [20]	0.550	0.385	0.264	0.178	0.173	0.390	0.443	0.117	0.280
	Eren *et al*. [9]	0.710	0.490	0.380	0.230	**0.290**	**0.590**	0.750	-	-
	Kim *et al*. [6]	0.713	0.552	0.421	0.309	0.240	0.503	0.733	0.177	0.455
	RECAP (w/ $\mathcal{DS}$)	0.721	**0.559**	**0.428**	**0.316**	0.252	0.521	0.750	0.183	**0.472**
	RECAP (w/ $\mathcal{DS}_{large}$)	**0.722**	0.557	**0.428**	0.313	0.256	0.525	0.751	**0.186**	0.471
(2) Clotho	Mei *et al*. [19]	0.415	0.219	0.121	0.063	0.134	0.303	0.149	0.066	0.107
	Gontier *et al*. [5]	0.425	0.223	0.124	0.061	0.128	0.298	0.147	0.060	0.104
	Chen *et al*. [20]	0.365	0.170	0.091	0.048	0.110	0.273	0.083	0.049	0.066
	Kim *et al*. [6]	0.449	0.266	0.157	0.084	0.144	0.330	0.211	0.083	0.147
	RECAP (w/ $\mathcal{DS}_{clotho}$)	0.427	0.224	0.148	0.065	0.112	0.281	0.191	0.078	0.136
	RECAP (w/ $\mathcal{DS}$)	0.501	**0.326**	**0.211**	0.104	0.164	**0.357**	0.359	**0.116**	0.198
	RECAP (w/ $\mathcal{DS}_{large}$)	**0.507**	0.321	0.206	**0.108**	**0.169**	**0.357**	**0.362**	0.111	**0.204**
(3) Clotho & AudioCaps	Mei *et al*. [19]	0.682	0.507	0.369	0.266	0.238	0.488	0.701	0.166	0.434
	Gontier *et al*. [5]	0.635	0.461	0.322	0.219	0.208	0.450	0.612	0.153	0.383
	Chen *et al*. [20]	0.489	0.292	0.178	0.106	0.152	0.346	0.265	0.093	0.179
	Kim *et al*. [6]	0.708	0.547	0.402	0.283	0.238	0.499	0.710	0.167	0.438
	RECAP (w/ $\mathcal{DS}$)	**0.728**	**0.563**	**0.425**	0.317	0.252	**0.529**	**0.764**	0.187	**0.469**
	RECAP (w/ $\mathcal{DS}_{large}$)	0.725	0.561	0.424	**0.319**	**0.256**	**0.529**	0.761	**0.190**	**0.469**

train and evaluate the model on the same dataset $\mathcal{D}$, (2) We train the model on a dataset $\mathcal{D}$ and evaluate the model on a different dataset $\hat{\mathcal{D}}$ (3) We train the model on a combination of both datasets and evaluate separately on individual datasets. For (1), the datastore $\mathcal{DS}$ consists of captions from either the training set of the source dataset $\mathcal{D}$ or a large curated datastore $\mathcal{DS}_{large}$. For (2), we use $\mathcal{DS}$ that has captions from either $\mathcal{D}$ ($\mathcal{DS}$), $\mathcal{DS}_{large}$ or from the other dataset. For (3), we either use $\mathcal{DS}$ that has captions from both datasets or use $\mathcal{DS}_{large}$. We list all the sources of $\mathcal{DS}_{large}$ with over 600,000+ text-only captions, on our GitHub. This includes 100,000+ new weakly labeled captions for the AudioSet strong subset and three new captions for each sample in AudioCaps and Clotho. All these captions were generated using GPT-4 and manually corrected by one expert human annotator. For retrieval-based prompt creation, we use k=4 and retrieve only the top 4 captions from the datastore. It is worth noting that RECAP does not use any additional training or data augmentation tricks. For both AudioCaps and Clotho, we train using Adam optimizer with a learning rate of $5e^{-5}$ for 100 epochs and a batch size of 32. We evaluate all our models on the metrics of BLEU, METEOR, ROUGE-L, CIDEr, SPICE, and SPIDEr.

Results. Table 1 and Table 2 compare the performance of RECAP against all our baselines evaluated on Clotho and AudioCaps, respectively. We train our models in different settings and evaluate them with different datastores. While RECAP shows decent margins of improvement in in-domain settings, RECAP outperforms all baselines by a significant margin in out-of-domain settings when an in-domain datastore is available. Without one, RECAP shows competitive performance with SOTA [6]. The presence of a larger data store ($\mathcal{DS}_{large}$) almost always improves performance. This opens possibilities to improve captioning performance by augmenting the datastore with diverse synthetically generated captions.

Results Analysis. Table 3 compares RECAP with Kim*et al*. [6] (SOTA) on compositional instances from Clotho (**1.**) and AudioCaps (**4.**) test set. While SOTA was able to caption only one audio event, due to conditioning on a prompt constructed from diverse retrieved captions, RECAP captures multiple. We also compared with a model trained on AudioCaps and inferred on a Clotho test instance with an audio event never seen during training (**2.**), and vice-versa (**3.**). By being conditioned on in-domain prompts, RECAP can caption these instances effectively.

Table 3. Comparing RECAP in 4 challenging settings.

Ground Truth	1: a engine roars in the background while pieces of metal are being dropped in. 2: a moving vehicle has some metal container in it clinging against each other. 3: nature sounds with a frog croaking. 4: a vehicle driving as a man and woman are talking and laughing.
SOTA [6]	1: a bell is ringing and a bell rings. 2: rain falling on a surface. 3: people are talking and laughing with a man speaking in the background. 4: a person is talking in the background.
RECAP	1: A person is using a chisel to cut wood and a car passes by. 2: Water splashes while a car drives by in the rain. 3: several vehicles move and a beep goes off. 4: an adult male is speaking, and a motor vehicle engine is running.

5. CONCLUSION AND FUTURE WORK

We present RECAP, a novel audio captioning system based on retrieval-augmented generation. While being competitive with state-of-the-art methods on benchmark datasets, RECAP outperforms SOTA by a huge margin on out-of-domain settings and shows unique capabilities of captioning novel audio events and compositional audios with two or more events. Additionally, RECAP is cheap to train and can exploit a replaceable text-caption-only datastore in a *training-free* fashion to further push performance. As part of future work, we would like to explore advanced techniques for efficient retrieval and build better audio-text models.

6. REFERENCES

[1] 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 *IEEE ICASSP 2023*.

[2] Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim, "Audiocaps: Generating captions for audios in the wild," in *ACL 2019*, pp. 119–132.

[3] Konstantinos Drossos, Samuel Lipping, and Tuomas Virtanen, "Clotho: An audio captioning dataset," in *IEEE ICASSP 2020*, pp. 736–740.

[4] A Sophia Koepke, Andreea-Maria Oncescu, Joao Henriques, Zeynep Akata, and Samuel Albanie, "Audio retrieval with natural language queries: A benchmark study," *IEEE Transactions on Multimedia*, 2022.

[5] Félix Gontier, Romain Serizel, and Christophe Cerisara, "Automated audio captioning by fine-tuning bart with audioset tags," in *DCASE2021 Challenge*, 2021.

[6] Minkyu Kim, Kim Sung-Bin, and Tae-Hyun Oh, "Prefix tuning for automated audio captioning," in *IEEE ICASSP 2023*, pp. 1–5.

[7] Yuma Koizumi, Yasunori Ohishi, Daisuke Niizumi, Daiki Takeuchi, and Masahiro Yasuda, "Audio captioning using pre-trained large-scale language model guided by audio-based similar caption retrieval," *arXiv preprint arXiv:2012.07331*, 2020.

[8] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al., "Retrieval-augmented generation for knowledge-intensive nlp tasks," *NeurIPS 2020*, pp. 9459–9474.

[9] Ayşegül Özkaya Eren and Mustafa Sert, "Audio captioning based on combined audio and semantic embeddings," in *IEEE International Symposium on Multimedia*, 2020.

[10] Xuenan Xu, Heinrich Dinkel, Mengyue Wu, Zeyu Xie, and Kai Yu, "Investigating local and global information for automated audio captioning with transfer learning," in *ICASSP 2021*.

[11] Wu et al., "Beats-based audio captioning model with instructor embedding supervision and chatgpt mix-up," Tech. Rep., DCASE2023 Challenge.

[12] Marek Kadlčík, Adam Hájek, Jürgen Kieslich, and Radosław Winiecki, "A whisper transformer for audio captioning trained with synthetic captions and transfer learning," *arXiv preprint arXiv:2305.09690*, 2023.

[13] Haoran Sun, Zhiyong Yan, Yongqing Wang, Heinrich Dinkel, Junbo Zhang, and Yujun Wang, "Leveraging multi-task training and image retrieval with clap for audio captioning," Tech. Rep., DCASE2023 Challenge.

[14] Jaeheon Sim, Eungbeom Kim, and Kyogu Lee, "Label-refined sequential training with noisy data for automated audio captioning," Tech. Rep., DCASE2023 Challenge.

[15] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang, "Semantic parsing on freebase from question-answer pairs," in *EMNLP 2013*, pp. 1533–1544.

[16] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al., "Natural questions: a benchmark for question answering research," *TACL 2019*, pp. 453–466.

[17] Sara Sarto, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara, "Retrieval-augmented transformer for image captioning," in *Proceedings of the 19th International Conference on Content-based Multimedia Indexing*, 2022, pp. 1–7.

[18] Rita Ramos, Desmond Elliott, and Bruno Martins, "Retrieval-augmented image captioning," *arXiv preprint arXiv:2302.08268*, 2023.

[19] Xinhao Mei, Xubo Liu, Qiushi Huang, Mark D. Plumbley, and Wenwu Wang, "Audio captioning transformer," in *DCASE2021 Challenge*.

[20] Kun Chen, Yusong Wu, Ziyue Wang, Xuan Zhang, Fudong Nian, Shengchen Li, and Xi Shao, "Audio captioning based on transformer and pre-trained cnn.," in *DCASE*, 2020, pp. 21–25.

[21] Andrew Koh, Xue Fuzhao, and Chng Eng Siong, "Automated audio captioning using transfer learning and reconstruction latent space similarity regularization," in *ICASSP 2022*.

[22] Izacard et al., "Few-shot learning with retrieval augmented language models," *arXiv preprint arXiv:2208.03299*, 2022.

[23] Huayang Li, Yixuan Su, Deng Cai, Yan Wang, and Lemao Liu, "A survey on retrieval-augmented text generation," *arXiv preprint arXiv:2202.01110*, 2022.

[24] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *ACL 2020*, pp. 7871–7880.