Speech Retrieval-Augmented Generation without Automatic Speech Recognition

Do June Min*†, Karel Mundnich‡, Andy Lapastora‡, Erfan Soltanmohammadi‡, Srikanth Ronanki‡, Kyu Han‡

*University of Michigan, dojmin@umich.edu

‡AWS AI Labs, {kmundnic, allapast, solterfa, ronanks, kyujhan}@amazon.com

Abstract—One common approach for question answering over speech data is to first transcribe speech using automatic speech recognition (ASR) and then employ text-based retrieval-augmented generation (RAG) on the transcriptions. While this cascaded pipeline has proven effective in many practical settings, ASR errors can propagate to the retrieval and generation steps. To overcome this limitation, we introduce SpeechRAG, a novel framework designed for open-question answering over spoken data. Our proposed approach fine-tunes a pre-trained speech encoder into a speech adapter fed into a frozen large language model (LLM)–based retrieval model. By aligning the embedding spaces of text and speech, our speech retriever directly retrieves audio passages from text-based queries, leveraging the retrieval capacity of the frozen text retriever. Our retrieval experiments on spoken question answering datasets show that direct speech retrieval does not degrade over the text-based baseline, and outperforms the cascaded systems using ASR. For generation, we use a speech language model (SLM) as a generator, conditioned on audio passages rather than transcripts. Without fine-tuning of the SLM, this approach outperforms cascaded text-based models when there is high WER in the transcripts.

Index Terms—speech retrieval-augmented generation, spoken content retrieval, cross-modal retrieval, multimodal retrieval, open question answering, audio language model, speech language model.

I. INTRODUCTION

Retrieval-Augmented Generation (RAG) [1] has enabled Large Language Models (LLMs) to generate responses using data not available during any of their training stages. This increases the helpfulness of these models since they can be used as an interface to explore topics released after the training, or to cite sources of information to improve factuality. However, RAG remains mainly used for text-based sources, which may contain images and tables [2], [3].

Over recent years, however, we have observed a surge in the creation of unstructured content and information such as audio recordings or videos containing spoken information. One tool that has the potential to enable efficient search through these expanding audio archives to structure the information is spoken content retrieval [4]. This method indexes and retrieves passages in audio format and offers a solution for searching large collections of speech data such as meeting recordings [5]. However, available spoken content retrieval systems treat the problem as a variant of text retrieval, where the source text is automatic speech recognition (ASR) transcriptions of audio produced by an ASR system [4], which usually contain errors. This system is often referred to as a *cascaded* model, where the output of the ASR step is fed to the text-based retriever as input. With the recent advancements in speech processing [6] and text retrieval [7], the cascaded model presents a robust baseline across many speech tasks, including spoken dialogue state tracking [8] and spoken language understanding (SLU) [9].

Despite the robustness of the cascaded approach, there are downsides in its use for retrieval from data in spoken form. For example, the errors from ASR can propagate downstream, and negatively

†Work done while the author was interning at AWS AI Labs.

impact retrieval and generation performance. This problem can be exacerbated in challenging topics for ASR systems such as named entity recognition [10], even more so considering that named entities are often central to accurate retrieval since often they are used to match queries to passages. In addition, applying ASR to speech results in the loss of paralinguistic information. To best represent the information contained in speech without loss, speech should be indexed and retrieved in its original form.

In this work, we propose a solution to overcome the limitations of the ASR-based cascaded retrieval systems and instead directly index and search audio passages in their original speech format. Specifically, we tackle the problem of text query to audio passage retrieval by using a text embedding model to encode audio passages in the same text embedding space, effectively allowing multimodal retrieval using a single embedding model.

Our main contributions are:

- We propose and implement an end-to-end speech retrieval system that embeds both text and audio in the same space, allowing retrieval of text and speech interchangeably,
- Our method used a lightweight adapter between an LLM-based text embedding model and a speech encoder, making it data efficient during training and avoiding cross-modal contrastive learning,
- We implement an end-to-end speech RAG framework that requires no ASR for open question answering from spoken passages.

II. BACKGROUND

Inspired by the success of CLIP [11] in the task of cross-modal retrieval, several audio embedding models have been proposed, leveraging large audio-caption pair datasets with contrastive learning techniques [12], [13], [14], [15]. Moreover, powerful audio and speech encoders have been made widely available, leading to better audio representations for a wide array of tasks [6], [16]. However, models like CLAP-LAION, CLAP-MS, and others are limited to matching audio to natural language descriptions of an audio event, instead of matching speech to its corresponding language content [17].

On the other hand, spoken content retrieval involves matching the linguistic and semantic content of speech contained in audio to queries [4], [18]. Recent models typically use contrastive learning [19], [20] to train speech embedding models that can either match spoken or text queries to audio passages. SpeechDPR tackles spoken query to spoken passage retrieval by training an end-to-end model with teacher distillation from dense text retrievers [21]. Our work tackles text query to audio passage retrieval and avoids expensive contrastive learning while achieving competitive performance with ground truth (GT) text retrieval.

Finally, another related work is ReSLM [22], where the authors propose to use a retrieval approach to boost the ASR performance of

(a) Speech retriever.　　　　　　　　　　　(b) Our SpeechRAG framework (top) and cascaded system (bottom).
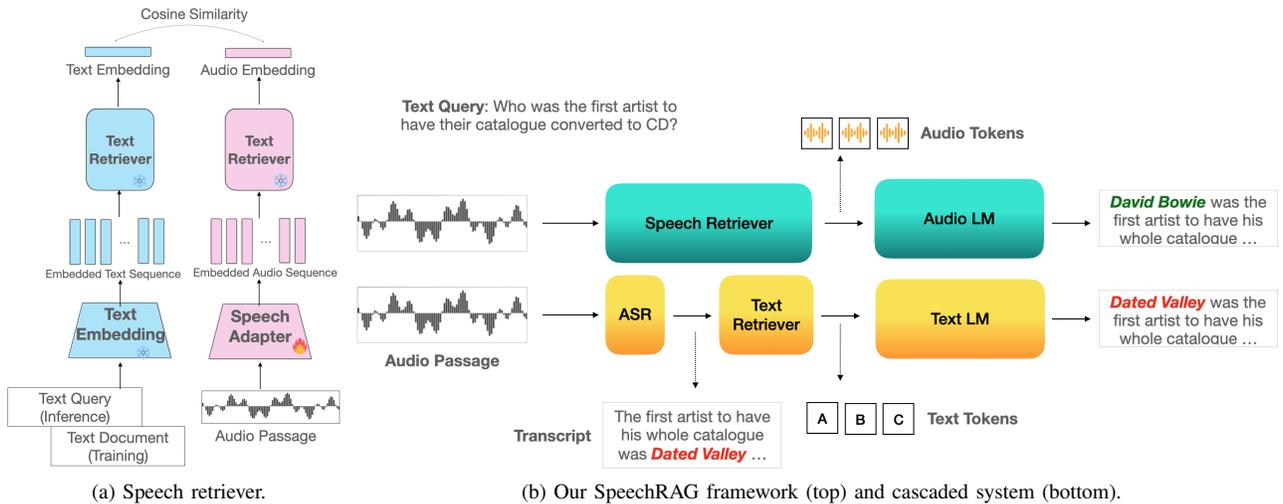
Fig. 1: **(a) Our speech retriever** consists of an adapter that projects speech into the embedding space of the text retrieval model. During training, we use distillation from the text embedding of the transcript of the speech to refine our speech embedding. This allows us to leverage the frozen text retriever's capacity during a similarity search with the query. During testing, we use the text branch (Fig 1a, left) to embed text queries and the speech branch (Fig 1a, right) to embed speech passages. **(b) Our SpeechRAG framework** consists of a speech retriever and an SLM (Figure 1b, top) and operates directly on speech. On the other hand, ASR-based cascaded baselines (Fig 1b, bottom) first transcribe the audio and use text-based RAG, leading to the propagation of ASR errors in retrieval and generation.

an SLM in named entity recognition. However, this approach differs from ours in that the retrieval is performed to obtain different text representations for specific spoken words to augment the prompt used for ASR.

III. RETRIEVAL FROM SPOKEN PASSAGES

Our framework for text-to-speech retrieval consists of a fine-tuned speech adapter and a frozen unimodal text retriever (shown in Fig. 1a). The speech adapter projects downsampled speech representations into the text embedding space of the text retriever model. This architecture allows us to leverage the retrieval capacity of state-of-the-art text retrievers without optimizing a cross-modal embedding model from scratch, which would require much larger datasets.

A. Speech adapter for text-based retriever

The function of the speech adapter is to adapt the speech representations of the input audio into the text embedding space of the downstream text model. In this work, we follow a similar approach as [23]. Specifically, the adapter, which consists of an encoder and a downsampler, is trained using a cosine embedding similarity loss propagated in an end-to-end manner from the text model.

Speech encoder: We use HuBERT [24], a pre-trained speech encoder, and feed the last hidden layer representations to the speech adapter.

Speech adapter: Typically, the text token sequence length of a speech transcript is much shorter than the sequence length of the corresponding discretized speech. Therefore, we use an average pooling layer of the time dimension to downsample the speech representations, and a projection layer to upsample the speech representation to the LLM embedding dimension [25].

B. Cross-modal retriever

To encode both modalities (text for queries and speech for passages), we use the same frozen text-based retriever as our backbone [26], [27]. The main difference across modalities is the embedding module, which maps the raw data (text tokens, audio features) into the embedding space of the text retriever.

Embedding text: We use the original token embedding module of the text retriever model. Then, the input embedding sequences are processed by the retriever, and the final layer's representations are pooled to create e_t, the fixed-sized representation of the text.

Embedding speech: Our speech adapter embeds speech into a sequence of embeddings by first deriving frame-level speech features with the pre-trained speech encoder, and downsampling the feature sequences over the temporal dimension. The embeddings are then consumed by the retriever, with e_s as the final embedding output.

Training loss: Given e_t, the text embedding of a ground truth transcript of an audio passage and e_s, the audio embedding of the audio passage, we compute the distillation loss using the cosine embedding loss:

$$\mathcal{L}(e_s, e_t) = 1 - \cos(e_s, e_t) = 1 - \frac{e_s \cdot e_t}{||e_s||||e_t||}. \qquad (1)$$

C. Audio-conditioned generator

For an end-to-end speech RAG model that does not require ASR, we use a pre-trained multi-task SLM trained on various speech tasks such as speech recognition or question answering [28]. Similar to our cross-modal retriever, the SLM consists of a speech adapter and a frozen text model.

IV. EXPERIMENTS

A. Data

We use two speech datasets, SpokenSQuAD [29], and VoxPopuli [30] for our RAG experiments. SpokenSQuAD is a spoken version of the SQuAD dataset, where Wikipedia text are converted into speech using text-to-speech systems [31]. It is annotated with text queries about the spoken passages and ground truth answers. The audio passages are pre-chunked in passage-level, with the average spoken passage duration of ∼60s.

VoxPopuli is a large collection of speech from the European Parliament events, with utterances averaging ∼10s. We use the English subset of VoxPopuli and create query and answer pairs from

Passage	The conclusion of the Framework Agreement provides a legally...
Query	What legally binding instrument provides for upgrading and strengthening EU-Australia bilateral relations as well as increasing cooperation between them?
GT Answer	Framework Agreement

TABLE I: Example of a generated query and answer in VoxPopuli.

the speech using LLM-prompting. We first identify potential answer candidates by extracting named entities from each utterance using a fine-tuned BERT model [32], and use answer-aware generation to mine text queries. Specifically, we prompt Claude 3.5 Sonnet to generate a question that is answered by the extracted named entity, given the utterance as context (Table I). For both datasets, each query has only one relevant passage.

B. Implementation details

Retriever: For our speech encoder, we use HuBERT-large [24], which uses self-supervised learning to generate deep representations of speech sampled at 16kHz. Our speech adapter downsamples the output of HuBERT 4 times (for a final frame length of 80ms). Both the speech encoder and adapter are unfrozen. For our frozen retriever backbone, we use E5-Mistral-7B-Instruct LLM-based retriever [27]. We train the model for 20 epochs, with a stopping criteria of validation loss, with a patience of 3. We use the Adam optimizer with a learning rate of 5e-5, and $\beta_1 = 0.9, \beta_2 = 0.999$. We train with a batch size of 4 and a gradient accumulation step of 16.

Generator: After audios are retrieved from the vector database, each query is combined with the top-k retrieved audios and a task instruction prompt as a prompt an instruction-tuned LLM [1] which generate answers. to an 7B-parameter SLM [2] to generate an answer [28]. We use the SLM as is, without fine-tuning its parameters to this specific task.

C. Evaluation

Retrieval: We use Recall@k as our evaluation metrics for the retrieval experiment:

$$\text{Recall@}k = \frac{\text{\# of relevant passages in top-}k}{\text{Total \# of relevant items}}, \quad (2)$$

where $k = 5, 10, 100$. Each query has exactly one relevant passage. For the relevance score, we use the cosine similarity between the text query embedding and the audio passage embedding.

Generation: For the generation experiment, we use top-5 retrieved passages as context provided along with the text query and the LLM instruction. To evaluate the correctness of the generated answer, we use

- ***Exact Match (EM)***, which assigns 1 if the ground-truth is in the answer, otherwise 0, and
- ***LLM Correctness***, implemented as machine-based evaluation of match to cover minor spelling alterations or other edge cases [33].

D. Baselines: fully-cascaded and semi-cascaded RAG

We implement two types of cascaded RAG baselines. The *fully-cascaded baseline* consists of a cascaded text retriever (an ASR module and a text retriever) and an LLM generator. This framework transcribes the audio passages and treats spoken content retrieval as

---

[1] https://hf.co/mistralai/Mistral-7B-Instruct-v0.2
[2] https://hf.co/Qwen/Qwen-Audio-Chat

a text retrieval problem. The *semi-cascaded baseline* uses our speech retriever to retrieve the audio passages, then uses the transcripts of the audios and uses them to condition generation. For our baselines, we use Qwen-7B-Chat as the text LLM generator.

To investigate the effect of transcription quality of the ASR step, we implement different versions of the baseline, each with varying levels of average WER. We use the transcriptions as ground truth text (where we assume that the WER is 0%), while the case uses the same audio encoder used for the speech adapter. To simulate a severe WER scenario, we use a small ASR module[3] trained on 100hrs of Librispeech [34].

Both baselines use the same frozen text retriever as the speech retriever and the text-only version of the SLM (Qwen-7B-Chat).

V. RESULTS

A. Retrieval results

	Passage WER	Recall@5	Recall@10	Recall@100
SpokenSQuAD				
GT Text Baseline	0%	0.9707	0.9871	0.9985
Low WER cascaded	~20%	0.9525	0.9745	0.9974
High WER cascaded	~35%	0.8768	0.9271	0.9926
Speech Retriever	N/A	**0.9702**	**0.9869**	**0.9986**
VoxPopuli				
GT Text Baseline	0%	0.9942	0.9971	1.0
Low WER cascaded	~17%	0.9826	0.9855	0.9961
High WER cascaded	~45%	0.7106	0.7493	0.8858
Speech Retriever	N/A	**0.9952**	**0.9981**	**0.9990**

TABLE II: Retrieval results. Passage WER reports the average WER of the text transcripts of the speech data. The speech retriever operates directly on audio passages and achieves performance on par with retrieval on ground truth text (WER 0%).

We show our retrieval experiment results in Table II. We observe that across both datasets, our proposed speech retriever outperforms the cascaded baselines. This shows that our method offers a retrieval performance advantage even when the ASR step is done using a relatively high-performance ASR model. While the comparison between the cascaded model and the ground truth text baseline indicates that the text retrieval model can be robust against low WER error rates, the large performance drop in the High WER results shows that noisy transcriptions can lead to severe retrieval performance degradation. Our speech retrieval removes the possibility of ASR error propagation at the retrieval step by operating directly on speech.

Moreover, our speech retriever achieves a retrieval performance that is on par with that of the ground truth transcript-based retrieval, even obtaining a slightly higher performance in one metric (Recall@10, VoxPopuli set), showing that our proposed speech retriever is a practical and powerful alternative to the cascading framework of spoken content retrieval.
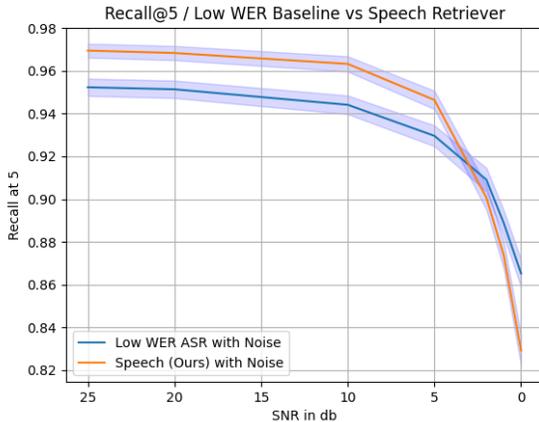
B. Retrieval performance under different noise levels

To study the effect of noise on retrieval performance, we plot the Recall@5 of the text baselines and our speech retriever at different noise levels, as shown in Fig. 2. We add Gaussian noise at different signal-to-noise (SNR) levels to the audio. We observe that across both datasets, our speech retriever is more robust to noise than the cascaded baseline that uses the same speech encoder as our end-to-end retriever, with the exception of a very high noise setting for the SpokenSQuAD dataset (which is TTS-based).
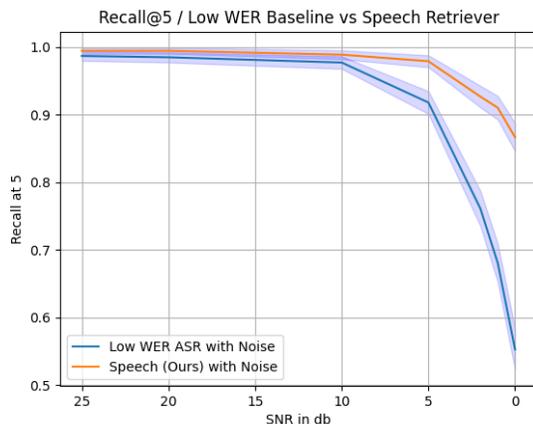
---

[3] https://hf.co/jkang/espnet2_librispeech_100_conformer_word

(a) Result on SpokenSQuAD

(b) Result on VoxPopuli

Fig. 2: Retrieval performance comparison with injected noise. We inject Gaussian noise to the speech signals and compare the text-based vs. our end-to-end retriever at different SNRs.

C. Generation results

Table III shows our comparison of SpeechRAG, fully-cascaded, and semi-cascaded RAG baselines. Promisingly, we find that the full SpeechRAG framework outperforms the High-WER cascaded baselines on both datasets, highlighting the potential of SpeechRAG. For example, Table 3 shows how SpeechRAG avoids the corruption of context information by an inaccurate transcription of named entities. However, we also find that it performs worse than the fully- and semi-cascaded baselines under settings, with a larger gap for the SpokenSQuAD dataset. This is possibly due to the difference in the durations of the retrieved audios, with an average SpokenSQuAD audio lasting 4 times as long as the average VoxPopuli utterance. This adversely affects performance since SLMs are typically not trained to handle multiple, long-context audios.

VI. CONCLUSION

In this work, we propose a first fully speech-based solution to question answering over speech. To achieve this, we implement a speech retriever consisting of a speech adapter and a frozen LLM-based text retriever and show that by indexing and retrieving speech

TABLE III: Generation results. We use top-5 passages from the previous retrieval experiment as context. SpeechRAG performs better than the baselines High WER cases, but not in cases.

	Passage WER	Exact Match	LLM Correctness
SpokenSQuAD			
GT Text Baseline	0%	0.7514	0.8352
Fully-cascaded Low WER	∼20%	0.5019	0.6987
Fully-cascaded High WER	∼35%	0.2684	0.3701
Semi-cascaded GT Text	0%	0.7364	0.8013
Semi-cascaded Low WER	∼20%	0.5057	0.7072
Semi-cascaded High WER	∼35%	0.2787	0.3842
SpeechRAG	N/A	0.3522	0.4811
VoxPopuli			
GT Text Baseline	0%	0.9080	0.9003
Fully-cascaded Low WER	∼17%	0.7473	0.7561
Fully-cascaded High WER	∼45%	0.4511	0.3950
Semi-cascaded GT Text	0%	0.9158	0.9071
Semi-cascaded Low WER	∼17%	0.7301	0.7561
Semi-cascaded High WER	∼45%	0.4327	0.3766
SpeechRAG	N/A	0.8045	0.7173

> **Passage:** *...whilst under Soviet rule, Armenian classical music composer **Aram Khatchaturian** became internationally well known for his music, for various ballets and the Sabre Dance...*
>
> **Text query:** *Who composed the Sabre Dance?*
>
> **Ground truth answer:** *Aram Khatchaturian*
>
> **ASR transcript (top-1 retrieved):** . . . *whilst under soviet rule armenian classical music composer **aram cocheterien** became internationally well known for his music for various ballets and the sabre dance* . . .
>
> **Fully-cascaded generation:** *Aram Cocheterien composed the Sabre Dance.*
>
> **SpeechRAG generation:** *The Sabre Dance was composed by Aram Khachaturian.*

Fig. 3: SpokenSQuAD generations of the fully-cascaded model vs our ASR-less SpeechRAG framework. The named entity transcription error in the ASR step propagates to the generation step, while the SLM of SpeechRAG correctly generates the named entity.

directly, our framework outperforms cascaded retrieval in noisy ASR scenarios, and matches ground truth text retrieval. In the generation step, our framework bypasses ASR by using an SLM conditioned on retrieved audio. While the SpeechRAG generation outperforms cascaded baselines in high WER scenarios, we identify the potential for improvement in the performance gap between our framework and the low WER baseline generations.

REFERENCES

[1] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela, "Retrieval-augmented generation for knowledge-intensive nlp tasks," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, Eds. 2020, vol. 33, pp. 9459–9474, Curran Associates, Inc.

[2] Wenhu Chen, Hexiang Hu, Xi Chen, Pat Verga, and William Cohen, "MuRAG: Multimodal retrieval-augmented generator for open question answering over images and text," in *Proceedings of the 2022 Conference*

on Empirical Methods in Natural Language Processing, Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, Eds., Abu Dhabi, United Arab Emirates, Dec. 2022, pp. 5558–5570, Association for Computational Linguistics.

[3] Pankaj Joshi, Aditya Gupta, Pankaj Kumar, and Manas Sisodia, "Robust multi model rag pipeline for documents containing text, table & images," 2024 3rd International Conference on Applied Artificial Intelligence and Computing (ICAAIC), pp. 993–999, 2024.

[4] Lin-shan Lee, James Glass, Hung-yi Lee, and Chun-an Chan, "Spoken content retrieval—beyond cascading speech recognition with text retrieval," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 23, no. 9, pp. 1389–1420, 2015.

[5] Ciprian Chelba, Timothy J. Hazen, and Murat Saraclar, "Retrieval and browsing of spoken content," IEEE Signal Processing Magazine, vol. 25, no. 3, pp. 39–49, 2008.

[6] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever, "Robust speech recognition via large-scale weak supervision," 2022.

[7] Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers, "MTEB: Massive text embedding benchmark," in Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, Dubrovnik, Croatia, May 2023, pp. 2014–2037, Association for Computational Linguistics.

[8] Léo Jacqmin, Lucas Druart, Yannick Estève, Benoît Favre, Lina M Rojas, and Valentin Vielzeuf, "OLISIA: a cascade system for spoken dialogue state tracking," in Proceedings of The Eleventh Dialog System Technology Challenge, Prague, Czech Republic, Sept. 2023, pp. 95–104, Association for Computational Linguistics.

[9] Dmitriy Serdyuk, Yongqiang Wang, Christian Fuegen, Anuj Kumar, Baiyang Liu, and Yoshua Bengio, "Towards end-to-end spoken language understanding," in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018, pp. 5754–5758.

[10] Piotr Szymański, Lukasz Augustyniak, Mikolaj Morzy, Adrian Szymczak, Krzysztof Surdyk, and Piotr Żelasko, "Why aren't we NER yet? artifacts of ASR errors in named entity recognition in spontaneous speech transcripts," in Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Toronto, Canada, July 2023, pp. 1746–1761, Association for Computational Linguistics.

[11] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever, "Learning transferable visual models from natural language supervision," in International Conference on Machine Learning, 2021.

[12] 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), 2023, pp. 1–5.

[13] 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.

[14] Joshua David Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka, "Contrastive learning with hard negative samples," in International Conference on Learning Representations, 2021.

[15] Changyou Chen, Jianyi Zhang, Yi Xu, Liqun Chen, Jiali Duan, Yiran Chen, S. Tran, Belinda Zeng, and Trishul M. Chilimbi, "Why do we need large batchsizes in contrastive learning? a gradient-bias perspective," in Neural Information Processing Systems, 2022.

[16] Paul-Ambroise Duquenne, Holger Schwenk, and Benoit Sagot, "SONAR: sentence-level multimodal and language-agnostic representations," 2023.

[17] Andreea-Maria Oncescu, A. Sophia Koepke, João F. Henriques, Zeynep Akata, and Samuel Albanie, "Audio retrieval with natural language queries," 2021.

[18] Yung-Sung Chuang, Chi-Liang Liu, and Hung yi Lee, "Speechbert: Cross-modal pre-trained language model for end-to-end spoken question answering," ArXiv, vol. abs/1910.11559, 2019.

[19] Sreyan Ghosh, Sonal Kumar, Chandra Kiran Reddy Evuru, Ramani Duraiswami, and Dinesh Manocha, "Recap: Retrieval-augmented audio captioning," in ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2024, pp. 1161–1165.

[20] Mingqiu Wang, Izhak Shafran, Hagen Soltau, Wei Han, Yuan Cao, Dian Yu, and Laurent El Shafey, "Retrieval augmented end-to-end spoken dialog models," ArXiv, vol. abs/2402.01828, 2024.

[21] Chyi-Jiunn Lin, Guan-Ting Lin, Yung-Sung Chuang, Wei-Lun Wu, Shang-Wen Li, Abdelrahman Mohamed, Hung yi Lee, and Lin shan Lee, "Speechdpr: End-to-end spoken passage retrieval for open-domain spoken question answering," 2024.

[22] Mingqiu Wang, Izhak Shafran, Hagen Soltau, Wei Han, Yuan Cao, Dian Yu, and Laurent El Shafey, "Retrieval augmented end-to-end spoken dialog models," 2024.

[23] Nilaksh Das, Saket Dingliwal, Srikanth Ronanki, Rohit Paturi, Zhaocheng Huang, Prashant Mathur, Jie Yuan, Dhanush Bekal, Xing Niu, Sai Muralidhar Jayanthi, Xilai Li, Karel Mundnich, Monica Sunkara, Sundararajan Srinivasan, Kyu J Han, and Katrin Kirchhoff, "Speechverse: A large-scale generalizable audio language model," 2024.

[24] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," 2021.

[25] Wonjune Kang and Deb Roy, "Prompting large language models with audio for general-purpose speech summarization," 2024.

[26] Nils Reimers and Iryna Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. 11 2019, Association for Computational Linguistics.

[27] Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei, "Improving text embeddings with large language models," 2024.

[28] 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.

[29] Chia-Hsuan Lee, Szu-Lin Wu, Chi-Liang Liu, and Hung-yi Lee, "Spoken squad: A study of mitigating the impact of speech recognition errors on listening comprehension," Proc. Interspeech 2018, pp. 3459–3463, 2018.

[30] Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux, "VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation," in Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online, Aug. 2021, pp. 993–1003, Association for Computational Linguistics.

[31] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang, "SQuAD: 100,000+ questions for machine comprehension of text," in Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas, Nov. 2016, pp. 2383–2392, Association for Computational Linguistics.

[32] Erik F. Tjong Kim Sang and Fien De Meulder, "Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition," in Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, 2003, pp. 142–147.

[33] Sujoy Roychowdhury, Sumit Soman, H G Ranjani, Neeraj Gunda, Vansh Chhabra, and Sai Krishna Bala, "Evaluation of rag metrics for question answering in the telecom domain," 2024.

[34] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2015, pp. 5206–5210.