Examining the Impact of Transcript Variation on Podcast Search and Re-Ranking

Watheq Mansour[0000−0002−9463−595X], J. Shane Culpepper[0000−0002−9463−595X], and Joel Mackenzie[0000−0001−7992−4633]

The University of Queensland

Abstract. Effectively retrieving and ranking spoken audio — such as podcasts — is an important problem in the field of Information Retrieval. A typical approach for the effective retrieval of podcasts is to reduce the problem to text retrieval by automatically transcribing the audio files to textual data, followed by segmentation, indexing, and retrieval. In this work, we examine how automatic transcription algorithms impact the effectiveness of podcast retrieval and re-ranking in dense and sparse retrieval configurations, motivated by the wide spectrum of quality-vs-cost transcription models currently available. Our results demonstrate that the choice of transcription model has a measurable impact on both end-to-end retrieval and late-stage re-ranking pipelines, for both dense and sparse retrievers. Our study highlights the issues and limitations of employing automatic speech recognition (ASR) models in podcast search and motivates future research on this important problem.

Keywords: Podcast search · Speech to text · Audio transcription.

1 Introduction

As the popularity of podcasts continues to grow, effectively representing spoken audio for ranked retrieval tasks has become an increasingly important research problem to the Information Retrieval (IR) community [5, 31]. Recent work demonstrates that, although podcasts often have rich metadata, such as show titles, episode titles, and descriptions provided by content creators, the most reliable signal for ranked retrieval tends to be the audio *speech* component of the content – which is transcribed using an *automatic speech recognition* (ASR) model [9]. However, many new ASR models are now available, each with a different quality-to-cost ratio. For example, public ASR APIs from Google and OpenAI cost fractions of a cent per minute of audio that is transcribed, but may be prohibitively expensive for certain tasks;[1] conversely, free, open-source options are readily available, but their quality can vary significantly. Therefore, exploring how this plethora of methods can impact the effectiveness of podcast retrieval is well-motivated. To the best of our knowledge, this is the first work to explicitly examine the end-to-end trade-offs of transcription cost versus retrieval quality. See Section 3 for a detailed list of our contributions.

[1] Transcribing 60,000 hours of audio – the size of the Spotify podcast corpus [10] — would cost between $8,000 to $22,000 USD as of October 2024.

2 Related Work

The first seminal work on spoken document retrieval dates back to the TREC spoken document retrieval track between 1997 and 2000, where manual and automatic transcripts of news were collected from radio and television broadcasts [13]. A key finding of this track was that automatic-transcription can achieve similar retrieval effectiveness to manually curated transcripts. In all subsequent work, various shared tasks have focused on speech retrieval in contexts such as personal testimonies [28], lectures [1, 2, 3, 4], and television shows [11, 12, 19, 34]. Refer to the work of Jones [16] for a comprehensive overview of spoken text retrieval, and to Jones et al. [18] for a discussion of future directions in podcast information access.

More recently, TREC introduced the Podcast Track in 2020 [17]. This track, along with the Spotify Podcast Dataset [10], provides further research and resources for podcast retrieval and summarization [9, 18]. Although track participants were offered the raw audio data, an automatically generated transcription was widely used for both summarization and ranking tasks. Intuitively, it seems likely that the quality of the ASR model would be correlated with the effectiveness of any downstream tasks using such automatically generated transcripts; this is highlighted by the work of Jones et al. [18] who discuss the importance of high-quality ASR methods on retrieval effectiveness [35]. Given that word-error rates for modern collections can be as high as 20% [10], we are motivated to revisit this fundamental and practical problem to determine how much recent advances in ASR impact end-to-end retrieval effectiveness. We hypothesize that more expensive ASR methods will generate more accurate transcripts, which will, in turn, lift retrieval quality.

3 Research Questions and Our Contribution

In this paper, we study the effect of the choice of ASR model on podcast search effectiveness. We aim to answer the following research questions:

RQ1: *How does the quality of an ASR model change re-ranking effectiveness?* Due to the prohibitive cost of transcribing terabytes of audio data, we present a preliminary experiment that explores how different transcription models can influence the effectiveness of podcast re-ranking, which requires only a subset of top-scoring results to be retranscribed. In particular, we assume the same, fixed, candidate document set, and re-rank this set with alternative document representations from different ASR models.

RQ2: *How does retranscribing the entire corpus impact end-to-end effectiveness?* An answer to RQ1 allows us to understand the impact of the ASR model on re-ranking effectiveness. However, this fails to capture the impact of the ASR model on *retrieval* effectiveness, where documents are not just being re-ranked, but are being retrieved from the whole corpus given a user query. Therefore, we identify the best-performing ASR models to retranscribe the entire corpus and measure end-to-end effectiveness.

RQ3: *Can dense retrieval alleviate the impact of ASR model choice for transcription?* In many real-world scenarios, resource limits require cheaper ASR models to be used. So, we are interested in exploring if dense *semantic search* methods can reduce the impact of ASR model choice. Our hypothesis is that dense models may be more robust to perturbations in the text caused by ASR methods than their lexical counterparts.

Reproducibility. In the interest of reproducibility, we make the code and experimental resources publicly available.

https://github.com/Watheq9/podcast_tr_vars

4 Experimental Setup

Here, we briefly outline our experimental setup, including the data and models used in our experiments.

Collection and Queries. We use the Spotify Podcast Dataset [10], which includes 100,000 episodes of English-language podcasts, comprised of nearly 60,000 hours of audio speech. Each episode is paired with metadata (description, etc.), and an automatically generated transcript using the Google Speech-to-Text API, as of early 2020. Each episode is split into 2 minute segments, with a 1-minute overlap; segments begin on the minute. In total, the collection contains 3.4 million segments, with an average word count of 340 ± 70 per segment. We used the TREC 2020 and 2021 podcast topics (50 from each year), reporting results on TREC 2020 for brevity. Our experiments use the keyword-only (title) queries. The 2020 topics are categorized into three types: topical (35), re-finding (8), and known-item (7). The 2021 topics are categorized into two types only: topical (40) and known-item (10).

ASR Models. We experiment using six different ASR models:

1. The **Baseline** is the default transcription provided in the Spotify podcast corpus [10], denoted as Spotify in the remainder of this work.
2. **Vosk**[2] is a lightweight open-source speech recognition toolkit with support for more than 20 languages. Our experiments use the small version that can be run on a single CPU; it is our only CPU-based model.
3. **Silero**[3] models are pre-trained enterprise-grade speech-to-text models. The models are lightweight and optimized for real-time applications. We use both the small and large models in this work.
4. **Whisper** [30] models are trained for speech recognition and translation tasks using weak supervision, using 680,000 hours of noisy audio data. Whisper

[2] https://alphacephei.com/vosk/
[3] https://github.com/snakers4/silero-models

models are robust against geographical accents, background noise, and technical language (law and medicine, for example). One limitation of Whisper is the chance of hallucinations, where the models may add words that are not spoken in the transcribed audio. We experiment with the multilingual version of tiny, base, small, and medium models.
5. **WhisperX** [8] is the state-of-the-art speech recognition model derived from Whisper. WhisperX is faster than Whisper and generates more accurate word-level timestamps using voice activity detection and forced phoneme alignment techniques.We test the base and large-v3 versions.
6. **Wav2Vec 2.0** [7] is a powerful, self-supervised speech representation model that uses raw audio input. We use the large model, which is fine-tuned using 960 hours of LibriSpeech data [27].

Ranked Retrieval and Re-ranking Systems. We use the `PyTerrier` toolkit [22, 23] for indexing, retrieval, and evaluation. For point-wise re-ranking, we use the MonoT5 model [26]; for list-wise re-ranking, we use the llm-based RankZephyr [29] re-ranker. Re-ranking was performed using the `llm-rankers` library [37].

Using the Massive Text Embedding Benchmark (MTEB) [25], we have selected `BAAI/bge-large-en-v1.5` [36] — an efficient yet effective pre-trained model — to encode the corpora for our ranked dense retrieval experiments. BM25 is used as a traditional lexical retrieval baseline [32]. One limitation of our analysis is that all models are applied in a *zero-shot* manner; no fine-tuning was conducted. We defer the question of whether our findings generalize under fine-tuning to future work.

Metrics. Early precision metrics including *rank-biased precision* (RBP) [24] with $\phi = 0.8$ and 0.9 (assuming a user model with expected browsing depths of 5 and 10 documents, respectively) and *normalized discounted cumulative gain* (NDCG) with a cut-off at 10 [15] are reported. These metrics are chosen as complementary *utility-* and *recall-oriented* metrics [21]. The degree of uncertainty caused by unjudged documents is also reported using the RBP residual. We measure first-stage retrieval effectiveness using recall at depth 100. Significance testing is performed using a two-tailed pairwise t-test, with a Bonferroni correction. We report significance at $p < 0.01$; the Spotify transcript is used as the baseline throughout. Effectiveness metrics are annotated using down arrows to represent negative significant differences.

5 Experiments

Re-Ranking Experiments. In order to answer RQ1, we use the top $k = 100$ segments (from BM25 on Spotify transcriptions) for each query, and map each one back to the original episodes; this results in a subset of around 2,000 episodes, totaling 54 GiB worth of audio. Next, we re-transcribed the episodes using each of the candidate ASR models, and measure the throughput (efficiency). Then, we

Table 1. Re-ranking the top 100 segments of a run generated by BM25 on the Spotify corpus using RankZephyr (list-wise) re-ranker. R@100 is 0.515.

ASR model	TSPS	RBP $\phi = 0.8$	RBP $\phi = 0.9$	NDCG@10	# Unret.
Spotify	–	0.569 + 0.142	0.491 + 0.174	0.473	–
Vosk-Small	21	0.527 + 0.160	0.459 + 0.190	0.451	404
Whisper-Tiny	42	0.530 + 0.160	0.463 + 0.189	0.441	256
Whisper-Base	35	0.542 + 0.147	0.472 + 0.182	0.452	191
Whisper-Small	21	0.544 + 0.141	0.476 + 0.175	0.453	166
Whisper-Medium	10	0.530 + 0.174	0.466 + 0.192	0.457	146
Wav2Vec2-Large	59	0.522 + 0.173	0.457 + 0.197	0.431	571
Silero-Small	635	0.500 + 0.188	0.446 + 0.207	0.422	738
Silero-Large	437	0.534 + 0.147	0.464 + 0.181	0.443	426
WhisperX-Base	115	0.527 + 0.166	0.465 + 0.190	0.443	74
WhisperX-LargeV3	52	0.549 + 0.149	0.476 + 0.184	0.450	43

re-segment each podcast episode — ensuring proper alignment with the original data — to generate *parallel segments*, which can be re-ranked.

To measure the re-ranking effectiveness, we pass the same 100 candidate segments (retrieved by BM25 on the Spotify corpus) to the re-ranker, replacing the segment text with the text generated by each ASR model. This allows us to isolate a static set of transcriptions and then measure the ranking quality. We only report results using RankZephyr as it was the most effective algorithm. We observed similar trends using MonoT5, albeit with lower overall performance.

Table 1 summarises the re-ranking results. The first column, TSPS, reports the throughput for each model in *transcribed seconds per second* — the number of seconds of audio that is transcribed per wall-clock second — showing a wide spectrum of cost profiles that are available for current commercial ASR models. Interestingly, in terms of early precision, the ASR model does not seem to have a significant effect on the quality of the re-ranking, even when significantly cheaper and presumably less accurate models are used. However, the rightmost column presents a very different picture – it reports the total number of segments that would not have been retrieved using a bag-of-words first-stage ranker due to having *no overlapping query terms* with the segments. In other words, although the ASR model does not appear to influence re-ranking effectiveness, large perturbations in the content generated from these methods could impact overall retrievability [6], at least for lexical search models.

End-to-End Retrieval. While the previous experiment highlights potential limitations of using alternative ASR models to transcribe the collection, the effectiveness is influenced more by the re-ranking model used than the transcript itself. To better understand how these effects translate to end-to-end retrieval, we select three representative models (each with different cost/quality trade-offs) and transcribe the entire 4TiB worth of audio data. After transcription,

Table 2. Effectiveness of lexical and dense retrieval after re-transcribing podcasts using three different ASR models. The top 100 segments are re-ranked using RankZephyr.

	ASR model	TSPS	Retrieval	Re-Ranking		
			R@100	RBP $\phi = 0.8$	RBP $\phi = 0.9$	NDCG@10
BM25	Spotify	–	0.515	0.569 + 0.142	0.491 + 0.174	0.473
	Silero-Small	635	$^{\downarrow}$0.348	$^{\downarrow}$0.416 + 0.306	$^{\downarrow}$0.362 + 0.370	$^{\downarrow}$0.352
	Silero-Large	437	$^{\downarrow}$0.382	0.460 + 0.238	0.396 + 0.297	$^{\downarrow}$0.382
	WhisperX-Base	115	0.500	0.542 + 0.187	0.464 + 0.237	0.458
Dense	Spotify	–	0.387	0.389 + 0.438	0.336 + 0.503	0.347
	Silero-Small	635	$^{\downarrow}$0.283	0.419 + 0.397	0.353 + 0.478	0.367
	Silero-Large	437	$^{\downarrow}$0.306	0.367 + 0.460	0.306 + 0.541	0.319
	WhisperX-Base	115	0.370	0.388 + 0.426	0.319 + 0.518	0.345

we segment the transcripts using the same methodology as the original Spotify collection.

We then index the resulting segments using both a classic inverted index (for BM25 retrieval) as well as a competitive dense retrieval model. Table 2 shows the results of end-to-end retrieval, including first stage recall (left), and early precision metrics on the re-ranked results (right). In contrast to Table 1, we observe that cheaper and faster ASR systems typically under-perform on baseline transcription. The only exception is WhisperX, which is not significantly worse. One potential reason is from bias in the judgment pool; it is clear from the RBP residuals that a significant proportion of high-ranking documents are not judged; this effect is even more pronounced in the lower half of the table (for dense retrieval). Thus, we can answer RQ2 in the affirmative: using an alternative transcription can significantly impact end-to-end retrieval and ranking, although it is not clear if these differences can be attributed to transcriptions that are *worse* – they may just be *different*. Similarly, the answer for RQ3 can be easily determined using Table 2: Dense retrieval performs significantly worse than simpler lexical retrieval (except for Silero-Small), and cannot — at least in a zero-shot setting — mitigate against the effectiveness loss induced from the transcription model choice. However, running the same experiment on the 2021 queries showed dense retrieval to outperform BM25. We attribute this difference to the out-of-pool issue, something that was also investigated by the top-ranked TU Wien team in TREC 2021 [14].

6 Failure Analysis

Finally, we present several failure cases that emerged when comparing the top-performing ASR model, WhisperX, to the Spotify baseline.

1. Incorrectly transcribing named entities is a common issue in speech recognition, which also causes a noticeable degradation in effectiveness in our

experiments. When comparing the top-retrieved segments using queries that contain named entities, we observed examples where WhisperX was more capable of transcribing named entities and/or selecting the correct homophones. A concrete example is the query *"Imran Khan career"* — In one of the relevant segments, Spotify transcribed *Imran Khan* as *American*, *Imran* as *everyone*, and *career* as *carrier*. However, WhisperX transcribed both words correctly, which boosts the rank of that segment from 28th (in the Spotify index) to 2nd.

2. Repetition is a well-documented issue in smaller language models, and we observed multiple non-relevant segments that contain repetitions of the query keywords, which boosts the ranking of the segment. For example, for the query *"podcast about podcasts"*, a non-relevant segment appears at rank 1 because the word *podcast* was erroneously repeated 106 times by the transcription algorithm.

3. Unjudged segments are a crucial but often overlooked limitation in the TREC podcast collection. Based on the (RBP) residual values in Table 2, we can observe that a large percentage of unjudged segments exist. Further investigation of the top ranking segments retrieved with WhisperX, it is clear that relevant segments are unjudged, which means they were not ranked highly by any pooled system in the initial TREC experiments. In the future, having participants use transcripts generated from a variety of ASR models would produce a more reusable collection.

7 Conclusion and Future Work

In this work, we examined the effects of transcript quality on podcast retrieval. We hypothesized that effectiveness might be improved by simply creating more accurate transcripts using newer, state-of-the-art audio-to-text models. However, our experiments show that variance in transcriptions generally leads to a loss in effectiveness, even when using state-of-the-art models. We attribute this to a bias primarily towards the original transcription method used in the judgment pool, and to repetition/hallucination in generative ASR models. In future work, we plan to explore the impact of transcription errors on retrieval further, with a specific focus on the issues identified in Section 6. We also plan to explore additional quality and cost measurements – such as modeling the exact dollar cost [20] or CO_2 emissions [33] — to provide a more holistic and complete view of current ASR options.

Acknowledgments. We thank the referees for their useful feedback. This project was supported by the Australian Research Council (Discovery Project DP220101434) and a Google research scholar award.

Disclosure of Interests. The authors have no competing interests of any sort.

Bibliography

[1] T. Akiba, H. Nishizaki, K. Aikawa, T. Kawahara, and T. Matsui. Overview of the IR for Spoken Documents Task in NTCIR-

9 Workshop. In *Proc. NTCIR*, pages 223–235, 2011. URL https://research.nii.ac.jp/ntcir/workshop/OnlineProceedings9/NTCIR/01-NTCIR9-OV-SpokenDoc-AkibaT.pdf.

[2] T. Akiba, H. Nishizaki, K. Aikawa, X. Hu, Y. Itoh, T. Kawahara, S. Nakagawa, H. Nanjo, and Y. Yamashita. Overview of the NTCIR-10 SpokenDoc-2 Task. In *Proc. NTCIR*, pages 573–587, 2013. URL https://research.nii.ac.jp/ntcir/workshop/OnlineProceedings10/pdf/NTCIR/OVERVIEW/01-NTCIR10-OV-SPOKENDOC-AkibaT.pdf.

[3] T. Akiba, H. Hishizaki, H. Nanjo, and G. J. Jones. Overview of the NTCIR-11 SpokenQuery&Doc Task. In *Proc. NTCIR*, pages 350–364, 2014. URL https://research.nii.ac.jp/ntcir/workshop/OnlineProceedings11/pdf/NTCIR/OVERVIEW/01-NTCIR11-OV-SPOKENQUERYDOC-AkibaT.pdf.

[4] T. Akiba, H. Nishizaki, H. Nanjo, and G. J. F. Jones. Overview of the NTCIR-12 SpokenQuery&Doc-2 Task. In *Proc. NTCIR*, 2016. URL http://research.nii.ac.jp/ntcir/workshop/OnlineProceedings12/pdf/ntcir/OVERVIEW/01-NTCIR12-OV-SPOKENQUERYDOC-AkibaT.pdf.

[5] G. S. Aluri, P. Greyson, and J. Delgado. Optimizing podcast discovery: Unveiling amazon music's retrieval and ranking framework. In *Proc. RecSys*, page 1036–1038, 2023. URL https://doi.org/10.1145/3604915.3610240.

[6] L. Azzopardi and V. Vinay. Retrievability: an evaluation measure for higher order information access tasks. In *Proc. CIKM*, pages 561–570, 2008. https://doi.org/10.1145/1458082.1458157. URL https://doi.org/10.1145/1458082.1458157.

[7] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Proc. NeurIPS*, 33:12449–12460, 2020. URL https://proceedings.neurips.cc/paper/2020/file/92d1e1eb1cd6f9fba3227870bb6d7f07-Paper.pdf.

[8] M. Bain, J. Huh, T. Han, and A. Zisserman. WhisperX: Time-accurate speech transcription of long-form audio. In *Proc. Interspeech*, 2023. URL https://www.isca-archive.org/interspeech_2023/bain23_interspeech.pdf.

[9] B. Carterette, R. Jones, G. F. Jones, M. Eskevich, S. Reddy, A. Clifton, Y. Yu, J. Karlgren, and I. Soboroff. Podcast metadata and content: Episode relevance and attractiveness in ad hoc search. In *Proc. SIGIR*, pages 2247–2251, 2021. https://doi.org/10.1145/3404835.3463101.

[10] A. Clifton, S. Reddy, Y. Yu, A. Pappu, R. Rezapour, H. Bonab, M. Eskevich, G. Jones, J. Karlgren, B. Carterette, and R. Jones. 100,000 podcasts: A spoken English document corpus. In *Proc. COLING*, pages 5903–5917, 2020. URL https://aclanthology.org/2020.coling-main.519.

[11] M. Eskevich, G. J. Jones, S. Chen, R. Aly, R. J. Ordelman, and M. Larson. Search and hyperlinking task at mediaeval 2012. In *Proc. MediaEval*, 2012. URL https://ceur-ws.org/Vol-927/mediaeval2012_submission_14.pdf.

[12] M. Eskevich, R. Aly, R. J. Ordelman, D. N. Racca, S. Chen, and G. J. Jones. SAVA at MediaEval 2015: Search and anchoring in video archives. In *Proc. MediaEval*, 2015. URL https://ceur-ws.org/Vol-1436/Paper11.pdf.

[13] J. S. Garofolo, C. G. Auzanne, and E. M. Voorhees. The TREC spoken document retrieval track: a success story. In *Content-Based Multimedia Information Access-Volume 1*, pages 1–20. 2000. URL https://dl.acm.org/doi/10.5555/2835865.2835867.

[14] S. Hofstätter, M. Sertkan, and A. Hanbury. TU Wien at TREC DL and Podcast 2021: Simple Compression for Dense Retrieval. In *Proceedings of Text REtrieval Conference (TREC)*, 2021.

[15] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Sys.*, 20(4):422–446, 2002. https://doi.org/10.1145/582415.582418. URL https://doi.org/10.1145/582415.582418.

[16] G. J. Jones. About sound and vision: CLEF beyond text retrieval tasks. *Information Retrieval Evaluation in a Changing World: Lessons Learned from 20 Years of CLEF*, pages 307–329, 2019. URL https://link.springer.com/chapter/10.1007/978-3-030-22948-1_13.

[17] R. Jones, B. Carterette, A. Clifton, M. Eskevich, G. J. Jones, J. Karlgren, A. Pappu, S. Reddy, and Y. Y. TREC. Trec 2020 podcasts track overview. In *Proc. TREC*, 2021. URL https://trec.nist.gov/pubs/trec29/papers/OVERVIEW.P.pdf.

[18] R. Jones, H. Zamani, M. Schedl, C.-W. Chen, S. Reddy, A. Clifton, J. Karlgren, H. Hashemi, A. Pappu, Z. Nazari, L. Yang, O. Semerci, H. Bouchard, and B. Carterette. Current challenges and future directions in podcast information access. In *Proc. SIGIR*, pages 1554–1565, 2021. https://doi.org/10.1145/3404835.3462805.

[19] M. Larson, M. Eskevich, R. Ordelman, C. Kofler, S. Schmiedeke, and G. Jones. Overview of MediaEval 2011 rich speech retrieval task and genre tagging task. In *Proc. MediaEval*, 2011. URL https://ceur-ws.org/Vol-807/Larson_RSR_and_Genre_me11overview.pdf.

[20] K. Liao, A. Moffat, M. Petri, and A. Wirth. A cost model for long-term compressed data retention. In *Proc. WSDM*, pages 241–249, 2017. URL https://doi.org/10.1145/3018661.3018738.

[21] X. Lu, A. Moffat, and J. S. Culpepper. The effect of pooling and evaluation depth on ir metrics. *Inf. Retr.*, 19(4):416–445, 2016.

[22] C. Macdonald and N. Tonellotto. Declarative experimentation in information retrieval using PyTerrier. In *Proc. ICTIR*, 2020. URL https://dl.acm.org/doi/10.1145/3409256.3409829.

[23] C. Macdonald, N. Tonellotto, S. MacAvaney, and I. Ounis. Pyterrier: Declarative experimentation in python from BM25 to dense retrieval. In *Proc. CIKM*, pages 4526–4533, 2021. URL https://dl.acm.org/doi/10.1145/3459637.3482013.

[24] A. Moffat and J. Zobel. Rank-biased precision for measurement of retrieval effectiveness. *ACM Trans. Inf. Sys.*, 27(1), 2008. https://doi.org/10.1145/1416950.1416952.

[25] N. Muennighoff, N. Tazi, L. Magne, and N. Reimers. MTEB: massive text embedding benchmark. *arXiv:2210.07316*, 2022. https://doi.org/10.48550/ARXIV.2210.07316. URL https://arxiv.org/abs/2210.07316.

[26] R. Nogueira, Z. Jiang, R. Pradeep, and J. Lin. Document ranking with a pretrained sequence-to-sequence model. In *Proc. EMNLP (Findings)*, pages 708–718, 2020. URL https://aclanthology.org/2020.findings-emnlp.63/.

[27] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur. Librispeech: An ASR corpus based on public domain audio books. In *Proc. ICASSP*, pages 5206–5210, 2015. https://doi.org/10.1109/ICASSP.2015.7178964.

[28] P. Pecina, P. Hoffmannova, G. J. Jones, Y. Zhang, and D. W. Oard. Overview of the CLEF-2007 cross-language speech retrieval track. In *Proc. CLEF*, pages 674–686, 2008. URL https://ceur-ws.org/Vol-1173/CLEF2007wn-CLSR-PecinaEt2007.pdf.

[29] R. Pradeep, S. Sharifymoghaddam, and J. Lin. RankZephyr: Effective and robust zero-shot listwise reranking is a breeze! *arXiv:2312.02724*, 2023. URL https://arxiv.org/abs/2312.02724.

[30] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever. Robust speech recognition via large-scale weak supervision. In *Proc. ICML*, pages 28492–28518, 2023. URL https://proceedings.mlr.press/v202/radford23a.html.

[31] R. Rezapour, S. Reddy, R. Jones, and I. Soboroff. What makes a good podcast summary? In *Proc. SIGIR*, pages 2039–2046, 2022. URL https://doi.org/10.1145/3477495.3531802.

[32] S. Robertson and H. Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Found. Trnd. Inf. Retr.*, 3(4):333–389, 2009. URL https://www.nowpublishers.com/article/Details/INR-019.

[33] H. Scells, S. Zhuang, and G. Zuccon. Reduce, Reuse, Recycle: Green information retrieval research. In *Proc. SIGIR*, pages 2825–2837, 2022. URL https://doi.org/10.1145/3477495.3531766.

[34] S. Schmiedeke, P. Xu, I. Ferrané, M. Eskevich, C. Kofler, M. A. Larson, Y. Estève, L. Lamel, G. J. Jones, and T. Sikora. Blip10000: A social video dataset containing spug content for tagging and retrieval. In *Proc. MMSys*, pages 96–101, 2013. URL https://dl.acm.org/doi/10.1145/2483977.2483988.

[35] D. Spina, J. R. Trippas, L. Cavedon, and M. Sanderson. Extracting audio summaries to support effective spoken document search. *J. Assoc. Inf. Sci. Technol.*, 68(9):2101–2115, 2017. https://doi.org/https://doi.org/10.1002/asi.23831.

[36] S. Xiao, Z. Liu, P. Zhang, and N. Muennighof. C-pack: Packaged resources to advance general Chinese embedding. In *Proc. SIGIR*, pages 641–649, 2023. URL https://dl.acm.org/doi/10.1145/3626772.3657878.

[37] S. Zhuang, H. Zhuang, B. Koopman, and G. Zuccon. A setwise approach for effective and highly efficient zero-shot ranking with large language models. In *Proc. SIGIR*, pages 38–47, 2024. URL https://doi.org/10.1145/3626772.3657813.