An efficient TF-IDF based Query by Example Spoken Term Detection

Akanksha Singh[*†], Vipul Arora[*], Yi-Ping Phoebe Chen[†]
[*]Department of Electrical Engineering, Indian Institute of Technology Kanpur, India
[†]Department of Computer Science, La Trobe University, Melbourne, Australia
{akankss20@iitk.ac.in, vipular@iitk.ac.in, Phoebe.Chen@latrobe.edu.au}

***Abstract*—In the present research, we tackle the problem of query by example spoken term detection (QbE-STD) in the zero-resource scenario. State-of-the-art methods typically use dynamic temporal warping (DTW) to match templates. In response, a novel approach is proposed that leverages term frequency and inverse document frequency (TF-IDF) to search over concise discrete representations of audio, obtained through advanced audio representation learning techniques. TF-IDF not only accelerates the search process but also enhances accuracy as compared to DTW, offering a balanced solution for navigating vast audio databases with improved efficiency and precision in retrieval results.**

Through rigorous experiments, the comparative results of retrieval performance using TF-IDF and DTW shed light on the method's efficacy in the context of Query-by-Example Spoken Term Detection (QbE-STD). This proposed approach showcases a promising direction for addressing the challenges associated with both the speed and accuracy of searching within large-scale audio datasets.

***Index Terms*—Discrete representations, Dynamic time warping (DTW), Inverse document frequency (IDF), Term frequency (TF), Retrieval, STD, QbE.**

I. Introduction

Detecting individual spoken queries within audio collections using Query-by-Example Spoken Term Detection (QbE-STD) is challenging, due to a variety of factors such as speaker differences, environmental conditions, and language-specific variations. Traditional approaches combine Automatic Speech Recognition (ASR) systems [1] with text-based retrieval techniques, which require a large volume of annotated spoken data for effective detection. However, the annotation procedure is time-consuming and requires linguistic competence, making it especially difficult to retrieve spoken documents in languages with few or no annotations. An alternate approach is pattern discovery, which seeks to identify similarities between spoken terms directly from acoustic feature representations. This method distinguishes itself by not relying on annotations, allowing for adaptability across languages and the capability to handle tasks without extensive language-specific labeled data.

When a spoken language is unknown (or, conversely, when numerous languages may occur), or when resources are insufficient to create reliable ASR systems [2], [3] pattern discovery techniques become crucial for searching untranscribed, multilingual, and acoustically unconstrained spoken materials.

The recent approaches for the spoken term detection complete the job in two stages. First, the audio signal is used to create an acoustic feature representation that highlights the spoken content regardless of the speaker or environmental variations. In the second stage, the features extracted are used to capture the likelihood between the spoken query and spoken documents by applying pattern discovery algorithms. In the context of acoustic feature representations, spectral, posterior, and bottleneck characteristics have been extensively studied for the STD task. Dynamic Temporal Warping (DTW) -centric methods [4]–[8] capture the similarity between spoken terms by aligning their acoustic features in time. However, global alignment presents challenges because it computes the optimal alignment path globally and ignores local alignments due to variability issues. Segmental DTW [9] was introduced to address this problem by focusing on segmental-level similarities. The DTW-centric systems despite being efficient, face limitations in terms of time complexity during retrieval and also suffer from the speech variability challenges that exist in natural speech. The lack of efficient indexing techniques results in linear searches across the entire database, impeding scalability, especially with short queries. Locality-sensitive hashes [10] and subspace-indexing attempt to tackle these issues for speech data storage, still scalability remains a challenge.

The proposed method also solves the problem in two stages and effectively addresses the scalability challenges associated with these systems. In the first stage, a self-supervised learning framework for representations from raw audio data, such as Wav2Vec2.0 [11] is employed to obtain concise discrete tokens. In the second phase, after obtaining the discrete tokens from unlabelled speech, term frequency and inverse document frequency (TF-IDF) technique [12], [13] is used to index and retrieve the spoken queries from the audio database. The effectiveness, resilience to noise, and readability of TF-IDF make it a flexible tool that can be used to extract important terms from documents in a variety of domains. This distinguishes it from DTW [14], which has drawbacks like quadratic time complexity, noise sensitivity, and parameter dependency.

The main contribution of this research is the introduction of a refined TF-IDF based framework for audio query search. This system is specifically designed to navigate token sequences created from various audio representation learning approaches, building upon earlier works [15], [16]. When compared to earlier audio search techniques, the improvements made in this new version of the query search framework are

979-8-3503-5409-6/24/$31.00 ©2024 IEEE
DOI 10.1109/CAI59869.2024.00039

evident in better mean average precision (MAP), recall, actual term weight value (ATWV), and decreased time complexity per query search as compared to other approaches.

II. Method

A. Audio tokenization techniques

Audio tokenization, a critical step in the analysis of audio data, involves the conversion of raw audio signals into a structured sequence of meaningful tokens. This process facilitates the handling and interpretation of complex audio information.

Wav2vec2.0 learns representations of speech audio through a contrastive learning approach. In the same way as masked language modeling, this approach encodes spoken audio using a multi-layer convolutional neural network and then masks portions of the resulting latent speech representations. In order to create contextualized representations, the latent representations are loaded into a transformer network. The model is then trained using a contrastive task in which the true latent has to be separated from distractions. For the purpose of representing the latent representations in the contrastive task, it learns discrete speech units using a gumbel softmax.

B. Pattern Matching Algorithms

From the token sequences

$$\mathcal{A} = \{a_i \mid 1 < i \leq N\} \tag{1}$$

$$a_i = \{(\tau_1, \tau_2, \ldots, \tau_n) \mid 1 < n \leq M\} \tag{2}$$

where $\mathcal{A}$ is a set of audio files, a_i portrays token sequence for a single audio file from the collection and τ_n stands for a token from the set of M unique tokens. The linguistic content of every audio file stored in the database is represented by these token sequences. After that, a TF-IDF matrix is produced to assess each token's importance concerning its recurrence within a particular audio clip as well as its broader context throughout the corpus. The TF-IDF matrix combines Term Frequency (TF) [17] and Inverse Document Frequency (IDF) [18] values. TF quantifies a token's frequency within a document, whereas IDF assesses its significance throughout the entire database.

The TF of each token τ_n in a document is determined as the ratio of the number of occurrences of τ_n to the total number of tokens in the document.

$$\text{TF}(\tau_n, a_i) = \frac{\text{Frequency of } \tau_n \text{ in the document}}{\text{Total number of tokens in the document}}$$

The IDF for each token τ_n is obtained by calculating the logarithm of the ratio of the total number of documents in the corpus to the number of documents containing τ_n.

$$\text{IDF}(\tau_n) = \log\left(\frac{\text{Total number of documents in the corpus}}{\text{Frequency of documents containing } \tau_i}\right)$$

The TF-IDF score for each token τ_i in a document is then obtained by multiplying the TF and IDF values:

$$\text{TF-IDF}(\tau_n, a_i) = \text{TF}(\tau_n, a_i) \times \text{IDF}(\tau_n) \tag{3}$$

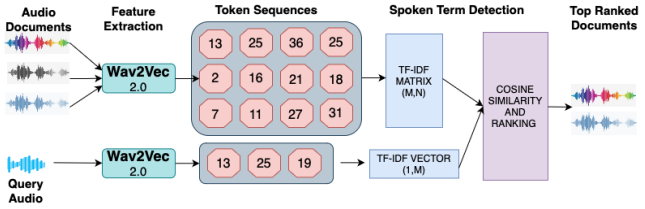

Fig. 1. Proposed Method: The TF-IDF matrix (M × N) obtained from the token sequences generated for an audio database. Similar process is undergone by the query that is followed by a similarity operation between the TF-IDF vector(1,M) of the query and the TF-IDF matrix (M, N) of the audio database. This operation determines the top-ranked audio documents in terms of cosine similarity.

Then the TF-IDF matrix $\mathcal{M}$ with dimensions $M \times N$ is formed by repeating this method for all tokens and documents, where N represents the total number of audio files and M represents the count of unique tokens across the corpus.

Fig. 1 depicts the specific approach mentioned where, the discrete token sequences of the audio database are obtained from pre-trained Wav2Vec2.0 model and the subsequent formation of the TF-IDF matrix of size (M × N), that allows the model to capture the importance of tokens in the audio data, emphasizing specific tokens that are significant within the context of the entire set of audio documents.

C. Query Search

To maintain uniformity during the query search phase, smaller audio queries are pre-processed using the pre-trained Wav2Vec2.0 model to provide token sequences that correspond to audio database tokens. The generated token sequences are treated as independent documents, and each token's TF-IDF score is calculated by considering it as a term. Using this method, one may describe the query as a TF-IDF vector in the same vector space as the dataset. The next step is to use a similarity metric, such as cosine similarity, to evaluate how similar the query's TF-IDF score is to the dataset entries' values. The top-ranked token sequences or audio segments can be retrieved from the dataset by identifying and sorting probable matches using the resulting similarity scores. For comparative analysis, DTW is also implemented on these query and audio token sequences. TF-IDF provides a clear depiction of word relevance in documents in relation to the database as a whole, while DTW is less convenient and computationally expensive, particularly for large datasets. Moreover, even in the face of unpredictability or noise, TF-IDF's robustness to small token fluctuations guarantees consistent performance.

III. Experimental Setup

A. Databases

- Hindi- The dataset comprises approximately 100 hours of 44.1kHz speech data, incorporating around 17.6 hours from the Common Voice Hindi (CVH) dataset. Annotators made enhancements, including corrections, transliterations, and improvements. An additional 78.21 hours

are sourced from the Prasar Bharati archive, covering diverse scenarios and are manually annotated. Time aligned segments for keywords are done with the help of TeLeS [19]. Thirty queries, each lasting roughly five seconds, are selected from the aligned word segments, with an average relevant occurrence of five, in order to aid in the evaluation process.

- English- LibriSpeech dataset, derived from LibriVox's public domain audio books, is also used [20]. The dev-clean subset of LibriSpeech is utilized to make search corpus consisting of approximately 5 hours of 16kHz audio with 2703 utterances. Time aligned word segmentation on the data is performed using kaldi [21]. A query approximately five seconds long has, on average, about 5 relevant occurrences in the database and thirty such queries are chosen to be searched.

B. Overview of the proposed approach

The large audio database is tokenized using Wav2Vec2.0 [11] pre-trained model and then, clustering is done on the frame level features to get concise, discrete and unique sequence of tokens. The model uses 256 unique tokens to capture the inherent features of the audio provided. Fig. 1 shows the form of token sequences generated for the audio database. For spoken query also the same approach is performed to get its token sequence. Inspired from the natural language processing techniques the token sequences are considered as sentences and the individual tokens as words, so an application of TF-IDF becomes possible on these audio token sequences.

The proposed method is compared with the Unsupervised query-by-example spoken term detection using segment-based Bag of Acoustic Words (BoAW + DTW) [22]. Inspired by the Bag of Words (BoW) approach, this paper presents an unsupervised framework for identifying spoken terms in large audio collections using a segment-based BoAW. The approach uses DTW to recover sequence information during retrieval, while disregarding it to efficiently index the database. The proposed method highlights the significance of individual audio segments within the total dataset using TF-IDF. This improves the system's ability to precisely locate particular audio queries. BoAW technique, on the other hand, while good, does not place as much emphasis on this feature, potentially making its representation of audio segments less comprehensive during the search process. Moreover, TF-IDF offers distinct advantages by obviating the necessity for the entire sequence during pattern matching. Instead, TF-IDF focuses on token importance, which effectively gauges similarity with the corresponding document.

C. Evaluation Metrics

The ATWV score, computed over a set of queries, is determined by the hit rate, miss rate, and false alarm rate. The hit rate (P_{hit}) is calculated as the ratio of correctly identified spoken terms $(N_{correct})$ to the actual number of terms in the corpus (N_{actual}). A higher hit rate suggests that the system effectively retrieves a substantial proportion of the relevant audio segments, minimizing the risk of overlooking pertinent content. The miss rate (P_{miss}) is the complement of the hit rate, representing the proportion of undetected spoken terms. The false alarm rate (P_{fa}) is determined by the ratio of falsely detected documents (N_{false}) to the total number of documents searched (N_{NT}).

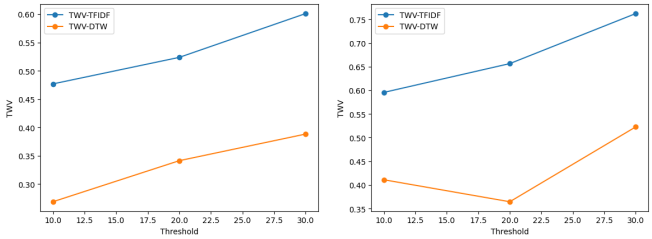

Fig. 2. Comparison of Term Weight Values for DTW and TF-IDF. The left plot shows the performance on English Data and the right one displays the performance on Hindi Data.

$$\mathbf{P_{hit}}(\tau,\lambda) = \frac{N_{correct}(\tau,\lambda)}{N_{actual}(\tau)} \tag{4}$$

$$\mathbf{P_{miss}}(\tau,\lambda) = 1 - P_{hit} \tag{5}$$

$$\mathbf{P_{fa}}(\tau,\lambda) = \frac{N_{false}(\tau,\lambda)}{N_{NT}(\tau)} \tag{6}$$

where, (τ) is the particular term and λ is the decision threshold confirming the spoken term availability in the retrieved document. So,

$$\mathbf{ATWV}(\lambda) = 1 - \frac{F}{Q} \tag{7}$$

where,
Q is the total number of terms in the search corpus and

$$F = \sum_{\forall \tau \in Q} P_{miss}(\tau,\lambda) + \beta \cdot P_{fa}(\tau,\lambda)$$

$$\beta = \frac{\omega_{miss} \times P_{target}}{\omega_{miss} \times P_{target} + \omega_{fa} \times (1 - P_{target})}$$

The β represents the weight factor computed based on the cost of the miss (ω_{miss}) and false rates (ω_{fa}) It is computed as 0.009 by applying $\omega_{fa} = 1, \omega_{miss} = 100$ and $P_{target} = 0.0001$. In the proposed approach, the β is constant for all terms. Fig. 2 compares the performance of DTW and TF-IDF for ATWV scores. It is evident that our technique performs better for both Hindi and English, achieving a maximum score of 0.76 for Hindi and 0.60 for English. In an ideal QbE-STD system, the ATWV score approaches unity.
The precision of the document D at the k_{th} position in the retrieved results is measured based on the number of hits in the retrieved result against the total number of documents retrieved till k_{th} position.

TABLE I
COMPARISON OF PERFORMANCE FOR PRECISION AND MAP SCORES FOR DIFFERENT SEGMENT SIZES OF QUERY.

Method	P@1			P@3			P@5			MAP		
	0.8s	1s	1.2s	0.8s	1s	1.2s	0.8s	1s	1.2s	0.8s	1s	1.2s
BoAW+DTW	0.5357	0.733	0.733	0.4405	0.5667	0.633	0.3643	0.4400	0.5200	0.329	0.4570	0.5051
TF-IDF	**0.5714**	**0.857**	**0.875**	**0.4809**	**0.576**	**0.649**	**0.3742**	**0.4441**	**0.5472**	**0.3375**	**0.4575**	**0.558**

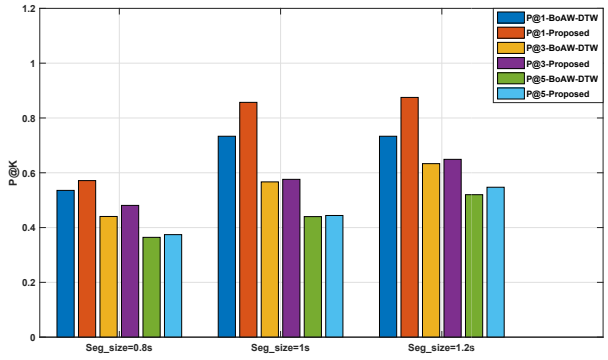

Fig. 3. Comparison of P@K scores between BoAW + DTW and proposed approach for different query groups based on segment duration.

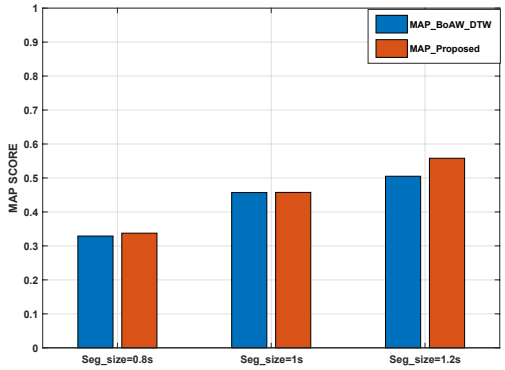

Fig. 4. Comparison of MAP scores between BoAW + DTW and proposed method for different query groups based on segment duration.

$$\mathbf{Precision(D_k)} = \frac{hit}{hit + false\ alarm} \tag{8}$$

Here, a "hit" is considered true if the detected document intersects with the actual relevant document; otherwise, it is labeled as a false alarm. The average precision for a query term with a given threshold (λ) is then calculated by averaging the precision values across all documents in the ranked retrieval result:

$$\mathbf{AveragePrecision}(\tau, \lambda) = \frac{\sum_{K=1}^{N} \text{Precision}(D_k)}{N} \tag{9}$$

The MAP score is computed by assessing the precision at each hit in the retrieved results and then averaging these precision values across all query terms. An optimal STD system aims to maximize the MAP to attain a perfect score of unity.

$$\mathbf{MAP}(\lambda) = \frac{\sum_{\forall \tau \in Q} \text{Average Precision}(\tau, \lambda)}{Q} \tag{10}$$

This formula captures the precision at various positions in the retrieved results and provides a comprehensive assessment of the method's performance across multiple query terms.

Mean Reciprocal Rank (MRR) is calculated by taking the reciprocal of the rank of the first correctly identified relevant item. In the context of the audio search system, this means determining the position at which the system correctly identifies a relevant audio segment among the retrieved results. The reciprocal of this rank is then computed. The MRR score is obtained by averaging these reciprocal ranks across multiple queries. A higher MRR score indicates that, on average, the system tends to identify relevant audio segments closer to the top of the ranked list.

TABLE II
EVALUATION METRICS FOR PERFORMANCE ANALYSIS FOR SPOKEN TERM DETECTION

Results for English			
Method	MAP	ATWV	MRR
DTW	0.24	0.33	0.14
TF-IDF	**0.55**	**0.55**	**0.35**
Results for Hindi			
DTW	0.36	0.36	0.20
TF-IDF	**0.69**	**0.62**	**0.41**

IV. RESULTS

Table I shows the effectiveness of the proposed method over BoAW + DTW on a test set of 800 files taken from TIMIT corpus using 30 queries of varying lengths [23]. A query has, on average, about 5 relevant occurrences in the database. Hence, the evaluation metrics used for comparison are: i) **P@1**: Average precision of the top result returned by the system; ii) **P@3**: Average precision of the top 3 results; iii) **P@5**: Average precision of the top 5 results and MAP. Fig. 3 and fig. 4 show that the suggested method performs comparably better than BoAW + DTW across segment sizes (0.8s, 1s, 1.2s) for P@K (K=1,3,5) and MAP, demonstrating its capacity to obtain more accurate audio files for a query. The method's greater ability to weigh term importance leads to its robust performance, making it a good choice for information retrieval tasks. BoAW + DTW is reasonable, but it falls behind TF-IDF, suggesting that it may not be able to capture complex links between terms and documents.

Fig. 5 and fig. 6 shows the recall rate at top 10, 20 and 30 files retrieved from the 5 relevant files per query in the audio dataset. Table II presents a comparative analysis of two distinct methods, Dynamic Time Warping (DTW) and (TF-IDF), within the domain of audio search for both English and

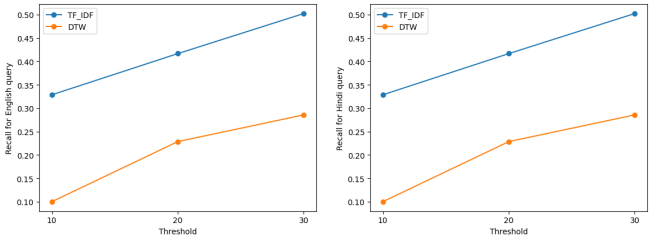

Fig. 5. Comparison of the Recall Rate, or the percentage of query-relevant documents that are successfully retrieved for DTW vs TF-IDF. The left plot depicts the performance in English, while the right plot depicts the performance in Hindi.

RECALL FOR ENGLISH WORDS									
METHOD	BEAUTIFUL			GLORIOUS			VANITY		
DTW	R@10	R@20	R@30	R@10	R@20	R@30	R@10	R@20	R@30
	0.2	0.2	0.2	0.2	0.2	0.4	0.0	0.0	0.0
TF-IDF	**0.8**	**0.8**	**0.8**	**0.4**	**0.8**	**0.8**	**0.4**	**0.8**	**0.8**
RECALL FOR HINDI WORDS									
	कारण			किसी			अब		
DTW	R@10	R@20	R@30	R@10	R@20	R@30	R@10	R@20	R@30
	0.0	0.0	0.0	0.66	0.66	0.66	0.0	0.0	0.25
TF-IDF	**1.0**	**1.0**	**1.0**	0.66	0.66	0.66	**0.75**	**1.0**	**1.0**

Fig. 6. Comparison of performance between DTW and TF-IDF for Recall rates at top 10, 20 and 30 files retrieved from the 5 relevant files for a sample of words in English and Hindi.

Hindi datasets. As can be observed in the case of Hindi, TF-IDF consistently outperforms DTW, with higher values across all metrics. The MAP for TF-IDF in Hindi is notably superior, indicating better precision in ranked results. Similarly, ATWV and MRR also reflect TF-IDF's superior performance. Higher recall rates suggest that most of the relevant files are being retrieved successfully by the TF-IDF based method. The results for English follow a similar trend, with TF-IDF surpassing DTW in all evaluated metrics. On comparing the two languages, TF-IDF exhibits better performance in Hindi than in English, as evidenced by higher values in MAP, ATWV, and MRR.

V. CONCLUSIONS AND FUTURE WORK

This paper presents an efficient QbE-STD system that improves search efficiency by applying TF-IDF to the discrete tokens of audio generated from advanced audio representation learning techniques. Extensive experiments demonstrate the superior performance of the proposed method compared to BoAW + DTW. The experimental results also demonstrate that TF-IDF based QbE-STD not only accelerates the search process but also enhances accuracy with less time complexity over the search space as compared to DTW and its variants, providing a balanced solution for navigating vast audio datasets with improved efficiency and precision in retrieval results. Further exploring the search approach with an integrated audio tokenization process and experimenting on several other languages opens a new direction of research.
The codes are available at https://github.com/madhavlab/2023_std_akankss.git.

VI. ACKNOWLEDGMENTS

This work was supported by a research grant from MeitY, Government of India.

REFERENCES

[1] X. Huang, A. Acero and H. -W. Hon, —Spoken Language Processing: a guide to theory, algorithm, and system development‖, Prentice Hall, (2001)
[2] Chunan Chanand, Lin-Shan Lee,"Model-Based Unsupervised Spoken Term Detection with Spoken Queries," IEEE Transactions on Audio, Speech Language Processing, vol. 21, no. 7, pp. 1330–1342, 2013.
[3] Haipeng Wang, Tan Lee, Cheung-Chi Leung, Bin Ma, and Haizhou Li, "Using parallel tokenizers with DTW matrix combination for low-resource spoken term detection," in ICASSP, 2013, pp. 8545–8549.
[4] Mantena, Gautam, Sivanand Achanta, and Kishore Prahallad. "Query-by-example spoken term detection using frequency domain linear prediction and non-segmental dynamic time warping." IEEE/ACM Transactions on Audio, Speech, and Language Processing 22.5 (2014): 946-955.
[5] A. Park and J. R. Glass, "Towards unsupervised pattern discovery in speech," in IEEE Workshop on Automatic Speech Recognition and Understanding, 2005. IEEE, 2005, pp. 53–58
[6] Y. Zhang and J. R. Glass, "Towards multi-speaker unsupervised speech pattern discovery," in 2010 IEEE international conference on acoustics, speech and signal processing. IEEE, 2010, pp. 4366–4369.
[7] V. Gupta, J. Ajmera, A. Kumar, and A. Verma, "A language independent approach to audio search," in Annual Conference of the International Speech Communication Association, 2011.
[8] P. D. Karthik, M. Saranya, and H. A. Murthy, "A fast query-by- example spoken term detection for zero resource languages," in 2016 International conference on signal processing and commu- nications (SPCOM). IEEE, 2016, pp. 1–5.
[9] A. S. Park and J. R. Glass, "Unsupervised pattern discovery in speech," IEEE Transactions on Audio, Speech, and Language Processing, vol. 16, no. 1, pp. 186–197, 2007.
[10] Gionis, Aristides and Indyk, Piotr and Motwani, Rajeev and others. "Similarity search in high dimensions via hashing." vol. 99, no. 6, pp. 518–529, 1999.
[11] Baevski, Alexei, et al. "wav2vec 2.0: A framework for self-supervised learning of speech representations." Advances in neural information processing systems 33 (2020): 12449-12460.
[12] Luhn, Hans Peter. "A Statistical Approach to Mechanized Encoding and Searching of Literary Information." IBM J. Res. Dev. 1 (1957): 309-317.
[13] Ke, Weimao. (2013). Information-theoretic Term Weighting Schemes for Document Clustering. Proceedings of the ACM/IEEE Joint Conference on Digital Libraries. 1-10. 10.1145/2467696.2467698.
[14] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," IEEE transactions on acoustics, speech, and signal processing, vol. 26, no. 1, pp. 43– 49, 1978
[15] P. Bachman, R. D. Hjelm, and W. Buchwalter. Learning representations by maximizing mutual information across views. In Proc. of NeurIPS, 2019
[16] A. Baevski, S. Schneider, and M. Auli. vq-wav2vec: Self-supervised learning of discrete speech representations. In Proc. of ICLR, 2020
[17] H. P. Luhn, "The Automatic Creation of Literature Abstracts," in IBM Journal of Research and Development, vol. 2, no. 2, pp. 159-165, Apr. 1958, doi: 10.1147/rd.22.0159.
[18] Jones, Karen. (2004). IDF term weighting and IR research lessons. Journal of Documentation - J DOC. 60. 521-523. 10.1108/00220410410560591.
[19] https://arxiv.org/abs/2401.03251
[20] V. Panayotov, G. Chen, D. Povey and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, QLD, Australia, 2015, pp. 5206-5210, doi: 10.1109/ICASSP.2015.7178964.
[21] Povey, Daniel, et al. "The Kaldi speech recognition toolkit." IEEE 2011 workshop on automatic speech recognition and understanding. No. CONF. IEEE Signal Processing Society, 2011.

[22] B. George and B. Yegnanarayana, ”Unsupervised query-by-example spoken term detection using segment-based Bag of Acoustic Words,” 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 2014, pp. 7133-7137, doi: 10.1109/ICASSP.2014.6854984.
[23] Garofolo, John S., et al. ”Acoustic-Phonetic Continuous Speech Corpus.”