Attention based Audio Embeddings for Query-by-Example

Anup Singh[1,2], Kris Demunynck[1], Vipul Arora[2]
[1]Ghent University, [2]Indian Institute of Technology-Kanpur

Full paper link

Introduction

- Audio fingerprinting applications include music recognition, broadcast monitoring, second screen applications and etc.
- Conventional audio fingerprinting systems rely on handicraft audio features, failing to deliver accurate results at high noise and reverberation levels.
- A well-known Shazam[1] method find spectral peaks in spectrograms as robust features and further transforms them into hash codes to expedite search.
- An ideal audio fingerprinting system must generate robust and compact fingerprints in computationally efficient manner to be scalable.

Contributions

- We deploy deep learning (CNN) to compute compact and robust audio fingerprints.
- We explore contrastive learning framework by creating pairs of clean audio segments and its corresponding distorted version.
- Inspired by Shazam method, we attempt to locate the salient peaks/patches in the CNN features using the proposed the spectral-temporal attention mechanism.
- Spectral-temporal attentions provides discriminative audio fingerprints.
- We devise our own custom resnet-like CNN architecture.
- We propose a simple yet effective subsequence search to precisely locate query timestamp.

Approach

- Contrastive loss:

$$\mathcal{L}_c = -\log \frac{e^{(\mathcal{F}_\theta(x)\cdot\mathcal{F}_\theta(x^+))/\tau}}{e^{(\mathcal{F}_\theta(x)\cdot\mathcal{F}_\theta(x^+))/\tau} + \sum_{x^-} e^{(\mathcal{F}_\theta(x)\cdot\mathcal{F}_\theta(x^-))/\tau}}$$

- Spectral-Temporal Attention:

$$a^{temp} = \mathrm{softmax}(X^T W_{temp})$$
$$a^{spect} = \mathrm{softmax}(X^T W_{spect})$$
$$A = a^{spect} \otimes a^{temp} \times S$$
$$X' = A * X$$

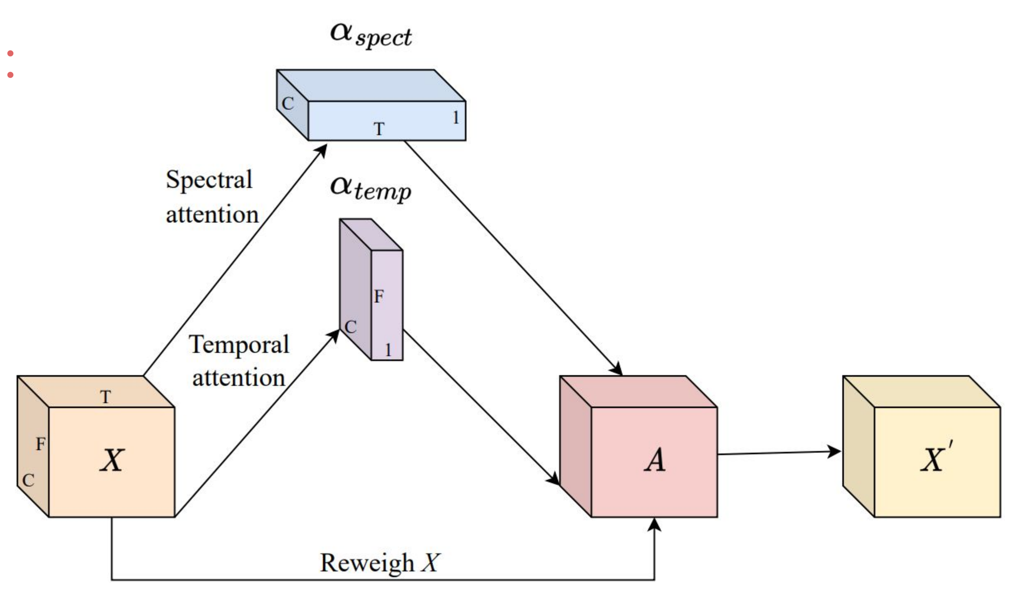

Figure 1: Spectral-Temporal Attention mechanism

- Subsequence Search:
  - Generate multiple sequence candidates C_i with their starting indice as $I_i = I_m - m$, where I_m is the retrieved index at m^{th} position.
  - Select I_i (time offset) with maximum agreement among candidates.

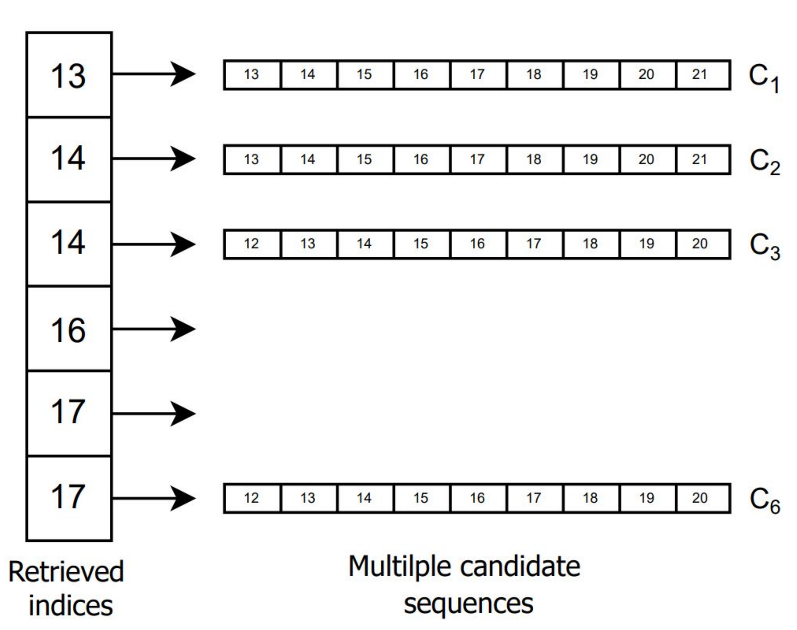

Figure 2: Subsequence search

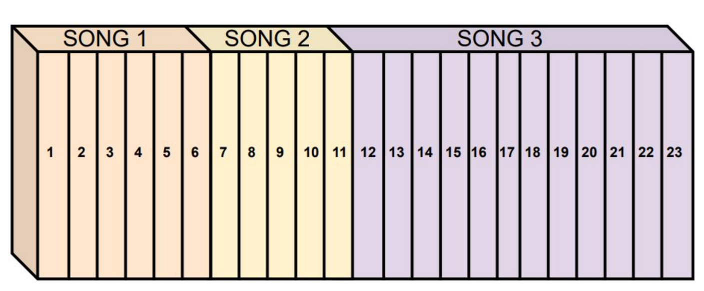

Figure 3: Index storing embeddings sequentially

Feature Encoder

- We design custom resnet-like CNN architecture.
- It consists of front-end and back-end. The Front-end consists of a CNN block with no subsampling in the spectral-temporal axis, and backend consists of sequentially stacked resnet blocks enhanced with spectral-temporal attention.

Layer	Input size	Output size
Encoder:		
CNN layer	1×64×96	32×64×96
ResBlock1	32×64×96	32×64×96
ResBlock2	32×64×96	64×32×48
...		
ResBlock6	512×4×6	1024×2×3
Flatten		6144
Projection Head:	$d*i$	$d*o$
Conv1D + ELU	128×48	128×32
Conv1D	128×32	128×1

Experiments and Results

- Database: Free Music Archival (FMA)
- Distortions: Noise, Reverberation and time offset.
- Evaluation Metric:
  - Recall @ audio-level: Coarse search
  - Recall @ segment-level: Fine-grain search, ie. located timestamp within +- 50 ms.
- Baselines: MIPS[2] and Audfprint[3]
- Indexing algorithm: Locality Sensitive Hashing (LSH)

Method	Query length(s)	0dB	5dB	10dB	15dB
Ours	0.96	**60.3**	**76.6**	**81.3**	**82.8**
MIPS		27.3	58.7	70.7	73.9
Ours	2	**66.4**	**83.5**	**86.9**	**88.0**
MIPS		39.0	69.6	76.5	78.7
Ours	3	**67.9**	**85.1**	**88.2**	**89.3**
MIPS		47.1	75.2	80.2	81.4
Ours	5	**69.5**	**87.1**	**90.5**	**91.9**
MIPS		54.7	77.3	81.8	82.8

Table 1. Top-1 hit rate (%) performance in the segment-level search for varying query lengths in noisy reverberant conditions.

Distortion	Method	0dB	5dB	10dB	15dB	
Noise	Ours	**95.0**	**98.7**	**98.9**	**99.2**	
	Audfprint	72.1	82.7	89.4	91.2	
Noise+ Reverb	Ours	**84.3**	**96.8**	**98.5**	**98.9**	
	Audfprint	64.8	79.4	87.2	92.3	
		0.2s	0.4s	0.5s	0.7s	0.8s
Reverb	Ours	**99.2**	**99.5**	**98.9**	**99.6**	**98.7**
	Audfprint	96.1	94.6	81.8	89.6	40.2

Table 2. Top-1 hit rate (%) performance in the audio-level search in different distortion conditions.

References

1. A. Wang, “The shazam music recognition service,” Communications of the ACM, vol. 49, no. 8, pp. 44–48, 2006.
2. S. Chang, D. Lee, J. Park, H. Lim, K. Lee, K. Ko, and Y. Han, “Neural audio fingerprint for high-specific audio retrieval based on contrastive learning,” in ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2021, pp. 3025–3029.
3. https://github.com/dpwe/audfprint