Overview
Apply semi-supervised method to audio tagging/retrieval.

Utilize untagged samples and reduce the needed number of expensive tagged samples.

What is Audio Tagging/Retrieval?

Audio Tagging
- Query: Audio Signal
- Output: Tags

Audio retrieval
- Query: Tags
- Output: Audio Signals

Technical Challenges: Use Inexpensive Untagged Audios

Audio samples having high-quality tags are very expensive!
Semi-supervised method utilizes inexpensive untagged audio samples!

Technical Challenges: Use Tag Co-occurrence Information

Tag co-occurrence information seems to be useful for tagging/retrieval task. But almost all of existing method cannot utilize this information.

SSCDE: Model Learning Framework

1. Extract features from audio signals and tags.
2. Generate topic space by SemiCCA.
3. Learn audio-topic model by kernel density estimation.
4. Learn tag-topic model by Multi-label SSKDE.
5. Use the learned model for annotation and retrieval.

Technical Points of SSCDE

1. Learn topic space with tagged and untagged samples:
   - SemiCCA
     
     \( \begin{aligned}
     z_n &= \lambda_n W_{x} x_n + \lambda_n W_{y} y_n \\
     \lambda_n &= M_{x} W_{xx} x_n + M_{y} W_{yy} y_n
     \end{aligned} \)

   - SSCDE

2. Propagate tag information: Multi-label SSKDE

   \( \begin{aligned}
   p(x, y) &= \frac{1}{N} \prod_{n=1}^{N} p(x_i | z_n) p(y | z_n) \\
   z_n &= \frac{1}{2} W_{xx} x_n + \ldots + t F(0)_{ij} + (1 - t) F(k - 1)_{ij}
   \end{aligned} \)

   \( \begin{aligned}
   F(0)_{ij} &= (t \mu_{yi,j} + (1 - t) N_{i,j} \mu_{y_i,j}) \\
   F_{ij} &= p(y | z_n) \\
   z_1, z_2, z_3, z_4, z_5, z_6 &\rightarrow \text{Unsupervised}
   \end{aligned} \)

   \( \begin{aligned}
   \hat{y} &= \arg\max_{y} p(y | x_q) = \arg\max_{y} p(x_q, y) \\
   \hat{x} &= \arg\max_{x} p(x | y_{\text{given}})
   \end{aligned} \)

   \( \begin{aligned}
   ^{\ddagger} p(y | z_n) &= \prod_{i=1}^{N} p(y \mid z_n) \cdot (z_n - z_i) \prod_{i=1}^{N} (z_n - z_i) \\
   p(x, y) &= \prod_{z} p(z) p(x | z) p(y | z) dz
   \end{aligned} \)

   \( \begin{aligned}
   S &\rightarrow \text{Sparse Matrix} \\
   C &\rightarrow \text{Neighboring Matrix}
   \end{aligned} \)

   \( \begin{aligned}
   S_{xx} &\rightarrow \text{Sparse Matrix}
   &\rightarrow \text{Neighboring Matrix}
   \end{aligned} \)

Experiment

Annotation performance of SSCDE is evaluated under following condition.

- **Dataset**: 2012 audio files taken from "Freesound" (http://www.freesound.org/).
  - Database of Creative Commons licensed sounds.
  - Annotated with vocabulary.
- **Evaluation condition**:
  - 2012 audio clips with WAV format.
  - 90% of clips are used for training, and the remaining 10% are used for testing.

- Tag feature: 230-dimensional binary vector.
  (Each element of the vector corresponds to specific tag.)

- Audio feature: bag-of-feature vectors extracted by the following process.
  1. Audio signals are split into half-overlapping 23ms windows, and 39-dimensional vector (including first 13 MFCC, MFCC-D, MFCC-Delta) is extracted from each window.
  2. 500 vectors are sampled from each audio signal (about 1,000,000 vectors in total).
  LBG algorithm (Linde-Buzo-Gray algorithm, algorithm for vector quantization : VQ) is applied to them, and VQ codebook (size: 1024) is obtained.

- 1024-dimensional normalized vector representing each audio signal is created by VQ.

Result

- All training samples are used as tagged samples. Our goal is to approach this performance with fewer tagged samples.
- SSCDE successfully improved the performance by utilizing untagged samples.

Use Sparse Matrix as Neighboring Matrix

- Fix the number of non-zero elements in each row, then required memory size to hold the neighboring matrix decreases.
  \( \Theta(N^2) \rightarrow \Theta(N) \)

- In this case, SSKDE is equivalent to Graph spectral method. [Joachims 2003]
- Apply the same idea to audio tagging and retrieval, then calculation needs only a few samples nearby query.

\( \begin{aligned}
\hat{y} &= \sum_{n=1}^{N} p(z_n | x_q) \cdot p(y = \text{tag} | z_n) \\
\hat{z} &= \sum_{n=1}^{N} p(z_n | x_q) \cdot p(y = \text{tag} | z_n)
\end{aligned} \)

Annotation/Retrieval task becomes equivalent to neighboring search problem (Computational complexity is \( \Theta(\log N) \))

Contact Information
E-mail: takagi@sg.es.titech.ac.jp

Jun Takagi† Yasunori Ohishi† Akisato Kimura† Masashi Sugiyama† Makoto Yamada† Hirokazu Kameoka†
†Graduate School of Information Science and Engineering, Tokyo Institute of Technology
‡NTT Communication Science Laboratories, NTT Corporation

AUTOMATIC AUDIO TAG CLASSIFICATION
VIA SEMI-SUPERVISED CANONICAL DENSITY ESTIMATION

Tools

- SSDE
- SemiCCA
- SSCDE

**Note:**
- SSCDE can cope with both tasks in the same framework!