An analysis and review of feature extraction and dimensionality reduction techniques for the visual exploration of large audio collections.

Engineering thesis

Author: Łukasz Piotrak (Student ID: s18002)

Supervisor: -

Co-Supervisor: -

Department of Information Technology,
Polish-Japanese Academy of Information Technolog

February 4, 2020

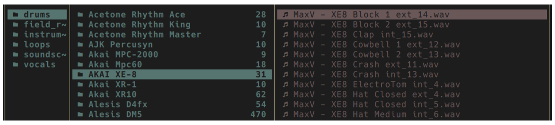
Abstract

Often when working with sound, musicians and producers start with an idea of what sound they would like to achieve. They navigate their sound libraries in search of files which might be similar to the sound in their imagination.

The current paradigm for the storage and classification of audio data is taken from the classic directory structure. This results in an attribute ontology; files are grouped together into classes (usually by directory name) i.e. "Drums", "Field recordings" and assigned a range of attributes (by means file name or tags) i.e. "Low", "Kick", "Noisy" etc.

This approach has many limitations and drawbacks when it comes to cataloging sound, e.g. The files might be mislabeled or the labels might not be an adequate description of the sound. Moreover, relationships between samples are unable to be sufficiently represented. Two audio samples might be in seperate branches of the taxonomy but be perceptually similar.

Figure 1: The typical way of exploring files.



```
■ drums      ■ Acetone Rhythm Ace         28   ♬ MaxV - XE8 Block 1 ext_14.wav
■ field_r~   ■ Acetone Rhythm King        10   ♬ MaxV - XE8 Block 2 ext_15.wav
■ instrum~   ■ Acetone Rhythm Master       7   ♬ MaxV - XE8 Clap int_15.wav
■ loops      ■ AJK Percusyn               10   ♬ MaxV - XE8 Cowbell 1 ext_12.wav
■ soundsc~   ■ Akai MPC-2000               9   ♬ MaxV - XE8 Cowbell 2 ext_13.wav
■ vocals     ■ Akai Mpc60                 18   ♬ MaxV - XE8 Crash ext_11.wav
             ■ AKAI XE-8                  31   ♬ MaxV - XE8 Crash int_13.wav
             ■ Akai XR-1                  10   ♬ MaxV - XE8 ElectroTom int_4.wav
             ■ Akai XR10                  62   ♬ MaxV - XE8 Hat Closed ext_4.wav
             ■ Alesis D4fx                54   ♬ MaxV - XE8 Hat Closed int_5.wav
             ■ Alesis DM5                470   ♬ MaxV - XE8 Hat Medium int_6.wav
```

However, this method of browsing highlights two crucial attributes of an effective file browser: Firstly, The samples should be visually represented in order to be navigable. Secondly, files must be available for immediate playback.

A more natural representation of the sound files might be to visualize them as points on a 2d plane, where perceptually similar sounds are grouped together, while different sounds would be further away. The user could then play back the sounds by hovering the mouse over the points. This mode of navigating the collection would have many advantages. The user would be able to quickly find files similar to the sound he is imagining. Sounds would be represented by their inherent characteristics instead of relying on arbitrary labels.

In this research paper I will attempt to construct a method of finding an optimal representation of this 2d map.

Contents

1 Algorithm Overview

1.1 Assesing the quality of visualizations

As proposed in (Sarwate, 2017), quality of the visualization is assessed using 3 main metrics:

The silhoutte coefficient. A measure of quality for clusters based on the ground truth labels to assess how well similar sounds are grouped together. (*sklearn metrics silhouette score*, n.d.)

$$\frac{(b-a)}{max(a,b)} \tag{1}$$

Where a is mean intra-cluster distance and b is the mean nearest-cluster distance for each sample.

The Polsby-Popper test. A measure of compactness of the convex hull of the representation as presented by Polsby, Popper (Polsby & Popper., 2008).

$$\frac{4\pi A(D)}{P(D)^2} \tag{2}$$

Where *A(D)* is the area of the convex hull and *P(D)* is area of the smallest circle which can be drawn around the area.
This metric defines what shape the representation should take. I reasoned that a round representation would be easiest to navigate.

Overlap of convex hulls The overlap of clusters as calculated by the normalized overlap of each clusters convex hull.

Running Time Because the algorithm is expected to run on a local system, the total running time of the pipeline will also be taken into account.

1.2 Choosing the best algorithm

In order to arrive at a an effective algorithm for visualization I will follow a pipeline similar to the one presented in (Sarwate, 2017) A two step process is adopted. First, one of the feature extraction techniques is applied on the data set, followed by one or several dimensionality reduction steps.

The most effective algorithm is then deemed to be the pairing of feature extraction and dimensionality reduction algorithms which maximise the metrics given above. This can be represented as:

$$\max F(a), a \in ExD \tag{3}$$

Feature Extraction

Two main feature extraction methods will be used:

- **Short-Term Fourier Transform:** a sound file is divided into equal segments. The Fourier transform is computed for each of the segments. Because this method is focused on frequency information, the timbral and perceptual information is reliably represented.

- **Mel-Frequency Cepstral Coefficients:** An extension of STFT. This method is predominantly used in speech recognition. A short overview is as follows:

  1. The signal is divided into equal segments and the fourier transform is computed for each segment as in STFT.
  2. Run the signal through a mel-scale filterbank.
  3. Take a *log* of each window from the filterbank.
  4. Take a discrete cosine transform of each window.

Homogenizing sample length

To collapse features extracted from samples of different length into a uniform representation the following process will be applied as presented as presented in this blog post by Leon Fedden (Fedden, 2017)

- Mean for each feature dimension

- Standard deviation for each feature dimension

- Mean first order difference for subsequent features frames:

  $$\sum_{n=0}^{N} x_d[n] \tag{4}$$

  where:

  $$x_d[n] = \frac{x[n] - x[n-1]}{T} \tag{5}$$

  and T is the sampling period of the signal.

Dimensionality Reduction

The following dimensionality reduction techniques will be applied for each feature set:

- **Principal Component Analysis.** Features are converted into *Principal components* - linearly uncorrelated variables with maximized variance. In our case, two principal components will be extracted from the features and used to plot the point.
- **T-sne** Two probability distribution are constructed, one for the high-dimensional data and one for the lower-dimensional representation. These distributions maximise the probability of similar objects being picked. The divergence between these representations is then minimized.
- **UMAP** From the umap documentation: "The algorithm is founded on three assumptions about the data:
  - The data is uniformly distributed on a Riemannian manifold;
  - The Riemannian metric is locally constant (or can be approximated as such);
  - The manifold is locally connected.

  From these assumptions it is possible to model the manifold with a fuzzy topological structure. The embedding is found by searching for a low dimensional projection of the data that has the closest possible equivalent fuzzy topological structure."

1.3 Dataset

Quite a few labeled audio datasets exist. The most interesting and expansive of these is the google-backed audioSet. This dataset consists of youtube clips labeled in an expansive ontology. However, the dataset is focused more towards classification and is too expansive (over 2.1 mln samples!) for visualization

In the end I have settled on the Medley-solo dataset, (Lostanlen, Cella, Bittner, & Essid, 2019) over 21,000 samples of solo instrument performances divided into 8 classes. Each sample in this dataset has a fixed length. However, my aim is to test the model with samples of differing lengths. Because of this have chosen to supplement the training of the model with the IRMAS dataset (Bosch, Fuhrmann, & Herrera, 2014). It also contains samples of performances classified into 11 classes. However, it differs from Medley in two main aspects:

Firstly, the performances are not solo but in an ensemble. The most prominent instrument is labeled, along other instruments present in the recording. Second, The samples are of different length, which will enable the validation of the method for homogenizing samples of different length.

References

Bosch, J. J., Fuhrmann, F., & Herrera, P. (2014, September). *IRMAS: a dataset for instrument recognition in musical audio signals.* Zenodo. Retrieved from `https://doi.org/10.5281/zenodo.1290750` doi: 10.5281/zenodo.1290750

Fedden, L. (2017). *Comparative audio analysis with wavenet, mfccs, umap, t-sne and pca.* Retrieved from `https://medium.com/@LeonFedden/comparative-audio-analysis-with-wavenet-mfccs-umap-t-sne-and-pca-cb8237bfce2f`

Lostanlen, V., Cella, C.-E., Bittner, R., & Essid, S. (2019, September). *Medley-solos-DB: a cross-collection dataset for musical instrument recognition.* Zenodo. Retrieved from `https://doi.org/10.5281/zenodo.3464194` doi: 10.5281/zenodo.3464194

Polsby, D. D., & Popper., R. D. (2008). The third criterion: Compactness as a procedural safeguard against partisan gerrymandering. *Yale Law & policy Review 9.2 (1991): 301-353.*, *9*, 2579-2605.

Sarwate, L. H. H. . A. (2017). *Klustr: A tool for dimensionality reduction and visualization of large audio datasets.* Retrieved from `https://github.com/lamtharnhantrakul/klustr`

sklearn metrics silhouette score. (n.d.). Retrieved from `https://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette_score.html`