Machine Learning for Audio Signals

ECE 272/472 Audio Signal Processing

Bochen Li
University of Rochester
• Introduction
• Audio Feature Extraction
• Audio Alignment and Matching
• Classifiers
• Evaluation Measures
• Application 1: Sound Classification
• Application 2: Keyword Spotting
## Introduction

### Audio Signal Processing

- **Speech**
  - Speech Recognition
  - Talker Recognition
  - Emotion Detection
  - Speech Enhancement

- **Music**
  - Pitch/Chord Estimation
  - Genre Classification
  - Source Separation

### Machine Learning

- **Other**
  - Sound Event Detection
  - Auditory Scene Classification
Introduction

Applications

Voice Assistant

"Hey, Siri"

"Ok Google"
Introduction

Algorithmic Music Recommendation

Applications
Introduction

Applications

Music Tutor

Play the following notes

[Image of music notation and piano keyboard]
Introduction

Applications

Security surveillance
Outline

- Introduction
- **Audio Feature Extraction**
- Audio Alignment and Matching
- Classifiers
- Evaluation Measures
- Application 1: Sound Classification
- Application 2: Keyword Spotting
Audio Feature Extraction

- Energy
- Zero-Crossing Rate
- Pitch
- Chromagram
- Spectrogram
- Log-Mel Spectrogram
- Mel-Frequency Cepstrum Coefficient
Audio Feature Extraction

Pitch

Single Pitch Detection Methods

- Time domain:
  \[ F_0 = \frac{1}{\text{periods}} \]

- Frequency domain:
  \[ F_0 = \text{greatest common divisor} \]

- Cepstrum domain:
  \[ F_0 = \text{frequency gap} \]

- Signal is periodic
- Spectral peaks have harmonic relations
- Spectral peaks are equally spaced
Audio Feature Extraction

Pitch
Audio Feature Extraction

Spectrogram

Waveform

Spectrum

Spectrogram

Time

Frequency
Audio Feature Extraction

Spectrogram

Waveform

Spectrogram

Time

Frequency
Audio Feature Extraction

Log-Mel Spectrogram

Mel-Frequency Analysis

• Human auditory systems respond to frequencies in log-scale
  • Finer frequency resolution for low frequencies
  • Coarser frequency resolution for high frequencies
• Mel-frequency (mel-scale) analysis is inspired by human auditory systems
  • More filters in low frequencies
  • Less filters in high frequencies
• Human auditory systems respond to amplitudes in log-scale → Log-mel spectrogram
Audio Feature Extraction

Log-Mel Spectrogram

Mel-Frequency Analysis
Audio Feature Extraction

Log-Mel Spectrogram

Waveform

Mel-filter Response

Log-Mel Spectrogram
Audio Feature Extraction

Mel-Frequency Cepstral Coefficients (MFCC)

Steps

1. Audio frame $\rightarrow$ FFT $\rightarrow$ Spectrum
2. Spectrum $\rightarrow$ Mel-Filters $\rightarrow$ Log-Mel Spectrum
3. Now perform cepstral analysis
4. Take the cepstral coefficients as MFCC
Audio Feature Extraction

Mel-Frequency Cepstral Coefficients (MFCC)
Outline

• Introduction
• Audio Feature Extraction
• **Audio Alignment and Matching**
• Classifiers
• Evaluation Measures
• Application 1: Sound Classification
• Application 2: Keyword Spotting
Audio Alignment and Matching

Motivation

- Audio signals are time sequences
- How to measure the similarity?
Audio Alignment and Matching

Motivation

- Audio signals are time sequences
- How to measure the similarity?

Pair-wise matching

Warped matching
Audio Alignment and Matching

Dynamic Time Warping

Find the warping path

Sequence X

Sequence Y
Dynamic Time Warping

Step 1: Calculate the local distance matrix $C \in \mathbb{R}^{M \times N}$
Audio Alignment and Matching

Dynamic Time Warping

Step 2: Calculate the accumulated distance matrix $D \in \mathbb{R}^{M \times N}$

\[
D(n, 1) = \sum_{k=1}^{n} C(k, 1) \quad \text{for} \quad n \in [1 : N],
\]

\[
D(1, m) = \sum_{k=1}^{m} C(1, k) \quad \text{for} \quad m \in [1 : M],
\]

\[
D(n, m) = C(n, m) + \min \left\{ D(n - 1, m - 1), D(n - 1, m), D(n, m - 1) \right\}
\]
Audio Alignment and Matching

Dynamic Time Warping

Step 3: Backward Tracing the path \( P^* = (q_L, q_{L-1}, \ldots, q_1) \)

\[
q_1 = (N, M) \\
q_{\ell+1} = (1, m-1) \quad \text{if } n = 1, \\
q_{\ell+1} = (n-1, m) \quad \text{if } m = 1, \\
q_{\ell+1} = \text{argmin} \begin{cases} 
D(n-1, m-1), \\
D(n-1, m), \\
D(n, m-1)
\end{cases}
\]
Audio Alignment and Matching

Application: Keyword Matching

“Strawberry”

“Strawberry”

“banana”

“apple”
Audio Alignment and Matching

Application: Keyword Matching

Accumulated cost:

The sum of local distance matrix values through the warping path
Outline

• Introduction
• Audio Feature Extraction
• Audio Alignment and Matching
• Classifiers
• Evaluation Measures
• Application 1: Sound Classification
• Application 2: Keyword Spotting
Classifiers

• K-Nearest Neighbor Classification
• Gaussian Mixture Model
• Support Vector Machine
Classifiers

K-Nearest Neighbor Classification
Classifiers

Gaussian Mixture Model

From P. Smyth
ICML 2001
Classifiers

Support Vector Machine
Outline

• Introduction
• Audio Feature Extraction
• Audio Alignment and Matching
• Classifiers
• Evaluation Measures
• Application 1: Sound Classification
• Application 2: Keyword Spotting
Evaluation Measure

Binary Classification

Model Output $y$ V.S. Ground-truth Label $t$

Two class labels: 1 and -1

- True Positive (TP): Model predicts 1, ground-truth is 1
- False Positive (FP): Model predicts 1, ground-truth is -1
- True Negative (TN): Model predicts -1, ground-truth is -1
- False Negative (FN): Model predicts -1, ground-truth is 1
Evaluation Measure

Binary Classification

Model Output $y$ V.S. Ground-truth Label $t$

- **Accuracy**: \( \frac{TP + TN}{TP + FP + TN + FN} \)  
  \[= \frac{TP + TN}{P + N} \]

- **Precision**: \( \frac{TP}{TP + FP} \)

- **Recall**: \( \frac{TP}{TP + FN} \)
**Evaluation Measure**

**Multi-label Classification**

Model Output $\mathbf{y}$ V.S. Ground-truth Label $\mathbf{t}$

Multiple class labels: A, B, C, D, ...

- Confusion Matrix

<table>
<thead>
<tr>
<th>Predicted Label</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>%</td>
<td>%</td>
<td>%</td>
<td>%</td>
</tr>
<tr>
<td>B</td>
<td>%</td>
<td>%</td>
<td>%</td>
<td>%</td>
</tr>
<tr>
<td>C</td>
<td>%</td>
<td>%</td>
<td>%</td>
<td>%</td>
</tr>
<tr>
<td>D</td>
<td>%</td>
<td>%</td>
<td>%</td>
<td>%</td>
</tr>
</tbody>
</table>
Outline

• Introduction
• Audio Feature Extraction
• Audio Alignment and Matching
• Classifiers
• Evaluation Measures
• Application 1: Sound Classification
• Application 2: Keyword Spotting
Sound Classification

General process to train and test a classifier

1. Data preparation
   • Divide into training set and test set
   • Feature extraction
   • Annotate the labels

2. Train a classifier on the training set

3. Evaluate the classifier on the test set
Dataset:

- Animal sound
- 4 animal categories: cat, dog, sheep, duck
- Each has 15 1-sec recording samples
- 16K sample rate, mono channel
- First 12 samples for training, the other 3 for test
Sound Classification

Data Preparation

Feature Extraction

- MFCC Feature

![Cat](image1)
![Dog](image2)
![Sheep](image3)
![Duck](image4)
Sound Classification

Data Preparation

Concatenate all of the samples

Tip: Remove low-volume frames

Sample 1  Sample 2  ......
Sound Classification

Data Preparation

Add labels

Tip: Label each frame

Cat

Dog

Sheep

Duck

Label 1

Label 2

Label 3

Label 4
Sound Classification

Train the Classifier

- Feed the concatenated features and labels to the classifier
- Multi-class Support Vector Machine (SVM)
- MATLAB built-in function
- Save the model (classifier parameters)
Sound Classification

Evaluate the Classifier

- Repeat the same data preparation process on the test set
- Load the model
- Feed the concatenated features to the model
- Get the model output and compare with labels
- Evaluate the model using the **confusion matrix**
## Evaluate the Classifier

- **Confusion Matrix**

<table>
<thead>
<tr>
<th></th>
<th>Cat</th>
<th>Dog</th>
<th>Sheep</th>
<th>Duck</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cat</strong></td>
<td>95.71%</td>
<td>0.00%</td>
<td>4.29%</td>
<td>0.00%</td>
</tr>
<tr>
<td><strong>Dog</strong></td>
<td>0.00%</td>
<td>94.20%</td>
<td>0.00%</td>
<td>5.80%</td>
</tr>
<tr>
<td><strong>Sheep</strong></td>
<td>7.17%</td>
<td>0.00%</td>
<td>92.83%</td>
<td>0.00%</td>
</tr>
<tr>
<td><strong>Duck</strong></td>
<td>4.92%</td>
<td>5.74%</td>
<td>7.38%</td>
<td>81.97%</td>
</tr>
</tbody>
</table>
Outline

• Introduction
• Audio Feature Extraction
• Audio Alignment and Matching
• Classifiers
• Evaluation Measures
• Application 1: Sound Classification
• Application 2: Keyword Spotting
Keyword Spotting

Overview

Keyword selection
- Not a common word
- Easy to pronounce

Record samples
- Usually >200 talkers, 2000 samples

Data preparation
- Split into training/test set
- Label the utterance onset/offset

Model training
- GMM-HMM

Test & Parameter tuning
- Trade-off between detection accuracy and false alarm (precision and recall)

Release
- Embedded into device
Keyword Spotting

Data Preparation

1. Collect recording, 16K Hz, mono-channel

2. Label the utterance onset/offset

3. Split training/test set
   - Training 85%, test 15%
   - Same ratio of male/female, native/non-native talker
   - No talker overlaps in two sets

4. Prepare background data (continuous non-keyword speech)
Keyword Spotting

Data Preparation

Data Expansion

- Raw Recording
- Formant
- Pitch
- Tempo

- Babble noise
- Highway noise
- Kitchen noise
- Music noise
- Meeting Room
- Car
- Hall Room

Voice Transformation
Noise Addition
Reverberation

Thousands of samples → Millions of samples
KeywordSpotting

Model Training

Model

• Hidden Markov Model (HMM)
• Gaussian Mixture Model (GMM)

Feature

• MFCC Feature

Keyword Sample

“O” “K” “ei”

24 states

frames
Keyword Spotting

Model Training

Model
- Hidden Markov Model (HMM)
- Gaussian Mixture Model (GMM)

Feature
- MFCC Feature

Training Samples
1
2
3

... frames

24 states

Viterbi decoding
Assign label
Train GMM
Keyword Spotting

Live Model Test

- Each coming audio stream $\rightarrow$ MFCC $\rightarrow$ GMM $\rightarrow$ State classification
- State probability $\rightarrow$ Local distance matrix
- Calculate global distance matrix in real-time
- Run back-warding tracking in real-time
- Thresholding the accumulated cost

State Likelihood Matrix
Keyword Spotting

Live Model Test

- Each coming audio stream $\rightarrow$ MFCC $\rightarrow$ GMM $\rightarrow$ State classification
- State probability $\rightarrow$ Local distance matrix
- Calculate global distance matrix in real-time
- Run **back-warding tracking** in real-time
- **Thresholding** the accumulated cost
Keyword Spotting

Live Model Test

- Tune the parameter: **Threshold**
- Precision & Recall trade-off
  - Threshold ↙️, Detection Recall ↗️, False Alarm ↗️
  - Threshold ↗️, Detection Recall ↙️, False Alarm ↙️
- Big Regression Test:
  - Test on 72-hour background speech, keep false alarm within 10
  - Fix the threshold, and observe the detection recall on keyword samples
Keyword Spotting

Release the Product

“Alexa”

“OK Google”

“Hi Siri”

“Bixby”

“天猫精灵”

“小艾同学”

“小渡小渡”