On Feature Combination for Music Classification

Zhouyu Fu | Guojun Lu | Kai Ming Ting | Dengsheng Zhang

Gippsland School of Information Technology, Monash University, Australia

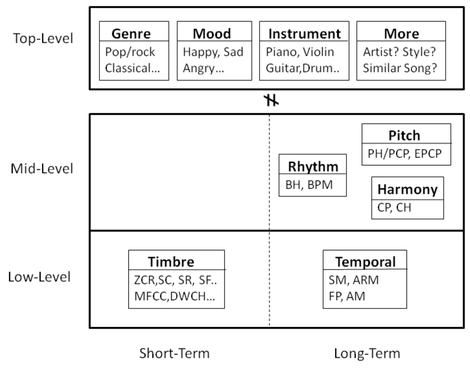
Problem

TASK: combining multiple types of features for music classification from raw audio signals

- Whether- we need multiple features?
- How- to combine different features?
- What- is the best feature and combination scheme?

Features

Taxonomy of audio features



- Timbre features capture the quality of the sound and has much to do with the instrumentation of the music.
- Temporal features capture the long-term variation of timbre and spectral features over time.
- Mid-level features are extracted on top of low-level features and more interpretable to human listeners.

Methods

Overview of Feature Combination

Problem Definition

- Input - data set $\{\left([\mathbf{x}_i^1, \ldots, \mathbf{x}_i^M], y_i\right)\}_{i=1,\ldots,N}$, with $\mathbf{x}_i^m \in \mathbb{R}^{d_m}$ and $y_i \in \{1, \ldots, K\}$
- Output - classification rule $f : \mathcal{X} \to \{1, \ldots, K\}$ with $\mathcal{X} \subset \mathbb{R}^{d_1} \times \cdot \times \mathbb{R}^{d_m}$

Overview of Methods

- Decision level - classifiers trained on individual features and fusion rules applied to the output of individual classifiers, e.g. majority voting, sum rule, stacked generalization, etc.
- Feature level - composite feature vector/similarity constructed from individual features, e.g. feature concatenation, multiple kernel learning, etc.

Decision Level Fusion

Principle

$$\{\mathbf{x}^1, \ldots, \mathbf{x}^M\} \overset{(1)}{\Longrightarrow} \{\mathbf{f}_1(\mathbf{x}^1), \ldots, \mathbf{f}_M(\mathbf{x}^M)\} \overset{(2)}{\Longrightarrow} f : \mathbf{f}_1 \times \cdot \times \mathbf{f}_M \to \{1, \ldots, K\}$$

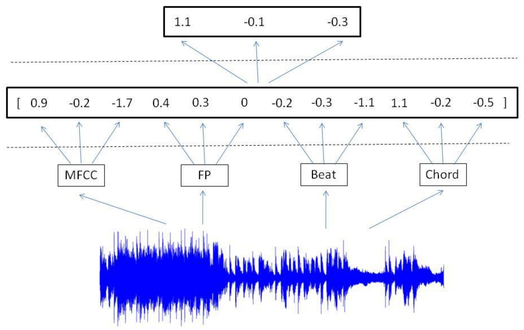
(1) Train a separate classifier $\mathbf{f}_m$ for each individual feature type $\mathbf{x}^m$

(2) Combine the decisions returned by individual classifiers

Assumption: decision scores are returned by each individual classifier $\mathbf{f}_m = [f_m^1, \ldots, f_m^C]$, where f_m^k is the score for kth class returned by the classifier trained on mth feature type

Methods

- Voting $\quad f : \arg\max_{k=1,\ldots,K} \sum_{m=1,\ldots,M} f_m^k$
  - Majority Voting : provides a winner-takes-all voting scheme
    $f_m^k = 1$ if mth classifier votes for class k and $f_m^k = 0$ otherwise
  - Sum Rule : provides a weighted voting scheme
    $f_m^k \in \mathbb{R}$ encodes the confidence value for/against class k returned by mth classifier
- Stacked Generalization
  - Stack the decision values returned by individual classifiers into a score vector
  - Train a classifier using the score vectors as new input features
  - Classifier on Classifier



Feature-Level Combination

Principle

$$\{\mathbf{x}^1, \ldots, \mathbf{x}^M\} \overset{(1)}{\Longrightarrow} \psi(\mathbf{x}^1, \ldots, \mathbf{x}^M) \overset{(2)}{\Longrightarrow} f(\psi)$$

(1) A composite feature (vector) is obtained by aggregating the individual features

(2) A single classifier is trained on the composite feature

Choice of feature-level combination methods is class specific, here we focus on SVM

Methods

- Feature Concatenation
  - Input Space $\quad \psi(\mathbf{x}) = [\mathbf{x}^{1T}, \ldots, \mathbf{x}^{MT}]^T$
  - Embedded Space (RKHS) $\quad \psi(\mathbf{x}) = \frac{1}{\sqrt{M}}[\phi(\mathbf{x}^1)^T, \ldots, \phi(\mathbf{x}^M)^T]^T$

    From kernel point of view, this is equivalent to averaging individual kernels

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle = \frac{1}{M} \sum_{m=1}^{M} \langle \phi(\mathbf{x}_i^m), \phi(\mathbf{x}_j^m) \rangle = \frac{1}{M} \sum_{m=1}^{M} K(\mathbf{x}_i^m, \mathbf{x}_j^m)$$

- Multiple Kernel Learning (MKL)
  - weighted feature concatenation in RKHS $\quad \psi(\mathbf{x}) = [\sqrt{\beta_1}\phi(\mathbf{x}^1)^T, \ldots, \sqrt{\beta_m}\phi(\mathbf{x}^M)^T]^T$
  - equivalent to weighted composite kernel $K_\beta(\mathbf{x}_i, \mathbf{x}_j) = \sum_m \beta_m K_m(\mathbf{x}_i^m, \mathbf{x}_j^m)$
  - kernel weights and SVM classifier weights are learned simultaneously

$$\underbrace{\min_{\beta \geq 0}}_{\text{Kernel}} \underbrace{\max_{0 \leq \alpha \leq \lambda}}_{\text{Classifier}} \sum_{i=1}^{N} \alpha_i - \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i y_i \alpha_j y_j K_\beta(\mathbf{x}_i, \mathbf{x}_j)$$

Results

Experimental Setup

- 1000 song clips equally distributed in 10 genres
- AU format with 22050Hz sampling rate, roughly 30 seconds in length for each clip
- Eightfeatures are extracted from each clip, including 3 timbre features (SMFCC, SASE, SOSC), 3 temporal features (TMFCC, TASE, TOSC) and 2 mid-level features of beat and chord.

Performance of Individual Features

	SMFCC	SASE	SOSC	TMFCC	TASE	TOSC	Beat	Chord
Blues	75.90	64.40	76.90	73.20	72.00	78.80	18.60	83.20
Classical	93.00	91.50	94.80	95.80	92.10	94.30	29.40	90.20
Disco	63.30	56.60	63.00	63.20	69.10	66.20	71.60	54.10
Hiphop	68.90	65.20	72.40	73.80	77.50	74.90	27.10	96.60
Metal	77.10	75.60	71.40	66.10	71.80	74.80	17.90	77.90
...								
10-class	73.55	68.00	73.10	73.81	73.20	75.00	24.66	78.92

- Chord is the best individual feature overall, and beat is the weakest
- No single type of feature performs consistently well for each individual class

Performance of Feature Combination

	Best	Vote	FC	AvgK	Sum	MKL	SG
Blues	83.20	86.30	89.60	94.20	91.70	93.70	95.70
Classical	95.80	96.80	97.00	97.20	96.60	97.50	97.00
Disco	71.60	77.60	83.00	83.70	86.10	86.30	86.60
Hiphop	96.60	85.90	86.60	91.90	93.40	93.00	93.30
Metal	77.90	78.60	80.60	88.00	87.80	89.70	87.80
...							
10-class	78.92	84.29	84.75	89.08	89.80	90.38	90.85

- Feature combination is effective in enhancing classification performance
- Classes for which all individual features perform weakly benefit more from feature combination
- Learning-based combination methods (SG, MKL) perform better than heuristics-based methods

Comparison with other methods

References	Accuracy	References	Accuracy
Feature Combination	$90.9 \pm 1.02\%$	T. Li *et. al.*	$78.5 \pm 4.07\%$
Bergstra *et. al.*	81%	I. Panagakis (2008) *et. al.*	$78.2 \pm 3.82\%$
Lee *et. al.*	$90.6 \pm 3.06\%$	T. Lidy *et. al.*	74.9%
Panagakis *et. al.*	$92.7 \pm 2\%$	G. Tzanetakis *et. al.*	61.0%