

Automatic Music Annotation

A Research Exam by

Douglas Turnbull

Department of Computer Science and Engineering

May 24, 2005

Research Exam Committee

Dean Tullsen

Charles Elkan

Serge Belogie

How do you search for music?

If you had heard this song on the radio and wanted to buy the album, how would you search for the song? You could search by

- **Name:** You might know the name of the **song** or the **artist**.
- **Genre:** You might try “Bebop,” “Latin Jazz,” or “Brazilian Jazz.”
- **Instrumentation:** The tenor sax, guitar, and double bass are all featured in the song.
- **Emotion:** The song has a “cool vibe” that is “sexy” and sounds like “elevator music.”

Musical Meta-Data

The song you have been listening to is “**Desafinado**” by performed by **Stan Getz** and **Charlie Byrd**.



AMG Allmusic has created a large database of **songs** and **musical meta-data**.

- The meta-data has been compiled by a large group of “experts.”
- A user can search the database by
 - Artist Stan Getz and Charlie Byrd
 - Song Desafinado
 - Instrument tenor sax, guitar, double bass
 - Moods “Summery,” “Wistful,” “Intimate”
 - Themes “At the beach,” “Relaxation,” “Summertime”
 - Time Period 1950’s or 1960’s
 - Country United States, Brazil, Latin America

MoodLogic uses a collaborative filtering approach to obtain meta-data for a large database of music files.

Musical Meta-Data

Musical meta-data can also be an entire song review.

Desafinado

by N. Mendonca and A.C. Jobim

Stan Getz (tenor sax), Charlie Byrd (guitar), Gene Byrd (rhythm guitar), Keter Betts (bass), Buddy Deppenschmidt (drums), Bill Reichenbach (percussion).

Recorded February 13, 1962

Featured on the album *Jazz Samba*

“The sensuous, airy saxophone of Stan Getz, Lester Young’s greatest disciple, proved the perfect voice to open the international door for the lifting, swinging Brazilian sambas of Antonio Jobim, Joao Gilberto, Luiz Bonfa and others. The album *Jazz Samba* was on the pop charts for seventy weeks, peaking at no. 1.”

- liner notes from Ken Burns Jazz collection.

Automatic Music Annotation

Most commercial systems (Apple Itunes, Amazon, AMG Allmusic, Moodlogic) use human experts or collaborative filtering to annotate music.

A number of research systems have been developed that **automatically annotate music**.

Automatic annotation uses low-level audio content to describe high-level musical concepts.

- **Low-level Content:** bitstream of audio samples
- **High-level Concepts:** genre, emotion, instrumentation, rhythm

Automatic Music Annotation

Annotation systems have two components

1. Feature Extraction
2. Learning/Modeling

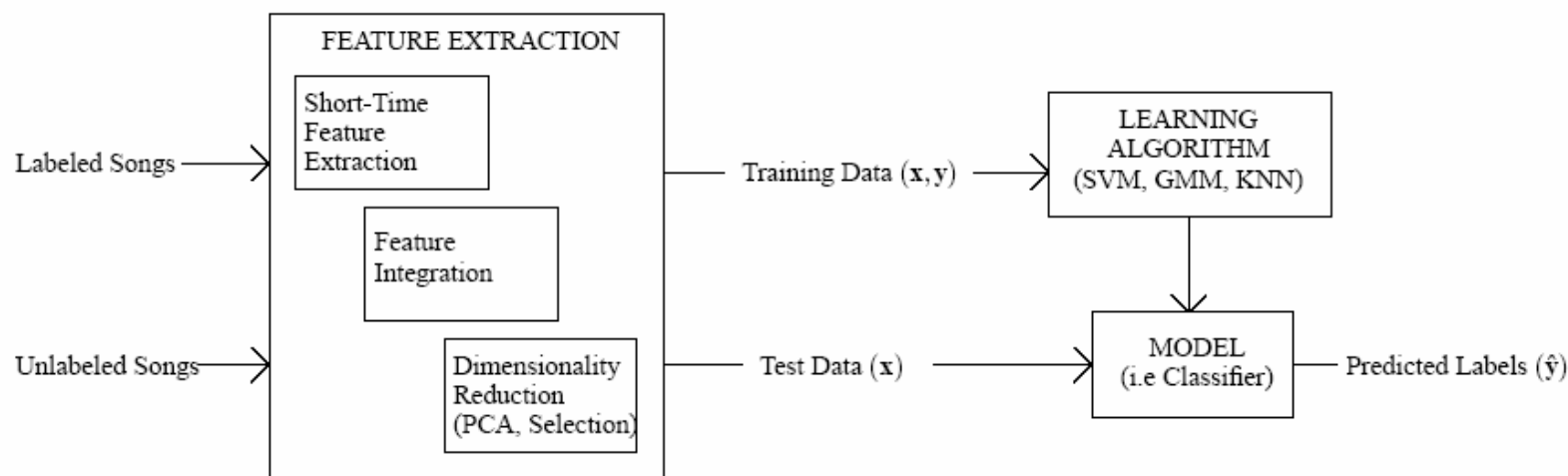
1. Feature Extraction: find a low dimensional representation of a song

- **Short-Time Feature Extraction:** extract a vector of features (\mathbf{x}_i) for each short-time ($\sim 25\text{ms}$) segment of the song using digital signal processing
- **Feature Integration:** combine a series of short-time feature vectors $\{\mathbf{x}_1, \dots, \mathbf{x}_t\}$ into one feature vector (\mathbf{x}) that represents the song
- **Dimensionality Reduction:** reduce the dimension of \mathbf{x} using feature subset selection or Principal Component Analysis (PCA)

Automatic Music Annotation

2. Learning/Modeling: use labeled feature vectors (\mathbf{x}, \mathbf{y}) to train a model. The model can then be used to predict labels $(\hat{\mathbf{y}})$ for an unlabeled song (\mathbf{x}) .

- Existing systems use a **supervised learning framework**.
- **Latent variables models** offer an alternative to supervised learning.



Outline

- 1. Digital signal processing background**
- 2. Three genre classification systems**
- 3. Latent variable models**
- 4. Final comments**

Outline

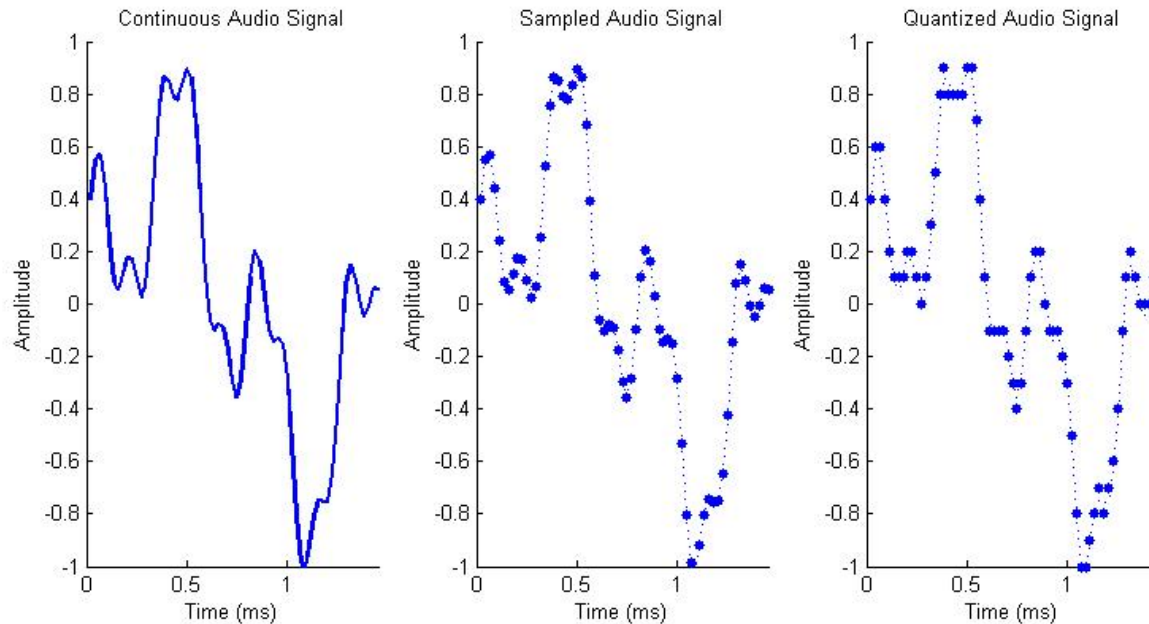
- 1. Digital signal processing background**
 - 1. Digital audio signals**
 - 2. Time and frequency domains**
 - 3. Transforms**
 - 4. Mel-frequency cepstral coefficients (MFCC)**
- 2. Three genre classification systems**
- 3. Latent variable models**
- 4. Final comments**

Digital Audio Signal

When music is recorded, the pressure from the acoustic wave is measured using a microphone.

Sampling: measurements are taken at a regular time interval

Quantization: each sample is rounded to the closest quantum



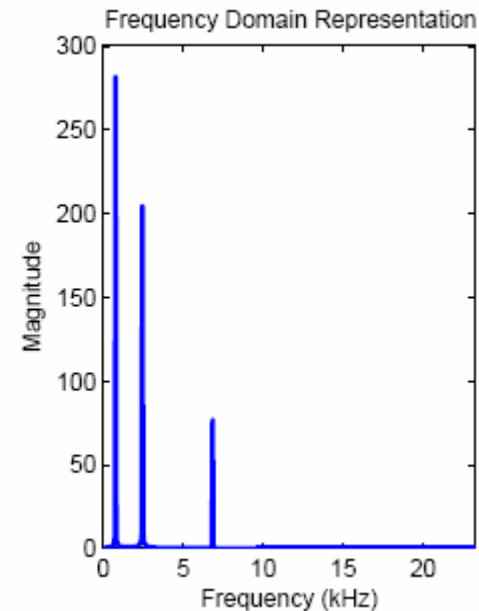
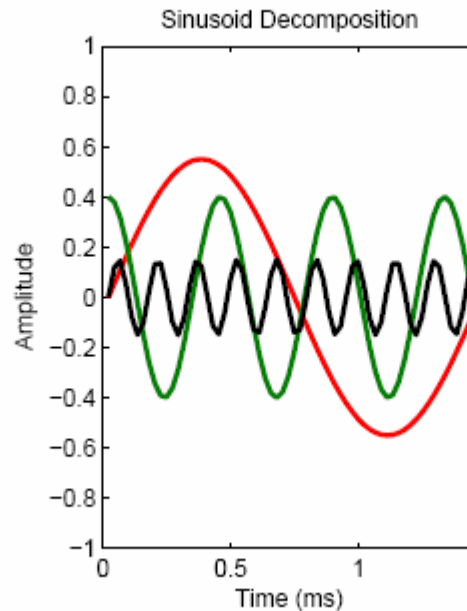
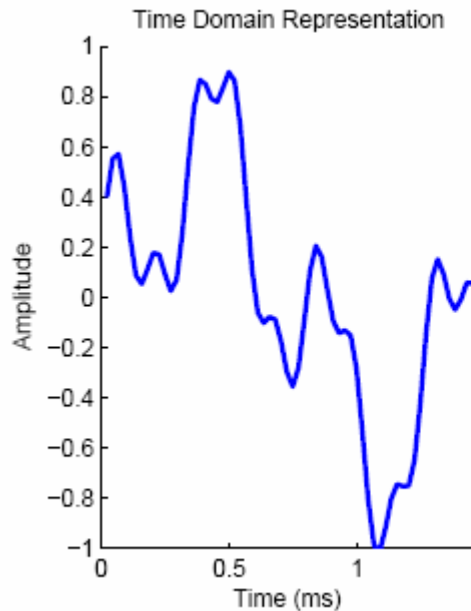
CD audio is sampled 44,100 times per second and stored as a time series of 16-bit values.

Time and Frequency Domains

Time Domain Representation: time series of pressure measurements

Frequency Domain Representation: sum of sine and cosine waves

$$x = \sum_{k=0}^{N/2} a_k^{(r)} \cos(2\pi(\frac{k}{N})) + a_k^{(i)} \sin(2\pi(\frac{k}{N})). \quad (1)$$



Real Discrete Fourier Transform (DFT)

The DFT is used to transform a time series into the frequency domain.

$$a_k^{(r)} = \frac{2}{N} \sum_{i=0}^{N-1} x[i] \cos(2\pi \frac{k}{N} i) \quad (1)$$

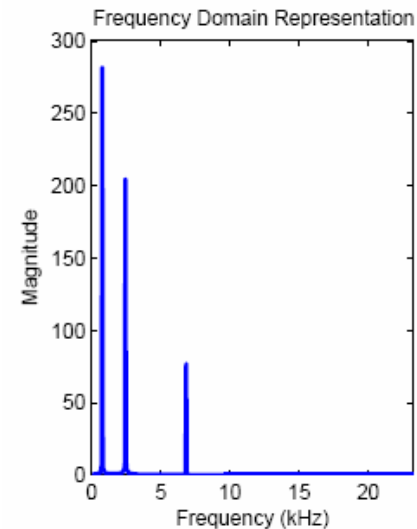
$$a_k^{(i)} = -\frac{2}{N} \sum_{i=0}^{N-1} x[i] \sin(2\pi \frac{k}{N} i) \quad (2)$$

The **magnitude** is calculated according to $X_M[k] = \sqrt{(a_k^{(r)})^2 + (a_k^{(i)})^2}$

Magnitude spectrum: a plot of the magnitudes for each frequency component

Power spectrum: a plot of magnitude² for each frequency component

Energy: the integral of the power spectrum



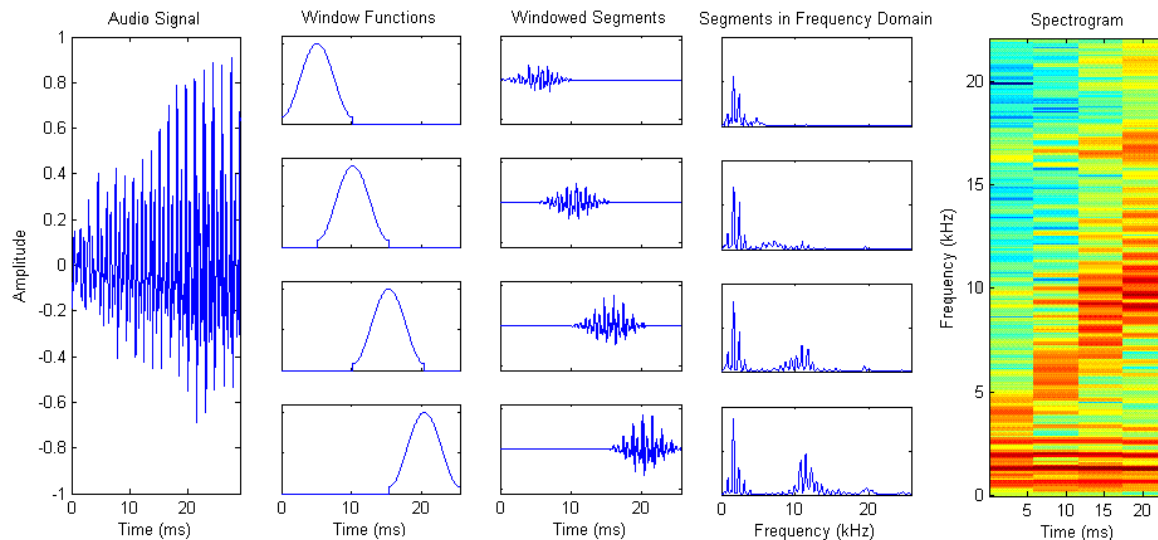
Short-Time Fourier Transform (STFT)

The STFT produces a time-frequency representation of an audio signal

- The signal is broken up into short segments using a window function
- The DFT of each segment is computed
- A **spectrogram** is the resulting time-frequency representation

$$a_{n,k}^{(r)} = \frac{2}{N} \sum_{i=0}^{N-1} x[n+i]w[i] \cos(2\pi \frac{k}{N}i) \quad (1)$$

$$a_{n,k}^{(i)} = -\frac{2}{N} \sum_{i=0}^{N-1} x[n+i]w[i] \sin(2\pi \frac{k}{N}i) \quad (2)$$



Short-Time Fourier Transform (STFT)

The STFT produces time-frequency representation of an audio signal

- The shorter the window length, the better the time resolution.
- The more samples in a windowed segment, the better the frequency resolution.
- There is a tradeoff between time and frequency resolution.

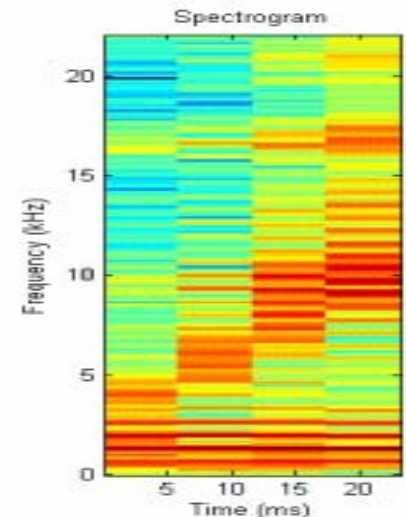
The time-frequency representation has a **fixed resolution** for all frequency bands.

Human hearing is approximately logarithmic in frequency.

- The perceptual difference between sounds oscillating at 50Hz and 60Hz is greater than the difference between sounds oscillating at 500Hz and 510Hz.

Given this perceptual model, the STFT produces

- High frequency bands with bad time resolution
- Low frequency bands with bad frequency resolution

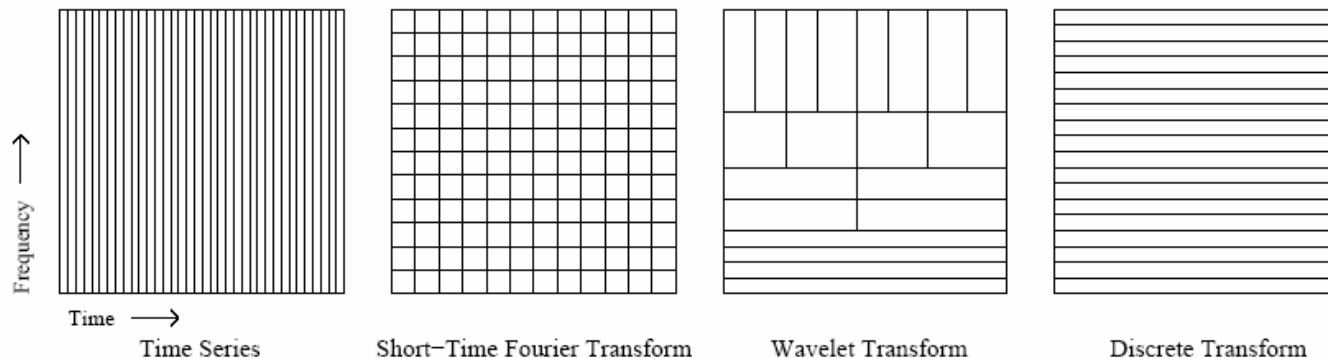


Wavelet Transform (WT)

Wavelet transforms produce representations that have **variable resolution**.

- Each frequency band is analyzed with wavelet (small oscillating waveforms) that are scaled to match the frequencies in the band.
- High frequency bands use short wavelets.
- Low frequency bands use long wavelets.

STFTs and WTs are different ways to break up the **time-frequency plane**.



Outline

1. **Digital signal processing background**
2. Three genre classification systems
3. Latent variable models
4. Final Comments

Outline

1. Digital signal processing background
2. **Three genre classification systems**
 1. **TC02: Tzanetakis and Cook (2002)**
 2. **LOL03: Li, Ogiwara, and Li (2003)**
 3. **MB03: McKinney and Breebaart (2003)**
3. Latent variable models
4. Final comments

‘Genre’

Musical genre is perhaps the most common concept used to classify music.

Genre is ill-defined due the subjective nature of music

- Flat vs. hierarchical taxonomy
- Number of genres
- Names of the genres
- The criteria for placing a song into a genre

In each of the following works, the authors bracket this problem in order to make progress with their research.

‘Genre’

The assumptions about genre (data sets, taxonomy) make it hard to compare classification performance.

The four works have been chosen based on their contribution to musical feature design.

Although the features have been designed for classification by genre, they are useful for other annotation tasks (emotion, instrumentation, rhythmic structure).

TC02 –Tzanetakis and Cook

Pachet and Cazaly (2003) review six genre classification systems that have been developed before 2002.

Tzanetakis and Cook implement a superset of features from these six systems as well as develop new features for modeling music.

The final feature vector is composed of 30 features

- Timbral texture features (19)
- Rhythm content features (6)
- Pitch content features (5)

TC02 –Tzanetakis and Cook

Five spectral features are calculated for each short-time window

Feature	Meaning	Formula
Centroid	brightness	$C = \frac{\sum_{k=1}^{N/2} X_M[k] * k}{\sum_{k=1}^{N/2} X_M[k]}$
Rolloff	low frequencies energy	$\sum_{k=1}^R X_M[k] = 0.85 * \sum_{k=1}^{N/2} X_M[k]$
Flux*	local spectral change	$F_t = \sum_{k=1}^{N/2} (X_M^{(t)}[k] - X_M^{(t-1)}[k])^2$
MS Energy	loudness	$MS = \frac{1}{(N/2)} \sum_{k=1}^{N/2} X_M[k]^2$
Zero Crossing Rate	noisiness	Rate that time series crosses zero amplitude

* Note that normalized frequency bins should be used for spectral flux.

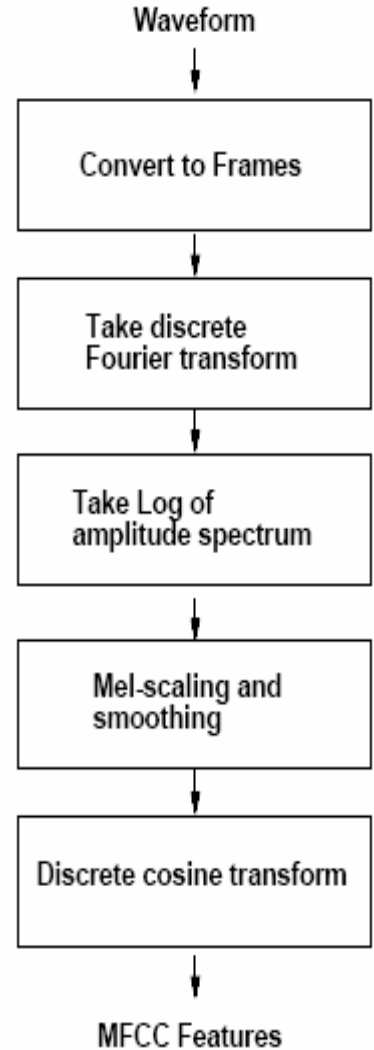
TC02 –Tzanetakis and Cook

Mel-frequency cepstral coefficients (MFCC) is a perceptually motivated features set that describes the shape of the spectrum for a short-time audio segment.

For each segment

1. Find the spectrum using the DFT
2. Calculate the log spectrum
3. Apply Mel-Scaling
 - Mapping between true frequency and perceived frequency
4. Separate frequency components into 40 bins
5. Apply discrete cosine transform (DCT)
 - Reduces dimensionality
 - Similar to PCA but does not require training

The result is 13 coefficients that characterize spectral shape.



Plot taken from [Log00].

TC02 –Tzanetakis and Cook

Feature Integration: the final feature vector is comprised of the **mean** and **variance** of each short-time timbral texture feature.

Timbral Texture features

- [1-8] Mean and variance of centroid, rolloff, flux, and zero-crossing rate
- [9] Low-Energy: percentage of windows with less RMS energy than average
- [10-19] Mean and variance of the second through sixth MFCCs.

The authors could use additional moments (i.e. skewness and kurtois) to summarize the the empirical distributions of these features.

TC02 –Tzanetakis and Cook

Rhythm content features are based on a beat histogram

- Histogram of beat strength versus beats-per-minute
- Analysis is done on a series of long audio segments (3 seconds)

Rhythm Content features

- [20] Total Beat Strength: sum of all beat strengths
- [21-22] Relative strengths of two highest peaks:
peak strength divided by total beat strength
- [23-24] Period of two highest peaks in BPM
- [25] Ratio of two peaks strength:
strength of second peak divided by strength of first peak

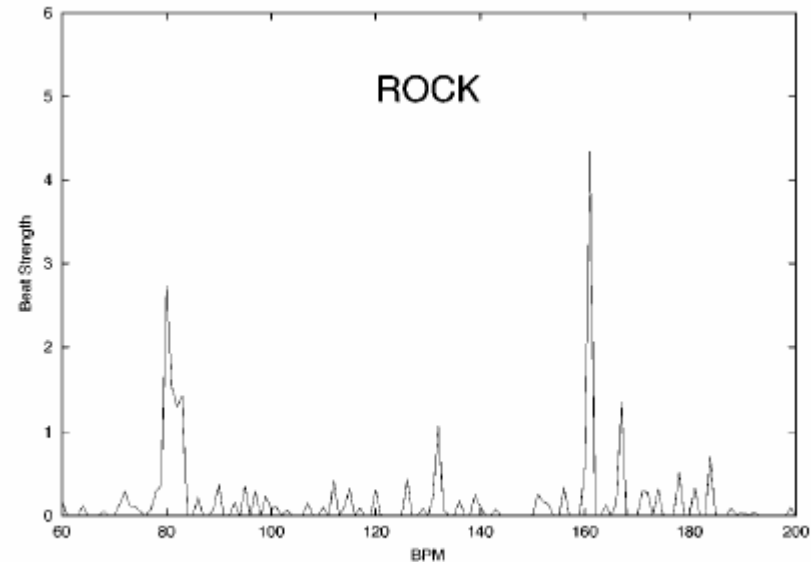


Fig. 2. Beat histogram example.

Plot taken from [TC02].

TC02 –Tzanetakis and Cook

Pitch content features are based on a pitch histogram

Histograms calculated using a multiple pitch detection algorithm

- Uses **autocorrelation** to determine pitch periods
- Reduces the effect of integer multiples of the pitch frequencies

Unfolded Pitch Histogram

- Pitch strength vs. frequency
- Frequencies are binned to musical notes
 - e.g. Middle A at 440Hz = [428 Hz, 453 Hz]

TC02 –Tzanetakis and Cook

Folded Histogram

- Musical note classes are merged.
 - e.g. A at 110 Hz, 220 Hz, 440Hz, 880 Hz, etc
- Notes are mapped according to circle of fifths.
 - e.g. A is adjacent to D and E

Pitch Content features

- [26] Amplitude of highest peak in the Folded Pitch Histogram
- [27] Period of maximum peak in the Unfolded Pitch Histogram
- [28] Period of maximum peak in the Folded Pitch Histogram
- [29] Interval of two highest peaks in the Folded Pitch Histogram
- [30] Total Pitch Strength: sum of pitch strength over all frequencies

TC02 –Tzanetakis and Cook

The authors collect a data set of 1000 songs.

- Each song is assigned one genre label for a set of ten genres.
- Genres: Classical, Country, Disco, Hip Hop, Jazz, Rock, Blues, Reggae, Pop, Metal

Using Gaussian mixture models (GMM) and K-nearest neighbor (KNN) classifiers, the authors achieve 61% classification accuracy.

Humans achieve 70% accuracy on an equivalent test.

On the same data set and using the same features

- Tzanetakis and Li (2003) achieve 72% accuracy using support vector machines (SVM)
- Turnbull and Elkan (2005) achieve 71% accuracy using radial basis function networks (RBF networks)

LOL03 – Li, Ogiwara, and Li

In LOL03, feature extract is performed using a **wavelet transform**.

- Break musical signal up into short-time segments
- For each segment, compute Daubechies wavelet transform
- Divide audio segments into seven frequency bands corresponding to seven musical octaves
- Create a histogram of wavelet coefficients for each band
- Calculate first 3 moments of each histogram (mean, variance, skewness)
- Calculate sub-band energy

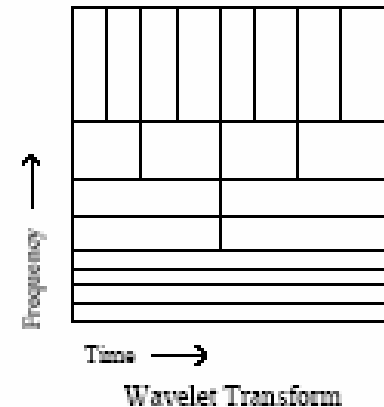
$$SBE = \sum_{k=B_i[1]}^{B_i[N]} |X_M[k]|$$

Daubechies Wavelet Coefficient Histogram (DWCH) features

[1-17] Daubechies Wavelet Coefficient Histogram

4 sub-bands * (mean, variance, skewness, energy)

[17-35] Timbral Texture Features from TC02



LOL03 – Li, Ogihara, and Li

Using the same data set as TC02, LOL03 achieve 78% classification accuracy using SVM classifiers.

There was a 6% improvement over the best results from previous works and an 8% improvement of human accuracy.

This suggests that the model might be fitting the nature of the individual who labeled the data set rather than some general notion of genre.

MB03 – McKinney and Breebaart

McKinney and Breebaart compare four feature sets

1. Standard low-level (SLL) features –
 - similar to TC02 timbral texture and pitch content features
2. Mel-frequency cepstral coefficients (MFCC)
3. **Psychoacoustic** (PA) features – approximations to perceptual models
 - Roughness: energy between 20-150Hz
 - Loudness: energy of spectrum
 - Sharpness: high frequency energy
4. **Auditory filterbank temporal envelop** (AFTE) features
 - Gammatone transform: variable resolution transform which models the human auditory system
 - 18 bandpass filters that are logarithmically spaced

MB03 – McKinney and Breebaart

McKinney and Breebaart compare two feature integration methods

1. **Static Features:** mean of a time series of feature vectors
2. **Dynamic Features:** filterbank transform of a time series of feature vectors
 1. Compute the power spectrum for each feature using DFT
 2. Calculate energy for four bands (0 Hz, 1-2 Hz, 3-15Hz, 20-43 Hz)

MB03 – McKinney and Breebaart

The authors collect a (small) data set of 188 songs

- Each song is given a label from {Jazz, Folk, Electronica, R&B, Rock, Reggae, Vocal}
- Some genres are poorly represented: Reggae (11), Vocal(9)

Feature subset selection (FSS) is used to reduce the feature set to 9 features.

- FSS is done using forward stepwise selection
- FSS is used as a control for dimensionality

Quadratic discriminate analysis (QDA) is the supervised learning algorithm

- Find quadratic decision boundary by minimizing the ratio of within-class scatter to between class scatter.
- Only one classifier is used to compare future sets.

MB03 – McKinney and Breebaart

The best results are found using the dynamic AFTE features: 74% accuracy

Observations

1. **Dynamic features** are always better than static features.
2. **AFTE features** are better than SLL, MFCC, and PA features.

Comments

1. Not limiting the final feature set to 9 features might produce interesting results.
2. Combining features from different feature sets might yield better performance.
3. Using other classifiers, such as SVMs, might effect results.

Genre Classification Summary

It is hard to directly compare classification results based on the

- **Assumptions** about genre
- **Size** and **contents** of the data sets

System	Task (# classes)	Database Size	Best Results
TC02	Genre(10)*	1000	61%
	Classical(4)	400	88%
	Jazz(6)	600	68%
LT03	Genre(10)*	1000	72%
TE05	Genre(10)*	1000	71%
LOL03	Genre(10)*	1000	79%
	Genre(5)	756	99%
MB03	Audio(5)	310	93%
	Genre(7)	188	74%
MAL05	Genre(5)	100	96%
	Genre(6)	354	69%

Outline

1. Introduction
2. Digital signal processing background
3. Three genre classification systems
- 4. Latent variable models**
5. Final Comments

Latent Variable Models

Existing music annotation systems are based on **supervised learning**

- Each song is assigned one label from a pre-established set of labels
- It is difficult to establish set of labels
- It is costly to hand label songs

Carneiro and Vasconcelos (2005) observe that early image annotation systems were also based on **supervised learning**.

- Holistic image concepts: ‘indoors’, ‘outdoors’, ‘landscape’, ‘cityscape’
- Objects: ‘building’, ‘horse’, ‘face’

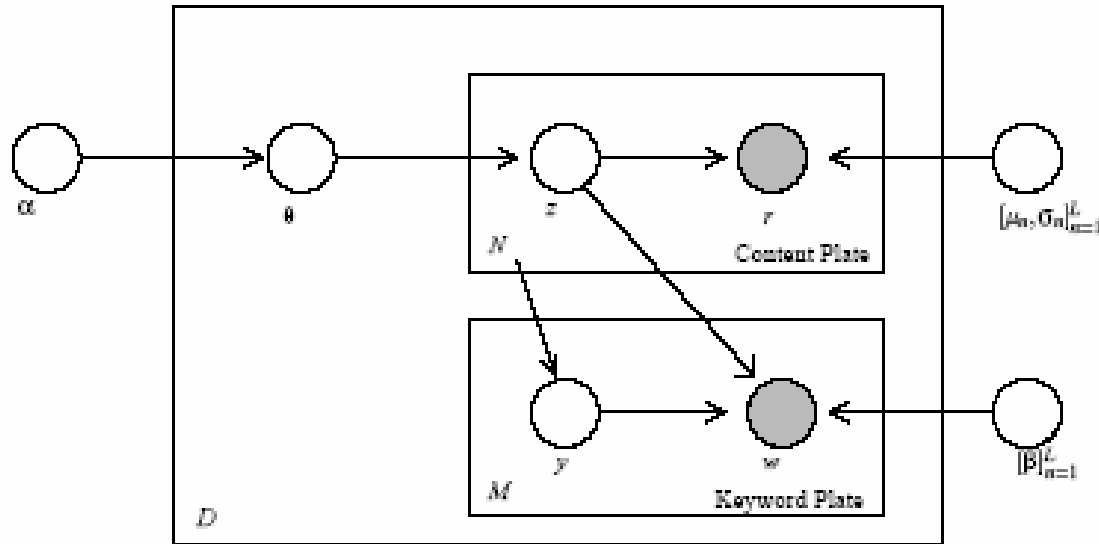
They also observe that more recent systems have been based on **latent variable models**.

Latent Variable Models

- A set of latent (i.e **hidden**) variables is introduced that encode hidden states.
- Each state (i.e **topic**) represents a joint distribution of keywords and feature vectors.
- A training set of keywords and feature vectors are presented to an unsupervised learning algorithm, such as Expectation Maximization (EM), to estimate the parameters of the model.
- During annotation, the keywords that maximize the joint probability of keywords and feature vectors are the predicted labels for the unlabeled song
 - The latent variables are marginalized during annotation

Correspondence Latent Dirichlet Allocation

Corr-LDA is a popular latent model developed for image annotation (Blei and Jordan)



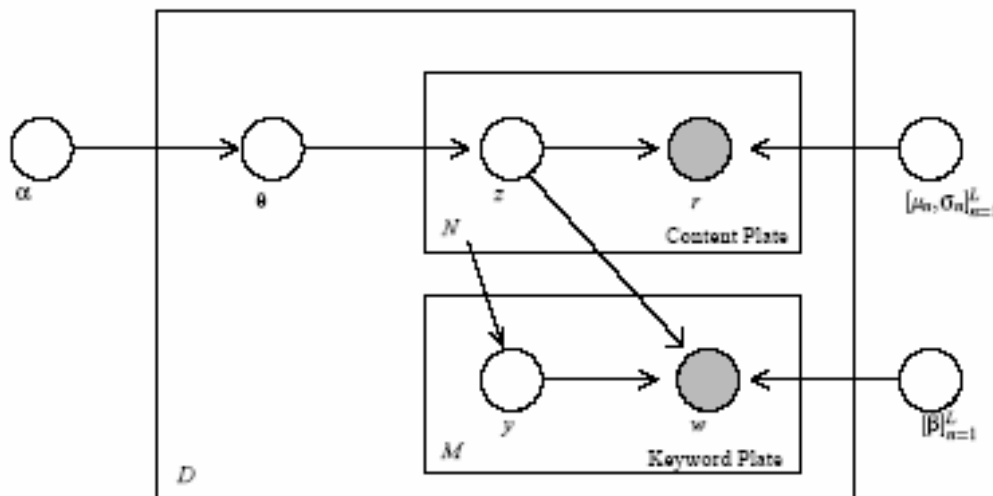
Each image is an (r, w) pair

- \mathbf{r} is a vector of N image region feature vectors
- \mathbf{w} is a vector of M keywords from an image caption vocabulary of W keywords

Correspondence Latent Dirichlet Allocation

The **generative process** for an image and image annotation according to the Corr-LDA model is

1. Draw θ from a Dirichlet distribution with parameter α
 - θ is a multinomial random variable and can be thought of as a “distribution over topics.”
2. For each of the N image regions r_n , draw a topic z_n . Draw an image region from the Gaussian distribution associated with topic z_n .
3. For each of the M keywords w_m , pick one of the topics that was chosen in step 2. Draw a keyword from the multinomial distribution associated with this topic.



Correspondence Latent Dirichlet Allocation for Image Annotation

Parameter estimation is done using variational EM (See Blei's Thesis – Chapter 3).

Annotation is performed by finding the M words that individually maximize

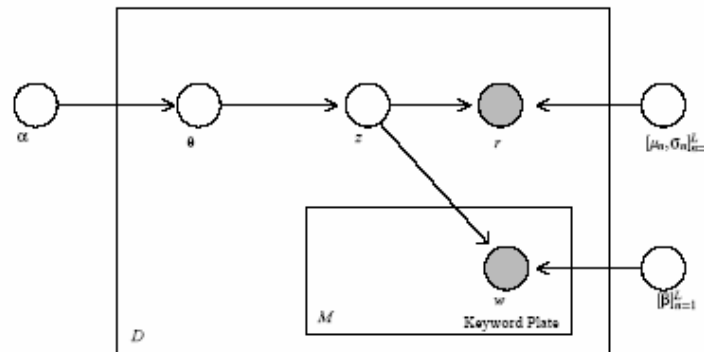
$$\operatorname{argmax}_{\mathbf{w} \in W} \prod_{i=0}^L \prod_{n=1}^N p(\mathbf{r}_n | \mu_i, \sigma_i) p(w | \beta_i)$$

where L is the number of hidden states.

Correspondence Latent Dirichlet Allocation for Music Annotation

If we do not segment the songs, a simplified version of Corr-LDA might be used for music annotation.

- Note that only one topic is used to generate an entire song.



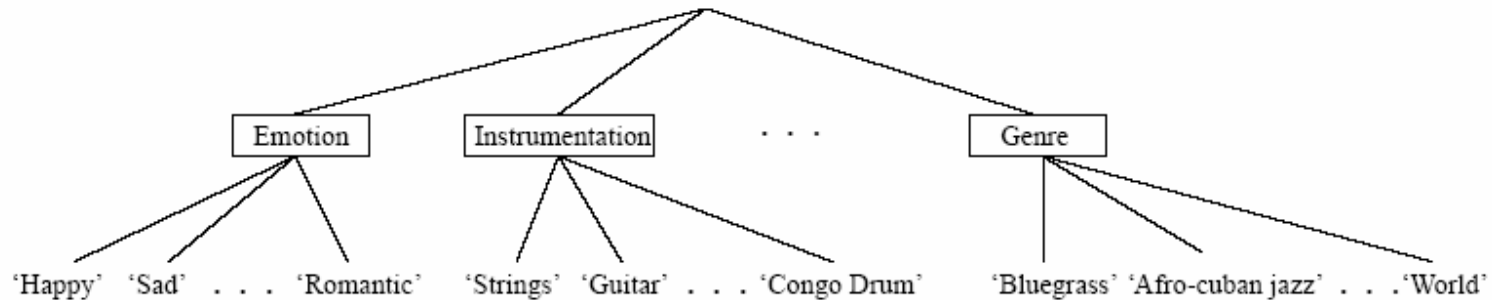
Segmentation would allow us to use the original the Corr-LDA model.

1. Fixed length segments
2. **Automatic Segmentation** techniques based on
 - Self-similarity (Perry and Essa)
 - Changes in short-time features

Correspondence Latent Dirichlet Allocation for Music Annotation

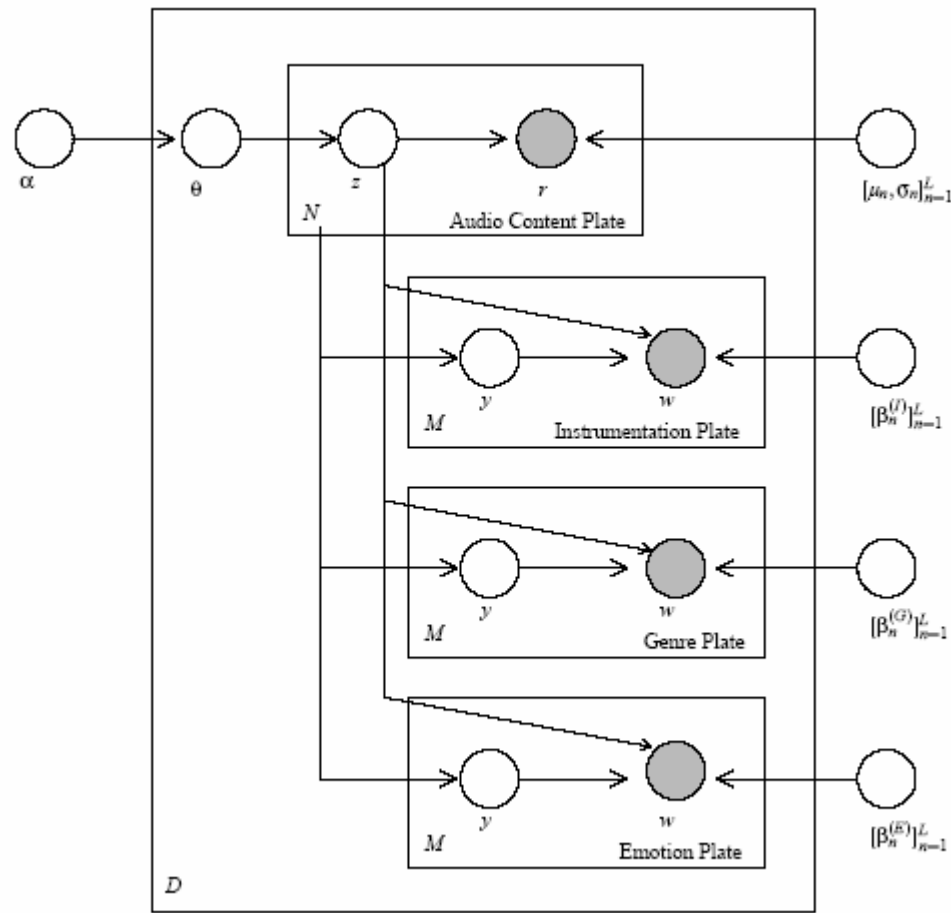
One draw back of the Corr-LDA model is that it assumes the vocabulary of image caption words is ‘flat’.

For image annotation, we might want to uses a ‘**hierarchical vocabulary**’.



Correspondence Latent Dirichlet Allocation for Music Annotation

If we adapt the Corr-LDA model so that it can make use of a hierarchical vocabulary, a new latent model is given by



Outline

1. Digital signal processing background
2. Three genre classification systems
3. Latent variable models
4. **Final Comments**

Summary

1. Existing Automatic Music Annotation

- Feature Extraction and Supervised Learning

2. Musical Feature Extraction for Genre Classification

- Short-time Extraction
- Feature Integration
- Dimensionality Reduction

3. Latent Variable Models for Music Annotation

Final Comments

1. Creating a **common data set**
 - Due to copyright laws, transmission (bandwidth) issues, and labeling assumptions, it is hard to create a common test data set.
2. The use of **latent variable models** are an alternative to supervised learning models
 - Latent variable models have been successful for image annotation
3. The use of **automatic segmentation** might aid music annotation in the future.

Thank You

Thank you to Dean Tullsen, Charles Elkan, and Serge Belongie for their time and energy while serving on my research exam committee.

Thank you to Charles Elkan, Dave Kauchak, and Shlomo Dubnov for their help during the research and writing of this research exam.

Other Transforms

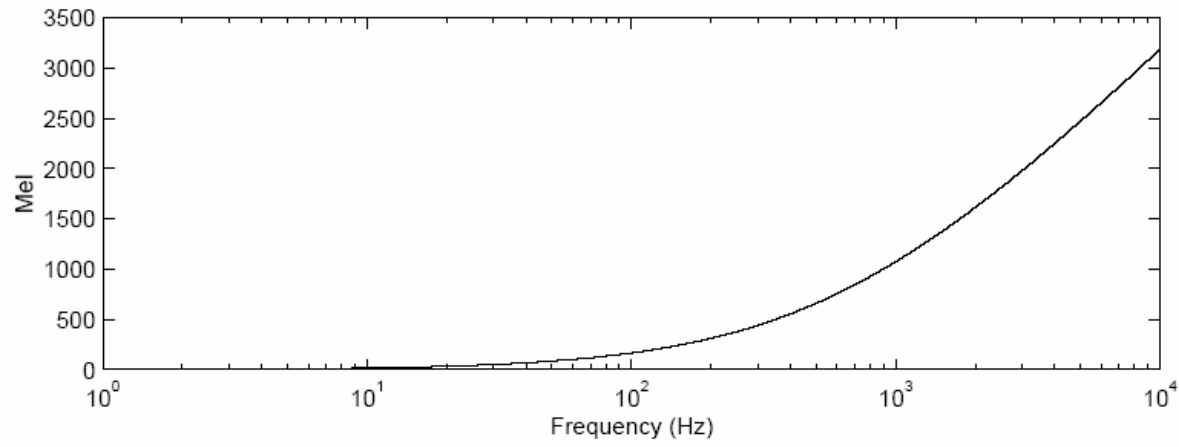
Discrete cosine transform (DCT)

- Uses a basis of N cosine waves that differ in frequency by π .
 - By comparison, the DFT uses a basis of $N/2$ cosine waves and $N/2$ sine wave, where each cosine wave (and sine waves) differ in frequency by 2π
- There are eight variants of the DCT.
- Assumes signal is periodic and has even symmetry
 - The DFT only assumes that the signal is periodic
 - Thus a N -point DCT is identical to the $(2N-2)$ point DFT of the symmetrically extended

Gammatone transform

- Variable Resolution like wavelets.
- Basis functions (filterbank) is designed to model human

The Mel Scale



MAL05 – Meng, Ahrendt, and Larsen (2005)

Meng, Ahrendt and Larsen compare feature integration techniques

- Early Fusion: create longer time features from shorter time features
- Late Fusion: combine outputs from series of classifiers to make a final decision

The authors consider short, medium and long time scale features

Timescale	Frame-size	Perceptual meaning
Short time	30ms	instant frequency (harmonics, pitch)
Medium time	740ms	timbre, modulation (instrumentation)
Long time	9.62s	beat, mood, vocal

Short-time features are the first 6 MFCC features

MAL05 – Meng, Ahrendt, and Larsen (2005)

Early Fusion Techniques

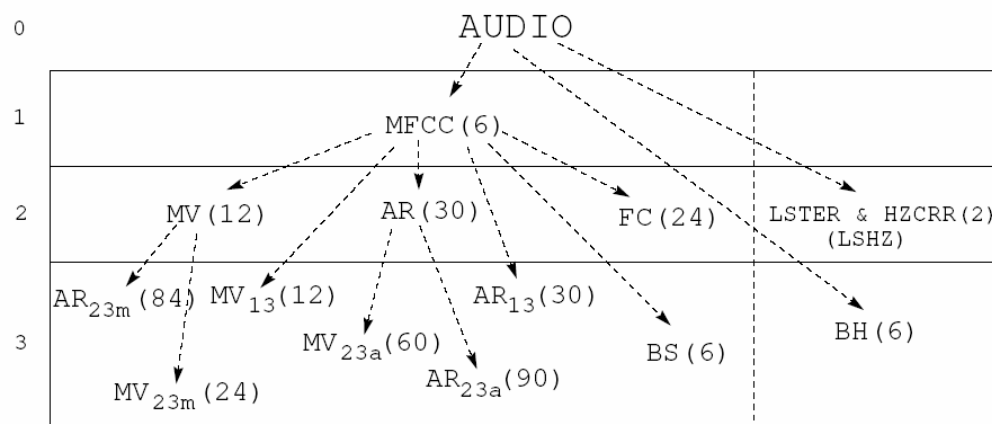
- Mean and variance (MV) – TC02, LOL03
- Filterbank transform coefficients (FC) – MB03
- Autoregressive model (AR) – MAL05
 - Based on Linear Predictive Coding (LPC)
 - Find a_i 's using Levinson-Durbin Algorithm
 - a_i 's characterize coloring filter for a random time series

$$x[n] = \left(\sum_{i=1}^P a_i x[n-i] \right) + \epsilon[n]$$

Early fusion techniques are combined to create medium and long-time feature vectors

PCA is used to reduce the dimension of high-dimensional feature vectors

The LPC model order P is found using cross-validation



MAL05 – Meng, Ahrendt, and Larsen (2005)

Late Fusion Techniques

The output from a series of shorter time (short, medium) classifiers are combined to make longer time decisions (medium, long).

The output from a classifier may be a score (binary label or real value) or a posterior probability.

Scores can be mapped to a probability distribution function using calibration (e.g. Isotonic regression).

Late Fusion Decision Rules

Majority Vote: genre with the most votes wins

Sum Rule: genre with the largest sum of posterior probabilities wins

Median Rule: genre with the largest median posterior probability wins

MAL05 – Meng, Ahrendt, and Larsen (2005) - Results

Data Set: 100 songs from 5 ‘perceptually separable’ genres

Classifiers: neural networks, Gaussian classifier

The best results are achieved using a three-step algorithm

- 1) Calculate short-time MFCC features
- 2) Use AR to obtain medium-time features
- 3) Use late fusion according to the sum rule to make long-time decision

Comments

1. Other classifiers might yield different performance
2. There are a large number of parameters to tune
 1. Number of MFCCs, LPC model order, PCA Eigenvectors, decision rule, etc
3. Small data set