JOHNS HOPKINS
WHITING SCHOOL of ENGINEERING

Music Emotion Predictions Using Back Propagation Neural Network and ABC Algorithm

PRESENTATION BY:
XINYU CHANG, XIANGYU ZHANG, YULU RAN, HAORUO ZHANG

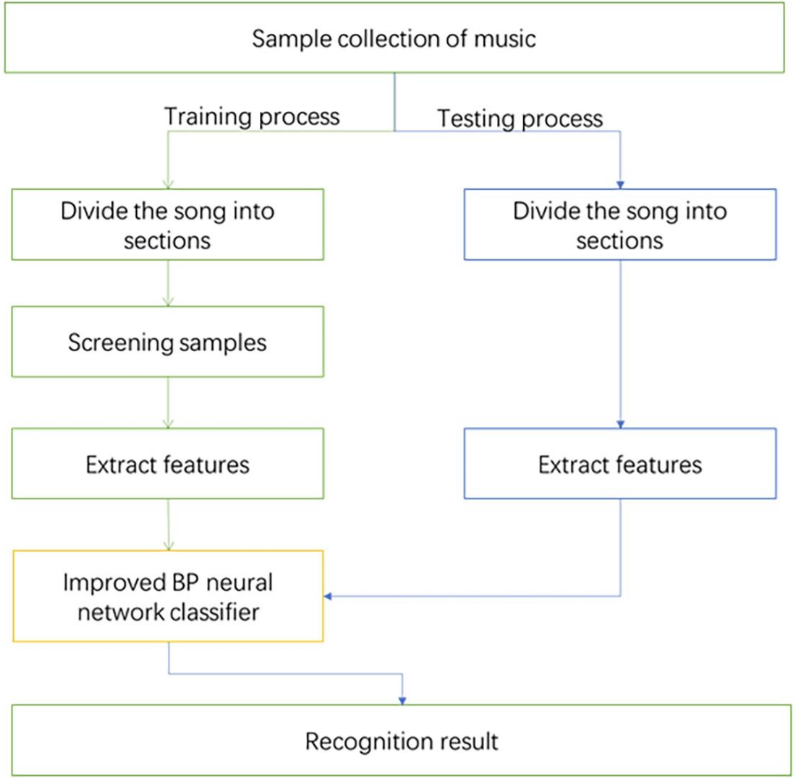
EN.553.602.01.SP24

Introduction

- Our projects examines using **Neural Networks** to interpret emotions in music.
- Improve music recommendations based on listeners’ moods
- Explore the therapeutic potential of music.

Procedure

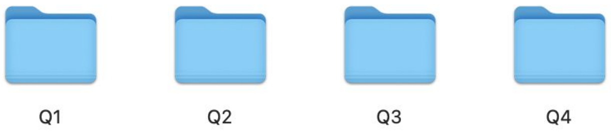
1. Sample a collection of music
2. Divide songs and pick slices
3. Feature extraction
4. Artificial bee colony (ABC) algorithm + Back propagation

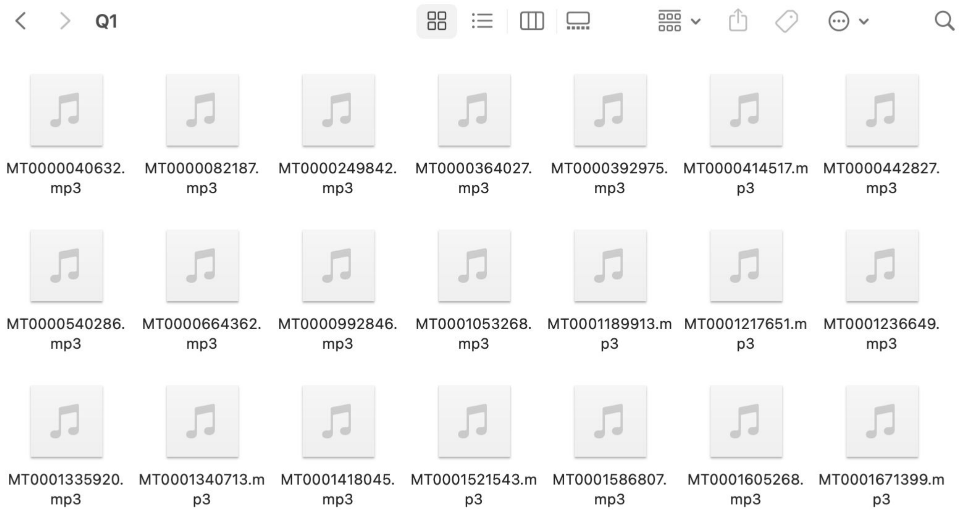


(Jing Yang, 2021)

Data

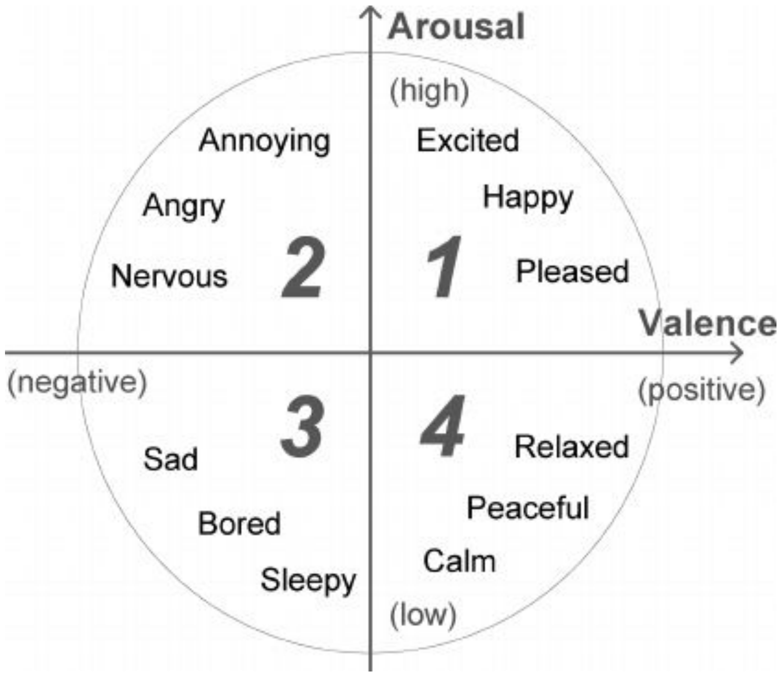
- 900 music samples with each sample 30 seconds long, randomly selected from the whole music.





Music Emotion Recognition Process

- **Input**: the features of each music sample after feature extraction
- **Output:** the 4 quadrants of the Russell's Valence-Arousal Emotion Model



(Jing Yang, 2021)

Dataset (1)

- panda_dataset_taffic_annotations.csv (900 rows, 2 columns)
  Music ID with their Quadrants (We will consider those as our labels).

Song	Quadrant
MT0000004637	Q3
MT0000011357	Q2
MT0000011975	Q2
MT0000040632	Q1
MT0000044741	Q3
MT0000054705	Q4
MT0000082187	Q1
MT0000088320	Q3
MT0000092267	Q4

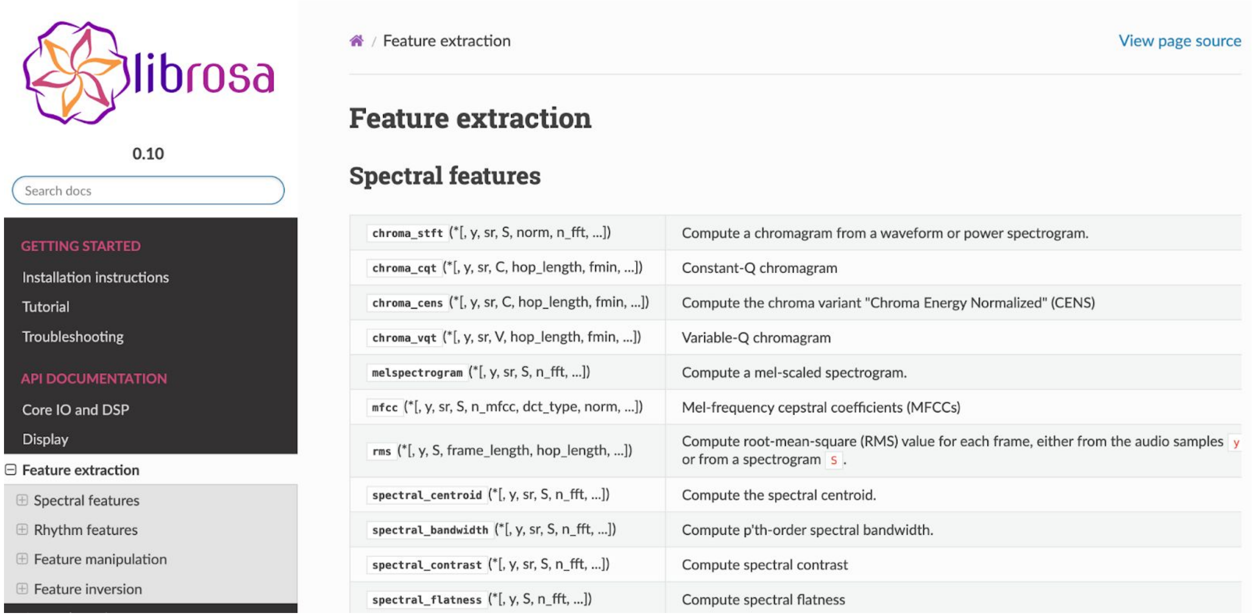
Dataset (2)

- panda_dataset_taffic_metadata.csv (900 rows, 14 columns)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Song	Artist	Title	Quadrant	PQuad	Moods Total	Moods	Moods FoundStr	MoodsStr	MoodsStr Split	Genres	GenresStr	Sample	SampleURL
2	MT0000004637	Charlie Poole	Bulldog Down in Sunny Tennessee	Q3	0.66666667	3	3	circular; greasy; messy	Circular; Greasy; Messy	Circular; Greasy; Messy	2	Country; International	1	http://rovimusic.rovicorp.com/playback.mp3?c=loVS25PIM05
3	MT0000011357	Dismember	Reborn in Blasphemy	Q2	0.66666667	3	3	jittery; negative;	Negative; Nervous/Jittery	Negative; Nervous; Jittery	3	Electronic; International; Pop/Rock	1	http://rovimusic.rovicorp.com/playback.mp3?

Featurization of music

- Turn audio clips into trainable vectors: Librosa package



librosa

0.10

Search docs

GETTING STARTED

Installation instructions

Tutorial

Troubleshooting

API DOCUMENTATION

Core IO and DSP

Display

⊟ Feature extraction

⊞ Spectral features

⊞ Rhythm features

⊞ Feature manipulation

⊞ Feature inversion

/ Feature extraction

View page source

Feature extraction

Spectral features

`chroma_stft` (*[, y, sr, S, norm, n_fft, ...])	Compute a chromagram from a waveform or power spectrogram.
`chroma_cqt` (*[, y, sr, C, hop_length, fmin, ...])	Constant-Q chromagram
`chroma_cens` (*[, y, sr, C, hop_length, fmin, ...])	Compute the chroma variant "Chroma Energy Normalized" (CENS)
`chroma_vqt` (*[, y, sr, V, hop_length, fmin, ...])	Variable-Q chromagram
`melspectrogram` (*[, y, sr, S, n_fft, ...])	Compute a mel-scaled spectrogram.
`mfcc` (*[, y, sr, S, n_mfcc, dct_type, norm, ...])	Mel-frequency cepstral coefficients (MFCCs)
`rms` (*[, y, S, frame_length, hop_length, ...])	Compute root-mean-square (RMS) value for each frame, either from the audio samples `y` or from a spectrogram `S`.
`spectral_centroid` (*[, y, sr, S, n_fft, ...])	Compute the spectral centroid.
`spectral_bandwidth` (*[, y, sr, S, n_fft, ...])	Compute p'th-order spectral bandwidth.
`spectral_contrast` (*[, y, sr, S, n_fft, ...])	Compute spectral contrast
`spectral_flatness` (*[, y, S, n_fft, ...])	Compute spectral flatness

https://librosa.org/doc/latest/feature.html

Featurization of music

- Alternative feature extraction package: musif, openSMILE

In this work, we introduce musif, a Python package that facilitates the automatic extraction of features from symbolic music scores. The package includes the implementation of a large number of features, which have been developed by a team of experts in musicology, music theory, statistics, and computer science. 2023年7月3日

arXiv
https://arxiv.org › cs

musif: a Python package for symbolic music feature extraction

openSMILE
audio feature extraction

release v3.0.2 | release date october 2023 | downloads 14k | docs 3.0

openSMILE (**open**-source **S**peech and **M**usic **I**nterpretation by **L**arge-space **E**xtraction) is a complete and open-source toolkit for audio analysis, processing and classification especially targeted at speech and music applications, e.g. automatic speech recognition, speaker identification, emotion recognition, or beat tracking and chord detection.

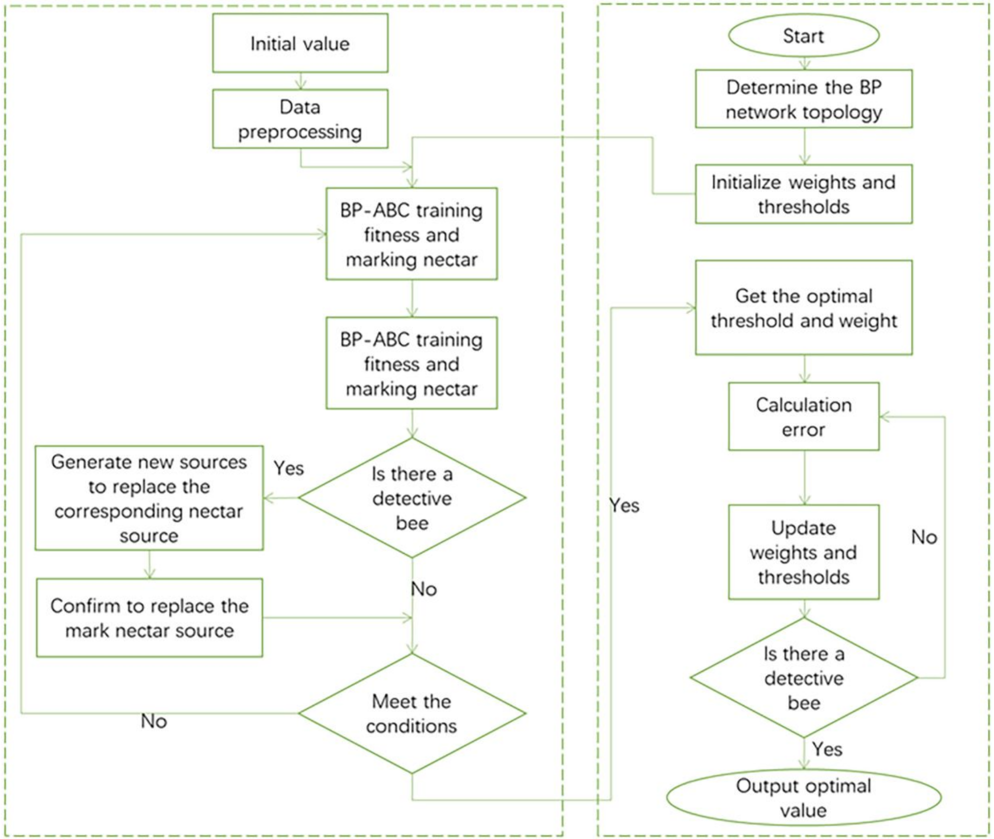
It is written purely in C++, has a fast, efficient, and flexible architecture, and runs on desktop, mobile, and embedded platforms such as Linux, Windows, macOS, Android, iOS and Raspberry Pi.

See also the standalone opensmile Python package for an easy-to-use wrapper if you are working in Python.

Model

- Back Propagation Neural Network(BP NN) + Artificial Bee Colony(ABC)

- The introduce of ABC can improve the **global search ability** of BP



Future Work

- Employ Librosa to convert audio clips into feature vectors.
- Construct a model combining Back Propagation Neural Networks (BP NN) with the Artificial Bee Colony (ABC) algorithm
- Assess the necessity of hyperparameter tuning to enhance our model's predictive accuracy.
- Select a test set of 100 audio clips at random from a collection of 900 samples for evaluation.

[1] Yang, J. (2021) ‘A novel music emotion recognition model using neural network technology’, *Frontiers in Psychology*, 12. doi:10.3389/fpsyg.2021.760060.
[2] Panda, R., Malheiro, R. and Paiva, R.P. (2020) ‘Novel audio features for Music Emotion Recognition’, *IEEE Transactions on Affective Computing*, 11(4), pp. 614–626. doi:10.1109/taffc.2018.2820691.