

# Multiple-part 学習による個人感性獲得機構

## A study on acquiring sensibility information by multiple-part learning

西川 敬之\*<sup>1</sup>    福井 健一\*<sup>2</sup>    森山 甲一\*<sup>2</sup>    栗原 聡\*<sup>2</sup>    沼尾 正行\*<sup>2</sup>  
 Takayuki Nishikawa    Ken-ichi Fukui    Koichi Moriyama    Satoshi Kurihara    Masayuki Numao

\*<sup>1</sup>大阪大学大学院 情報科学研究科 情報数理学専攻

Department of Information and Physical Science, Graduate School of Information Science and Technology, Osaka University

\*<sup>2</sup>大阪大学産業科学研究所

The Institute of Scientific and Industrial Research, Osaka University

We develop a system that automatically composes music adapted to one's sensibility as a part of a study to learn human sensibility. The system learns music structures by first order logic that evokes subjects' specific feelings from their evaluations of existing tunes and the music structures of them. In this paper, we view this learning problem as a problem of multiple-part learning, where the task is to learn a concept given positive and negative instances composed of many parts. We view tunes as data composed of many chords and calculate the importance of each chord, so that we incorporate it into the search heuristics.

### 1. はじめに

本研究室では、人の感性を学習する研究の一環として、個人の感性に合わせた音楽を自動作曲するシステムの開発を行ってきた。本システムは、既存の楽曲に対するユーザの形容詞対ごとの評価データと、それらの楽曲の構造から、ユーザに特定の感情を喚起させる音楽構造を一階述語論理形式で学習する。

従来の研究では、ユーザの評価は楽曲全体の構造に対して与えられたものとして学習を行っていたが、本研究では、楽曲内の部分構造、すなわち一部の和音の結びつきがその楽曲に対するユーザの評価決定に強く影響していると考え、その部分構造からユーザの感性を喚起する楽曲構造を学習することを考える。(図 1) そのため、楽曲を multiple-part データ、すなわち複数の和音によって構成されているデータであると見なし、この学習問題の枠組みを従来の教師付き学習から multiple-part 学習へと変更する。この変更により、楽曲内の部分構造ごとの差を考慮することになり、より一般的な感性情報の獲得が可能になる。

### 2. Multiple-part データと学習

Multiple-part データ [2] とは、1 つのデータ例が複数の要素 (parts) で構成され、それらの要素間に関係 (relation) が存在するデータのことを指す。例として、化合物を multiple-part データとして一階述語論理形式で表現した例を図 2 に示す。これは、化合物  $c_1$  が 6 個の炭素 (c) 原子  $a_1, \dots, a_6$  から成り、単結合と二重結合が交互に並んでいることを示している。本研究では、楽曲を構成する和音を part、楽曲

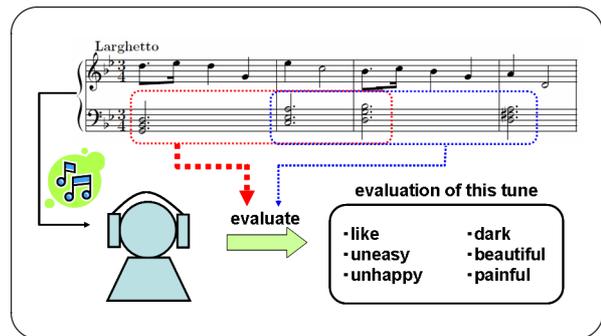


図 1: 本研究におけるユーザの楽曲評価の考え方

の枠組み構造 (拍子, テンポなど) を relation として扱う。4.1 節で詳説する。

Multiple-part 学習 [2] とは、multiple-part データの中のある部分構造、すなわちある parts 同士の結びつきがそのデータ全体のラベルを決定する考え、その部分構造から仮説を発見することを目的とする。そのため、データを構成する part ごとに重みを与え、その重みをヒューリスティックとして用いて学習を行う。これにより、学習における探索空間を効果的に絞り込むことができる。Nattee ら [2] は multiple-part 学習における学習手法として、part の重み付けに Diverse Density (DD) [3] を利用し、与えた重みに基づいて帰納論理プログラミング (ILP) の一つである FOIL の探索を行う手法 (FOILMP, FOIL for Multiple-Part data) を提案している。

#### FOIL

FOIL [1] は Quinlan により開発されたトップダウン型の ILP システムである。トップダウン型 ILP システムは、目標概念 (目標述語) を頭部とし、空の本体部を持つホー

連絡先: 沼尾研究室, 大阪大学産業科学研究所,  
 〒 567-0047 大阪府茨木市美穂ヶ丘 8-1,  
 Tel:06-6879-8426, Fax:06-6879-8428,  
 E-mail:nishikawa@ai.sanken.osaka-u.ac.jp

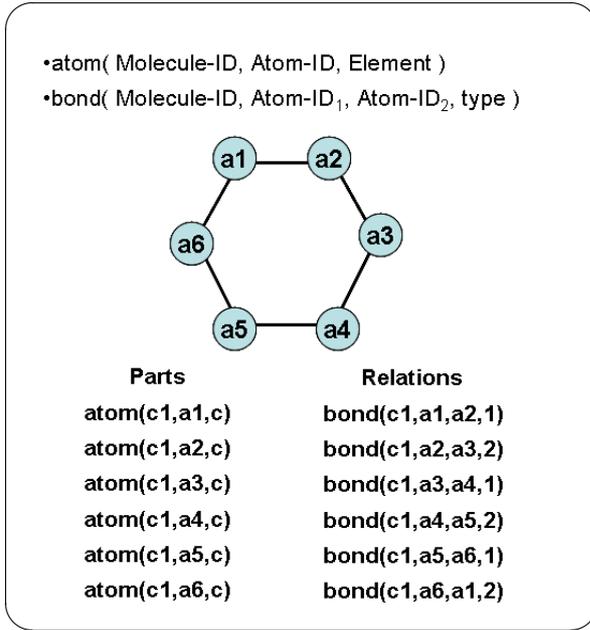


図 2: Multiple-part データの例

ン節に、背景知識として定義されたりテラルを追加して特殊化することにより、目標概念をホーン節の形で獲得する。FOIL は追加するリテラルの評価に情報理論に基づく Gain ヒューリスティックを利用したものである。

生成中の節にリテラル  $L_i$  を追加するとき、 $L_i$  を追加する時点で節によって説明されている正例の数を  $T_i^+$ 、負例の数を  $T_i^-$  とすると、全訓練例の中の一つの例が正例であるかどうかを知るのに必要な情報量  $I(T_i)$  は

$$I(T_i) = -\log_2 \frac{T_i^+}{T_i^+ + T_i^-} \quad (1)$$

である。同様に  $L_i$  を追加した後の情報量  $I_{i+1}$  は

$$I(T_{i+1}) = -\log_2 \frac{T_{i+1}^+}{T_{i+1}^+ + T_{i+1}^-} \quad (2)$$

である。このとき Gain ヒューリスティックは

$$Gain(L_i) = T_i^{++} \times (I(T_i) - I(T_{i+1})) \quad (3)$$

となる。ここで、 $T_i^{++}$  は  $T_i^+$  のうち  $L_i$  を満たす正例の数である。

#### FOILMP

FOILMP では、part の重みとして DD 値を計算し、その値を探索に用いる。DD は正の訓練例内の多くの parts が持っている属性を持つ part に対して大きな値を与え、負の訓練例内の多くの parts が持っている属性を持つ part に対して小さな値を与える。

DD 値を探索に用いるために FOIL のヒューリスティック関数を以下のように変更する。ここで、 $B_i^+$  は正例、 $B_{ij}^+$

は  $B_i^+$  に含まれる  $j$  番目の part、 $B_{ijk}^+$  は  $B_{ij}^+$  の  $k$  番目の属性を表すとする。同様に、 $B_{ij}^-$  は負例に含まれる part を表す。

$$Gain(L_i) = DD_s(T_i^{++}) \times (I(T_i) - I(T_{i+1})) \quad (4)$$

$$DD_s(T) = \sum_{T_i \in T} DD(T_i) \quad (5)$$

$$I(T_i) = -\log_2 \frac{DD_s(T_i^+)}{DD_s(T_i^+) + |T_i^-|} \quad (6)$$

但し

$$DD(x) = \prod_i (1 - \prod_j (1 - \exp(-\|B_{ij} - x\|^2))) \cdot \prod_i \prod_j (1 - \exp(-\|B_{ij}^- - x\|^2)) \quad (7)$$

$$\|B_{ij} - x\|^2 = \sum_k (B_{ijk} - x_k)^2 \quad (8)$$

$\|B_{ij} - x\|^2$  は、述語で表現されている属性の差に基づいて計算される。同じ属性であれば 0、違う属性であれば 1 という値を与える。

探索の際、生成中の節に候補リテラルが追加されると、追加後の節が説明する全ての parts の DD 値の和から Gain が計算される。追加リテラルが relation に関するリテラルである場合は、それを追加した節によって説明される relation を構成している parts の DD 値の平均をその relation の DD 値として扱う。

FOILMP の基本のアルゴリズムは FOIL アルゴリズムと同じであるが、FOIL では探索アルゴリズムとして山登り探索を採用しているのに対し、FOILMP では探索アルゴリズムとしてビーム探索を採用している。さらに、生成するルールの適用範囲を広げるために被覆率下限パラメータ  $\gamma$  を設定し、それによる精度の低下を防ぐために精度下限パラメータ  $\epsilon$  を設定している。FOILMP のルール生成部のアルゴリズムを図 3 に示す。

### 3. 帰納論理プログラミングによる感性情報の学習

本研究における目的は、ある感性を喚起する音楽構造の発見である。高い精度で感性情報を獲得できるかどうかは以下で述べる音楽知識表現と目標述語の定義に依存する。

#### 3.1 楽曲の述語表現

本研究では、楽曲の枠組み構造と和音構造の 2 種類を扱う。楽曲を以下の述語で表現する (述語名/引数の数)。

- song\_frame/7: 楽曲の枠組を表す述語であり、その引数はテンポ、拍子、楽器等について示す。

```

Initialize Beam with an empty rule
Do
  NewBeam ← {}
  For each clause C in Beam
    Generate Candidate by adding all possible
    literals to C
    For each new clause nC in Candidates
      Calculate heuristic of nC using DD values
      Append nC to NewBeam
  Beam ← Best BeamWidth clauses in NewBeam
  R ← Best clause in Beam
Until Accuracy(R) > ε and PositiveCoverage(R) > γ
Return R
    
```

図 3: FOILMP のルール生成アルゴリズム

- chord/9: 和音構造を表す述語であり、主調、根音、形  
体指数等について示す。
- music/2: 楽曲全体を表す述語であり、第一引数に  
song\_frame/7 を、第二引数に chord/9 のリストをと  
る。訓練例となる楽曲は、全てこの music/2 の形式で  
作成されている。

### 3.2 目標述語

以上に述べた知識表現を用い、ユーザの感性を刺激する楽曲構造についての述語獲得を行う。述語獲得には、楽曲の枠組み構造に関する frame/1、楽曲中に含まれる和音進行に関する pair/2、triplet/3 の 3 種類の述語を個別に獲得する手法と、楽曲の枠組み構造と和音進行の両者に関する述語 tune/1 のみを獲得する手法がある。[4]

### 3.3 ユーザによる楽曲評価と訓練例生成

目標述語の学習には FOIL を使用する。そのために、訓練例楽曲から正例と負例を生成する。

まず、ユーザは楽曲を 6 つの形容詞対について 5 段階評価する。評価の仕方については、例えば「好き」な曲ほど 5 段階評価の高い数値を与えるように、また「嫌い」な曲ほど 5 段階評価の低い数値を与えるようにユーザに指示する。他の形容詞対についても同様に「正方向」「負方向」をあらかじめ定義し、5 段階評価と対応させている。

得られた評価値と 3.1 節で説明した述語で表現された楽曲構造から、3.2 節で述べた目標述語それぞれ専用の訓練例を生成する。正方向の形容詞については、先の評価により一定値以上の評価を得た楽曲の構造から生成した訓練例を正例、それ未満の評価を得た楽曲の構造から生成した訓練例を負例とする。負方向の形容詞については、先の評価により一定値以下の評価を得た楽曲の構造から生成した訓練例を正例、それより高い評価を得た楽曲の構造から生成した訓練例を負例とする。

frame/1 に対する訓練例は、song\_frame/7 から生成する。pair/2 に対する訓練例は、2 つの連続する chord/9 から生成する。

triplet/3 に対する訓練例の生成法は、2 つではなく 3 つの連続する chord/9 を使用するという点を除いて pair/2 の場合と同様である。また、tune/1 に対する訓練例は、music/2、すなわち song\_frame/7 と chord/9 から生成する。

## 4. 提案手法

楽曲構造からの感性情報の獲得を multiple-part 学習の問題として捉え、2. で説明した FOILMP を用いて学習を行う。以下で、楽曲を multiple-part データとして扱い、FOILMP で学習を行うための手法を説明する。

### 4.1 楽曲の multiple-part データ表現

楽曲を multiple-part データとして扱うために、新たに 2 つの述語 has\_chord/3 と successive\_chords/4 を定義する。has\_chord/3 は楽曲内の 1 和音を 1 つの part として扱うための述語、successive\_chords/4 は楽曲中で連続する 2 和音の relation を表すための述語である。それぞれ以下の引数を持つ。

- has\_chord(Music-ID, Chord-ID, chord/9)
- successive\_chords(Music-ID, Chord-ID1, Chord-ID2, song\_frame/7)

これらの述語を用いて楽曲を multiple-part データとして表現した例を図 4 に示す。

```

ex(tune(1,+).
has_chord(1,1,chord(c(0),dur,2,5,0,nil,nil,nil,subdominant)).
has_chord(1,2,chord(c(0),dur,4,5,0,nil,nil,nil,subdominant)).
has_chord(1,3,chord(c(0),dur,1,7,0,nil,nil,nil,tonic)).
successive_chords(1,1,2,song_frame(lento,four_four,piano,piano,c(0),dur,other)).
successive_chords(1,2,3,song_frame(lento,four_four,piano,piano,c(0),dur,other)).
    
```

図 4: 楽曲の multiple-part データ表現の例

chord/9 の属性の差に基づいて DD を計算し、part としての和音に重みを与える。

### 4.2 目標述語

Multiple-part データとして表現した楽曲を訓練例として扱うには、和音と和音間の関係を同時に扱えるように目標述語を定義する必要がある。そのため、目標述語には 3.2 節で述べた tune/1 を用いる。以下にあるユーザが「明るい」と感じる楽曲構造を示す。

```

tune(M) :- has_chord(M,A,B),
           inversion_I(B),
           successive_chords(M,A,C,D),
           tempo_allegro(D),
           has_chords(M,C,E),
           inversion_I(E).
    
```

上記の述語例は、あるユーザが「明るい」と感じる楽曲は、テンポがアレグロのときに連続する2和音が共に第一転回形であるという楽曲構造を持つということを表している。

tune(A) :- has\_chord(A,\_,B),napori\_VI(B).

tune(A) :- has\_chord(A,B,C),inversion\_II(C),moll(C),  
successive\_chords(A,B,\_,D),tempo\_andante(D).

tune(A) :- has\_chord(A,\_,B),inversion\_II(B),moll(B),  
dominant(B).

## 5. 実験

4. で提案した学習手法で実際に個人の感性情報を学習できるかどうかを確かめるための実験を行う。以下に実験の流れを示す。

- 32 人分の被験者データ (訓練曲 75 曲に対する 5 段階評価データ) を用いて、表 1 に示す 6 形容詞対・12 形容詞についての感性情報を FOIL を使って学習する。
- 同じデータを用いて、12 形容詞についての感性情報を FOILMP を使って学習する。なお、FOILMP の被覆率下限パラメータと精度下限パラメータはそれぞれ 0.15 と 0.95 とし、ビーム幅は 1 とする。
- それぞれの学習結果の正答率及び被覆度を、5-fold の Cross-Validation により評価し、両者を比較する。

表 1: 形容詞対

種類	正方向	負方向
嗜好度	好き	嫌い
明るさ	明るい	暗い
安心度	安心	不安
美しさ	美しい	汚い
嬉しさ	嬉しい	悲しい
切なさ	切ない	切なくない

実験結果を表 2 に示す。表の正答率及び被覆度の値は、全被験者データに対する 5-fold Cross-Validation の全 fold の平均正答率と平均被覆度である。

表 2: 平均正答率と平均被覆度

形容詞	正答率 (%)		被覆度 (%)	
	FOIL	FOILMP	FOIL	FOILMP
好き	78	75	19	28
嫌い	89	87	29	31
明るい	78	72	29	33
暗い	84	82	36	48
安心	79	76	19	21
不安	87	86	21	35
美しい	76	73	17	25
汚い	92	90	21	12
嬉しい	80	77	25	30
悲しい	85	83	28	43
切ない	76	73	27	30
切なくない	75	72	34	38

これより、被覆度において従来の学習手法から改善が見られ、より一般的な学習結果が得られたと言える。

FOILMP による学習の結果得られた、ある被験者が「切ない」と感じる楽曲構造の例を以下に示す。

これは、ある被験者が「切ない」と感じるのは、(1) ナボリの 6(和音の特殊変化形) であるような和音を持つか、(2) テンポがアンダンテのときに第二転回形で調性が短調の和音を持つか、もしくは(3) 第二転回形で調性が短調、機能がドミナントの和音を持つような楽曲であることを表している。それぞれのルールの正答率は順に 91%, 91%, 92%, 被覆度は順に 45%, 36%, 55%であった。

## 6. まとめ

本研究では、より一般的な感性情報の獲得を目指し、当研究室で開発されている自動作曲システムの学習段階において、学習の枠組みを従来の教師付き学習から multiple-part 学習へと変更した。その結果、学習結果として簡潔で適用範囲の広いルールを得ることができた。

## 参考文献

- [1] J.R.Quinlan. Learning logical definitions from relations. *Machine Learning*, Vol.5, pp. 239-266,1990.
- [2] Cholwich Nattee, Sukree Sinthupinyo, Masayuki Numao, and Takashi Okada. Learning First-order Rules from Data with Multiple Parts: Applications on Mining Chemical Compound Data. *Proc. the 21st International Conference on Machine Learning*, pp. 606-614,2004.
- [3] Oded Maron and Tomás Lozano-Pérez. A Framework for Multiple-Instance Learning. *Advances in Neural Information Processing Systems*, The MIT Press,1998.
- [4] Roberto Legaspi, Yuya Hashimoto, Koichi Moriyama, Satoshi Kurihara and Masayuki Numao. Music Compositional Intelligence with an Affective Flavor. *Proc. ACM International Conference on Intelligent User Interfaces*, pp. 216-224,2007.