

感性に基づく楽曲生成のための背景知識自動作成

Generation of Background Knowledge for Music Composition Based on Personal Sensibility

土屋 直樹
TSUCHIYA, Naoki

概要: 個人の感性を反映した楽曲生成システムの研究が進められている。本システムでは、評価用楽曲に対する個人の評価をもとに訓練例を作成し、評価用楽曲の構造に関する記述を背景知識として、帰納論理プログラミングにより感性モデルを獲得する。獲得した感性モデルに基づき、進化計算アルゴリズムによって枠組構造、和音進行、メロディからなる楽曲を生成する。現在、背景知識は楽譜を基に手作業で作成しており、大量の評価用楽曲を用意することができない。また、楽譜にコードネームが記載されていない場合、コードネームの導出が必要となる。本研究では、感性に基づく楽曲生成のための背景知識自動作成システムを構築する。入力データに MusicXML を用い、コードネームが記述されていない場合、コードネームを推定し背景知識を自動作成する。

Summary: Automatic composition system that composes music adapting personal sensibility has been proposed. The system induces personal sensibility models using inductive logic programming from training data and background knowledge and composes music based on the models. In order to generate more precise personal sensibility models, background knowledge of many musical pieces should be prepared. However, it is difficult to generate background knowledge by hand. In addition, when a score has no chord names, it is necessary to derive chord names. In this study, a system is developed to generate background knowledge for music composition based on personal sensibility. In the case that a score has no chord names, the system estimate chord names and generate background knowledge from MusicXML.

キーワード: 条件付き確率場・機械学習・系列ラベリング・和音推定

Keywords: Conditional Random Field, Machine Learning, Sequential Labeling, Chord Estimation

1. はじめに

現在、個人の感性を反映した楽曲生成システムの研究が進められている[1]。本システムでは、評価用楽曲に対する個人の評価をもとに訓練例を作成し、評価用楽曲の構想に関する記述を背景知識として、帰納論理プログラミング (Inductive Logic Programming; ILP) により枠組構造、モチーフ、和音進行、メロディの感性モデルを獲得する。獲得した感性モデルに基づき、楽曲を生成する。感性モデルは、聴者のある感性を反映した評価用楽曲に共通する楽曲の構造である。評価用楽曲が多いほど、聴者の特定感性を想起させる楽曲数が増え、多様な楽曲生成が可能になると考えられる。現在、背景知識は楽譜をもとに手作業で作成しており、大量の評価用楽曲を用意することができない。また、楽譜にコードネームが記載されていない場合、コードネームの導出が必要となる。

本研究では、入力データとして MusicXML を用い、感性に基づく楽曲生成のための背景知識自動作成システムを構築する。本システムは、コンバータ機能とコードネーム推定機能の2つの機能を有する。コンバータ機能では、MusicXML から背景知識を自動で作成し、コードネーム推定機能では、事前学習したコードネーム推定モデ

ルに MusicXML から抽出した素性を入力し、コードネームを推定する。

コードネームの推定には、条件付き確率場 (Conditional Random Field; CRF) を用いる。CRF によるコードネームの推定手法として、陳らの研究[2]が存在するが、ソプラノとバスの2声部から構成された楽曲を解析対象としているため、本研究において解析対象とする楽曲に適用することができない。本研究では、解析対象とする楽曲に適用させるために、コードネームの表現、素性関数の設計を新たに定義し、コードネーム推定機能で利用する。

2. 感性に基づく楽曲生成システム

感性に基づく楽曲生成システムにおける楽曲は、枠組構造、複数のモチーフから構成される和音進行、メロディの3つの要素から構成される。モチーフとは、楽曲を構成する最小単位であり、2小節で構成される。枠組構造は、楽曲の調、テンポの2つの要素から構成される。和音進行は、和音の並びを表すもので、和音は Root, Type, Tension の3要素の組 (Root, Type, Tension) として表現される。Root は和音の根音、Type は和音の種類、Tension は付加的に用いられる音を表す。

楽曲の生成手順を図1に示す。まず、評価用楽曲に対

本研究の一部は、情報処理学会第79回全国大会にて発表予定である。

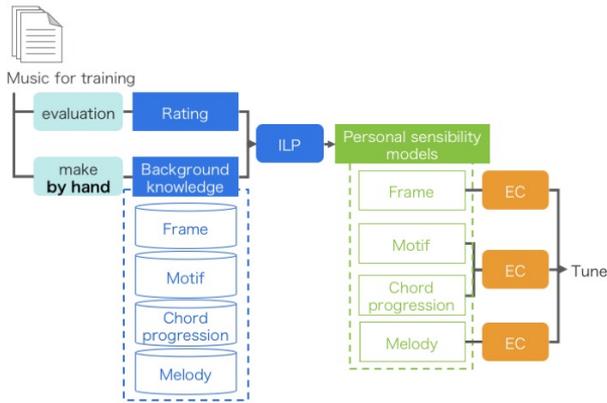


図1 楽曲生成手順

する聴者の評価をもとに訓練例を作成し、評価用楽曲の構造に関する記述を背景知識とし、ILPにより枠組構造、モチーフ、和音進行、メロディの感性モデルを獲得する。

獲得した感性モデルから、進化計算アルゴリズムにより、枠組構造、和音進行、メロディを生成し、3つを組み合わせる楽曲とする。

3. 背景知識自動作成システム

本システムは、評価用楽曲の充実化を目的としている。入力データとして MusicXML を用い、背景知識を自動作成する。MusicXML から背景知識を作成するコンバータ機能と、コードネームが記述されていない MusicXML からモチーフ、和音進行の背景知識を作成するためのコードネーム推定機能の2つの機能を有する。システムの構成図を図2に示す。

なお、本システムで解析対象とする楽曲の形式は、大谷らの楽曲生成システムにおける楽曲と同様とする。本研究では、backup 要素以前の note 要素をメロディ、backup 要素後の note 要素を和音進行、harmony 要素をコードネームとする。

4. コンバータ機能

コンバータ機能は、MusicXML から背景知識を作成する機能である。

4.1 枠組構造作成処理

枠組構造は、楽曲の調と BPM から構成される。楽曲の調は式(5.1)で決定する。ここで、 $fifth$ は $fifth$ 要素の値、 $n2f()$ は、音名を $fifth$ 要素の値に変換する関数。 $note1$ は楽曲の最後に登場するベース音、 $note2$ は、楽曲の最初に登場するベース音を表す。BPMは MusicXML の tempo 要素の値で決定する。

$$key = \begin{cases} major & \text{if } fifth = n2f(note1) \\ major & \text{if } fifth = n2f(note2) \\ minor & \text{otherwise} \end{cases} \quad (5.1)$$

4.2 メロディ作成処理

メロディは measure 要素内の backup 要素以前に出現する note 要素内の要素の値を用いる。note 要素内の octave 要素の値でオクターブ、pitch 要素と alter 要素の値で音名、式(5.2)で音価が決定する。ここで、 $duration$ は

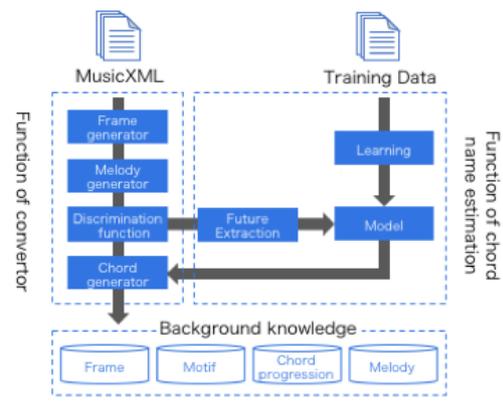


図2 システム構成図

$duration$ 要素の値、 $divisions$ は $divisions$ 要素の値を示す。ただし、note 要素内に rest 要素がある場合、オクターブ、音名は共に rest とする。例えば、octave 要素が「4」、pitch 要素が「C」、alter 要素が「1」、duration 要素が「4」、divisions 要素が「4」であれば、オクターブは「4」、音名は「C#」、音価は「4」となる。

$$actualDuration = \frac{(duration * 4)}{divisions} \quad (5.2)$$

4.3 和音進行作成処理

和音進行作成処理では、measure 要素内の harmony 要素の数や位置によって和音の作成方法が異なる。なお、コードネーム記述の有無は、楽曲の最初の measure 要素内の attributes 要素直後、backup 要素の直後のどちらかに harmony 要素があればコードネームあり、なければなしと判別される。

harmony 要素が measure 要素内に存在しないときには、直前の和音の音価を1小節分増やす。

harmony 要素が measure 要素内に存在する場合、harmony 要素の位置によっては追加処理が発生する。

- harmony 要素が backup 要素の前にある場合
harmony 要素の値から和音を作成し、backup 要素または次の harmony 要素までに存在する note 要素の音価の合計を和音の音価とする。ただし、measure 要素内の先頭に harmony 要素が存在しないときは、直前の和音の音価を measure 要素内の先頭から harmony 要素の直前までに存在する note 要素の音価の合計分増やした後に上記の処理が発生する。
 - harmony 要素が backup 要素の後にある場合
harmony 要素の値から和音を作成し、measure 要素の末尾または次の harmony 要素までに存在する note 要素の音価の合計を和音の音価とする。ただし、backup 要素の直後に harmony 要素が存在しないときは、直前の和音の音価を measure 要素内の先頭から harmony 要素の直前までに存在する note 要素の音価の合計分増やす。和音の値は、harmony 要素内の値をもとに決定する。
- Root は、root-step 要素と root-alter 要素、Type は kind 要

素, Tension は degree-value 要素と degree-alter 要素の値で決定する。

例えば, root-step 要素が「C」, root-alter 要素が「1」, kind 要素が「major-seventh」, degree-value 要素が「9」, degree-alter 要素が「-1」であれば, Root は「C#」, Type は「M7」, Tension は「b9」となる。

5. コードネーム推定機能

5.1 条件付き確率場

条件付き確率場 (Conditional Random Field; CRF) は, 対数線形モデルを系列ラベリング問題に適用した手法であり, さまざまな素性を学習に用いることができる点, 系列全体におけるグローバルな最適化を求める点に特徴がある。

CRF では, 式(6.1)で条件付き確率 $P(\mathbf{y}|\mathbf{x})$ が表現される。ここで, \mathbf{w} は素性に対する重みベクトル, \mathbf{x} は事例, \mathbf{y} はラベル列, $\phi(\mathbf{x}, \mathbf{y})$ は素性ベクトルを表す。また, $Z_{\mathbf{x}, \mathbf{w}}$ は $\sum_{\mathbf{y}} p(\mathbf{y}|\mathbf{x}) = 1$ を保証する係数で, 式(6.2)と定義される。

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_{\mathbf{x}, \mathbf{w}}} \exp(\mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{y})) \quad (6.1)$$

$$Z_{\mathbf{x}, \mathbf{w}} = \sum_{\mathbf{y}} \exp(\mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{y})) \quad (6.2)$$

CRF 事例を分類するには, 式(6.3)の最大化問題を解くこととなる。しかし, 式(6.3)を解くには一般的に膨大な時間がかかるため, CRF では, 式(6.4)や式(6.5)を仮定し, それぞれ式(6.6), 式(6.7)の最大化問題を解けばよいことになる。ここで y_t は現在のラベル, y_{t-1} は直前のラベルを表す。式(6.6), 式(6.7)は Viterbi アルゴリズムで効率的に解くことができる。

$$\begin{aligned} \mathbf{y}^* &= \operatorname{argmax}_{\mathbf{y}} \frac{1}{Z_{\mathbf{x}, \mathbf{w}}} \exp(\mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{y})) \\ &= \operatorname{argmax}_{\mathbf{y}} \mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{y}) \end{aligned} \quad (6.3)$$

$$\phi_k(\mathbf{x}, \mathbf{y}) = \sum_t \phi_k(\mathbf{x}, y_t) \quad (6.4)$$

$$\phi_k(\mathbf{x}, \mathbf{y}) = \sum_t \phi_k(\mathbf{x}, y_t, y_{t-1}) \quad (6.5)$$

$$\begin{aligned} \mathbf{y}^* &= \operatorname{argmax}_{\mathbf{y}} \mathbf{w} \cdot \sum_t \phi(\mathbf{x}, y_t) \\ &= \operatorname{argmax}_{\mathbf{y}} \sum_t \mathbf{w} \cdot \phi(\mathbf{x}, y_t) \end{aligned} \quad (6.6)$$

$$\begin{aligned} \mathbf{y}^* &= \operatorname{argmax}_{\mathbf{y}} \mathbf{w} \cdot \sum_t \phi(\mathbf{x}, y_t, y_{t-1}) \\ &= \operatorname{argmax}_{\mathbf{y}} \sum_t \mathbf{w} \cdot \phi(\mathbf{x}, y_t, y_{t-1}) \end{aligned} \quad (6.7)$$

CRF では, 式(6.6)や式(6.7)におけるパラメータ \mathbf{w} を学習し, 学習した \mathbf{w} を用いて事例のラベルを予測する。 \mathbf{w} は解析的に解を求めることができないため, 最急降下法

表 1 Type と Tension の構成要素

| カテゴリ | 構成要素 | |
|---------|--|----------------------|
| Type | Type1 | M, m |
| | Type2 | null, aug, dim, sus4 |
| | Type3 | null, 7, M7, 6, add9 |
| Tension | null, b9th, #9th, 9th, 11th, b11th, #11th, b13th, 13th | |

や準ニュートン法などによる数値解析手法を用いる。

5.2 関連研究

陳らは, CRF でソプラノとバスの2声部で構成された楽曲のコードネーム推定を行った。ダイアトニックコードとセカンダリドミナントコードの三和音を解析対象のコードとし, 素性関数として, ソプラノ, バスそれぞれの現在音, 前後音, 前後和音, 前音+次音, 前和音+次和音, 前音+前和音, 次音+次和音, 前音+現在音+次音, 前々和音+前和音, 次和音+次々和音の12種と現在和音との組み合わせを用いている。

5.3 提案手法

5.3.1 コードネームの表現

提案手法では, Root, Type, Tension それぞれでモデルを構築し, 3つの組み合わせにより1つのコードネームを表現する。また, Type はさらに3つのカテゴリ Type1, Type2, Type3 に分割し, それぞれの組み合わせで表現する。Type, Tension の構成要素を表1に示す。ここで, null は該当要素がないことを表す。例えば, コードネームが「C7」であったとすると, 各カテゴリにおける正解ラベルは, Root は「C」, Type1 は「M」, Type2 は「null」, Type3 は「7」, Tension は「null」となる。

5.3.2 素性

和音進行は, 複数の音が同時に演奏されることもあるため, 単純に音名を記述するだけでは, 同時演奏される音を表現することができない。本研究では, 同時演奏される音を表現するために, 12次元クロマベクトルを用いる。12次元クロマベクトルは, 12の音名の出現頻度を表すベクトルで, さまざまな音楽情報処理の研究で利用されている。

扱う素性は, 和音進行, メロディの12次元クロマベクトル, 和音進行とメロディの12次元クロマベクトルの和, ベース音, 調の計38要素とする。入力系列には, 前後2つの素性を含め, 合計190の要素が入力系列となる。

素性関数には, 式(6.4)と式(6.5)を用いる。したがって, 190の要素とラベル数の組み合わせ数と, 190の要素とラベル数の二乗の組み合わせ数の合計が, 素性ベクトルの次元となる。また, 素性関数は, 1または0を返す関数とする。

6. 評価実験

6.1 コンバータ機能の評価

作曲家が本研究のために新たに作成した楽曲48曲を入力とし, 背景知識を作成する。楽曲はすべての調が2曲ずつ含まれており, 1台のピアノで演奏できるようにな

