

Web から入手したデータに基づく

コード進行を利用した楽曲類似度の提案と楽曲視聴支援システムの開発

長澤 慎子 (指導教員：渡辺知恵美)

1 はじめに

近年では、MP3 などの音声圧縮技術の発展や記憶媒体の大容量化に伴い、世の中の人々は大量の楽曲を所有し、様々な音楽を手軽に楽しむことが可能となってきている。それに伴い、音楽情報処理に関する様々な研究が行われている。例えば、小杉ら [1] のハミング検索システムに関する研究や後藤 [2] のサビ区間検出といった研究などが挙げられる。これらは、MIDI データやオーディオデータを対象に行っているため、個人が所有している楽曲に限られてしまう。また、Web 上にある情報を利用した研究として、大坪 [3] の Goromi-Music が挙げられるが、アーティスト名やタイトル名といったメタデータの情報を基に行っているため、音楽的要素を利用して類似した楽曲を検索することができない。そこで我々は、Web 上には膨大な量の情報があり、Web 環境さえあれば誰でも様々な情報の取得が可能であるといった点に着目し、Web 上からアーティスト名やタイトル名といったメタデータの情報に加えて、音楽的要素の一つであるコード進行データを取得し、ヒット曲の音楽的傾向の分析やリスナーの音楽的嗜好の分析を行う。また分析結果に基づいてリスナーの嗜好に合った楽曲の提示を行うシステムを提案する。

2 システム概要

我々は、Web から入手可能なデータをもとにヒット曲の音楽的傾向の分析やリスナーの音楽的嗜好の分析を行う。また分析結果に基づいたリスナーの嗜好に合った楽曲の提示を行うシステムを提案する。我々は、普段生活していて一番身近であり、リアルタイムに情報が更新されている Web 上の情報に対して適していると考えられるポピュラー音楽の楽曲を対象に分析を行う。キーとなる音楽的要素としては楽曲中のコード進行データに着目した。コード進行とは楽曲の旋律に伴う和音(コード)の変化のことであり、コード進行が同一でリズムやメロディをアレンジするだけで同一の楽曲でも印象は異なったものとなる。このことからコード進行はポピュラー音楽において楽曲のベースとなる不変性の高いものであると言える。以上のことから我々はコード進行から楽曲の特徴量を計算し分析を行うこととした。

図 1 に我々が提案するシステムの構成を示す。まず、リスナーの好みの楽曲情報やヒット曲情報をもとに Web 上から対象となる楽曲のコード進行データを抽出する。特徴抽出部ではコード進行データをもとに楽曲の特徴量を抽出し、楽曲情報とともに楽曲の特徴量を楽曲 DB に格納する。楽曲視聴支援に関しては、楽曲 DB 内の特徴量に基づいてクラスタリングを行い、楽曲の分析や検索をすることにより行う。クラスタリングに関しては、楽曲をある一定のサイズで分割を行い、分割したものを Block と名づけ、Block 単位で行う。

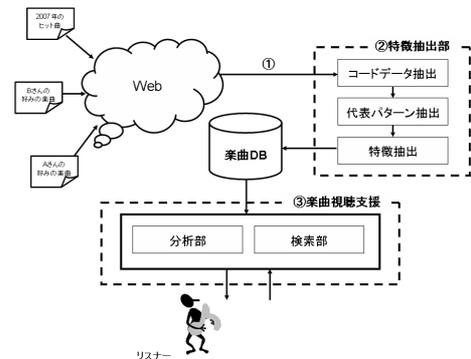


図 1: システム構成

3 楽曲の特徴抽出

3.1 コード間の類似度

コード間の類似度に関しては、音楽理論にて定義されている近親調を用いて定義を行った。コード間の類似度は、図 2 の五度圈上にて、もとなるコードから近いものは類似した関係となっている。図中 で表されているものは近親調の中の「平行調」と呼ばれる関係であり、図中 で表されているものは近親調の中の「属調」と呼ばれる関係である。この二つの関係をもとにコード間の類似度を求める。例えば、C と Cm の類似度は、C と Am の平行調関係による距離 $\text{dist}(\text{平行調}, C \rightarrow Am)$ と、Am と Cm の属調関係による距離 $\text{dist}(\text{属調}, Am \rightarrow Cm)$ の和として計算する。ここに出てくる距離計算方法は文献 [4] に従う。以上の手順で求めたコード間の類似度をもとに Block 間の類似度を求める。

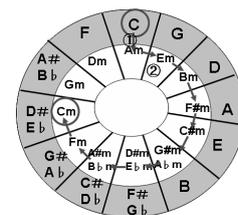


図 2: 五度圈

3.2 定型コード進行パターンを考慮した Block 間の類似度

楽曲や Block は定型コード進行パターンの組合せで構成されている。このことから我々は、定型コード進行パターンを考慮して Block の特徴量を計算し、求めた特徴量を基に Block 間の類似度を求めることとした。

以下に、Block の特徴量抽出手順を示す。

- Step1: Block の分割を行い、Block を「コード進行パターン」の集合として表す。「コード進行パターン」とは分割を行って出来たそれぞれの要素のことである。
- Step2: 「コード進行パターン」をクラスタリングし、「代表コード進行パターン」の抽出を行う。
- Step3: Block 中に出現する「コード進行パターン」の

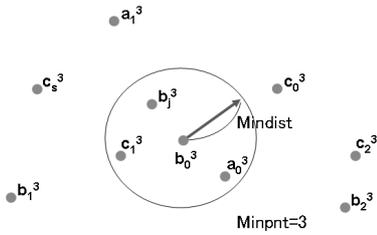


図 3: クラスタリング例 (n=3)

頻度情報と「代表コード進行パターン」のアレンジ情報をもとに Block の特徴量である特徴ベクトルを求める。

次節以降にてそれぞれの Step の詳細を述べる。

3.2.1 Step1: Block の分割

定型コード進行パターンは主に 3 つや 4 つのコードで構成される。このことから、Block を構成するコード進行列を 3-gram, 4-gram に分割することとした。例えば、Block A=(C, G7, Am, F) という Block の分割を行うと以下ようになる。

$$\text{Block A} = \{(C, G7, Am), (G7, Am, F), (C, G7, Am, F)\}$$

以下説明のため、上記の通り分割したものを以下のように表す。

$$\text{Block A} = \{a_0^3, a_1^3, a_0^4\}$$

a_m^n の m は任意の数, n は「コード進行パターン」の長さを表す。よって、 a_1^3 の「コード進行パターン」の長さは 3 となる。

3.2.2 Step2: 代表コード進行パターンの抽出

全ての Block の「コード進行パターン」に対し、長さ n ごとにクラスタリングを行う。例えば、前節にて挙げた Block A の他に、

$$\text{Block B} = \{b_0^3, \dots, b_j^3, b_0^4, \dots, b_k^4\}$$

$$\text{Block C} = \{c_0^3, \dots, c_s^3, c_0^4, \dots, c_t^4\}$$

という Block があつた場合、Block B と Block C の「コード進行パターン」も全て集めて同一の長さの「コード進行パターン」ごとにクラスタリングを行う。クラスタリングを行った際に、半径 Mindist 以内に Minpnt 個以上の「コード進行パターン」を含むものを「代表コード進行パターン」とする。図 3 に、n=3, Minpnt=3 でクラスタリングを行った場合の例を示す。この場合、 b_0^3 が「代表コード進行パターン」となる。

3.2.3 Step3: Block の特徴量抽出

「コード進行パターン」の頻度情報と Step2 で抽出した「代表コード進行パターン」のアレンジ情報をもとに各 Block の特徴ベクトルを求める。前節までに例に挙げた Block とクラスタリング結果をもとに具体例を示す。Block A は a_0^3, a_1^3, a_0^4 が一回ずつ出現し、その他の「コード進行パターン」である $b_0^3, \dots, b_k^4, c_0^3, \dots, c_t^4$ は出現しない。よって、 a_0^3, a_1^3, a_0^4 の部分に関しては 1, その他の部分に関しての頻度情報は 0 となる。ここで、Block 内の各「コード進行パターン」が Step2 で求めた「代表コード進行パターン」と同一のクラスターに含まれているかを調べ、含まれていればそのクラスターの「代表コード進行パターン」に対して重みを加算する。重みの値は、 $1/(\text{「代表コード進行パターン」}$

	a_0^3	...	a_0^4	b_0^3	b_0^4	...
A		...		$1/(\text{dist}(a_0^3, b_0^3)+1)$	0	...

表 1: Block A の特徴ベクトル

と「コード進行パターン」の距離+1) とする。現在示している例では、図 3 より「代表コード進行パターン」は b_0^3 であり、Block A 中の「コード進行パターン」である a_0^3 は b_0^3 と同一のクラスター内にあることから、 b_0^3 をアレンジしたものであると考えられる。よって、 b_0^3 の部分に対して $1/(\text{dist}(a_0^3, b_0^3)+1)$ の重みが加算される。さらにこれらの情報に重みを加えることにより、どのような情報を特徴として重視するかを判別することが可能である。本研究では、「代表コード進行パターン」をどのようにアレンジしているのかといった情報に重きを置きたいと考えているので、アレンジ情報に関する重みの値を大きくしている。頻度情報に関しては、アレンジ情報に関しては重みを加えるとすると、Block A の特徴ベクトルは「コード進行パターン」つまり a_0^3 から a_0^4 の部分に関しては出現回数に重みを掛け合わせた値となり、「代表コード進行パターン」つまり b_0^3 の部分に関しては、 $1/(\text{dist}(a_0^3, b_0^3)+1)$ に重みを掛け合わせたものとなる。表 1 に Block A の特徴ベクトルを示す。

4 楽曲視聴支援

第 3 章の手順にて抽出した楽曲の特徴量をもとに、楽曲の分析や検索を行うことにより楽曲視聴支援を行う。楽曲の分析は、楽曲の特徴量をもとにクラスタリングをすることにより行う。クラスタリング結果を可視化し、可視化結果を基にユーザが対話的に分析を行うことにより楽曲分析を行う。可視化には、伊藤らによって開発された平安京ビュー [5] を用いた。楽曲の検索は楽曲の特徴量をもとに対象の楽曲の A メロ, B メロ, サビと似た楽曲の検索を行う。

5 まとめ

本研究では、Web から入手したデータに基づいた楽曲の分析を行うための、コード進行に関する様々な定義を行い、楽曲視聴支援ツールの開発を行った。これにより、今までとは違った側面から音楽を楽しむことが可能となった。

参考文献

- [1] 小杉尚子, 小島明, 片岡良治, 串間和彦: “大規模音楽データベースのハミング検索システム,” 情報処理学会論文誌, Vol.43, No.2, pp.287–298, 2002.
- [2] 後藤真孝: “リアルタイム音楽情景記述システム: サビ区間検出手法,” 音楽情報科学研究会研究報告, Vol.2002, No.100, pp.27–34, 2002.
- [3] 大坪五郎: “Goromi-Music 音楽をより楽しむためのインタフェース,” WISS2007, 2007.
- [4] 長澤慎子, 渡辺知恵美, 伊藤貴之: “ポピュラー音楽クラスタリングのための近親調を用いたコード進行類似度の提案,” 音楽情報科学研究会研究報告, Vol.2007, No.37, pp.69–79, 2007.
- [5] 伊藤貴之, 山口裕美, 小山田耕二: “長方形の入れ子構造による階層型データ視覚化手法の計算時間および画面占有面積の改善,” 可視化情報学会論文集, Vol.26, No.6, pp.51–61, 2006.