

定型コード進行パターンに着目した ポピュラー音楽クラスタリング手法の提案

長澤 慎子[†] 渡辺 知恵美[‡] 伊藤 貴之[‡]

[†]お茶の水女子大学大学院 人間文化研究科 数理・情報科学専攻 〒112-8610 東京都文京区大塚 2-1-1

[‡]お茶の水女子大学大学院 人間文化創成科学研究科基幹部門 〒112-8610 東京都文京区大塚 2-1-1

E-mail: [†] makiko@db.is.ocha.ac.jp, [‡] {chiemi, itot}@is.ocha.ac.jp

あらまし 世の中には様々なジャンルの楽曲があり、クラシック、ジャズ、ポップスなどが挙げられる。我々はその中のポップスに着目し楽曲と作曲者、ヒット曲、年代、ユーザの好みの楽曲との関係のマイニングを行う。これらの実現に向け、ポピュラー音楽において、楽曲のベースとなり不変性の高いコード進行に着目しクラスタリングを行う。本稿では、先行研究で行った予備実験の結果から他属性との相関関係についての考察を可視化により行う。また、先行研究で定義したコード進行の類似度の定義を拡張し、コード進行をつながりとしてみた場合のコード進行間の類似度測定法を提案する。

キーワード クラスタリング、マイニング、音楽情報処理、可視化

Proposal on Popular Music Clustering Method Focused the Chord Progression Pattern

Makiko NAGASAWA[†] Chiemi WATANABE[‡] and Takayuki ITO[‡]

[†] Graduate Division of Mathematics and Computer Science, Ochanomizu University

2-1-1 Otsuka, Bunkyo-ku, Tokyo, 112-8610 Japan

[‡] Department of Natural and Advanced Science, Ochanomizu University

2-1-1 Otsuka, Bunkyo-ku, Tokyo, 112-8610 Japan

E-mail: [†] makiko@db.is.ocha.ac.jp, [‡] {chiemi, itot}@is.ocha.ac.jp

Abstract There are various music in the world, classical music, jazz, and popular music. Target of our study is mining of relativity among popular music and their attributes, such as composer, raking of hit charts, date of release, and among features of listener's favorite music, focusing on the popular music. For these achievements, we perform clustering focusing on the chord progression that based on music and invariance is high. In the previous work, we proposed similarity measure of chord progression of music data and we classified several Japanese pop music data. In this paper, we reports two experiments using the similarity measure. First, we visualize the relation clustering results to analyze the relation between the chord progression and their song's metadata such as title, performer and producer. Second, we improve the similarity measure of chord progression using typical chord progression pattern. In our proposed measure, each block of chord progression is represented as combination of typical chord patterns, a distance of two chord progressions is calculated based on vector space model.

Keyword Clustering, Mining, Music information processing, Visualization

1. はじめに

世の中には様々なジャンルの楽曲が存在し、クラシック、ジャズ、ポップスなどが挙げられる。我々は、普段生活していて、一番身近であるポップスに着目し、楽曲と作曲者、ヒット曲、年代、ユーザの好みの楽曲との関係のマイニングを行う。これらの実現に向け、ポピュラー音楽において、楽曲のベースとなり不変性の高いコード進行に着目しクラスタリングを行う。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ことだけで同一の楽曲でも印象は異なったものとなる。このことより、上記のことが言えるのでコード進行に着目した。先行研究[1]では、コード進行の類似度に関して、音楽理論にて定義されている近親調をもとに定義を行い、近親調を考慮してクラスタリングを行った場合と、考慮せずにクラスタリング

を行った場合についての比較予備実験を行った。本稿では、先行研究にて行った予備実験の結果から、他属性との相関関係について可視化をすることによって考察を行う。また、先行研究ではコード一つ一つの距離を計算し、それらの和をコード進行間の類似度としていたが、本研究では先行研究で定義したコード進行の類似度の定義を拡張し、コードの前後のつながりも考慮した場合のコード進行間の類似度測定法を提案する。

第3章にて、可視化による考察に関して、第4章にて、コード進行間の類似度測定法に関して述べる。

2. 先行研究

2.1. 先行研究の概要

先行研究では、楽曲と作曲者、ヒット曲、年代、ユーザの好みの楽曲との関係のマイニングを行うため、音楽的特徴であるコード進行に着目し、ポピュラー音楽を対象とした楽曲のクラスタリングを行った。楽曲クラスタリングの処理の流れを図1に示す。先行研究では、クラスタリングを行うために、楽曲を4小節ごとに分割し、分割したものをBlockと名づけ、Blockを対象にクラスタリングを行った。Blockは、四分音符を単位とした16コードネーム c_i による配列、 $Block = (c_0, \dots, c_{15})$ として表される。Block $A = (c_{a_0}, \dots, c_{a_{15}})$ 、Block $B = (c_{b_0}, \dots, c_{b_{15}})$ 間の距離は各コードネーム間の距離の和 $\sum dist(c_{a_i}, c_{b_i})$ で表される。先行研究

では、音楽理論にて定義されている近親調をもとにコード間の類似度の定義を行い、Block同士の距離を計算し、k-means法を用いてクラスタリングを行い、近親調を考慮した場合と考慮しない場合の比較予備実験を行った。比較予備実験より、近親調を考慮した方が、音楽的観点から見て類似したもので構成されたクラスタが出来上がるという結果になった。

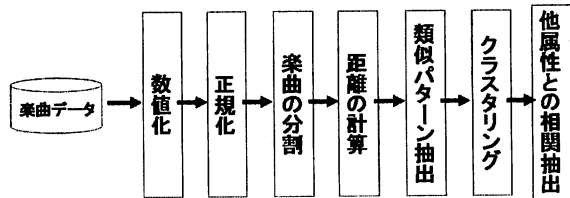


図1 楽曲クラスタリングの流れ

2.2. 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。コードの類似度に関しては、図2の五度圏上にて、もとなるコードから近いものは類似している関係となっている。図中①で表されているものは平行調の関係であり、図

中②で表されているものは属調の関係である。この二つの関係の類似度をもとに、コード間の類似度を求めている。距離計算の式の詳細は省略するが(文献[1]参照)、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 $dist(平行調, C \Rightarrow Am)$ と、AmとCmの属調関係による距離 $dist(属調, Am \Rightarrow Cm)$ の和として計算する。一方、近親調を考慮しなかった場合のCとCmの距離は、「m」が加わった分を距離としている。

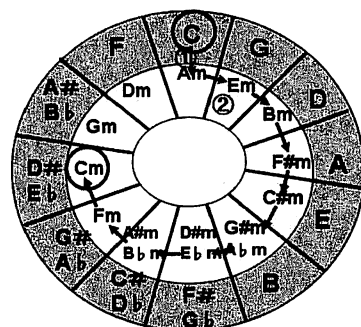


図2 五度圏

3. 可視化による他属性の相関関係抽出

本章では、先行研究にて行った近親調を考慮してクラスタリングを行った際の予備実験の結果を可視化することにより、他属性との相関関係についての考察を行う。コード進行の類似度や楽曲間の類似関係といった、人間の主観や感性により評価が異なる可能性のあるものに対して、人間の主観を排除して定量的に評価することは必ずしもいい評価方法であるとは言えない。そこで我々は、本当に似ているか否かは可視化結果を見た人の判断に委ねることが可能であるという点から可視化は有効であると考え、可視化による評価を行った。

3.1. 平安京ビュー

可視化には、伊藤らによって開発された平安京ビュー[2]を利用する。平安京ビューとは、階層型データを構成する葉ノードをアイコンで表示し、枝ノードを入れ子状の長方形の枠で表現するというものであり、階層型データを可視化するための手法である。平安京ビューは階層構造の葉ノードを塗りつぶされたアイコンで、枝ノードを長方形の枠で表現し、長方形の入れ子構造で階層構造を表現する。平安京ビューは、階層構造をもつウェブサイトのアクセス傾向や計算機ネットワークの不正侵入検知情報などの可視化、原子カシステムの計測情報や薬物実験情報の可視化など、多くの分野に適用が試みられている汎用的な可視化手法である。

図3に、平安京ビューを用いてクラスタリング結果

を可視化したものを示す。本研究では、各クラスタを長方形の枠で表現し、楽曲ごとにアイコンを色分けすることによって可視化を行った。本稿では、分かりやすくするため色分けではなく楽曲ごとに異なるテクスチャで表示している。第 3.2 節にて可視化によるクラスタリング結果の分析について述べる。

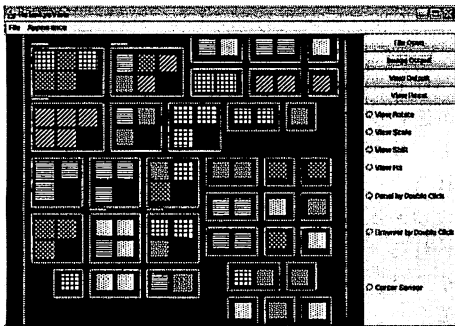


図 3 平安京ビューによるクラスタリング結果の可視化

3.2. クラスタリング結果の分析

3.2.1. 同一楽曲のクラスタ

可視化結果より、同色のアイコンのみで構成されたクラスタは同一楽曲のみで構成されたクラスタであることが分かる(図 4)。合計 36 個のクラスタのうち 13 個のクラスタは同一楽曲のみの Block で構成されていることが見て分かる。同一楽曲の判断については、一目見ただけで分かり、可視化は大変効果的であることが分かる。

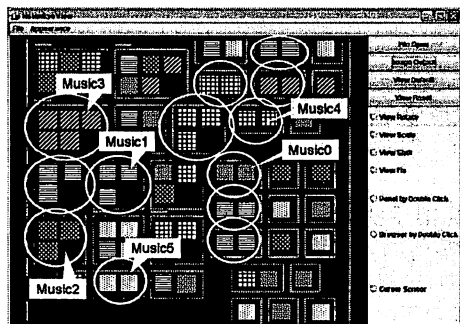


図 4 同一楽曲の Block で構成されたクラスタ

3.2.2. 同一作曲者のクラスタ

一見、異なる色のアイコンで構成されたクラスタではあるが、アイコンの詳細を見てみると、楽曲は異なるが作曲者が同じであるアイコンで構成されているクラスタが見られた(図 5)。さらに、これらのクラスタを構成しているコード進行データを見て分析を行った。図 6 に Cluster31 のクラスタリング結果を示す。図より、

Cluster31 は、ほぼ同一のコード進行で構成されたクラスタとなっており、どの Block も楽曲のサビの部分で使われていることが分かった。このことから、Cluster31 を構成している Block の楽曲の作曲者は、サビに同じようなコード進行を使う傾向があるであろうと考えられる。このコード進行は Cluster31 を構成している楽曲の作曲家特有のコード進行として知られているコード進行であった。図 7 は Cluster30 のクラスタリング結果である。Cluster30 については、B メロの部分で使われている Block で構成されていた。

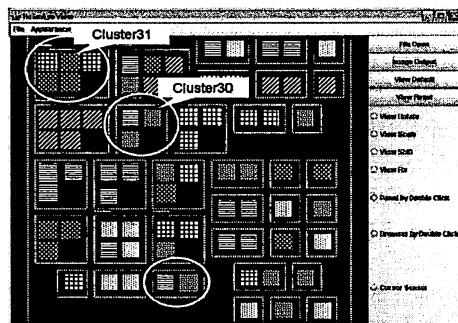


図 5 同一作曲者の Block で構成されたクラスタ

Music No	Block No	コード進行															
4	0	Am	Am	Am	Am	Em7	Em7	Em7	Em7	F	F	G	G	C	C	C	C
2	0	Am	Am	Am	Am	Am7	Am7	Am7	Am7	F	F	G	G	C	C	C	C
4	9	Am	Am	Am	Am	Em	Em	Em	Em	F	F	G	G	C	C	C	C
2	9	Am	Am	Am	Am	Am7	Am7	Am7	Am7	F	F	G	G	C	C	C	C
2	7	Am	Am	Am	Am	Am7	Am7	Am7	Am7	F	F	G	G	C	C	C	C

図 6 クラスタリング結果 (Cluster31)

Music No	Block No	コード進行															
1	6	Ea	Ea	Ea	Am	Am	Am	Am	D7	D7	D7	D7	D7	D7	D7	D7	D7
0	4	Ea	Ea	Am	Am	G7	G7	C7	C7	F	F	C	C	D7	D7	D7	D7
0	12	Aa	Aa	Aa7	Aa7	Aa7	Aa7	Aa7	Aa7	F	F	F	F	D7	D7	D7	D7

図 7 クラスタリング結果 (Cluster30)

3.2.3. 一つの楽曲の分散の仕方

一つの楽曲がどのように分散しているか分析を行った。実験に使った楽曲の内 Music3 の Block は、4 つのクラスタに分散していた(図 8)。それぞれのクラスタでのその楽曲の Block について詳細を見たところ、Cluster24 はサビ、Cluster3 は B メロ、Cluster11 は A メロ、Cluster2 は 1 番と 2 番の移り変わりの部分のように、楽曲の構成要素ごとに分散しているという結果になった。このことから、楽曲の構成要素が同じ場合、似たようなコード進行を用いているのではないかと考えられる。

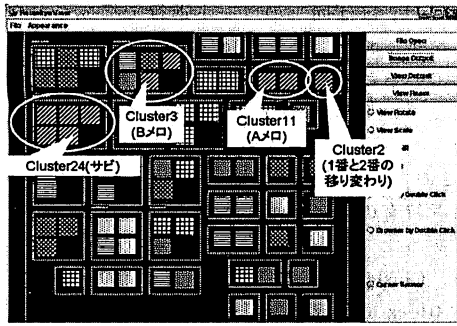


図8 一つの楽曲のBlockの分散

4. 定型コード進行パターンを考慮した類似度測定法の提案

先行研究では、コード進行間の類似度を測る際に、コード一つ一つの距離を計算し、それらの和をコード進行間の類似度としていた。しかし、音楽情報処理研究会での発表[1]やその後の調査により我々は以下の見解を得た。

- (1) コード進行には代表的なコード進行パターンが存在し、楽曲やBlockは、コード進行パターンやそれをアレンジしたものの組合せとして認識される。
- (2) コード進行パターンは教科書に載るような代表的なものから、特定の作曲者のみが使うようなものまで多岐に渡る。
- (3) 先行研究で行ったクラスタリングは、Blockの傾向を見るよりも、コード進行パターンの検出として用いるのに適していると思われる。

見解(1)の例として、図9に示すBlockのコード進行を用いて説明する。図9は、Cメジャーの場合の例であり、このBlockのコード進行は、図中の定型コード進行パターン C_1 、 C_2 、 C_3 により構成される。 C_1 は「T→D→T」、 C_2 は「T→S→D→T」、 C_3 は「T→S→T」という定型コード進行パターンであることを表している。Tはトニックを表しており、トニックとはコード進行において、スタートにもゴールにもなりうる和声の中心となる機能である。Dはドミナントを表し、Tの5度上の和音であり、Tに移行しようとする力が非常に強い。Sはサブドミナントを表し、Tの4度上の和音であり、Dに進むことが多いが、Tに進むことも可能である。表1に、Cメジャーの場合のT、S、Dにどのようなコードが使われるのかを示す。ここで、T、S、Dが1つのコードではなく、数種類のコードの選択肢があることに注意されたい。表中の□で囲まれているコードはベースとなるコードであり、「T→D→T」の場合、代表的なコード進行は「C→G→C」である。しかし、3つ目のCに、同じくTの関係であるAmを用

いて、「C→G→Am」というようにアレンジされて使われる場合もある。よって、同じ「T→D→T」でも何通りかのコード進行が存在する。その他の定型コード進行パターンに関しても同じことが言える。

機能が同じコード同士がどのような関係にあるのか、図2の五度圈上で判断できるものに関して考察を行う。まず、3和音の場合について、トニックに関して、CとEmは属調平行調、CとAmは平行調、EmとAmは属調の関係にあり、DmとFは平行調にある。つまり、同じ機能とみなされているものは、五度圈上で距離が近い関係になっていることが分かる。よって、コードは異なるが、同じ定型コード進行パターンに分類されるコード進行は先行研究にて定義したコード進行間の類似度においてある閾値以内に含まれるものと考えられる。

本章では、以上に述べたことをもとに、先行研究にて定義した、コード進行の類似度に関しての定義を拡張し、コードの前後のつながりも考慮した場合、つまり、定型コード進行パターンを考慮し、コード進行間の類似度の測定法を提案する。



図9 定型コード進行パターンの使用例

	3和音	4和音
T(トニック)	C, Em, Am	CM7, C6, Em7, Am7
S(サブドミナント)	Dm, F	Dm7, FM7, F6
D(ドミナント)	G, Em-5	G7, Em7-5

表1 Cメジャーの場合に用いられるコードの例

4.1. 提案手法の概要

定型コード進行パターンを考慮したコード進行間の類似度の測定法を提案するにあたり、我々は、Blockや楽曲の類似度について、以下の3段階に分けて処理を行う。

- Step1: Blockをさらに細かく分割し、「コード進行パターン」の集合として表す。
- Step2: 「コード進行パターン」をクラスタリングし、定型コード進行とそのアレンジを発見する。
- Step3: Block中に出現する「コード進行パターン」と「定型コード進行パターン」の頻度情報を用いてBlockの類似度を測る。

次節以降にて、それぞれのStepの詳細を述べる。

4.1.1. Block の分割

まず Step1 の Block の分割について説明する。

先行研究では、四分音符を単位として Block を 16 個のコードで表していたが、今回提案する手法においては、長さは考慮せずに出現コードの順序関係のみを扱う。例えば、図 6 の「MusicNo.4 - BlockNo.0」は、「Am - Em7 - F - G - C」というように表される。また、分割数に関しては、定型コード進行パターンを構成しているコードの出現個数で分割を行う。定型コード進行パターンには、例えば、「T→D→T」、「T→S→D→T」のような 3, 4 連続のものや、「II→V」のように 2 連続の組でパターン化されるものが多いので、コード進行列を、2 分割、3 分割、4 分割にする。例えば、A = (Am, Em7, F, G, C) という Block の場合、分割を行うと以下ようになる。

$$A = \{(Am, Em7), (Em7, F), (F, G), (G, C), (Am, Em7, F), (Em7, F, G), (F, G, C), (Am, Em7, F, G), (Em7, F, G, C)\}$$

以下説明のため、上記の通り分割したものを以下のよう表す。

$$A = \{a_0^2, a_1^2, a_2^2, a_3^2, a_0^3, a_1^3, a_2^3, a_0^4, a_1^4\}$$

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_1^2 の「コード進行パターン」の長さは 2 となる。

4.1.2. コード進行パターンのクラスタリング

分割された「コード進行パターン」の長さ n ごとに、全ての Block の「コード進行パターン」を集めてクラスタリングを行う。例えば、第 4.1.1 節にて挙げた Block A の他に、

$$B = \{b_0^2, \dots, b_i^2, b_0^3, \dots, b_j^3, b_0^4, \dots, b_k^4\}$$

$$C = \{c_0^2, \dots, c_r^2, c_0^3, \dots, c_s^3, c_0^4, \dots, c_t^4\}$$

という Block があった場合、Block B と Block C の「コード進行パターン」も集めてクラスタリングを行う。図 10 に Block A, B, C を対象に 2 次元、3 次元、4 次元でそれぞれクラスタリングを行った結果の例を示す。次元とは、「コード進行パターン」の長さのことであり、2 次元の場合は「コード進行パターン」の長さ 2 であるものにおいてクラスタリングを行った結果である。ここで、それぞれのクラスタの中心である P_i は定型コード進行パターンであり、その周辺は定型コード進行パターンをアレンジしたものを見なすこととする。

4.1.3. ブロックの類似度の計算

「コード進行パターン」の頻度情報と、Step2 で行ったクラスタリング結果をもとに、それぞれの Block の特徴ベクトルを求める。オリジナルのコード進行パターンについては、Block 中に出現する回数の特徴ベクトルとする。例えば、Block A において、「コード進行パターン」 $a_0^2, a_1^2, \dots, a_1^4$ が一回ずつ出現し、その他のオリジナルのコード進行パターンは出現しない場合、図に示すように、 a_0^2 から a_1^4 の部分は 1、その他の部分は 0 となる。ここで、Block 内の各コード進行パターンが Step2 で求めたクラスタに含まれているかを調べ、あればその代表パターンに対して重みを加算する。その重みの値は、コード進行パターンの重みを 1 としたとき、 $1/(\text{代表パターンとコード進行パターンの距離})$ とする。

例えば Block A の a_0^2 は図 10 のクラスタ 2 に含まれ

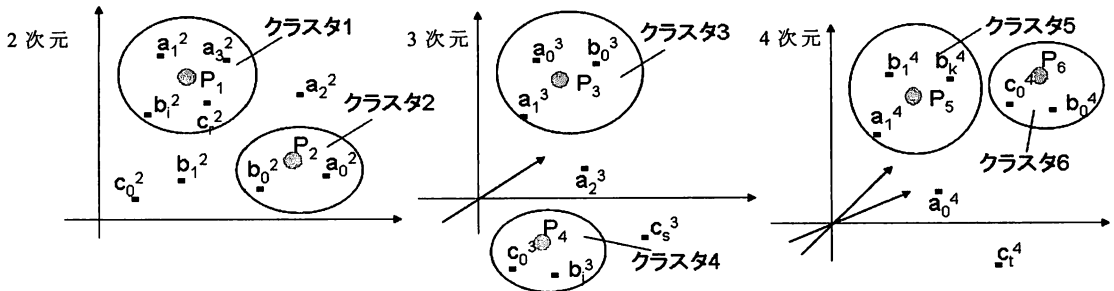


図 10 クラスタリングの例

	a_0^2	...	a_1^4	b_0^2	...	b_k^4	c_0^2	...	c_t^2	P_1	P_2	...
A	1	...	1	0	...	0	0	...	0	$\frac{1}{d(a_0^2, P_1)} + \frac{1}{d(a_1^4, P_1)}$	$\frac{1}{d(a_0^2, P_2)}$...
B	0	...	0	1	...	1	0	...	0	$\frac{1}{d(b^2, P_1)}$	$\frac{1}{d(b_k^4, P_2)}$...
C	0	...	0	0	...	0	1	...	1	$\frac{1}{d(c^2, P_1)}$	0	...

図 11 特徴ベクトルの例

ておりその代表パターンは P_2 であるため、 P_2 に $1/\text{dist}(P_2, a_0^2)$ が加算される。Block A において、クラスタ 1 の定型コード進行パターンである P_1 は a_1^2, a_3^2 を含んでいるため、Block A における P_1 の値は $1/d(P_1, a_1^2) + 1/d(P_1, a_3^2)$ となる。第 4.1.2 節に用いたクラスタリング例の特徴ベクトルを、以上に述べた方法に従って求めたものを図 11 に示す。特徴ベクトルを用いて、それぞれにどのような重みを加えるかにより、オリジナルのコード進行パターンと Block の関係や、定型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関係を知ることが可能である。例えば、図 11 の特徴ベクトルの例では、 a_0^2 から c_1^4 の部分の重みを大きくすれば、オリジナルのコード進行パターンがどれ位 Block 中に出現するかといった部分を特徴として重要視し、 P_0 の部分の重みを大きくすれば、どのようにアレンジをしているかといった部分を特徴として重要視することになる。

4.2. 提案手法に関する考察

現在のところ、本手法はまだ十分な検証がされておらず、実験もまだ行っていないため、本手法の十分な検討が必要となる。現段階で本手法に関して、まだ取り入れているが重要と思われる項目について本節で示しておく。

(1) 出現頻度のカウント数の問題

定型コード進行パターンは複数存在するが、一つの Block の中には 1 種類の定型コード進行パターンのみであったり、2 種類の定型コード進行パターンが一回ずつ出現したりという場合が考えられるため、ほとんどの特徴ベクトルが、同じようなものとなり、それぞれの Block を識別できるような、特徴ベクトルを得られない可能性がある。

(2) 定型コード進行パターン間の前後関係の欠落

定型コード進行パターンの出現回数をカウントするため、一つのブロックに複数の定型コード進行パターンが存在する場合、出現する位置を考慮していないため、定型コード進行パターンの出現位置が前後していたとしても、同じ Block であるとみなされる。

この二つの問題点に関しては今後検討していく。またクラスタリング手法についても今後検討を行う。

4.3. 関連研究

同一 Block に含まれる定型コード進行パターン間の類似度を考慮するための手法として、情報検索の観点からコードとコード進行の関係性を文字と単語の関係と考えた場合、以下の論文が関連研究として挙げられる。

谷口[3]は情報検索をする際、検索に語の同意語や関連語により索引づけされた文献が検索されないという

問題を挙げ、索引語間の関連性を考慮した情報検索モデルの提案を行っている。実際には、索引語間の関連性が関連度として、数値として与えられていることを前提にして、質問に対する各文献の類似度測定にその関連度を反映させることが出来るよう、既存の検索モデルの拡張を試みている。

定型コード進行パターンを「索引語」としてみなすと、上に述べた手法に基づいて、一つの Block に含まれるコード進行間の類似度を測ることが可能となるのではないかと考えられる。つまり、定型コード進行パターンとの関連性を Block 間の類似度測定に反映させ、定型コード進行パターンとの関連性により Block 同士の類似度を測りクラスタリングを行うという手法が考えられる。

5. まとめと今後の課題

本稿では、ポピュラー音楽を対象に、楽曲と他属性の相関関係の抽出を行うため、先行研究にて行った予備実験の結果の可視化することにより、クラスタリング結果の分析を行った。この分析により、クラスタを構成している楽曲の関係、作曲家や楽曲を構成している要素とコード進行との関係を分析することが出来た。また、楽曲や Block はコード進行パターンやそれをアレンジしたものの組合せとして認識されるという見解から、定型コード進行パターンを考慮してクラスタリングを行うための類似度測定法を提案した。

今後は、提案手法を適用する上で問題点として挙げられる、出現頻度のカウント数に関する問題や定型コード進行パターン間の前後関係についての検討、クラスタリング手法についての検討を行う。また、本稿にて提案した手法を用いて、定型コード進行パターンを考慮したクラスタリングを行い、楽曲と他属性との関係のマイニングを行う。

[謝辞]

本研究を進めるにあたりご指導いただいた、静岡文化芸術大学デザイン学部の長嶋洋一教授に深く感謝致します。

6. 参考文献

- [1] 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視調を用いたコード進行類似度の提案”, 音楽情報処理学会研究報告, Vol.2007, No.37, pp.69-76, May.2007.
- [2] 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
- [3] 谷口祥一, “索引語間の関連性を考慮した情報検索モデル”, Library and Information Science, No.28, pp.105-119, 1991