

# 複数ユーザー間での楽曲推薦を実現するミュージックプレイヤー：楽曲類似度の導入と有効性の検証

鈴木 潤一<sup>†1,a)</sup> 北原 鉄朗<sup>†1,b)</sup>

**概要：**音楽を嗜むシチュエーションの一つとして、複数人が同じ場所に集まり同じ楽曲を聴く場合がある。このようなシチュエーションにおいて、個々人のスマートフォン内に所持する楽曲から適した楽曲を自動的に選択し、再生することができれば有用であると考えられる。以前の我々の論文では、スマートフォンに記録されている楽曲の再生履歴とアーティスト間の類似度を用いて、複数ユーザーが好む可能性のあるアーティストの楽曲を選曲したが、アーティストのどの楽曲を再生するかについては、十分に検討しておらずアーティスト単位の推薦に至った。本研究では、以前では用いていない楽曲自身のデータを用いた類似度も導入し、ユーザーの楽曲を好む度合い、期待度の推測についての妥当性の検証を標準的データベースを用いて行い楽曲単位の楽曲の推薦手法を提案した。実験を行った結果、ユーザーが楽曲に対する再生回数が十分に付与されている場合において、ユーザーの楽曲に対する期待度と推測した期待度に弱い相関関係が現れた。

**キーワード：**楽曲, 推薦, グループ, 複数人, スマートフォン

## Music player to recommend songs among multiple users: Introduction of music similarity and verification of effectiveness

JUNICHI SUZUKI<sup>†1,a)</sup> TETSURO KITAHARA<sup>†1,b)</sup>

**Abstract:** When situations where multiple people to gather, they often play background music (BGM) stored in individual devices owned by one or more users. However, because one user typically selects the music from a personal device, music preferences tend to be biased. It is preferred that the musical preferences of all the people be used to determine the background music (BGM). In our previous the method, system recommended music using the playback history of music and the similarity between artists. In this paper, we propose recommendation using similarity between songs along with the similarity between artists. And also, we validate the appropriateness of recommended method using standard database.

**Keywords:** Music, Recommendation, Group, Multiple people, Smartphone

### 1. はじめに

近年、大容量の小型記憶媒体の登場やインターネット上の音楽コンテンツの多様化によってユーザーは場所や時間を問わず、容易に個人で音楽を楽しむことが可能になっ

た [1]。

さらにスマートフォンのような小型記憶媒体の発展により、音楽を個人で鑑賞するだけではなく、パーティーやドライブのような、複数人が同じ場所に集まり同じ音楽を鑑賞するシチュエーションにおいて、個人の所有するスマートフォン（以降、端末と呼称）から音楽を再生する場面も増えている。しかし、そのようなシチュエーションにおいて、ユーザー同士の音楽の嗜好の違いにより、再生する楽曲について気に入らないユーザーが出てしまう場合がある。

<sup>†1</sup> 現在, 日本大学  
Presently with Nihon University

a) junichi@kthrlab.jp

b) kitahara@chs.nihon-u.ac.jp

なぜなら端末に保存されている楽曲は、ユーザーの音楽の嗜好によって異なるからである。本来は、その場にいる全員の嗜好を満たす楽曲の選択を行うべきであるが、その処理を手動で行うのは時間や手間を要するため自動的に行われるのが望ましい。そのため複数人を対象にした選曲・再生システムに関する研究事例はいくつか存在する [2][3][4]

Jamson[2] と Crossen[3] らは、1つのサーバーやPCに統合した楽曲を用いて複数人に対する楽曲推薦を行っている。しかし、一部のユーザーにとっては未知な楽曲を含む場合があり、そのような楽曲の評価をユーザーが直接行うには、楽曲を聴く必要がある。そのような問題に対して、再生された楽曲についてユーザーが何度も評価を行うような手法を用いているが、そのようなシチュエーションにおいてユーザーが自発的に何度も評価をする行為は、煩わしいことや手間を要する場合がある。また Joseph[4] らは、複数人が集まるフィットネスジムにおいてユーザーの嗜好を事前に調査し、同一の部屋にいるユーザーの好みに合わせて、音楽ラジオ配信サービスの楽曲を変更する手法を用いている。しかし、この手法では事前に嗜好の調査を行うことが必要であり、調査が困難なシチュエーションにおいては現実的ではない。以上から、上記で述べた3つの複数人に対する楽曲推薦の研究において、ユーザーの嗜好の不足は課題の一つである。

このような課題について我々は、複数人が同一の空間に集まり各ユーザーが所有する端末から楽曲を再生し、一緒に同じ楽曲を聴く環境に着目し、端末に記録されている情報から、ユーザーの嗜好の推測を行ってきた [5]。我々は、ユーザーの端末に記録されているアーティストを、ユーザーの嗜好するアーティストとみなし、複数人の嗜好するアーティストに類似しているアーティストが多く含まれる楽曲リストの生成を行った。その結果、ランダムに選択した楽曲リストと比べ、複数人の嗜好に合う楽曲を多く含ませる事ができた。しかしアーティスト単位の選曲でありアーティストについて、どの楽曲を選曲するかについての検討は行っていない。

本研究では、以前に我々が行ったアーティスト情報を用いた手法 [5] に加えて、楽曲のデータの情報を用いる。さらに、選曲の妥当性の検証を標準的なデータベースを用いて行う。我々の提案するシステムでは、各ユーザーが所有する端末同士を Bluetooth で接続し、端末内に記録されている再生履歴や楽曲情報の収集を行う。次に収集した端末の情報から、全ての端末に保存されている楽曲に対して各ユーザーの気に入る度合い（以下、期待度）を推測し、推測した全てのユーザーの期待度を基に楽曲リストの生成を行う。生成した楽曲リストに沿って、各端末に保存されている楽曲を次々と再生を行う。全員の嗜好を満たした楽曲を推薦・再生ができるシステムと推薦手法の実現に必要な、ユーザーの端末に記録されている情報から、楽曲に対して

の期待度の推測の検証と、それを実現するためのアプリケーションの制作を行う。

## 2. 推薦手法

我々の想定するシチュエーションにおいて、楽曲の推薦手法として考えられる手法の一つは、一人一人の楽曲に対する嗜好評価を統合し、全員の嗜好評価が高い楽曲を選出する手法である。本研究では、ユーザーが楽曲を気に入る度合いを期待度と定義する。我々は、以下の2つの仮説の基、推薦手法の提案をする。

### 仮説 1

ユーザーの期待度は、端末に記録されている再生履歴が多い楽曲ほど高い。

### 仮説 2

ユーザーの楽曲に対する期待度は、類似している楽曲の期待度に類似する。

本研究では、仮説 1 の基、ユーザーの所持楽曲の再生履歴が多い楽曲ほど期待度が高くなるよう決定する。次に仮説 2 の基、ユーザーの楽曲に対する期待度は、類似した楽曲の期待度に近似すると考えた。そのためユーザーの所有していない楽曲の期待度を、ユーザーの所有する楽曲との類似度に応じて推測する。推測対象の楽曲に類似している所有曲の期待度の値が多く反映されるものである。

上記の提案システムは、ユーザーが日常的に自身の端末を用いて楽曲を聴いており、楽曲に対してユーザーの再生履歴が十分に付与されていることを前提とする。そのため我々の提案手法では、ユーザーが再生した楽曲の再生履歴を保存する機能を実装する。機能の詳細については 3 章で述べる。

### 2.1 所持楽曲に対する期待度

ユーザー  $n$  人を  $U_{\text{all}} = \{u_1, u_2, \dots, u_n\}$  とする。仮説 1 に基づいて、ユーザーの所有楽曲は再生履歴が多い楽曲ほど期待度が高くなるように決定する。本研究では再生履歴に出現した楽曲の回数を、その楽曲の再生回数とする。再生履歴に出現しない再生回数がゼロである楽曲は、所持楽曲とみなさない。ユーザー  $u_i$  が  $x$  曲所持しているとき、 $u_i$  の所持楽曲を  $M_{(u_i)} = \{m_{(u_i,1)}, m_{(u_i,2)}, \dots, m_{(u_i,x)}\}$  とし楽曲  $m_{(u_i,x)}$  の再生回数を  $F(m_{(u_i,x)}) > 0$  とする。 $m_{(u_i,x)} \in M_{(u_i)}$  を満たすとき、 $m_{(u_i,x)}$  に対するユーザー  $u_i$  の期待度  $W(m_{(u_i,x)})$  は以下のように計算する。

$$W(m_{(u_i,x)}) = 1 - \frac{1}{F(m_{(u_i,x)})^{0.35}} \quad (1)$$

式 (1) よりユーザーの所持している楽曲に対しての期待度を決定する。

## 2.2 楽曲間の類似度の算出

本研究では楽曲間の類似度を求めるのに、楽曲特徴量を用いた楽曲類似度と、楽曲のアーティスト情報を用いたアーティスト類似度を使用する。

### 2.2.1 楽曲類似度

E. Pampalk ら [6] は楽曲特徴量を楽曲データの中心の 30 秒間を対象にメル周波数ケプストラム係数 (Mel-Frequency Cepstrum Coefficient: MFCC) の抽出を行い、ベクトル集合である MFCC をクラスタリングする。そしてクラスタ集合間の距離を EMD (Earth Mover's Distance) を用いて算出し、得られた値を楽曲間の距離としている。本研究では楽曲間の距離の算出に E. Pampalk らの手法を用いる。なお MFCC の抽出は、サンプリング周波数 16kHz、シフト幅 160 サンプル、1 フレームの 20 次元 MFCC ベクトル (19 次元+パワー) として行う。EMD を用いて得られた楽曲  $\alpha, \beta$  間の距離を  $D(\alpha, \beta)$  とするとき、 $\alpha, \beta$  間の楽曲類似度を  $Similarity_{MFCC}(\alpha, \beta)$  とし、以下のように計算する。

$$Similarity_{MFCC}(\alpha, \beta) = \frac{1}{1 + D(\alpha, \beta)} \quad (2)$$

なお、本研究で扱う音響的特徴の MFCC は、抽出に時間を要するため 4 章で述べる楽曲特徴量収集機能において楽曲類似度の算出を行う前に、既に取得している事とする。

### 2.2.2 アーティスト類似度

Last.fm [7] の WebAPI を用いて、指定したアーティストに類似したアーティスト上位 100 位までの情報を取得する。類似アーティストには指定したアーティストからの類似度が 0 ~ 1 の値で付与されている。楽曲  $\alpha, \beta$  のアーティストをそれぞれ  $A(\alpha), A(\beta)$  とし、 $A(\alpha)$  の類似アーティスト 100 件を  $S_{A(\alpha)}$  とする。また  $A(\beta) \in S_{A(\alpha)}$  を満たす時の  $A(\alpha), A(\beta)$  の類似度を  $L(A(\alpha), A(\beta))$  とする。 $\alpha, \beta$  のアーティスト類似度  $Similarity_{Artist}(A(\alpha), A(\beta))$  を以下の様に定義する。

$$Similarity_{Artist}(A(\alpha), A(\beta)) = \begin{cases} 1.0 & A(\alpha) = A(\beta) \\ L(A(\alpha), A(\beta)) & A(\beta) \in S_{A(\alpha)} \\ 0.01 & (else) \end{cases} \quad (3)$$

なお、アーティスト類似度は、取得に時間を要するため 4 章で述べる類似アーティスト収集機能において、既に取得している事とする。

### 2.2.3 楽曲間類似度の統合

楽曲  $\alpha, \beta$  の楽曲間の類似度  $Similarity(A(\alpha), A(\beta))$  の算出を、式 (2) で得られた楽曲類似度と式 (3) で得られたアーティスト類似度を次の方法のように掛け合わせる。

$$Similarity(\alpha, \beta) = Similarity_{Artist}(A(\alpha), A(\beta)) \times Similarity_{MFCC}(A(\alpha), A(\beta)) \quad (4)$$

## 2.3 未所持楽曲に対する期待度

仮説 2 に基づいて、ユーザーの所有楽曲の期待度は、類似する楽曲の期待度に類似すると考える。ユーザー  $u_i$  が  $x$  曲所持しているとき、 $u_i$  の所持楽曲を  $M_{(u_i)} = \{m_{(u_i,1)}, m_{(u_i,2)}, \dots, m_{(u_i,x)}\}$ 、ユーザー  $u_j$  が  $y$  曲所持しているとき、 $u_j$  の所持楽曲を  $M_{(u_j)} = \{m_{(u_j,1)}, m_{(u_j,2)}, \dots, m_{(u_j,y)}\}$  とする。ユーザー  $u_i$  の楽曲  $m_{(u_i,y)}$  に対する嗜好評価  $R(u_i, m_{(u_i,y)})$  は、式 (1)、(4)、を用いて以下の式で行う。

$m_{(u_i,y)} \notin M_{(u_i)}$  を満たすとき、

$$R(u_i, m_{(u_i,y)}) = \frac{\sum_{k=1}^x W(m_{(u_i,k)}) \times Similarity(m_{(u_i,k)}, m_{(u_i,y)})}{\sum_{l=1}^x W(m_{(u_i,l)})} \quad (5)$$

$m_{(u_i,y)} \in M_{(u_i)}$  を満たすとき、

$$R(u_i, m_{(u_i,y)}) = W(m_{(u_i,y)}) \quad (6)$$

以上の式により、ユーザーの所持する楽曲から、対象となる楽曲の期待度の推測を行う。

## 2.4 楽曲リストの生成

式 (5)、(6) より、複数のユーザーが所持する楽曲について、全員の期待度が既知な状態とし、全員の所持する楽曲を統合した楽曲集合を  $M_{All} = \{m_1, m_2, \dots, m_t\}$  とするとき、楽曲  $m_t$  に対するユーザー全員の期待度の統合を以下の式で行う。

$$R_{All}(m_t) = \sum_{l=1}^n R(u_l, m_t) \quad (7)$$

式 (7) より  $M_{All}$  に対する複数人の統合した期待度が、付与されている状態とし、複数人の期待度が大きい順に  $M_{All}$  を並び替えたものを楽曲リストとする。生成した楽曲リストの再生方法については、3 章のアプリケーションの構成において述べる。

## 3. アプリケーションの概要

本研究ではユーザーが使用する端末をスマートフォンと想定している為、開発が容易である Android OS を対象にアプリケーションの制作を行った。本アプリケーションは、大きく分けて次の 3 つの機能を実装している。

- (1) 音楽プレイヤー機能
- (2) 楽曲特徴量・類似アーティスト収集機能
- (3) グループ楽曲推薦機能

### 3.1 音楽プレイヤー機能

音楽プレイヤー機能には、楽曲の再生する機能に加え、ユーザーが日常的に音楽を聴いた再生履歴を端末に記録する機能を実装している。

### 3.2 楽曲特徴量・類似アーティスト収集機能

3章で述べた音響的特徴である MFCC の取得は、時間を要するため日常的に端末内に保存されている楽曲の MFCC の抽出を行い端末内に保存する。また Last.fm Web API を用いたユーザーの所持するアーティストの類似アーティストの情報の取得も、時間を要するため日常的に収集を行い端末内に保存する。

### 3.3 グループ楽曲推薦機能

ユーザー  $n$  人が所持する端末を  $D_{\text{all}} = \{D_1, D_2, \dots, D_n\}$  と称し  $D_1$  を親機とする。 $D_1$  はグループ楽曲推薦機能を使用すると端末内に保存されているペアリング済みである BluetoothID を参照し、Bluetooth 機能を用いて  $D_1$  周辺で本アプリケーションを起動している端末の探索を開始する。 $D_1$  は発見した子機と接続を開始する。この一連の処理をグループ化と呼び、後述の処理においてデータ受信する際はこのグループ化が完了している状態が必要である。グループ化を行っているアプリケーションの画面を図 4.5 に示す。 $D_1$  は  $D_{\text{all}}$  の端末内に記録、保存されている楽曲名、楽曲の再生回数、楽曲の MFCC データ、アーティスト名、類似アーティスト情報を集約する。 $D_1$  は集約した情報を基に、3章で述べた推薦システムを用いて楽曲リストの生成を行う。

複数ユーザー所有する各端末に別々に保存されている楽曲を再生するために、親機は楽曲リストの順序に従って楽曲を保存している端末へ再生指示を出す。再生指示された端末は、Bluetooth スピーカーへ自動的に接続を行い再生を行う。

## 4. 実験

### 4.1 仮説 1 の検証

#### 4.1.1 収集したユーザーデータセット

我々は、SNS(ソーシャルネットワーキングサイト)である Last.fm からユーザーのプロファイルを収集した。ユーザーは Last.fm 上において、好きな楽曲に対してポジティブな評価である「Loved」という評価を行うことができる。楽曲に対して評価経験のある 34,671 ユーザーを対象にプロファイルをランダムに収集を行った後、仮説 1 の検証において、再生回数と期待度の関係性を調べるために一定の評価数のある、楽曲の評価数が 20 以上 5000 以下の 10,201 ユーザーに限定した。次に評価経験のある 10,201 ユーザーの楽曲の再生履歴を収集した。

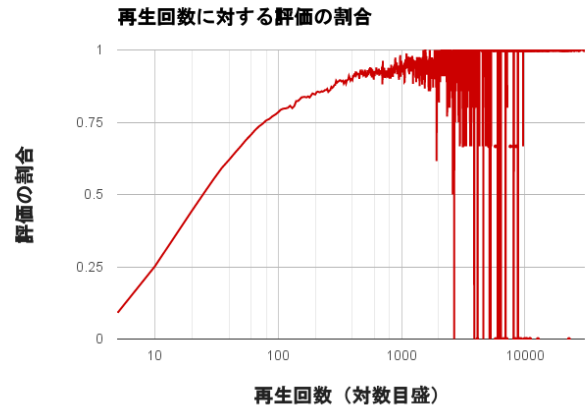


図 1 再生回数に対する評価の割合

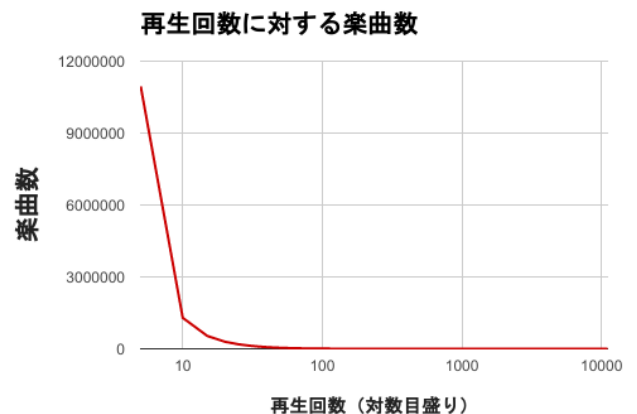


図 2 再生回数に楽曲数

### 4.1.2 検証結果

10,201 ユーザーの楽曲についての再生回数と評価の対データ、計 98,504,128 個に対して再生回数と期待度に関する調査を行う。楽曲を 5 再生回数ずつに分類し、再生回数に対する評価楽曲を含む割合を調べる。10,201 ユーザーの中で、一つの楽曲に対して最も高かった再生回数は 30,145 回であった。そのため 0 ~ 30,145 再生回数に対する楽曲に評価が付いている割合を図 1 に示す。図 1 から再生回数が増加するにつれて、楽曲に評価が付いている割合も増加し、ユーザーは、100 再生回数を越えると楽曲に評価を付けている割合が 7 割に達し、また 1000 再生回数を越えると約 9 割の楽曲に評価を付けている事が分かる。本研究では、再生回数に対するユーザーが楽曲に評価を付ける割合を、ユーザーが楽曲を好む度合いと解釈し、楽曲を好む度合いは再生回数に対して単調増加で、100%に近似していくと図 1 から十分に言える。再生回数が 1000 を超えた辺から評価の割合に差が大きく生じるのは、図 2 から再生回数に対する楽曲数が減り、評価の割合に差が出てしまったためである。以上から、楽曲の再生回数と評価の割合から仮説 1 で述べた再生回数の楽曲を好む度合い、期待度の関係性

は、正しいと結論付ける。

## 4.2 The Last.fm Dataset を用いた実験

### 4.2.1 The Last.fm Dataset の詳細

The Last.fm Dataset[8] には、last.fm に登録している 500 ユーザーの再生した楽曲名、アーティスト名を含む再生履歴が記録されている。本研究では、ユーザーの再生履歴に出現した楽曲を、そのユーザーの所持楽曲とみなし、再生履歴に出現した回数をその楽曲の再生回数として用いる。提案システムで用いる楽曲特徴量の抽出に必要である楽曲データについて、The Last.fm Dataset には楽曲データは含まれていない為、7digital[9] の API を利用して The Last.fm Dataset のユーザーの再生履歴にある楽曲の取得を行った。7digital API を利用することで 30 秒程の長さの試聴用楽曲データを手に入れる。本実験ではこの試聴用データをもとに、楽曲特徴抽出および The Last.fm Dataset を用いた実験を行う。

### 4.2.2 実験手法

The Last.fm Dataset のデータを基に、ユーザーの所持楽曲が 300 曲以上被っている 3 ユーザー 1 組のグループを 102 グループ作成した。本実験では、グループ内のユーザーの所持楽曲が被っている楽曲をグループ所持楽曲と呼称する。グループ所持楽曲に対する 3 ユーザーの再生回数を、期待度へ正規化を行い、後に推測するユーザー期待度と比較するための、正解になる期待度とする（以降、正解データ）。図 1 よりユーザーの楽曲に対する評価率は、再生回数に対して単調増加で 100% に近似していく為、近似する式を式 (1)、(8) の 2 つ用意した。

$$W(m(u_i, x)) = \left( \frac{1}{1 + e^{0.037 \times F(m(u_i, x))}} - 0.5 \right) \times 2 \quad (8)$$

図 1 の楽曲の対する評価の割合のグラフと、それに近似させた式 (1)、式 (8) を図 3 に示す。次にグループ楽曲をランダムに 3 等分し、それぞれをグループ内の 3 ユーザーの所持楽曲とみなす。所持楽曲とみなさなかつたグループ所持楽曲については、ユーザーの未所持楽曲とする。提案システムを用いて未所持楽曲に対する期待度の推測を行い、正解データと推測した期待度の相関関係を調べる。

### 4.3 実験結果

102 グループに含まれる 306 ユーザーを対象に式 (1)、式 (8) を用いて所持楽曲に対する期待度の正規化を行い、提案する推薦手法を用いて推測した期待度と、正解データの相関係数の平均を表 1 に示す。式 (1) を用いて推測した期待度と正解データの相関係数の平均は 0.348 と強い相関があるという結果には至らず、弱い相関を示す結果になった。式 (8) を用いた場合の相関係数は、0.258 であった。式 (1) を用いたときの、ユーザー毎の相関係数の平均のヒストグ

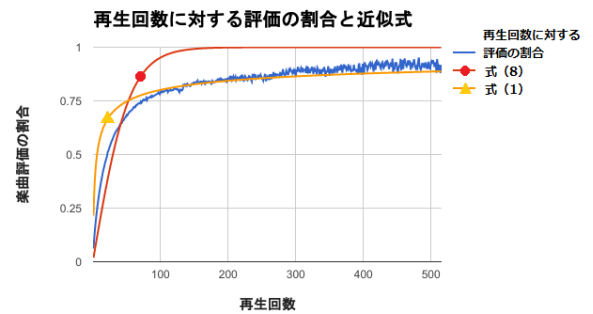


図 3 再生回数に評価の割合と式 (1)(8)

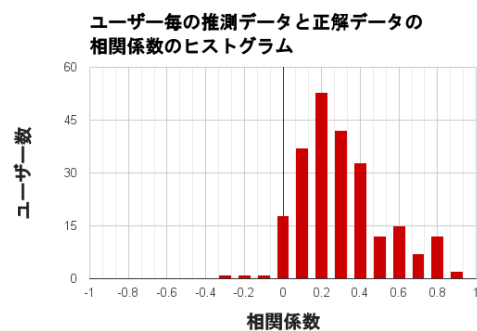


図 4 推測した嗜好評価と正解嗜好評価のユーザー毎の相関係数のヒストグラム

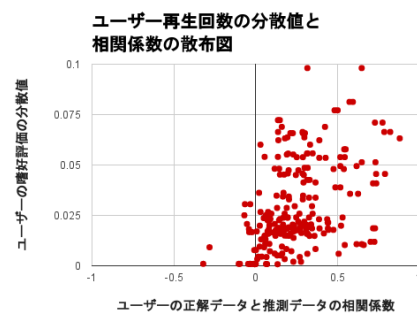


図 5 ユーザーの期待度の分散と、ユーザー毎の嗜好評価と推測データの相関に関する散布図

ラムを図3に示す。図4から全体的に正の相関関係の傾向があると見て取れる。

ユーザーの正解データと推測した期待度の相関係数値と、正解データの分散値に関する散布図を図5に示す。図4からユーザーの再生回数を正規化した正解データの分散値が高いほど相関係数値が高い傾向にある。これは、ユーザーの所持楽曲の再生回数にバラツキの大きいほど、ユーザーの楽曲に対する期待度がより明確になり期待度の推測において、より良い結果が出たと考える。しかし、今回の実験においてグループ数の確保の為、グループ所持楽曲をThe Last.fm Dataset から共通の所持楽曲が300以上被っているユーザーの共通の楽曲とした。そのためグループ所持楽曲にはユーザー毎の再生回数の考慮をしておらず、グループ所持楽曲に限定したときのユーザーの所持楽曲に対する再生回数が、本来のユーザーが所持する楽曲の再生回数に比べて分散が小さくなってしまった為と考える。これから、グループ所持楽曲の再生回数が従来のユーザーが所持する楽曲に対する再生回数と比べて、小さいことが分かる。

次に、3章で述べた仮説2について本章の実験から、提案システムにおいてユーザーの楽曲に対する再生回数が十分に付与されている場合、推薦精度が向上する傾向があることが分かり、そのような状況に置いてはユーザーの所持している楽曲と類似している楽曲はその楽曲の評価に類似する傾向にあると言える。今後のより詳しい分析を行うために、ユーザーの本来の所持している楽曲から、期待度の推測を行うべきと言える。また、所持楽曲に対する期待度の導出式について、単調増加で1に収束する近似式(1)、(8)を用意したが、式の妥当性についても検討すべきである。

表1 The Last.fm Dataset:推測した嗜好評価と正解データの相関係数

グループ数	ユーザー数	推測した期待度と正解データと相関係数平均	
		(式1)	(式8)
102	306	0.351	0.258

## 5. 結 論

本研究では、複数人が同じ場所へ集まり同じ楽曲を聴くシチュエーションで、複数人に楽曲を推薦するグループ楽曲推薦を行う際の、嗜好評価の不足の改善を試みた。複数のユーザーの所有する端末に記録されている情報から、楽曲に対する嗜好評価の推測を行い、複数人の所有する端末をBluetoothを用いて相互に接続を行い、楽曲リストの生成から同一のスピーカーへ自動再生をするスマートフォン用アプリケーションを開発した。

実験では、ユーザーの楽曲に対する再生回数が、十分に付与されている場合において、推測した嗜好評価とユーザーの嗜好評価に正の相関関係が現れることがわかり、本

研究で提案した端末内に記録されている楽曲情報と、再生回数のデータを用いた、グループ楽曲推薦が一定の有効性があることが分かった。

本稿で述べた実験ではユーザーの使用する楽曲に対して、共通の楽曲が一定以上あるグループに絞った。そのため、推測した期待度と正解データに弱い相関が現れる結果に成ったが、本来のユーザーの所有する楽曲に対する再生回数のデータが多く省かれてしまう結果に成った。今後の課題として、ユーザーの本来の所持している楽曲から決定した期待度を用いて、対象の楽曲の期待度を推測できるか検証を行い、期待度の推測について精度の向上を目指す。

## 参考文献

- [1] 梶 克彦, 平田 圭二, 長尾 確: “コミュニケーションメディアとしてのプレイリストを目指して”, FIT, 2005.
- [2] A. Jameson. “ More than the Sum of its Members: Challenges for Group Recommender System, ” Proc AVI '04, ACM, 2004, New York, NY, USA, pp.48-54.
- [3] A. Crossen, J. Budzik, and K. Hammond. “ Flytrap: intelligent group music recommendation ”, Proc IUI '02., 184-185, 2002.
- [4] F. Joseph, D. Theodore. “ MUSICFX: An Arbiter of Group Preference for Computer Supported Collaborative Workouts ”, Proc CSCW '98., ACM, New York, NY, USA, 363-372, 1998.
- [5] J. Suzuki, T. Kitahara. “ A Bluetooth-Networked Music Player for Playing Musical Pieces Stored in Separate Devices ”, Proc KSE, Vietnam 2016.
- [6] E. Pampalk, “ Computational Models of Music Similarity and their Application ” in Music Information Retrieval, Doctoral Thesis, Vienna University of Technology, Austria, March 2006.
- [7] Last.fm(accessed 2017. 1), <http://www.lastfm.jp/api>
- [8] Last.fm datasets(accessed 2017. 1), <http://ocelma.net/>
- [9] 7digital(accessed 2017. 1), <https://www.7digital.com/>