

平成 28 年度 修士論文

複数ユーザーが同一楽曲を聴くための
選曲・再生システム

指導教員 北原鉄朗准教授

日本大学大学院総合基礎科学研究科

6115M17 鈴木潤一

2017 年 2 月 提出

概 要

音楽を鑑賞するシチュエーションの一つとして、複数人が同じ楽曲を聴く場面が存在する。そのような場面の選曲手法の一つとして、各自の楽曲に対する気に入る度合いを統合し全員の音楽の嗜好を考慮する方法がある。しかしユーザーの楽曲に対する嗜好のデータは、シチュエーションによって収集が困難である場合がある。そのため楽曲に対する嗜好データの不足は、選曲を行う際において問題の一つであり、嗜好データの推測について様々な手法が提案されてきた。

本研究では、ユーザーが普段からスマートフォンのような端末に、好きな楽曲を保存し聴いている事を想定し、複数人で同じ音楽を聴く際に各自所有する端末から楽曲を再生するようなシチュエーションに着目する。各自のスマートフォンに保存されている楽曲を気に入る度合いである期待度を、自身の所有するスマートフォンに記録されている鑑賞履歴や楽曲情報を用いて推測し、選曲を行い楽曲一覧を生成する。生成した楽曲一覧に沿って同一スピーカーから再生することでユーザー間で楽曲を聴かせ合うことができる。このシステムを利用することで、複数人で同一の楽曲を聴く際に、特定の個人の嗜好のみが反映されたBGMが続く状況や、とあるユーザーにとって嗜好に合わない楽曲が選択されるのを回避し、その場にいる全員の嗜好を満たすことを期待する。

実験において、我々は「再生回数が多い楽曲ほど期待度も高い」という仮定1と、「類似している楽曲は、ユーザーの期待度も類似する」という仮定2の基、ユーザーが所持していない楽曲に対する期待度の推測を行い、推測した期待度と再生回数から決定した期待度の関係を分析した。実験結果では、再生回数から決定し

た期待度と推測した期待度の相関係数について0.5以上の値が得られた。更にユーザー毎の相関係数を求めた結果、354名の内157名が相関係数について0.5以上の値であり、正の相関関係であった。この結果からユーザーの好きな度合いである期待度が再生回数に対して単調増加すると仮定した場合に、本研究で提案する手法は、ユーザーの期待度の推測に有効である事が分かった。

目 次

目 次	iii
図目次	vii
表目次	ix
第1章 序 論	1
1.1 本研究の背景	1
1.2 本研究の目的	2
1.3 本論文の構成	2
第2章 関連研究・関連システム	5
2.1 プレイリスト生成・楽曲推薦に関する関連研究	5
2.1.1 環境情報を用いたプレイリスト生成アルゴリズムの提案 [1]	5
2.1.2 開始曲と終了曲を使用したプレイリストの生成 [2]	5
2.1.3 スキップ動作に基づく動的プレイリスト生成 [3]	6
2.1.4 視聴履歴を用いた楽曲推薦システム [4]	6
2.2 複数人に向けた楽曲推薦に関する関連研究・アプリ	6
2.2.1 複数人に向けたBGM選曲手法に関する研究 [5]	6
2.2.2 リズムやテンポの入力によるグループ楽曲推薦 [6]	7
2.2.3 嗜好する音楽ジャンルの入力によるグループ楽曲推薦の提案 [7]	7

2.2.4	フィットネスジム内における複数ユーザーに向けた楽曲推薦 [8]	7
2.2.5	ユーザークラスタリングに基づくグループ推薦 [9]	8
2.2.6	嗜好の欠如を考慮したグループ推薦システム [10]	8
2.2.7	FESS[11]	8
2.2.8	Listenwith[12]	9
2.3	関連研究との類似点・相違点	9
第3章	選曲手法	11
3.1	選曲手法の概要	11
3.2	所持楽曲に対する期待度	13
3.3	楽曲間の類似度の算出	13
3.3.1	音響類似度	13
3.3.2	アーティスト類似度	14
3.3.3	楽曲間類似度	15
3.4	未所持楽曲に対する期待度	16
3.5	楽曲一覧の生成	16
第4章	アプリケーション	19
4.1	アプリケーションの概要	19
4.1.1	音楽プレイヤー機能	20
4.1.2	楽曲特徴量・類似アーティスト収集機能	20
4.1.3	グループ楽曲推薦機能	20
第5章	実験	25
5.1	所持楽曲に対する期待度計算手法の評価	25
5.1.1	収集したユーザーデータセット	26
5.1.2	仮定1の分析	28

5.2	未所持楽曲に対する期待度計算手法の評価	34
5.2.1	使用するデータセット	34
5.2.2	実験手法	34
5.2.3	実験結果	35
5.3	被験者データセットを用いた実験	36
5.3.1	被験者データセット	36
5.3.2	実験結果	38
5.4	アプリケーションの動作確認実験	39
第6章 結 論		43
参考文献		45
付録 A 被験者実験で用いた CD 一覧		51

目 次

4.1	プレイリスト生成の流れ	22
4.2	ミュージックプレイヤー：再生画面	23
4.3	ミュージックプレイヤー：楽曲選択画面	23
4.4	ミュージックプレイヤー：アルバム選択画面	24
4.5	グループ化：周辺端末の探索画面	24
5.1	評価楽曲数に対するユーザー数の累積度数分布	27
5.2	収集したデータセットの再生回数に対する楽曲数の累積度数分布	30
5.3	再生回数に対する評価の割合	30
5.4	アーティストの累計再生回数とアーティストの評価数の相関係数を ユーザー毎に算出したヒストグラム	31
5.5	式 (3.1) を用いた場合の期待度のグラフと、再生回数に対する評価 の割合の比較	32
5.6	式 (3.2) を用いた場合の期待度のグラフと、再生回数に対する評価 の割合の比較	32
5.7	グループ楽曲を対象とした、全ユーザーの所持楽曲の再生回数に対 する楽曲数のヒストグラム	37
5.8	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	38

表 目 次

5.1	図 5.5 に対するデータの詳細：アーティスト再生回数とアーティスト評価数の相関係数	29
5.2	再生回数と評価の割合に対して式 (3.1)、(3.2) をについてカーブフィッティングを検証を行った結果	33
5.3	式 (3.1) について、図 5.5 の α 値を用いた場合の相関係数と平均二乗誤差。(max-min) は RMSE の算出前に正解データの最大値-最小値で正規化	36
5.4	式 (3.2) について、図 5.6 の β 値を用いた場合の相関係数と平均二乗誤差。(max-min) は RMSE の算出前に正解データの最大値-最小値で正規化	36
5.5	被験者データセット:被験者 3 人の楽曲に対するアンケート嗜好評価	40
5.6	被験者データセット:被験者 3 人のアンケート嗜好評価の詳細	41
5.7	被験者データセット:被験者 3 人再生履歴の詳細	41
5.8	被験者データセット:被験者 3 人の推測した未所持楽曲に対する期待度とアンケート嗜好評価の相関係数	41
A.1	被験者データセット含まれる 3 人の被験者に用意した CD・アーティスト一覧	52

第1章 序 論

本章では、研究の背景、目的を述べた後、本論文の構成を述べる。

1.1 本研究の背景

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、容易に音楽を個人で楽しむことが可能になった。それに伴い、ユーザー自身の嗜好に合うコンテンツを取捨選択する際、必要以上に時間を要したり、適切なコンテンツを選択できない場合があるなどの問題が考えられるようになった。そのため、単一のユーザーに対してのプレイリスト生成を想定した研究 [1][2][3] や楽曲推薦の研究 [4] が盛んに行われてきた。

また個人で音楽を鑑賞するだけでなく、複数人で同じ音楽を鑑賞するシチュエーションも存在する。そのようなシチュエーションでは、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。その解決策として、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がいくつか存在する [5][6][7][8][9][10]。

先行研究では、複数人の嗜好を考慮したアイテムの推薦を行うために、各ユーザーの嗜好が少ない場合には、嗜好を収集することや推測することを行っている。例えば、ユーザ全員の嗜好を事前に調査する方法や [6][7][8]、各自の嗜好情報がある程度存在する場合には、協調フィルタリングや内容ベースフィルタリング、また

組み合わせたハイブリットフィルタリングを用いた手法がある [9]。それでも嗜好に欠損が出る場合には、全てのメンバーが好きなアイテムや少なくとも一人のメンバーが好きなアイテムを推薦する手法を用いた手法も存在する [10]。

1.2 本研究の目的

本研究では、各ユーザーが所有するスマートフォンから楽曲を再生し、同じ楽曲を聴くような環境に着目する。我々の提案するシステムでは、各ユーザーが所有するスマートフォン（以下、端末と呼称する）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選曲を行い、楽曲一覧の生成を行う。楽曲一覧に沿って、各端末内に保存されている楽曲を次々と再生を行う。この想定では、個々人の所有する楽曲が、それぞれの端末に分散して存在することになる。このとき、これらの楽曲をどれか1つの端末あるいは別途用意したPCやサーバに複製する方式にすると著作権上の問題が生じる。そのため、複製せずに各端末から直接再生することが望ましい。このシステムを利用することで楽曲に対する、全員の嗜好を満たした楽曲を推薦・再生ができることを目的とする。ユーザーの楽曲に対する期待度を、ユーザーの所有する端末内の情報から推測し、全員の楽曲に対する期待度を考慮した選曲・楽曲一覧の生成と、楽曲一覧に沿った楽曲の自動再生が課題である。

1.3 本論文の構成

本論文は次の構成からなる。第2章では、楽曲推薦やプレイリスト生成に関する先行研究を述べ、本研究との類似点と相違点について述べる。第3章では、本

研究で提案する選曲手法の概要と構成について述べる。第4章では、選曲手法を組み込んだアプリケーションの機能について述べる。第5章で提案する選曲手法の有効性の検証を行い、その考察について述べる。第6章で本研究の結論と今後の課題について述べる。

第2章 関連研究・関連システム

ここでは関連する研究やシステムを紹介する。これらの課題について述べたあと、本研究との類似点と相違点について述べる。

2.1 プレイリスト生成・楽曲推薦に関する関連研究

2.1.1 環境情報を用いたプレイリスト生成アルゴリズムの提案 [1]

この研究は、刻一刻と変化するユーザーを取り巻く環境情報や心理情報に合わせ、その都度適切な選曲を行わなければならないという課題に着目した、自動プレイリスト生成に関する研究である。照明、湿度やスケジュールなどの環境情報を選曲の際の基準に帰着させ、それらを機械学習によってモデル化させることでプレイリスト生成を実現しようとしている。実験が行われていないので、システムの有用性を提示する主旨の研究ではないが、環境情報の有用性などを述べている。

2.1.2 開始曲と終了曲を使用したプレイリストの生成 [2]

プレイリストの開始曲と、終了曲を設定すると、開始から終了までの曲調がスムーズに遷移するように途中の楽曲を自動選択する手法を提案している研究である。主観的評価、客観的評価ともに研究のコンセプトに沿って機能したと言えるが、結果を平均すると少なくとも1曲は生成したプレイリストの流れなどに相応しくない楽曲が含まれる結果も得られた。したがって、プレイリストの編集の機能やユーザーインターフェースの実装を今後の課題とした。

2.1.3 スキップ動作に基づく動的プレイリスト生成 [3]

この研究は、自動プレイリスト生成システムに関する論文である。プレイリストを作成する際、ライブラリをランダムにシャッフルする手法、ユーザー自身が半手動で楽曲を選択していく手法が一般的であった。仮にユーザーが聴いている楽曲が嗜好に合わない場合、ユーザーは楽曲に対してスキップ動作を行う。これを「ユーザーからのフィードバック」と解釈し、スキップされた楽曲と類似度の高い楽曲はプレイリストから削除し、同時にスキップされず受け入れられた楽曲と類似度の高い楽曲はプレイリストに追加していくというアプローチである。結果としてこれらのアプローチによってスキップ回数を大幅に低減することが可能であると結論づけた。

2.1.4 視聴履歴を用いた楽曲推薦システム [4]

この研究では協調フィルタリングと内容ベースフィルタリングの2つの手法を連動させることで双方の問題点を補い、推薦精度の向上を狙いとしている。連動させることによる有効性は確認されたが、双方の推薦手法がともに有効であるときに初めて有効に働くため、それぞれの手法の推薦精度が十分でないとは有効に機能しない。

2.2 複数人に向けた楽曲推薦に関する関連研究・アプリ

2.2.1 複数人に向けたBGM選曲手法に関する研究 [5]

この研究では、複数人がBGMを共有するためのBGM選曲手法を提案している。複数人が嗜好する楽曲集合から過半数がお気に入り指定した楽曲を選択することで、その場にいるユーザー全員が楽しめるプレイリストを生成することを

目的としたものである。嗜好楽曲の取得は Youtube のお気に入り機能を用い、各ユーザのお気に入り楽曲情報を取得する。プレイリスト生成に十分な曲数が確保できない場合は、アーティストレベルでの共通楽曲を抽出し、プレイリストに追加する機能を実装しこれを回避している。結果として共通のお気に入りの楽曲を抽出し、十分な曲数を確保できるプレイリストを生成することができたとしている。

2.2.2 リズムやテンポの入力によるグループ楽曲推薦 [6]

Jameson は、複数のユーザーがそれぞれ聴きたいと思うシチュエーションに沿ったリズムやテンポ、またはテーマを事前に指定し、それに近い楽曲を再生することで複数ユーザに向けた BGM の楽曲推薦を行った。その結果、複数ユーザーの入力情報を考慮したプレイリストの生成ができたとしている。

2.2.3 嗜好する音楽ジャンルの入力によるグループ楽曲推薦の提案 [7]

Crosson らは、複数ユーザーの嗜好する楽曲のジャンルを予め設定することで、複数ユーザーの設定したジャンルに近い楽曲を次々に再生するシステムの提案を行っている。結果として、複数ユーザーの入力したジャンルを考慮したプレイリストの生成が可能と結論づけた。

2.2.4 フィットネスジム内における複数ユーザーに向けた楽曲推薦 [8]

Joseph らは、複数人が集まるフィットネスジム内で再生する音楽ラジオチャンネルを決定するシステムを開発した。ユーザーはフィットネスジム内で再生されている楽曲に対して任意に評価を行う。ユーザーにバッジを付けることで部屋の出

入りをチェックし、現在部屋にいるユーザーに合った音楽ラジオチャンネルを推薦する。このシステムの用いることで、リアルタイムに部屋にいるユーザーに合わせた楽曲を再生が可能になり、ユーザーの楽曲の満足度が向上した。

2.2.5 ユーザークラスタリングに基づくグループ推薦 [9]

Ntoutsisらは、MovieLens データセットを用いて、グループ推薦に対する映画の推薦を提案している。凝集階層クラスタリングを用いた協調フィルタリングを使用することで、従来手法より類似ユーザ検索の計算量を削減することができた。

2.2.6 嗜好の欠如を考慮したグループ推薦システム [10]

複数人グループに、アイテムを推薦する場合に、全員の嗜好情報が不明である時、不明である情報を使用しないでグループに対するアイテムの推薦を行った。全てのユーザーが好きなアイテムや、少なくとも一人のメンバーが好きなアイテムを推薦するようにモデルを作ることで、ユーザーの嗜好が一部分存在しない場合でも複数人に対する推薦を可能にした。

2.2.7 FESS[11]

FESSは最大5人で音楽が共有できるiOS 端末向けのアプリケーションである。各種通信はBluetoothを介して行われる他、親機、子機のあることから本研究に類似したアプリケーションであるといえる。しかし、楽曲の選択はすべてユーザーの手動によるもので選曲を自動で選曲を行う機能は無い。

2.2.8 Listenwith[12]

FESS[11]と同様、Bluetoothを介して音楽を共有できるiOS端末向けのアプリケーションである。最大接続可能人数が2名であり、ユーザーが手動による選曲を行う必要があり、自動で選曲をする機能はない。

2.3 関連研究との類似点・相違点

ユーザーの鑑賞履歴を用いるものなど本研究と類似している点を含む研究はためられている。しかし、本章で挙げた研究のうちプレイリスト生成に関する研究と楽曲推薦に関連する研究の双方ですべての研究[1][2][3][4]が単一のユーザーを対象としたものであり、ユーザー全員が気に入るであろうという楽曲を選択し、聴きあうという主旨の研究は十分に為されていない。

複数人に対して楽曲の推薦を行っている研究[5][6][7]について、ユーザーの楽曲に関する嗜好評価を事前に調査を行い複数人への楽曲の選曲を行っている。各ユーザーの楽曲に対する嗜好評価を推測し、複数人に向けた楽曲リストの生成を行う試みではない。Josephらの研究[8]も複数のユーザを対象とした楽曲の推薦を行っているが、複数人で音楽を聴くシチュエーションの場面において、システムを繰り返し使用しながらユーザーの楽曲に対する評価の収集をしている。しかしユーザーが直接楽曲の評価を行うには、直接楽曲を聴く必要があり、手間と時間がかかる。また複数人の嗜好をMovieLensデータセットを用いて予測し、映画の推薦を行っている研究[9]もあるが、事前にドメインに応じたデータセットを用意する必要がある。グループのユーザーの嗜好に一部欠損がある状態で複数人に対する推薦を行う際に、全員、または少なくとも一人のメンバーが好きなアイテムを推薦するようにする研究[10]もあるが、我々の想定しているシチュエーションでは、各ユーザーの所持楽曲に対して、楽曲を所持していないユーザーの嗜好が殆ど欠損に当たるため、我々の想定しているシチュエーションでは、使用できない。

楽曲共有に関するシステム [11][12] は、いずれも複数ユーザーの所有する楽曲を参照するものではあるが、ユーザーが手動で楽曲を選択する必要があり、プレイリストを自動で生成する主旨のものでは無い。PC やサーバへのアップロードを前提としている研究 [6][7] は、楽曲のコピーによる著作権の問題が生じる場合がある。我々の想定では、個々人の所有する楽曲が、それぞれの端末に分散して存在することになる。このとき、これらの楽曲をどれか1つの端末あるいは別途用意したPC やサーバに複製を行わずに各端末から直接再生を行う。

第3章 選曲手法

本章では、我々の提案する選曲手法について述べる。

3.1 選曲手法の概要

本稿では、ユーザーの楽曲を気に入る度合いを期待度と定義する。本研究の目的は、複数人が同じ楽曲を聴くというシチュエーションにおいて、各自の所有する端末に保存されている楽曲から全員の嗜好を満たす選曲を行い、楽曲の一覧を生成・再生をすることである。

こうした状況において選曲手法として考えられる手法の一つとして、一人一人の楽曲に対する期待度を統合し、全員の期待度が高い楽曲を選出する手法が挙げられる。しかし、このようなシチュエーションにおいて、ユーザーの所有していない、または聴いたことのない楽曲に対しての期待度は未知であるため、ユーザーの期待度を推測する必要がある。

我々は、以下の2つの仮定の基、選曲手法を提案する。

仮定1

再生回数が多い楽曲ほど期待度も高い。

仮定2

類似している楽曲は、ユーザーの期待度も類似する。

我々の提案する選曲手法（以降、提案手法）は、仮定1の基、再生履歴からユーザーの所有する楽曲に対する期待度を推測する。次に仮定2の基、ユーザーの所持

していない楽曲に関しては、ユーザーの所持している楽曲の期待度を用いて、楽曲間の類似している度合いに応じて推測する。楽曲に対する全ユーザーの期待度が高い楽曲を選曲し楽曲一覧を生成する。

この提案手法を実現するには次の2つの課題を解決する必要がある。

課題1

ユーザーの所持楽曲に対する期待度を再生履歴からどのように決定するか

課題2

ユーザーの所持していない楽曲に対する期待度をどのように算出するか

本研究では、課題1に対しては、仮定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。課題2に対してのアプローチとして、本研究では仮定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類似すると考える。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。

この章では選曲手法による楽曲一覧の生成手法について述べる。選曲手法を組み込んだ音楽プレイヤーのアプリケーションの構成を4章で述べ、提案手法の有効性の検証については5章で述べる。また3.2節~3.5節において n 人のユーザーの集合を $U_{\text{all}} = \{u_1, u_2, \dots, u_n\}$ とし、ユーザー u_i が v 曲所持しているとき、 u_i の所持楽曲の集合を $M_{(u_i)} = \{m_{(u_i,1)}, m_{(u_i,2)}, \dots, m_{(u_i,v)}\}$ 、ユーザー u_j が z 曲所持しているとき、 u_j の所持楽曲の集合を $M_{(u_j)} = \{m_{(u_j,1)}, m_{(u_j,2)}, \dots, m_{(u_j,z)}\}$ と定義する。

3.2 所持楽曲に対する期待度

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し、仮定1に基づいて、ユーザーの所有楽曲は再生回数の増加と共に期待度が単調増加するように決定する。ユーザー u_i の所持楽曲 $m_{(u_i,k)}$ の再生回数を $F(m_{(u_i,k)})$ とする。 $m_{(u_i,k)} \in M_{(u_i)}$ を満たすとき、 $m_{(u_i,k)}$ に対するユーザー u_i の期待度 $W(m_{(u_i,k)})$ として以下の2式を提案する (e をネイピア数とする)。

$$W(m_{(u_i,k)}) = 1 - \frac{1}{\{F(m_{(u_i,k)}) + 1\}^\alpha} \quad (3.1)$$

$$W(m_{(u_i,k)}) = \left(\frac{1}{1 + e^{-\beta \times F(m_{(u_i,k)})}} - 0.5 \right) \times 2 \quad (3.2)$$

α 、 β の値を変化させた時については、5.2節の実験において述べる。式(3.1)(3.2)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。

3.3 楽曲間の類似度の算出

本研究では、楽曲間の類似度に音響特徴量を用いた音響類似度と、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。

3.3.1 音響類似度

音響的特徴の類似度を楽曲の類似度として用いる手法は多くの音楽情報検索システムで用いられており、代表的な特徴量の一つとしてメル周波数ケプストラム係数 (Mel-Frequency Cepstrum Coefficient: MFCC) がある。E. Pampalkら [13] は音響特徴量を楽曲データの中心の30秒間を対象に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 $m_{(u_i,k)}$ 、 $m_{(u_j,l)}$ 間の距離を $D(m_{(u_i,k)}, m_{(u_j,l)})$ とするとき、 $m_{(u_i,k)}$ 、 $m_{(u_j,l)}$ 間の音響類似度を $Similarity_{MFCC}(m_{(u_i,k)}, m_{(u_j,l)})$ とし、以下のように計算する。

$$Similarity_{MFCC}(m_{(u_i,k)}, m_{(u_j,l)}) = \frac{1}{1 + D(m_{(u_i,k)}, m_{(u_j,l)})} \quad (3.3)$$

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 章で述べる音響特徴量収集機能において音響類似度の算出を行う前に、既を取得している事とする。

3.3.2 アーティスト類似度

Last.fm[15] 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の情報を取得することができる。類似アーティストを求める為に使用した Last.fm の WebAPI のメソッドは `artist.getSimilar` である。`artist.getSimilar` は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 XML もしくは json 形式で取得できる。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 $m_{(u_i,k)}$ 、 $m_{(u_j,l)}$ のアーティストをそれぞれ $A(m_{(u_i,k)})$ 、 $A(m_{(u_j,l)})$ とし、 $A(m_{(u_i,k)})$ の類似アーティスト 100 件の集合を $S_{A(m_{(u_i,k)})}$ とする。また $A(m_{(u_j,l)}) \in S_{A(m_{(u_i,k)})}$ を満たす時の $m_{(u_i,k)}$ 、 $m_{(u_j,l)}$ の類似度を $L(A(m_{(u_i,k)}), A(m_{(u_j,l)}))$ とする。本来 $m_{(u_i,k)}$ 、 $m_{(u_j,l)}$ のアーティスト類似度 $Similarity_{Artist}(A(m_{(u_i,k)}), A(m_{(u_j,l)}))$ は、 $Similarity_{Artist}(A(m_{(u_j,l)}), A(m_{(u_i,k)}))$ と等しくなるべきだが Last.fm の仕様上、一般に

$$L(A(m_{(u_i,k)}), A(m_{(u_j,l)})) = L(A(m_{(u_j,l)}), A(m_{(u_i,k)}))$$

を満たさない。そこで、以下の様に定義する。

$A(m_{(u_i,k)}) = A(m_{(u_j,l)})$ を満たすとき、

$$\underset{\text{Artist}}{\text{Similarity}}(A(m_{(u_i,k)}), A(m_{(u_j,l)})) = 1.0 \quad (3.4)$$

$A(m_{(u_j,l)}) \in \mathcal{S}_{A(m_{(u_i,k)})} \wedge A(m_{(u_i,k)}) \in \mathcal{S}_{A(m_{(u_j,l)})}$ を満たすとき、

$$\underset{\text{Artist}}{\text{Similarity}}(A(m_{(u_i,k)}), A(m_{(u_j,l)})) = \frac{L(A(m_{(u_i,k)}), A(m_{(u_j,l)})) + L(A(m_{(u_j,l)}), A(m_{(u_i,k)}))}{2} \quad (3.5)$$

$A(m_{(u_j,l)}) \in \mathcal{S}_{A(m_{(u_i,k)})} \wedge A(m_{(u_i,k)}) \notin \mathcal{S}_{A(m_{(u_j,l)})}$ を満たすとき、

$$\underset{\text{Artist}}{\text{Similarity}}(A(m_{(u_i,k)}), A(m_{(u_j,l)})) = L(A(m_{(u_i,k)}), A(m_{(u_j,l)})) \quad (3.6)$$

$A(m_{(u_j,l)}) \notin \mathcal{S}_{A(m_{(u_i,k)})} \wedge A(m_{(u_i,k)}) \in \mathcal{S}_{A(m_{(u_j,l)})}$ を満たすとき、

$$\underset{\text{Artist}}{\text{Similarity}}(A(m_{(u_i,k)}), A(m_{(u_j,l)})) = L(A(m_{(u_j,l)}), A(m_{(u_i,k)})) \quad (3.7)$$

$A(m_{(u_j,l)}) \notin \mathcal{S}_{A(m_{(u_i,k)})} \wedge A(m_{(u_i,k)}) \notin \mathcal{S}_{A(m_{(u_j,l)})}$ を満たすとき、

$$\underset{\text{Artist}}{\text{Similarity}}(A(m_{(u_i,k)}), A(m_{(u_j,l)})) = 0.01 \quad (3.8)$$

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。

3.3.3 楽曲間類似度

$m_{(u_i,k)}$ 、 $m_{(u_j,l)}$ 間の楽曲間類似度 $\text{Similarity}(m_{(u_i,k)}, m_{(u_j,l)})$ を式 (3.3) ~ (3.8) を用いて以下の式で算出する。

$$\text{Similarity}(m_{(u_i,k)}, m_{(u_j,l)}) = \underset{\text{MFCC}}{\text{Similarity}}(m_{(u_i,k)}, m_{(u_j,l)}) \times \underset{\text{Artist}}{\text{Similarity}}(A(m_{(u_i,k)}), A(m_{(u_j,l)})) \quad (3.9)$$

3.4 未所持楽曲に対する期待度

仮定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 の楽曲 $m_{(u_j,l)}$ に対する期待度 $R(u_i, m_{(u_j,l)})$ は、式(3.1)、(3.2)、(3.9)、を用いて以下の式で行う。

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

$$R(u_i, m_{(u_j,l)}) = \sum_{q=1}^v \frac{W(m_{(u_i,q)}) \times \text{Similarity}(A(m_{(u_i,q)}), A(m_{(u_j,l)}))}{\sum_{t=1}^v \text{Similarity}(A(m_{(u_i,t)}), A(m_{(u_j,l)}))} \quad (3.10)$$

$m_{(u_j,l)} \in M_{(u_i)}$ を満たすとき、

$$R(u_i, m_{(u_j,l)}) = W(m_{(u_j,l)}) \quad (3.11)$$

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。

3.5 楽曲一覧の生成

式(3.6)、(3.7)より、複数ユーザーの所持する楽曲について、 n 人の期待度が推定済みとする。全員の所持する楽曲を統合した楽曲集合を $M_{all} = \bigcup_{t=1}^n M_{(u_t)} = \{m_1, m_2, \dots, m_h\}$ とするとき、楽曲 m_g に対する n 人のユーザーの期待度を統合したものを $R_{all}(m_g)$ とする。その計算を以下の式で行う。

$$R_{all}(m_g) = \prod_{q=1}^n R(u_q, m_t) \quad (3.12)$$

式(3.12)より M_{all} に対する n 人の統合した期待度が、付与されている状態とし、統合した期待度が大きい順に M_{All} を並び替えたものを選曲した楽曲一覧とする。統合した期待度が同値である楽曲は、辞書順で先に出現したものを優先する。生

成した楽曲一覧の再生方法については、4章のアプリケーションの構成において述べる。

第4章 アプリケーション

本章では、3章で述べたシステムを組み込んだアプリケーションの概要と構成について述べる。

4.1 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[14] に限定し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

音楽プレイヤー機能は、ユーザーが端末を用いて楽曲を聴く機能であり、再生された楽曲の情報を再生履歴として端末へ記録する。楽曲特徴量・類似アーティスト収集機能は、3章で述べた楽曲データから音響的特徴 MFCC の抽出する機能と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取得を行う機能である。グループ楽曲推薦機能は、本研究で想定している複数人が同じ音楽を聴くシチュエーションにおいて、複数人に向けた楽曲リストの生成する機能である。本章ではそれぞれの機能のについての詳細を述べる。また、グループ楽曲推薦の機能の流れを図 4.1 で示す。

4.1.1 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。音楽プレイヤーの再生機能画面と楽曲選択画面、アルバム別の楽曲選択画面をそれぞれ図 4.2、図 4.3、図 4.4 に示す。

4.1.2 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。

4.1.3 グループ楽曲推薦機能

各ユーザーの持つ端末を Bluetooth を用いて相互に通信を行う。次に各端末に保存されている楽曲の情報を集約した後に楽曲リストを生成を行い、楽曲リストに沿って Bluetooth 対応のスピーカーから次々に再生するシステムとなっている。

ユーザー n 人が所持する端末の集合を $D_{\text{all}} = \{D_1, D_2, \dots, D_n\}$ 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_{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。

次に生成された楽曲リストに基づいて同一の Bluetooth 対応スピーカーから楽曲を再生する処理について述べる。楽曲リストは、個々人の所有する端末に別々に保存されている楽曲から構成されている。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。楽曲の再生が終了した端末は再生終了の通知を親機へ伝達し、Bluetooth スピーカーとの接続を切断する。このように楽曲リストに沿って再生する端末が Bluetooth スピーカーへ自動接続を行うことで各端末に保存されている楽曲が、同一スピーカーから再生され楽曲を聴くことができる。Bluetooth スピーカーとの接続に用いた Bluetooth のプロファイルはステレオ再生が可能な A2DP(Advanced Audio Distribution Profile) を使用した。また再生をしていない端末には、再生中の端末名やアーティスト名、楽曲名などの再生状況を送信し、端末の画面に表示することでリアルタイムに確認することができる。親機が行うデータの集約や親機から子機への再生状況の送信、子機から親機への再生終了の通知、グループ化など Bluetooth スピーカーとの通信以外には、SPP(Serial Port Profile) を使用した。

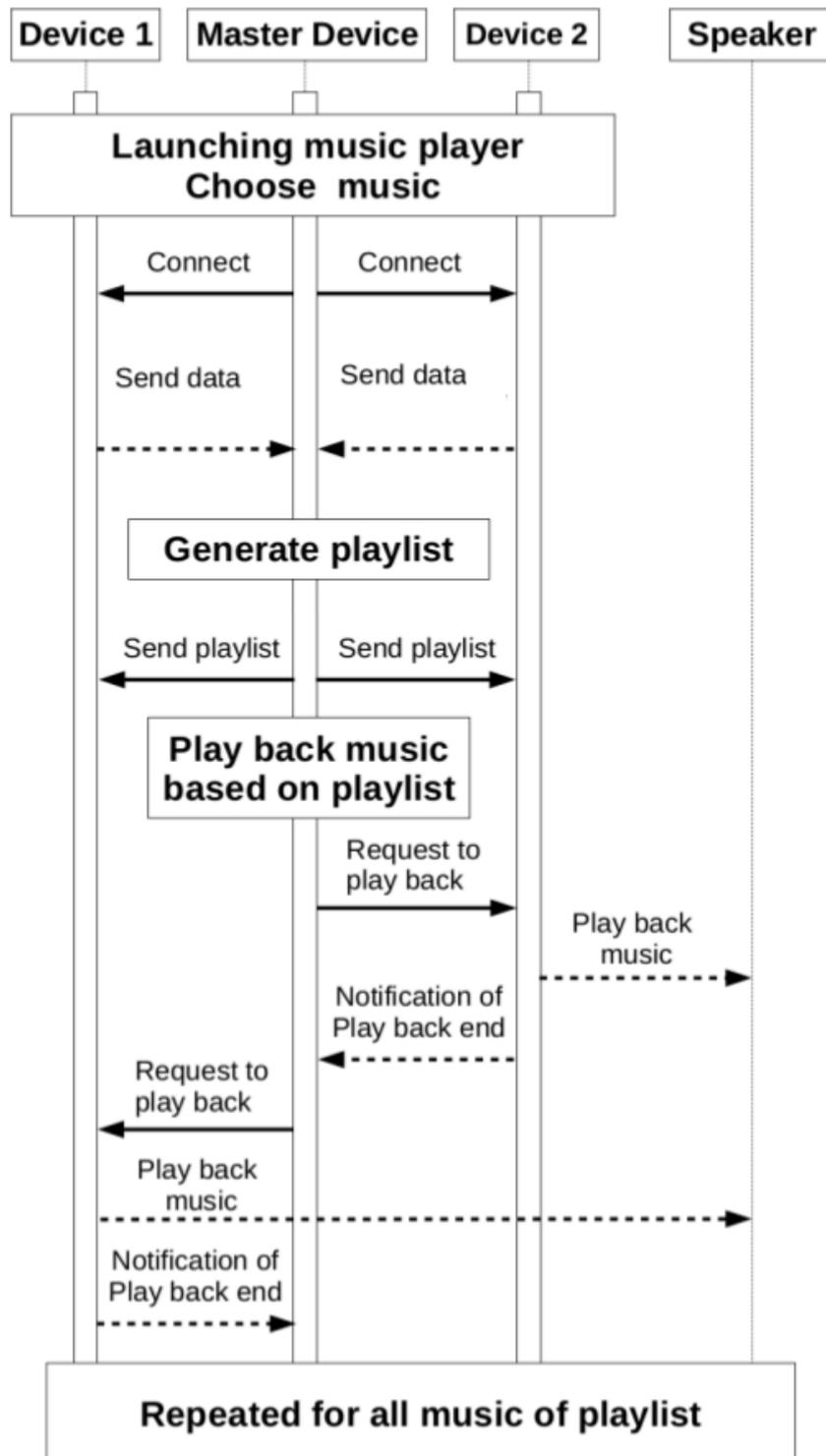


図 4.1: プレイリスト生成の流れ

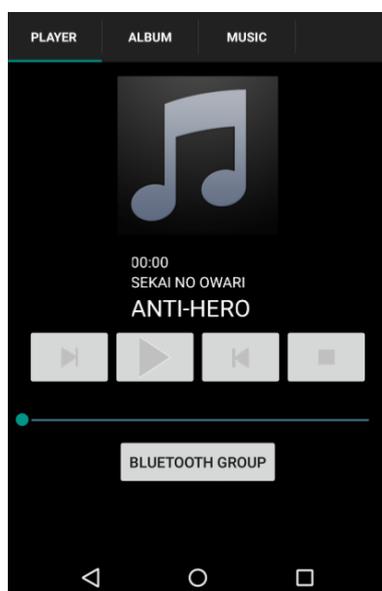


図 4.2: ミュージックプレイヤー：再生画面



図 4.3: ミュージックプレイヤー：楽曲選択画面

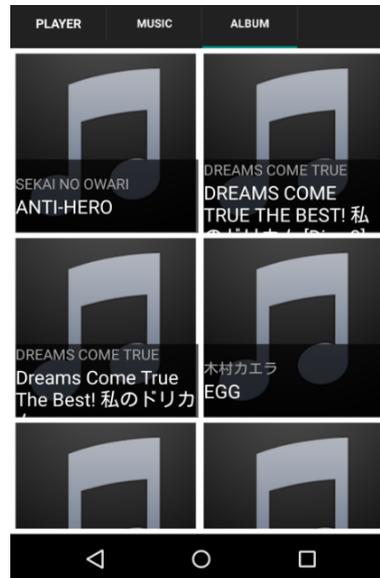


図 4.4: ミュージックプレイヤー：アルバム選択画面

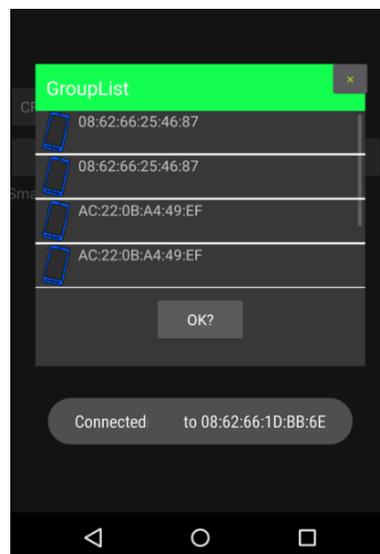


図 4.5: グループ化：周辺端末の探索画面

第5章 実 験

本章では3.1節で述べた仮定1、仮定2に基づいて期待度の推測を行い、選曲手法の有効性を検証する。5.1節の仮定1では、ユーザーの楽曲に対する再生回数と楽曲を好む度合いの関係を調べ、再生回数を期待度へ正規化する手法について考察する。5.2節では、標準的データベース Last.fm DataSets[17] を用いて3人1組である計118グループを対象に実験を行い、実験結果から、本研究で提案手法の有効性を検討する。5.3節では、3人1グループの被験者を1組（以降、被験者データセット）に対して実験を行う。本研究で想定している、複数人が端末を用いて同じ楽曲を聴くシチュエーションと近い環境を構築し、提案手法を用いて実験を行う。本研究の選曲手法の有効性について考察を述べる。5.4節では、作成したアプリケーションの動作確認実験を行った結果について述べる。

5.1 所持楽曲に対する期待度計算手法の評価

仮定1の内容である「再生回数が多い楽曲ほど期待度も高い。」について再生回数から楽曲を好む度合いである期待度へ正規化を行う式(3.1)、(3.2)について分析をする。仮定1について、再生回数の増加に伴いユーザーが楽曲を評価する割合も単調増加であると考えられる。そこでユーザーのプロファイル、再生履歴と楽曲評価についてのデータを Last.fm API を用いて収集し、楽曲の再生回数と楽曲を好む度合いの関係を調査する。5.1.1では収集したデータセットの概要を述べ、5.1.2では式(3.1)、(3.2)を用いて算出した期待度について分析と考察を述べる。

5.1.1 収集したユーザーデータセット

我々はユーザー間で音楽の嗜好の共有を主とした、SNS（ソーシャルネットワーキングサイト）である Last.fm の「フレンド機能」を用いてユーザーのプロファイルの収集を行った。「フレンド機能」とは、Last.fm 上のユーザーが互いに「フレンド機能」を用いてリストへ追加し合うことで、ユーザー同士の楽曲の趣味や嗜好を共有できる機能である。Last.fm 上でランダムに選んだ1ユーザーから深さが2つまでのフレンド機能のリストに記録されているユーザーのプロファイルの収集を行い、合計 45,745 ユーザーのプロファイルを集めた。また Last.fm には、ユーザーが楽曲に対する評価を行える機能が存在する。ユーザーは、ポジティブな評価である「Loved」と、ネガティブな評価である「Ban」という2つの機能を用いて楽曲に対して評価を行うことができるが、「Ban」の評価が付いている楽曲の収集が可能で、Last.fm API は現段階では公開されていない。よって収集可能である「Loved」が付いている楽曲の情報を 45,745 ユーザーを対象に収集した。楽曲を1曲も評価していない 11,074 ユーザーを取り除いたユーザー数は 34,671 ユーザーである。楽曲を1曲以上評価したことのある 34,671 ユーザーを対象にユーザー毎の合計楽曲評価数の累積度数分布を図 5.1 に示す。仮定 1 の検証において、再生回数と期待度の関係を調べるために、一定の評価数のあるユーザー限定する必要がある。そのため、楽曲の評価数が 20 以上 5,000 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 Last.fm API を用いて収集した。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呼び、ユーザーの楽曲をアーティスト単位で集約し合計した再生回数を、そのアーティストの再生回数とする。またアーティスト単位で集約した楽曲の内、「Loved」の評価が付いている楽曲数を、そのアーティストの評価数とする。以降、5.1.2 節において「Loved」の評価が付いている楽曲を、評価楽曲と呼ぶ。

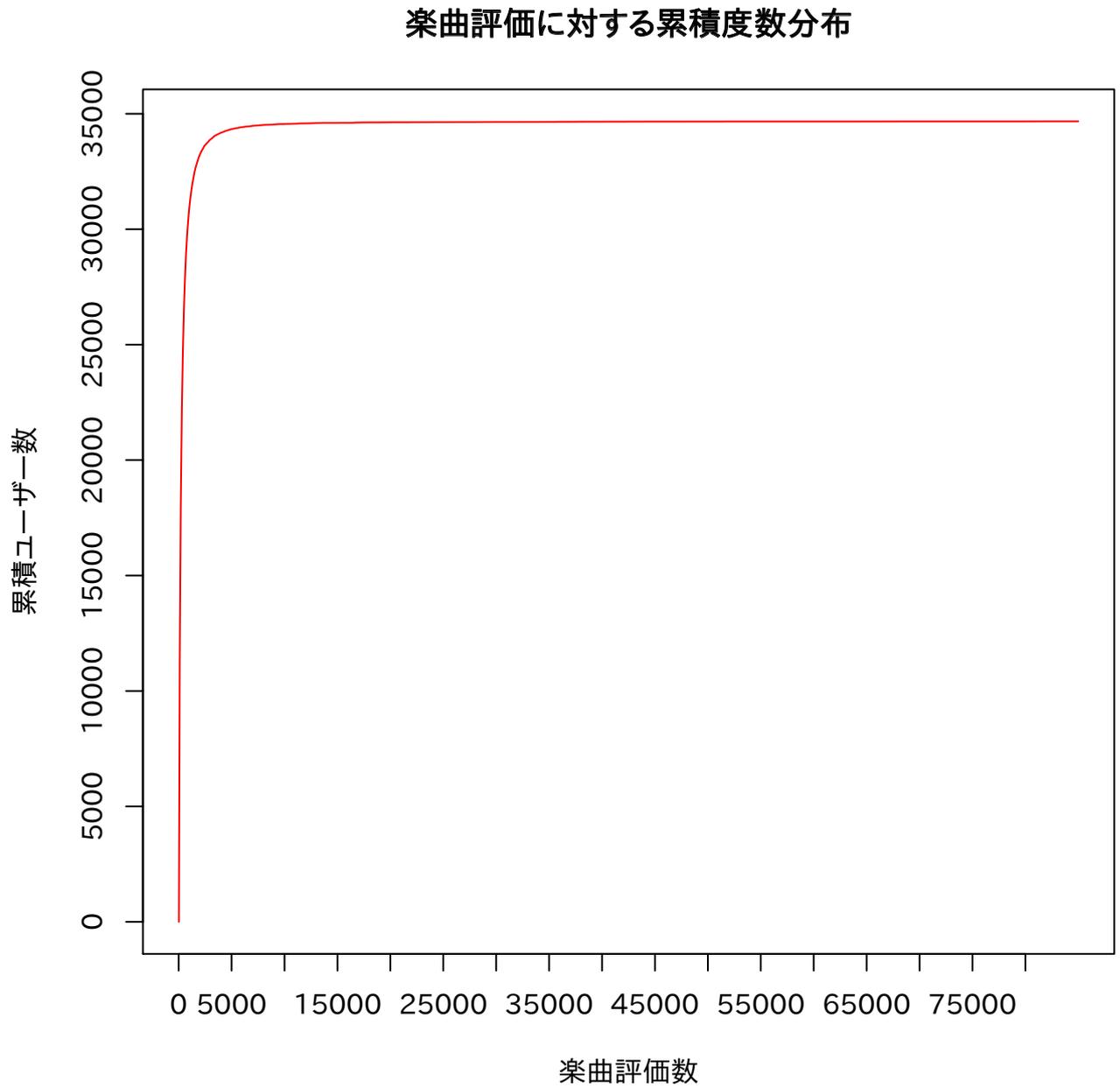


図 5.1: 評価楽曲数に対するユーザー数の累積度数分布

5.1.2 仮定1の分析

10,201 ユーザーの楽曲の再生回数と評価の対データ、計 98,504,128 個について再生回数と期待度の関係を調査を行う。しかし、Last.fm の「Loved」評価は、段階的な評価ではない為、再生回数の増減による楽曲を好む度合いの関係を調べる事ができない。そのため、再生回数に対する楽曲に評価を付ける割合を楽曲を好む度合いと解釈し、この2つの関係を調査する。再生回数の増加に伴い楽曲を評価する割合が増加するならば仮定1は正しいと言える。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。図5.2から100再生回数を超える楽曲数は、少ない事が分かる。評価の割合を調査する際に楽曲の母数が減少すると、評価の割合が正確に求められないため、楽曲を10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再生区切り区間の楽曲数が10曲以下である場合、不採用とする。採用された区間の楽曲に評価を付けている割合を図5.3に示す。採用区間の最大再生回数は6,720回であった。図5.3から再生回数が増加するに伴い、楽曲に評価を付ける割合も増加し、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割に達する。また1,000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、楽曲を好む度合いである期待度は再生回数の増加と共に、単調増加していくと図5.3から十分に言える。

次にアーティスト単位で、再生回数と評価数の関係性を調べる。アーティストの再生回数とアーティストの評価数の相関関係を確認する。アーティスト再生回数とアーティスト評価数の相関係数の平均をユーザー毎に算出し、その相関係数に対する10,201ユーザーのヒストグラムを図5.4、データの詳細を表5.1に示す。図5.4、表5.1から、アーティストの再生回数の増加に伴い、アーティスト評価数も増加しているため、相関関係にあると言える。

次に、仮定1の分析結果から再生回数の増加に伴い、期待度の増加率も図5.3の

再生回数に対する評価の割合に従うと考えた。図 5.5、図 5.6 は式 (3.1)、(3.2) の α 、 β の変化による、図 5.2 の再生回数と評価の割合の関係グラフを比較したものである。評価の割合については 1 再生区切り算出している。楽曲数が減少すると評価の割合を正確に求めることが難しい為、カーブフィッティングを再生回数が 515 回で打ち切る。表 5.2 は、図 5.5、図 5.6 のグラフに沿って式 (3.1)、(3.2) の α 、 β の値を変動させた時の再生回数に対する期待度と、再生回数に対する楽曲に評価を付けている割合について平均二乗誤差 (RMSE) を用いてカーブフィッティングを行った結果を示している。式 (3.1)、(3.2) について、それぞれ $\alpha = 0.35$ 、 $\beta = 0.1$ の場合に、再生回数に対する評価の割合のグラフに近づいたことが分かる。

以上、楽曲の再生回数と評価の割合、アーティスト再生回数とアーティスト評価数の相関関係のことから、再生回数の増加に応じて期待度も単調増加するような正規化を行う式 (3.1)、(3.2) を提案した。式 (3.1)、(3.2) について、表 5.2 の α 、 β の値に沿って期待度を推測した場合の結果については、5.2 節で述べる。

表 5.1: 図 5.5 に対するデータの詳細：アーティスト再生回数とアーティスト評価数の相関係数

ユーザー数	ユーザー毎の 相関係数の平均値	ユーザー毎の 相関係数の最頻値	全ユーザーの アーティスト数
10,201	0.79	0.886	3317477

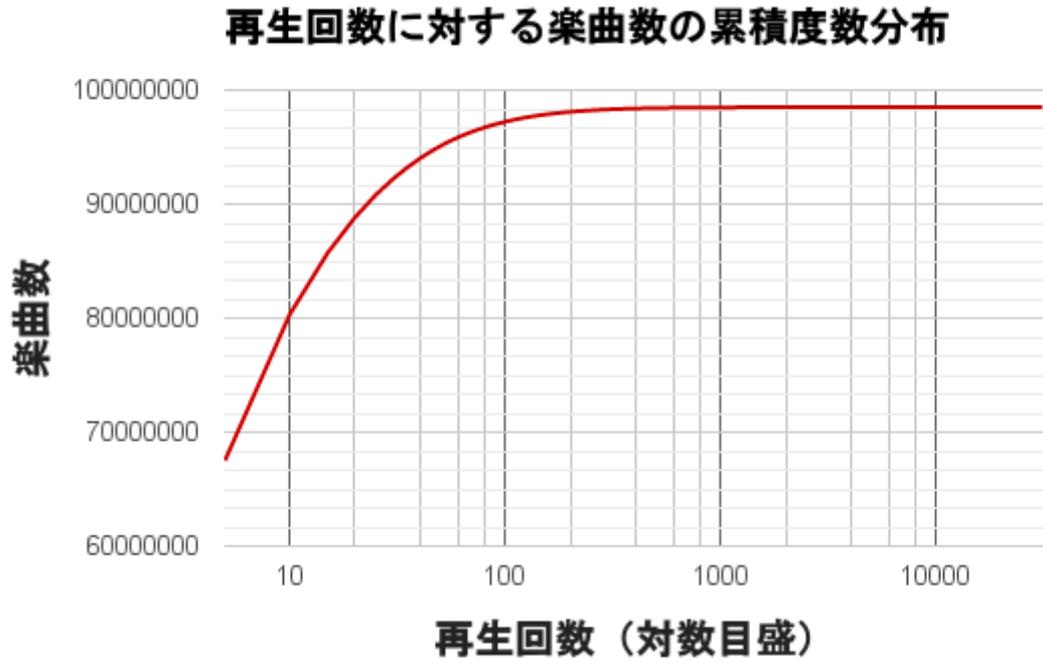


図 5.2: 収集したデータセットの再生回数に対する楽曲数の累積度数分布

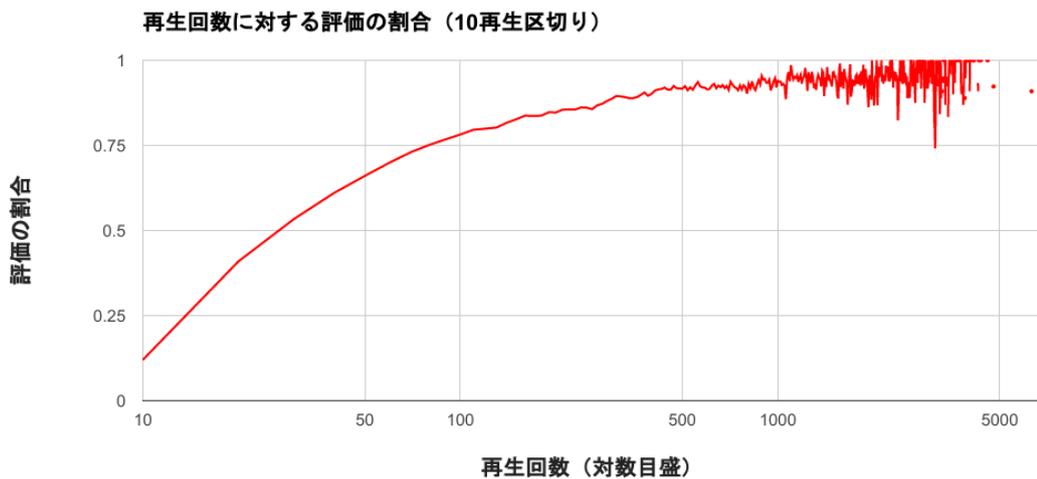


図 5.3: 再生回数に対する評価の割合

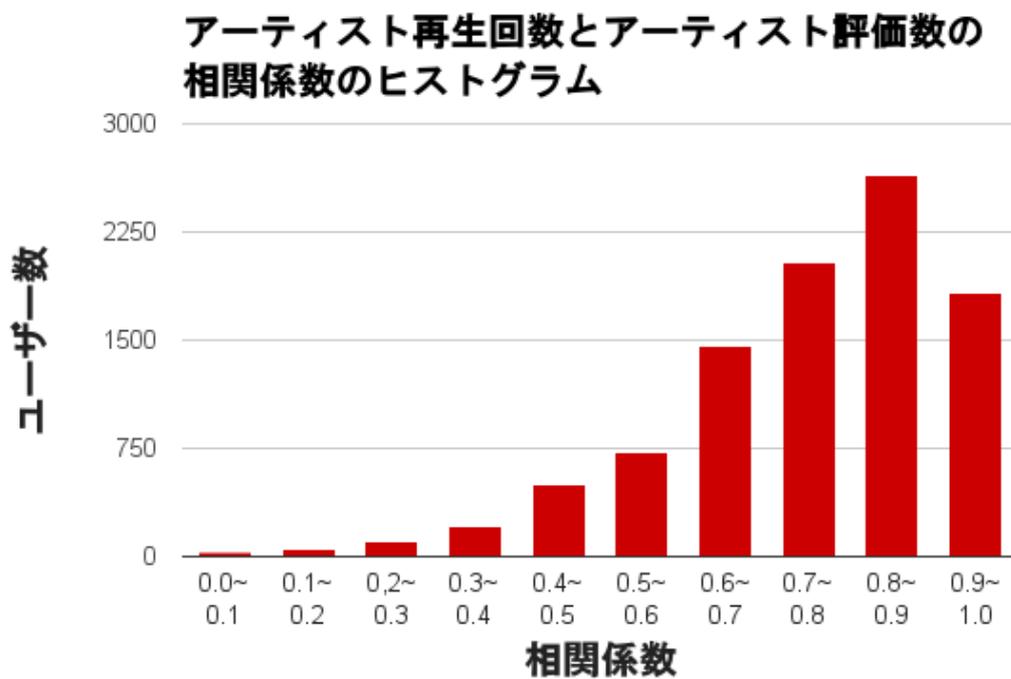


図 5.4: アーティストの累計再生回数とアーティストの評価数の相関係数をユーザー毎に算出したヒストグラム

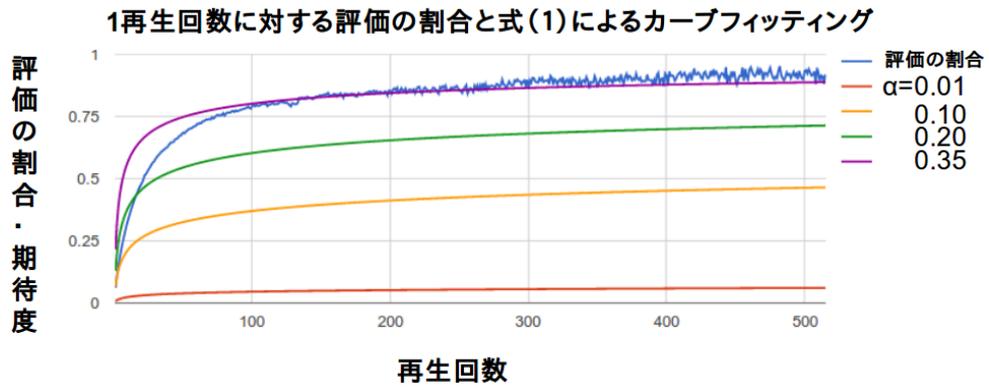


図 5.5: 式 (3.1) を用いた場合の期待度のグラフと、再生回数に対する評価の割合の比較

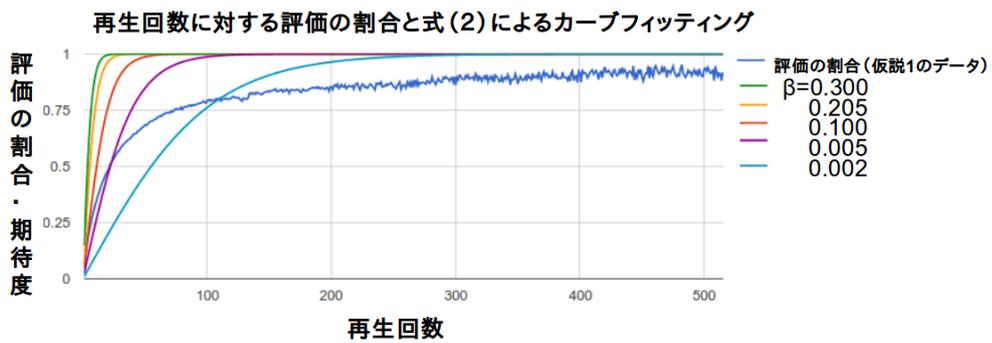


図 5.6: 式 (3.2) を用いた場合の期待度のグラフと、再生回数に対する評価の割合の比較

表 5.2: 再生回数と評価の割合に対して式 (3.1)、(3.2) をについてカーブフィッティングを検証を行った結果

	式 (3.1)			
	$\alpha = 0.01$	$\alpha = 0.10$	$\alpha = 0.20$	$\alpha = 0.35$
RMSE	0.78520	0.42659	0.19063	0.05670
	式 (3.2)			
	$\beta = 0.002$	$\beta = 0.005$	$\beta = 0.100$	$\beta = 0.205$
RMSE	0.58692	0.34897	0.16295	0.18708

5.2 未所持楽曲に対する期待度計算手法の評価

5.2.1 使用するデータセット

The Last.fm Dataset には、last.fm に登録している 1000 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音響特徴量について、The Last.fm Dataset には楽曲データは含まれていない為、7digital[16] 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入手できる。本実験ではこの試聴用データをもとに、音響特徴抽出および The Last.fm Dataset を用いた実験を行う。

5.2.2 実験手法

The Last.fm Dataset のデータを基に、共通の楽曲を 300 曲以上所持している 3 ユーザー 1 組のグループを 118 グループ作成した。本実験において、グループ内のユーザーの共通の所持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、式 (3.1)、(3.2) に従い期待度へ正規化を行い、それぞれのユーザーの正解となる期待度のデータとする（以降、正解データ）。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っ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ったものを推測データとし、正解データと推測した期待度の相関関係を調べる。

5.2.3 実験結果

118 グループ 354 ユーザーを対象に式 (3.1)、(3.2) を用いてユーザーの再生回数を期待度へ正規化を行い、本研究で提案した推薦手法を用いた時の推測データと、正解データの関係について調査する。

式 (3.1) における α の値を変化させたときの、正解データと推測した期待度の相関係数および平均自乗誤差 (RMSE) を表 5.3 に示す。ただし、期待度の取りうる値が α によって大きく異なるため、平均二乗誤差は正解データの最大値と最小値の差で除算することで、正規化を行っている。相関係数については、 α がいずれの値のときも 0.59 から 0.61 の間の値をとっており、ほとんど変動することはなかった。平均二乗誤差については、 α の値が小さくなるにつれて、減少していく傾向がみられた。同様に、式 (3.2) の α の値を変化させたときの、正解データと推測した期待度の相関係数および平均二乗誤差を表 5.4 に示す。正規化についても表 5.3 と同様である。相関係数については、式 (3.1) と同様、 α の値がいずれの場合も相関係数は 0.56 以上 0.61 未満であり、ほとんど変動することはなかった。平均二乗誤差も式 (3.1) と同様、 α の値が小さくなるにつれて減少する傾向がみられた。

図 5.7 は 118 グループ計 354 ユーザーのグループ所持楽曲の再生回数に対する楽曲数を示す。図 5.7 から、本実験で用いたグループユーザーの再生回数データは 1 ~ 15 回が多く占めている事が分かり、今回の実験において期待度の推測の対象が再生回数が 1 ~ 15 回程度の楽曲に対して行った結果であることが分かる。

図 5.8 は、(3.1) の α 値が 0.01 の時の、ユーザー毎の推測データと正解データの相関係数のヒストグラムを表したものである。図 5.10 から全体的に正の相関関係の傾向がある。

以上の結果から、再生回数を期待度へ正規化した際の値よりも、正規化後の期待度の大小関係が合っていれば、式 (3.1)、(3.2) のどちらを用いても相関係数は大きく変動しない事が言える。また、本実験で主に検証の対象となった再生回数 1 ~ 15 回の楽曲に対して、仮定 1 と仮定 2 を用いて期待度を推測した時に正解データ

と推測データに相関関係があり、本研究で提案した期待度の推測について有効であると言える。

表 5.3: 式 (3.1) について、図 5.5 の α 値を用いた場合の相関係数と平均二乗誤差。

(max-min) は RMSE の算出前に正解データの最大値-最小値で正規化

α 値	0.01	0.1	0.2	0.3	0.4
相関係数	0.60548	0.60419	0.60217	0.59960	0.59658
RMSE(max-min)	0.15717	0.16951	0.18332	0.197013	0.21038

表 5.4: 式 (3.2) について、図 5.6 の β 値を用いた場合の相関係数と平均二乗誤差。

(max-min) は RMSE の算出前に正解データの最大値-最小値で正規化

β 値	0.002	0.005	0.100	0.205	0.3
相関係数	0.56625	0.56708	0.60060	0.60132	0.59658
RMSE(max-min)	0.05850	0.05975	0.19225	0.25625	0.28837

5.3 被験者データセットを用いた実験

5.3.1 被験者データセット

被験者は男性 3 名 (全員年齢 23 歳) の 1 組のグループである。Android 端末は我々が用意したものを使用する。我々は、事前に被験者が購入したい CD をそれぞれ 10 枚調査し、合計 30 枚の CD に収録されている計 167 曲を用意した。各被験者の端末には各自が購入したいと回答した CD の楽曲のみを格納した。格納した楽曲をユーザーの所持楽曲とし、格納していない楽曲をも所持楽曲とする。端末間の楽曲の重複は 1 曲もなかった。実験を行う前に各被験者は一ヶ月間、我々が用意したアプリケーションを用いてこれらの楽曲を日常的に聴取してもらい、楽曲に対して再生回数の付与を行った。

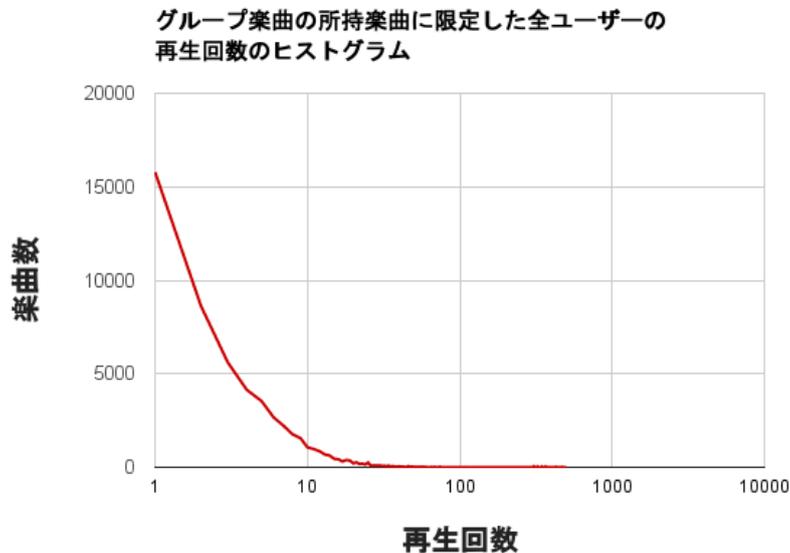


図 5.7: グループ楽曲を対象とした、全ユーザーの所持楽曲の再生回数に対する楽曲数のヒストグラム

次に、被験者の負担を考慮し、2時間程度で実験が終わるよう 287 曲の中から 28 曲を再生し、被験者は再生された楽曲に対して次の設問に 1～5 の 5 段階評価で回答する。以降、このアンケートによる嗜好評価をアンケート嗜好評価と呼称する。28 楽曲に対するアンケート嗜好評価の結果と詳細を表 5.5、表 5.6、表 5.7 に示す。また、被験者に用意した CD の一覧を本稿の末尾の「付録 A・被験者実験で用いた CD 一覧」に示す。

Q1 再生された楽曲は好みの曲であるか (アンケート嗜好評価)

5 好みであり、ぜひ CD や MP3 などの音源を購入したい。

4 無料であればぜひ聴きたい。

3 自分から選曲しようとは思わないが、流れてきたら聴き入ってしまいそうだ。

2 嫌いでは無いが、積極的に聴こうとは思わない。

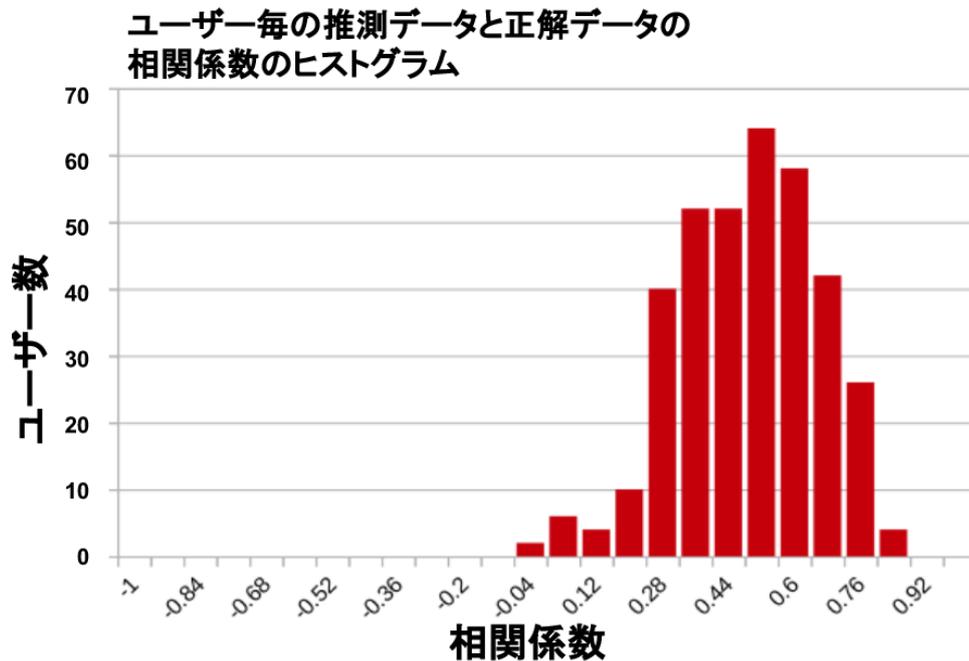


図 5.8: 推測した嗜好評価と正解嗜好評価のユーザー毎の相関係数のヒストグラム

1 好みでは無い。

5.3.2 実験結果

式 (3.1)、(3.2) を用いて再生回数を期待度へ正規化し提案システムを用いて、被験者データセットについて被験者 A、B、C の未所持楽曲である楽曲の期待度を推測する。式 (3.1)、(3.2) について α 、 β 値を変化させた時のアンケート嗜好評価との相関係数を表 5.8 に示す。表 5.8 から、推測した期待度とアンケート嗜好評価の相関係数が高かった被験者は、C であった。次に B、A の順である。ユーザーによって相関関係が異なるが、式 (3.1)、(3.2) の α 、 β 値の変更における変化は大きく現れなかった。

この実験では、5.2 節のような正解データの期待度と、推測データの期待度の比

較ではなく、5段階嗜好評価と推測した期待度との関係を調査している為、平均二乗誤差についての検証は、行っていない。

今後の課題として今後のより詳しい分析を行うために、共通の楽曲を所持しているユーザーグループを生成する際に、グループ所持楽曲について、再生回数の多い楽曲を含ませ、本研究の推薦手法が有効であるかの検証すべきである。

5.4 アプリケーションの動作確認実験

我々は、提案した選曲手法を用いて生成した楽曲一覧に沿って、楽曲の再生を行う端末が自動で Bluetooth スピーカーへ接続し、再生、そして切断を行うことで、同一のスピーカーから楽曲を自動再生するようなアプリケーションの作成を行った。Android 端末を 3 代用いて我々が作成したアプリケーションを用いて自動再生を 1 時間動作したところ、アプリケーションに問題はなかった。今後の課題として、アプリケーションの操作性を考慮した、ユーザービリティ評価を行う必要がある。

表 5.5: 被験者データセット:被験者3人の楽曲に対するアンケート嗜好評価

曲番号	曲名	アーティスト	B	C	A
1 曲目	虹	いきものがかり	4	3	3
2 曲目	サウダージ	ポルノグラフィティ	4	3	3
3 曲目	もしも運命の人がいるのなら	西野カナ	5	5	2
4 曲目	赤黄色の金木犀	フジファブリック	2	4	5
5 曲目	私以外私じゃないの	ゲスの極み乙女	5	5	1
6 曲目	ヒールに願いを	東京カランコロン	3	5	4
7 曲目	神様の暇つぶし	GO!GO!7188	2	3	5
8 曲目	シュガーソングとビターステップ	UNISON SQUARE GARDEN	5	5	1
9 曲目	GUT!	嵐	5	4	3
10 曲目	Cry OUT	ONE OK ROCK	5	5	2
11 曲目	なんでもねだり	KANA-BOON	4	5	2
12 曲目	ええねん	ウルフルズ	5	2	5
14 曲目	F	マキシマムザホルモン	5	5	5
15 曲目	別世界	UVERworld	3	4	2
16 曲目	ドタン場でキャンセル	GO!GO!7188	3	2	5
17 曲目	誰も知らない	嵐	5	3	2
18 曲目	ヴォイス	ポルノグラフィティ	5	4	5
19 曲目	不思議	スピッツ	3	2	5
20 曲目	0 CHOIR	UVERworld	2	5	1
21 曲目	ヒトリノ夜	ポルノグラフィティ	5	5	3
22 曲目	Asterisk	嵐	5	3	2
23 曲目	相愛	ウルフルズ	2	1	4
24 曲目	マイステージ	いきものがかり	3	2	3
25 曲目		東京カランコロン	2	5	2
26 曲目	ビューティフルデイズ	SPYAIR	4	5	3
27 曲目	考え事	GO!GO!7188	3	1	5
28 曲目	現状ディストラクション	SPYAIR	5	5	4

表 5.6: 被験者データセット:被験者 3 人のアンケート嗜好評価の詳細

ユーザー	未所持楽曲に対する評価数	未所持楽曲に対する評価平均	所持楽曲に対する評価数	所持楽曲に対する評価平均
A	21	3.25	7	4.77
B	17	2.70	11	4.40
C	18	4.21	10	5.00

表 5.7: 被験者データセット:被験者 3 人再生履歴の詳細

ユーザー	所有曲数	合計再生回数	最大再生回数
A	130	565	29
B	84	216	26
C	65	578	18

表 5.8: 被験者データセット:被験者 3 人の推測した未所持楽曲に対する期待度とアンケート嗜好評価の相関係数

ユーザー	所有曲数	未所持楽曲に対する評価数	式 (3.1)			
			$\alpha = 0.01$	$\alpha = 0.10$	$\alpha = 0.20$	$\alpha = 0.35$
A	130	16	-0.05870	-0.05872	-0.05894	-0.05915
B	84	13	0.27332	0.27089	0.26802	0.26500
C	65	16	0.44797	0.45136	0.45470	0.45759
ユーザー	所有曲数	未所持楽曲に対する評価数	式 (3.2)			
			$\beta = 0.01$	$\beta = 0.10$	$\beta = 0.20$	$\beta = 0.35$
A	130	16	-0.05636	-0.05723	-0.05758	-0.05765
B	84	13	0.25232	0.25189	0.25162	0.251032
C	65	16	0.42575	0.42902	0.43109	0.43155

第6章 結 論

本研究では、複数人が所有する端末から楽曲を再生するような、同一の音楽を聴くシチュエーションに着目し、そのような場面における選曲手法の提案と、手法を実現するアプリケーションの開発をした。

我々の作成したアプリケーションでは、各自の楽曲に対する好きな度合いである期待度を統合し、全員の嗜好を考慮した選曲を行っている。想定しているシチュエーションでは、とあるユーザーの所有する端末から楽曲が再生されるが、その楽曲に対する各自の嗜好を示す明示的な評価が存在しない場合がある。そこで我々は、各自の楽曲に対する期待度を端末に記録されている楽曲情報と行動履歴を用いて推測し、楽曲一覧を生成・再生するアプリケーションを作成した。

実験では、期待度の推測結果について検証を行った結果、ユーザーの再生回数から決定した期待度と推測した期待度に関して、正の相関関係が現れた。この結果から、ユーザーの楽曲に対する期待度について、端末内に記録されている楽曲情報と再生回数を用いた推測が有効であることが分かった。しかし、本実験において、ユーザーの楽曲に対する再生回数が、主に1~15回である楽曲でしか検証が行われていない。そのため今後の検証において、再生回数が1~15回以上の楽曲についての期待度の推測について検討を行う必要がある。また提案システムを用いて選曲した楽曲一覧について現状では、統合した期待度が高い順番に再生を行っている。今後は、楽曲のリズムやテンポを用いて、楽曲の遷移がスムーズになるような曲順を考慮することや、端末から再生したい楽曲、また再生したくない楽曲をユーザーが選択し楽曲一覧に反映されるようなプレイリストについて検

討も行うべきである。

作成したアプリケーションについては、提案手法を実現する為に、複数人の所有する端末を Bluetooth を用いて相互に接続を行い、提案手法を用いて選曲した楽曲一覧に沿って、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。しかしアプリケーションのユーザビリティ評価について実験を行えていない。そのため今後の課題として、アプリケーションについて利便性を追求する必要がある。

参考文献

- [1] Gordon Reynolds, Dan Barry, Ted Burke, and Eugene Coyle: “Towards a Personal Automatic Music Playlist Generation Algorithm: the need for Contextual Information”, DIT, 2007.
- [2] Arthur Flexer, Dominik Schnitzer, Martin Gassar, and Garhard Widmer: “Playlist Generation Using Start and End Songs” , Proc. ISMIR, pp. 173–178, 2008.
- [3] Elias Pampalk, Tim Pohle, and Garhard Widmer: “Dynamic Playlist Generation Based on Skipping Behavior” , Proc. ISMIR, pp. 634–637, 2005.
- [4] 黒瀬 崇弘, 梶川 嘉延, 野村 康雄: “視聴履歴を用いた楽曲推薦システム”, HIP2003-84, pp. 41–44, 2003.
- [5] 松本 大希, 滝口 恭平, 小高 大典: “複数人が共有する場のためのBGM選曲手法に関する研究”, 日本大学卒業論文, 2012.
- [6] A. Jameson: “ More than the Sum of its Members: Challenges for Group Recommender System ”, Proc AVI '04, ACM, New York, pp.48–54, 2004.
- [7] A. Crossen, J. Budzik, and K. Hammond: “Flytrap: intelligent group music recommendation”, Proc IUI, pp. 184–185, 2002.

- [8] F. Joseph, D. Theodore: “MUSICFX: An Arbiter of Group Preference for Computer Supported Collaborative Workouts”, Proc CSCW, ACM, New York, pp. 363–372, 1998.
- [9] Irene Ntoutsis and Kostas Stefanidis: gRecs: “A group recommendation system based on user clustering”, Proc. DASFAA, South Korea, pp. 299–303, 2012.
- [10] Gorla, J., Lathia, N., Robertson, S., and Wang, J: “Probabilistic Group Recommendation via Information Matching”, Proc WWW, pp. 495–504, Brazil, 2013.
- [11] 面白法人カヤック: FESS,

<http://fess.kayac.com/>
- [12] Yuuki Furuyama: Listen with,

<https://itunes.apple.com/jp/app/listen-with/id479816778?mt=8>
- [13] E. Pampalk: “Computational Models of Music Similarity and their Application”, Proc. MIR, Austria, 2006.
- [14] Android(accessed 2017. 1),

<https://www.android.com/>
- [15] Last.fm(accessed 2017. 1),

<http://www.lastfm.jp/api>
- [16] 7digital(accessed 2017. 1),

<https://www.7digital.com/>

- [17] Last.fm datasets (accessed 2017. 1),

<http://ocelma.net/>

- [18] G. Bonnin and D. Jannach: “Automated Generation of Music Playlists: Survey and Experiments”, Proc. ACM, Article No. 26, 2015.

- [19] K. Seyerlehner, P. Knees, D. Schnizer, G. Widmer: “Browsing Music Recommendation Networks”, Proc. ISMIR, pp. 129–134, 2009.

- [20] M. A. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney: “Content-Based Music Information Retrieval: Current Directions and Future Challenges”, Proc. IEEE, Volume:96, Issue:4, pp. 668–696, 2008.

謝 辞

本論文を作成するにあたり、北原鉄朗准教授から、丁寧かつ熱心なご指導を賜りました。また、北原研究室のメンバーとは苦楽を共にし、時には切磋琢磨し、成長を共にしました。特に、同学年である栗原拓哉氏とは、お互いが研究が行き詰まった時には議論をし、様々な助言をいただきました。また、本論文の審査に当たり、副査を担当してくださったまた尾崎知伸准教授と宮田章裕准教授には、本質的な鋭いご指摘やご意見を頂きました。他にも様々な人々の助けがあり、研究を進めることができました。実験の際に被験者を快く引き受けてくださった被験者の皆様、日常の議論を通じて多くの知識や示唆を頂いた北原研究室の皆様感謝します。

付 録 A 被験者実験で用いた CD 一 覧

本稿 4.2 節で述べた、被験者データセットに含まれる各被験者に用意した楽曲の CD のタイトルとアーティストを示したものである。

表 A.1: 被験者データセットに含まれる 3 人の被験者に用意した CD・アーティスト一覧

被験者	アーティスト	CD タイトル
A	チリヌルヲワカ	あ可よろし
	GO!GO!7188	ベスト・オブ・ゴー!ゴー!
	LOVE PSYCHEDELICO	This is LOVE PSYCHEDELICO U.S.BEST
	風味堂	風味堂 3
	森山直太郎	傑作撰 2001-2005
	スピッツ	CYCLE HIT 1991-1997 Spitz Complete Single Collection
	ウルフルズ	ベストやねん
	松任谷由実	松任谷由実 40 周年記念ベストアルバム 日本の恋と、ユーミンと。
	フジファブリック	SINGLES 2004-2009
	鬼束ちひろ	the ultimate collection
B	UVERworld	0 CHOIR
	嵐	THE DIGITALIAN
	NEWS	White
	KAT-TUN	KISS KISS KISS
	Acid Black Cherry	L-エル-
	いきものがかり	FUN! FUN! FANFARE!
	ONE OK ROCK	35xxxv
	ももいろクローバーZ	Z 伝説 - 終わりなき革命
	ポルノグラフィティ	PORNOGRAFFITTI 15th Anniversary " ALL TIME SINGLES
	ONEREPUBLIC	Native
C	KANA-BOON	なんでもねだり
	ゲスの極み乙女	私以外私じゃないの
	WEAVER	くちづけ diamond
	MAN WITH A MISSION	Seven Deadly Sins
	UNISON SQUARE GARDEN	シュガーソングとビターステップ
	西野カナ	もしも運命の人がいるのなら
	マキシマムザホルモン	予襲復讐
	UVERworld	LAST
	SPYAIR	BEST
	東京カランコロン	UTUTU