

3N-5

大量の音楽プレイリストに基づく楽曲推薦システムの試作

園田 亮[†] 伊東 栄典^{††} 池田 大輔[‡] 竇 ギョクホウ^{‡‡} 笠原 義晃^{‡‡}

[†]九州大学理学部物理学科情報理学コース ^{††}九州大学情報基盤センター [‡]九州大学システム情報科学研究院
情報理学部門 ^{‡‡}日立コミュニケーションズテクノロジー ES 事業部

1 はじめに

音楽のプレイリストでは、個人の嗜好に基づいて楽曲が選択される。個人は意識無意識に関わらず、何らかのテーマに沿って楽曲を選択していると考えられ、そのためプレイリスト内の楽曲間には何らかの関係があると仮定できる。この仮定に基づけば、大量のプレイリストから類似楽曲を発見可能である。

本研究では、プレイリスト群での楽曲およびアーティストの共起関係から類似関係を推定し、それに基づく音楽推薦の手法を提案する。提案手法では、協調フィルタリングでは推薦されにくいマイナーな部分の未だ知らない曲の推薦を行うことが可能であると考えられる。また、約25万個のプレイリストに基づく音楽推薦システムも開発した。

2 プレイリスト

実験のために Apple 社の提供するプレイリストの公開サイト iMix から 312,013 個のプレイリストを入手した。そのうちデータの存在しないなど無効なものを除いた 261,853 個のプレイリストについて分析を行った。iMix のプレイリストでは曲、アーティスト、ジャンルなどへ Apple 社が一意的 ID が割り付けている。261,853 プレイリスト内の一意な曲数は 396,595 曲であった。261,853 プレイリスト内の一意なアーティスト数は 70,905 であった。

iMix の提供するプレイリストは XML の形式をしている。XML の解析によりプレイリストの中からアーティスト名・ID、曲名・ID を抽出した。プレイリストには他にもジャンル ID や、発売された日、値段等の詳細なデータが含まれている。

3 頻度解析

今回の分析では 5~20 曲の曲数を含む 165,309 プレイリストのみを対象に分析を行った。これは全体の 60% にあたる。同時に含む曲が少なすぎるプレイリストはテーマに沿って曲を選んでいるとは考えにいため除外した。一方、曲が多すぎるプレイリストはテーマに

対する曲の選別が甘く、かつ共起する組合せ爆発的に増加するので除外した。

165,309 プレイリスト内の一意な曲数は 230,837 曲で一意的アーティスト数は 42,944 であった。音楽の推薦を行うため前準備のため、曲 ID と曲名、アーティスト ID とアーティスト名、曲 ID とアーティスト ID の対応表を作成した。

3.1 TF (単語頻度), DF (文書頻度)

まず、プレイリストを文書と見立て、曲とアーティストについて、 TF と DF を数えた。通常文書ファイル群を扱う場合、 $DF(w)$ と $TF(w)$ は同じではない、しかしプレイリストの曲については同じとなった。これは、同じプレイリスト内に同じ曲を 2 曲以上入れている人はいないことを意味する。一方アーティストでは $DF(w)$ と $TF(w)$ は異なる値をとった。

図 1 に各曲の FR (頻度, 順位) グラフを示す。X 軸 Y 軸はいずれも対数尺度で示している。また表 1 に上位の 10 曲を示す。

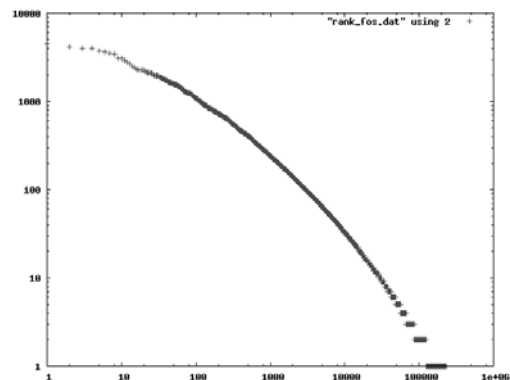


図 1: 曲の FR グラフ。X:出現頻度順位, Y:出現頻度

次に、各曲および各アーティストの共起を調査した。アーティストについては、全ての共起するアーティストの対と、その共起頻度を調べた。曲についても共起の調査を試みた。しかし、曲の場合は共起する組み合わせが多くなりすぎ、解析に用いた PC のメモリに収まらなかった。そのため共起頻度が低くなる出現頻度の低い曲 (10 回未満) を切り捨てて共起頻度を調査した。このときの一意な曲数は 31,060 曲であった。

A Group management system for secure distributed collaborative works

[†] Ryo SONODA (ryo.sonoda@i.kyushu-u.ac.jp)

^{††} Eisuke ITOH (itou@cc.kyushu-u.ac.jp)

[‡] Daisuke IKEDA (daisuke@i.kyushu-u.ac.jp)

^{‡‡} Yufeng DOU

⁰ Yoshiaki KASAHARA

表 1: 曲ランキング

順位	頻度	曲目
1	4836	Boulevard of Broken Dreams
2	4471	Ocean Avenue
3	4132	The Reason
4	4010	She Will Be Loved
5	3944	One, Two Step
6	3700	This Love
7	3646	Mr. Brightside
8	3499	Let's Get It Started
9	3434	Float On
10	3093	American Idiot

4 楽曲推薦の手法

4.1 コサイン類似度

Amazon.com [3] では顧客 A と顧客 B の類似度を A, B それぞれをベクトルとして次の式で表している。これは協調フィルタリングと呼ばれる手法で用いられている。

$$\text{similarity}(\vec{A}, \vec{B}) = \cos(\vec{A}, \vec{B}) = \frac{\vec{A} \cdot \vec{B}}{\|\vec{A}\| * \|\vec{B}\|}$$

上の式を適用するために、曲 A と曲 B (アーティスト A とアーティスト B) の各プレイリストでの出現の有無についてのベクトルを考える。すなわちプレイリスト 1 ~ n のうちで、プレイリスト i に曲 A が出現したか否かを 1, 0 で表現する。A についてのベクトルは次のように表現できる。

$$\vec{A} = (a_1, a_2, a_3, \dots, a_n), \text{ただし } a_i = \{0, 1\}.$$

曲 A と曲 B のコサイン類似度は以下のように計算できる。

$$\begin{aligned} \cos(\vec{A}, \vec{B}) &= \frac{a_1 * b_1 + a_2 * b_2 + \dots + a_n * b_n}{\sqrt{a_1^2 + a_2^2 + \dots + a_n^2} * \sqrt{b_1^2 + b_2^2 + \dots + b_n^2}} \\ &= \frac{Co(A, B, D)}{\sqrt{DF(A, D)} * \sqrt{DF(B, D)}} \end{aligned}$$

ここで $Co(A, B, D)$ とは文書集合 D 内で A と B が共起した回数である。 $\cos(\vec{A}, \vec{B})$ は 0 ~ 1 までの値を取り、1 に近いほど A と B の類似度が高い。

4.2 推薦手法の提案

我々は既に、共起頻度と出現頻度を基にした推薦システムを試作している [5]。あるアーティスト名 x を入力すると、 x と共起する頻度順に x に近いアーティスト名が表示される。 x と共起するアーティスト群を、 x より出現頻度の高いものと低いもので分ける。 x に近い順に 2 つのランキング表示を行うことで推薦を行う。

現在、新たな推薦のシステムの構築について検討している。コサイン類似度は全体での各曲/アーティスト

トの出現回数を考慮したものである。これを用いることで、マイナーな部分についても推薦ができると思われる。そこで、調べたい曲/アーティストとコサイン類似度が高い順に候補を表示するシステムを開発している。

調べたい曲/アーティスト名と共起する曲/アーティストとは何らかの関係があり、さらに共起した曲/アーティストとは元の曲/アーティストともなんらかの関係があると考えられる。この関係を用いた推薦を考えている。この場合、既知ではない曲/アーティストが推薦される可能性が高くなると考えている。現状では、共起頻度、出現頻度、コサイン類似度を用いて曲と曲 (アーティストとアーティスト) 間の距離を何らかの方法で測りそれを基に推薦を行うことを考えている。

5 おわりに

我々は、プレイリスト群での楽曲およびアーティストの共起関係から類似関係を推定し、それに基づいた音楽推薦の手法についての研究を行なっている。本稿では収集したプレイリストの頻度分析の結果と、それを用いた類似度の計算法について紹介した。曲の共起分析では、計算機の性能から多くの曲を切り捨てている。今後、解析手法を改良することで全曲についての共起頻度を調査したい。

また、新たな推薦システムの構想についても記述している。今後は推薦手法の詳細化を行なう予定である。また、具体的な推薦システムを実装し、その結果から提案した推薦手法の評価を行う予定である。

参考文献

- [1] Upendra Shardanand, Pattie Maes: "Social Information Filtering Algorithms for Automating "Word of Mouth"," Human Factors in Computer Systems CHI'95 Conference Proceedings, pp.210 - 217, 1994.
- [2] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, John Riedl: "GroupLens: An Open Architecture for Collaborative Filtering of Netnews," Proc. of The Conf. on Computer Supported Cooperative Work, pp.175-186, 1994.
- [3] Greg Linden, Brent Smith, Jeremy York: "Amazon.com Industry Report Amazon.com Recommendations",
- [4] 廣川佐千男, 伊東栄典, 下司義寛, Yufeng Dou, 池田大輔: "プレイリストからの曲目やアーティストの相互関連抽出," 情報研報, 第 81 回情報学基礎研究会, pp.??-??, Nov, 2005.
- [5] Yufeng Dou, Eisuke Itoh, Sachio Hirokawa, Daisuke Ikeda: "An Approach to Analyzing Correlation between Songs/Artists Using iTunes Playlists", Proc. IAWTIC'2005, vol.1, pp.951-956, Nov., 2005.
- [6] 北研二, 津田和彦, 獅子堀正幹, "情報検索アルゴリズム", 共立出版, 2002.(ISBN:4-320-12036-1)