

Linked Data を用いた音楽キュレーター推薦

北矢 光紀[†] 黄 宏軒^{††} 川越 恭二^{††}

[†] 立命館大学大学院 理工学研究科 〒 525-8577 滋賀県草津市野路東 1-1-1

^{††} 立命館大学 情報理工学部 〒 525-8577 滋賀県草津市野路東 1-1-1

E-mail: [†]kitaya@coms.ics.ritsumeai.ac.jp, ^{††}huang@fc.ritsumeai.ac.jp, ^{†††}kawagoe@is.ritsumeai.ac.jp

あらまし 近年, Web 等のインターネットサービス上で, 音楽キュレーターと呼ぶ, 自らコンテンツを収集し音楽の評論を行う利用者が増加している. 同時に, 音楽キュレーターの評論を読む利用者也増加している. しかし, キュレーターの急激な増加により, 利用者の嗜好に合った音楽キュレーターを探すのが困難である. 本論文では, キュレーターの評論や音楽関連情報等の音楽データ同士がリンクされた Linked Data を考え, その Linked Data 上での意味的距離を用いることで利用者の嗜好に合った音楽キュレーターを推薦する手法を提案する. 具体的には, Passant が提案した Linked Data 意味的距離を, 利用者の楽曲視聴履歴, 音楽キュレーターの楽曲評論, 楽曲情報データにより形成されたリンク構造に適用し改良する. この改良距離を用いた方法によって, 利用者は自分の視聴楽曲を考慮した音楽キュレーターの発見が可能となる.

キーワード Linked Data, セマンティックウェブ, キュレーション, 推薦

Music Curator Recommendation on Linked Data

Koki KITAYA[†], Hung-Hsuan HUANG^{††}, and Kyoji KAWAGOE^{††}

[†] Graduate School of Science and Engineering, Ritsumeikan University
Nojihigashi1-1-1, Kusatsu, Shiga, 525-8577 Japan

^{††} Graduate School of Science and Engineering, Ritsumeikan University
Nojihigashi1-1-1, Kusatsu, Shiga, 525-8577 Japan

E-mail: [†]kitaya@coms.ics.ritsumeai.ac.jp, ^{††}huang@fc.ritsumeai.ac.jp, ^{†††}kawagoe@is.ritsumeai.ac.jp

Abstract Recently, the music curator who criticize music is increasing. It is difficult to discover the music curator suitable for the user's taste in connection with it. People who collect contents by human power and create criticism are called the curator. On the other hand, Linked Data which is the big network structure to linked data exists. By the LDS algorithm which used Linked Data, Passant calculated the distance between music. In this paper, we propose the method of recommending the music curator who suited the user's taste using Linked Data. Link structure is formed using the listening history of user's music, music curator's musical criticism data, and music information data. And we calculate the distance between user and curator using the Linked Data.

Key words Linked Data, Semantic Web, Curation, Recommendation

1. はじめに

最近の急速なインターネットの普及により, 利用者は膨大な情報に容易にアクセス可能となった. 膨大な情報の中から所望の情報を取得するために検索サービスが活用されているが, 最近, コンテンツを取捨選択して Web で公開するキュレーターと呼ぶ評論を行う専門的知識を持つ利用者が出現している. キュレーターは, 人力でコンテンツを探索し選択し評論を行っている. キュレーターが評論した情報を利用者が読むことで, 利用者はその評論結果からコンテンツにアクセスすることができ

る. キュレーターを経由してコンテンツを利用者が取得する利用形態は, 従来の検索サービスに加えて今後増加すると予想されている [1]. しかし, 多数のキュレーターの中から各利用者の嗜好に合ったキュレーターを探し出すのは手間のかかることである. キュレーター検索サービスは現時点では存在しない. しかし, 音楽などのマルチメディアコンテンツを対象とするキュレーターをキーワード検索だけで見つけ出したとしても, そのキュレーターが評論した文章の中にたまたまそのキーワードが記述され利用者に結果として提供されるような状況が発生する可能性がある. すなわち, 利用者にとって利用者の嗜好に合っ

適切なキュレーターを提供することが重要である。

一方、インターネットで公開されている膨大な情報を個々に互いにリンクしあうことで多様で膨大な情報を共有する考え方 (Linked Data) が急速に定着しつつある。Linked Data はデータ同士がリンクされた超大規模情報のネットワークであり、2011年9月の時点で、音楽情報、地理情報、学術情報、政府情報、写真情報など、295のデータセットによるネットワークが形成されている。代表例として DBpedia^(注1) や MusicBrainz^(注2) がある。Linked Data を使用することで、異なるデータの間の意味的距離を求めることが可能となり、すでに Passant らの研究例が発表されている。

本稿では、Linked Data [2] を用いて、利用者に対して自分の嗜好と類似している音楽キュレーターを推薦する新たな方法を提案する。この方法を用いた推薦システムにより、利用者は嗜好の合った音楽キュレーターを短時間で発見して音楽評論の閲覧を行うことができる。また、従来の楽曲推薦とは違い、音楽キュレーターを推薦して継続的に閲覧することで、今後、音楽キュレーターの新たな評論を閲覧できる、新たな音楽の発見できると考える。

2. Linked Data を用いた音楽キュレーター推薦

本章では、本研究にて用いる Linked Data を用いた音楽キュレーター推薦について述べる。

2.1 Linked Data と意味的距離

Linked Data を用いて様々なデータ間の意味的な距離を求める研究がおこなわれている [3] [4] [5]。具体的には、任意のリンクしたデータセット上で意味的距離を計算する LDS (Linked Data Semantic Distance) [6] が提案されている。LDS は [0,1] の値をとり、2つのデータ間の LDS が小さいほどこの2つのデータは非常に類似しているということを意味する。

今、ノードの集合 R とノード間のリンクの集合 I とリンクの型の集合 L から構成されるグラフ G (すなわち、 $G = (R, L, I)$) が与えられているとする。ひとつのリンク $i_k (i_k \in I)$ が開始ノード $r_s (r_s \in R)$ と終了ノード $r_e (r_e \in R)$ およびリンク型 $l_j (l_j \in L)$ であるとき、 $i_k = \langle l_j, r_s, r_e \rangle$ と記述する。このとき、ある2つのノード $r_a, r_b (r_a, r_b \in R)$ 間の意味的距離 $LDS(r_a, r_b)$ は次の式で定義される [6]。

$$LDS(r_a, r_b) = \frac{1}{1 + \alpha + \beta + \gamma + \delta} \quad (1)$$

$$\begin{cases} \alpha = \sum_i \frac{C_d(l_i, r_a, r_b)}{1 + \log(C_d(l_i, r_a, r_b))} \\ \beta = \sum_i \frac{C_d(l_i, r_b, r_a)}{1 + \log(C_d(l_i, r_b, r_a))} \\ \gamma = \sum_i \frac{C_{ii}(l_i, r_a, r_b)}{1 + \log(C_{ii}(l_i, r_a, r_b))} \\ \delta = \sum_i \frac{C_{io}(l_i, r_a, r_b)}{1 + \log(C_{io}(l_i, r_a, r_b))} \end{cases}$$

ここで、 $C_d(l_i, r_a, r_b)$ は、 r_a から r_b にリンク型 l_i のリンクが存在するときに1の値をとり、それ以外の場合には0となる関数である。また、 $C_{ii}(l_i, r_a, r_b)$ は、 $\langle l_i, x, r_a \rangle$ と $\langle l_i, x, r_b \rangle$

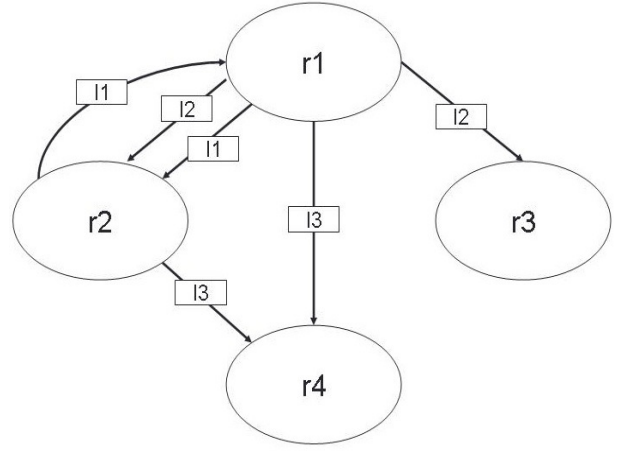


図1 音楽 Linked Data を用いた例

の2つのリンクとノード x が存在するときに1の値を持ち、それ以外は0となる関数である。 $C_{io}(l_i, r_a, r_b)$ は、 $C_{ii}(l_i, r_a, r_b)$ とは逆に、 $\langle l_i, r_a, x \rangle$ と $\langle l_i, r_b, x \rangle$ の2つのリンクとノード x が存在するときに1の値を持ち、それ以外は0となる関数である。また、 $C_d(l_i, r_a, n)$ は r_a からリンク型 l_i でリンクされているノード数である。同様に、 $C_{io}(l_i, r_a, n)$ は $C_{ii}(l_i, r_a, n)$ は、 r_a から r_b へリンク型 l_i で間接的にリンクされているノード数を出力する関数である。

例えば、図1のようなグラフの場合に、 r_1 と r_2 との間の意味的距離 $LDS(r_1, r_2)$ は次のような値となる。

$$LDS(r_1, r_2) = \frac{1}{1 + \frac{1}{1 + \log 3} + \frac{1}{1 + \log 3}} = 0.42$$

2.2 Linked Data を用いた音楽推薦

Passant は、LDS を用いて音楽のアーティストとバンドに限定した Linked Data に適用した音楽推薦 dbrec を開発した [6] [7]。dbrec では、例えば Jonny Cash というアーティスト名を入力することで、このアーティストと他のアーティスト間の距離を LDS により計算することで、このアーティストとよく似たアーティストを求めることができる。dbrec では DBpedia から Artist と Band についてノードを抽出した。39000 以上からなる Artist ノードが構築された。さらに、これらのノード間に18種類のリンク型で合計1073077件のリンクを生成した Linked Data として推薦に使用した。他の Artist と直接リンクされていない Artist ノードが21211ノードもあり通常はアーティスト推薦の対象外となるが、LDS による間接リンクの考慮によってこのような孤立ノードも含めた推薦が可能となる点が、dbrec の特徴である。Linked Data を用いた推薦に関して、dbrec のようなアーティスト推薦以外の他の音楽関連資源を推薦する研究は、筆者らが調査した範囲ではこれまで報告されていない。

2.3 キュレーターとキュレーション

キュレーションとは、一般に、人力で良質なコンテンツを集めて、編集、関連付け、意味付けを行い、それを公開し、利用者と共有することである [8]。本稿では音楽推薦を対象としているため、音楽評論を作成・編集・公開している利用者を音楽

(注1) : <http://dbpedia.org/>

(注2) : <http://musicbrainz.org/>

キュレーターと定義する。

日本の主要なキュレーションサービスとして、Naver まとめ^(注3)が存在する。約 10 万人のキュレーター数が登録され、月間約 2 億 5 千万ページビューされている非常に総アクセスの多いサービスである。なお、本稿では、音楽評論を特化しているキュレーションサービス itunes ping^(注4)で評論を 10 本以上作成している利用者を音楽キュレーターと定義する。

今後は、急激にキュレーター数およびアクセス数が増加するものと考えられる。

2.4 Linked Data を用いた音楽キュレータ推薦

アーティストや楽曲を推薦する従来の音楽推薦と異なり、本研究では、Linked Data によって今後重要となるキュレータの推薦を行うことを目指す。従来の楽曲推薦ではなく、ユーザ、キュレーターに付加されたメタデータを利用してユーザにキュレーター推薦を行うために Linked Data を使用する。しかし、Linked Data を用いた音楽キュレータ推薦において Passant らが提案した LDSO を適用する場合に次のような問題点が存在する。

まず、最初の問題点を説明する。dbrec では特定アーティストに似たアーティストを提示するアーティスト推薦のみを實現しているが、同様の方法で特定キュレーターに似たキュレーターを提示する推薦は可能である。しかし、dbrec でも孤立アーティストは存在するが、キュレーターは各々独立して評論することが多いため、キュレーター間での直接リンクはほぼ皆無と予想できる。したがって、キュレーター間の間接リンクを中心に LDSO を改良することが重要となる。

次の問題点は、キュレーターが評論する対象はアーティストというよりも通常は大多数は楽曲であるという点から発生する問題である。利用者はキュレーターが評論した楽曲とその評論内容をみる行動を行う。しかし、キュレーター数は膨大であり、その中から自身の嗜好にあった特定のキュレーターを見つけ出すことは容易なことではない。すなわち、特定のキュレーターから類似のキュレーターを見つけ出すのではなく、利用者の嗜好にあったキュレーターを推薦する必要がある。このとき、どのような情報から利用者の嗜好に似たキュレーターを求めることができるかという点が重要となる。

本稿では、この 2 つの問題を解決するために、Linked Data のネットワーク構造を定義すると同時に LDSO の改良を行う。これにより、利用者の嗜好に似た音楽キュレーターの推薦を可能とする。なお、本稿では、次のような前提を設定する。1) 利用者は楽曲の視聴履歴データを Last.fm^(注5)に登録しておく。2) MusicBrainz で管理されている楽曲データを対象とする。3) 音楽キュレーターおよびその楽曲評論に関する情報はキュレーションサービス itunes ping から入手し、Linked Data のノードとして抽出する。

これらのデータは Linked Data として推薦に使用する。Linked Data の構造は次章で説明する。

3. 提案手法

本章では、Linked Data を用いた音楽キュレーター推薦について述べる。

3.1 基本的事項

利用者ノード集合 U を $U = \{U_1, U_2, \dots, U_n\}$ 、音楽キュレーターのノード集合 C を $C = \{C_1, C_2, \dots, C_n\}$ 、楽曲のノード集合 M を $M = \{M_1, M_2, \dots, M_n\}$ 、アーティストのノード集合 A を $A = \{A_1, A_2, \dots, A_n\}$ 、ジャンルのノード集合 J を $J = \{J_1, J_2, \dots, J_n\}$ 、アルバムのノード集合 X を $X = \{X_1, X_2, \dots, X_n\}$ 、リンク型集合 l を $l = \{l_1, l_2, \dots, l_n\}$ 、利用者 U_i 、音楽キュレーター C_j 、利用者と音楽キュレーター間のエッジ l_k のリンク関係を $\langle l_i, U_i, C_i \rangle$ とする。

$K_{ii}(l_i, a, b)$ は、2 章で述べたように、2 つのノード (a と b) 間に存在するリンク型 l_i による間接ノードの総数を出力する関数である。正確には、 $K_{ii}(l_i, a, b)$ は $\langle l_i, a, x \rangle$ と $\langle l_i, b, x \rangle$ となる間接ノード x の総数である。したがって、 $K_{ii}(l_i, U_j, C_k)$ は、利用者ノード $U_j (U_j \in U)$ からキュレーターノード $C_k (C_k \in C)$ へリンク型 $l_i (l_i \in L)$ で楽曲ノード $M_x (M_x \in M)$ を経由して辿り付くときの、楽曲ノード M_x の総数を出力する関数である。

一方、 $K_{ii}(l_i, U, C)$ は $\langle l_i, U_j, M_x \rangle$ と $\langle l_i, C_k, M_x \rangle$ となる間接楽曲ノード M_x の総数である。ここで、 $\langle l_i, U_j, M_x \rangle$ や $\langle l_i, C_k, M_x \rangle$ は、リンク型 l_i で、利用者ノード U_j から楽曲ノード M_x へのリンク、および、キュレーターノード C_k から楽曲ノード M へのリンクを表現している。これは、 $\langle l_i, U_j, M_x \rangle$ は利用者がリンク型 l_i で楽曲 M_x を視聴したこと、 $\langle l_i, C_k, M_x \rangle$ はキュレータ C_k がリンク型 l_i で楽曲 M_x を評論したことを意味している。

$K_{io}(l_i, U_i, C_i)$ とは楽曲ノード M_i からリンク型 l_i を経由して辿り着く利用者ノード U_i と、楽曲ノード M_i からリンク型 l_i を経由して辿り着くキュレーターノード C_i からの集合の総数を出力する関数である。

このとき、利用者 U_a と音楽キュレーター C_b の間の意味的距離を計算するために、2 章の ((1) 式で定義した LDSO [6] を使用すると、(2) に示す式となる。

$$dis(U_a, C_b) = \frac{1}{1 + \sum_i \frac{K_{ii}(l_i, U_a, C_b)}{1 + \log(K_{ii}(l_i, U_a, n))} + \sum_i \frac{K_{io}(l_i, U_a, C_b)}{1 + \log(K_{io}(l_i, U_a, n))}} \quad (2)$$

(1) 式から (2) 式を導出するのは、LDSO の定義式 (1) 式では、リンク型 l_i でリンクされているノード間を考慮していたが、本稿で対象とする Linked Data では、利用者ユーザノードとキュレーターノード間が直接リンク型 l_i でリンクされることはないためである。このことから、間接的にリンクされているノード間のみを考慮する。

(2) 式において利用者 U_a と音楽キュレーター C_b 間の距離が 0 に近いほど U_a と C_b は類似している。よって、(2) 式の計算結果から、距離が 0 に近い音楽キュレーターを利用者に推薦する。

(注3) : <http://matome.naver.jp/>

(注4) : <http://www.apple.com/itunes/ping/>

(注5) : <http://www.lastfm.jp/>

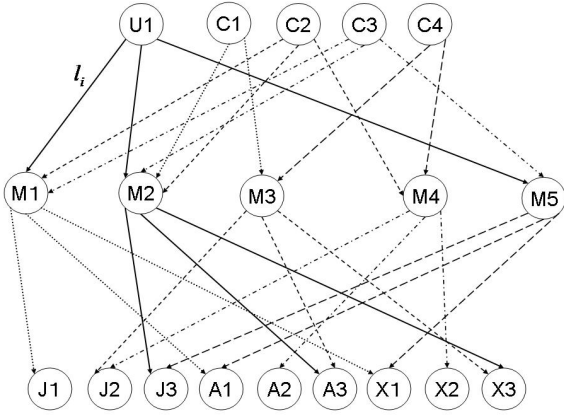


図2 音楽 Linked Data を用いた例

音楽 Linked Data の例を図2に示す。ここで、利用者 $U = \{U_1\}$ 、キュレータ $C = \{C_1, C_2, C_3, C_4\}$ 、楽曲 $M = \{M_1, M_2, M_3, M_4, M_5\}$ 、ジャンル $J = \{J_1, J_2, J_3\}$ 、アーティスト $A = \{A_1, A_2, A_3\}$ 、アルバム $X = \{X_1, X_2, X_3\}$ である。

以下に、LDSO を基本とする (2) 式による意味的距離を計算する方法に加えて、この方法を基本に他の情報を考慮した3種類の改良方法を提案する。

3.2 利用者と音楽キュレーターと楽曲を考慮したリンク構造

まず、Naïve な方法として、利用者と音楽キュレーターと楽曲をのみを考慮したリンク構造を用いて意味的距離を算出する方法を用いる。

図2に示すグラフの場合に、 U_1 と C_1, C_2, C_3, C_4 との間の意味的距離 $dis(U_1, C_1), dis(U_1, C_2), dis(U_1, C_3), dis(U_1, C_4)$ は次のような値となる。

本基本方法による利用者と音楽キュレータ間の意味的距離を算出する。

$$dis(U_1, C_1) = \frac{1}{1 + \frac{1}{\log 3}} = 0.60$$

$$dis(U_1, C_2) = \frac{1}{1 + \frac{2}{\log 3}} = 0.43$$

$$dis(U_1, C_3) = \frac{1}{1 + \frac{3}{\log 3}} = 0.33$$

$$dis(U_1, C_4) = \frac{1}{1 + \frac{0}{\log 3}} = 1$$

この例では、 $dis(U_1, C_3)$ のが最も小さい値となるため、利用者 U_1 に最も近いキュレータは C_3 となる。したがって、 U_1 にはまず C_3 が推薦されることとなる。

3.3 アーティスト、アルバム、ジャンルを含めたリンク構造 (改良方法1)

Naïve 方法では、利用者と音楽キュレーターと楽曲のみを考慮したリンク構造を扱っているが、これに加えて、ジャンル、アーティスト、アルバムをの3種類のノードに関するリンクを考慮する構造を考える。このとき、以下の (3) 式により、利用者 U_a と音楽キュレータ C_b との間の意味的距離を算出する (以降、本方法を改良方法1と呼ぶ)。このように、ジャンル、アーティスト、アルバムを考慮することで、より精度の高い推薦を

行うことができると考える。

$$dis(U_a, C_b) = \frac{\sqrt{\sum_{i=1}^n (U_i - \bar{U})} \sqrt{\sum_{i=1}^n (C_i - \bar{C})}}{\sum_{i=1}^n (U_i - \bar{U})(C_i - \bar{C})} \times \frac{1}{1 + \sum_i \frac{K_{ii}(l_i, U_a, C_b)}{1 + \log(K_{ii}(l_i, U_a, n))} + \sum_i \frac{K_{io}(l_i, U_a, C_b)}{1 + \log(K_{io}(l_i, U_a, n))}} \quad (3)$$

なお、 $\bar{U}, \bar{C}$ はそれぞれ相加平均 $\bar{U} = \sum_{i=1}^n (\frac{U_i}{n})$, $\bar{C} = \sum_{i=1}^n (\frac{C_i}{n})$ を示す。

図2の例を用いて、本改良方法1による利用者と音楽キュレータ間の意味的距離を算出する。 U_1 と C_1, C_2, C_3, C_4 との間の意味的距離 $dis(U_1, C_1), dis(U_1, C_2), dis(U_1, C_3), dis(U_1, C_4)$ は以下に示す値となる。

$$dis(U_1, C_1) = \frac{1}{1 + \frac{1}{\log 3}} \times \frac{3}{50} = 0.036$$

$$dis(U_1, C_2) = \frac{1}{1 + \frac{2}{\log 3}} \times \frac{1}{3} = 0.141$$

$$dis(U_1, C_3) = \frac{1}{1 + \frac{3}{\log 3}} \times \frac{1}{25} = 0.013$$

$$dis(U_1, C_4) = \frac{1}{1 + \frac{0}{\log 3}} \times \frac{3}{8} = 0.375$$

この例でも、Naïve 方法と同様に、 $dis(U_1, C_3)$ が最も小さい値となるため、利用者 U_1 に最も近いキュレータは C_3 となる。したがって、 U_1 にはまず C_3 が推薦されることとなる。

3.4 楽曲間で双方向リンクするリンク構造 (改良方法2)

この方式では、楽曲 M_a と楽曲 M_b がアーティスト、ジャンル、アルバムのリンク構造上、類似している場合、楽曲 M_a と楽曲 M_b 間で双方向にリンク型を付加する。楽曲間でリンク型を付加する理由として、楽曲間でアーティスト、ジャンル、アルバムのリンク構造上、類似している場合、2つの楽曲間の意味的距離が近いと考えられるためである。この方式によって、3.2節からさらに高精度の結果が得られると考えられる。

2つの楽曲 M_a, M_b の間の意味的距離 $dis(M_a, M_b)$ を (4) に示す。(4) 式の計算結果からあらかじめ設定した閾値 δ_1 を超える場合に、楽曲間の双方向にリンク型を付加する。次に、ノード間の長さを3以下にして、利用者と音楽キュレータ間の距離を (2) 式を用いて計算する。 $K_{ia}(l_i, U_i, C_i)$, $K_{ij}(l_i, U_i, C_i)$, $K_{ix}(l_i, U_i, C_i)$ はそれぞれ、アーティスト A_i 、ジャンル J_i 、アルバム X_i のノードからリンク型 l_i を経由して辿り着く楽曲ノード M_i 間の集合とする。

$$dis(M_a, M_b) = \frac{1}{1 + \alpha + \beta + \gamma} \begin{cases} \alpha = \sum_i \frac{K_{ia}(l_i, r_a, r_b)}{1 + \log(K_{ia}(l_i, r_a, n))} \\ \beta = \sum_i \frac{K_{ij}(l_i, r_a, r_b)}{1 + \log(K_{ij}(l_i, r_a, n))} \\ \gamma = \sum_i \frac{K_{ix}(l_i, r_a, r_b)}{1 + \log(K_{ix}(l_i, r_a, n))} \end{cases} \quad (4)$$

閾値を0.3として、図2のようなグラフの場合に計算すると、 U_1 と C_1, C_2, C_3, C_4 との間の意味的距離 $dis(U_1, C_1), dis(U_1, C_2), dis(U_1, C_3), dis(U_1, C_4)$ は以下に示す値となる。

$$\begin{aligned} dis(U_1, C_1) &= \frac{1}{1 + \frac{2}{\log 3}} = 0.43 \\ dis(U_1, C_2) &= \frac{1}{1 + \frac{2}{\log 3}} = 0.43 \\ dis(U_1, C_3) &= \frac{1}{1 + \frac{3}{\log 3}} = 0.33 \\ dis(U_1, C_4) &= \frac{1}{1 + \frac{1}{\log 3}} = 0.60 \end{aligned}$$

この例でも、Naïve 方法と同様に、 $dis(U_1, C_3)$ が最も小さい値となるため、利用者 U_1 に最も近いキュレータは C_3 となる。したがって、 U_1 にはまず C_3 が推薦されることとなる。

3.5 利用者と楽曲間をバーチャルリンクするリンク構造 (改良方法 3)

この方式では、利用者 U_n と M_a がリンク型でリンクしている、楽曲 M_a と楽曲 M_b がアーティスト、ジャンル、アルバムのリンク構造上、類似している場合に、利用者 U_n と楽曲 M_b 間はバーチャルリンクとしてリンク型を付加する。バーチャルリンクとしてリンク型を付加する理由として、2つの楽曲間の意味的距離が近い場合、楽曲 M_a とリンク型でリンクしている利用者 U_n は楽曲 M_b も同様、意味的距離が近いと考えられるためである。この方式によって、3.2節からさらに高精度の結果が得られると考えられる。

2つの楽曲間の意味的距離は改良方法2と同様の(4)式を用いる。(4)式の計算結果からあらかじめ設定した閾値 δ_2 を超える場合に、ユーザと楽曲間にリンク型を付加する。次に、(2)式を用いて、利用者と音楽キュレーター間の距離を(2)式を用いて計算する。

閾値を0.3として、図2のようなグラフの場合に、 U_1 と C_1, C_2, C_3, C_4 との間の意味的距離 $dis(U_1, C_1), dis(U_1, C_2), dis(U_1, C_3), dis(U_1, C_4)$ は次のような値となる。

$$\begin{aligned} dis(U_1, C_1) &= \frac{1}{1 + \frac{2}{\log 4}} = 0.17 \\ dis(U_1, C_2) &= \frac{1}{1 + \frac{2}{\log 4}} = 0.17 \\ dis(U_1, C_3) &= \frac{1}{1 + \frac{3}{\log 4}} = 0.13 \\ dis(U_1, C_4) &= \frac{1}{1 + \frac{1}{\log 4}} = 0.23 \end{aligned}$$

この例でも、Naïve 方法と同様に、 $dis(U_1, C_3)$ が最も小さい値となるため、利用者 U_1 に最も近いキュレーターは C_3 となる。したがって、 U_1 にはまず C_3 が推薦されることとなる。

4. 評価実験

本章では、本研究にて用いる Linked Data を用いた音楽キュレーター推薦の評価実験について述べる。

4.1 実験条件

本稿で提案した Linked Data を用いた音楽キュレーター推薦の有効性を確認するため、実データを用いて評価実験を行った。

ロック、ジャズ、J-POP といった複数のジャンルの楽曲を視聴している last.fm の利用者5名からなる利用者データを実験

表1 実験で利用した利用者データ

利用者名	再生楽曲数
U_1	153
U_2	104
U_3	130
U_4	89
U_5	215

で使用する利用者データとして用いた。実験で利用した使用者データを表1に示す。

次に、ロック、ジャズ、J-POP といった複数のジャンルの楽曲の評論を作成している itunes ping から音楽キュレーター20名を選び、実験で使用する音楽キュレーターとして用いた。実験で利用した利用者データを表2に示す。

上記の利用者と音楽キュレーターのデータを用いて、3章で説明した基本方法と改良した意味的距離を用いた3種類の改良方法により、利用者と音楽キュレーター間の類似度算出する。その結果、利用者と音楽キュレーター間の距離が0に近い音楽キュレーターを利用者に推薦する。

使用したテストデータの具体例を利用者 U_1 を用いて説明する。利用者 U_1 がリンクしている楽曲ノードの集合 M^1 は、楽曲 $M^1 = \{FirstLove, COLORS, InMyRoom, \dots, StayGold\}$ である。これにより、 U_1 と M^1 の要素の間にエッジが作成されている。また、 M^1 の楽曲がリンクしているジャンルノードの集合 J^1 、アーティストノードの集合 A^1 、アルバムノードの集合 X^1 は、各々、ジャンル $J^1 = \{J-POP, R \& B, ROCK, \dots, TECHNO\}$ 、アーティスト $A^1 = \{宇多田ヒカル, 倅田来未, 絢香, \dots, 安室奈美恵\}$ 、アルバム $X^1 = \{SPARK, FirstMessage, FirstLove, \dots, Fresh\}$ である。これにより、 M^1 のノードである個々の楽曲と J^1, A^1, X^1 の要素の間にエッジが作成されている。

ジャンルのノード数は8個であり、アルバムのノード数は112個、アーティストのノード数は46個である。また、利用者、キュレーターと楽曲間で生成したリンク数は927である。テストデータの総リンク数は1093である。なお、使用した閾値 δ_1 と δ_2 の値は、各々、0.3と0.3である。

実験の評価尺度として、提案手法である音楽キュレーター推薦の個々の方法を比較するために、情報検索の評価に頻繁に使用される再現率と適合率、およびF値を用いた。なお、正解集合は予め人手によって作成した。具体的には、表1に示す各利用者に対して、アーティストの視聴する割合、楽曲の視聴する割合、ジャンルの視聴する割合、アルバムを視聴する割合を考慮して、推薦する音楽キュレーターを正解として決定した。

4.2 実験結果ならびに考察

本提案手法の有効性を示すための評価実験の結果と考察をそれぞれ示す。本提案手法の音楽キュレーター推薦結果を表3、評価結果を表4に示す。

表4より、Naïve な方法と提案した3種類の改良方法のすべてで、再現率とF値は高いことがわかる。このように提案した改良手法が利用者の嗜好に合った音楽キュレーターを抽出する

表 2 実験で使った音楽キュレーターデータ

音楽キュレーター名	評論数
C_1	13
C_2	11
C_3	10
C_4	8
C_5	15
C_6	20
C_7	9
C_8	11
C_9	7
C_{10}	9
C_{11}	11
C_{12}	12
C_{13}	9
C_{14}	15
C_{15}	11
C_{16}	10
C_{17}	19
C_{18}	7
C_{19}	17
C_{20}	12

ことができた。提案した3種類の改善方法の中では、利用者と楽曲間をバーチャルリンクするリンク構造を使用した改良方法3が、再現率、適合率、F値のいずれも高い値を示した。これにより、利用者と楽曲間を実際にリンクするのではなく、類似度を算出するために仮想的なリンクを生成することで柔軟に音楽キュレータの検索が可能であることを示している。

その結果、利用者の嗜好に合った音楽キュレーターの見逃しを防ぐことができると考えられる。以上より、本提案手法は情報検索における評価より、Linked Dataを用いた音楽キュレーター推薦は有効であると言える。しかし、さらに、以下の2つの視点から検討することが必要である。

まず、用いたテストデータでは、対象とする音楽キュレーターが作成する評論の実データ数が少ないことである。その結果、全体的に適合率が低くなったと考える。今後、音楽キュレーター数を増加したテストデータによる詳細な実験が必要である。

次に、本提案手法では、利用者の楽曲視聴履歴データと音楽キュレータの評論データから意味的距離を算出したが、さらに利用者の嗜好に合った音楽キュレーターを推薦するために、楽曲視聴履歴データの視聴した日時と音楽キュレーターの評論データの作成日時を考慮する必要がある。そうすることによって、利用者の嗜好の変化にも対応できると考えられる。そのため、今後、楽曲視聴履歴データの視聴した日時と音楽キュレーターの評論データの作成日時も意味的距離算出式に取り入れるか検討する。さらに、利用者毎に何を重視しているか本提案手法が考慮し、利用者毎に適した類似度算出式を導くようにする必要がある。そのため、今後は本提案手法をシステム化して実装していく必要がある。

表 3 音楽キュレーター推薦結果

利用者	基本方法	改良方法 1	改良方法 2	改良方法 3	正解のキュレーター
U_1	C_3	C_3	C_{18}	C_{18}	C_{18}
U_2	C_2	C_2	C_5	C_2	C_2
U_3	C_{17}	C_{17}	C_{17}	C_{17}	C_{17}
U_4	C_6	C_6	C_9	C_9	C_7
U_5	C_1	C_1	C_{15}	C_{15}	C_{15}

表 4 評価結果

	基本方法	改良方法 1	改良方法 2	改良方法 3
再現率	0.20	0.20	0.60	0.80
適合率	0.17	0.21	0.23	0.26
F 値	0.18	0.20	0.33	0.39

5. おわりに

本稿では、Linked Dataを用いた音楽キュレーター推薦を提案した。Linked Dataによって、利用者とキュレーター間の意味的距離を算出可能となった。その結果、人力で良質なコンテンツを集めて音楽の評論を行っている音楽キュレーターを利用者に推薦することが可能になった。

今後は、提案した3つの方式から大規模な実験を行い、より評価の高い方式を用いる。及び、利用者自身の評価も考慮して実験を行う。発展として、利用者の嗜好に類似している音楽キュレーターの評論を元に音楽プレイリストを作成する。プレイリストを作成することで、さらに効率的に楽曲を視聴できると考えられる。また、楽曲間の類似においてフーリエ変換などの音響的特徴からみられる類似度も考慮して、有用性を確認する。

文 献

- [1] R. Steven. Curation Nation: How to Win in a World Where Consumers are Creators, McGraw-Hill(2011)
- [2] "Linked data - connect distributed data across the web", <http://linkeddata.org/>.
- [3] Alexander Budanitsky and Graeme Hirst: Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures, Workshop on Wordnet and Other Lexical Resources, Second Meeting of The North American Chapter of The Association for Computational Linguistics(2001)
- [4] Hao Sheng, Huajun Chen, Tong Yu, Yelei Feng: Linked Data based Semantic, IEEE IRI2010, pp.104-108, (2010)
- [5] Victor Penela, Guillermo Alvaro, Carlos Ruiz, Carmen Cordoba, Francesco Carbone, Michelangelo: miKrow: Semantic Intra-enterprise Micro-Knowledge Management System, The Semantic Web: Research and Applications, Lecture Notes in Computer Science, 2011, Volume 6644/2011, 154-168, Similarity and Data Mining, IEEE IRI2010, pp.104-108, (2010)
- [6] A. Passant: "dbrec-Music Recommendations Using DBpedia", ISWC, pp.209-224, (2010)
- [7] A. Passant: "Measuring Semantic Distance on Linking Data and Using it for Resources Recommendations", AAAI, pp.93-98, (2010)
- [8] 佐々木俊尚. キュレーションの時代「つながり」の情報革命が始まる, 筑摩書房, (2011)
- [9] S. Debnath, N. Ganguly, and P. Mitra, Feature Weighting in Content Based Recommendation System Using Social Network Analysis, WWW'08, pp.1041-1042, (2008)
- [10] A. Passant, Stefan Decker: Hey! Ho! Let's Go! Ex-

planatory Music Recommendations with dbrec, ESWC(2),
pp.411-415, (2010)