

Multi Armed Bandit モデルに基づく エンドポイント探索支援システムの試作

A Preliminary Prototype System for Discovering LOD Endpoints
based on Multi Armed Bandit Model

稜野 寿章*¹ 福田 直樹*²
Yoshiaki Kadono Naoki Fukuta

*1*2 静岡大学大学院情報学研究科
Graduate School of Informatics, Shizuoka University

There are emerging needs to realize semantic-based heterogeneous data integration via Linked Open Data (LOD) technologies. Although users can obtain various data from multiple open data sources as LOD endpoints, it is theoretically a hard problem to effectively select endpoints without previously knowing what information would be contained in those endpoints. Therefore, it is necessary that to understand what contents is contained in each endpoint by SPARQL query to select proper one. In this paper, we present our preliminary prototype system that supports determining appropriate endpoints to be retrieved based on Multi Armed Bandit model in order to discovery useful endpoint on searching LOD.

1. はじめに

Linked Open Data(LOD)が急速の普及が進み、2014年3月現在、400を超える情報提供源(エンドポイントと呼ぶ)が公開されている*¹。利用者はエンドポイントに対し、規格化されたクエリ言語によるクエリを実行することによって、様々な情報を得ることができる。

エンドポイントがどのような問い合わせに対して有効なデータを持っているかを知ることは、スキーマを事前に決めてデータをテーブル形式で格納する関係データベースとは異なり、LODへの検索では必ずしも容易ではない[稜野 14]。エンドポイントの持つ外形的な情報のみからでは、無数にあるエンドポイントの中から適切なものを選ぶことは困難であり、なおかつ、LODの相互接続性という特性を考えれば、ユーザが得たい情報は必ずしも一つのエンドポイント内で完結するとは限らないため、複数のエンドポイントにまたがる検索(横断的検索)を効果的にかつ少ない負担で行えるようにすることが、重要な課題の1つとなる[Noguchi 13]。

情報源としてのエンドポイントが検索対象として特定の問い合わせにとって有益であるかどうかを判断するには、なんらかの別の事前の問い合わせによりその有益さを推定する必要がある。その予備的な事前の問い合わせを、対象が無数にある場合に行おうとすると、それぞれのエンドポイントやその通信路となるネットワークに大きな負荷がかかってしまうため、その効率化の必要がある。本研究では、これらの問題に対して、Multi-Armed Bandit モデルに基づく手法を用いてエンドポイント探索を支援するシステムを試作する。

2. 関連研究

2.1 SuPARQooL

SuPARQooL[Noguchi 13]は、語彙発見手法を用いて重み付きオントロジーマッピングに基づく SPARQL、及び SPARQLoid[Fujino 12][Fujino 13]クエリの作成支援システムである。このシステムにおけるクエリの作成支援機能の一

つに、エンドポイント検索支援が含まれている。SuPARQooLでは、必要なオントロジーやプロパティが不明な場合には、該当する箇所に URI を記載する代わりにキーワードを入れ、そのキーワードを基に探索対象となるエンドポイントがユーザにとってどれだけ有益かを推定するための評価クエリを生成し、評価クエリを実行して得られた結果によってユーザに推薦するエンドポイントを決定する。

SuPARQooLにおけるエンドポイント検索では、探索対象となるエンドポイント全てにキーワードの数と等しい評価クエリを実行するため、多数のエンドポイントが探索対象となる場合の効率化が課題となる。

2.2 Multi-Armed Bandit(MAB)

標準 MAB モデル [Robbins 52] では、プレイヤーが K 個のアームが付いたスロットマシンに対し、アームを実行するとそれぞれのアームに設定された独立かつ未知の分布に従った報酬が得られる中で、どのアームを選択するかを考える問題を扱う。この問題の目的は、得られる報酬の合計を最大化することであり、焦点は、最も良いアームを探しつつも可能な限り得られる報酬を増やさなければならない点にある。

この問題のモデルは、アーム数を K 、実行タイムステップ数を $T(> 0)$ 、各タイムステップを $t(= 1, 2, 3, \dots)$ 、 t に選択するアームを $i(t)$ 、 $i(t)$ を実行することで得られる $i(t)$ に設定された報酬分布に従う報酬を $r_{i(t)}(t)$ としたとき、MAB 問題は次の式を最適化する問題と定義される。

$$\max \sum_{t=1}^T r_{i(t)}(t) \quad (1)$$

この問題において、もし全てのアームに設定された報酬分布が既知であるならば、最適な方針は最も期待値の高いアームを引き続けることである。MAB 問題では、その方針を、アームの選択方針の性能を評価するために利用する。 μ_i をアーム i に設定された報酬の期待値としたとき、

$$\mu^* = \max_i \mu_i \quad (2)$$

とし、 T 回実行した後におけるアーム選択方針 A の機会損失

連絡先: 稜野 寿章, 静岡大学情報学研究科, 〒432-8011 静岡県浜松市中区城北 3-5-1, passcut2000@yahoo.co.jp

*1 <http://sparqls.okfn.org/>

$R^T(A)$ を,

$$R^T(A) = T\mu^* - \sum_{t=1}^T r_{i(t)}(t) \quad (3)$$

と定義する.

MAB問題は、情報収集と情報活用のトレードオフがある中で報酬を最大化することを考える。エンドポイント探索においても、評価クエリによる情報収集と情報活用のバランスを考慮することで効率化を図りたいため、MABモデルの利用を考えるが、エンドポイント探索においては、評価クエリの実行時間や、ネットワークへの負荷といったコストを考慮しなければならない。そこで本研究では、MABモデルに予算とコストの概念を追加した Budget-Limited MAB(BLMAB) モデル [Tran-Thanh 10][Tran-Thanh 12a] を利用する。BLMABモデルでは、各アームにコストが設定されており、アームを引くためにはそのコストを予算から支払わなければならない。プレイヤーは予算に収まるようにアームを引かなければならない。

アームを選択するアルゴリズムを A とし、 $N_i^A(B)$ を A によって決定されるアーム i の実行回数、 c_i をアーム i を実行するコストとしたとき、予算 B を超えてはならないという制約は次の式で表される。

$$P\left(\sum_i^K N_i^A(B)c_i \leq B\right) = 1 \quad (4)$$

3. 拡張 BLMAB モデルの導入

エンドポイント探索における BLMAB のモデルは、探索する際に生じるネットワークの負荷と、評価クエリの実行にかかる時間という二重のコストが存在するため、従来の BLMAB モデルに対し、時間の制約という概念を追加したモデルを導入する。

拡張 BLMAB のモデルでは、プレイヤーが K 個のアームを持つスロットマシンにおいて、1ステップごとに任意の個数のアームを選択し、実行する。それぞれのアーム i にはコスト c_i が設定されており、プレイヤーはアーム i を実行するごとにコスト c_i を支払い、アーム i に設定されたばらつきのある報酬 μ_i を得る。このときプレイヤーは報酬 μ_i を事前には知らない。また、プレイヤーは予算 B を持ち、支払うコストの合計は予算 B に収まっていないと同時に、ステップ数は $T(\in \mathbb{N})$ に制限される。プレイヤーの目的は、 T ステップ以内に予算 B を駆使し得られる報酬を最大化するようにアーム i を選択・実行することである。

拡張 BLMAB モデルでは、従来の BLMAB モデルにおける制約式 4 を次のように再定義する。

$$P\left(\sum_i^K N_i^A(B)c_i \leq B \cap t \leq T\right) = 1 \quad (5)$$

4. 拡張 BLMAB モデルの適用

4.1 エンドポイント探索における適用

エンドポイント探索においては、探索対象となる各エンドポイントがアーム i に相当するが、それらのエンドポイントに対して実行される評価クエリの個数は有限であるため、アーム i の実行回数には限りがあるとみなすことができる。エンドポイント探索における拡張 BLMAB モデルでは、評価クエリの個数を Q としたとき、アーム i は $N_i^A(B) \leq Q$ を満たすべきである。

MAB に属する問題の目的は、得られる報酬の合計を最大化することであるが、エンドポイント探索における目的は、ユーザにとって最も良いエンドポイントを見つけることであるため、 $i(A)$ を A が定めた最も良いアーム、 $i(A^*)$ を理論上の最も良いアームとしたとき、エンドポイント探索における拡張 BLMAB モデルでは、機会損失式を次のように再定義する。

$$R(A) = \mu_{i(A^*)} - \mu_{i(A)} \quad (6)$$

4.2 ϵ -greedy アルゴリズムの適用

ϵ -greedy アルゴリズムは、強化学習の分野においてよく知られたアルゴリズムであり、 $(1-\epsilon)$ の確率で貪欲法に従い、 ϵ の確率でランダムサーチを行う。

Algorithm 1 は、エンドポイント探索における拡張 BLMAB モデルを適用した ϵ -greedy アルゴリズムである。

Algorithm 1 エンドポイント探索における ϵ -greedy

Require: $0 \leq \epsilon \leq 1$

```

1:  $t = 1, B_t = B, T_t = T$ 
2: while  $B_t > \min_i c_i$  かつ  $t \leq T$  do
3:    $s(B_t, T_t)$  を同時実行数とする
4:   for  $j = 1$  to  $s(B_t, T_t)$  do
5:     if  $(1-\epsilon) >$  乱数値  $\in [0, 1]$  then
6:        $\{i(t)\}$  に選択可能なアームの中から  $t$  の時点で最も良い
       評価値  $H_i(t)$  のアームを追加する
7:     else
8:        $\{i(t)\}$  に選択可能なアームの中からランダムにアームを
       追加する
9:     end if
10:  end for
11:  選択したアームの集合  $\{i(t)\}$  に属するアームを実行
12:   $t, B_t, T_t$  を更新
13: end while
14: 最良のアームを決定

```

$t = 1, 2, 3 \dots$ とし、 B_t を t における残りの予算、 $\{i(t)\}$ を t において選択するアームの集合と定義する。

Algorithm 1 の 3 行目における $s(B_t, T_t)$ は、 t においていくつのアームを選択するかを決定する関数である。

t の時点におけるアーム i の平均報酬を $\hat{\mu}_{i, n_{i, t}}$ とし、Algorithm 1 の 6 行目における評価値 $H_i(t)$ は次のように定義する。

$$H_i(t) = \frac{\hat{\mu}_{i, n_{i, t}}}{c_i} \quad (7)$$

Algorithm 1 の 6 行目と 8 行目における選択可能とは、アーム i において $N_i^A(B) \leq Q$ かつアーム i を $\{i(t)\}$ に追加した時点で、残り予算から $\{i(t)\}$ の実行にかかるコストを引いても 0 を下回らないことである。

Algorithm 1 の 14 行目における最良のアームとは、 ϵ -greedy アルゴリズムによって最も選択されたアームとし、選択された回数が同じであれば、平均報酬が高いほうを最良のアームと定義する。

4.3 KDE アルゴリズムの適用

KDE アルゴリズム [Tran-Thanh 12b] は BLMAB 問題に対する強力なアルゴリズムの一つである。

Algorithm 2 は、エンドポイント探索における拡張 BLMAB モデルを適用した拡張 KDE アルゴリズムである。

アームが実行可能であれば、次の式 8 で表される問題を、密度順貪欲法により、残り予算 B_t を使った最適なアームの組み

Algorithm 2 エンドポイント探索における KDE

Require: $\gamma > 0, \alpha > 0$
 1: $t = 1, B_t = B$
 2: **while** $B_t > \min_i c_i$ かつ $t \leq T$ **do**
 3: 密度順貪欲法を用いて $M^*(B_t) = \{m_{i,t}^*\}$ を計算することによって、最適なアームの組み合わせを近似的に求める
 4: $\epsilon_t = \min\{1, \frac{\gamma}{\alpha t}\}$
 5: $s(B_t, T_t)$ を同時実行数とする
 6: **for** 1 to $s(B_t, T_t)$ **do**
 7: $\{i(t)\}$ に選択可能なアームの中から確率 $P(i(t) = i) = (1 - \epsilon_t) \frac{m_{i,t}^*}{\sum_{k=1}^K m_{k,t}^*} + \frac{\epsilon_t}{K}$ に従って選択したアームを追加する
 8: **end for**
 9: 選択したアームの集合 $\{i(t)\}$ に属するアームを実行
 10: t, B_t, T_t を更新
 11: **end while**
 12: 最良のアームを決定

合わせを近似的に求める.

$$\max \sum_{i=1}^K m_{i,t} \hat{\mu}_{i,n_{i,t}} \quad s.t. \quad \sum_{i=1}^K m_{i,t} c_i \leq B_t, \forall i, t : m_{i,t} \text{ integer} \quad (8)$$

上記の式 8 における $\hat{\mu}_{i,n_{i,t}}$ は t におけるアーム i の平均報酬を示しており、 $n_{i,t}$ は t までのアーム i の実行回数を表している。この問題は式 8 を最大化するような $\{m_{i,t}\}_{i \in K}$ を残り予算 B_t に注意しながら見つけることが目的であるが、この問題はナップザック問題と同一であり、NP 困難であるため、Algorithm 2 の 3 行目では密度順による貪欲法を用いて近似的に組み合わせを求めている。 t におけるアーム i の密度 $\rho(i, t)$ は、次の式で与えられる。

$$\rho(i, t) = \frac{\hat{\mu}_{i,n_{i,t}}}{c_i} \quad (9)$$

また、その組み合わせを $M^*(B_t) = \{m_{i,t}^*\}$ と定義する。この $\{m_{i,t}^*\}$ を用い、KDE アルゴリズムでは、確率的に次に実行するアーム $i(t)$ を選択する。その確率は次の式で表される。

$$P(i(t) = i) = (1 - \epsilon_t) \frac{m_{i,t}^*}{\sum_{k=1}^K m_{k,t}^*} + \frac{\epsilon_t}{K} \quad (10)$$

ϵ_t の値は t とともに減少するため、 t が増えるごとに優先度の高いアームが選択されやすくなる。

5. シミュレーションによる評価の検討

この章では、新しく定義した拡張 BLMAB モデル、及びエンドポイント探索に向けた拡張 BLMAB モデルの適用により、目的関数が従来のモデルと異なるため、本研究におけるモデルの性質を示すためのシミュレーションによる評価を試みる。

5.1 評価クエリ

本研究では、SuPARQooL と同様に、図 1 で示すようにユーザは未知のノードを“<<”と“>>”によって明示することを仮定する。

評価においては、図 2 で示されるような評価クエリを用いる。これらのクエリは、図 1 のユーザクエリと対応しており、未知のノードをキーワードとして、そのキーワードが含まれるようなデータを検索するクエリとなっている。

```
SELECT *
WHERE {
  ?x ?y <<Object>>.
  <<Subject>> <<Predicate>> ?x.
}
```

図 1: ユーザからのクエリ例

```
SELECT ?a
WHERE {
  ?x ?y ?a.
  FILTER regex(str(?a), "Object", "i")
}

SELECT ?a ?b
WHERE {
  ?a ?b ?x.
  FILTER regex(str(?a), "Subject", "i")
  FILTER regex(str(?b), "Predicate", "i")
}
```

図 2: 評価クエリの例

5.2 用意したエンドポイント

本評価は、主に拡張 BLMAB モデルの性質を明らかにする意味で、仮想的なエンドポイントを 100 個用意した。一つのエンドポイントには、10000 件のトリプルがあり、それらのトリプルは、全てのエンドポイントにおいて、100 種類の主語、目的語に当たるデータと、100 種類の述語に当たるデータを無作為に選択し、構成したものである。

5.3 ϵ -greedy アルゴリズムによる初期的评价

Algorithm 1 に対して本シミュレーション条件を適用した際の結果を図 3 と図 4 に示す。横軸は ϵ 、縦軸は式 6 で定義された機会損失 $R(A)$ である。 $R(A)$ の値が低いほど、良い結果であることを示している。ここでは、図 3 のパラメータを $Q = 10, B = 200, \forall c_i = 1, T \in \{20, 40, 200\}$ 、図 4 のパラメータを $Q = 10, B = 400, \forall c_i = 1, T \in \{40, 80, 400\}$ とし、両方とも $s(B_t, T_t) = \frac{B_t}{T_t}$ と定義した。図 3 と図 4 より、従来

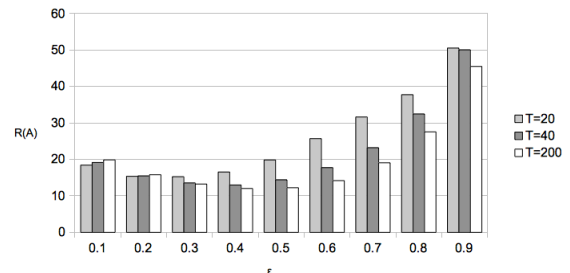


図 3: $Q = 10, B = 200, \forall c_i = 1, T \in \{20, 40, 200\}$ としたときの Algorithm 1 の結果

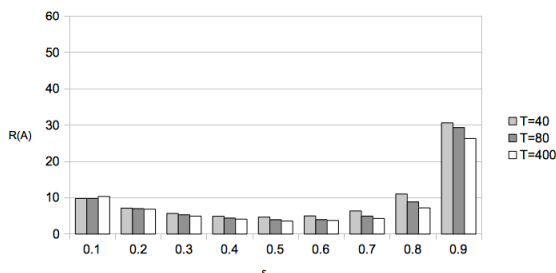


図 4: $Q = 10, B = 400, \forall c_i = 1, T \in \{40, 80, 400\}$ としたときの Algorithm 1 の結果

の BLMAB モデルと同様に、予算が多いほど良い結果が得られていることが分かる。また、本研究における拡張 BLMAB モデルに追加したタイムステップ数上限 T をより強く制限するにつれ、性能が低下する傾向があることを確認した。また、従来の MAB 問題における ϵ -greedy アルゴリズムの最適パラメータは、おおよそ $\epsilon = 0.1$ 程度であるが、図 3 と図 4 を見ると、最適な ϵ の値は 0.4 から 0.6 程度となっている。エンドポイント探索においては、目的関数が最良のアームを見つけることになっていることと合わせて考えると、従来のモデルより情報収集に重きを置いたほうが良い結果が得られるものと思われる。

6. エンドポイント探索支援システム

本研究におけるエンドポイント探索支援システムは、SuPARQooL のエンドポイント推薦機能を、MAB モデルに基づいたエンドポイント推薦機能へ拡張することで実現している。図 5 に拡張したエンドポイント探索支援システムの構成を示す。エンドポイント探索に MAB モデルに基づく手法を用いることにより、探索対象となるエンドポイントの数やキーワードの数が増大しても、ネットワークや評価クエリの実行にかかる時間の制約がある中で効率的なエンドポイントの発見を行い、推薦を行うことが可能となる。

7. まとめと今後の課題

本研究では、エンドポイントの探索を支援するために、ネットワークへの負荷、および評価クエリの実行時間の二つのコストを考慮した上で、決定理論の一つである MAB モデルとそのアルゴリズムを拡張して適用し、エンドポイント推薦機能の効率化への適用を試みた。

今後の課題としては、本研究で提示した拡張 BLMAB モデルは横断的検索を考慮していないモデルであるため、横断的検索を考慮したモデルへの拡張を行うこと、および、実データを考慮した定量的な評価手法の確立と実験を行うことが挙げられる。

参考文献

- [稜野 14] 稜野 寿章, 福田 直樹. LOD 検索におけるエンドポイント探索への Budget-Limited Multi Armed Bandit に基づく最適化手法適用への検討. 第 76 回情報処理学会全国大会, 2014.
- [Noguchi 13] Noguchi, H. Fujino, T. and Fukuta, N. On Implementing a SPARQLoid Query Coding Support

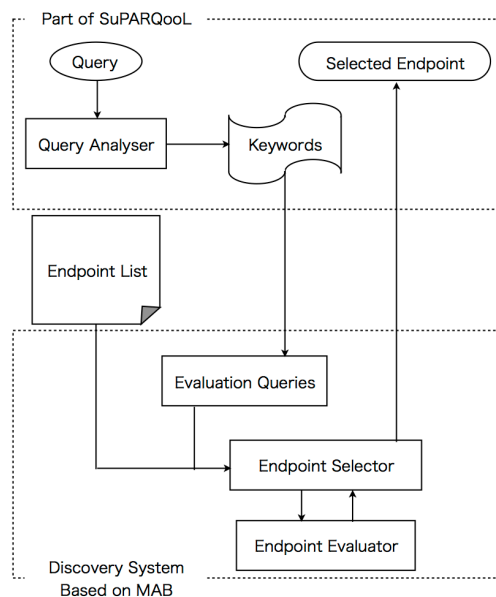


図 5: エンドポイント探索支援システムの構成

Vocabulary Discovery for Queries with Weighted Ontology Mappings *The 3rd Joint International Semantic Technology Conference*, 2013.

[Fujino 12] Fujino, T. and Fukuta, N. SPARQLoid - a Querying System using Own Ontology and Ontology Mappings with Reliability. *The 11th International Semantic Web Conference (Posters and Demos)*, 2012.

[Fujino 13] Fujino, T. and Fukuta, N. On Implementing a SPARQLoid Query Coding Support Vocabulary Discovery for Queries with Weighted Ontology Mappings. *The 3rd Joint International Semantic Technology Conference*, 2013.

[Robbins 52] Robbins, H. Some aspects of the sequential design of experiments. *Bulletin of the AMS* 55:527–535, 1952.

[Tran-Thanh 10] Tran-Thanh, L. Chapman, A. Munoz De Cote Flores Luna, J. Rogers, A. and Jennings, N. Epsilon-First Policies for Budget-Limited Multi-Armed Bandits. *In, Twenty-Fourth AAAI Conference on Artificial Intelligence, Atlanta, USA, Georgia, 11 - 15 Jul 2010.*, 1211-1216, 2010.

[Tran-Thanh 12a] Tran-Thanh, L. Chapman, A. Rogers, A. and Jennings, N. Knapsack based optimal policies for budget-limited multi-armed bandits. *In, Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI-12), Toronto, CA, 22 Jul 2012.*, 1134-1140, 2012

[Tran-Thanh 12b] Tran-Thanh, L. Budget-limited multi-armed bandits. *University of Southampton, Faculty of Physical and Applied Sciences, Doctoral Thesis*, 2012