クリックに基づく選好グラフを用いたバーティカル適合性推定

門田見　侑大†　　吉田　泰明††　　藤田　澄男††　　酒井　哲也†

† 早稲田大学大学院 基幹理工学研究科 情報理工・情報通信専攻
〒 169–9555 東京都新宿区大久保 3–4–1
†† ヤフー株式会社
〒 102–0094 東京都千代田区紀尾井町 1-3
E-mail: †kdtm-783640@ruri.waseda.jp, ††{yayoshid,sufujita}@yahoo-corp.jp, †††tetsuyasakai@acm.org

あらまし　検索意図の理解に関する研究のひとつに，ユーザーの要求するバーティカルを推定するタスクがある．バーティカル推定にはクエリとバーティカルの対に関する適合性の正解データが必要であるが，その作成コストが膨大となることや，スパース性が問題となっている．本研究では，バーティカルを含むモバイル検索結果に対するクリックログをもとに，与えられたクエリに対する各バーティカルの適合性を自動推定するタスクを扱う．具体的には，クリックログからウェブページの選好グラフを構築し各ページの適合性ラベルを自動付与するアルゴリズムを応用し，モバイル検索におけるクリックログからバーティカルの選好グラフを構築し，各バーティカルの適合性を自動推定した．結果として，与えられたクエリに対し，各バーティカルに高適合・適合・不適合のいずれかのラベルを自動付与するタスクではクリック数ベースの単純な手法にかなわなかったが，バーティカルを適合度順に並べる評価においてはある程度有効性を示すことができた．

キーワード　情報検索，クエリログマイニング

1. はじめに

情報要求はいつでも，いかなる場所でも発生する．情報要求を満たすために日常的に行われる検索行為は音声検索や自然言語検索の普及により複雑化し，検索される内容もスラングの発生や新商品の開発などとともに多様化している．検索エンジンのランキングアルゴリズムを改善するには正確な学習データが大量に必要であるが，上記の状況により学習データの作成コストは多大なものとなっている．

一方，近年，アプリやキュレーションサイト等の特定のバーティカルに特化した情報収集ツールが多く出現している．検索クエリから情報要求の種類を特定することができれば，そのようなツールの情報提供をユーザーに行うことで，ユーザーの情報要求の充足に役立つと考えられる．検索意図の理解に対する研究のひとつに，ユーザーの要求するバーティカルの推定や各クエリに関連するバーティカルのランクリストを返すといったタスクが存在する．しかし，バーティカル検索アルゴリズムの改善やそれらの評価を行うためのクエリとバーティカルの関連性に関する正解データのスパース性が問題となっている．

本研究では，上記の問題を解決するため，Yahoo! JAPAN (注1)から提供されたバーティカル情報が付与されたページに対するクリックログの集合を入力とし，各バーティカルの関連性の度合いを 3 段階で示すラベルを出力するタスクを扱う．クリックログの集合から作成される選好グラフを用いてバーティカルに着目した関連性ラベルの自動付与アルゴリズムを提案し，その有用性の評価を行った．(注2)評価の結果，与えられたクエリに対し，各バーティカルを高適合・適合・不適合に自動的に振り分けるタスクではクリック数ベースの単純な手法にかなわなかったが，バーティカルを適合度順に並べる評価においてはある程度有効性を示すことができた．また，知見を共有するために行った失敗分析の結果も本論文に記載する．

Arguello ら [1] はバーティカルを「ニュース・旅行・ローカル検索のような特定の分野もしくは画像や動画のような特定のメディアのために特化された部分コレクション」と説明しており，我々もこの定義に従う．

2. 関連研究

クエリの分析とバーティカルに関する研究として，二項クエリモデルを用いた情報タイプ (本論文におけるバーティカル) の抽出がある [2]．検索システムの利用者はクエリを物事の下位範疇と情報の下位範疇の 2 項形式で表現すると考え，クエリログの集合から十分な頻度を持つ第 2 項の単語を収集し，情報要求の経年変化を分析している．

また，クエリログを用いた研究として，Jansen ら [3] [4] [5] は Dogpile.com や Excite のログを用いて絞り込みや汎化といったクエリの改変パターンの変遷やバーティカルからバーティカルへの遷移確率，クエリに含まれる単語数やページの閲覧数などを分析している．近年では様々なデバイスの普及に伴い，デバイス毎のユーザーの振る舞いの分析も行われている．Song ら [6] はモバイル，タブレット，デスクトップのそれぞれのクエリログをクエリの長さや検索する時間，場所などに着目し分析

(注1)：http://www.yahoo.co.jp

(注2)：本論文は Yahoo!JAPAN 研究所でのインターン中の研究内容である．

を行い，各デバイスごとに特徴があることを活かし，それぞれのデバイスに最適化したランキングの生成を試みている．

バーティカルを用いたタスクに，TREC Federated Web Search Track [7] の Vertical Selection Task や NTCIR の IMine-2 Task [8] の Vertical Incorporating subtask がある．これらは，与えられたクエリに対して適切なバーティカル，もしくはバーティカルのモジュールを含むランキングを返すシステムを構築し，検索有効性を競うタスクである．TREC の参加チーム [9] [10] は，機械学習手法や，言語モデルを用いるなど様々な手法でこのタスクに取り組んでいる．その中で，最も F 値の高かったチーム [11] は「あるバーティカルにおいて多く出現する単語はそのバーティカルを表す可能性が高い」という仮定のもと FTR (Frequent Term Rank) という指標を用いている．これにクエリ拡張や IR モデルを組み合わせた手法が最高の F 値を示している．

関連性ラベルの付与に関しては，Agrawal ら [12] や Xu ら [13] がクリックログを用いて研究を行っている．検索エンジンのランキング関数はクエリと URL のペアを学習データとして機能の改善を図る [14]．現在は人手によって学習データのラベリングを行っているが，学習に必要とされる正確なデータの量は日々増えている．さらに，人手によるラベリングは協力者の確保や時間的かつ金銭的コストが掛かることが問題となっている．そこで Agrawal ら [12] はクリックログから SERP(Search Engine Result Page) に出現するドキュメントの選好グラフを作成し，そのエッジの重みやグラフから算出されたスコアやページランクを用いて，与えられたクエリに対し，各 URL の上位下位関係および関連性ラベルの自動作成を提案している．

Xu ら [13] はクエリログからドキュメントがクリックされた回数や滞在時間などを素性として抽出し，ラベルを予測するための最適なラベルの学習を行っている．ここでユーザーはドキュメントを比較しながら情報収集をするためドキュメント間で関連性の依存性があるとして 2 つのモデルを提唱し，関連性ラベルの学習，予測，エラー検出を行っている．

本研究では Agrawal らのラベル付与アルゴリズムをバーティカル情報が付与されたクリックログに適応し，バーティカルの関連性ラベルの自動生成およびその結果に対する分析を行った．

3. 提 案 手 法

本論文では Agrawal ら [12] の研究をもとにクエリ毎に選好グラフを作成し，そのグラフを用いてクエリとバーティカルの関連性ラベルを作成する事を目的とする．作成したシステムは複数レコードのクリックログを入力として，各クエリのバーティカルに対する関連性ラベルを自動付与するシステムである．

3.1 入力データ

3.1.1 データの素性

Yahoo! JAPAN の 2016 年 3 月のモバイル端末におけるクエリログを用いて実験を行った．ヘッドクエリの中からクエリをランダムに抽出したデータを用いた．モバイルのデータに限定している理由は，モバイルの場合，検索結果が 1 カラムであるためランキング通りにユーザーが視認するが，PC の場合，検索結果が 2 カラム表示される場合があり，クリックにバイアスがかかってしまう，ランキングの位置を利用する手法を用いることが出来ないといった問題を考慮したためである．

これらのデータは SERP の表示 1 回に付き 1 レコードであり，以下に示すクリック情報が含まれている．

- バーティカル情報
- クリック先 URL
- クリックした時間
- 埋め込み位置
- モジュール名
- モジュール内ランク

SERP にはバーティカル情報の付与されていない web ページの検索結果が 10 件含まれており，バーティカル情報が付与されているリンクの集合はそれらの間に挿入されている．埋め込み位置はどのランキングの web ページの下に挿入されているかを表す (ランキング 1 位と 2 位の web ページの間に挿入されていれば埋め込み位置は 1 となる)．モジュール名は埋め込まれたバーティカルが付与されたリンクの集合内の表示のされ方を示す文字列であり，モジュール内ランクはその中での表示順を表す．実際には 1 レコードにクリック情報が複数含まれており，バーティカルが紐付けられていないリンクのデータは含まれていない．これらの配列を埋め込み位置，モジュール内ランクでソートし，SERP 内バーティカルランキングとして扱った．アクセス先のページのバーティカル情報を以下に示す．

- 画像
- レシピ
- 質問サイト
- 動画
- 周辺地域情報
- ニュース
- ショッピング
- オークション
- リアルタイム
- タレント情報
- 地図
- 辞書

バーティカルの情報が付与されているリンクは同じバーティカルで連続しているが，異なるバーティカルの埋め込み位置が連続しているとは限らない．例えば，[画像 1, 画像 2, 画像 3, web1, web2, web3, 動画 1, 動画 2] というようにバーティカルの集合の間にバーティカルの付与されていない web 分類の URL が埋め込まれている．また，クリック情報内のクリック先の URL はクリックされているものにのみ付与されている．そのため，同じクエリ，同じ埋め込みランクであっても同一の URL であることは保証されない．したがって，本研究では各 URL に関する正確な選好関係は抽出できないことからバーティカルに焦点を当てている．図 1 に，本研究で実際に扱うデータの形式を簡略化したものを示す．クエリ毎に複数の SERP 情報が含まれており，SERP 毎に表示されたリンクのクリック情報が配列で含まれている．

3.2 入力データの作成

今回用いた手法では上記のデータから 2 種類のランクリストを作成して実験を行った．

URL-List バーティカルの付与されたリンクを表示された順番に並べた．今回は全ての URL 情報が存在しているわけではないので同じ位置に表示されていたリンクだとしても URL が異なる場合があるが，その差は無視することとした．図 1 を例に挙げると SERP1 に関しては [レシピ 1, レシピ 2, 画像 1, ... , 動画 1]，SERP2 に関しては [レシピ 1, レシピ 2, レシピ 3, 画

```
# query が味噌汁のとき
SERP1 = [
    [レシピ, http://example.abc, 143147260, 1, example_module, 1],
    [レシピ, null, null, 1, example_module, 2],
    [画像, http://example2.abc/1, 143147360, 5, example_module2, 1],
    ...
    [動画, null, null, 10, example_module3, 1],
]
SERP2 = [
    [レシピ, null, null, 1, example_module, 1],
    [レシピ, null, null, 1, example_module, 2],
    [レシピ, http://example2.abc/5, 143148921, 1, example_module, 2],
    [画像, null, null, 5, example_module2, 1],
    ...
    [動画, http://example3.abc/8, 14318980, 10, example_module3, 1],
]
SERP3 = [
    ...
..
```

図 1　本研究で実際に扱うデータの形式を簡略化したもの

像 1, ... , 動画 1] というようなリストを生成した．

Vertical-List　バーティカル毎に連続したリンクを 1 つの要素とし，それらを表示された順番に並べた．利用したデータの中には同一バーティカルであっても連続していないリンクも存在したが，今回は連続していない同一バーティカルは別々の要素として扱った．図 1 を例に挙げると SERP1，SERP2 ともに [レシピ, 画像, 動画] というようなリストを生成した．

本論文では与えられたランクリストの 1 つの要素を「バーティカル要素」と呼ぶ．ランクリストは 1 レコードにつき 1 つ作成され，クエリごとに複数のランクリストが作成される．Agrawal らの研究では文書のみに焦点をあてており，文書のみで構成されたランクリストを用いて実験を行っている．

3.3　選好ルール

本論文では Agrawal らと同様に Joachims ら [15] が提示しているルールを適応し，選好グラフの作成を行った．ルールの一覧を表 1 に示す．

表 1　適応した選好ルール

ルール ID	ルール内容
R1	Click > Skip Next
R2	Click > Skip Above
R3	Click > Skip Previous
R4	Last Click > Skip Above
R5	Click > Click Above
R6	Click > Skip Other

これらのルールは「>」の左側のほうがユーザーがより好むものであるということを表す．例えば，「Click > Skip Next」はあるランクのバーティカル要素がクリックされた場合，そのバーティカル要素は次のランクのバーティカル要素よりも適合性が高いとみなす，ということを表す．

3.4　選好グラフの作成

SERP 情報から作成された複数のランクリストとクリック情報を用いてクエリごとに選好グラフを作成する．選好グラフはノードがバーティカル要素に相当し，エッジは適合度が高いとみなされるノードから適合度が低いノードへ向けて張られる．つまり，ノード v_i から v_j へのエッジは v_i の方が v_j よりクエリとの適合度が高いことを表す．エッジの重みはエッジでつながっている 2 つのノードに対応する両方のバーティカル要素の情報を読んだユーザーの数を表す．この情報とはバーティカル要素がリンクであればタイトルやスニペット等のページに関する情報，画像や動画などであれば含まれる画像やタイトルなどを表す．Agrawal らの研究に則り，以下の様にエッジの重みを増やす．

SERP のクリック情報が与えられ，ある位置 j のバーティカル要素 E_j がクリックされている時，ルールに対応し，エッジが張られる対象となる位置 i のバーティカル要素を E_i とする．Agrawal らは E_j から全ての E_i に張られるエッジに対し，$P(i,j)$ の確率で対応するエッジの重みを 1 増やし，$1-P(i,j)$ の確率で何もしないという方法で重みを増やしている．本研究ではデータの量が十分であればエッジの重みの合計は $P(i,j)$ とルールの適応回数の積に近似できるという考えのもと，エッジの重みをルールが適応される度に $P(i,j)$ の値だけ増加させる方法を取っている．ここで $P(i,j)$ は，ユーザーが E_i のページに関する情報を読み，E_j をクリックする確率を表す．

前節の例をもとに補足すると，「Click > Skip Next」のルールを適応した場合，E_j がクリックされ，次のランクのバーティカル要素 E_{j+1} がスキップされたとすると，E_j に相当するノードから E_{j+1} に相当するノードへのエッジの重みが $P(j+1,j)$ の値だけ増える．

選好グラフのイメージを図 2 に示す．この図はバーティカル要素 A に着目している．赤い矢印は A の方が優位である事を示すエッジ，青い矢印は劣位であることを示すエッジを表す．各矢印に付随している数字が各エッジの重みを表している．したがって，A に接続されているエッジだけを見れば A は B，D よりも関連度が高く，C より関連度が低いと言える．しかし，実際にはエッジはほぼ全てのノードに相互的に張られており，直感的に関連度の優劣を推測するのは不可能である．

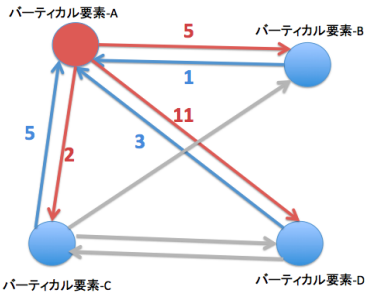

図 2　選好グラフ

3.5　ユーザーモデル

Agrawal らの研究では，前節の $P(i,j)$ はアイトラッキングの研究 [16] [17] に基づくクリックログのクリックと閲読の関係を表す確率が用いられている．この確率は SERP 内においてクリックされた文書より下位にある文書は下位にあるほどスニペットが読まれている割合が減少しているという事実をもとに作成されている．ランキング 1 位の文書からクリックされた文書の 1 つ下の文書は必ずスニペットが読まれており，その結果

ユーザーが比較を行っているが，それ以下の文書のスニペットを必ずしも読んでいるわけではないので選好ルールを適応する際にスニペットを読んでいるユーザーの割合を用いて選好グラフの重み付けを行っている．しかし，用いられている確率のデータは PC の検索結果に対するものであるのに対し，本論文で用いているログデータはモバイルのものであり，ランクリストに含まれるページ数も異なる．さらに，文書だけに焦点を当てているわけではない．そのため，我々は以下のようなユーザーの行動を表す確率モデルを 3 種類作成した．ここで i は選好グラフ作成時にエッジを張る対象となるバーティカル要素の位置，j はクリックされた最下位のバーティカル要素の位置を表す．

model1　ユーザーが全てのバーティカル要素の情報を閲覧しているとする一様なモデル (全ての i，j に対して $P(i,j)=1$)

model2　クリックされたバーティカル要素よりランクが下位の情報閲覧率が指数関数にしたがって減少するモデル

model3　クリックされたバーティカル要素よりランクが下位の情報閲覧率が線形関数にしたがって減少するモデル

指数関数によって表現されるモデルを以下に示す．

$$P(i,j)=\begin{cases} 1 & (i-j-1 \leqq 0 \text{ のとき}) \\ 2^{-(i-j-1)} & (i-j-1>0 \text{ のとき}) \end{cases}$$

線形関数によって表現されるモデルを以下に示す．

$$P(i,j)=\begin{cases} 1 & (i-j-1 \leqq 0 \text{ のとき}) \\ -0.1\cdot(i-j-1)+1 & (10 \geqq i-j-1>0 \text{ のとき}) \\ 0 & (i-j-1>10 \text{ のとき}) \end{cases}$$

これらのモデルは全てのルールに適応できるが，全てのモデルに関して $P(1,i)$ から $P(i+1,i)$ が全て 1 となっている．また，R1〜R5 はルールによってエッジを張る対象がクリックされたバーティカル要素の 1 つ下までであるため，R6 以外はモデルによる影響は受けない．

3.6　関連性ラベルの判定

作成した選好グラフを用いた関連性ラベルの判定方法を示す．

3.6.1　問 題 定 義

ノード集合を V，順序付けされた K 個のラベルの集合を $L=\{L_1,L_2,...L_K\}$，あるノード v のラベルを $L(v)$ とする．また，ノード u から v へ出て行く各エッジを e_{uv}，あるエッジ e の重みを $w(e)$ と表す．

K 個のラベルと選好グラフが与えられ，$L(u)>L(v)$ であるとき，e_{uv}, e_{vu} はラベルの境界を跨ぐエッジである．直感的には，e_{uv} はラベル付けを肯定することを表し，e_{vu} はラベル付けを否定することを表す．ここで肯定的なエッジの集合を F，否定的なエッジの集合を B とすると我々が解くべき問題は，F に含まれるエッジの重みの合計を最大化し，B に含まれるエッジの重みの合計が最小となるようにラベル付けを行う事である．

ラベル L_k に関するもっともらしさを $Ag(L_k)$ とすると以下で表され，これを最大化する問題となる．

$$Ag(L_k)=\sum_{e_{uv}\in F} w(e_{uv})-\sum_{e_{vu}\in B} w(e_{vu})$$

3.6.2　順序付きリストの作成

まず，ノードを順序付けしたリストを求める．順序付けは以下の 3 つの方法で行った．

スコアによる順序付け　ここであるノード u に関するユーザーの選好を反映したスコアを以下の式で表す．

$$Score(u)=\sum_{v\in V} w(e_{uv})-\sum_{v\in V} w(e_{vu})$$

このスコアを降順でソートした．

ページランク　Google の検索結果の表示に用いられている文書の被リンク数から文書の重要度を算出する手法であるページランクアルゴリズム [18] を用いた．今回作成した選好グラフはエッジが出て行くノードの方が重要視される．しかし，ページランクアルゴリズムはページがどれほど他のページから参照されているかに着目しているため，エッジが入ってくるノードが重要視される．そのため，今回は作成した選好グラフのエッジの向きを全て逆にした後にページランクアルゴリズムを適応する．ある文書を D，Ti を D にリンクを張っている文書，$C(Ti)$ を Ti の持つリンクの総数，n をグラフに存在する文書数としたときページランクを導出する式を以下に示す．

$$PageRank(D)=(1-d)+d\sum_{i=1}^{n}\frac{PageRank(Ti)}{C(Ti)}$$

今回は減衰定数 d を 0.85 として収束するまで計算した．D に関するページランクを計算し，降順でソートした．

重み付きページランク　ページランクの更新に今回作成した選好グラフの重みを用いる．ページランクアルゴリズムはノード間の移動はエッジが出ている他のノードに同確率で行われるが，今回はエッジの重みの比率をサーファーが移動する確率として用いる．導出の式を以下に示す．

$$WeightedPageRank(D)=$$
$$(1-d)+d\sum_{i=1}^{n}\frac{WeightedPageRank(T_i)\times w(e_{T_iD})}{\sum_{j=1}^{n} w(e_{T_jD_j})}$$

オリジナルのページランクと同様に減衰定数 d を 0.85 とし，降順でソートした．

3.6.3　分割アルゴリズム

次に作成した順序付きのリスト $[v_1, v_2,...v_n]$ を K 個のクラスに分割するアルゴリズムについて述べる．K 個のクラスに分けるためには $K-1$ 個の分割点が必要であり，分割点は $n-1$ 個の候補を持つことになる．ここで二次元配列 OPT を定義する．$OPT[k,i]$ はリストを k 個に分割する時に最後の分割点が i である時の尤もらしさを表す．したがって，リストを K 個に分割するときの最後の分割点は $OPT[K-1,i]$ が最大となるときの i ということになる．さらに，B という二次元配列を定義する．$B[j,i]$ は最後の分割点が j であるリストの i の位置に分割点を足す際の利得を表す．$OPT[k,i]$ の導出を以下に示す．

$$OPT[k,i]=\max_{k-1\leqq j\leqq i}\{OPT[k-1,j]+B[j,i]\}$$

ここで $k=1$ のとき，$OPT[1,i]=B[0,1]$ である．B は

作成した選好グラフのエッジの向きと重みによって計算される．あるエッジ $e_{v_x v_y}$ が与えられたとき，$l = \min(x, y)$, $r = \max(x, y)$(つまり，最前に近い方の位置を l，最後に近い方の位置を r) とする．この時，$0 \leqq j \leqq l-1$ かつ $l \leqq i \leqq r-1$ を満たす $B[j,i]$ を更新する．$x > y$ の場合，$B[j,i]$ を $w(e_{v_x v_y})$ だけ増やし，$x < y$ の場合，$B[j,i]$ を $w(e_{v_x v_y})$ だけ減らす．このアルゴリズムは K 個のクラスに分割する際に，それぞれのクラス間にまたがるエッジの重みの合計が最大になるような分割点を見つけるアルゴリズムである．

3.7 バーティカルへのラベル付け

上記アルゴリズムによって多段階に分割された順序付きリストから，各バーティカルの関連性ラベルを判定する．まず，分割された各リストに，最前から L_1, L_2,...L_K とラベルを付け，それに属する各ノードにも同じラベルを割り当てる．その後，各バーティカルに属するノードに割り当てられたラベルのうち，最も関連性の高いラベルをそのバーティカルのラベルとした．本研究では $K = 3$ とし，L_1 から順番に「高適合」,「適合」,「不適合」というラベル名をつけている．既存研究では 5 段階のラベル付けを行っているが，今回はバーティカルという抽象度の高いものをターゲットとしている上に全体数も文書に比べると少ないため 3 段階のラベル付けで実用上充分であると考えた．

3.8 バーティカルのランクリストの出力

本研究ではラベル付けのタスクだけでなく，バーティカルのランクリストの出力とその評価も行った．ラベル付けのタスクがバーティカルに関する適合性判定の絶対的評価であるのに対し，こちらは相対的評価をするものである．各バーティカルに属するノードのうち，順位が高いものから順に並べ，各バーティカルに変換したものをランクリストとして出力する．

4. 評価実験

4.1 正解データ

システムの有用性を評価するにあたり，人手による正解データの作成を行った．関連性有無のラベル付けを行うにあたり提供したデータは，クエリとその検索結果に表示されるバーティカルのリストのペアである．今回のデータにおいて，バーティカルのリストは 1 から 8 個のバーティカル情報を含んでいる．これらのデータに対し，以下の観点からアノテータに 3 段階の点数付けを行ってもらった．

2 点　クエリに対し，このカテゴリの情報を閲覧すると思う
1 点　文脈 (時間や場所) によっては閲覧することがあると思う
0 点　関連性がなく閲覧することはないと思う

アノテータは早稲田大学に所属する男女 9 人であり，この内 8 名には 2 名ずつに同じデータを 300 クエリずつ，合計 1200 クエリの点数付けを依頼し，1 名には 1200 クエリの点数付けを行ってもらった．一つのクエリに付き 3 人が点数付けを行っている．この結果を用いて多数決により正解データの作成を行った．その後，2 点，1 点，0 点をそれぞれ「高適合」,「適合」,「不適合」としてラベルをつけた．3 人の点数がそれぞれ 0，1，2 点であった場合は「適合」を正解とした．バーティカルの総数は 3937 個であった．それぞれのバーティカルにおけるラベルの数，配点の分布を表 2，3 にそれぞれ示す．表 3 の配点のカラムはアノテータが付けた点数を表している．例えば「2-2-1」であれば 2 人が 2 点をつけ，1 人は 1 点をつけたことを表す．この分布を見ると一致率は高くなく，人間にとってもラベルを推定するタスクは難しいのではないかと言える．

表 2　バーティカル毎のラベルの数

バーティカル	高適合	適合	不適合
画像	407	209	100
レシピ	43	4	0
オークション	21	54	109
辞書	24	24	43
質問サイト	194	397	235
動画	170	61	24
ニュース	118	283	330
地図	24	1	3
リアルタイム	43	122	502
ショッピング	73	42	43
タレント情報	113	58	19
周辺地域情報	32	9	3
合計	1262	1264	1411

表 3　アノテータの点数付けの分布

配点	ラベルの割合
2-2-2	10.62%　(418 個)
2-2-1	12.40%　(488 個)
2-2-0	9.04%　(356 個)
1-1-1	3.05%　(120 個)
1-1-2	8.31%　(327 個)
1-1-0	9.17%　(361 個)
0-0-0	12.52%　(493 個)
0-0-2	8.18%　(322 個)
0-0-1	15.14%　(596 個)
0-1-2	11.58%　(456 個)

4.2 評価指標

ラベル付けの評価指標としてまず正答率を求めた．正答率は付与された関連性ラベルの正しさを評価する指標であり，今回はマクロ平均とマイクロ平均の両方を求めた [19]．クエリの集合を Q，各クエリを q，q におけるのシステムの正解数を $CorrectCount(q)$，q におけるバーティカル数を $VerticalCount(q)$，評価したクエリの数を N とした時，それぞれの算出方法を以下に示す．

$$\text{マクロ平均} = \frac{1}{N} \sum_{q \in Q} \frac{CorrectCount(q)}{VerticalCount(q)}$$

$$\text{マイクロ平均} = \frac{\sum_{q \in Q} CorrectCount(q)}{\sum_{q \in Q} VerticalCount(q)}$$

次に $nDCG$ を求めた [19]．$nDCG$ はシステムの出力結果の DCG をシステムが出力しうる最大の DCG で正規化したものであり，システムが出力した順位付けが関連性の度合い順であるかを評価する指標である．算出方法を以下に示す．

$$DCG@k = \sum_{i=1}^{k} \frac{2^{rel} - 1}{\log_2(i+1)}$$

$$nDCG@k = \frac{DCG@k}{idealDCG@k}$$

ここで k は上位何件のバーティカルを用いて DCG を計算するかを示し，今回は出力されたバーティカルの数とした．そのため今回の実験ではクエリ毎に k の値は異なる．また，rel はバーティカルの関連性を示す数値であり,「高適合」,「適合」,「不適合」のラベルに対し 2，1，0 とした．$idealDCG@k$ をシステムが出力しうる最大の $DCG@k$ とする．

4.3 ベースライン

本研究では比較対象として 2 つのシステムを構築した．一つはラベルをランダムに出力するシステムであり，一つはバーティカル情報が付与されているページのクリック情報を用いて，それぞれのバーティカルのクリック数の合計が多いものを順番に出力するシステムである．ここで，今回はクリック数が最大値の 2/3 以上のバーティカルを「高適合」，1/3 以上のものを「適合」，それ以外を「不適合」として出力した．前者を random，後者を click-num とそれぞれ表記して評価を記載する．

4.4 評 価 結 果

URL-List を用いた場合の正答率の結果を表 4，5，Vertical-List を用いた場合の正答率の結果を表 6，7 に示す．正答率はマクロ平均，ミクロ平均共に click-num の結果が一番良いものとなった．選好グラフを作成した手法の中では Vertical-List を用いて R6 と model2 において選好グラフを作成し，WeightedPageRank でソートしたものが一番良い結果であった．

次に URL-List，Vertical-List を用いた場合の $nDCG$ の結果を表 8，9 にそれぞれ示す．$nDCG$ は URL-List を用いて R6 と model1 において選好グラフを作成し，PageRank を用いてソートを行ったものの結果が一番良い結果となった．この結果から，選好グラフを R6 において作成した手法はバーティカルのランクリストの出力結果は click-num よりも良いが，その分割が上手くいっていないことがわかる．また，確率モデルや扱うバーティカル要素に着目すると扱うバーティカル要素や確率モデルによって優劣は異なるがあまり差がないことが分かる．URL-List，Vertcal-List を用いた手法において $nDCG$ の値が最高のものをそれぞれ URL-List-Best，Vertical-List-Best とする．

表 4　URL-List を用いた提案手法の正答率のマクロ平均

random	33.14%		
click-num	**46.30%**		
利用ルール	Score	PageRank	WeightedPageRank
R1	39.30%	**42.01%**	41.48%
R2	35.77%	37.50%	39.88%
R3	38.47%	41.57%	41.54%
R4	38.01%	39.63%	40.27%
R5	36.24%	37.95%	39.83%
R6-model1	38.27%	41.28%	41.13%
R6-model2	35.80%	40.78%	39.44%
R6-model3	36.14%	40.45%	39.44%

4.5 統計的検定

システム間の有意差について調べるため，click-num，URL-List-Best，Vertical-List-Best の 3 つのシステムの出力結果について試行回数を 5000 回としたランダム化 Tukey HSD 検定を行った [19]．正答率のマクロ平均，$nDCG$ のランダム化 Tukey HSD 検定の結果の p 値を表 10 に示す．一般に p 値が 0.05 以下の時，システム間の結果は有意であるとされている．結果を見ると正答率のマクロ平均 (絶対評価) では click-num が提案手法よりも優れ，$nDCG$(相対評価) では提案手法の方が click-num よりも優れていることが統計的に有意であると言え，提案手法同士では有意な差はないと言える．

表 5　URL-List を用いた提案手法の正答率のマイクロ平均

random	33.35%		
click-num	**46.69%**		
利用ルール	Score	PageRank	WeightedPageRank
R1	38.32%	**41.25** %	40.56%
R2	34.11%	35.59%	38.63%
R3	37.34%	41.15%	41.24%
R4	36.83%	39.07%	39.57%
R5	34.44%	35.94%	38.63%
R6-model1	37.33%	40.46%	40.41%
R6-model2	33.93%	39.62%	38.35%
R6-model3	34.62%	39.60%	38.66%

表 6　Vertical-List を用いた提案手法の正答率のマクロ平均

random	33.14%		
click-num	**46.30%**		
利用ルール	Score	PageRank	WeightedPageRank
R1	39.42%	39.86%	40.46%
R2	35.77%	34.91%	36.25%
R3	35.18%	34.99%	34.91%
R4	35.35%	35.35%	35.35%
R5	33.54%	33.54%	33.54%
R6-model1	40.43%	42.90%	42.84%
R6-model2	40.83%	41.84%	**43.19%**
R6-model3	41.27%	41.81%	42.71%

表 7　Vertical-List を用いた提案手法の正答率のマイクロ平均

random	33.35%		
click-num	**46.69%**		
利用ルール	Score	PageRank	WeightedPageRank
R1	39.07%	39.83%	40.56%
R2	34.75%	33.43%	35.10%
R3	35.15%	34.77%	34.65%
R4	33.53%	33.53%	33.53%
R5	31.98%	31.98%	31.98%
R6-model1	40.23%	42.77%	43.05%
R6-model2	40.26%	41.30%	**43.23%**
R6-model3	40.92%	41.25%	42.72%

表 8　URL-List を用いた提案手法の $nDCG$

random	0.7550		
click-num	0.8430		
利用ルール	Score	PageRank	WeightedPageRank
R1	0.8307	0.8208	0.8215
R2	0.8536	0.8355	0.8434
R3	0.7868	0.7949	0.7951
R4	0.8233	0.8149	0.8193
R5	0.8511	0.8349	0.8446
R6-model1	0.8630	**0.8713**	0.8671
R6-model2	0.8623	0.8707	0.8590
R6-model3	0.8617	0.8707	0.8662

表 9　Vertical-List を用いた提案手法の $nDCG$

random	0.7550		
click-num	0.8430		
利用ルール	Score	PageRank	WeightedPageRank
R1	0.8143	0.8129	0.8158
R2	0.7900	0.7680	0.7792
R3	0.5944	0.5858	0.5852
R4	0.6877	0.6750	0.6878
R5	0.7980	0.7818	0.7831
R6-model1	0.8482	0.8657	0.8631
R6-model2	0.8570	0.8640	**0.8703**
R6-model3	0.8576	0.8640	0.8699

表 10　ランダム化 Tukey HSD 検定の結果

システム対	正答率のマクロ平均の p 値	$nDCG$ の p 値
click-num，URL-List-Best	0	0
click-num，Vertical-List-Best	0.0034	0
URL-List-Best，Vertical-List-Best	0.1026	0.979

4.6 分　　析

4.6.1 正解データとの比較

click-num，URL-List-Best，Vertical-List-Best における正解データと出力結果の分布を表 11 に示す．click-num は「不適合」，URL-List-Best は「高適合」に出力が偏っており，Vertical-List-Best は他の 2 つに比べると偏りは少ないことが分かる．URL-List-Best の出力に「高適合」が多い理由は，各バーティカル毎のページが多く，相対的にクリックのデータが疎であることである．1 つのバーティカルに属するノードが多く存在することがあり，クリック数が少なくともクリックされていればそのバーティカル要素が上位に入ってしまう．すると，分割する際にも他のバーティカル要素との差が大きくなり，「高適合」のラベルが多くなることが分かった．その他の選好ルールに着目したところ，R1 と R3 はエッジの張られるバーティカル要素が 1 クリックにつき 1 つであることからグラフ全体のノード数は少なくなり，「高適合」の割合は多いものの他の選好ルールを用いた場合よりも偏りは少ないことが分かった．この結果，正答率が少し高くなったと考えられる．

Verical-List を用いた出力は，バーティカルの偏りは一番少なくなったが評価結果としてはベースラインの方が良い結果となった．Vertical-List を用いた場合，バーティカル要素の数が 2 となるクエリも存在し，そうした場合に今回はクリックが多いものから強制的に「高適合」，「適合」のラベルが付与される．そのため「不適合」が正解に含まれているものは精度が落ちる結果となっている．「高適合」「不適合」のラベルを順に付与したとしても改善される見込みはなく，重みの割合などからラベル間の適合度の差を判断することも重要であると考えられる．選好ルールに着目すると R6 を適応した正答率が高い．バーティカル要素が少なくなったことでノードの重み和の過剰な増加が少なくなっていることが原因だと考えられる．R2〜R5 では逆に重み和が少ないためにソート時に誤った結果となるものも見られた．URL-List を用いた場合よりも全体的に正答率が下がっている 1 要因であると考えられる．

また，正解データの中に「高適合」のラベルの付与されたバーティカルを持たないクエリが 248 個存在することわかった．click-num も選好グラフを用いた手法も「高適合」がある前提であるため，全体的に正答率が低い原因であると考えられる．Vertical-List を用いた場合の問題と合わせて考えると，バーティカルに対する関連性ラベルの自動付与はソートしたものを分割するのではなく，各バーティカルにそれぞれ関連性判定を行えば結果の向上に繋がるのではないかと考えられる．

4.6.2 エラーの傾向分析

バーティカル毎のエラーの傾向を前述の 2 つのシステムについても分析した．不正解だったラベルの数，正解が「高適合」で出力が「不適合」の数，正解が「不適合」で出力が「高適合」の数をそれぞれバーティカル毎に集計したものを表 12，13，14 にそれぞれ示す．バーティカル毎の不正解数を比較するとベースラインと URL-List-Best の比較では URL-List-Best が 4 勝 6 敗 2 分と負け越している．さらに，単純に「高適合」が多いバーティカルのみで優っているため，バーティカルの違いによる特性ではないと考えられる．

加えて，$nDCG$ が 1 で正答率が 0 であるクエリとラベルの特徴について調べた．クエリと属するバーティカルに関しては特に傾向が得られなかったが，ラベルに関して見てみると全てのクエリに対する正解に「高適合」ラベルがなく，出力に「高適合」が多いことが分かった．また，$nDCG$ が 1 で正答率が 0 に近いクエリに関しても同様の傾向が見られた．

最後に，アノテータのラベル付けと出力結果を比較した．まず，リアルタイムやニュースのバーティカルにおいて時事的なクエリによるエラーが多く発生していることが分かった．例えば「パルミラ遺跡」は利用したデータの収集された 2016 年 3 月現在テロにより破壊され関心が高まった．そのためクリック数はニュースやリアルタイムに集まっているが，正解ラベルは両方とも「不適合」で画像や辞書が「高適合」となっていた．その他にもスキャンダルが報じられた芸能人の名前などが同様の結果となっている．エラー率の高い質問サイトに関しても分析をしたが，扱ったデータが主に固有表現からなるヘッドクエリであるため情報要求を詳細化する語が含まれておらず，クエリに付加情報がなく人間にとっても難しいタスクであると考えられる．

表 11　分　　布

		click-num			URL-List-Best			Vertical-List-Best		
		高適合	適合	不適合	高適合	適合	不適合	高適合	適合	不適合
正解データ	高適合	712	174	376	1002	204	56	705	408	149
	適合	495	194	575	769	385	110	528	470	266
	不適合	320	159	932	686	519	206	321	563	527

5. まとめと今後の課題

本研究では，選好グラフを用いたバーティカルに対する関連性ラベルの自動付与を行い，評価・分析した．実験の結果として正答率はクリック数から単純に求めたベースラインにも及ばぬ結果となったが，バーティカルのランクリストに関する $nDCG$ の結果は選好グラフを用いた手法でベースラインより優れた結

表 12　バーティカル毎の不正解の割合

バーティカル	click-num	URL-List-Best	Vertical-List-Best
画像	57.68%	51.12%	54.19%
レシピ	14.89%	8.51%	14.89%
オークション	48.36%	64.67%	53.26%
辞書	47.25%	65.93%	52.75%
質問サイト	60.89%	61.74%	58.23%
動画	55.68%	42.74%	54.90%
ニュース	54.17%	64.84%	61.01%
地図	39.29%	39.29%	32.14%
リアルタイム	44.22%	77.21%	61.77%
ショッピング	41.77%	43.03%	39.87%
タレント情報	63.68%	51.05%	69.47%
周辺地域情報	29.55%	29.55%	25.00%

表 13　正解が「高適合」で出力が「不適合」の数

バーティカル	click-num		URL-List-Best		Vertical-List-Best	
画像	33.66%	(137 個)	7.37%	(30 個)	15.23%	(62 個)
レシピ	6.98%	(3 個)	0%	(0 個)	0	(0 個)
オークション	61.90%	(13 個)	4.76%	(1 個)	38.10%	(8 個)
辞書	33.33%	(8 個)	0%	(0 個)	16.67%	(4 個)
質問サイト	33.50%	(65 個)	5.15%	(30 個)	3.60%	(7 個)
動画	32.94%	(56 個)	1.76%	(3 個)	18.82%	(32 個)
ニュース	25.42%	(30 個)	4.24%	(5 個)	6.78%	(8 個)
地図	33.33%	(8 個)	4.17%	(1 個)	16.67%	(4 個)
リアルタイム	39.53%	(17 個)	6.98%	(3 個)	16.28%	(7 個)
ショッピング	17.81%	(13 個)	1.36%	(1 個)	4.11%	(3 個)
タレント情報	21.24%	(24 個)	0%	(0 個)	11.50%	(13 個)
周辺地域情報	6.25%	(2 個)	0%	(0 個)	3.13	(1 個)

表 14　正解が「不適合」で出力が「高適合」の数

バーティカル	click-num		URL-List-Best		Vertical-List-Best	
画像	37.00%	(37 個)	63.00%	(63 個)	45.00%	(45 個)
レシピ	0%	(0 個)	0%	(0 個)	0%	(0 個)
オークション	9.17%	(10 個)	31.19%	(34 個)	11.00%	(12 個)
辞書	18.60%	(8 個)	27.91%	(12 個)	16.28%	(7 個)
質問サイト	22.13%	(52 個)	39.14%	(92 個)	18.72%	(44 個)
動画	29.17%	(7 個)	62.5%	(15 個)	41.67%	(10 個)
ニュース	23.03%	(76 個)	45.45%	(150 個)	23.03%	(76 個)
地図	33.33%	(1 個)	33.33%	(1 個)	33.33%	(1 個)
リアルタイム	23.10%	(116 個)	53.19%	(267 個)	22.91%	(115 個)
ショッピング	6.98%	(3 個)	30.23%	(13 個)	11.63%	(5 個)
タレント情報	47.37%	(9 個)	63.15%	(12 個)	26.31%	(5 個)
周辺地域情報	33.33%	(1 個)	66.66%	(2 個)	33.33%	(1 個)

果を出力するものもあった．選好グラフのうちの幾つかの手法は関連性によるソートは上手くいっているのだが，ラベルの分割が上手くいっていないと言える．分析の結果，付与される適合ラベルの偏りや全てのバーティカルに「高適合」のラベルが必須ではないといった問題が見られた．

今後の課題としてはラベル付けをする際の効果的な手法の提案や最高のラベルの判定，アノテータを増やした実験の再評価が考えられる．また，今回は実験に利用するデータに制約が多いことも懸念事項の一つであると考えられるため，実験に利用するデータの整形なども課題であると言える．最後に，ニュースやリアルタイムのバーティカルに対するラベル付けの分析結果から，時事的なクエリの判定などが今後の応用的なタスクとして挙げられる．

文　献

[1] Arguello, Jaime, et al. “Sources of evidence for vertical selection.” Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2009.

[2] 中渡瀬 秀一，大山 敬三，“サーチエンジンクエリ分析による情報タイプの抽出:Web 検索利用者の情報要求に即した Web 情報空間の再構成に向けて”，人工知能学会全国大会論文集 25，2011.

[3] Jansen, Bernard J., et al. “Real life information retrieval: A study of user queries on the web.” ACM SIGIR Forum. Vol. 32. No. 1. ACM, 1998.

[4] Jansen, Bernard J., Mimi Zhang, and Amanda Spink. “Patterns and transitions of query reformulation during web searching.” International Journal of Web Information Systems 3.4 2007.

[5] Jansen, Bernard J., Danielle L. Booth, and Amanda Spink. “Patterns of query reformulation during Web searching.” Journal of the american society for information science and technology 60.7 2009.

[6] Song, Yang, et al. “Exploring and exploiting user search behavior on mobile and tablet devices to improve search relevance.” Proceedings of the 22nd international conference on World Wide Web. ACM, 2013.

[7] Thomas Demeester, Dolf Trieschnigg, Dong Nguyen, Ke Zhou, Djoerd Hiemstra, “Overview of the TREC 2014 Federated Web Search Track”, Proceedings of TREC 2014, 2015.

[8] Takehiro Yamamoto, Yiqun Liu, Min Zhang, Zhicheng Dou, Ke Zhou, Ilya Markov, Makoto. P. Kato, Hiroaki Ohshima, Sumio Fujita. “Overview of the NTCIR-12 IMine-2 Task”, Proceedings of NTCIR-12, 2016.

[9] Jin, Shan, and Man Lan. “Simple May Be Best-A Simple and Effective Method for Federated Web Search via Search Engine Impact Factor Estimation.” Proceedings of TREC 2014. 2015.

[10] Emanuele Di Buccio, Massimo Melucci, “University of Padua at TREC 2014: Federated Web Search Track”, Proceedings of TREC 2014, 2015.

[11] Feng Guan, Shuiyuan Zhang, Chunmei Liu, Xiaoming Yu, Yue Liu, Xueqi Cheng, “ICTNET at Federated Web Search Track 2014”, Proceedings of TREC 2014, 2015.

[12] Agrawal, Rakesh, et al. “Generating labels from clicks.” Proceedings of the Second ACM International Conference on Web Search and Data Mining. ACM, 2009.

[13] Xu, Jingfang, et al. “Improving quality of training data for learning to rank using click-through data.” Proceedings of the third ACM international conference on Web search and data mining. ACM, 2010.

[14] Hang Li, “A Short Introduction to Learning to Rank” IEICE TRANS. INF. & SYST. VOL.E94D, 2011.

[15] Thorsten Joachims, Laura A. Granka, Bing Pan, Helene Hembrooke, Filip Radlinski, and Geri Gay. “Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search”, ACM Trans. Inf. Syst, 25(2), 2007.

[16] Edward Cutrell and Zhiwei Guan. “What are you looking for? An eye-tracking study of information usage in Web search”, In CHI, pages 407416, 2007.

[17] Guan, Zhiwei, and Edward Cutrell. “An eye tracking study of the effect of target rank on web search.” Proceedings of the SIGCHI conference on Human factors in computing systems. ACM, 2007.

[18] Brain, Sergey, and Lawrence Page. “The anatomy of a large scale hypertextual web search engines.” Computer Networks and ISDN System30 (1-7), 107-117, 1998.

[19] 酒井哲也. “情報アクセス評価方法論〜検索エンジンの進歩のために〜”，コロナ社，2015.