検索クエリログのセッション情報を利用した属性語句抽出

関口裕一郎† 田中 智博† 内山 匡† 藤村 滋†† 望月 崇由††
鈴木 智也††

† 日本電信電話株式会社 NTT サイバーソリューション研究所 〒 239–0847 神奈川県横須賀市光の丘 1–1
†† NTT レゾナント株式会社 〒 108–0023 東京都港区芝浦 3–4–1 グランパークタワー 8F
E-mail: †{sekiguchi.yuichiro,tanaka.tomohiro,uchiyama.tadasu}@lab.ntt.co.jp,
††{fujimura,mochizuki,to.suzuki}@nttr.co.jp

あらまし ショッピングサイトの商品探索やニュースサイトでの記事探索など，ディレクトリ構造に基づいたサイトナビゲーションは数多く存在する．しかし日々新たなコンテンツやジャンルが生まれる中で，これらの階層構造を維持するのは非常にコストのかかる作業である．本論文ではウェブ検索エンジンに入力される検索クエリのログを用いて，あるカテゴリの下位に入るような語句の抽出を行い，ユーザの探索目的にあったカテゴリ構成を自動抽出することを目的とする．具体的には，商用検索エンジンに入力された検索クエリログから取得されるユーザが入力するクエリの遷移に注目し，ある特定のクラスに所属する検索クエリ語句に対して絞り込みの意図を持って追加された語句を取得することにより，各クラスの属性語句を取得する手法について述べる．また商用検索エンジンに入力された約 2 億のクエリ遷移情報を用い，8 クラスに対して各々10 のシードインスタンスを用いて提案する属性語句抽出の評価を行った．その結果，Precision@20 において最高で 0.95，最低で 0.60，平均で 0.76 の精度を確認した．
キーワード Web マイニング，検索クエリログ，情報抽出

Acquisition of Class Attributes from Search Engine Query Logs

Yuichiro SEKIGUCHI†, Tomohiro TANAKA†, Tadasu UCHIYAMA†, Shigeru FUJIMURA††,
Takayoshi MOCHIZUKI††, and Tomoya SUZUKI††

† NTT Cyber Solutions Laboratories, NTT Corporation Hikarinooka 1–1, Yokosuka-shi, Kanagawa, 239–0847 Japan
†† NTT Resonant Inc. Granpark Tower 8F, Shibaura 3–4–1, Minato-ku, Tokyo, 108–0023 Japan
E-mail: †{sekiguchi.yuichiro,tanaka.tomohiro,uchiyama.tadasu}@lab.ntt.co.jp,
††{fujimura,mochizuki,to.suzuki}@nttr.co.jp

Abstract There are many services having a directry like navigation interfaces, such as artile lists of news sites, item directories of e-commerce sites. These directiries requires a lot of costs to construct and maitain them useful. The aim of this paper is to extract attribute words of each category contained in those directiory and make the cost down. We describe the way to extract attribute words from commercial search engine logs by extracting following queries of specific class words. We evaluate our method using almost 200 million search query sequece data. The reluts shows that the highest precision@20 value is 0.95, lowest is 0.60, and average is 0.76.
Key words Web Mining, Search Engine Log, Information Retrieval

1. はじめに

検索エンジンの普及によって，ウェブ全体においてはディレクトリ型の検索は減ってきた．しかしニュースサイトにおける記事の閲覧や，EC サイトによる商品の閲覧といった，各階層に所属するアイテムを眺めながら目的のコンテンツを探し出す用途のサービスにおいては，依然としてディレクトリ構造を用いたナビゲーションが一般的に用いられている．

一般的にディレクトリ構造を構築するのはコストのかかる作業である．特にウェブの世界では日々コンテンツが追加されあらたなジャンルが発生するため，ディレクトリ構造を使いやすい形に維持をするのにも人手によるメンテナンスが必要となる．

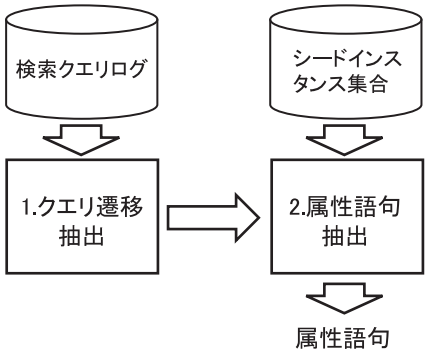

図 1　処理の流れ
Fig. 1　The image of our methd.

本論文ではこのようなディレクトリ構造の構築・維持のコストを低減するため，あるカテゴリの下層にくる語句を自動抽出する手法を提案する．

これまでにもセマンティックウェブの分野や，テキストマイニングの分野において，意味的に下位の語句を抽出する技術の研究が行われてきた．例えば，ウェブ文書等の大量のコーパスを用いて語句の概念を表すベクトルを構築し，そのカルバックライブラー距離を用いて意味的な階層の上下関係を用いる手法 [1] や，検索エンジンの結果をもちいて各語句間の関係性を抽出する手法 [2] が提案されている．これらのウェブ文書を用いた手法は,「スポーツ」に対して「野球」が下位になるといった，語句の文中の意味における上下関係を抽出することが可能である．しかし EC サイトなどにおいて探索にもちいるディレクトリ構造は，ファッションブランドのカテゴリの下には,「バッグ」「指輪」「アクセサリ」といった商品タイプによる分類が必要になるといったことがあり，意味的な上下関係が必ずしも有用ではない．

本論文では，商用検索エンジンに入力された検索クエリログからユーザが入力した検索クエリの遷移を抽出することにより，実際にユーザが絞り込みに用いている語句を下位語句として抽出する手法を提案する．検索クエリログからユーザが探索する際に用いる語句を抜き出すことによって，ユーザが考えるコンテンツの絞り込みのイメージにあったディレクトリ構造を構築可能になると考えられる．

2. 提 案 手 法

提案手法の概要を図 1 に示す．本手法はディレクトリ構造の 1 要素となるカテゴリをクラス，クラスに所属する商品名等の固有名詞で表される事柄をインスタンス，クラスに所属するインスタンスに共通する絞り込み語句を属性語句と呼び，シードとなるいくつかのインスタンス語句を元に．属性語句を抽出する．例えば「ファッションブランド」クラスの場合，各ブランドの名称がインスタンスにあたる．抽出対象となる属性語句は,「コート」「バッグ」といったブランドの下で絞り込み探索をするのに用いられる語句となる．

提案手法は 2 段階の処理によって行われる．まず検索クエリログから，ユーザの入力する検索クエリの遷移の情報を抽出する．次にクエリ遷移から，シードインスタンスの絞り込みに使われるている語句を属性語句として抽出する．

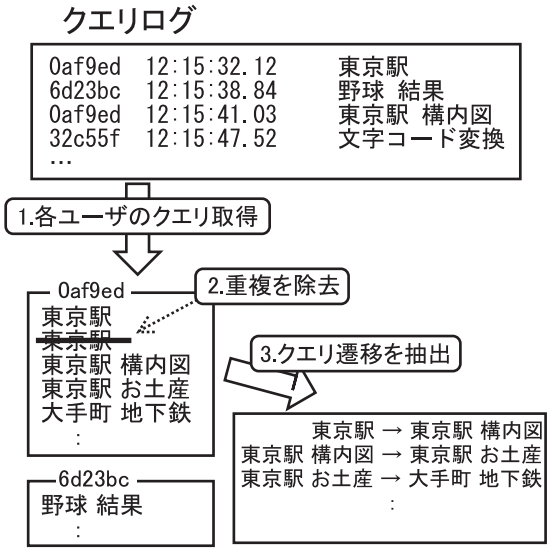

図 2　クエリ遷移情報の抽出手順
Fig. 2　The process of extracting query sequences.

以下，各段階の処理について説明を行う．

2.1　検索クエリログからのクエリ遷移情報の抽出

本手法は，商用のウェブ検索エンジンに入力された検索クエリのログを処理対象とする．利用した検索ログには既存の検索クエリログ調査 [4] に用いられているログと同様に，以下の 3 つのデータが含まれている．

（1） 検索クエリ：ユーザによって入力されたクエリ文字列

（2） 検索時刻：ウェブサーバに記録される，検索クエリが入力された時刻

（3） ユーザ ID：サーバによって割り振られる，匿名化されたユーザ識別 ID

クエリ遷移情報の抽出手順を図 2 に示す．

まずユーザ ID 毎に入力された検索クエリを時刻順に取得することによって，各ユーザが入力した検索クエリの遷移を取得する．次に得られた検索クエリの遷移から，同じクエリが連続して入力されていた場合に生じる重複を除去する．最後に時系列順に 2 つづつ検索クエリの組を抽出し，クエリ遷移として取り扱う．

例えば,「東京駅，東京駅，東京駅 構内図，大手町 地下鉄，大手町乗り換え，大手町乗り換え，八重洲」という順番にクエリが入力されていた場合，連続して入れられている「東京駅」「大手町乗り換え」というクエリは除去されて,「東京駅 → 東京駅 構内図」「東京駅 構内図 → 大手町 地下鉄」……といったクエリ遷移が抽出される．この際，クエリ遷移における先に入力されたクエリを先行クエリ，後に入力されたクエリを後続クエリと呼ぶこととする．

2.2　属性語の抽出

属性語句の抽出手順を図 3 に示す．

表 1　各クラスのシード一覧
Table 1　The list of seed words

クラス	シード集合
地名	大阪, 東京, 京都, 札幌, 横浜, 名古屋, 福岡, 仙台, 北海道, 広島
車名	BMW, bB, オデッセイ, レクサス, トラック, コロナ, ベンツ, プリウス, エスティマ, クラウン
スポーツ	ゴルフ, サッカー, 野球, テニス, スポーツ, バドミントン, 卓球, マラソン, ソフトボール, 陸上
番組名	あいのり, ガンダム, ドラゴンボール, 24, エヴァンゲリオン, 相棒, マクロス, クレヨンしんちゃん, プリキュア, ヘキサゴン
ギャンブル	パチンコ, 宝くじ, toto, 競馬, パチスロ, 麻雀, スロット, 競艇, 競輪, ナンバーズ
映画	タクシー, リング, ハリーポッター, 007, バットマン, マトリックス, ゴジラ, スパイダーマン, カーズ, タイタニック
芸能人	EXILE, NEWS, ジャニーズ, SMAP, B'z, 東方神起, AKB48, コブクロ, ミスチル, aiko
作家	村上春樹, 宮沢賢治, 三島由紀夫, 井上雄彦, 宮部みゆき, 手塚治虫, 夏目漱石, 京極夏彦, 相田みつお, 内田春菊

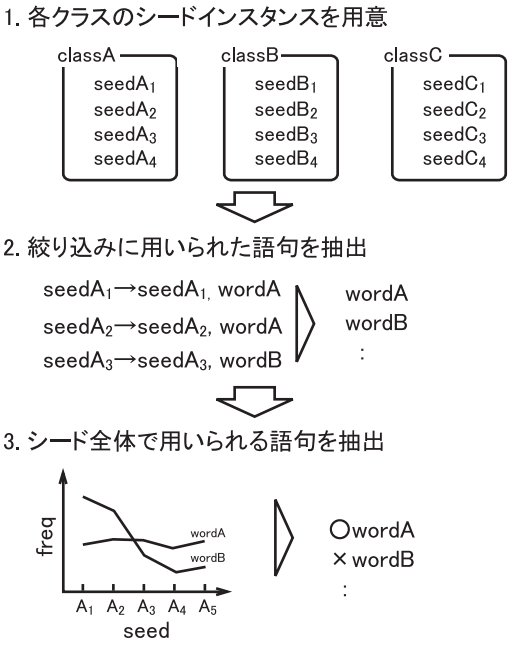

図 3　属性語句抽出の処理の流れ
Fig. 3　The process of attribute extraction.

最初に抽出対象となるクラス毎のシードとなるインスタンスの集合を用意する．例えば「駅」クラスであれば,「東京駅」「新宿駅」「新大阪駅」等がシードインスタンスとして考えられる．クラス c_i のシードインスタンスの集合を $\mathcal{S}_i = \{s_{i1}, s_{i2}, ..., s_{in}\}$ と記述する．

次に各シードインスタンスについて，ユーザがシードインスタンスでの検索結果に対して，さらに絞り込もうとして用いた語句を属性候補語句として抽出する．例えばシードインスタンスが「東京駅」であった場合,「東京駅→東京駅 構内図」や「東京駅→東京駅 お土産」といった，先行クエリがシードインスタンスであり後続クエリがシードインスタンスと他の語句からなるクエリ遷移を選び出し，後続クエリで追加された語句を抽出することによって属性候補語句が得られる．検索クエリログに含まれる全てのクエリ遷移中で，語句 w_k がシードインスタンス s_{ij} の絞り込み語句となっている回数を $freq(s_{ij}, w_k)$ と表記することとする．

最後に全てのシードクエリにおいて偏りなく用いられている属性候補語句を，クラスの属性語句として抽出する．これは特定のシードインスタンスのみに特徴的に用いられる絞り込み語句を属性語句として抽出すると，クラス全体に共通する属性語句とならずノイズとなるためである．例えば「東京駅」「新宿駅」「新大阪駅」がシードインスタンスであった場合,「構内図」「時刻表」「終電」といった語句はどのシードに対しても用いられクラス全体の属性語句として適切である．一方「八重洲口」といった語句は「東京駅」のみで用いられるのでクラス全体の属性語句としてふさわしくない．

この偏りのなさを数値化するため，w_k が絞り込み語句として使われている際に，どのシードインスタンスの絞り込み語句となっているかの確率 $P(s_{ij}, w_k)$ の平均情報量を用いて，属性候補語句 w_k の属性らしさを表すスコア $Score_i(w_k)$ を求める．この際，単純にはシードインスタンス間における偏りを見ればよいので，以下の式のように平均情報量を計算すればよい．

$$Score_i(w_k) = -\sum_j P(s_{ij}, w_k) log P(s_{ij}, w_k) \quad (1)$$

$$P(s_{ij}, w_k) = \frac{sf(s_{ij}, w_k)}{\sum_j sf(s_{ij}, w_k)} \quad (2)$$

(3)

しかし実際にはシードインスタンスごとにクエリログ中での出現頻度は大きく異なるため，$sf(s_{ij}, w_k)$ の値もその影響をうけて大きく変動してしまっている．この影響をさけるために，情報量計算の確率に $sf(s_{ij}, w_k)$ をそのまま用いるのではなく，一度各シードインスタンスの中で w_k が絞り込みで用いられる確率 $sf_p(s_{ij}, w_k)$ を求めてから，その値を元に改めて補正した確率 $P'(s_{ij}, w_k)$ を求める．以上の考え方から，下の式 (3) を用いて属性スコアを計算する．

$$Score_i(w_k) = -\sum_j P'(s_{ij}, w_k) log P'(s_{ij}, w_k) \quad (4)$$

$$P'(s_{ij}, w_k) = \frac{sf_p(s_{ij}, w_k)}{\sum_j sf_p(s_{ij}, w_k)} \quad (5)$$

$$sf_p(s_{ij}, w_k) = \frac{sf(s_{ij}, w_k)}{\sum_{w \in \mathcal{W}} sf(s_{ij}, w_k)} \quad (6)$$

3. 評価実験

提案手法を用いて，8 クラスについて属性語句の抽出を行い，

表 2　地名クラスにおける抽出された属性語の一覧
Table 2　The list of attribute words belong to the place class

rank	word	score	rank	word	score
1	ホテル	2.240	11	風俗	2.131
2	おすすめ	2.237	12	地図	2.118
3	グルメ	2.233	13	買い物	2.114
4	観光	2.227	14	食事	2.113
5	温泉	2.220	15	お土産	2.104
6	観光地	2.219	16	ランチ	2.104
7	病院	2.184	17	歴史	2.098
8	不動産	2.150	18	おみやげ	2.090
9	土産	2.149	19	求人	2.087
10	イベント	2.132	20	宿泊	2.067

表 3　作家クラスにおける抽出された属性語の一覧
Table 3　The list of attribute words belong to the author class

rank	word	score	rank	word	score
1	作品	1.849	11	イラスト	1.015
2	写真	1.678	12	猫	0.995
3	画像	1.352	13	年表	0.969
4	あらすじ	1.345	14	ランキング	0.925
5	wiki	1.247	15	サーチ	0.825
6	小説	1.078	16	感想	0.693
7	新刊	1.068	17	父	0.693
8	本	1.047	18	直木賞	0.693
9	映画	1.031	19	新作	0.693
10	女	1.019	20	人気	0.693

その精度を評価した．

3.1　実験条件

表 1 に示す 8 クラスについての抽出を行った．各クラスのシードインスタンスはそれぞれ 10 インスタンスとし，検索クエリログ中の頻度上位クエリから各クラスのインスタンスにふさわしい語句を人手で抽出して用意した．

処理対象には商用の検索エンジンに入力された検索エンジンログを用いた．2.1 で述べた手法を用いて 190,715,322 件のクエリ遷移を抽出し，それを用いて属性語句の抽出を行った．

表 4 に得られた各クラスの属性上位 20 件を示す．また，表 2 に,「地名」クラスに対して抽出した結果の上位 20 件とそのスコアを，表 3 に,「作家」クラスに対して抽出した結果の上位 20 件とスコアを示す．

3.2　評価基準

得られた各クラスの属性語句上位 20 語に対して，人手で適・不適の判別を行い，Precision@5, Precision@10, Precision@20 で精度を算出した．

適・不適の判別においては，シードインスタンス全てに対して属性となりうる語句を適とすることとした.「作家」クラスにおける「小説」のような，一部のインスタンス (この場合であれば「村上春樹」等の小説家) のみにあてはまる属性は不適とした．また「地名」クラスにおける「おすすめ」や「作家」クラスにおける「あらすじ」のような，属性語句と言うよりも属性を修飾するような語句についても不適とした．例えば「おすすめ」は「東京 ホテル おすすめ」といった使われ方で現れる語句であり,「あらすじ」は「村上春樹 新作 あらすじ」といった使われ方で現れる．どちらも属性語句である「ホテル」や「新作」に対するもう一段階意味階層が下に来る属性語句と考えられ，取得対象としているクラスの属性語句としては不適である．

表 5　各クラスにおける Precision@5,@10,@20
Table 5　Precision for each class.

クラス	precision@5	precision@10	precision@20
地名	0.80	0.90	0.95
車名	0.80	0.80	0.90
スポーツ	0.80	0.80	0.80
番組名	0.60	0.70	0.70
ギャンブル	0.80	0.70	0.60
映画	0.80	0.70	0.75
芸能人	0.80	0.90	0.80
作家	0.80	0.70	0.60

3.3　抽出精度

得られたを精度を表 5 に示す．

Precision@20 での値は最も高い「地名」クラスで 0.95，最も低い「作家」「ギャンブル」クラスで 0.6 となった．Precision@5, Precision@10, Precision@20 の間では，Precision@5 の値が最も悪かった「番組名」などをはじめ，クラスごとに値の上下関係に差が生じていた．

クラスによって精度に差が出たのは，クラス毎に抽出された上位 20 語のスコアに差が存在するためと考えられる．図 4 に Precision@20 と各クラスの上位 20 語のスコアを平均した値 (以下平均スコアと記す) の相関を示す．平均スコアと Precision@20 の間に正の相関があることが読み取れ，その相関係数は 0.772 となった．Precision@20 でのスコアが低い「作家」「ギャンブル」は，平均スコアにおいても 8 クラス中で下 2 つになっており，また Precision@5 から Precision@20 に向けて値が悪くなる傾向を持つ．そのため，スコア値を元に取得する属性語句の足きりを行うことによって，精度が向上する可能性があると考えられる．

クラスによって平均スコアにばらつきが生じているのは，シードインスタンスの均一性による影響と考えられる.「ギャンブル」クラスで不適な属性とされたうちの「ゲーム」「メーカー」「ネット」は，シードインスタンスの中でログ中での出現頻度の大きい語句に特有の属性語が上位に混ざっているものである．ログ中での出現数の近い語句をシードとして選ぶことにより対応可能であったと考えられる．また「作家」クラスで不適な属性とされた「小説」「直木賞」といった語句はシードインスタンスのうち大半となる小説家に特有の属性語となる.「作家」クラスを小説家，漫画家と細分化することにより対応可能と考えられる．

3.3.1　スコア算出時の出現頻度正規化の効果

2.2 でのべた式 (3) による効果を確認するため，シードインスタンスの出現頻度に対する正規化を行わない式 (1) でスコア計算をした場合の精度も求めた．その結果を表 6 に示す．

表 4　抽出された属性語の一覧
Table 4　The list of attribute words

クラス	属性語句
地名	ホテル, おすすめ, グルメ, 観光, 温泉, 観光地, 病院, 不動産, 土産, イベント, 風俗, 地図, 買い物, 食事, お土産, ランチ, 歴史, おみやげ, 求人, 宿泊
車名	中古, カタログ, 中古車, 車, 価格, ドレスアップ, 値引き, ナビ, トヨタ, エアロ, 試乗, 改造, ホイール, カスタム, 画像, 新車, 値段, カーナビ,cm, 評価
スポーツ	結果, イラスト, 大阪, 歴史, 北京オリンピック, 練習, 北京五輪, 放送, 福岡, ルール, 東京, 日本, オリンピック, 画像, 速報, ショップ, 試合, 上達, 女子, 五輪
番組名	ユーチューブ, 画像, 本, 無料動画, 壁紙, 写真, 無料,op, 主題歌, 動画, 2008, 曲, イラスト,wiki,you, 公式, ストーリー,dvd, 待ち受け,youtube
ギャンブル	ブログ, 動画, 初心者, 結果, ニュース, 画像, 初めて, 販売, 新潟, 予想, 攻略, ゲーム, 攻略法, メーカー, 購入, ランキング, プロ, 入門, ネット, 全国
映画	映画, 公式,dvd, サントラ, 画像, ゲーム,youtube, 曲, あらすじ, 登場人物, ポスター, 動画, 歌, イラスト, テレビ,wii, 公式サイト, 新作, 音楽, グッズ
芸能人	ランキング,dvd, チケット, 写真, 動画, 曲, ジャケット,cd, 壁紙, 新曲, pv, 画像, 歌, ライブ, ラジオ, アマゾン, アルバム, プロフィール,tube,you
作家	作品, 写真, 画像, あらすじ,wiki, 小説, 新刊, 本, 映画, 女, イラスト, 猫, 年表, ランキング, サーチ, 感想, 父, 直木賞, 新作, 人気

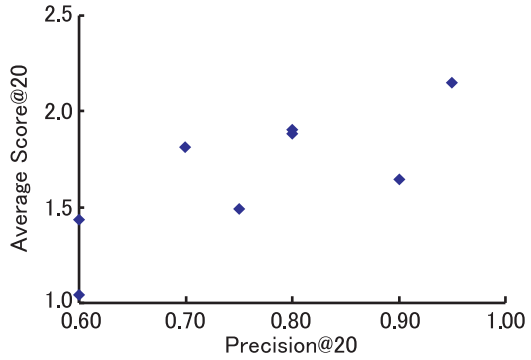

図 4　平均スコアと精度の相関
Fig. 4　Relations between average scores and precision@20.

表 6　頻度正規化なしでの Precision@5,@10,@20
Table 6　Precision without normalization.

クラス	precision@5	precision@10	precision@20
地名	1.00	1.00	1.00
車名	0.80	0.70	0.55
スポーツ	0.80	0.80	0.75
番組名	0.40	0.40	0.25
ギャンブル	0.20	0.40	0.35
映画	0.40	0.60	0.60
芸能人	0.80	0.60	0.55
作家	0.60	0.50	0.35

表 5 と比べると,「地名」クラスを除いて全体的に精度が落ちていることが分かる．誤抽出された属性語句としては，特定のシードインスタンスに関係する語句に加え，本来インスタンスとして取れるのが妥当な語句やシードインスタンスとして入力した語句自身が属性国として取れているクラスもあった．特に精度の低下が激しかった,「車種」クラスと「ギャンブル」クラスについての抽出結果を表 7 に示す．シードインスタンスであった「Bb」が含まれているのに加え,「当選番号」をはじめとして，特定のシードインスタンスとしか関係しない語句が多く含まれている．

表 7　頻度正規化なしでの抽出される属性語句の例
Table 7　Example of attributes without normalization.

rank	車名	ギャンブル
1	中古	予想
2	中古車	当選番号
3	トヨタ	エヴァンゲリオン
4	モデルチェンジ	攻略
5	価格	ゲーム
6	bmw	北斗の拳
7	新型	結果
8	bb	無料
9	燃費	動画
10	ハイブリッド	ビッグ

これらはシードインスタンスのログ中の出現頻度にばらつきがあるため，たまたま各シードインスタンスでの絞り込み語句としての使用回数が近くなってしまった語句がランダムに取得されてしまったことによると考えられる．

4. 関 連 研 究

商用の検索エンジンの検索クエリログを解析して，語句間の関係性などの知識を抽出する研究はいくつか行われている．

Yates らは検索クエリとその検索クエリにおける検索結果からユーザがクリックした URL を含むログを用いて，2 つの語句のクリック URL の重なり方を解析することにより，それらの意味的な上下関係を抽出する手法を提案している. [5]

小町らは，検索クエリログ中の文脈パターンを利用することにより，特定のクラスに属するインスタンスを，あらかじめ与えられたいくつかのインスタンスシードを元に自動的に収集する手法を提案している. [6] ここでいう文脈パターンは「JAL 時刻表」「JR 時刻表」といった複数語句からなる検索クエリを一般化した,「#+時刻表」といったクエリのパターンを用いる．

Pasca らは，検索クエリログ中の文脈パターンを用いることにより，特定のクラスに属する属性を，あらかじめ与えられたインスタンスシードと属性シードを元に自動的に収集する手法

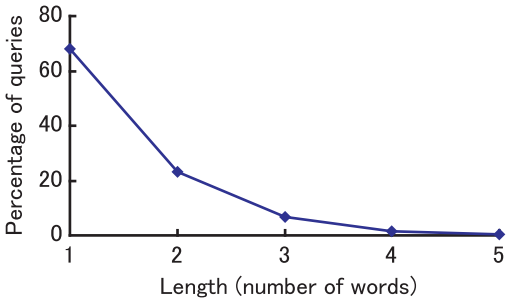

図 5　検索クエリの構成語数の割合
Fig. 5　Percentage of queries' lengths

を提案している. [3] [7] これは特定のクラスの属性語句を抽出するという点で，本研究との非常に関連が高い手法である．

この手法においては，インスタンスと属性を含む文脈パターンをクエリログから抽出する．例えば，与えられたシードインスタンス「東京」とシード属性「ホテル」を含む検索クエリ「東京 格安 ホテル」から,「¡シード¿ 格安 ¡属性¿」という文脈パターンを抽出する．その後抽出された文脈パターンが他のシードでも出現するかを確認した後に，文脈パターンの属性の位置に入る語句を新たな属性として抽出する．このようにインスタンスと属性 2 語を含んだ文脈パターンを抽出する必要があるため，最低でも 3 語以上クエリのみが処理対象となる．

しかし，今回実験に用いたクエリ遷移ログ中の先行クエリの構成語数の割合を求めると図 5 のようになり，1 語で構成されるクエリが全体の 68%を占め，3 語以上含むクエリは全体の 8.8%となる．Pasca らの報告 [3] によると，英語の検索エンジンに入力されたクエリのうち 1 語，2 語，3 語，4 語，5 語で構成されるクエリの割合が，それぞれ 14.5%, 31.3%, 26.5%, 13.2%, 7.5%となり，英語クエリにおける 3 語以上含むクエリの割合は 52.2%となっている．このことから，日本語検索は 1 語による検索がきわめて多くなる傾向があり，3 語以上の語句からなる検索クエリを必要とする文脈パターンを用いた手法は日本語クエリログにおいては有効性が低くなると考えられる．

一方，本論文で提案したクエリの遷移を利用する手法は少ない語数でも処理可能となるため，語数の少ない日本語クエリログに向いた手法であると考えられる．

実際に Pasca の手法 [3] で,「車名」「作家」クラスに対して属性語句を抽出した例を，表 8 に示す．シードインスタンスには評価実験に用いたのと同じ 10 語句を，シード属性には評価実験で得られた属性語句のうち適と判断された語句の上位 5 語句を用いた.「車名」クラスの様な検索ログ中における検索件数が比較的多いクラスについては，属性を表す語句が取れているのに対して,「作家」クラスのような検索件数が比較的少ないクラスについては「母親の正体」など，特定のシードインスタンスにしか当てはまらないような語句が多く上位に来てしまっているのが分かる．

5. まとめと今後の課題

検索クエリログから抽出されるクエリ遷移の情報を用いることによって，シードインスタンスからクラスの属性語句を抽出する手法を提案した．クエリ遷移中で絞り込み語句として用いられている語句を属性語句候補として抽出し，ログ中での出現頻度を正規化した平均情報量を用いることにより，属性語句をスコア付けする手法を提案した．

商用検索エンジンから得られた約 2 億のクエリ遷移情報に対して提案手法を適用し，8 クラスについて評価を行った結果，Precision@20 において，最大で 0.95，最低で 0.60，平均で 0.76 の精度が出ることを確認した．

今後は，今回の評価で得られたスコアと精度の正の相関関係に注目し，不正解な属性語句を自動で除去するための足きり手法について取り組みたい．また提案手法によって得られた属性語句を用いた，ディレクトリ構造の構築補助の効果を確認していきたい．

表 8　Pasca の手法によって抽出した属性語句の例
Table 8　Example of attributes extracted by Pasca's method.

rank	車名	作家
1	ドレスアップ	新刊
2	値引き	母親の正体
3	燃費	主題
4	新古車	同性
5	モデルチェンジ	新しい命
6	ハイブリット	pdf
7	ハイブリッド	読解
8	全長	親族
9	評価	天皇制の容認
10	サービスマニュアル	大阪書籍

文　献

[1] 別所 克人, 内山 俊郎, 片岡 良治, 単語・意味属性間共起に基づく単語間の階層関係の抽出, 信学技報, NLC, vol.106(518), NLC2006-92, Jun 2007.
[2] D. Bollegala, Y. Matsuo, M. Isizuka, Measuring Semantic Similarity between Words using Web Search Engines, Proc. of WWW2007, May 2007.
[3] M. Pasca, Organizing and Searching the World Wide Web of Facts Step Two: Harnessing the Wisdom of the Crowds, In proc. of WWW2007, pp 101-110, May 2007.
[4] B.d J. Jansen , A. Spink, An analysis of web searching by European AlltheWeb.com users, Information Processing and Management: an International Journal, v.41 n.2, pp.361-381, March 2005.
[5] R. Baeza-Yates, A. Tiberi, Extractiong Semantic Relations from Query Logs, Proc. of WWW2007, pp 76-85, 2007.
[6] 小町 守, 鈴木 久美, 検索ログからの半教師あり意味知識獲得の改善, 人工知能学会論文誌, vol. 23, no. 3, pp 217-225, 2008.
[7] M. Pasca, B. V. Durme, What You Seek is What You Get: Extraction of Class Attributes from Query Logs, Proc. of IJCAI07, pp. 2832-2837, 2007.