2005－DBS－137（Ⅱ）（46）
2005／7／14

クエリ変形のモデル化および対話的クエリ精密化手法の提案

山野邊 大嗣† 遠山 元道‡

†慶應義塾大学大学院 理工学研究科 開放環境科学専攻
‡慶應義塾大学 理工学部 情報工学科
E-mail: † daishi@db.ics.keio.ac.jp, ‡ toyama@ics.keio.ac.jp

近年, データベースを中核とするウェブサイトが増加している. こういったウェブサイトの情報検索において, データ分布に関する知識や明確な検索目標をもたない検索者が, 嗜好錯誤を強いられるケースが多くある. これに対し, システム側でクエリの自動変形を行う様々な手法が提案されているが, 離散値もしくは連続値に特化していたり, 扱うクエリの形式がユーザの意図を厳密に表すのに適していないなどといった欠点がある. 本論文では, 多様な特徴をもつ e コマースサイトにおいて, 検索者が直感的な検索を通じて有益なデータを得られるような, 検索モデルおよびクエリ変形手法の枠組みを提案する. 具体的には, まず始めに, 条件指定インターフェイスを介して生成されるクエリを, ユーザからのフィードバックに基づいて選言形式に精密化する. そして, この選言クエリを構成する各タームの代表的なキーとなる特殊属性の緩和を行う. 最後に, 精密化されたクエリ範囲内のデータ分布および緩和した特殊属性のデータ分布の相対関係を基に特殊属性以外の条件を確定し, 最終的な結果をユーザに提示する.

キーワード :クエリ緩和, クエリ精密化, 対話的, 離散値

Query Relaxation Based On User Feedback and Data Distribution

Daishi YAMANOBE † Motomichi TOYAMA ‡

†School of Science for OPEN and Environmental Systems,
Faculty of Science and Technology, Keio University.
‡Department of Information and Computer Science, Faculty of Science and Technology,
Keio University.
E-mail : †daishi@db.ics.keio.ac.jp ‡toyama@ics.keio.ac.jp

Many web applications have been developed database-centric recently, especially in the e-commerce domain. Users of such applications often meet difficulties with expressing their intensions and resort to exploratory "trial-and-error" queries. Though the query composed is rigid enough, it may fail to return satisfactory number of results. In such cases, it may be better for the applications to produce approximate answers rather than enforcing users to relax or tighten the query constraints manually. This paper presents an approach to automatically reformulate queries based on user's feedback, in order to extract results that are likely to fulfill user's desires. First, initial query is reformulated into disjunctive form. The system then relaxes query condition on the key attirbute which represent each term in that disjunctive query. Finally, constraints on the other attributes are determined according to data distribution, to produce the final results.

keyword :Query Reformulation, Interactive, Discrete Attribute

1 はじめに

近年, データベースを中核とするウェブアプリケーションが増加している. 特に e コマースの領域において, 膨大な商品情報を扱うショッピングサイトや, 不動産の物件情報を扱う賃貸検索サイトなどが代表的な例としてあげられる. そういったウェブサイトでは, データベースやクエリ記述に関する知識を持たないユーザを想定し, 情報の簡易検索を実現する条件指定インターフェイスが提供されている. 一方で検索システムの内部では, 検索者が指定した条件を基に従来の SQL を生成して問合せを行う方式を採用するサイトが多く, 検索を行う上で重要となるデータ分布等の提示や, 一連の検索プロセスを効率化するような支援は提供されないのが一般的である. 従って, 検索者が意図に合致する結果を得るまでに, 結果を吟味した上で条件指定を繰り返すといった試行錯誤を強いられるケースが多くある. こういった「trial-and-error クエリ」の原因として, 以下の二つが考えられる.

1. 結果件数が多すぎる, もしくは少なすぎる
2. 検索の目標が不明瞭である事などが原因で, 意図をうまく表す事が出来ない

1 の問題に関しては, データ分布などの情報を基にクエリ条件を自動調節する事によって, 結果件数を最適化する手法 [4, 3] や, 抽象階層木を用いて離散値属性に対する条件を緩和する手法 [6] が提案されている. これらの手法の欠点は, 連続値属性もしくは離散値属性のいずれかに特化している点や, 検索者の意図を厳密に反映できない連言型のクエリを対象としている点であり, 本論文でとりあげる e コマースサイトのように多様な特徴を持つ対象には不向きであるといえる. 又, 検索者の要求とは無関係な指針を用いてクエリ緩和を行う為, 問合せの結果に, 意図に反するデータが多く含まれてしまう可能性がある. 更に 2 の問題への取り組みとしては, フィードバックを基に検索者の隠れた意図を汲み取り, それをクエリ変形の指針として利用するといった方法が考えられる. Ishikawa[7] らは, ユーザにより与えられる複数の例示を基に, 検索式を自動生成する手法を提案している. この手法はユークリッドなどの距離関数がベースとなっている為, 順序付けされていない離散値属性等は対象外である.

それらを踏まえた上で本論文では, 多様な特徴をもつ e コマースサイトにおける情報検索にて, 検索者がデータ分布を意識したり複雑な条件指定に試行錯誤する事なく, 有益なデータを得られるような, 検索モデルおよびクエリ変形手法を提案する. 具体的には, まず始めに, 条件指定インターフェイスを介して生成されるクエリを, ユーザからのフィードバックに基づいて選言形式に精密化する. そして, この選言クエリを構成する各タームの代表的なキーとなる特殊属性の緩和を行う. 特殊属性とは, 検索者が最終的な結果を選定する上でキーとなる属性を指す. 最後に, データ分布の相対関係を基に特殊属性以外の属性に対する制約を確定し, 新たな条件をクエリに追加する.

以下, 2 章にてクエリ精密化手法について述べ, 3 章で離散値の内挿および外挿によるクエリ緩和手法について解説を行う. 最後に, 4 章および 5 章にて, 今後の課題とまとめについてそれぞれ述べる.

2 クエリ精密化手法

条件指定インターフェイスを介して生成されるクエリは, 連言形式で表されるのが一般的であり, 検索者の複雑な要求を表現するのは難しい. 意図を網羅するクエリ記述は可能ではあるが, 検索範囲が広くなる分, 意図に反するデータを多く抽出してしまう事は避けられない. 簡単な例を以下に示す.

$$Q_D : (\text{駅} = \text{日吉} \wedge \text{賃料} \leq 7\text{万}) \vee (\text{駅} = \text{自由が丘} \wedge \text{賃料} \leq 15\text{万})$$

$$Q_C : (\text{駅} = \text{日吉} \vee \text{駅} = \text{自由が丘}) \wedge \text{賃料} \leq 15\text{万}$$

検索者が「日吉だったら 7 万円以下が良いが, 自由が丘だったら 15 万円まで良い」といった意図を持っているとする. Q_D はこの意図を精密に表現した選言型クエリである. 厳密には, 一つのドメイン制約を一タームとした際の, ターム間の選言で表される. 一方でインターフェイスを介して生成されるクエリは, Q_C のようなドメイン制約間の連言により表現されるのが一般的である. Q_C の解集合には, 7 万円以上の日吉の物件が含まれてしまうため, 検

索者は結果選定に余分な時間と労力を費やすこととなってしまう.
クエリ緩和にも同様のリスクが伴う. 例えば, 「駅=田園調布」という述語の追加に伴い賃料を 30 万まで緩和させた場合, 日吉と自由が丘に対応する賃料まで緩和されてしまい, 意図に反するデータを多く含んでしまう結果となる. 1 章で前述した先行研究 [4, 3, 5, 6] の手法は, このような問題が致命的な欠点となっている. つまり, 検索者の要求が複雑になればなるほど, クエリを Q_D のような選言形式で扱える事が重要となる. そこでまず始めに, Q_C のような初期クエリを, ユーザのフィードバックおよびデータ分布を基に Q_D のような選言形式に精密化する事を考える.

2.1 初期クエリの発行

対象となるデータベースを構成する属性の集合を $\{X_1, X_2, \cdots, X_n\}$ とし, X_i はそれぞれ連続値, 離散値, 真理値のいずれかをとるものとする. この時, ユーザが指定する条件を基に生成される初期クエリ Q_I を, 以下の形式に限定する.

$$Q_I = C_1 \wedge C_2 \wedge \cdots \wedge C_n \qquad where$$

$$C_i = \begin{cases} cond_1(X_i) \vee \cdots \vee cond_k(X_i) & if X_i discrete, \\ cond(X_i) & otherwise \end{cases}$$

$$\forall i = 1, 2, \cdots, k$$

$cond(X_j)$ は, 属性 X_j に課される条件であり, 演算子として等価演算子, 比較演算子のいずれかをとるものとする. 又, 属性 X_i が離散属性の場合に限り, C_j を複数の等価条件の選言として表現する事が可能である. 又, 初期クエリに対する結果集合を $V_I = \{t_1, t_2, \cdots, t_{M'}\}$ と表す.

2.2 ユーザフィードバック

検索者は, 結果集合 V_I の各タプルに対して, "Good", "Bad" の評価を与える. "Good" と評価されたタプルの集合を $S_G = \{g_1, g_2, \cdots, g_n\}$, "Bad" と評価されたタプルの集合を $S_B = \{b_1, b_2, \cdots, b_m\}$(ただし, $S_G, S_B \subset V_I$) とする. いずれにも当てはまらないと判断された場合は, 評価値は与えられない. 評価値のないタプル集合は $S = V_I - (S_G \cup S_B)$ と表す事が出来る. この様なフィードバックは, 条件指定の反復作業に比べ直観的であり, 検索者への負担を多少は軽減する事が可能であると考えられる.

2.3 ユーザの興味抽出

次に, ユーザの興味抽出を行う. 検索者から与えられるフィードバック, S_G, S_B, S より得られる情報を以下に示す.

- 各属性の重要度
- 属性間の相関関係
- 各タプルの特徴量

これらの情報を基に, n 次元のデータ空間上で拡張対象となる領域, 及び, 排除すべき領域を導き出す事が可能となる.

2.4 クエリ精密化

最後に, 2.3 節で取得した情報を基に, クエリの精密化を行う. 精密化されたクエリ Q は以下の様に表される.

$$Q = C_1 \vee C_2 \vee \cdots \vee C_m$$
$$where \;\; C_i = cond(X_1) \wedge \cdots \wedge cond(X_n)$$
$$\forall i = 1, 2, \cdots, m$$

3 離散値の内挿および外挿

e コマースにおける情報検索には, 最終的な結果を選定する上での代表的なキーとなる属性が一般的に存在する. 本論文では, この属性を特殊属性と定義する. 例えば, 不動産の賃貸検索における「駅」, コンサートのチケット検索における「アーティスト」などがこれにあてはまる. 本章では, それら特殊属性に対して検索者が初期に選定する値の内挿および外挿を求めることによって, その検索者が興味を持ちそうなデータを含むような条件を導き出す手法について述べる.

3.1 特殊属性のマッピング手法

前述した例からもわかる通り, 特殊属性は離散値属性であるケースが多い. 離散値属性は, 数値的な特徴や順序といった概念を含まないため, オントロジーなどの抽象階層関係を用いて緩和を行う方法が考えられる [6]. しかし, 値を選ぶ基準はユーザに

より異なる為, 個々の意図に特化した緩和が行われないケースがほとんである. そこで, まず始めに離散値に数値的な特徴を持たせる事によって順序付けを行い, ユーザから得られる値の内挿・外挿を新しく提示する値の候補として抽出する. ただしその際に, 特殊属性を特徴付ける, 付随的な数種類のデータを用いる事を前提とする. 又, これらのデータは数値であるか, もしくは順序付けされている必要がある. 以下に不動産データベースの「駅」関係のスキーマ例を示す.

駅 (都心からの距離, 利用者数, コンビニ件数)

駅を特徴づけする情報として, 関係で定義された情報以外に, 物件データの集約を利用する事も考えられる. 例えば,

```
SELECT avg(賃料)
FROM 物件
GROUP BY 駅
HAVING 駅='日吉';
```

といった集約演算により, 駅ごとの賃料平均や標準偏差などを扱う事が可能である.

ここで, 特殊属性値の集合を $V_s = \{v_1, v_2, \cdots, v_n\}$, その付随属性の集合を $A = \{A_1, A_2, \cdots, A_k\}$ とする. そこでまず, V_s 内の各離散値 v_i を A の属性軸 A_j に射影, もしくは集約値の軸に写像する. これを $f_j : S = X$(ただし, $j = 1, \cdots, k$) と表す (図 1 参照).

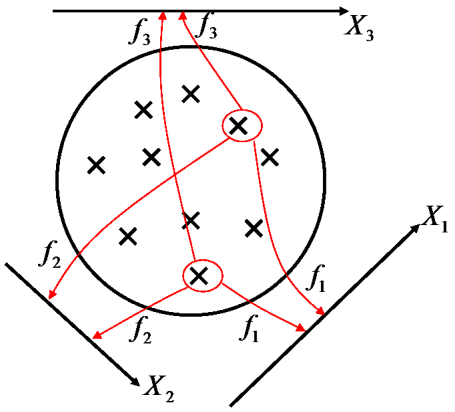

図 1: 特殊属性値の写像

こうする事で離散値を連続値として扱う事が出来, その内挿および外挿を導き出すことが容易となる.

3.2 内挿および外挿の候補値

次に, ユーザが初期クエリの条件として特殊属性に対して選定した値を基に, その内挿および外挿を抽出する手法について述べる. ここで重要となるのは, 検索者がどういう基準でそれらの値を選んでいるかといった, 検索者の隠れた意図を発見する事である. つまり, 検索者が持つ個々の軸に対する重要度を算出する事である.

ここで, ユーザが選定した値の集合を $U = \{u_1, u_2, \cdots, u_k\}$ とする. 更に, A_j に写像された離散値の集合を $V'_j = \{v'_{j1}, v'_{j2}, \cdots, v'_{jn}\}$, その区間を $[min(V'), max(V')]$ とすると, ユーザが注視する属性を V' の標準偏差に対する U の標準偏差によって特定する事が可能となる. つまり, その区間において選定値が一箇所に集中するような場合は, その属性の重要度が高い, と仮定できる. 重要度 w_j は以下のようにもとめる.

$$w_j = \frac{D'_j}{D_j}$$

$$D = \frac{1}{\bar{v'}} \sqrt{\frac{1}{n} \sum_{i=1}^{n} (v'_i - \bar{v'})}$$

$$D' = \frac{1}{\bar{u}} \sqrt{\frac{1}{k} \sum_{i=1}^{l} (u_i - \bar{u})}$$

各付随属性に対する重要度 w_j が確定したら, 次に各特殊属性値のスコアを求める. スコアは, 以下のように定義する.

$$RP_i = \sum_{j=1}^{k} w_j^{-1} \frac{v'_{ij} - \bar{u'}}{SD}$$

$\bar{u'}$ は, ユーザが選定した値の写像を平均したものである. SD は, 軸 A_i に写像されたデータの標準偏差であり, 異なる軸間での単位のばらつきを除去するための正規化を目的としたものである. ユーザが着目している属性に対して, ある離散値の写像の偏差が大きい場合, スコアは大きくなるべきである. 逆に, 写像の偏差が小さい場合は, スコアが小さくなる. これを踏まえると, 重要度 w_j の逆数を採用するのが最適である.

3.3 相対比較に基づくクエリ条件の確定

新たに確定した特殊属性値に対して, その他の属性に課すべき条件を確定する方法について述べる. 3.1 で精密化されたクエリの条件が以下の Q_D であるとする.

$$Q_D : (\text{駅} = \text{日吉} \wedge \text{賃料} \leq 7\,\text{万}) \vee (\text{駅} = \text{自由が丘} \wedge \text{賃料} \leq 15\,\text{万})$$

これに対し, 新たに確定した「駅」の値が「田園調布」である場合, 賃料の値は, 日吉や自由が丘の全データ件数に対して, 条件に合致する件数の比率を適用する事によって確定する. 例えば, 日吉の物件が 100 件あるのに対し, 賃料が 7 万円以下の物件が 50 である場合, その比率は $\frac{1}{2}$ となる. そして田園調布の物件件数が 50 である場合, その比率分のデータ件数, つまり, 25 件の物件を含むような条件を導きだす.

4 今後の課題

まず第一に, クエリ精密化アルゴリズムの詳細についての考察が必要である. 検索者からのフィードバックを基に, 興味の対象になりそうな領域とそうでない領域を分類する方法として, 決定木など学習機能を用いる事が考えられる. 詳細については今後の課題として検討する. 更に, 特殊属性値の内挿および外挿を抽出する際に用いる, 重要度計算式の妥当性について検証を行う必要がある. 多次元尺度構成法や主成分分析などといった分析法を用いる事も考慮にいれ, 研究を進めていく. 最後に, 今回提案するクエリ緩和手法の有用性を証明すべく, 実装および評価を行う. その際に妥当な評価方法について検討をする必要がある.

5 まとめ

データベースを中核とするウェブサイトの情報検索において, ユーザの意図およびデータ分布に基づいた自動クエリ緩和手法のフレームワークを提案した. 不動産情報や商品情報などといった複雑な性質を持つデータベースを対象とし, 離散値や連続値のいずれかに特化しない手法である事を示した. 又, クエリ緩和を行った際に, 意図に合わないデータを多く拾ってしまう事を避ける為, クエリを選言形式に精密化しながら緩和を行う手法について解説した.

参考文献

[1] Shinwa-J. *http://www.shinwa-j.co.jp/*

[2] Yahoo Auction!. *http://auction.yahoo.co.jp/*

[3] Abhijit Kadlag, Amol V.Wanjari, Juliana Freire, and Jayant R.Haritsa. Supporting Exploratory Queries in Databases. In *Database Systems for Advanced Applications: 9th International Conference*, pages 594-605, 2004.

[4] Nicolas Bruno, Surajit Chaudhuri, Luis Gravanjo. Top-k Selection Queries over Relational Databases: Mapping Strategies and Performance Evaluation. In *ACM Transactions on Database Systems*, pages 153-187, 2002.

[5] Ion Muslea. Machine Learning for Online Query Relaxation. In *Proceedings of the 2004 ACM SIGKDD international conference*, pages 246-255, 2004.

[6] S.-Y. Huh, K.-H. Moon, H.Lee. A Data Abstraction Approach for Query Relaxation. In *Infromation and Software Technology 42*, pages 407-418, 2000.

[7] Yoshiharu Ishikawa, Ravishankar Subramanya, Christos Faloutsos. Mindreader: Querying databases through multiple examples. In *Proceedings of the 24rd International Conference on Very Large Data Bases*, pages 218-227, 1998.

[8] 中島伸介, 田中克己. 相対的マッピング処理に基づく相対的情報検索手法. In *Transaction on Databases*, pages 63-75, 2004.