1G1-4

Web 上の表情報の例示検索における機械学習手法の検討

Machine Learning on Query by Example Searching for Web Information in Tabular Formulation

前島一弥[*1] 横川智浩[*1] 吉田稔[*2] 山田剛一[*1] 絹川博之[*1] 中川裕志[*2]
Kazuya Maejima Tomohiro Yokokawa Minoru Yoshida Koichi Yamada Hiroshi Kinukawa Hiroshi Nakagawa

[*1] 東京電機大学大学院 Graduate School of Engineering, Tokyo Denki University
[*2] 東京大学情報基盤センター Information Technology Center, University of Tokyo

Abstract: In this paper, we have discussed the method of searching web information in tabular formulations by displaying the information in tabular forms. We have used machine learning method to improve the ranking of search results, and we have evaluated our search method. In this paper we have evaluated a new feature which we added into our recent work which manages to see through whether a tabular in the search result is a tabular in real form or a layout in tabular form to improve the ranking of the search results.

1. はじめに

1.1 目的

Web 上の表情報は構造化されているため良質な情報であることが多く，情報が構造化されていることを利用することで，従来の単語群を検索質問とする全文検索よりも精度の高い検索を行うことができると考えられる．そこで，検索対象を Web の表情報とし，ユーザの検索意図である情報内容を表形式で例示し検索する，表情報の例示検索方式を検討している．[1]

本論文では，Web 上の表情報の例示検索における機械学習のフィーチャー（特徴）の最適化を行い，対象分野の偏りのないデータを用いてシステムを評価した．

1.2 関連研究

関連研究として，まず，野口らによる「Web 上の表検索」が挙げられる．野口らは検索対象を Web 上の表情報に定め，特化させることで，より粒度の細かい情報を検索することができる細粒検索システムについて考察している．[2]

この手法では表情報が構造化されていることを利用していないため，ユーザが入力した検索単語から，ユーザが所望する表情報を正確に捉えづらいという問題点がある．また，表情報を順序付けにおいてユーザの情報要求に関する情報が少ないため，Web 上の表情報のパラメータに依存することになる．

野口らの研究では検索手法の評価が行われていないため，本研究との比較は行っていない．

次に，Web 上の表情報の抽出手法として，林らによる「機械学習を用いた WWW からの製品性能表の分類と抽出」が挙げられる．嶋田らは，Web 上にある製品のスペックを記述した表から，製品の情報を抽出する手法について研究しており，製品選択支援システムの構築に向けて，システム入力部にあたる HTML 文書からの製品性能表の情報抽出処理について述べている．[3]

この手法では，情報を抽出する対象を Web ページ上の製品の性能表に限定しており，汎用性が低いという問題点がある．本研究は，製品性能表も含めて複数の分野を対象としており，この問題点を解決している．

連絡先：〒101-8457 東京都千代田区神田錦町 2-2
東京電機大学大学院 工学研究科
情報メディア学専攻 計算言語学研究室
電話：03-5280-3332 内線：2810

2. 例示検索方式

例示検索方式とは，検索質問を表形式で例示する検索方式である．これは，RDB の Query By Example の考え方を取り入れたものである．Query By Example の問い合わせ対象がデータベースの情報であることに対して，本研究は Web 上の表情報に対して問い合わせを行う．上記の Query By Example とは，データベースへの問い合わせの際，ユーザの情報要求と結果の例を指定する方式である．機能的なレベルは SQL とほぼ同じであるが，視覚的な設計がされている点が SQL と異なる．[4]

2.1 例示表の入力方式

ユーザは，あらかじめ用意した表形式の検索インタフェース（図 1）に，検索を所望する表の例を入力することで，検索条件を与える．検索インタフェースは，表を属性と値のペアの集合として捉え，表題を入力するセル 1 つ，属性を入力するセル 2 つ，値を入力するセル 2 つで構成している．属性のセルと値のセルは上下でペアとする．属性のセルには表頭や表側に出現すべき単語を入力し，値のセルには属性に対する値が入力されることを想定している．

表題		
属性		
値		

図 1 表形式の検索インタフェース

ユーザから与えられた単語と，その単語が入力された位置情報を条件として，Web 上から表情報を検索する．このとき，ユーザが入力した表を例示表と呼称するものとする．

2.2 検索の実行

検索には GoogleAPI を使用する．GoogleAPI に検索条件を与え，検索された Web ページの URL を取得する．[5]

GoogleAPI に例示表をそのまま渡すことはできないため，検索インタフェースに入力された例示表を検索条件式に変換し，検索する．以下に，例示表の検索条件式への変換手順を示す．

(1) 左から順に，属性→値の順で検索単語を AND でつなぐ．

(2) 表題と(1)で生成した文字列を AND でつなぐ.

図 2 のような例示表の場合,「航空会社」,「JAL」,「料金」の順で, 検索単語を AND でつなぐ. 次に表題の「ツアー」と, (1)で生成した文字列をつなぐ. つまり, 図 2 の例示表は, 検索条件式「ツアー AND 航空会社 AND JAL AND 料金」に変換される.

表題	ツアー	
属性	航空会社	料金
値	JAL	

図 2 検索インタフェースにおける入力例

2.3 検索結果からの表情報の抽出

GoogleAPI により検索した Web ページから表を抽出する.

Web 上にはレイアウトを目的として使われている表タグが数多く存在するが, これらは検索の対象とすべきでない. 具体例としては, 表を包含しているもの, 箇条書きを表で実現しているもの, などがあげられる.

このようなレイアウトを目的とした表を除去するために, 入れ子になっている一番内側の表のみを検索の対象とする. さらに, 箇条書きのレイアウトを実現するために使われている表を除去するために, 1 行(あるいは 1 列)の表を検索の対象から外す.

2.4 表情報の順序付け

検索された表情報を, よりユーザの要求を満たすものが上位になるよう順序付ける. この順序付けには, SVM(Support Vector Machine)の機械学習によって生成された分類モデルを使用する. SVM として, TinySVM を使用している. [6]

SVM の機械学習には, 予備実験により有効と判断された 80 のフィーチャー[2]に, 新たに行った予備実験にて利用できると判断された境界フィーチャーを追加し, 81 のフィーチャーを使用する(表 1). 境界フィーチャーについては次節にて説明する.

81 個のフィーチャーを使用し機械学習を行う. 機械学習によって生成された分類モデルが与える『2 クラス間の境界面』からの距離を用いて, 表が検索意図に合致している度合いに沿う順序付けを行う. つまり, 分類モデルが定義する境界面付近より, 正解方向に離れれば上位へ, 不正解方向に離れれば下位へ順序付けを行う(図 3).

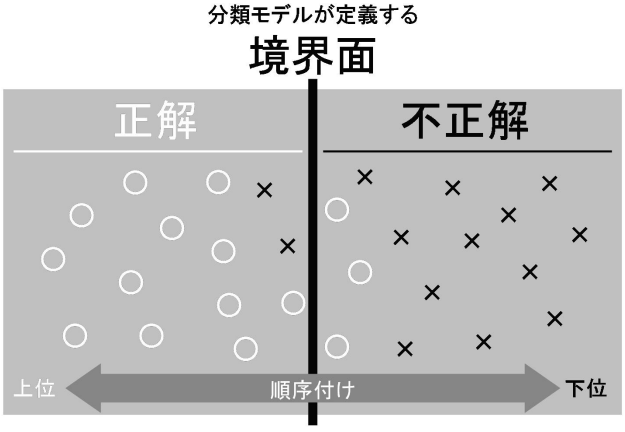

図 3 順序付けの仕方

表 1 フィーチャーの種類および数

カテゴリ	予備実験によって選別されたフィーチャー数
検索単語	10
強調・意味付け	11
画像	3
形状	3
セルの結合	3
文字数	12
改行	13
段落	4
文字修飾	9
フォーム	4
背景色	3
文字フォント	5
境界フィーチャー	1
計	81

2.5 境界フィーチャー

(1) 役割

境界とは, 属性と値の堺のことで, 図 4 の二重線の部分をいう. 図 4 において境界より上の行は属性を, 下の行は値を示す.

境界があると判断できれば, その表は本質的な表といえ, 逆に境界がないと判断できれば, その表はレイアウト目的の表といえる.

このフィーチャーを利用することで, 本質的な表を上位へ順序付ける.

商品番号	商品名	価格
1	りんご	100 円
2	バナナ	80 円
3	みかん	50 円

図 4 境界の例

(2) アルゴリズム

(a) 境界探索の手順

以下に境界フィーチャーにおける境界探索手順を示す.

1. 表の正規化を行う
2. 行に対しての境界を探す
   2-2. 列単位に境界を探す
   2-3. もっとも多かった位置を行の境界とする
3. 列に対しての境界を探す
   3-1. 行単位に境界を探す
   3-2. もっとも多かった位置を列の境界とする
4. 行・列のどちらかに境界があれば本質的な表とする

(b) 表の正規化

(a)の 1. に示した表の正規化について説明する. 表の正規化とは図 5 のようにセルが結合された(rowspan, colspan 属性を使用した)表に対して, セルを分割し, 図 6 のように 2 つ以上のセルにすることである. その際, 分割後新たにできたセルには分割前の内容をコピーする.

商品名	価格	
	会員	非会員
りんご	100 円	120 円
バナナ	80 円	100 円

図 5 結合表

商品名	価格	価格
商品名	会員	非会員
りんご	100 円	120 円
バナナ	80 円	100 円

図 6 正規化した表

(c) 境界探索

(a)の 2. ～4. の境界探索について説明する. まず, 行に対しての境界を探すため, 各列において行の境界を探し, もっとも境界と判定されることの多かった位置を算出する. 1 列目を取り出し, 1 行目と 2 行目, 2 行目と 3 行目, 3 行目と 4 行目・・・の各組み合わせの類似度を算出する. 類似度がもっとも小さかった位置を 1 列目の境界とする. これを各列行い, もっとも多かった位置を行の境界とする. もし, 同数の場合は, 値の小さい位置を境界とする.

次に, 列に対しての境界を探すため, 各行において列の境界を探し, もっとも境界と判定されることの多かった位置を算出する.

行・列共に境界を算出したら, 機械学習への学習データとして利用できるように, 境界があったら「1」に, 境界がなかったら「0」に変換する.

本アルゴリズムは境界探索に多数決を採用している. 多数決ではなく平均を利用すると, 他とは極端に異なる行(列)単位における境界位置があった場合, 平均化されるため境界の位置がずれてしまう. その点, 多数決ではカバーできる.

(d) 類似度計算

(c)で使用した類似度の算出方法について説明する. 類似度には以下の 7 つを足し, 算出する.

- 文字種の編集距離. 文字列を文字種に変換し, 比較する. 変換する際, 同じ文字種が続いた場合は, 集約する. 例えば「100 円」の場合, 「(数字)(漢字)」と変換する.
- 品詞の編集距離. 文字列を形態素解析「Sen」[7]を利用して, 形態素に分け, 各形態素の品詞へと変換し, 比較する. 文字種同様変換する際は, 同じ品詞が続いた場合は, 集約する. 例えば, 「100 円」の場合, 「名詞」と変換する.
- セルの背景. セルの背景色を比較する. 属性にあたるセルは高確率で, 値にあたるセルとは別の背景色となるという特性を利用して, 背景色が異なれば, 類似度が低いとする.
- 文字列の長さ. 文字列の長さを比較する. 属性にくる文字列と値にくる文字列には差が生じる可能性がある. 差が大きければ, 類似度が低いとする.
- 接頭辞. 「第」, 「昭和」, 「平成」, 「㈱」, 「株式会社」が前方一致し, かつ, 対象セルも前方一致した場合, 類似度が高いとする.
- 接尾辞. 「円」, 「ドル」, 「大学」, 「県」, 「㈱」, 「株式会社」, 「学校」など計 66 個が後方一致し, かつ, 対象セルも後方一致した場合, 類似度が高いとする. 自然言語研究会
- 強調タグ. 属性に当たる文字列は高確率で, 強調タグが使用されているという特性を利用して, 強調タグを含んでいたら, 属性の可能性が高いとする.

(3) 精度

予備実験として, 5 つの分野を選び, それらの表に対して, 精度を算出する. 精度は式(1)のように算出する.

$$精度 = \frac{①+②}{③} \quad (1)$$

①は本質的な表に対して境界を引いた数,
②はレイアウト表に対して境界を引かなかった数,
③は取得した表(本質的な表＋レイアウト表)の総数.

精度を表 2 に示す. 表 2 の結果より, 機械学習のフィーチャーとして, 境界フィーチャーを採用した.

表 2 境界フィーチャーの精度

分野	精度
趣味とスポーツ	71%
政治と社会	68%
ビジネスと経済	74%
文化	68%
健康	79%

2.6 結果の提示

順序付けた表情報をユーザに提示する. それぞれの表には付属情報としてタイトルと URL を表示する.

3. 評価実験

3.1 対象分野

分野の偏りをなくすため, 様々な分野を網羅しているポータルサイトのカテゴリに注目し, 4 つのポータルサイト(Yahoo!Japan, infoseek, goo, ライブドア)に共通する 12 のカテゴリを対象分野とする. それら分野に対して検索を行い, 表を収集する. 以下に 12 のカテゴリを示す.

- エンターテイメント
- メディアとニュース
- 趣味とスポーツ
- ビジネスと経済
- 文化
- 暮らし
- コンピュータとインターネット
- 健康
- 教育
- 政治と社会
- 学習
- ショッピング

3.2 表の収集

検索質問の構成を統一するため, 各分野に対して, 3 つの例示表構成パターンを用意し, 表を収集する. 3 つの例示表構成パターンは分野に関連した異なる内容である. 例示表構成パターンについて以下に示す.

- 属性 A と値 A のペア(以下, 例示表構成パターン 1)
- 属性 A と値 A のペアかつ属性 B のみ(以下, 例示表構成パターン 2)

- 属性 A のみ(以下, 例示表構成パターン 3)

12 分野×3 例示表構成パターン×200 = 7,200 表を収集する.

3.3 実験方法

以上収集した Web 上の表を用いて, 機械学習によって生成された分類モデルの正解・不正解の分類精度の検証を行う. 評価の方法として, 総合評価と例示表構成パターン別の評価を行った.

(1) 総合評価

収集した 7,200 個の表に対して 4-fold クロスバリデーションを行う. このとき, クロスバリデーションの 1 群は, 3 分野で構成された 1,800 個の表からなる. 1,800 個の表は 3.2 節の 3 つの例示表構成パターン(600 個)×3 分野で構成されている.

(2) 例示表構成パターン別の評価

収集した 7,200 個の表を例示表構成パターン別に分け, 各例示表構成パターンにつき 4-fold クロスバリデーションを行う. つまり, 2,400 個の表に対して 4-fold クロスバリデーションを行う. クロスバリデーションの1群は, 3 分野で構成された 600 個の表からなる.

3.4 正解・不正解の基準

収集した表は事前に正解と不正解に判別する. 判別基準は, 表構成パターンごとに異なる. 例示表構成パターン 1 の場合は属性 A と値 A のペアを含んだ表を正解とする. 例示表構成パターン 2 の場合は属性 A と値 A のペアを含む表で, かつ属性 B に対して値があった場合正解とする. 例示表構成パターン 3 の場合は属性 A に対して値があった場合正解とする. また, 共通の判別基準として表題がタイトルや前の文章に表れている場合を正解とする. 例えば, 図2の場合は表題「ツアー」がタイトルや前の文章にあり, かつ属性「航空会社」と値「JAL」のペアを含み, かつ属性「料金」に対して何らかの値があれば, 正解となる.

3.5 比較対象

比較対象として Google の検索結果を表単位に変換したものを用いる. これを Google'と呼称する. このとき Google'は, 検索結果の上位のページから表を順番に抽出し, 表に対しての検索単語の出現頻度順にソートしたものとする. 検索精度の検証における対象分野は, 3.1 節で示した分野である.

3.6 実験結果

精度を求めるため, 平均精度を利用する. 平均精度 v の定義を式(2)に示す.

$$v = \frac{1}{\sum_{i=1}^{N} x_i} \sum_{i=1}^{N} \left[\frac{x_i}{i} \left(1 + \sum_{k=1}^{i-1} x_k \right) \right] \qquad (2)$$

ここで, N は表情報の総数, x_i は出力順第 i 位の表情報の正解と不正解の状態を示す変数とする. 正解ならば x_i = 1, 不正解ならば x_i = 0 とおく.

表 3 に, 機械学習によって生成された分類モデルの精度と Google'の精度の比較を示す.

表 3 検索精度の比較

	平均精度		①−②
	①例示検索方式	②Google'	
総合評価	67.06%	60.54%	+6.52
例示表構成パターン 1	72.93%	61.75%	+11.18
例示表構成パターン 2	71.25%	58.83%	+12.42
例示表構成パターン 3	63.94%	61.04%	+2.90

4. 考察

3 の評価実験において, 例示検索方式が総合評価に関しては+6.52%, 例示表構成パターン 1 に関しては+11.18%, 例示表構成パターン 2 に関しては+12.42%, 例示表構成パターン 3 に関しては+2.90%, Google'より精度が良かった. これによって, 例示検索方式が Google'に比べ有効であることがわかる. 一方, 例示表構成パターン 3 について, +2.90%にとどまっているが, このパターンは属性のみからなるので, 学習のためのフィーチャーが少なく, その効果が出難いと考えられる. 今後は, 検索単語に関連しないフィーチャーの最適化を図り, さらに精度を向上させることが必要である.

5. おわりに

入力した例示表の情報から, Web 上の表を検索する方式を検討した. 本論文では, 有効と考えられるフィーチャーの追加を行った.

機械学習によって生成された分類モデルの精度の検証実験を行い, 表情報の例示検索方式において, 例示表の構造特性を利用することの有効性を示すことができた.

今後は, 境界フィーチャーおよび例示検索方式の精度の向上と, 検索インタフェースの使い勝手の向上を目指す.

参考文献

[1] 前島一弥,横川智浩,吉田稔,山田剛一,絹川博之,中川裕志: Web 表情報の例示検索方式とその評価,2008 年電子情報通信学会総合大会論文集 情報・システム第 1 分冊,p.69,2008.

[2] 野口正人,廣川左千男:Web 上の表検索,人工知能学会 1C4-02,2003.

[3] 林晃司,島田和孝,遠藤勉:機械学習を用いた WWW からの製品性能表の分類と抽出,言語処理学会第 10 回年次大会論文集,pp.733-736,2004.

[4] C.J.DATE: An Introduction to Database Systems(Third Edition),ADDISON-WESLEY PUBLISHING COMPANY,1982.

[5] Google: http://www.google.com/intl/ja/

[6] TinySVM: http://chasen.org/~taku/software/TinySVM/

[7] Sen: http://ultimania.org/sen/