HTML テーブル情報の XML による統合

Integration of Multiple HTML Table Information into one XML List

板井 久美* 高須 淳宏** 安達 淳**

Kumi ITAI Atsuhiro TAKASU Jun ADACHI

本稿では，複数の HTML ページに存在し，内容的には類似していても，構造の全くバラバラな表情報を，全て一つの共通の XML の表構造に変換し，統合する手法について提案する．これにより，独立して存在する表の情報を一覧表とし，一見して，各々の特徴を比較することができる．
今回は，その中でも，各表の内容を解析することにより，「表構造から情報を抽出」し，「それらを意味によって分類」するというタスクに焦点を当てている．情報の意味による分類方法として，本稿では，(I)Support Vector Machine による分類，(II)隠れマルコフモデルを用いた表構造推定による分類，という 2 通りの手法を試み，分類の精度の比較を行った．

In this paper, We propose a method of transformation of HTML tables, which have various kinds of structure into a common XML list structure. This integration enables us to browse and compare all information that is in separate HTML pages.
This paper focuses on the tasks of information extraction from tables and data categorization. For this purpose, we applied two algorithms, (I) data classification by Support Vector Machine and (II) a table structure estimation and data categorization by Hidden Markov Model, and report the experimental results.

1. はじめに

本稿では，WWW ページ上に別々に存在する，複数の「内容の類似した表」を，その中のデータの意味を解析し，分類することにより，それらを 1 つの共通の表構造に変換し，統合するための手法を提案する．

現在，WWW の普及により，世界中に分散された多様な文書情報に簡単にアクセスすることが可能となったが，その多様さゆえにユーザは何らかの形で，これらの情報を整理する必要がある．情報をどのように整理するし，統合するかについては様々な方針が考えられるが，本研究では，まず第一段階の試みとして，表，すなわち HTML の Table タグ(<TABLE></TABLE>)で囲まれた部分に着目し，表の統合を行うことを目指した．

* 非会員 東京大学大学院情報理工学系研究科
kumi@nii.ac.jp
** 正会員 国立情報学研究所
takasu@nii.ac.jp, adachi@nii.ac.jp

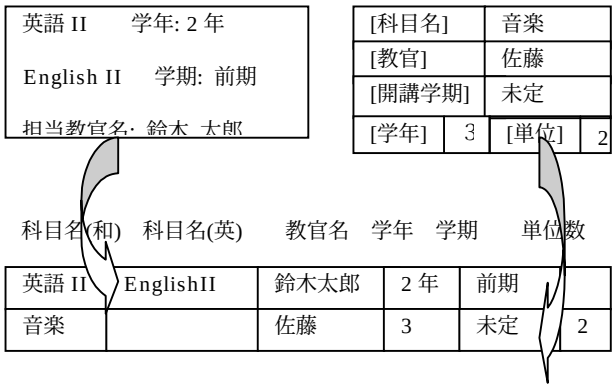


図 1 表の統合の動作例

Fig.1 An example of integration of tables

2. 表の統合

2.1 問題設定

WWW 上に点在する，内容の類似した表を，XML の形で 1 つに統合することができれば，それは各ページの表情報の一覧となり，それらを一目で比較することが出来るようになる．XML では HTML と異なり，目的に応じて自由にタグを定義することが可能である．そのため，各データに対し，意味によるタグ付けをすることができ，表の統合を実現する手段として最適であると考えられる．図 1 に本手法が目標とする動作例を示す．この例では，各大学の授業科目情報についての表(HTML)を，1 つの表(XML)にまとめている．Web 上の表形式のデータは多く，授業科目情報だけでなく，例えば，各種の製品情報の一覧(カタログ)，あるいは各航空会社のフライト情報の一覧といったものへの適用も考えられるため，情報統合の最も有効な例として表を取り上げた．

本研究が目指す，HTML テーブルのXMLによる統合を行うためには，まず各ページの表が，「どのような構造で，どのような情報を表現しているか」の解析が必要となり，さらに解析された情報から，意味的に類似したデータ同士を分類する必要がある．例えば，「住所」と「住居」，「メールアドレス」と「E-mail」と「eメール」，「誕生日」と「生年月日」，等は同一の内容を表しているはずであり，このように，表現が異なっていても，内容的には同じであるというように自動分類されるような方法の提案を試みる．

2.2 用語の定義

表は，一つまたは複数の実世界に存在するオブジェクトを属性により表現したものである．図 2 に授業科目の表の例を載せる．この表では，科目オブジェクトを，「科目名」「英文科目名」「教官名」「学期」「単位数」という 6 種類の属性と，その値を用いて記述している．つまり，スキーマとはある表に表記された属性の集合である．また，セル内に表記されている細かい単語を，ここでは要素語と呼ぶことにする．

同じ種類のオブジェクト(例えば科目情報)を持つ，複数の表を統合するためには，まず各表の中で，「属性を記述している部分」と「値を記述している部分」を判別する必要がある．表中の各セルの要素語が，属性であるか値であるかに関する仮説を，表の論理構造と呼ぶ．図 2 に表の論理的構造を示す．

[科目名]	音楽
[教官]	佐藤
[学期]	未定
[学年] 3	[単位] 2

属性	値
属性	値
属性	値
属性 催	属性 催

図 2　授業科目の表の例 と その論理構造
Fig.2　An example of class information table and its logical structure

3. 実験概要

本章では，各表から要素語を抽出し，その各々に記述されている内容を自動判別し，分類するための提案手法について述べる．またそれらを 1 つの XML の表にまとめる手法についても言及する．

3.1 対象データ

今回は, 日本の 20 大学における授業科目情報の HTML ページ，約 10,100 ページを対象に実験を行った．図 3 に本手法の流れを示している．

各 HTML ページの表から要素語を抽出するため，まず <TD><TH>タグを区切りにデータを切り出す．各要素語に分けるため，「：」「；」「・」などのデリミタをセパレータとして，データを切り出す．

次に，上で取り出された各要素語を，その意味により幾つかのグループに分類することになるが，詳細については，次節以降で述べる．ただし，この 2 つのどちらの手法を実行するにあたっても，前もって全ての要素語に対し，形態素解析を施しておく必要がある．

各要素語を，その指し示す意味によって幾つかのグループに分類した後，その各グループに対し，XML タグを付加する. XML では, タグを自由に定義することができるため,「データの意味によるタグ付け」が可能となる．その結果，各 HTML ページの全く構造の異なる表は，規定された共通の表に変換されることとなり，全てを 1 つにつなげることが可能となる．今回，この XML ページを表示するにあたって，XSLT スタイルシートを用いた．

また，今回の実験で使用した，様々な大学の授業科目情報の表の構造は全くバラバラであり，その例を図 4 に示す．

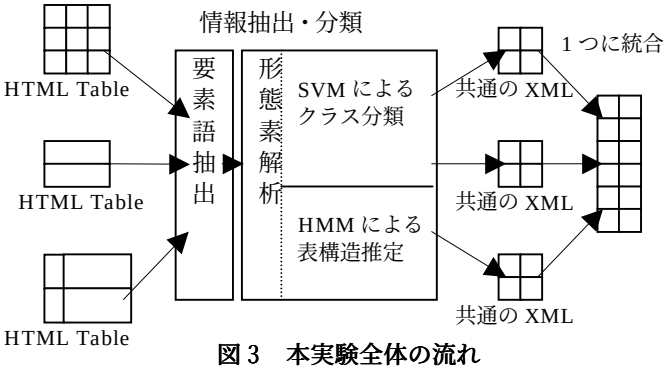
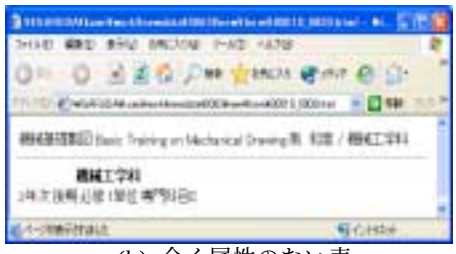


図 3　本実験全体の流れ
Fig.3　The overview of the experiment


(a) 全て属性がついている表


(b) 全く属性のない表


(c) 部分的に属性がついている表

図 4　実験で使用した表の例
Fig.4　The examples of HTMLtables

3.2 SVM による要素語分類

Support Vector Machine(SVM)は，高性能なクラス分類器として注目を集めている．現在，様々な AI タスクに用いられており，例えば手書き数字の認識，顔画像の検出，話者特定など，その範囲は幅広い．SVM をテキスト分類に応用した例も数多くあるが[3], 本研究ではテキストのような長い文章ではなく，要素語という短いフレーズに対して，SVM を適用し，その内容によって分類することを試した．

今回の実験では，属性に対しては<属性>クラスを，そして値に対しては，<科目名(和)>，<科目名(英)>，<教官名>，<学年>，<学期>，<単位数>の 6 つのクラスを，つまり各要素語が 7 つのクラスのいずれかに分類されるようにした．

まず，ある 1 大学の全ての授業科目の HTML ページ(約 2500 ページ)をトレーニングセットとして SVM に学習させる. その際, 学習データから抽出された各要素語に対しては, 人手によって，正解クラスを与えた．そして，残り全ての大学の約 7600 ページの授業科目情報のページをテストセットとし，それらの各要素語を上に挙げた 7 つのクラスに分類した．

本来 SVM は 2 値分類のためのクラス分類器である．しかし今回の実験の場合, 各要素語を 7 つのクラスに分類しなければならない．そこで 7 つのクラスそれぞれについて SVM 学習モデルを作成する．そして，各要素語について作成した 7 つの学習モデルからの数値を算出し,「各クラスの数値の中で最も大きな値をつけた要素語をそのクラスに分類する」という手法をとった. 具体的には図 5 に図示するような値を各列で選択する．この結果については，第 4 章で述べる．

	科目名(和)	科目名(英)	教官名	・・・	単位数
英語	5.2324	3.2870	0.23		0.1256
English	0.9845	6.5324	−1.24		−1.5672
佐藤	1.1231	−1.4527	3.67		−2.4578
2 単位	−1.2378	−0.9523	−2.57		0.9987

図 5 SVM による 多クラス分類

Fig.5 Multi-Classification using SVMs

3.3 HMM による表構造推定

前節と同様に，表中の各要素語を意味によって幾つかのグループに分類する方法として，隠れマルコフモデル(Hidden Markov Model, HMM)の状態遷移を適用することを考えた．

HMM は確率的な状態遷移と確率的な記号出力を備えた有限状態オートマトンである．日本語の形態素解析において，観測可能な言語データから言語現象の背後にある隠れた構造を推定する場合に有効なモデルである．そのため，単語分割モデルや音声認識のための音響モデル，あるいは英語の品詞タグ付けに使われる統計的言語モデルによく用られる．本研究では，表中の各要素語に対し，この HMM を適用することを考える．

本実験では，「状態」を「属性あるいは値の種類」，そして「出力記号」を「要素語を形態素解析したもの」とする．そのモデルの様子を図 6 に示す．矢印は状態遷移を表現しており，全部で 13 の状態を用意した．

ここでは，20 大学それぞれから全体の約 20%，約 2000 の授業科目情報の HTML ページをランダムに選びトレーニングセットとした．トレーニングセットから要素語を抽出し形態素解析したものを出力記号とした．その結果，状態数 13，出力記号数 約 10500 語の学習モデルが作成された．この学習モデルを基に，Viterbi アルゴリズムを用いて残りの科目情報の表の状態遷移系列を復元することを試みた．トレーニングセットに用いなかった約 8100 の表について，各表における要素語のみの連続列を入力とし，そこから推定される状態遷移系列が出力となる．つまり各要素語が 13 の状態のうちのどの状態からの出力であるかを推定するのである．その結果については，第 4 章で述べる．

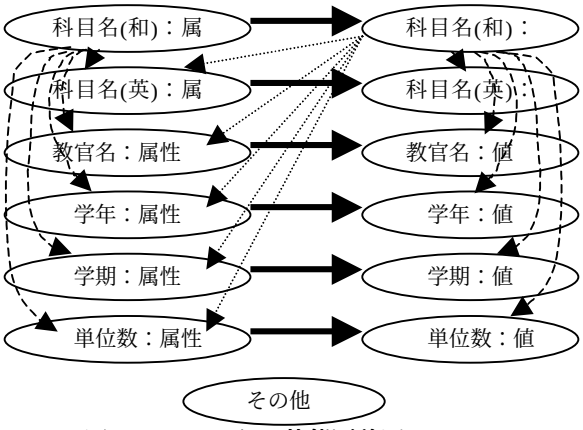


図 6 HMM モデル 状態遷移図

Fig.6 The state transitions of HMM

```
<VSUBJ> 科目名(和) </VSUBJ>
<VSUBE> 科目名(英) </VSUBE>
<VTEAJ> 教官名 </VTEAJ>
<VGRADE> 学年 </VGRADE>
<VSEM> 学期 </VSEM>
<VUNIT> 単位数 </VUNIT>
```

図 7 付与した XML タグ

Fig.7 The added XML tags

3.4 XML タグの付与

表から抽出された各要素語の値は，<科目名>，<英文科目名>，<教官名>，<学年>，<学期>，<単位数>の 6 つのグループに分類される．

XML の大きな特徴は，「タグを自由に設定し，そのタグに意味情報を与えることが出来る」という点にある．例えば，<血液型>A 型</血液型>というタグを使うことで，A 型という文字列に「血液型」という意味を与えているのである．

このような XML タグのメタデータ的な性質を，本研究では分類された要素語の意味付けとして用いる．具体的には，3.2 節 3.3 節でグループ分類された各要素語に対し「科目名の値」として分類されたものについては<VSUBJ></VSUBJ>というタグで挟み，「教官名の値」として分類されたものについては<VTEAJ></VTEAJ>というタグで挟む．ここで，V は Value(値)の意味で，A: Attribute(属性)と区別するために付けている．図 7 に今回の実験で用いた XML タグを示す．このように全ての表から要素語を抽出し，図 7 で定めた 6 つのタグ構造に埋め込むことにより，1 つの共通の XML の表に統合される．

またさらに，一つの表からの情報は，一つの object として<OBJECT></OBJECT>というタグで挟まれる形となる．そして XSLT を用いて Web ページ上に表示させた．

4. 実験結果・考察

第 3 章で述べた手法を，C, C++, Perl を用いて実装し，実験データとして得られた表の集合に適用した．

4.1 分類結果

SVM による要素後分類によって作成された XML の表(図 8)と，HMM による要素語分類によって作成された XML の表(図 9)を，それぞれ各項目について正解判定を行い正解率を計算したところ，表 1 のような比較結果が得られた．(図 8，図 9 に示す例は，共に図 1 の表を統合させたものである)

値の分類	SVM	HMM
<科目名(和)>	0.8013	0.9186
<科目名(英)>	0.9256	0.8490
<教官名>	0.7270	0.4289
<学年>	0.9425	0.7243
<学期)>	0.9556	0.9457
<単位数>	0.9478	0.6610

表 1 分類正解率 (SVM，HMM)

Table 1 The results of classification (SVM，HMM)

SVM の方が，全体的に精度良く分類されていることがわかる．SVM の場合，各値の要素語そのものだけを見て分類していくのに対し，HMM の場合状態遷移系列からその表の構造を推定し要素語を分類していくという，全く異なるアプローチである．表構造全体を推定しなければならない HMM の方が精度が下がるのは当然の結果といえるであろう．

4.2 考察

SVM, HMM のそれぞれの分類結果を詳しく見ていく．まず SVM の場合，科目名(和)，科目名(英)，教官名は比較的良く分類されている．また「1 年次」,「2 単位」あるいは「前期」というような,「年」,「期」,「単位」がついている語は，ほぼ 100%確実に分類できていた．一方，誤って分類された例としては，

- 「2」(単位数)と「001」(講義番号)のような数値のみの要素語が複数存在した場合,「001」の方が単位数として分類されてしまう．
- 「1 年前期」という要素語があった場合，その要素語のまま学期に分類されてしまい，学年には一つも分類されない

という現象が起こってしまった．

一方 HMM では，要素語の前後に記述されている情報を加味して各要素語を分類するように働くと考えられる．その場合，確かに「単位数：2」「講義番号：001」とあった場合，「2」という要素語は単位数の値としてきちんと分類される．しかし，HMM では，トレーニングセットに存在しなかった単語が出現した場合，12 状態以外であると認識されてしまうため，非常に正解率が下がってしまうという難点がある．特に，教官名での下が方は激しい．

以上の考察より，新たに SVM と HMM を組み合わせるような手法が考えられる．具体的には，まず SVM による分類を行う．その際，各要素語に対して 7 つの学習モデルから算出される数値をすべて 1 つのベクトルとする．次にそのベクトルを HMM に適用する．すなわち，HMM の各状態からの出力記号を，SVM で求められたベクトルとする．現在その手法で実験を新たに進めているが，2 つの手法を組み合わせることにより，各手法単独での分類よりも分類の精度が改善されることが期待される．

5. おわりに

本稿では，WWW 上の情報統合の第一段階として，複数の「内容の類似した表」を解析し，それらを XML の 1 つの表に統合する手法について述べた．

今回は，大学の授業科目情報の Web ページの表部分を実験データとして使用したが，その他に，各社の製品情報(カタログ)や，各航空会社のフライト情報の時刻表など，異なるジャンルにおける表の統合についても今後取り組み，本手法の有効性を評価したいと考えている．

さらに，統合の対象として現在は Web ページ上の表構造のみを考えているが，例えば，今回の実験データの授業科目情報のページについて言えば，表構造の部分だけでなく,「授業概要」「シラバス」といった長い文書も含まれる．そのような Web ページ上のテキスト文書も統合の対象として考えていく予定である．

[謝辞]

本研究は，文部科学省科学研究費補助金特定領域研究「情報学」(課題番号 13224087) の助成のもとに行われました．

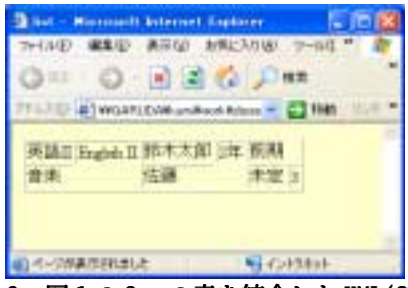


図 8 図 1 の 2 つの表を統合した XML(SVM)

Fig.8 XML list using SVM

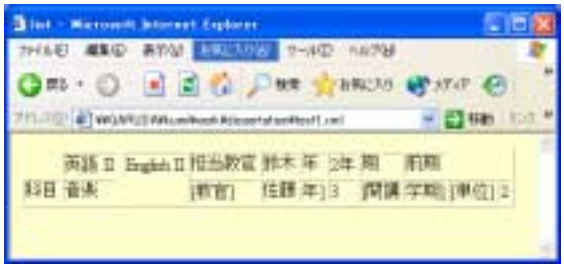


図 9 図 1 の 2 つの表を統合した XML(HMM)

Fig.9 XML list using HMM

[文献]

[1]. Wei Han, David Buttler, Calton Pu, “Wrapping Data into XML”, ACM SIGMOD Record Vol.30, No.3 Sep (2001).

[2] Minoru Yoshida, Kentaro Torisawa, Jun'ichi Tsujii, “Extracting ontologies from World Wide Web via HTML tables”, Proceedings of the Pacific Association for Computational Linguistics 2001 page332-341(2001).

[3] T.Joachims, “Text Categorization with Support Vector Machines: Learning with Many Relevant Features”, Proceedings of the European Conference on Machine Learning, Springer, (1998).

[4] Dane Freitag, Andrew McCallum, “Information Extraction with HMM Structures Learned by Stochastic Optimization” Proceedings of AAAI(2000)..

[5] Vladimir N. Vapnik, “The Nature of Statistical Learning Theory”, Springer, (1995).

[6] 吉田稔,鳥澤健太郎,辻井潤一, “表形式からの情報抽出手法”, 言語処理学会第 6 回年次大会発表論文集, pp. 252--255, 石川, March, (2000).

板井 久美 Kumi ITAI

現在，東京大学大学院情報理工学系研究科修士課程在学中．

高須 淳宏 Atsuhiro TAKASU

国立情報学研究所助教授．1989 東京大学大学院工学系研究科博士課程終了，工学博士．データベースシステム，文書画像処理，機械学習の研究に従事．電子情報通信学会，人工知能学会，ACM, IEEE 各会員．

安達 淳 Jun ADACHI

国立情報学研究所教授．東京大学大学院情報理工学系研究科教授を併任．1981 東京大学大学院工学系研究科博士課程終了，工学博士．オンライン情報システム，情報検索，電子図書館システムの研究に従事．電子情報通信学会，情報処理学会，日本データベース学会，ACM, IEEE 各会員．