社団法人 情報処理学会　研究報告
IPSJ SIG Technical Report

2003－FI－ 72　(12)
2003－NL－157　(12)
2003／9／30

機械学習による科学技術論文からの書誌情報の自動抽出

阿辺川 武[†]　難波英嗣[‡]　高村大也[§]　奥村 学[§]

概　要

本論文では，電子化された学術論文から，その論文ファイルの書誌情報および参考文献の書誌情報を抽出する手法を提案する．両書誌情報の抽出ともにサポートベクトルマシンによる機械学習手法を使用し，論文ファイルの書誌情報には，視覚的素性と言語的素性を用いることで，また参考文献の書誌情報抽出には各フィールドの出現順を制約に組み入れることで高精度で抽出が出来るようになった．

Automatic extraction of bibliography with machine learning

Takeshi ABEKAWA[†]　Hidetsugu NANBA[‡]　Hiroya TAKAMURA[§]　Manabu OKUMURA[§]

Abstract

In this paper, we propose an extraction method of bibliography using support vector machines. We use visual and linguistic features for extracting bibliography of a paper, and use field order for extracting reference infomation. Our method leads to high precision extraction.

1　はじめに

従来，学術論文の流通は印刷物を媒体にして行なわれていたが，近年の WWW での公開や CD-ROM などによる配布によって論文が電子的に流通するようになった．また e-Print archive*のように論文の著者が自分で投稿できるシステムや，WWW 上で公開されている論文を収集し，その論文の検索機能を提供するサイトが存在する．情報科学分野の論文を対象にした検索システムでは，CiteSeer(Research Index)[3] などが有名である．しかし日本語の情報科学分野の論文の検索を可能にした WWW 上のシステムは，筆者の知る限り存在しない．

そこで筆者らは，WWW 上で公開され，日本語もしくは英語で記述された論文を検索するシステム PRESRI[†] を開発している．現在，タイトル・著者名からの検索機能を提供する他，論文間の参照関係を解析しグラフ表示することができる．また参照個所周辺の文脈を解析し参照理由を提示する機能も有している [7].

WWW で公開されている論文の数は非常に多く，手作業で 1 つずつ書誌情報を登録していく手段は現実的ではない．そこで本システムでは，自動的に電子論文から書誌情報を抽出する手段を有し，効率的に書誌情報 DB の構築を行う．本論文では，機械学習手法を用いて電子化された学術論文から，論文自体の書誌情報 (以下論文ファイルの書誌情報と呼ぶ) および，参考文献中の書誌情報 (参考文献の書誌情報と呼ぶ) を自動的に抽出する手法を説明する．以下 2 節では本手法におけるテキスト切り出しまでの流れを説明し，3,4 節では，論文ファイルの書誌情報および参考文献抽出手法の有効性を調べる．最後に 5 節でまとめを行う．

2　抽出の流れ

本節では，電子媒体で提供された論文ファイルから，書誌情報を抽出していく流れを説明する．図 1 にその流れ図を載せる．WWW および CD-ROM から収集された論文ファイル (フォーマットは PS もしくは PDF) は，まず PDF ファイルに統一する．PS ファイルは Ghostscript [‡]に付属の ps2pdf を用いて PDF ファイルに変換する．

次に PDF ファイルをプログラム pdftohtml[§] を用いて，位置，フォント情報つきの XML ファイルに変換する．このプログラムは PDF ファイルから行単位でテキストを抽出し，各要素に対して，ページ中の座標，フォントサイズやボールド・イタリックなどのフォント属性を付与した XML ファイルを出力する．

[†]東京工業大学大学院 総合理工学研究科
Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology
abekawa@lr.pi.titech.ac.jp
[‡]広島市立大学 情報工学部
Faculty of Information Sciences, Hiroshima City University
nanba@its.hiroshima-cu.ac.jp
[§]東京工業大学 精密工学研究所
Precision and Intelligence Laboratory, Tokyo Institute of Technology
{takamura,oku}@pi.titech.ac.jp

*http://arxiv.org/
[†]http://peter.pi.titech.ac.jp:8000/
[‡]http://www.cs.wisc.edu/~ghost/
[§]http://pdftohtml.sourceforge.net/

pdftohtml は，PDF ファイルに含まれている順番のままテキストを出力するが，PDF ファイルを作成する環境によっては，人間がページを読み進める順番とは異なる出力がなされる場合がある．そこで独自プログラムにより，正しい順番に並べ変えたり，“ffi” や “fl” のような合字を正しい文字に変換する処理を行なう．

引き続いて，本システムでは “はじめに”，“Introduction” などを手がかり語として，論文の本文が開始される前までを切り出し，論文ファイルの書誌情報として用いる．またファイル末尾の “参考文献”，“References” などの語を手がかり語として，参考文献部分を切り出す．

以上のような処理を行い書誌情報が含まれている該当箇所を切り出していくが，次のようなファイルは，現状では正しく処理されない．例えば，パスワードによって暗号化されている PDF ファイルや，ファイルにフォントが直接埋め込まれている場合などである．特に日本語の環境で作成された PDF に多く見られるが，全ての文字間に空白が挿入されていたり，逆に英単語列にも関わらず空白が全て抜けていたりする場合がある．

英語で書かれた学術論文から書誌情報を抽出する関連研究では，正しく区切られた単語の情報を使う手法が多いが，以上のようなことから本手法ではテキスト文字列を単語として扱うのではなく 1 文字単位で扱う手法をとる．

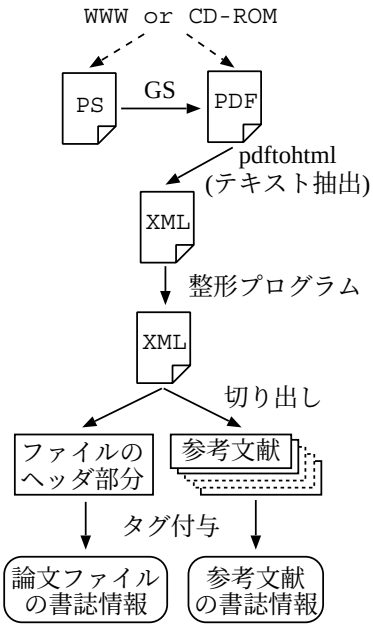


図 1: 抽出の流れ

3 論文の書誌情報の抽出

本節では，論文ファイルの 1 ページ目から書誌情報を抽出する手法を説明し，実験により本手法の評価を行なう．

学術論文では通常 1 ページ目に，その論文のタイトルと著者が必ず明記され，必要に応じて所属，連絡先，メールアドレスなどといったフィールドが記述される．そして本文が始まる論文もあれば，概要が述べられキーワードが付与されている論文もある．このように論文の種類により，特定のフィールドの有無，順番，ページ中の位置などが異なり，単純な規則だけでは書誌情報を抽出することが難しい．しかし，これらは無意味に並んでいるのではなく，特定の規則を持って配置されている．例えばタイトルは一番大きいフォントで記述され，タイトルの次に著者が記述されるといった規則である．文献 [2] では，これらの規則をもとにフローチャートを作成，書誌情報を抽出している．また文献 [6] においても同様なレイアウト知識を元にテンプレートを作成し書誌情報を抽出している．

しかし論文の書式の多様性のために，あらかじめ作成した規則だけでは対応できない場合が生じる．そこで隠れマルコフモデル (HMM) を利用した解析手法が提案されている [5]．単語を記号列とみなし，訓練データから各フィールドを 1 つの状態とした遷移モデルを構築し，状態を次々に統合していくことでモデルの簡略化を行なっていく．HMM は学習に利用する情報として，観測された記号列および状態の遷移を用いるが，今回変換したテキストには記号列の他に位置・フォント属性などが含まれ，これらの情報を統合して HMM の枠組で用いることは容易ではない．

最近，サポートベクトルマシン (以下，SVM) と呼ばれる学習モデルが，テキスト分類をはじめとした多くの分野で用いられ，大きな成果をあげている．SVM は，素性空間の次元数に影響しない高い汎化性能と，カーネル関数によって素性の組合せまでも考慮した学習が可能である．このことから，文字による言語的な素性の他に位置・フォント属性など視覚的な素性を複数有するこの書誌情報抽出問題に対して，SVM を使用することは有効であると考えられる．

3.1 SVM について

SVM は，各素性を要素とする特徴ベクトル x_i と正・負の 2 値タグ y_i のペア (x_i, y_i) で表現された n 個の訓練事例 $(0 < i < n)$ に対して，以下の式で表せる超平面で正・負のタグを正しく分離する二値線形分類器である:

$$\mathbf{w} \cdot \mathbf{x} + b = 0.$$

SVM は，訓練事例の中で他クラスと最も近い位置にいる事例 (これをサポートベクトル “SV” と呼ぶ) を基準として，そのユークリッド距離 (マージン $1/||\mathbf{w}||$) が最大となるような分離平面を求めるアルゴリズムである．実際の問題では訓練事例を正しく線形分離できる場合は稀であり，様々な改良が行なわれている．1 つはソフ

トマージン法でマージンを最大としながら，訓練事例の多少の誤りは許すものである．しかしソフトマージン法だけでは，本質的に複雑な分類問題に対して良い性能を得られるとは限らない．そこで特徴ベクトルを写像 $\phi(\mathbf{x})$ により非線形変換し，別の特徴空間で線形分離を行なう手法が用いられる．写像 $\phi(\mathbf{x})$ の内積が高次元のベクトル演算になってしまうので，カーネルトリックと呼ばれる手法を用いて，この内積をあるカーネル関数に置き換える．カーネル関数を用いることにより高次元に写像しながら，写像された特徴空間での演算をうまく避けることができる．カーネル関数には，多項式カーネルやシグモイドカーネルがよく用いられる．本手法では以下に示す d 次の多項式カーネル関数を用いる:

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j + 1)^d.$$

3.2 実験データの作成

論文ファイルの 1 ページ目からプログラムにより書誌情報を含むテキストを切り出す．切り出したテキストは行ごとにまとめられている．多くの場合，切り出した個々の行は，1 つのフィールドのみを表し，複数のフィールドが混在することは少ない．本手法ではその各行に対してフィールドのタグを付与していく．タグは表 1 にある 12 種類を設定した．

表 1: タグの種類

フィールド名	日本語・英語	日本語のみ
表題	TITLE	TITLE_E
著者	AUTHORS	AUTHORS_E
所属・連絡先	AFFILIATION	AFFILIATION_E
概要	ABSTRACT	ABSTRACT_E
キーワード	KEYWORD	KEYWORD_E
E メール	EMAIL	
その他	OTHER	

今回実験の対象とした論文ファイルは，本文が日本語もしくは英語で記述された論文を利用している．書誌情報が日本語だけ，あるいは英語だけで記述されている場合は，表 1 の左側のカラムにあるタグのみを付与する．一方，日本語で既にそのフィールドが記述され，同じ内容が英語でも記述されている場合は，“_E” のついたタグを付与し，英語のみで記述されたフィールドと区別する．AFFILIATION は所属・連絡先を表し，電話番号・FAX・URL などもこれに含める．EMAIL は連絡先の 1 つだが，他の連絡先とは違い単独で出現することが多いので AFFILIATION とは異なるタグとした．OTHER は，論文ファイルの発表日時や出典やスペースのみの行など上記タグに当てはまらない行に付与した．

例として，本論文の原稿に対して行ごとにタグ付与を行うと以下のようになる．

```
<TITLE>機械学習による科学技術論文か...</TITLE>
<AUTHORS>阿辺川 武† 難波英嗣‡ ...</AUTHORS>
<ABSTRACT>概 要</ABSTRACT>
<ABSTRACT>本論文では，電子化され...</ABSTRACT>
<ABSTRACT>を抽出する手法を提案す...</ABSTRACT>
<ABSTRACT>使用し，論文ファイルの...</ABSTRACT>
<ABSTRACT>そして両者を混合した手...</ABSTRACT>
<ABSTRACT>書誌情報抽出には各フィ...</ABSTRACT>
<TITLE_E>Automatic extraction o...</TITLE_E>
<AUTHORS_E>Takeshi ABEKAWA † ...</AUTHORS_E>
<ABSTRACT_E>Abstract</ABSTRACT_E>
<ABSTRACT_E>In this paper, we...</ABSTRACT_E>
<ABSTRACT_E>We use visual and...</ABSTRACT_E>
<ABSTRACT_E>In this paper, we...</ABSTRACT_E>
<ABSTRACT_E>extracting refere...</ABSTRACT_E>
```

3.3 素性の展開

論文ファイルのヘッダ部分から切り出した各行に対して，素性を展開する．素性の種類は，テキスト行のページ中での位置・フォント属性などの視覚的な素性と，テキストに含まれる文字や単語などの言語的な素性に大別される．それぞれについて説明を行なっていく．

視覚的素性

先に述べたように，プログラム pdftohtml により変換されたテキストからは，ページ左上を原点とした座標，テキストを囲むボックスの高さ，幅，テキストのフォント属性 (サイズ，イタリック，ボールド) が得られる．本手法ではこれらの中から次のものを素性として用いる．

- ページ上部からの距離
  ページの高さで除算し，$(0 \leq x \leq 1)$ に正規化する．
- ページ左部からの距離
  ページの幅で除算し，$(0 \leq x \leq 1)$ に正規化する．
- テキストを囲むボックスの幅
  ページの幅で除算し，$(0 \leq x \leq 1)$ に正規化する．
- センタリングされているか？
  ボックスがページ左右の端から均等な距離にあれば 1，それ以外は 0.
- フォントサイズの順番
  ページ内のフォントの大きさ順を求め，その順番を 1 としそれ以外を 0 とする 5 値のベクトルに変換する．例えばフォントサイズが 3 番目に大きい場合 (0,0,1,0,0) をとる．5 番目以降はすべて (0,0,0,0,1) とする．
- 行にボールド体の文字を含むかどうか？
  テキストがボールドなら 1，それ以外なら 0.

言語的素性

言語的素性では，単語の情報や文字情報を利用する．先で，変換したテキストでは単語として扱うことが難

しいことを述べたが，行頭の単語だけは正規表現によって識別することが可能である．そこで本手法ではタグの特徴を表す行頭の文字列を素性として用いる．

- 概要
  行頭に “概要—要旨—要約” もしくは “abstract” の文字列が存在するか？
- キーワード
  行頭に “キーワード” もしくは “Keyword” の文字列が存在するか？

そしてテキスト行を構成する文字列に対しての素性を用いる．

- 表 2 に分類された文字クラスが存在するか？

言語的素性は以上であげた 12 種類あり，それぞれ $\{0, 1\}$ の 2 値をとる．なお全角英数字は半角に変換している．

表 2: 文字クラス

文字種	正規表現
アルファベット	[A-Za-z]
数字	[0-9]
ひらがな	[ぁ-んゝゞ]
記号類	[~-;:[]{}&/　。- 〓？%…◆○／（）]
区切り文字	[’, ”‘、,．()“ ”「」『』:˚]
アットマーク	@
ピリオド	.
スペース	␣
コンマ	,
漢字・カタカナ	上記以外

3.4 評価実験

以上で述べた手法に対して評価実験を行なう．実験に用いた論文は，表 3 に示される出典から収集をおこなった．筆者の手元にあるまとまった数の論文ファイルは，情報系の分野 (特に自然言語処理) が多く，また同じ学会では論文のフォーマットも指定されている場合があることから書式の多様性があるとはいえない．そこで WWW から収集した論文ファイルを実験データに追加した．

実験には上記の媒体から収集した論文 945 ファイルについて書誌情報の抽出をおこなった．本実験では修士・博士論文や報告書のような表紙の存在する論文ファイルは対象外とした．実験データは，言語に関係なく両者を統合して 1 つのデータ集合とした．これを 5 分割し交差検定を行なった．

SVM の学習モデルの作成，およびタグ付与には YamCha¶ を利用した．視覚的素性における距離を正規化した素性については，小数第 3 位以下を切り捨て，そのまま YamCha に対する素性とした．YamCha はこれをそのまま 100 個の異なる素性として扱う．各パラメーターは予備実験を行ない精度の良かったものを用いている．多項式カーネル関数の次元数は $d = 2$ を使用し，素性として与えるウィンドウサイズは $-4..+4$，そして直前 4 個までの既に付与されたタグを素性として与えている．

実験では，素性の集合により 3 つの学習モデルを構築した．

- 素性セット A
  視覚的素性のみを用いたもの．言語に依存しない．
- 素性セット B
  言語的素性のみを用いたもの．各行に含まれる文字列情報のみ．
- 素性セット A+B
  すべての素性を用いたもの．

評価については次の 2 つの尺度を用いる．

- F-measure
  行単位での Recall,Precision を求め，$\beta = 1$ として算出したものを使用．通常複数のタグは 1 つにまとめて評価を行なうが，本実験では行単位で行なった．
- タグ正解率
  1 論文ファイル中に含まれる同一のタグがすべて一致したものを正解とする．

実験の結果を表 4 に掲載する．

3.5 考察

まず素性集合別についてわかったことについて述べる．視覚的素性と言語的素性を比較すると全体の正解率では大きな差はないが，個々のタグについて見るとそれぞれ特徴が見受けられる．視覚的素性では TITLE の正解率の精度が格段に良い．これは論文ではタイトルは目立つようにページ上部の中央なおかつ一番大きいフォントで記述されるといった多くの特徴があるからである．ついで ABSTRACT の正解率がよい．ABSTRACT は行数が多い上，前後の行と左右の位置は同じである場合が多いので前後を含めて ABSTRACT だと認識できると考えられる．しかし ABSTRACT の直後に出現しやすい KEYWORD については，視覚的素性が ABSTRACT とほぼ同一なため誤認識してしまう傾向がある．

一方，言語的素性を用いることにより KEYWORD は精度良く識別できた．これは KEYWORD の始まりを示す単語を正しく認識できているからである．また EMAIL の精度が良いのは “@” の有無により容易に識別できたからであると思われる．

視覚的素性と言語的素性をすべて使用した素性セット A+B がどのタグにおいても一番精度がよかった．これは SVM の多数の素性を効果的に取り扱える汎化性能の良さを示している．

¶http://cl.aist-nara.ac.jp/~taku-ku/software/yamcha/

表 3: 実験に用いた論文

出典	ファイル数	論文ファイル 英語	論文ファイル 日本語	参考文献 英語	参考文献 日本語
Association for Computational Linguistics(ACL2003)	65	65	0	150	0
Computational Linguistics(COLING2002)	140	140	0	150	0
電気情報通信学会 2003 年総合大会	150	8	142	223	147
情報処理学会第 65 回全国大会 (2003)	177	1	176	150	236
第 17 回人工知能学会全国大会 (2003)	208	5	203	152	244
自然言語処理研究会第 146〜155 回	98	2	96	150	232
WWW から収集した論文	107	73	34	147	96
計	945	294	651	1122	955

表 4: 論文ファイルから書誌情報の抽出

タグ	行数	タグ数	素性セット A (視覚的素性) F 値	素性セット A (視覚的素性) タグ正解率	素性セット B (言語的素性) F 値	素性セット B (言語的素性) タグ正解率	素性セット A+B (すべて) F 値	素性セット A+B (すべて) タグ正解率
TITLE	1,215	945	0.962	0.959	0.900	0.884	0.976	0.972
AUTHORS	1,661	940	0.870	0.817	0.835	0.767	0.931	0.899
AFFILIATION	2,124	882	0.838	0.821	0.876	0.805	0.935	0.906
EMAIL	528	323	0.643	0.538	0.964	0.960	0.969	0.960
ABSTRACT	6,777	598	0.954	0.898	0.974	0.910	0.986	0.959
KEYWORD	103	70	0.483	0.361	0.882	0.863	0.909	0.858
OTHER	1,481	651	0.948	0.902	0.938	0.914	0.968	0.932
TITLE_E	570	455	0.846	0.830	0.928	0.926	0.960	0.962
AUTHORS_E	939	459	0.820	0.747	0.876	0.837	0.925	0.886
AFFILIATION_E	722	426	0.802	0.809	0.853	0.834	0.912	0.892
ABSTRACT_E	773	99	0.806	0.573	0.851	0.719	0.895	0.794
KEYWORD_E	47	37	0.449	0.394	0.790	0.786	0.840	0.786
全体	16,940	945	0.894	0.532	0.920	0.527	0.959	0.692

4 参考文献からの書誌情報の抽出

次に，論文の末尾に記述されている参考文献から書誌情報を抽出する方法を説明する．1 ページ目における論文ファイルの書誌情報と同様に，多様な書式が用いられる．参考文献の例を以下に挙げる．

1. S. Lawrence, C.L. Giles, K. Bollacker, “Digital libraries and autonomous citation indexing”, *IEEE Computer*, vol. 6, no.4, pp. 67-71, 1999.
2. Lawrence, S., Giles, C.L., Bollacker, K.(1999). Digital libraries and autonomous citation indexing. *IEEE Computer, 32* (**6**),67-71.
3. S.Lawrence,C.Giles,K.Bollacker,Digitallibrariesandautonomouscitationindexing,IEEEComputer32(6):67-71(1999)

同じ論文に対する参考文献でも，論文ファイルにより大きく書式が異なる．代表的なものでは出版年の位置が著者の後ろにあるか，末尾にあるかといったものである．また参考文献例 3. のように PDF ファイルからテキストを抽出する際に，空白がまったく認識されないといった場合も存在する．

4.1 実験データの作成

参考文献の書誌情報に付与するタグは次の 6 つのタグである．OTHER を除き，1 つの参考文献につき 2 回以上は出現しない．

- AUTHORS
  論文の著者．複数人数からなる場合でもまとめて 1 つのタグとする．
- TITLE
  論文の表題
- SOURCE
  論文の出典・ボリューム・ナンバー，書籍の場合は出版社．URL があるときもある．出典を表わ

していると思われる情報が連続していればすべて1つのタグとみなす.

- DATE
  出版年．“September 2003” のように月が入る場合もある．1つの参考文献で複数の日付を表す文字列がある場合，同じ論文ファイル中の参考文献を参考にしながら決定する．
- PAGE
  文献が掲載されているページ．“pp.1–8”，“2138-2152”，“pp.34” のような文字列をページとしている．また日本語の文献では “10-18 ページ” などのような記述もある．
- OTHER
  “to appear” など上記以外で分類できない文字列．また出典に関する情報が PAGE や DATE により分割されている場合，後半を OTHER とする場合がある．

参考文献例の 1 についてタグ付与を行なうと次のようになる．ここでは記述していないが各タグの間にある文字列に対しては “NONE” というタグを付与している．

- <AUTHORS>S. Lawrence, C.L. Giles, K. Bollacker </AUTHORS>, “<TITLE>Digital libraries and autonomous citation indexing</TITLE>”, <SOURCE> *IEEE Computer*, vol. 6, no.4 </SOURCE>, <PAGE> pp. 67-71</PAGE>, <DATE> 1999</DATE>.

4.2 HMM を用いたタグ付与

本手法では，論文ファイルの書誌情報と同様に SVM を用いてタグ付与を行なっていくが，先行研究で用いられている HMM を用いた手法 [1, 4] を比較する対象として用いる．

HMM による手法を実装する際，どんな単位で状態遷移を考えていくかが問題となる．先の関連研究では，単語単位で状態遷移を行なっている．論文ファイルの書誌情報の抽出でも述べたが，日本語では形態素解析を行なわなければならない点，そして参考文献例 3 のように PDF ファイルからのテキスト抽出において正しく単語が認識できなくなる点などから，単語を単語として識別することは難しいので，文字単位で状態遷移を行うことにする．つまり 1 文字が出力記号列になり，その文字を含む文字列に付与されているタグが状態となる．

HMM は状態遷移確率と出力確率の 2 つの確率を組み合わし，観測された記号列に対して最大の確率を与える状態遷移列を求める手法である．状態 q_i から状態 q_j への遷移の回数を $c(q_i \to q_j)$ とし，それぞれの状態で出現した記号の回数 $c(q \uparrow \sigma_k)$ としたときの遷移確率，出力確率は次のようになる:

- 遷移確率

$$P(q_i \to q_j) = \frac{c(q_i \to q_j)}{\sum_{q_i, q_j \in Q} \quad c(q_i \to q_j)}$$

- 出力確率

$$P(q_i \uparrow \sigma_k) = \frac{c(q_i \uparrow \sigma_k)}{\sum_{\sigma_k \in \Sigma} \quad c(q_i \uparrow \sigma_k)}$$

訓練データから 2 つの確率を求め，参考文献を構成する文字列に対し，Viterbi アルゴリズムを用いて最大尤度を有する状態遷移列を算出する．実際には論文ファイルの書誌情報抽出時に用いた表 2 を用いて文字クラス化した記号列を HMM の入力としている．

モデルの構築

HMM を利用する際は，状態遷移のモデルを構築することが望ましい．全ての状態から全ての状態へ遷移可能なエルゴード的モデルも可能であるが，事前知識より状態遷移になんらかの制約を課すことが出来る場合，精度向上が期待できる．参考文献の書誌情報では，AUTHORS が最初に出現しやすいといった制約を課すことが出来る．モデルを構築する手法には，訓練事例で観測された状態遷移を 1 つの大きなモデルとして結合し，次々と状態をマージしていく方法 [4] がある．ただこの方法では，DATE のように複数の場所で出現するタグを考慮するとモデルが複雑になってしまい，少ない訓練事例では精度が得られないと予想される．

そこで本手法では，DATE や PAGE などの文字列に規則性があることに注目し，HMM を用いる前に，あらかじめ正規表現を使用してこれらのタグを抽出してしまう方法をとる．つまり状態遷移モデルに DATE と PAGE の状態を含めない．すると参考文献中のタグの出現順はある程度固定され，簡単な状態遷移モデルを構築できる．今回の手法で構築したモデルを図 2 に示す.

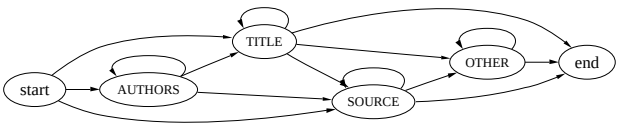


図 2: HMM の状態遷移モデル

ただし，このモデルを使用すると，モデルで示される状態遷移列以外のタグ順を持つ参考文献を正しく解析することが出来なくなる．今回使用した訓練事例に対してタグの出現順を求めると表 5 のようになり，割合にすると 29/2077=約 1.4%の参考文献が正しく解析できないことになる．

表 6: 評価単位

正解	AUTHORS		DATE		TITLE		
	J. Connan and C.W. Omlin	(	2000	)	Bibliography Extraction with Hidden Markov	Models	.
結果	○ AUTHORS		○ DATE		× TITLE	× SOURCE	

表 5: フィールドの出現順 (DATE,PAGE は除く)

個数	図 2 のモデルで認識可能
1670	AUTHORS, TITLE, SOURCE
138	AUTHORS, TITLE, SOURCE, OTHER
107	AUTHORS, SOURCE
40	AUTHORS, TITLE
38	TITLE, SOURCE
23	SOURCE
15	AUTHORS, SOURCE, OTHER
6	AUTHORS, TITLE, OTHER
6	TITLE
2	TITLE, SOURCE, OTHER
2	SOURCE, OTHER
個数	図 2 のモデルで認識不可能
15	TITLE, AUTHORS, SOURCE
6	AUTHORS, SOURCE, TITLE
2	AUTHORS, TITLE, OTHER, SOURCE
1	TITLE, OTHER, SOURCE
1	AUTHORS, SOURCE, TITLE, OTHER
1	SOURCE, TITLE
1	AUTHORS, OTHER, SOURCE
1	AUTHORS, OTHER, TITLE, SOURCE
1	AUTHORS

4.3 SVM を用いたタグ付与

HMM を使った手法に続き，SVM を使った手法を説明する．HMM の手法と同様に，文字単位でタグ付与を行なっていく．素性には HMM で用いた文字クラスと，加えて文字そのものを使用し，素性として与えるウィンドウサイズは予備実験で精度のよかった前後 7 文字を用いた．また 7 つ前までの既に付与されたタグを素性として利用する．SVM のカーネル関数には論文ファイルの書誌情報の抽出と同様に多項式関数を使用したが，次数 d は 3 に設定した．解析手法の違いにより次の 3 つの手法を作成した．

- SVM1
  訓練事例に対して，何も制限をせずに解析する手法
- SVM2
  HMM と同様に正規表現により，あらかじめ DATE, PAGE を抽出する手法
- SVM3
  SVM2 に加え，付与すべきタグ候補を状態遷移モデルにあわせて制限する手法

SVM3 についてもう少し詳しく説明する．SVM1,SVM2 では，どの位置でもすべてのタグ候補から，SVM により最大の分離平面との距離を持つタグを採用する．一方 SVM3 では現在位置の 1 つ前までの既に付与されたタグ列をモデルと照合し，モデルに適合するタグ候補だけの中から最大の距離を持つタグを採用するという手法をとる．よって HMM と同様，解析できる参考文献は状態遷移モデルに依存し，すべての参考文献が正しく解析できるわけではない．

4.4 評価について

提案した手法についての評価実験を行なう前に評価する単位について説明する．提案手法により 1 文字単位でタグ付与を行ない，同じタグを持つ文字を連結して文字列を作成する．表 6 のように同じタグを有する文字列単位で正解データの文字列と比較する．アルファベット，数字，漢字，かな以外の記号だけが異なる場合は，一致しているとみなす (ex. AUTHORS).

次に評価尺度について説明する．

- タグの正解率

$$\frac{\text{文字列が一致したタグ数}}{\text{テストセット中のタグ数}}$$

- 参考文献全体の正解率

$$\frac{\text{タグがすべて一致した参考文献数}}{\text{テストセット中の参考文献数}}$$

4.5 評価実験

実験データには，論文ファイルの書誌情報抽出に用いた論文と同じものを使用し，参考文献を抽出している．しかし抽出したすべての参考文献を使用せず，ランダムに選択した参考文献のみを使用しており，その総数を表 3 の参考文献のカラムに示す．論文ファイルの書誌情報と異なり，日本語と英語では学習されるモデルが大きく異なると予想されることから，言語別に分割して，それぞれの言語に対して実験を行なった．

評価は，実験データを 5 分割した上で交差検定を行なった．日本語・英語の参考文献に対する結果を表 7 に掲載する．

4.6 考察

HMM と SVM1, SVM2 を比較すると，HMM の方が精度がよい．これは HMM に使用した状態遷移モデルが有効に機能しているからであると思われる．SVM に対し HMM と同様なタグの並びの制約を組み込んだ

表 7: 実験結果

	日本語					英語				
タグ	個数	HMM	SVM1	SVM2	SVM3	個数	HMM	SVM1	SVM2	SVM3
AUTHORS	919	**0.913**	0.897	0.903	0.903	1084	0.907	0.893	0.898	**0.981**
TITLE	883	0.818	0.818	0.824	**0.840**	1044	0.785	0.834	0.839	**0.941**
SOURCE	923	0.756	0.756	0.794	**0.805**	1100	0.674	0.743	0.830	**0.848**
DATE	853	**0.988**	0.942	**0.988**	**0.988**	1061	**0.957**	0.886	**0.957**	**0.957**
PAGE	465	**0.989**	0.945	**0.989**	**0.989**	652	**0.956**	0.868	**0.956**	**0.956**
OTHER	64	**0.538**	0.313	0.313	0.201	106	0.538	0.538	**0.769**	0.461
参考文献全体	955	0.738	0.706	0.732	**0.748**	1122	0.651	0.700	0.781	**0.816**

SVM3 は，参考文献全体の精度を見ると日本語・英語共に HMM より精度が良い．

これにより参考文献のタグの並びのように順番に制約がある場合，その制約を組み込む方が精度は向上することがわかる．SVM は素性として一定のウィンドウサイズを扱えること可能であることから，そのウィンドウサイズ以内のタグ列の制約を識別に反映させることが出来る．それでもウィンドウの外の情報は扱うことが出来ないため，タグ列の制約は局所的なものになってしまう．例えば参考文献では，AUTHORS,TITLE の後にもう一度 AUTHORS が来ることはない．TITLE を構成する文字列がウィンドウサイズ以上の場合，以前に AUTHORS のタグが存在したという情報が扱えなくなる．したがって今回の実験のように，HMM のモデルをタグ列の並びの制約として扱った場合，精度が向上したものと思われる．

実験結果を詳しくみると英語と日本語では，タグ列の制約を導入した場合の精度向上の度合が異なる．これは各タグの文字列長が原因であると考えられる．英語と日本語の参考文献で各タグの平均文字列長を求めたものが表 8 であるが，英語と日本語では大きく異なる．英語の平均文字列長は今回のウィンドウサイズ $15(-7..+7)$ を大きく越えており，タグ列の制約が精度向上につながったと考えられる．一方，日本語のタグの平均文字列長は，ウィンドウサイズとほぼ同じ長さであり，すでに素性の中にタグ列の制約が含まれていたと思われる．英語の参考文献の実験においてタグの制約を反映するウィンドウサイズを拡大すれば，状態遷移モデルによるタグの制約を導入することなく高精度を維持できるかは，今後確認していきたい．

表 8: タグ文字列の平均長

	英語	日本語
AUTHORS,TITLE,SOURCE のみ	43.44	17.74
全て	30.93	14.01

5 まとめ

本論文では，電子化された学術論文から，その論文ファイルの書誌情報および参考文献の書誌情報を抽出する手法を提案した．両書誌情報ともにサポートベクトルマシンを使用し，論文ファイルの書誌情報には，視覚的素性と言語的素性を用いることで，また参考文献の書誌情報抽出には各フィールドの並び順を考慮することで高精度で行なうことが出来るようになった．本システムは多言語化を想定しており，言語的素性について単語の素性を対応させれば，その他の素性を修正せずに容易に拡張できると思われる．参考情報に関しては訓練データさえ用意し，正しく言語に適した学習モデルが構築できれば現時点でも多言語化に対応できると思われる．また複数ファイルから得られた書誌情報の同定をする際に，抽出した個々のフィールドの情報を利用できれば効果的であると思われる．

参考文献

[1] J. Connan and C.W. Omlin. Bibliography extraction with hidden markov models. 2000.

[2] Ying Ding, Gobinda Chowdhury, and Schubert Foo. Template mining for the extraction of citation from digital documents. In *Proceedings of Second Asian Digital Library Conference*, pp. 47–62, 1999.

[3] Steve Lawrence, C. Lee Giles, and Kurt Bollacker. Digital libraries and autonomous citation indexing. *IEEE Computer*, Vol. 32, No. 6, pp. 67–71, 1999.

[4] Andrew McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Building domain-specific search engines with machine learning techniques. In *Proceedings of AAAI-99 Spring Symposium on Intelligent Agents in Cyberspace, 1999.*, 1999.

[5] Kristie Seymore, Andrew McCallum, and Roni Rosenfeld. Learning hidden Markov model structure for information extraction. In *AAAI 99 Workshop on Machine Learning for Information Extraction*, 1999.

[6] 冨田陽明, 大囿忠親, 新谷虎松. レイアウト知識を用いた PDF 形式論文からの情報抽出システム. 情報処理学会第 65 回全国大会, pp. 2–229 – 2–230, 2003.

[7] 難波英嗣, 奥村学. 論文間の参照情報を考慮したサーベイ論文作成支援システムの開発. 自然言語処理, Vol. 6, No. 5, pp. 43–62, 1999.