2004－NL－162　(6)
2004／7／15

Web ページ内の目的部分の自動抽出

新納浩幸　　佐々木稔

茨城大学工学部システム工学科　　茨城大学工学部情報工学科
shinnou@dse.ibaraki.ac.jp　　sasaki@cis.ibaraki.ac.jp

本論文では Web ページから目的部分のテキストを自動抽出する手法を提案する．本論文で扱うタスクは，Web ニュースのページからそのニュース記事のタイトルと本文を抽出するというタスクである．

本手法ではまずテキストブラウザを利用して， Web ページをテキスト化する．このテキストファイルをもとに抽出規則の学習を行なう．具体的には行を事例とした START/END 法とクラス間の出現順序や位置情報などの制約を取り入れた状態遷移図を利用する．本手法は Wrapper 学習の一種であるが，従来までの Wrapper 学習とは異なり，HTML のタグを抽出手がかりとして使わない．そのためにサイトの異なるページに対しても適用できる抽出規則を学習することが期待できる．実験では訓練データの元になったサイトから取り出したページと別サイトから取り出したページを使って抽出実験を行なった．単純なレイアウトのページであれば，高精度に抽出できたが，複雑なレイアウトのページでは抽出に失敗していた．また本手法は様々な応用が可能である．ここでは対訳コーパスの自動構築に応用できることを示した．今後は自然言語の情報を素性に組み入れる．本タスクに関しては，タイトルの判定の精度を高めて改善を行なう．

Automatic extracion of target parts from a web page

Hiroyuki Shinnou
Department of Systems Engineering, Ibaraki University
shinnou@dse.ibaraki.ac.jp

Minoru Sasaki
Department of Computer and Information Sciences, Ibaraki University
sasaki@cis.ibaraki.ac.jp

This paper proposes a new method to extract target parts from a web page. Our task is to extract the title and the article from a web news page.

First, our method translates the HTML formatted web page into the plain text file, and then learns the extraction rule by using such plain text files. In concrete, we use the START/END method using a line as an instance and the state transition diagram incorporating constrains of the class sequence, the distance between classes and so on. Our method is a Wrapper learning method. However, our method does not use HTML tags as clues for extraction, unlike traditional Wrapper learning methods. Therefore, our method might be expected to learn the extraction rule which can be applied to other various site pages. We conducted experiments using other pages on the same site and pages on the other site. The extraction rule learned by our method worked well for pages with the simple layout. However, it failed for pages with the complex layout. Moreover, we also conducted the experiment constructing a bilingual corpus automatically, to introduce the wide usefulness of our method. In future, we will use the language information as the features, and improve the judgment of the title part for this task.

1 はじめに

本論文では Web ページ，すなわち HTML で記述されたテキストから目的部分のテキストを自動抽出する手法を提案する．本論文で扱うタスクは，Web ニュースのページからそのニュース記事のタイトルと本文を抽出するというタスクである．

Web は情報の宝庫である．そこから特定の情報を収集し，再構成してユーザーに提示するシステムが数多く提案されている (例えば [3])．そのようなシステムで鍵となる技術は Web ページから目的部分を自動的に抽出する技術であり，その技術は自然言語処理を使うアプローチと Wrapper 構築のアプローチに分類できる[1]．前者では，Web ページを文書データとして扱い，ページに書かれている内容に関する部分を抽出する．後者では，Web ページを (半) 構造化データとして扱い，ページ内の表のデータや，定型的なレイアウトをもつ文書データなどを抽出対象とする．本論文のタスクに対しては後者のアプローチが使われる．

Web ニュースのページからタイトルと本文を抽出するのは，一見，簡単な処理に思われる．確かに，同じサイト内のニュース記事であれば，レイアウトが固定されているので，目的を達する Wrapper を書くのはそれほど困難なことではない．現に Web ニュースを収集，再構成してユーザーに提示するシステムは Google News[2] や Bulknews[3]で既に行なわれている[4]．Yahoo や Goo などのポータルサイトでも Web ニュースを収集して，記事の一覧を提供している．これらのシステムは基本的には RSS (Rich Site Summary/RDF Site Summary) を利用していると思われるが，RSS を提供していないニュースサイトへは対応できない．また RSS は文字通り Summary であり記事本文の抽出はできない．記事本文を抽出する場合には，サイト毎の何らかの Wrapper を作成していると思われる．Web ニュースを収集して要約を提供する NewsInEssence においてもサイト毎に Wrapper が手作業で作られている [8] [5]．しかし，サイト毎に Wrapper を作成するのは手間がかかるし，新しいサイトが現れればまた Wrapper を作成しなくてはならない．また同一サイトでレイアウトが変更されることもある．

そこで本論文では機械学習を利用した Wrapper の自動構築を行なう．

従来から機械学習を利用した Wrapper の (半) 自動構築技術が研究されてきている．Kushmerick の LR Wrapper[4]，Muslea の STALKER[6]，Soderland の WHISK[10] などが代表的な研究である．しかしこれらの研究の抽出対象は，表形式のリストのように 1 ページ内にレコードのインスタンスが繰り返し出現するタイプのものであり，我々のタスクには合わない．当然，それらの手法でも本タスクに適用可能ではあるが，その場合には，サイト毎に，訓練データを作成し，学習を行なう必要がある．これはかなり煩わしい．HTML データを解析しながら，訓練データを作成するくらいなら，Wrapper を直接作った方が簡単とも思える．もちろん，複数のサイトからページを集めて学習を行なえば，それらサイトをカバーする Wrapper が作成できる可能性はある．しかし従来の Wrapper では，抽出箇所を挟むタグを抽出の手がかりとして利用しているので，異なるサイト間で共通に使える Warpper の構築は困難である．

本論文は異なるサイト間で共通に使える抽出規則の自動構築を目指す．基本となるアイデアは HTML ファイルをテキストブラウザーを利用してテキストファイルに変換し，そこから抽出規則を学習させることである．HTML ファイルはサイトが異なると全く異なるが，テキストファイルではその違いがある程度吸収される．またテキストブラウザーから作成されたテキストファイルには HTML のタグが取り除かれているので，ここで学習される抽出の手がかりとなる情報はサイトを通して有効な情報であることが期待できる．またテキストファイルに変換すると訓練データを作成する際のタグ付けが容易であるというメリットもある．テキストファイルに変換した後は，行を事例とした START/END 法 [9] を適用する．本タスクの場合，削除箇所と抽出箇所が 1 つの行の中に混在することはない．そのため各行毎にその行を抽出するかどうかを判定すればよい．START/END 法では各事例にクラスの確率を与え，動的計画法によって最適なクラス列を決定する．各行のクラスが決定されればそこから抽出する行，つまり抽出箇所が特定できる．また本タスクではクラスの出現順序にある程度制約があるため，そのような制約を取り込んだ状態遷移図を作成し，その後に動的計画法を適用する．

本論文では 2 つの実験を行う．

1 つは学習した抽出規則の評価実験である．まずレイアウトが異なる 5 つのサイトを選び，各サイトから Web ニュースを約 50 ページづつ集め，これらを訓練データとして，抽出規則の学習を行う．そして同じ 5 つのサイトから，再び，Web ニュースを約 50 ページづつ集め，これらを評価データとして評価を行なう．さらにそれら 5 つのサイトとは別の 6 サイトを選び，各サイトから Web ニュースを約 10 ページづつ集め，これらのページに対しても抽出実験を行ない，評価を行う．

もう 1 つの実験は，本手法の有用性を示した実験である．本手法は Web ニュースの収集，再構築だけに応用されるわけではない．ここでは新しい応用として対訳コーパスの自動構築を取り上げる．日本

[1] Wrapper とは Web ページから特定部分を抽出するプログラムのことである．

[2] http://news.google.com/

[3] http://bulknews.net/

[4] Bulknews は RSS を利用して，Web ニュースからタイトルだけ取りだして提示している．タイトルに元のページへのリンクがある．

[5] NewsInEssence では HTML のニュース記事を XML に変換する．このプログラムが本論文での Wrapper に相当する．

語の Web ニュースの中には英語の Web ニュースを翻訳して提供しているものがある．そのようなページはオリジナルのニュースページへのリンクが張られている．そのため対訳関係にあるページを特定でき，それら Web ニュースからタイトルと本文を抽出することで，対訳コーパスを自動構築することができる．本実験ではこのアプローチにより対訳コーパスを自動構築できることを示す．

2 抽出規則の学習

2.1 テキストブラウザーの利用

抽出対象の Web ニュースのページ例を図 1 と図 2 に示す．図には抽出する部分も示している．

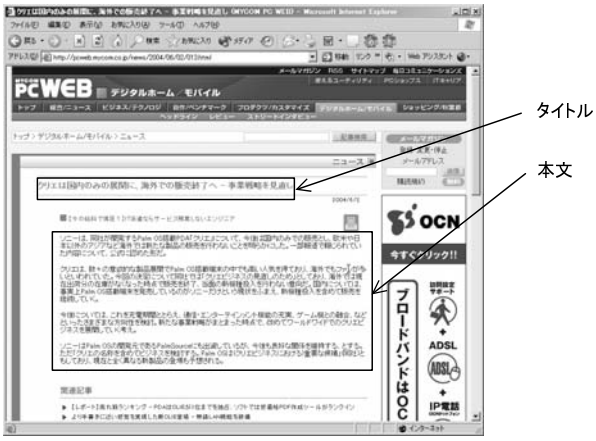


図 1: Web ニュース (1)

図 2: Web ニュース (2)

図 1 と図 2 は異なるサイトから取り出した Web ニュースなので，レイアウトが異なっている．タイトルや本文を挟むタグを手がかりにするアプローチでは，図 1 と図 2 に共通して使える抽出規則を作成することは難しい．

本論文では，Web ニュースの HTML ファイルを扱わず，テキストブラウザー `lynx` [6]を利用する．`lynx` に `-dump` のパラメータを与えることで、HTML のタグが除去され，ある程度整形されたプレーンなテキストを得ることができる。

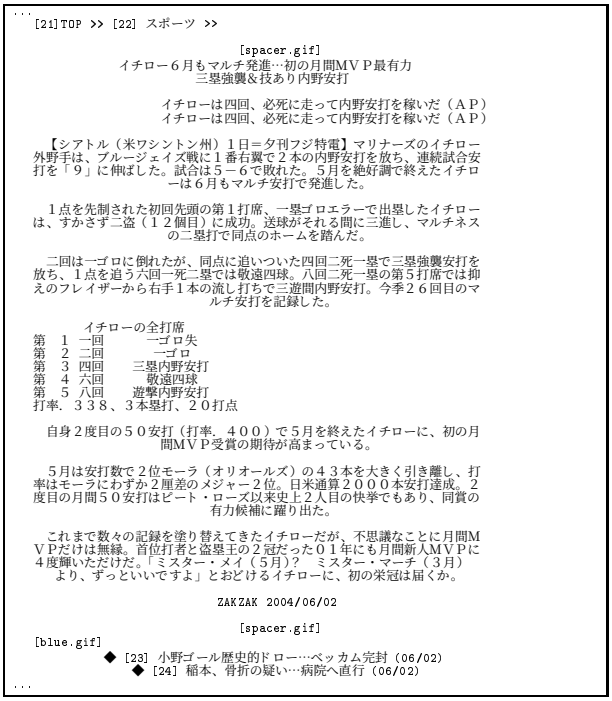
図 1 の HTML ファイルをテキストファイルに変換した結果の一部を図 3 に示す．また図 2 の HTML ファイルをテキストファイルに変換した結果の一部を図 4 に示す．

こうして作成されたテキストファイルから記事のタイトルと本文を抽出する規則を構築する．この場合，抽出規則が利用する情報は，例えば次の行が空行であるとか，その行が字下げされているかなどといった情報になり，その多くはその文字列自身がもつ情報となる．これは HTML のタグの情報とは本質的に異なる．また HTML のタグの違いを吸収した情報ともみなせる．例えば，
のタグや<P>のタグの違いは吸収されている．このようなことから，図 3 や図 4 のテキストファイルに共通して使える抽出規則を構築できる可能性がある．

```
...
    * [33]【特別企画】撮りっぱなしでは終わらない、とことんできる「デジタ
      ルカメラ徹底活用術」!! デジカメシアター Play@TV for PC

    * [34]【特別企画】業界初の新機能「バックアップファイルマネージャー」
      を搭載。バックアップしたイメージファイルから、直接 PC を起動ができる
      「HD 革命/BackUp Ver.5」

クリエは国内のみの展開に、海外での販売終了へ - 事業戦略を見直し

   2004/6/2
   [36] 印刷

   ソニーは、同社が開発する Palm OS 搭載 PDA「クリエ」について、今後は国内の
   みでの販売とし、欧米や日本以外のアジアなど海外では新たな製品の販売を行
   わないことを明らかにした。一部報道で報じられていた内容について、公式に
   認めた形だ。

   クリエは、数々の意欲的な製品展開で Palm OS 搭載端末の中でも高い人気を得
   ており、海外でもファンが多いといわれていた。今回の決定について同社では
   「クリエビジネスの見直しのため」としており、海外では現在出荷分の在庫が
   なくなった時点で販売を終了、当面の新機種投入を行わない意向だ。国内につ
   いては、事実上 Palm OS 搭載端末を発売しているのがソニーだけという現状を
   ふまえ、新機種投入を含めて販売を継続していく。

   今後については、これを充電期間ととらえ、通信・エンターテインメント機能
   の充実、ゲーム機との融合、などといったさまざまな方向性を検討。新たな事
   業戦略がまとまった時点で、改めてワールドワイドでのクリエビジネスを展開
   していく考え。

   ソニーは Palm OS の開発元である PalmSource にも出資しているが、今後も良好
   な関係を維持する、とする。ただ「クリエの名称を含めてビジネスを検討する
   。Palm OS は（クリエビジネスにおける）重要な候補」（同社）ともしており、現
   在と全く異なる新製品の登場も予想される。

関連記事

    * [36]【レポート】売れ筋ランキング - PDA は CLIE が 3 位までを独占、ソフ
      トでは低価格 PDF 作成ツールがランクイン
    * [37] より手書きに近い感覚を実現した新 CLIE 登場 - 無線 LAN 機能を装備

...
```

図 3: テキスト化されたページ (1)

[6] `http://lynx.browser.org/`



```
...
  [21]TOP >> [22] スポーツ >>

                                [spacer.gif]
                イチロー６月もマルチ発進…初の月間ＭＶＰ最有力
                        三塁強襲＆技あり内野安打

                        イチローは四回、必死に走って内野安打を稼いだ（ＡＰ）
                        イチローは四回、必死に走って内野安打を稼いだ（ＡＰ）

    【シアトル（米ワシントン州）１日＝夕刊フジ特電】マリナーズのイチロー外野手は、ブルージェイズ戦に１番右翼で２本の内野安打を放ち、連続試合安打を「９」に伸ばした。試合は５－６で敗れた。５月を絶好調で終えたイチローは６月もマルチ安打で発進した。

    １点を先制された初回先頭の第１打席、一塁ゴロエラーで出塁したイチローは、すかさず二盗（１２個目）に成功。送球がそれる間に三進し、マルチネスの二塁打で同点のホームを踏んだ。

    二回は一ゴロに倒れたが、同点に追いついた四回二死一塁で三塁強襲安打を放ち、１点を追う六回一死二塁では敬遠四球。八回二死一塁の第５打席では抑えのフレイザーから右手１本の流し打ちで三遊間内野安打。今季２６回目のマルチ安打を記録した。

          イチローの全打席
第　１　一回　　一ゴロ失
第　２　二回　　一ゴロ
第　３　四回　　三塁内野安打
第　４　六回　　敬遠四球
第　５　八回　　遊撃内野安打
打率．３３８、３本塁打、２０打点

    自身２度目の５０安打（打率．４００）で５月を終えたイチローに、初の月間ＭＶＰ受賞の期待が高まっている。

    ５月は安打数で２位モーラ（オリオールズ）の４３本を大きく引き離し、打率はモーラにわずか２厘差のメジャー２位。日米通算２０００本安打達成。２度目の月間５０安打はビート・ローズ以来史上２人目の快挙でもあり、同賞の有力候補に躍り出た。

    これまで数々の記録を塗り替えてきたイチローだが、不思議なことに月間ＭＶＰだけは無縁。首位打者と盗塁王の２冠だった０１年にも月間新人ＭＶＰに４度輝いただけだ。「ミスター・メイ（５月）？　ミスター・マーチ（３月）より、ずっといいですよ」とおどけるイチローに、初の栄冠は届くか。

                                ZAKZAK 2004/06/02

                                [spacer.gif]
[blue.gif]
                ◆ [23] 小野ゴール歴史的ドロー…ベッカム完封（06/02)
                ◆ [24] 稲本、骨折の疑い…病院へ直行（06/02)
...
```

図 4: テキスト化されたページ (2)

2.2 行単位の START/END 法

START/END 法 [9] は固有表現抽出のような Chunk 同定に対して利用される手法である．例えば，人名を抽出する場合には，以下のような 4 つのクラスを用意する．

HS: 人名単語列の始まりの単語
HM: 人名単語列の中間の単語
HE: 人名単語列の最後の単語
HI: 人名単語列が 1 単語

他の固有表現も同時に抽出したければ，同様に 4 つのクラスを追加すればよい．最後に，抽出対象の単語列とは無関係という N というクラスを用意する．そして入力テキストの各単語に上記で設定したクラスになる確率を与える．クラスの並びにはある制約 (例えば HM の直後には HS は現れない) があるので，その制約を満たしたクラス列で生起確率が最大になるようなものを求めれば，固有表現の抽出が行なえる．

本論文ではこの START/END 法を利用する．本タスクの場合，削除箇所と抽出箇所が 1 つの行の中に混在することはない．そのため各行毎にその行を抽出するかどうかを判定すればよい．そのため，START/END 法の事例としては，テキストファイルの行を設定すればよい．START/END 法によってタイトルの行と本文の開始行，終了行を見つけることで，タイトルと本文の抽出が可能となる．

2.2.1 クラスの設定

START/END 法では，まずクラスを設定しなくてはならない．本タスクではタイトルと本文の 2 種類のクラスがあるので 8 (= 2 ∗ 4) つのクラスと N を設定すればよいが，精度を高めるために，制約を設けてクラスの数を減らす．

まずタイトルは 1 行で書かれると仮定する．実際に 2 行になることもあるが，1 行目さえ取り出せておけば，最終的な後処理で簡単に 2 行目は取り出せる．そのためタイトルに関するクラスは H だけになる．また 1 つのニュース記事の中がいくつかに分割され，各部分にサブタイトルがつくような場合がある．このためサブタイトルというクラス K も設定する．これも 1 行であると仮定する．また本文 (の一部) が 1 行であることはないとする．このため本文に関しては，本文の開始行の S，本文の中間行の M，本文の最終行の E の 3 つを設定する．

以上より，N,H,K,S,M,E の 6 つが本論文で設定したクラスとなる．

2.2.2 素性の設定

START/END 法では各事例が各クラスになる確率を求めなくてはならない．これは行を事例として，事例を上記 6 つのクラスのどれかに識別する分類問題として扱えばよい．分類問題として扱うには，注目する素性が必要となる．素性は，概略，各行が上記 6 つのクラスのどれになるかを判断する手がかりに対応するものである．

ここでは以下の 13 種類の素性を利用することにした．括弧ないの数値は素性のとり得る値である．

f1: センターに位置するか (1 or 0)
f2: 右づめになっているか (1 or 0)
f3: 。,」。,), [*] で終っているか (1 or 0)
f4: [??] が行の中にあるか (1 or 0)
f5: 左余白があるか (1 or 0)
f6: ○●■ ではじまってるか (1 or 0)
f7: 直前の行が水平区切りか (1 or 0)
f8: 直前の行に *.jpg があるか (1 or 0)
f9: 直前の行が FULL か (1 or 0)
f10: その行が FULL か (1 or 0)
f11: 直後の行が水平区切りか (1 or 0)
f12: 行の長さ (0 〜 80)
f13: 全角文字の個数 (0 〜 40)

素性を設定する際には，そのサイトに特有な手がかりを使うことを避けることを注記しておく．そのような素性を使えば，訓練データで扱ったサイト内のページに関しては，満足いく結果が得られるかも知れないが，他のサイトでは有効に機能しないからである．例えば，形態素解析してある単語が出現するかどうかの情報を利用した場合，あるサイトでは不要部分の文面が同一であるので，そのサイト内のページに関しては，その部分を排除する効果は高いが，他のサイトではその単語が出現するかどうかは識別の情報にならない．

2.2.3 決定木による学習

各行がどのクラスになるかを識別するための規則を学習するために，本論文では決定木 [7] を利用する[7]．

ただし決定木では識別の結果が決定的であり，各クラスの確率を与えることはできない．ここでは以下のような方法をとった．

訓練データから構築された決定木を利用して，訓練データの識別を行なう．訓練データの各事例は決定木のあるリーフノードに達することで，識別が行なわれるので，決定木の各リーフノードに達した事例を集め，そのクラスの割合をそのリーフノードに与えておく．実際の識別では，入力事例はあるリーフノードに達するので，そのリーフノードが持つ先のクラスの割合を出力とする．

2.3 状態遷移図の作成

START/END 法では出力となるクラス列を推定する際に，本質的には，クラスの状態遷移図を作り，それを HMM としてモデル化して最適なクラス列を求める．

固有表現抽出の場合，作成すべき状態遷移図は単純である．それはクラスの出現に制約がほとんどないからである．例えば,「人名の次に現れる固有表現は地名である」や「地名は 1 回しか現れない」や「人名と地名の間が 10 単語以上離れない」などといった制約はない．しかし本タスクではそのようなクラスの順序や位置関係に関して，いくつかの制約があり，状態遷移図を作成するには工夫が必要である．

まず単純に生成するクラス列を考えると，

```
N+ H N* S M* E N+
```

の列となる．これをそのまま状態遷移図で書くと図 5 のようになる．

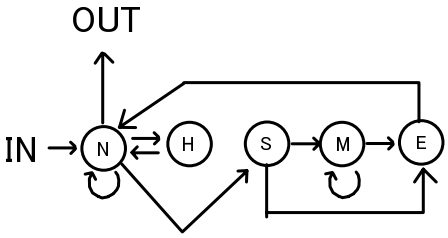


図 5: 単純な状態遷移図

これは H が 1 回しか現れないことや H が S の前に出現する制約が効いていない．なぜなら図 5 では N の後に H や S が出現可能であるために，H は複数回現れ得るし，H より前に S が出現可能でもある．

この問題は，H の前の N や E の後の N を別名にすれば解決できる．例えば，図 6 のようにすれば，H が 1 回しか現れないことや H が S の前に出現する制約を取り込めている．ここで J や LN は仮想的状態であり，本質的には N である．そのためその状態の持つ確率は N の状態が持つ確率と等しい．

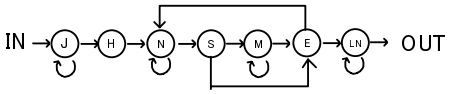


図 6: クラスの順序の制約を入れた状態遷移図

また通常， H と S が極端に離れることはないので，例えば，H と S の間には 5 行以上離れないという制約を入れる場合には，図 7 のような仮想的状態 (N1,N2,N3,N4,N5) を追加すればよい．これらの状態の持つ確率も N の状態が持つ確率と等しい．

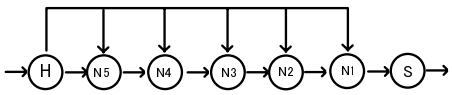


図 7: クラス間の距離の制約を入れた状態遷移図

上記のような工夫をいくつか追加して，最終的な状態遷移図を作成した．最終的に作られた状態遷移図を図 8 に示す．ここからの最適なクラス列の生成は動的計画法を用いて算出できる．

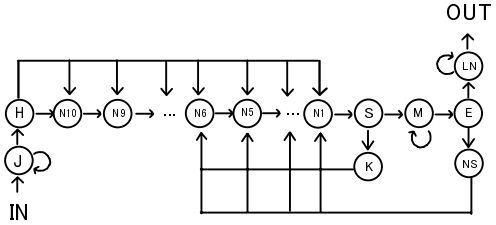


図 8: 作成した状態遷移図

[7] 他の学習手法も利用可能だが，決定木が使える場合には，決定木を使うのは無難な選択である．

3 実験

3.1 訓練データの作成

ここでは以下の 5 つのサイトからニュースページを取り出した．取り出したページ数を括弧で示す．合計 261 ページであった．

サイト 1 http://www.hotwired.co.jp (50)
サイト 2 http://pcweb.mycom.co.jp (50)
サイト 3 http://www.sponichi.co.jp (50)
サイト 4 http://japan.internet.com (61)
サイト 5 http://www.zakzak.co.jp (50)

これら 261 ページをテキストブラウザ lynx によりテキストファイルにし，設定したクラスを各行に与えることで訓練データを作成した．この訓練データを決定木構築ソフト C4.5 [7] に与えることで，決定木を構築した．

3.2 同サイト別ページによる評価

まず訓練データの作成元になったサイトから訓練データとは別に訓練データと同じ数だけニュースページを取り出し，それらに対しても lynx によりテキストファイル化した．

次に構築された決定木と設定した状態遷移図を利用することで，テキストファイルの各行にクラスを付与し，このクラスをもとに記事のタイトル (とサブタイトル) と本文を抽出した．

ここでは各行に与えたクラスの正解率，再現率，F 値をみることで評価を行なう．結果を表 1 に示す．

	抽出個数	正解個数	全体個数	正解率	再現率	F 値
H	262	196	262	0.748	0.748	0.748
K	35	11	35	0.314	0.314	0.314
S	380	281	320	0.740	0.878	0.803
M	7173	7109	7535	0.991	0.944	0.967
E	380	263	321	0.692	0.819	0.750

表 1: 同サイト別ページの評価

全体の評価は表 1 の通りであるが，正解率や再現率はサイトによって若干異なる．最も成績の良かったサイト 1 と最も成績の悪かったサイト 5 の結果を示す．他のサイトはサイト 1 とサイト 5 の中間位の値を出していた．

大雑把にまとめると，比較的単純な構造をしているページに関しては抽出精度が高い．ここで言う単純な構造とは本文中に図やサブタイトルが含まれないページである．サイト 1 のページがそのようなページであり，タイトルと本文が誰の目にもわかりやすい．一方，図や表が入った場合，図のキャプションが本文中の文章と区別し難しくなるし，本文が図や表によって分断されてしまうので，抽出が困難に

	抽出個数	正解個数	全体個数	正解率	再現率	F 値
H	50	49	50	0.980	0.980	0.980
K	0	0	0			
S	58	49	50	0.845	0.980	0.907
M	2769	2768	2815	1.000	0.983	0.991
E	58	48	50	0.828	0.960	0.889

表 2: 最良の結果 (サイト 1)

	抽出個数	正解個数	全体個数	正解率	再現率	F 値
H	51	36	51	0.706	0.706	0.706
K	9	0	0	0.000		
S	72	43	51	0.597	0.843	0.699
M	718	699	832	0.974	0.840	0.902
E	72	45	52	0.625	0.865	0.726

表 3: 最悪の結果 (サイト 5)

なる．サイト 5 のページはそのような場合が多く，抽出が困難であった．

3.3 別サイトによる評価

次に訓練データの元になったサイトとは別の以下の 6 つのサイトからニュースページを取り出した．括弧内の数値は取り出したページ数である．合計で 76 ページである．

サイト 6 http://www.asahi.com/ (14)
サイト 7 http://ascii24.com/ (10)
サイト 8 http://www.nikkansports.com/ (10)
サイト 9 http://www.nikkei.co.jp/ (13)
サイト 10 http://www.tvguide.or.jp/ (9)
サイト 11 http://www.yomiuri.co.jp/ (20)

これら 76 ページを lynx によりテキストファイル化した．以下，本手法を用いて，ニュース記事のタイトル (とサブタイトル) と本文を抽出した．結果を表 4 に示す．

	抽出個数	正解個数	全体個数	正解率	再現率	F 値
H	76	50	76	0.658	0.658	0.658
K	140	0	1	0.000	0.000	0.000
S	207	79	84	0.382	0.941	0.543
M	1155	1015	1088	0.879	0.933	0.905
E	207	51	84	0.246	0.607	0.351

表 4: 別サイトページの評価

F 値が同サイト別ページのものよりも悪くなっているが，最良の正解率を出したサイト 8 の成績は表 5 の通りである．これは同サイト別ページの最良のものと同等以上である．サイト 9 もほぼ同じ結果であった．これらのサイトのページのレイアウト

も比較的単純であったために，抽出がうまくいっている．

	抽出個数	正解個数	全体個数	正解率	再現率	F 値
H	10	10	10	1.000	1.000	1.000
K	1	0	0	0.000		
S	10	10	10	1.000	1.000	1.000
M	70	70	73	1.000	0.959	0.979
E	10	8	10	0.800	0.800	0.800

表 5: 別サイトでの最良結果 (サイト 8)

逆にサイト 6，サイト 7 は極端に精度が悪かった．これらのサイトでうまくいかなった原因については考察の章で述べる．

3.4 対訳コーパスの自動構築

本手法は様々な応用が可能である．ここでは新しい応用の 1 つとして，対訳コーパスの自動構築を提案する．

Web ニュースでは英語のニュースサイトの記事をそのまま翻訳して提供する日本語のニュースサイトがある．翻訳されたページにはオリジナルの英語のページへのリンクが張られているので，対訳関係にあるページが収集できる．それらのニュース記事からタイトルと本文を抽出することで，対約コーパスが作成できる．

本実験で利用したオリジナルとなる英語のサイトとその翻訳サイトの日本語のサイトを以下に示す．

英語のニュース記事のサイト:
Today on HotWired
http://hotwired.wired.com/

日本語のニュース記事のサイト:
HotWired Japan
http://www.hotwired.co.jp/

今回の実験では日本語のニュース記事のサイトの以下のディレクトリの下にあった 1,247 の記事を対象にした。

```
http://www.hotwired.co.jp/\
news/news/technology/story
```

この実験に関しては各々のサイトに特有の素性を導入して，それぞれのサイト毎に決定木と状態遷移図を作成した．

上記の記事の中でオリジナルの記事へのリンクがあったものは 1,139 記事であった．それらの記事をすべてテキストブラウザを用いてテキストファイルにした後，日本語のページと英語のページから，それぞれ 50 ページをランダムにとりだし，各行にクラスの正解タグを付与した．これを訓練データとした．

ここから決定木を作成し，残りページについて各行にクラスの確率を付与し，作成した状態遷移図を利用して，記事のタイトルと本文の抽出を行なった．結果として 1,089 記事の英日対訳テキストを得ることができた。得られたコーパスうち日本語の方を調べると，全部で 37,600 文，1 文平均 58.9 文字であった．

抽出結果の評価を行なうために，日本語，英語，各 20 ページをランダムに選び，正しく記事部分を取り出せているかどうかを調べた．結果，すべて正しく取り出せていた．

4 考察

本手法により構築した抽出規則は，ある程度単純なページに対しては精度が高かったが，図やサブタイトルが含まれるような複雑なページに対しては，精度はそれほど高くはなかった．しかしこれは本手法のアプローチに無理があることを示しているわけではなく，利用した素性の不備であると考えている．

記事本文が分断されたテキストファイルを読んでも，人間は記事部分を把握できる．つまり抽出規則は存在する．利用する手がかりの情報をうまく組み合わせてゆけば，その規則を近似することは可能なはずである．ただし，人間が複雑なレイアウトからもタイトルや記事を特定できるのは，内容を少しは読んでいる，少なくとも，文字列が意味のある文字列であるかどうかくらいは判断しているからだと思われる．つまり抽出規則の手がかりには，本質的に文字列に対する自然言語の情報を利用する必要がある．しかしここで利用した素性には，自然言語の情報を使っていない．テキストファイルに変換することは自然言語処理を利用できる状態にすることも狙っていたのだが，どのような素性を含めればよいかが検討段階であったため，実装することはできなかった．今後はこの点の検討を深め，自然言語の情報を組み込んでいく必要がある．

また別サイトのページに関しては同サイトのページよりも精度が低かった．レイアウトの問題があることは，同サイトのページと同じであったが，それ以外に，H の判定で大きく間違ったことが原因であった．ここで利用した状態遷移図では H の後に S,M,E が必ず現れる．また H と S の間隔は 10 行以内という制約を入れている．そのためテキストファイルの最初の方である行を H と判定してしまい，実際の H がその行より 10 行以上，下方に位置する場合には，無用の S, M, E の抽出も行なう．更に本来の H にはほとんどの場合 K (サブタイトル) のクラスを与えるために，その部分でも精度が下がる．本手法の状態遷移図を利用する場合には H の判定の精度を高める必要がある．

ただし H の判定を正確に行なうには，その行の文字列だけの情報では限界がある．実際に別のページではタイトルとなっている文字列が，他のページ

では無用な行となっていることもある[8]．1行だけをみた判断では，自然言語処理を導入しても，この問題の解決は難しい．この問題には，2つの回避策が考えられる．1つは学習で前後の行の情報を素性として組み込むことである．もう1つは動的計画法での最適なクラス列を生成する部分で工夫することである．前者の回避策は場当たり的であるし，人間はテキスト全体から見た場合の，タイトルや本文の位置を考慮していて判定していると思われるので，後者のアプローチを検討したい．

本手法の抽出精度は高くはないが，それは従来のWrapper 学習に対して劣っていることを意味しない．本手法もサイト毎に学習を行なえば，抽出規則の精度は向上する．またサイト毎の特有な素性を導入すれば更に精度は向上する．実際に対訳コーパスを作成した実験では，そのサイト特有の素性を利用することでほぼ 100% の抽出精度を達成している．今回の実験で精度の低かったサイト 5 に関しても，サイト特有の素性を導入すれば，精度の高い抽出規則の学習は容易である．

本手法は様々なサイトで適用可能な抽出規則を学習することを目標としたが，その背景には，サイト毎の学習を行なう際に訓練データを構築することが多大な負荷であることがあった．この問題に対しては，本手法のアプローチとは別に，Wrapper 学習に教師なし学習を利用するという研究もある [1][2]．本手法の核は単純な分類問題であるので，教師なし学習が利用可能だと思われる．その点も検討したい．

最後に本手法は様々な応用が可能であることを述べておく．本タスクのように Web ニュースの中からニュースコンテンツを取り出す処理自体は，Web ページの読み上げに利用できる．サーチエンジンはページの収集の際に，本文とは無関係の部分を省くことで検索精度を高められる．また論文のトップページから書誌情報を抽出する処理にも応用できる [5]．

5 おわりに

本論文では Web ページから目的部分のテキストを自動抽出する手法を提案した．本論文で扱ったタスクは，Web ニュースのページからそのニュース記事のタイトルと本文を抽出するタスクである．

提案手法では Web ページをテキストブラウザを利用してテキスト化する．このテキストファイルをもとに抽出規則の学習を行なう．具体的には行を事例とした START/END 法と制約を取り入れた状態遷移図を利用する．本手法は Wrapper 学習の一種であるが，通常の Wrapper 学習とは異なり，HTML のタグを抽出手がかりとして使わない．そのためにサイトが異なるページに対しても適用できる汎用的な抽出規則を学習できる可能性がある．

実験では訓練データの元になったサイトから取り出したページと別サイトから取り出したページを使って抽出実験を行なった．単純なレイアウトのページであれば，高精度に抽出できていたが，複雑なレイアウトのページでは抽出に失敗していた．

また本手法は非常に応用が広い．ここでは新しい応用として対訳コーパスの自動構築を提案した．

今後は自然言語の情報を素性に組み入れることを検討する．また本タスクに関してはタイトルの判定の精度を高めて改善を図る．

[8] 他のニュース記事へのリンクなどを考えれば明らか．

参考文献

[1] Chia-Hui Chang and Shao-Chen Lui. Iepad: Information extraction based on pattern discovery. In *10th international conference on World Wide Web*, pp. 681–688, 2001.

[2] D.W. Embley, Y. Jiang, and Y-K Ng. Record-boundary discovery in web documents. In *ACM SIGMOD international conference on Management of data*, pp. 467–478, 1999.

[3] Atsushi Fujii and Tetsuya Ishikawa. Organizing encyclopedic knowledge based on the web and its application to question answering. In *Porc. of the 39th Annual Meeting of the Association for Computational Linguistics*, pp. 196–203, 2001.

[4] Nicholas Kushmerick. Wrapper induction: Efficiency and expressiveness. *Artificial Intelligence*, Vol. 118, No. 1-2, pp. 15–68, 2000.

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

[6] Ion Muslea, Steve Minton, and Craig Knoblock. Hierarchical wrapper induction for semistructured information sources. *Autonomous Agents and Multi-Agent Systems*, No. 4, pp. 93–114, 2001.

[7] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publisher, 1993.

[8] Dragomir R. Radev, Sasha Blair-Goldensohn, Zhu Zhang, and Revathi Sundara Raghavan. Interactive, domain-independent identification and summarization of topically related news articles. In *5th European Conference on Research and Advanced Technology for Digital Libraries*, 2001.

[9] Satoshi Sekine, Ralph Grishman, and Hiroyuki Shinnou. A Decision Tree Method for Finding and Classifying Names in Japanese Texts. In *Proceedings of the 6th Workshop on Very Large Corpora*, 1998.

[10] Stephen Soderland. Learning information extraction rules for semi-structured and free text. *Machine Learning*, Vol. 34, pp. 233–272, 1999.