平成23年度茨城大学工学部情報工学科卒業
研究論文

人名構成文字確率を用いた文字ベースCRFによる中国語人名検出

茨城大学工学部情報工学科
執筆者：全　太俊
指導教官：新納　浩幸

目 次

第1章　はじめに

1.1　概要

情報化が浸透した現在，個人が入手できる情報の量は，人間の処理能力の限界を超えている. 情報の形態としては，テキスト，画像，音声など様々のものが存在するが，現在において伝達の主情報として用いられているのはテキストである.WWW(World Wide Web) 上に存在するテキストや電子化された新聞記事は膨大な量となるため，人間が全てのテキストを読むことによって必要な情報を探し出すことは事実上不可能である. そこで，大量の文書を構造化・組織化する技術が研究されている. それらの技術を用いることによって，人間が直接文書を読む前に，必要な文書を絞り込むことが可能になると考えられる．　文書の構造化手法として近年では，情報抽出[1]（Information Extraction）が注目を集めている. 情報抽出とは，文書から中心的な情報だけを抽出する技術である. その中で，基礎技術となるのが固有表現抽出[2]（NamedEntity Extraction）である．　固有表現とは，人名，地名，組織名などの固有名詞や，時間表現，日付表現などを指す. 非常に多くの数の固有表現が存在するため，形態素解析の段階で用いる辞書に登録されていない表現が多く出現し，解析時に未知語になりやすいという問題がある. そのため，解析誤りを避けるためには，多くの固有表現を辞書に登録しておくことが有効である. 人名検出は固有表現抽出の一部であり, 情報抽出の要素技術として重要である [1]. 本論文では, 中国語の人名検出を行う.

一般に, 固有表現抽出は系列ラベリング問題として定式化し, 条件付き確率場（CRF: Conditional Random Field）[2, 3, 4] を用いることで精度よく行える. ただし, 通常,CRF の素性として単語の品詞を利用するが [5], 中国語の場合, 標準的に利用できる形態素解析システムが存在しないために, 単語分割, 品詞付与の処理が容易ではない. そこでここでは文字列ベースの CRF [6] を用いる. また中国語のコーパスと中国人名のデータベースを利用して, 各文字が人名の構成要素となる確率を推定し, その確率がある閾値よりも高いか低いかのラベルを素性として加えることで, 単純な文字列ベースの CRF を改善する.

[1] http://ja.wikipedia.org/wiki/情報抽出に乗っている.
[2] http://ja.wikipedia.org/wiki/固有表現に乗っている

1.2　本論文の構成

本論文の構成は以下の通りである.
２章では系列ラベリング問題の固有表現, 情報抽出などについて述べる.
３章では系列ラベリングモデルについて紹介する.
４章では文字ベースを使った CRF について述べるが、道具である CRF++ の使い方も紹介する.
５章では中国人名のデータベースを利用して, 各文字が人名となる確率を推定する.
６章では文字ベースの CRF と人名構成確率を用いて文字ベースの CRF 実験を行う.
７章では実験結果を元に間違えて抽出したもの, 抽出できなかったものについて考察を行う.
８章では結論として今後の課題について述べる.

第2章　固有表現抽出

2.1　情報抽出

2.1.1　情報抽出の定義

本研究でいう情報抽出は, その分野で中心的な存在である米国の Message Understanding Conference （MUC）に準ずる．MUC における情報抽出とは，新聞記事のようなテキストからあらかじめ指定されたイベントや事柄に関する情報を抽出し，その情報をデータベースに入力する，という技術である．まずは具体例を見てみよう．表 2.1 に人事異動に関する新聞記事を基にした情報抽出結果を載せる．ここでは，抽出したい情報は，企業の重役の異動（昇進，降格，退任等）に関する情報であり，抽出したい情報の内容としては，該当者の人名，会社，異動前役職名，異動後役職名，異動理由，異動発生日というように与えられている．ご覧のように，面倒な文章で描かれた人事異動の情報がすっきりとしたデータベース形式で抽出されている．このような技術は，特定の情報を簡単に調べたいときなどに非常に役に立つ．たとえば，過去 10 年の新聞記事から，企業の重役の異動に関する情報を得たいという場面を想定してみよう．現在よくある情報検索の技術を利用すると，適当な検索式を作成し，異動に関連した記事を引っぱり出すことはできる．しかし，多くの読者が簡単に想像できるように，その結果は膨大であり，実際に必要な情報は 1 つ 1 つの記事を読んでみなければ得ることができない．また，自動要約という技術を利用して，その手間を減らすことは可能であるが，それでも文章を読まなければいけないことには変わりない．また，今の技術では，特定の視点に注目した要約はまだ開発途上であると言わざるを得ない．これに対し情報抽出の技術を利用した場合には，表 2.2 のような表形式で，複数の異動の情報を視覚的に一度に見ることができる．表形式になっているので，特に注目したい欄を対象にフィルタリングすることなども可能である．つまり，情報検索結果の記事を 1 記事あたり数分かけて読みながら調べるというのに比べて，非常に高速に必要な情報を得ることができる．ここでは一例として，重役の人事異動をとりあげたが，対象となる情報は，もちろん，それだけに限らない．一般的に前もって,「抽出したい情報の型」が決められるものなら特にこだわらない．たとえば，合弁事業の情報，新製品の情報，新事業の情報など産業界に役に立ちそうな内容や，研究者向けには科学技術論文における技術内容の情報抽出や，医療カルテ，ゲノムといった特定分野の文

章における注目される情報（たとえばタンパク質の役割の情報抽出），またアイドル歌手の活動情報抽出，スポーツなどの特定イベントの情報など個人的な利用も含め広く考えることができる．この技術のポイントは，一般的にテキストを理解するという技術に比べて，あらかじめ抽出したい情報の型が与えられているということにより，比較的に実現容易だという点にある．すでに，いくつかのデモシステムは作成されており，実際に商用に使われ始めているシステムもある．

表 2.1: 情報抽出の例

〈新聞記事〉 ABC 株式会社は十二日，臨時取締役会で田中一郎社長が代表権のある会長に就任し，山田次郎副社長が社長に昇格する人事を内定したと発表した．鈴木三郎会長は代表取締役にとどまる．三月二十五日に開く株主総会後の取締役会で正式決定する．田中社長は五期十年社長を務め，年齢も七十一歳と高齢になったため，若返りをはかる． 〈異動イベント情報〉 人名： 田中一郎 会社名： ABC 株式会社 異動前役職名： 社長 移動後役職名： 会長 異動理由： 昇格 異動発生日： 3 月 25 日 人名： 山田次郎 会社名： ABC 株式会社 異動前役職名： 副社長 移動後役職名： 社長 異動理由： 昇格 異動発生日： 3 月 25 日 人名： 鈴木三郎 会社名： ABC 株式会社 異動前役職名： 会長 移動後役職名： 代表取締役 異動理由： 降格 異動発生日： 3 月 25 日

表 2.2: 情報抽出の結果

人名	会社名	異動	異動理由	発生日
田中一郎	ABC 株式会社	社長から会長	昇進	1999 年 3 月 25 日
山田次郎	ABC 株式会社	副社から長社長	昇進	1999 年 3 月 25 日
鈴木三郎	ABC 株式会社	会長から代表取締役	降格	1999 年 3 月 25 日

表 2.3: パターンの例

［固有名詞 — カタカナ列 — アルファベット列］ ”株式会社” -¿ @ 企業数字 ”月”数字”日” → @日時 ［”社長”|”副社長”|”会長”|...］ → @役職@日時”,”@企業名”の” @ 人名@役職”は”@役職”に””昇進した.” → @昇進イベント

2.1.2 利用されている技術

先ほどこの技術がテキストを理解するよりも容易であると述べたが，実際にどのような点で容易なのか，また，逆にシステムを開発したり使用する際に制約が生じていないかというようなことについて述べる．

•パターンマッチングによる情報抽出

一般にテキスト解析というと，自然言語処理における構文解析や意味解析を行うと考えられがちであるが，これらの技術はまだ困難な問題が残り，なんら制約のない環境で広く利用可能であるとは言い難い．それに対して，情報抽出の実用が広く可能になっているのは，これらの難しい技術を使用して深い理解を試みることなく情報抽出を実現できる技術が生まれたという点に集約できる．それがパターンマッチングによる情報抽出という技術である．ちなみに，初期の MUC では構文解析などの技術を用いる方式が主流であったが，パターンマッチングの方式が性能的に優れていたため淘汰されてしまった．現在は一部で統計的構文解析等を利用した新しい試みはあるが，広くは利用されていない．
パターンマッチングは，その情報抽出の対象に関係する文や文の一部のマッチするパターンを用意しておいて，それを決まった順に適応し，決定的に情報を把んでいくという技術である．たとえば，表 2.3 の最初のパターンを言葉に直すと,「固有名詞かカタカナ列かアルファベット列の後に，株式会社と

いう文字列があった場合にはそれらをまとめて企業と判断する.」ということである．@で始まる要素は，なんらかのカテゴリを示す．ここでは，プロセスや定義の詳細には立ち入らないが，表 2.3 の後半では，入力文が次第にパターンマッチングにより@の付いたカテゴリに変換され，最終的に昇進イベントを得るという様子が描かれている．まず,「XYZ 株式会社」が最初のパターンにより「@企業」に変換され，その他，日時，人名，役職等が随時変換されていく．そして，最後のパターンを利用して，昇進イベントを得るという流れになる．最後の昇進イベントを得る際には，それぞれの項の内容を記憶し，それぞれの役割りの抽出も同時に行うということがされる．これによって図-1 に示したような情報が抽出されることになる．ここでは，説明のために簡略化してパターンマッチングの流れを示したが，実際にはこれだけでは十分ではない．人名等の抽出はなかなかパターンのみでは困難であり，辞書的な情報との組合せが必要になってくる．また，パターンは注意深く設計しないと，意図しない文字列とマッチングしてしまうことも考えられる．そして，正確に情報を把握するためにはパターンマッチングの順序も重要である．また，現実的にはこのような 1 文に対するパターンマッチングだけですべての情報を得ら

れるとは限らない．1 つのイベントが複数の文にまたがって書かれていることもある．このような問題を解決するためには照応解析の技術が必要になってくる．これはたとえば，表 2.1 の例で，2 つ目の文にある鈴木三郎が ABC 株式会社の会長であったということを理解するためには，その前の文の情報が必要であるといったことである．このような照応を厳密に行うには，難しい問題があるが，ある程度までの照応解析は，シンプルなルールで解決することができる．情報抽出は，これまで述べたパターンマッチングの技術を利用して，実用化に向って大きく成長してきた．しかし，このパターンマッチングの枠組みから，システム開発や利用においてある種の制約を生じさせてしまっている．次に，その制約を技術的な課題として紹介し，その解決に向けての取り組みを紹介する．

表 2.4: パターンマッチングの例

入力文： 2 月 7 日，XYZ 株式会社の高橋四郎副社長は社長に昇進した.
↓
2 月 7 日,（@企業）の高橋四郎副社長は社長に昇進した.
↓
（@日時）,（@企業）の（@人名）（@役職）は（@役職）に昇進した.
↓
（@昇進イベント）

2.1.3 技術的な課題

情報抽出においてパターンマッチング技術を利用したことによって生じた最大の問題は，パターンを情報抽出の課題ごとに作成しなければいけないという点である．つまり，人事異動のパターンは人事異動にしか使用できず，新たに企業合併に関する情報抽出を行いたいとしたときにはそのためのパターンを新たに作らなければいけないということである．課題によって必要なパターンの数は異なるが，ある程度複雑な課題の場合には，少なくとも 500 から 1000 近いパターンが必要になってくる．これらのパターンを課題が与えられるごとに，1 から手作業で作成していくのでは，そのコストが膨大になり，あまり実用的であるとは言い難い．システムがある特定の情報抽出の要望だけを処理するのならばいいが，望むべくは幅広い課題に対処できるようにしたい．この問題に対しては大きく 3 つの方向で研究が進められている．
1 つは，課題の種類に依存しないパターンと，課題に依存したパターンを切り分け，前者をライブラリとして用意しておくという方法である．たとえば，表 2.4 の例では重役の昇進という課題以外でも十分に有効であり，課題に依存しないパターンとして利用できる．また，課題に依存するパターンも課題の種類によっては流用できるものもあると思われる．そのような分野依存性を的確に判断しパターン作成の労力を少しでも減らそうというのが 1 つ目の方針である．
2 つ目は，パターンを作成するためのツールを用意し，言語処理や計算機システムなどに詳しくない人でも簡単にパターンを作成できるような環境を用意するという方法である．たとえば，ニューヨーク大学では例文を基に，簡単にパターンを作成するツールを開発している．パターンの要素の一般化や構文的なバリエーションの自動的な作成などをサポートし短時間で特定の課題に対するパターンを作成できるようになっている．
最後には，パターンを自動的に作成するという方法である．基本的には大量の文章を基に，動詞の使われ方や固有名詞の出現などの情報からパターンを自動的に作成するという取り組みが行われている．この分野はまだ研究段階であり，今後が楽みな分野である．

2.1.4 要素技術の分割

最初に紹介した MUC では，システムのモジュラリティやポータビリティを向上させることを目的に，情報抽出における要素技術を分割し，それぞれの技術をそれのみで評価しようという試みが行われた 3)．この分割は，単に情報抽出の開発に役に立っているだけではなく，それぞれの要素技術の問題点を整理したり，また単独で応用システムに組み込まれたりするのに役に立っている．
まずは，表 2.4 の例からも明らかであるが，人名，組織名，時間などの表現の

抽出は 1 つの要素技術として分割できる．この課題は，テキストの中にある固有表現を抽出し，種類を認定することが課題とされており，すでに情報検索の分野で応用されている．たとえば，地名とも人名ともとれる表現（たとえば，成田やワシントン）を地名として検索したいか，人名として検索したいかをあらかじめ指定しておくことによって，検索ミスを減らそうという応用である．また，組織名や人名には課題によらない決まった型があると考えられる．たとえば，組織名なら，所在地，社長名，事業分野などがある．そのような情報を分野によらず決めておいて，その情報を抽出することを目的とした課題というのもある．この技術はテンプレート抽出技術と呼ばれている．最後に，パターンマッチングのところで触れたが，照応解析というのは情報抽出によって重要な技術である．これも独立した課題として扱うことができるし，この技術は他の自然言語処理応用システムに利用することができる．

2.1.5 応用分野

情報抽出は技術的には成熟期に手が届いているという状態であり，米国はもとより，日本でも製品化やデモシステムの実例がある．ここでは，それらを大雑把にまとめ，応用分野として紹介したい．

●高度情報検索としての応用
本稿の最初にも述べたように，現在の情報検索が持つ問題点を解決していこうという目的が，今一番ホットな対象である．インターネットを始めとする電子上のテキストが膨大になるにつれ，非常に有望な応用課題となってきている．新聞記事や特定の情報リソースからの情報抽出はもとより，電子化図書館での応用なども考えられる．

●データベースの（半）自動構築
世界のデータベースコンテンツのシェアは米国が断突であり，日本でもその作成が急務であると伝えられている．このようなデータベースの作成は非常に手間のかかる仕事であるが，それを推進するツールとしての利用価値が考えられる．

2.2 固有表現

固有表現とは, 人名や地名などといった固有名詞や, 日付表現, 時間表現などに関する総称である. 以下は固有表現について解説する.

1. 組織名 $\langle ORGANIZATION \rangle$
複数の人物で構成され，共通の目的を持った組織等の名称のこと.
例 :「共和党」「茨城大学」

2. 人名 $\langle PERSON \rangle$
固有の人物を表す名称.
例 :「中村俊輔」

3. 地名 $\langle LOCATION \rangle$
固有の場所を表す名称．国名，都道府県名，市町村名などの行政単位となりうるものから，河川名，山脈名など地形に関するものを含む.
例 :「東京都千代田区」「富士山」

4. 固有物名 $\langle ARTIFACT \rangle$
人間の活動によって作られた具体物，抽象物を含む物の固有の名前．例 :「ペンティアムプロセッサ」,「日米安保条約」

5. 日付表現 $\langle DATE \rangle$
特定の時間を表現するもので，単位が 24 時間以上のもの.
例 :「２００１年９月１１日」「江戸時代」

6. 時間表現 $\langle TIME \rangle$
特定の時間を表現するもので，単位が 24 時間以下のもの.
例 :「午前７時」「正午」

7. 金額表現 $\langle MONEY \rangle$
金額を表す表現.
例 :「１０９円５７銭」「１００億ドル」

8. 割合表現 $\langle PERCENT \rangle$
割合を表す表現.
例 :「５０％」「３割２分５厘」

2.3 系列ラベリング問題として固有表現抽出

m 個のデータの系列 $\{x_1, x_2, \cdots, x_m\}$ が与えられたときに, 系列中の各データ x_i に対する適切なラベル $y_i \in C$[3]を求めて, ラベルの系列 $\{y_1, y_2, \cdots, y_m\}$ を出力する問題を系列ラベリング問題という.

人名抽出の問題は, 系列ラベリング問題として取り扱うことが可能である[4]. 例えば, 以下の単語切りされた文は, 各単語がデータとなるデータの系列である.

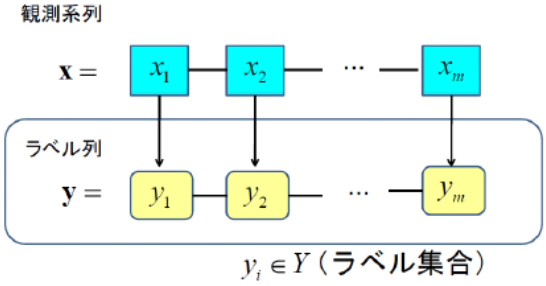


図 2.1: 系列ラベリング問題

私 / の / 名前 / は / 全 / 太俊 / と / 申し / ます

系列中の各単語に対して BI , BO , O のいずれかラベルを与える. ラベル BI は人名の開始単語であることを意味し, ラベル BO は人名内の単語であることを意味し, ラベル O は人名ではない単語を意味する. この例の場合, 以下のようなラベルが与えられ, ラベルの系列が完成する.

```
私 / の / 名前 / は / 全 / 太俊 / と / 申し / ます
O    O     O      O   BI    BO    O    O      O
```

このラベルの系列から「全太俊」が人名として抽出できる.

[3] C はラベルの集合
[4] 一般にチャンキング抽出の問題が扱え, 固有表現抽出はチャンキング抽出の一種である.

2.4　パターン照合に基づく固有表現抽出

パターン照合に基づく固有表現抽出とは，人手で明示的なパターンを作成し，それらを適用することによって固有表現を抽出するものである [7]．具体的には,「〜さん」,「〜大学」「株式会社〜」など，固有表現に含まれる接尾辞・接頭辞などの表記パターンを利用してテキスト中の固有表現を探索する．パターン照合に基づく手法の利点は,（1）人手で作成された規則を用いるため，実際にどのような規則が適用されているかが容易に観察することができる.（2）限られた分野に対しては高精度であることが期待できることが挙げられる．一方欠点としては，規則の変更や追加に対して多大なコストを要することが挙げられる．

2.5　教師あり学習による固有表現抽出

一般に固有表現抽出は教師あり学習によって実装される. この場合, 教師データとして人手でタグ付けされたデータを与え, それを分類器が学習することで, 入力されたデータに対してタグが付与される．機械学習を用いない最も単純な抽出方法として, 予め作成した各固有表現クラスの辞書による単純マッチでも固有表現を抽出することは可能であるが, 機械学習を用いた手法には以下のような利点がある．多義語の処理人名・地名のいずれとしても出現しうる“ Washington ”のように, 複数のクラスをとり得る語に関しては, 辞書でマッチングを図るだけではいずれのクラスに分類すべきか判断がつかない. 人間にとって判断するにたる情報が含まれる文においては, 何らかの形で文脈を考慮することで適当なクラスを判断することができると考えられるが, 当該語句のあらゆる出現に関して妥当な判定ルールを陽に作成することは一般に困難である. 一方機械学習の枠組みでは, 前後の単語を素性に含めることで文脈を判断材料に含めることができると考えられる．未知語の処理辞書によるマッチングでは, 各クラスに属する語のうち辞書に含まれていない物を正しく判定することができない. 機械学習では, 素性設計次第で, 人間が未知の語に対してある程度どのような物を指すのか推定できるのと同様に, 教師データ中の他の語の出現例から得た知識で新たな語を分類できる可能性がある．

第3章　系列ラベリングモデル

3.1　最大エントロピー (ME)

最大エントロピー法 [7] は，訓練データ $C(t,h)$ から，確率モデル P(t—h) を推定するアルゴリズムである．ここで，h は条件付き確率の条件となるべき事象 (以下，履歴事象 (history) と呼ぶ)，t は確率モデルが予測するべき事象 (以下，目標事象 (target) と呼ぶ) であり，$C(t,h)$ は t, h が同時に起こる頻度である．単語 bigram においては，t, h はともに単語であり，h が存在したときにそれが t と共起する確率 $P(t|h)$ を推定することになる．確率モデル $P(t|h)$ の値は式 (3.1) で計算される．

$$P(t|h) = \frac{\prod_{i=1}^{|F} \alpha_i^{f_i(t,h)}}{\sum_t \prod_{i=1}^{|F|} \alpha_i^{f_i(t,h)}} \tag{3.1}$$

fi(t, h) は素性 (feature) と呼ばれ，目標事象，履歴事象の組に対して 1 または 0 を返す任意の関数である．また，F はこのような素性の集合である．本稿では素性をの (3.2) ように定義する．

$$f(t,h) = \begin{cases} 1 & t \to C_t h \to C_h \\ 0 & \text{そのほかの場合} \end{cases} \tag{3.2}$$

Ct, Ch は意味クラスであり，$t \to Ct$ は単語 t が意味クラス Ct に属することを意味する．例えば，Ct が「生物」という概念を表わす意味クラスで，Ch が「食べる」という概念を表わす意味クラスであるとき，この素性は「生物」を表わす単語と「食べる」という意味を表わす単語が共起しやすいということを表わしている．以下，意味クラス Ct, Ch によって定義される素性を f(Ct,Ch) と記す．このとき，Ct, Ch として任意の抽象化レベルの意味クラス (もしくは単語) を用いることができ，またグラフ構造を持つシソーラスの意味クラスも利用可能である．また，式 (3) 中の α i は素性 fi のパラメタである．最大エントロピー法による確率モデル P(t—h) の推定は，これらのパラメタ α i を推定することに他ならない．次に，素性の期待値 $E(fi)$, $\hat{E}(f_i)$ を式 (3.3), (3.4) のように定義する．

$$\begin{aligned} E(f_i) &= \sum_{t,h} P(t,h) f_i(t,h) \\ &= \sum_{t,h} \hat{P}(h) P(t|h) f_i(t,h) \end{aligned} \tag{3.3}$$

$$\hat{E}(f_i) = \hat{P}(t,h) f_i(t,h) \tag{3.4}$$

ただし,

$$\hat{P}(t,h) == \frac{C(t,h)}{\sum_{t,h} C(t,h)} \tag{3.5}$$

$$\hat{P}(h) = \sum_{t} \hat{P}(t,h) \tag{3.6}$$

すなわち，$E(f_i)$ とは，fi が 1 を返す事象の確率モデルにおける確率 P(t, h) の総和である．最大エントロピー法による確率モデルの推定は，1. 素性に関する (3.7) の制約を満たしつつ,

$$P(f_i) = \hat{P}(f_i) \forall (f_i) \in F \tag{3.7}$$

2. $P(t|h)$ のエントロピー $H(P)$ が最大となる

$$H(P) = -\sum_{t,h} \hat{P}(h) P(t|h) \log P(t|h) \tag{3.8}$$

ようにパラメタを推定することにより行われる．すなわち，最大エントロピー法による確率モデルの推定とは，ある素性が 1 を返す事象 (t,h) についてはその確率の和を訓練データのそれに近付け，かつ確率分布が一様分布になるべく近くなるように (意味クラスが支配する単語の確率が全て等しくなるように) 確率モデルを推定することである．

3.2 生成モデル:隠れマルコフモデル (HMM)

3.2.1 隠れマルコフモデル

隠れマルコフモデル (Hidden Markov Model，HMM) [8, 9] とは，確率的な状態遷移と記号出力を備えたオートマトンである．次の 5 項組 $M = (X, Y, A, B, \pi)$ で定義される．

1. 出力記号系列 $X = x_1, ..., x_n$ であり，観測可能である．
2. 状態遷移系列 $Y = y_1, ..., y_n$ であり，観測不可能である．
3. 状態遷移確率分布 $A = a_{ij}$ であり，状態 y_i から状態 y_j への遷移確率である．単純マルコフを仮定している．
4. 記号出力確率分布 B $= b_i(x_t)$ であり，状態 y_i で記号 x_t を出力する確率である．

出力は現在の状態にのみ依存すると仮定している．

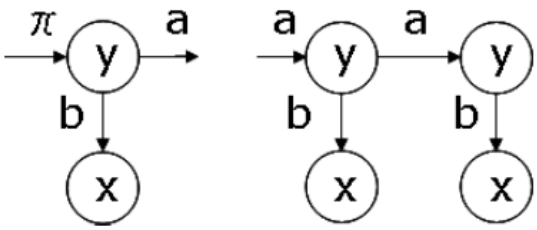


図 3.1: 隠れマルコフモデル

5. 初期状態確率分布π = π_i であり，状態 yi が初期状態である確率である．すなわち，図 3.1 のようなモデルが構成される．

本稿では，あらかじめ正解が与えられた学習データからモデルを生成し，これを基に，最適な状態遷移系列の推定を行う教師あり機械学習の手法を用いることとする．

また，その推定には，Viterbi アルゴリズムを用いる.Viterbi アルゴリズムは以下のようなステップを持つ．

1. 各状態 i=1,⋯,N に対して，変数の初期化を行う．

$$\delta_1(i) = \pi_i b_i(o_1)) \tag{3.9}$$

$$\psi_1(i) = 0$$

2. 各状態の遷移ステップ t=1,⋯,T-1, 各状態 j=1,⋯,N について，再帰的に計算を行う．

$$\delta_{t+1}(j) = max_i[\delta_t(i)a_ij](a_t + 1) \tag{3.10}$$

$$\psi_{t+1}(j) = argmax_i[\delta_t(i)a_ij] \tag{3.11}$$

3. 再起計算の終了

$$\hat{P} = max\delta T(i)] \tag{3.12}$$

$$\hat{q}T = argmax\delta T(i) \tag{3.13}$$

4. バックトラックによる最適状態遷移系列を復元する.$t = T - 1, ..., 1$ に対して行う．

$$\hat{q}t = \psi_{t+1}(\hat{q}_{t+1}) \tag{3.14}$$

Viterbi アルゴリズムにより，確率値を最大にする状態遷移系列を得ることができる．

3.2.2 HMM とタグ付けの対応

英語の文章は，観測できない状態遷移系列として，品詞を持っている．例えば，The man saw a girl という文章の場合，その観測できない (隠れ) 状態列として，図 3.2 のように，品詞を持っていると考えられる．したがって，本稿で扱う品詞推定は，隠れマルコフモデルにおいて，状態遷移系列を品詞，出力記号系列を単語とし，学習データから，(A,B, π) を以下で示した式のように学習することにより，モデルを生成する．ただし，P(y—x) は条件付き確率であり，条件 x の下で y が生起する確率を示したものである．また C(x) は生起頻度であり，語 x が当該文書で出現する頻度を表す．BOS(Beginning Of Stream) は初期状態であり，文の先頭であることを表す．

$$\pi_i = P(y_i|BOS) = \frac{C(y_i)}{C(BOS)} \tag{3.15}$$

$$a_{ij} = P(y_j|y_i) = \frac{C(y_i, y_j)}{C(y_i)} \tag{3.16}$$

$$b_i(x_t) = P(x_t|y_i) = \frac{C(y_i, x_t)}{C(y_i)} \tag{3.17}$$

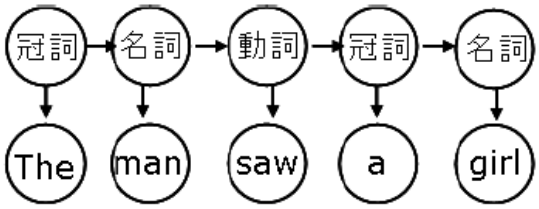


図 3.2: HMM の対応図

3.3 識別モデル:条件付き確率場 (CRF)

CRF は対数線形モデルの一種であり, データの系列 $\boldsymbol{x}$ に対応するラベルの系列が $\boldsymbol{y}$ となる確率 $P(\boldsymbol{y}|\boldsymbol{x})$ を以下の式でモデル化する.

$$P(\boldsymbol{y}|\boldsymbol{x}) = \frac{1}{Z}\exp(\boldsymbol{w}\cdot\phi(\boldsymbol{x},\boldsymbol{y})) \tag{3.18}$$

ここで$\boldsymbol{w}$は素性に対する重みベクトル,ϕ は素性ベクトルに変換する関数, Z は正規化のための以下で定義される数である.

$$Z = \sum_{\boldsymbol{y}}\exp(\boldsymbol{w}\cdot\phi(\boldsymbol{x},\boldsymbol{y})) \tag{3.19}$$

訓練データ $\boldsymbol{D}$ は以下の形をしている.

$$\boldsymbol{D} = \{(\boldsymbol{x^{(1)}},\boldsymbol{y^{(1)}}),(\boldsymbol{x^{(2)}},\boldsymbol{y^{(2)}}),\cdots,(\boldsymbol{x^{(|D|)}},\boldsymbol{y^{(|D|)}})$$

$\boldsymbol{x^{(i)}}$ はデータの系列,$\boldsymbol{y^{(i)}}$ は $\boldsymbol{x^{(i)}}$ に対応するラベルの系列である. CRF はこの $\boldsymbol{D}$ を利用して $\boldsymbol{w}$を学習する実際の識別は以下で行える.

$$\boldsymbol{y^*} = \arg\max_{\boldsymbol{y}} P(\boldsymbol{y}|\boldsymbol{x}) = \arg\max_{\boldsymbol{y}}\exp(\boldsymbol{w}\cdot\phi(\boldsymbol{x},\boldsymbol{y})) \tag{3.20}$$

ここで

$$\phi_k(\boldsymbol{x},\boldsymbol{y}) = \sum_t \phi_k(\boldsymbol{x},y_t,y_{t-1}) \tag{3.21}$$

となるように ϕ を設計すれば, 以下の形となり, ビタビアルゴリズムなどで解が求まる.

$$\boldsymbol{y^*} = \arg\max_{\boldsymbol{y}} \sum_t \boldsymbol{w}\cdot\phi(\boldsymbol{x},y_t,y_{t-1}) \tag{3.22}$$

CRF と通常の識別モデルとの違いは, 出力が出力集合 Y の部分集合ではなく, 系列となる点にある. CRF は, 品詞付与, テキストチャンキング, 固有表現抽出, HTML からの情報抽出, 書誌データからの情報抽出, といった系列ラベリング問題に適用され, いずれにおいても高い精度を示している.

第4章　文字ベースのCRF

4.1　文字ベース

通常，CRF の素性として単語の品詞を利用するが，中国語の場合，標準的に利用できる形態素解析システムが存在しないために，単語分割，品詞付与の処理が容易ではない．この背景の一つは，中国語解析の困難性であると考える．中国語では英語のような単語の分かち書きを行わない．また，日本語では文字種が単語分割のための大きな情報を持つが，中国語はほぼ単一文字種（漢字）である．さらに，複数品詞を持つ語が多いため品詞付与も容易ではない．例えば，中国語の介詞（前置詞）のほとんどは動詞からの転成であるため，内容語と機能語との間で品詞付与の曖昧性が生じる．これは，日本語や英語ではほとんど生じない現象である．また，日本語における「－する」（動詞）「－い」（形容詞）などの明確な文法標識を持たないため，内容語間の曖昧性も比較的多い．

そこで, ここでは文字ベースを用いる. 文字ベースは形態素解析が必要なく, 辞書なしに CRF が実行可能な特徴がある．

4.2　利用する素性

ここでは文字ベースの CRF なので, 文字だけを利用して素性を構成する. ここで利用した素性は 6 種類である．注目している文字が c_0 として, 以下の文字列を例とする．

$$\cdots c_{-2}c_{-1}c_0c_1c_2\cdots$$

1. c_{-2}：2 文字前の文字
2. c_{-1} :1 文字前の文字
3. c_0：注目している文字
4. c_1：1 文字後の文字
5. c_2：2 文字後の文字
6. $c_{-1}c_0$：直前文字と注目文字の 2 文字列
7. c_0c_1：注目文字と直後文字の 2 文字列

4.3 CRF++の使い方

4.3.1 ダウンロード

CRF++はフリーソフトであり, 以下のサイトからダウンロードできる.

`http://crfpp.sourceforge.net/`

4.3.2 形態素解析の場合

形態素解析による CRF の実行手順[5]は以下のとおりである.
a) トレーニングセットとテストセット
事前に訓練データ, テストデータ準備する. 訓練データは形態素解析により品詞タグを付け, ラベルも貼る. 図 4.1 のように訓練データには 3 列になっている. 1 行目は単語, 2 行目は品詞, 3 行目はラベルである.

```
He        PRP  B-NP
reckons   VBZ  B-VP
the       DT   B-NP
current   JJ   I-NP
account   NN   I-NP
deficit   NN   I-NP
will      MD   B-VP
narrow    VB   I-VP
to        TO   B-PP
only      RB   B-NP
#         #    I-NP
1.8       CD   I-NP
billion   CD   I-NP
in        IN   B-PP
September NNP  B-NP
.         .    O

He        PRP  B-NP
reckons   VBZ  B-VP
```

図 4.1: トレーニングデータ

```
Rockwell NNP B-NP
International NNP I-NP
Corp. NNP I-NP
's POS B-NP
Tulsa NNP I-NP
unit NN I-NP
said VBD B-VP
it PRP B-NP
signed VBD B-VP
```

図 4.2: テストデータ

b) 素性テンプレート
どのような素性を採用するか表記したものである.

`%x[i,j]`

現在の位置からの相対位置で i 行目の j 番目の列の要素を指す.

[5]http://crfpp.sourceforge.net/を参考にした.

```
# Unigram
U00:%x[-2,0]
U01:%x[-1,0]
U02:%x[0,0]
U03:%x[1,0]
U04:%x[2,0]
U05:%x[-1,0]/%x[0,0]
U06:%x[0,0]/%x[1,0]

U10:%x[-2,1]
U11:%x[-1,1]
U12:%x[0,1]q
U13:%x[1,1]
U14:%x[2,1]
U15:%x[-2,1]/%x[-1,1]
U16:%x[-1,1]/%x[0,1]
U17:%x[0,1]/%x[1,1]
U18:%x[1,1]/%x[2,1]

U20:%x[-2,1]/%x[-1,1]/%x[0,1]
U21:%x[-1,1]/%x[0,1]/%x[1,1]
U22:%x[0,1]/%x[1,1]/%x[2,1]
```

図 4.3: テンプレート

c) トレーニング

crf_learn コマンドを使ってトレーニングモデルを作成する. テンプレートファイルが $template\_file$, トレーニングファイルが $train\_file$ の時, 以下のコマンドでトレーニングモデルが生成し,$model\_file$ に入る.

```
% crf_learn template_file train_file model_file
```

```
CRF++: Yet Another CRF Tool Kit
Copyright(C)2005-2007 Taku Kudo, All rights reserved.

reading training data:
Done!0.00 s

Number of sentences: 2
Number of features:  1800
Number of thread(s): 1
Freq:                1
eta:                 0.00010
C:                   1.00000
shrinking size:      20
Algorithm:           MIRA

iter=0 terr=0.66667 serr=0.50000 act=2 uact=0 obj=0.30126 kkt=12.00000
iter=1 terr=0.16667 serr=0.50000 act=2 uact=0 obj=0.36494 kkt=2.84937
iter=2 terr=0.00000 serr=0.00000 act=2 uact=0 obj=0.36494 kkt=0.00000
iter=3 terr=0.00000 serr=0.00000 act=2 uact=0 obj=0.36494 kkt=0.00000

Done!0.00 s
```

図 4.4: トレーニング学習

d) テスト

crf_test コマンドを使ってテストを行う. テストデータが $test\_files$ の時以下のようなコマンドの入力によりシステムからラベルを付けてくれる. テスト結果は図 4.5 のようで, 単語, 品詞タグ, ラベル, システムから付けたラベルの 4 列になっている.

```
% crf_test -m model_file test_file
```

```
He      PRP     B-NP    B-NP
reckons VBZ     B-VP    B-VP
the     DT      B-NP    B-NP
current JJ      I-NP    I-NP
account NN      I-NP    I-NP
deficit NN      I-NP    I-NP
will    MD      B-VP    B-VP
narrow  VB      I-VP    I-VP
to      TO      B-PP    B-PP
only    RB      B-NP    B-NP
#       #       I-NP    I-NP
1.8     CD      I-NP    I-NP
billion CD      I-NP    I-NP
in      IN      B-PP    B-PP
September       NNP     B-NP    B-NP
.       .       O       O

He      PRP     B-NP    B-NP
reckons VBZ     B-VP    B-VP
```

図 4.5: テスト結果

第5章　人名構成文字確率

5.1　確率の推定

日本語や中国語では漢字文字に意味がある場合が多く, 人名に使われる漢字には傾向があると思われる. 例えば, あまり悪い意味の漢字を人名に使うようなことはない.

ここでは漢字 c が人名になる確率 $P(c)$ を推定し,$P(c) \leq \theta$ となる c には H というラベルをつけ,$P(c) < \theta$ となる c には L となるラベルを付けて, このラベルの情報を素性に加えて CRF を学習する.

$P(c)$ の推定手順として, まず以下のサイトから中国語の 15,147 文書を得た.

`http://download.csdn.net/detail/finallyliuyu/2123131`

各文書はプレーンな中国語のテキストであり, 平均 1196.5 文字からなっている. このコーパス中の文字 c の頻度 $f(c)$ をカウントする. 次に, このコーパス中で文字 c が人名の構成文字として使われた回数 $g(c)$ をカウントすることで,$P(c) = g(c)/f(c)$ と推定できる.

$f(c)$ のカウントは容易だが,$g(c)$ の正確なカウントは困難であり, ここでは以下のような処理で $g(c)$ を求めた.

まず, 全ての文字 c に対して $g(c) = 0$ とおく. 次に, 以下のサイトから中国語の人名のリストを得る. このリストには約 65537 人の人名が記載されている.

`http://www.gangzi.org/article/463.htm`

この中から文字列の長さが k の人名に注目する.

$$c_1 c_2 \cdots c_k$$

単純な文字列一致を利用して, この人名が先のコーパスに現れる頻度 h をカウントし, $g(c_i) += h(i = 1 \sim k)$ とする. これを $k = 2$ から 7 まで動かすことで, $g(c)$ を得る. 得られた $P(c)$ の上位 10 個を以下に示す.

人名の文字列長とその種類数を以下に示す.

図 5.1: 人名構成文字確率の上位 10 文字

表 5.1: 人名の文字列長とその種類数

長さ	種類数
2	15,438
3	48,056
4	1,488
5	408
6	70
7	30
8 以上	53
合計	65,537

5.2 素性の追加

上記で得た $P(c)$ を用い, その値が $\theta = 0.01$ 以上となる c には H というラベルをつけ,θ 未満となる c には L というラベルを付ける. このラベルをここではラベル P と呼ぶことにする. このラベル P を品詞のように扱い, 以下に示す 12 種類の素性を CRF の学習に追加した.

注目している文字が c_0 として, 以下の文字列を例とする.

$$\cdots c_{-2}c_{-1}c_0c_1c_2\cdots$$

c_i に対するラベル P を $p_i(p_i \in \{H, L\})$ とおく.

1. p_{-2} : 2 文字前の文字のラベル P
2. p_{-1} : 1 文字前の文字のラベル P
3. p_0 : 注目している文字のラベル P
4. p_1 : 1 文字後の文字のラベル P
5. p_2 : 2 文字後の文字のラベル P
6. $p_{-2}p_1$: 2 文字前の文字のラベル P と直前文字のラベル P の列
7. $p_{-1}p_0$: 直前文字のラベル P と注目文字のラベル P の列
8. p_0p_1 : 注目文字のラベル P と直後文字のラベル P の列
9. p_1p_2 : 直後文字のラベル P と 2 文字後の文字のラベル P の列
10. $p_{-2}p_{-1}p_0$: 2 文字前の文字のラベル P と直前文字のラベル P と注目文字のラベル P の列
11. $p_{-1}p_0p_1$: 直前文字のラベル P と注目文字のラベル P と直後文字のラベル P の列
12. $p_0p_1p_2$: 注目文字のラベル P と直後文字のラベル P と 2 文字後の文字のラベル P の列

第6章　実験

6.1　実験データ

6.1.1　コーパスと人名リスト

まずは以下のサイトから 15147 個の文書を得た. こちらはは 2009 年 12 月の Tengxun という中国サイトの記事である.

http://download.csdn.net/detail/finallyliuyu/2123131

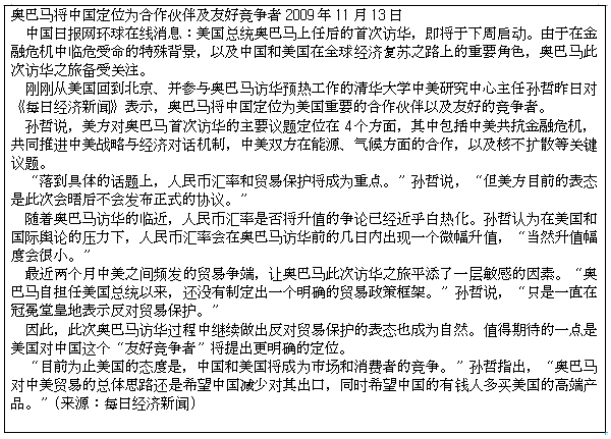

奥巴马将中国定位为合作伙伴及友好竞争者 2009 年 11 月 13 日
　中国日报网环球在线消息：美国总统奥巴马上任后的首次访华，即将于下周启动。由于在金融危机中临危受命的特殊背景，以及中国和美国在全球经济复苏之路上的重要角色，奥巴马此次访华之旅备受关注。
　刚刚从美国回到北京、并参与奥巴马访华预热工作的清华大学中美研究中心主任孙哲昨日对《每日经济新闻》表示，奥巴马将中国定位为美国重要的合作伙伴以及友好的竞争者。
　孙哲说，美方对奥巴马首次访华的主要议题定位在 4 个方面，其中包括中美共抗金融危机，共同推进中美战略与经济对话机制，中美双方在能源、气候方面的合作，以及核不扩散等关键议题。
　“落到具体的话题上，人民币汇率和贸易保护将成为重点。”孙哲说，“但美方目前的表态是此次会晤后不会发布正式的协议。”
　随着奥巴马访华的临近，人民币汇率是否将升值的争论已经近乎白热化。孙哲认为在美国和国际舆论的压力下，人民币汇率会在奥巴马访华前的几日内出现一个微幅升值，“当然升值幅度会很小。”
　最近两个月中美之间频发的贸易争端，让奥巴马此次访华之旅平添了一层敏感的因素。“奥巴马自担任美国总统以来，还没有制定出一个明确的贸易政策框架。”孙哲说，“只是一直在冠冕堂皇地表示反对贸易保护。”
　因此，此次奥巴马访华过程中继续做出反对贸易保护的表态也成为自然。值得期待的一点是美国对中国这个“友好竞争者”将提出更明确的定位。
　“目前为止美国的态度是，中国和美国将成为市场和消费者的竞争。”孙哲指出，“奥巴马对中美贸易的总体思路还是希望中国减少对其出口，同时希望中国的有钱人多买美国的高端产品。”（来源：每日经济新闻）

図 6.1: コーパスの例

次に以下の中国サイトから 65537 人の人名が記載されている文書を得た.

http://www.gangzi.org/article/463.htm

6.1.2　テキストへのタグ付け

先ほど述べた 15147 個の文書からランダム 1000 個の文書を選べ, タグ付けを行った. 次の図はタグ付けしたテキストの一部である. ここで人名は $<hc><$ $/hc>$ で囲んだ.

id	name
1	阿宝宝
2	阿爆
3	阿倍仲麻吕
4	阿比
5	阿碧
6	阿扁
7	阿炳
8	阿波罗
9	阿伯拉默夫
10	阿博德
11	阿卜杜尔
12	阿卜杜热合曼
13	阿不都
14	阿不都哈地尔
15	阿不都卡德尔·尤努斯
16	阿不都克里木·热合满
17	阿不都克力木·吾卖
18	阿不都麻木提
19	阿不都尼衣木
20	阿不都热合曼·乃买提
21	阿不都热苏力
22	阿不都热西提
23	阿不都热西提·买买提
24	阿不都热依木
25	阿不拉江
26	阿不来提
27	阿不来提·阿不都热西提

図 6.2: 人名リストの例

11 月 18 日，中国国务院总理<hc>温家宝</hc>在北京钓鱼台国宾馆会见正在中国进行国事访问的美国总统<hc>奥巴马</hc>。新华社记者<hc>刘建生</hc>摄 11 月 18 日，中国国务院总理<hc>温家宝</hc>在北京钓鱼台国宾馆会见正在中国进行国事访问的美国总统<hc>奥巴马</hc>。新华社记者<hc>马占成</hc>摄中新网北京 11 月 18 日电中国总理<hc>温家宝</hc>18 日上午在北京钓鱼台国宾馆会见美国总统<hc>奥巴马</hc>。<hc>温家宝</hc>说，总统先生这次访问的成果是有深远意义的。中美两国在世界上是有重要影响的国家。总统先生在上海对青年学生讲话时引用孔子的话“温故知新”，这句话说得很好，继往开来需要回顾中美关系的历史。它告诉我们一个道理，中美和则两利，斗则俱损；互信则进，猜忌则退。合作比遏制好，对话比对抗好，伙伴比对手好。<hc>温家宝</hc>表示，我真诚希望通过总统先生的这次访问，使中美全面合作关系进入新的阶段。两人见面时，<hc>奥巴马</hc>用中文“你好”向<hc>温家宝</hc>致意。<hc>奥巴马</hc>表示，我这两天的访问非常有成果。我和<hc>胡锦涛</hc>主席在会谈中达成一致，加强中美两国的战略互信。以前，中美关系的焦点集中在经济、贸易方面，现在已经拓展到各个领域。这是<hc>奥巴马</hc>就职美国总统后，首次与中国总理会晤。

図 6.3: タグ付きテキスト

6.1.3 トレーニングデータとテストデータ

文字ベースの CRF を使うためにタグ付けした 1000 個のデータを 1 文字 1 行にし, 人名の先頭には B-I, 人名内の単語には B-O, 人名ではない単語には O とラベルをつける.

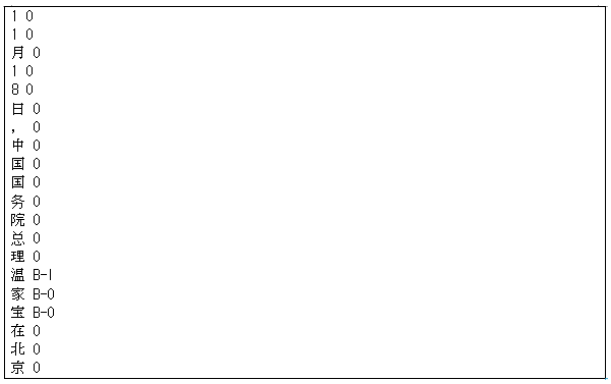


図 6.4: 1 文字 1 行

このデータセットを用いて 10 分割の交差検定を行った. 交差検定法は重回帰分析などの応用に用いられている検定法の 1 種である. 具体的には, サンプル集合 X から平均二乗誤差を推定する方法であり, あるサンプルを用いて学習と実験を行いたい場合に, サンプル量が少なくて偏りが予想されるときに使う. 以下に 3 種類の精度推定法[6]について述べる.

a) 代替推定法（Resubstitution estimate） これは非常に単純な考え方で, 決定木 (d) で表されたモデルに, そのモデルを構築したケース（N 件とする）を再度あてはめ, どのくらい正確に分類される（t 件とする）かをそのままモデル精度の推定値としてやろうという方法である. 代替推定法による精度 A(d) は次の式で表される.

$$A(d) = \frac{t}{N}$$

b) テストサンプル法（Test sample estimate）

モデルを構築するためのデータと, その精度を推定するためのデータを予め分けて用意する方法がテストサンプル法である. まず基のデータを, トレーニングデータとテストデータと呼ばれる二つのデータにランダムに分割する（分割の比率は, テストデータを 1/3, トレーニングデータを 2/3 とするのが一

[6] http://musashi.sourceforge.jp/tutorial/mining/xtclassify/accuracy.html に載っている.

般的である）. また, 両データとも, 元のデータを代表するようなデータであることが望まれる. 例えば, トレーニングトデータに「ゴルフをする」ケースが全く含まれていなかったり, 気温が高いケースばかりが含まれていては問題である. そこで通常は「層化（Stratification）」と呼ばれる手法が用いられる. もとデータからサンプリングを行なう際に, 各属性の値の分布が, 元データの分布と同じようになるようにサンプリングするのである.

モデルの生成はトレーニングデータ（解答付きデータ）を用いて作成し, そのモデルにテストデータを当てはめる. そしてテストデータにおける分類精度を, そのモデルの精度とする（図 6.3 参照）. 受験勉強で言えば, 参考書のいくつかの問題を解答を見ずに解いて見て, どのくらい正解するかを確かめることに当たる.

テストデータの総件数を N^ts とし, そのうち, モデルに当てはめて実際に正解であった件数を t^ts とすると, テストサンプル法による推定精度 $A^ts(d)$ は次式で表される.

$$A^ts(d) = \frac{t^ts}{N^ts}$$

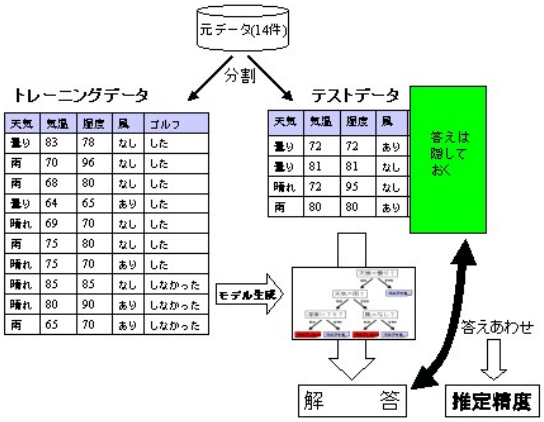


図 6.5: テストサンプル法

c) 交差検証法（n-flod cross-validation estimate）

テストサンプル法は, データ量が十分ある場合には問題になりませんが, データ量が少ない時には, テストデータの選び方によって, 推定精度に大きな誤差が生じる可能性が高くなる. そのような場合には, ここで説明する交差検証法が使われる. この方法では, まず元のデータをｎ個のブロックに分割する（通常ｎとしては 10〜20 の値が用いられる）. その際, 各ブロックに割り当て

られる件数が同程度になるようにする. また各ブロックとも, 層化されていることが望まれる.

そしてまず, 一つ目のブロックをテストデータ, その他のブロックをトレーニングデータとして, モデルの構築と精度の算出を行なう. 次に二つ目のブロックをテストデータとし, その他のブロックをトレーニングデータとし, モデル構築を行なっていく. このような手続きをｎ回繰返し, 各回で算出された精度の平均を, モデルの推定精度としようというものです. お気づきと思うが, 各回の試行は, テストデータの割合が 1/n によるテストサンプル法であることがわかる.

交差検証法を使うと, 全てのケースが, 一回はテストデータとして選ばれ, なおかつ見かけ上元データのｎ－１倍の件数のトレーニングデータを用いてモデルを構築したことになり, 少ないデータであっても推定誤差が少くなる.
図 6.4 でｎ回目に構築された決定木を d^n とし, その精度を $A^{ts}(d^n)$ とすると, 交差検定による推定精度 $A^{cv}(d)$ は次の式で与えられます.

$$A^{cv}(d) = \frac{1}{n}\sum_{i}^{n} A^t s(d^i)$$

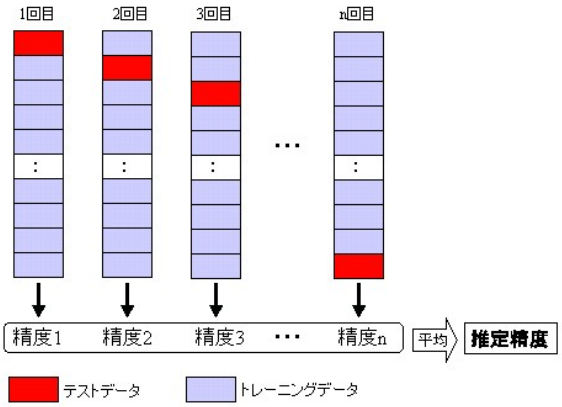
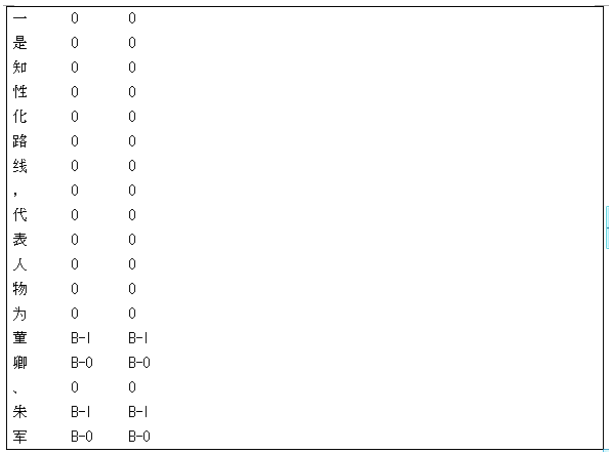


図 6.6: 交差検定法

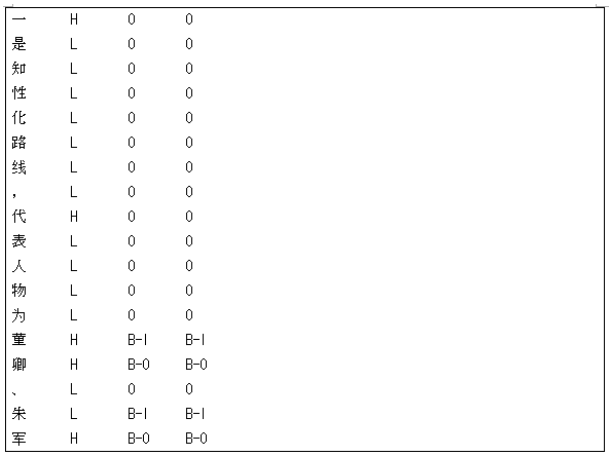
6.2 実験方法

本稿で提案した手法の有効性を示すために, 実験を２回行った. まずはCRF++ソフトを使って, 文字ベースの CRF 実験を行った. 次に文字構成確率を素性として与えた文字ベースの CRF 実験を行った. ２回の実験手順は同じあるが訓練データ, テンプレートファイルが違う.



一	0	0
是	0	0
知	0	0
性	0	0
化	0	0
路	0	0
线	0	0
,	0	0
代	0	0
表	0	0
人	0	0
物	0	0
为	0	0
董	B-I	B-I
卿	B-0	B-0
、	0	0
朱	B-I	B-I
军	B-0	B-0

図 6.7: 文字ベースによる CRF 実験



一	H	0	0
是	L	0	0
知	L	0	0
性	L	0	0
化	L	0	0
路	L	0	0
线	L	0	0
,	L	0	0
代	H	0	0
表	L	0	0
人	L	0	0
物	L	0	0
为	L	0	0
董	H	B-I	B-I
卿	H	B-0	B-0
、	L	0	0
朱	L	B-I	B-I
军	H	B-0	B-0

図 6.8: 人名構成文字確率の情報を追加利用した CRF 実験

6.3 実験結果と評価

固有表現抽出の精度を評価する代表的な指標は, 精度 P, 再現率 R,F 値 F の 3 つである.

$$P = \frac{\text{正しく抽出できた人名数}}{\text{システムが抽出した人名数}}$$

$$R = \frac{\text{正しく抽出できた人名数}}{\text{正しい人名数}}$$

$$P = \frac{2 \times P \times R}{P + R}$$

つまり, 精度はモデルが人名だとして出力したものの正答率, 再現率はモデルが全人名のうちいくつを見つけることができたかを意味する. そして適合率と再現率は一般にトレードオフの関係にあるので, これらのバランスを考慮した全体的な認識精度のよさを与える指標として F 値がある.

表 6.1: 文字ベースのみの CRF

data	人名数	抽出数	正解数	精度	再現率	F 値
1	914	622	561	0.9019	0.6138	0.7305
2	1088	738	685	0.9282	0.6296	0.7503
3	895	550	497	0.9036	0.5553	0.6879
4	1144	668	596	0.8922	0.5210	0.6578
5	911	596	531	0.8909	0.5829	0.7047
6	1040	764	706	0.9241	0.6788	0.7827
7	979	595	530	0.8908	0.5414	0.6734
8	642	465	414	0.8903	0.6449	0.7480
9	879	633	565	0.8926	0.6428	0.7474
10	722	580	525	0.9052	0.7271	0.8065
平均	921.4	621.1	561.0	0.9020	0.6138	0.7289

表 6.2: 人名構成文字確率を追加利用した CRF

data	人名数	抽出数	正解数	精度	再現率	F 値
1	914	653	596	0.9127	0.6521	0.7607
2	1088	771	715	0.9274	0.6572	0.7692
3	895	577	513	0.8891	0.5732	0.6970
4	1144	782	713	0.9118	0.6233	0.7404
5	911	614	551	0.8974	0.6048	0.7226
6	1040	790	730	0.9241	0.7019	0.7978
7	979	595	531	0.8924	0.5424	0.6747
8	642	488	434	0.8893	0.6760	0.7681
9	879	673	598	0.8886	0.6803	0.7706
10	722	622	556	0.8939	0.7701	0.8274
平均	921.4	656.5	593.7	0.9027	0.6481	0.7528

第7章　考察

本論文では人名構成文字確率を H と L の二値の素性として導入したが, 素性の形式には連続値のまま導入したり, 三値以上にすることも考えられる. ただし連続値は CRF の素性としては使えないので, なんらかの方法で離散化する必要がある. 何種類の離散値にすればよいかは問題だが, 訓練データの量に依存していると考える. つまり種類数が多い場合は, その素性の並びパターンが多いという長所があるが, そのパターンの頻度が少なくなるという短所がある. 本論文では少量の訓練データを想定したので二値としたが, 三値以上の精度は未確認である.

また H と L を区別する閾値の設定の問題もある. 本論文では訓練データ中の人名構成文字の人名構成文字確率の分布を調べ, 訓練データ中の人名構成文字文字の約半数が 0.01 以上であったので, この値を閾値とした. この閾値を小さくすれば, 人名にのラベルが付くので, 抽出が容易になりそうだが, 人名以外の文字にも H のラベルが付くケースも増えるために, 単純ではない. 最適な閾値は交差検定などを行うことで推定できる.

またここでは比較的大規模な人名リストを利用できた. そのような人名リストがあれば, 文字列一致だけを利用して文書中の人名候補を取り出せる. 候補になった文字列の文字に H, それ以外の文字に L の素性を与えれば, 本論文と厳密には異なるが, ほぼ同様の情報を学習に取り入れることになる. このような素性は冗長でも同時に利用することができるので, その点で改良が可能だと考える.

本実験の未検出と誤検出の原因として文字ベースゆえの学習能力の低さがあると思われるが, 誤り検出についてはタグ付けの誤りも多かった. 本論文で使用した人名タグのコーパスは中国人学生である私が手作業で構築したものであり, 多くの誤りが含まれている. 実際に交差検定で使われた data1 のテストデータについて調べると, 文字ベースの CRF では 61 個の誤り中 35 個はタグ付けの間違いであり, 実際には正解となるべきものであった. また人名構成文字確率の情報を付与した本方法では, 同じテストデータで,57 個の誤り中 37 個が実際は正解であった. このようにかなりの数のタグ付与されていない人名があるために, 本実験での実際の正解率は, ここで出された正解率よりもかなり高いと思われる. ただし本手法の優位性は変化はないはことは注記しておく. また未検出については 1 文字で構成される人名も目立った. 例えば data1 のテストデータについて調べると,179 個の 1 文字による人名が存在したが, 文

字ベースの CRF では 105 個, 本手法では 103 個が未抽出となっている.1 文字で構成される人名の検出が困難であること [10] はでも論じられている. 中国語の姓は 1 文字で構成されるものが多いために, この点は何らかの対策が必要である.

今後の課題としては, 人名以外の固有表現 (地名や組織名など) への応用が考えられる. 他の固有表現であっても構成文字ある傾向が存在すると考えている. 本手法は比較的大規模な固有表現のリストがあれば利用可能である. そのようなリストは Web 上から比較的容易に収集できるために, 本手法を人名以外の固有表現の抽出に試したい.

第8章　おわりに

本論文では中国の人名抽出のタスクを文字ベースの CRF を用いて行った. 中国語のコーパスと中国語の人名リストを利用して, 各文字が人名文字列の構成要素になる確率を推定した. この確率がある閾値以上のものに H, その閾値 $ 未満のものに L というラベルを与え, その素性を文字ベースの CRF に追加りようすることで, 単純な文字ベースの CRF の精度を改善することができた. 今後の課題としては人名以外の固有表現にも, その表現要素として利用できるかどうか調べたい.

謝辞

本研究を遂行するにあたり，日頃より適切な御指導をいただいた，茨城大学工学部情報工学科の新納浩幸教官に深く御礼申し上げます．また, 佐々木稔先生にも多くの有益なご助言をいただきました．深く感謝いたします.

参考文献

[1] 関根　聡. テキストからの情報抽出.

[2] J. Lafferty, A. McCallum, and F. Pereira. Conditional Random Fields: Probabilistic models for segmenting and labeling sequence data. In ICML, pp. 282?289, 2001.

[3] 工藤拓, 山本薫, 松本裕治.Conditional Random Fields を用いた日本語形態素解析

[4] 黄　根倡，万　如. 基于 SVM 和 CRF 的双模型中文.2010

[5] L. Li, Z. Li nad Z. Ding, and D. Huang. A Hybrid Model Combining CRF with Boundary Templates for Chinese Person Name Recognition. Advanced Intelligence, Vol. 2, No. 1, pp. 73?80, 2010.

[6] CW. Wu, SY. Jan, RTH. Tsai, and WL. Hsu. On Using Ensemble Methods for Chinese Named Entity Recognition. In the Fifth SIGHAN Workshop on Chinese Language Processing, pp. 142?145, 2006.

[7] 白井清昭, 乾健太郎, 徳永健伸, 田中穂積. 最大エントロピー法による確率モデルのパラメタ推定に有効な素性の選択について.

[8] 揚石亮平, 三浦孝夫. 固有名詞の認識を含む HMM による英文形態素解析.2008

[9] 坪井祐太, 鹿島久嗣, 工藤拓. 言語処理における識別モデルの発展-HMM から CRF まで

[10] X. Zhu, M. Li, J. Gao, and CN. Huang. Single character Chinese named entity recognition. In the Second SIGHAN Workshop on Chinese Language Processing, pp. 125?132, 2003.

[11] 高村大也. 言語処理のための機械学習入門. コロナ社,2010.

付 録A　プログラム

```
/*make_name.py(ファイル名前をつける)*/
#!/usr/bin/env python
import glob
filelist = glob.glob("*.txt")
for x in range(len(filelist)):
    f = open(filelist[x],'r')
    all=f.read()
    f.close()
    f = open('/home/taisyun/python/corpus/corpus/data/
'+str(x+1)+'.txt','w')
    f.write(all)
    f.close()
```

```
/*make_1000.py(1000 個の人名リスト作成):/
#!/usr/bin/env python
x=1
count=1
while x<15148:
    if(count>1000):
        break
    f = open(str(x)+'.txt','r')
    all=f.read()
    f.close()
    f = open('/home/taisyun/python/data3/'+str(count)+'.txt','w')
    f.write(all)
    f.close()
    x+=10
    count+=1
```

```
/*template.py(テンプレートを作成)*/
#!/usr/bin/env python
#coding=utf-8
x=1
f1 = open('mozi.txt','w')
while x<15148:
    f = open(str(x)+'.txt','r')
    all=f.read()
    f.close()
    u1 = unicode(all,'utf8')
    for i in range(2,len(u1)-2):
        word=u1[i].encode("utf8")
        e1=u1[i-1].encode("utf8")
        e2=u1[i-2].encode("utf8")+e1
        e3=u1[i+1].encode("utf8")
        e4=e3+u1[i+2].encode("utf8")
        f1.write(word+''+e1+' '+e2+' '+e3+' '+e4+'\n')
    x+=1
f1.close()
```

```
/*make_tag.py(１列１文字にラベルを付ける)*/
#!/usr/bin/env python
#coding=utf-8
x=1
while x<1001:
    f = open(str(x)+'.txt','r')
    lines=f.readlines()
    f.close()
    f = open('1.dat', 'w')

    for line in lines:
        i=0
        u = unicode(line,"utf8")
        while i<len(line):
            if u[i].encode("utf8")=='<' and u[i+1].encode("utf8")=='h'
                and u[i+2].encode("utf8")=='c' and u[i+3].encode("utf8")=='>':
                i+=4
                j=0
                while u[i].encode("utf8")!='<' and u[i+1].encode("utf8")!='/'
                    and u[i+2].encode("utf8")!='h'and u[i+3].encode("utf8")!='c'
                    and u[i+4].encode("utf8")!='>':
                    if j==0:
                        f.write(u[i].encode('utf8'))
                        f.write(' B-I\n')
                        j+=1
                    else:
                        f.write(u[i].encode('utf8'))
                        f.write(' B-O\n')
                    i+=1
                i+=5
            else:
                f.write(u[i].encode("utf8"))
                f.write(' O\n')
                if u[i].encode("utf8")=="":
                    f.write('\n')
                i+=1
f.close()
```

付 録B　テンプレート

```
/*文字ベース CRF*/
# Unigram
U00:%x[-2,0]
U01:%x[-1,0]
U02:%x[0,0]
U03:%x[1,0]
U04:%x[2,0]
U05:%x[-1,0]/%x[0,0]
U06:%x[0,0]/%x[1,0]

# Bigram
B
```

/*構成文字確率の情報を付与した CRF*/

```
# Unigram
U00:%x[-2,0]
U01:%x[-1,0]
U02:%x[0,0]
U03:%x[1,0]
U04:%x[2,0]
U05:%x[-1,0]/%x[0,0]
U06:%x[0,0]/%x[1,0]

U10:%x[-2,1]
U11:%x[-1,1]
U12:%x[0,1]
U13:%x[1,1]
U14:%x[2,1]
U15:%x[-2,1]/%x[-1,1]
U16:%x[-1,1]/%x[0,1]
U17:%x[0,1]/%x[1,1]
U18:%x[1,1]/%x[2,1]

U20:%x[-2,1]/%x[-1,1]/%x[0,1]
U21:%x[-1,1]/%x[0,1]/%x[1,1]
U22:%x[0,1]/%x[1,1]/%x[2,1]

# Bigram
B
```