

## HTMLの木構造を利用した条件付確率場による固有表現分類: Wikipediaからのシソーラス半自動構築

渡邊陽太郎 浅原正幸 松本裕治  
奈良先端科学技術大学院大学 情報科学研究科  
{yotaro-w, masayu-a, matsu}@is.naist.jp

本稿では、Wikipedia 内に出現する固有表現を獲得し、精度よく分類する手法を提案する。Wikipedia の記事に出現するアンカーテキストの単語および句は、リンク先の記事に語釈が記述されている。この Wikipedia の特性を用いて、我々は、固有表現の分類問題を固有表現を表すアンカーテキストに対するラベル付与問題として定式化する。まず、アンカーテキストをノードとして定義されるグラフを構成する。次に、グラフに HTML の構造を取り入れるため、HTML の DOM 構造に基づく 3 種類のエッジを導入する。このようにして構成したグラフのノードに対するラベル付与を教師あり学習器である Conditional Random Fields (CRFs) を用いて行う。しかし、構成したグラフは閉路を含むため、CRFs の正確な演算を行うことは計算量が大きく困難である。そこで、Tree-based Reparameterization (TRP) を用いて近似的に演算をおこなう手法を導入する。実施した評価実験において、提案手法が 2 つ組に対する Support Vector Machines の順次適用による手法と比較して高い精度で固有表現の分類ができたことを報告する。

### Named Entity Categorization Using Conditional Random Fields on HTML Tree Structure: Semi-Automatic Thesaurus Construction from Wikipedia

Yotaro Watanabe Masayuki Asahara Yuji Matsumoto  
Graduate School of Information Science, Nara Institute of Science and Technology  
{yotaro-w, masayu-a, matsu}@is.naist.jp

This paper presents a method for categorizing named entities in Wikipedia. In Wikipedia, an anchor text is glossed in a linked HTML text. We formalize named entity categorization as a task of categorizing anchor texts with linked HTML texts which glosses a named entity. Using this representation, we introduce a graph structure in which anchor texts are regarded as nodes. In order to incorporate HTML structure on the graph, three types of cliques are defined based on the HTML DOM structure. We propose a method with Conditional Random Fields (CRFs) to categorize the nodes on the graph. Since the defined graph include cycles, the exact inference of CRFs is computationally expensive. We introduce an approximate inference method using Tree-based Reparameterization (TRP) to reduce computational cost. Experimental results show that the proposed method outperforms a baseline method that uses Support Vector Machines.

#### 1 はじめに

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、日付表現などを指し、自然言語処理の応用分野である関係抽出や情報検索、質問応答などにおいて非常に重要や役割を持つ。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。

固有表現抽出は近年では機械学習に基づく手法が主に用いられるが、テキスト中の固有表現を全て網羅できるわけではなく限界がある。そこで、獲得できる固有表現については、既存の資源などを利用して獲得することが必要となる。

固有表現獲得のための資源として我々は Wikipedia<sup>\*1</sup> に注目した。Wikipedia は Web 上の多言語百科事典であり、

日々新たな記事が追加されている。記事の見出し語には固有表現が数多く含まれ、DOM 構造やカテゴリなど、抽出の手がかりとなる情報が豊富である。また、各記事からアンカーによって別の関連する記事を参照することができる。このような特徴を持つ Wikipedia は、固有表現の獲得に適した資源であると考えられる。

Wikipedia 中の記事は HTML(半構造データ)であり、生テキストには無い特徴を持っている。特に注目すべき点として、文書中に存在するアンカーの出現に依存関係があることが挙げられる。例えば、リスト<LI>において列挙されているアンカーは、同じクラスに属するような固有表現が記述されている記事を参照している傾向がある。そのようなアンカー相互の依存関係を捉えた分類をおこなうことで、高精度な固有表現の獲得が期待できる。

このような依存関係を考慮した分類手法としては、局所的な分類を繰り返すことで全体の分類をおこなう Iterative Classification、大域的最適化に基づく分類手法である Collective Classification に大別される。Iterative Clas-

\*1 <http://ja.wikipedia.org/>

sificationの手法としては、HTMLの各文書のカテゴリを、リンク関係にある文書のカテゴリを考慮して推定する手法[6]などがあるが、これらは分類の順序によって結果が変化するという問題点がある。Collective Classificationの手法としては、GetoorらのProbabilistic Relational Models (PRMs) [2]があるが、Bayesian Networksに基づく有向グラフモデルであるため、HTMLのリンクなど循環するような対象を扱う場合や、対象が相互に依存しているような場合は直接のモデル化が困難であるという問題点がある。一方、TaskarらのConditional Markov Networks(またはConditional Random Fields (CRFs))の特殊形であるRelational Markov Networks (RMNs) [11]は無向グラフモデルであるため、PRMsのような問題は生じない。したがって、大域的最適化に基づく無向グラフモデルが最も優れたモデルであると考えられる。

そこで本稿ではアンカー間の依存関係を考慮した分類をおこなうために、大域的最適化に基づく無向グラフモデルであるConditional Random Fields (CRFs) [4]を用いてWikipedia中の固有表現の分類をおこなう。CRFsを適用する上で重要であるのは、どのようなグラフ構造を構成するかであるが、本稿では依存関係を考慮した分類をおこなうために、DOM構造によって捉えられるアンカー間の依存関係をグラフ構造に反映させたCRFsのモデルを提案する。

## 2 Wikipedia 文書中の固有表現分類

Wikipediaでは、一記事につきある特定の事柄に関して記述され、各記事には、見出し語、その事柄の定義文、その事柄が属する1つ以上のWikipediaのカテゴリが付与されている。ここで、定義文やカテゴリなどの情報を手がかりに、その文書で述べられている事柄が何であるかを個別に分類する文書分類の問題として扱うことで固有表現を獲得するという方法が考えられる。

一方、WikipediaはHTML文書(半構造データ)であるため、生テキストには無い特徴がある。その特徴の中で特に重要な特徴を持っているのは、リスト(<UL><OL>など)およびテーブル(<TABLE>)であり、それらの中に出現する要素間には依存関係がある。本稿では、Wikipediaにて頻繁に出現するリストに焦点を当てることにする。図1はHTML文書の一部と、対応するDOM構造の例である。ここで、<UL>タグ直下のリスト<LI>において最初に出現するアンカー(「ディラード&クラーク」、「カーペンターズ」)は、同じ固有表現カテゴリに属する。このような関係にあるアンカー間には同じカテゴリに属する傾向があり、これは分類精度向上に寄与する特徴である。また、Wikipediaの各記事にはその記事に関連する記事を参照するアンカーが存在し、アンカーに記述されているテキスト(アンカーテキスト)は、リンク先の記事の実体または概念を表している。

ここで、上記のようなリストの特徴およびアンカーテキストの存在を踏まえると、Wikipediaから固有表現抽出の問題を、アンカーテキストに対するラベリング問題として扱うということが考えられる。リストなどの規則性によって出現するアンカーテキストの依存関係を分類に利用することで、個別に分類をおこなうアプローチと比較して高精度な分類が期待できる。

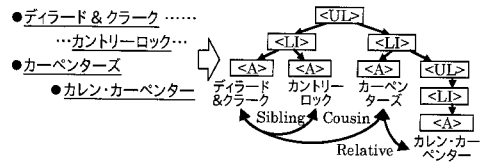


図1: DOM構造と定義したクリークとの対応関係。

依存関係を考慮した分類をおこなうための手法としてCRFsを用い、ノード集合 $V$ をアンカーテキスト、アンカーテキスト間の依存関係を $\mathcal{G}$ のエッジ $E$ とするような無向グラフ $\mathcal{G} = (V, E)$ を構成し、各ノード $v_i \in V$ へのラベル割り当てが全体で最適となるような割り当て $\arg \max_{\mathbf{y}} p(\mathbf{y} | \mathbf{x})$ を求める問題として扱うことで、Wikipedia中の記事から固有表現を獲得する。

### 2.1 DOM構造に基づくグラフ構造の構築

HTMLやXMLといった半構造化された文書は、DOM (Document Object Model) 構造として表現される。ここでDOMとは、HTML文書やXML文書を扱うためのApplication Programming Interface (API) であり、DOMにおいてHTMLおよびXML文書の論理構造は木構造の形で扱われる。DOM構造は、唯一の親を持ち、子が順序をもつ順序木として扱うことができる。図1では、唯一の他と区別される頂点が<UL>に対応し、各タグの子に対応するタグには出現順序があるため、順序木の定義を満たす。よって以後、DOM構造を順序木として扱う。

次に、DOM構造からCRFsを構成する無向グラフ $\mathcal{G}$ のクリーク $C$ を定義するために、DOM構造上において依存関係を持つようなアンカーの組について考えることにする。

図1の例において、

1. 「ディラード&クラーク」と「カントリーロック」は、DOM構造上では兄弟 (Sibling) の関係であり、同一のリストの中で出現している。この間には、「ディラード&クラーク」の楽曲のジャンルが「カントリーロック」であるというような関係が成り立つ。このような出現の仕方には、先に出現したものが持っている属性、または関連する事柄が後ろに来るといった傾向がある。
2. 「ディラード&クラーク」と「カーペンターズ」は、DOM構造上では従兄弟 (Cousin) の関係になっており、これらは双方共に組織名である。このような出現の仕方は、双方が同じようなクラスに属するような傾向がある。
3. 「カーペンターズ」と「カレン・カーペンター」は、DOM構造上では、「カーペンターズ」から見て「カレン・カーペンター」が兄弟の孫という関係になっている。これら間には「カーペンターズ」の構成員として「カレン・カーペンター」が存在するという関係があり、このような出現の仕方は、先に出現したものを構成している要素が後にくる傾向がある。

このような、相互に依存関係があるようなアンカーの集合をクリークを構成するノードとすることで、依存関係を捉えた分類をおこなうことができると考えられる。ここで、上記の観察に基づき、DOM構造上において以下で述べる3つのクリークの定義に該当するものを、CRFsのクリーク

とする。

DOM 構造に対応する順序木を  $\mathcal{T}^{ordered} = (V^T, E^T)$  とする。頂点  $v_i^T, v_j^T \in V^T$  間の距離を  $d(v_i^T, v_j^T)$ 、順序木の頂点  $v_i^T \in V^T$  から  $k$  回辿った先祖を  $pa(v_i^T, k)$ 、頂点  $v_i^T$  の  $k$  番目の子を  $ch(v_i^T, k)$ 、頂点  $v_i^T, v_j^T \in V^T$  の共通の先祖を  $cpa(v_i^T, v_j^T)$ 、と表すことにする。図 2 に示す 3 つの関係をクリークとして定義する。

ここで Sibling は 1 番目、Cousin は 2 番目、Relative は 3 番目の観察に対応したクリークであり、 $E_S$  は Sibling の関係にある頂点の組の集合、 $E_C$  は Cousin の関係にある頂点の組の集合、 $E_R$  は Relative の関係にある頂点の組の集合を表している。図 1 に、DOM 構造とクリークの対応関係を示す。ここで、上記の関係の定義にて、関係として扱う頂点の組を、ある頂点とその頂点から最も近い頂点に限定している。その理由は、「最も近い」という条件を除いた場合、ある頂点と特定の関係にある頂点の数は複数個該当する可能性が出てくるが、それらの組を全てクリークとした場合、計算量の問題が生じるためである。

図 2 の定義を満たす頂点の組を、構成する CRFs のクリークとする。すなわち、 $C = E_S \cup E_C \cup E_R$  である。

## 2.2 DOM 構造に基づくグラフ構造上の条件付確率場

前節で定義したクリーク  $C$  を持つ無向グラフ  $\mathcal{G} = (V, E)$  のクリークに対してポテンシャル関数を定義することで  $p(\mathbf{y}|\mathbf{x})$  を与える CRFs を構成する。観測集合  $\mathbf{x}$  が与えられたときのラベル集合  $\mathbf{y}$  の条件付分布を以下の式で与える。

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \left( \prod_{(v_i, v_j) \in E_S, E_C, E_R} \Phi_{SCR}(y_i, y_j) \right) \left( \prod_{v_i \in V} \Phi_V(y_i, \mathbf{x}) \right) \quad (1)$$

ここで  $Z(\mathbf{x})$  は正規化項、 $\Phi_{SCR}(y_i, y_j)$  は Sibling, Cousin, Relative  $\{(v_i, v_j) \in E_S, E_C, E_R\}$  に対応するクリークのポテンシャル関数、 $\Phi_V(y_i, \mathbf{x})$  はノード  $v_i \in V$  に対応するポテンシャル関数であり、それぞれ以下の式で与える。

$$\Phi_{SCR}(y_i, y_j) = \exp \left( \sum_k \lambda_k f_k(y_i, y_j) \right) \quad (2)$$

$$\Phi_V(y_i, \mathbf{x}) = \exp \left( \sum_{k'} \lambda'_{k'} f'_{k'}(y_i, \mathbf{x}) \right) \quad (3)$$

ここで、 $k \in \{(y_i, y_j) | \mathcal{Y} \times \mathcal{Y}\}$  であり、これはラベル集合  $\mathcal{Y}$  の直積集合のある要素に対応する。また、 $k' \in \{(y_i, x_j) | \mathcal{Y} \times \mathcal{X}\}$  であり、これは観測系列に出現する素性  $x_j$  と、ラベル  $y_i \in \mathcal{Y}$  の共起をとらえるための素性である。

CRFs のパラメータ  $\Lambda = \{\lambda_k, \dots, \lambda'_{k'}, \dots\}$  は、訓練データ  $\mathcal{D} = \{\langle \mathbf{x}^{(1)}, \mathbf{y}^{(1)} \rangle, \langle \mathbf{x}^{(2)}, \mathbf{y}^{(2)} \rangle, \dots, \langle \mathbf{x}^{(N)}, \mathbf{y}^{(N)} \rangle\}$  の条件付対数尤度を最大化するように推定される。対数尤度関数

固有表現クラス		記事数
名前	イベント名 (EVENT)	121
	人名 (PERSON)	3315
	単位名 (UNIT)	15
	地名 (LOCATION)	1480
	施設名 (FACILITY)	2449
	称号名 (TITLE)	42
	組織名 (ORGANIZATION)	991
	職業名 (VOCATION)	303
	自然物名 (NATURAL.OBJECT)	1132
	製品名 (PRODUCT)	1664
	名前_その他 (NAME.OTHER)	24
時間・数値表現	時間・数値表現	2749
固有表現以外		1851
全体		16136

表 1: 訓練・評価用データの固有表現クラスと Wikipedia の記事数一覧

は以下のように定義できる。

$$\begin{aligned} \mathcal{L}_\lambda = & \sum_{d=1}^N \left[ \sum_{(v_i, v_j) \in E_S^{(d)}, E_C^{(d)}, E_R^{(d)}} \sum_k \lambda_k f_k(y_i, y_j) \right. \\ & \left. + \sum_{v_i \in V^{(d)}} \sum_{k'} \lambda'_{k'} f'_{k'}(y_i, \mathbf{x}^{(d)}) - \log Z(\mathbf{x}^{(d)}) \right] - \sum_k \frac{\lambda_k^2}{2\sigma^2} - \sum_{k'} \frac{\lambda'_{k'}{}^2}{2\sigma^2} \end{aligned} \quad (4)$$

ここで、最後の 2 つの項はパラメータの正則化をおこなうための項であり、Gaussian prior[1] によるものである。これはパラメータの事後確率最大化 (MAP) に対応し、このような項を導入することで過学習を防ぐことができる。パラメータの最適化は L-BFGS[5] などを用いておこなうことができる。

構成するグラフは閉路を含むため、周辺確率の正確な計算が困難である。本稿では、周辺確率を近似的に求める手法として Tree-based Reparameterization (TRP) [13] を用いる。TRP は、閉路を含むグラフから構成できる全域木の集合  $\Upsilon = \{\mathcal{T}\}$  を列挙し、各全域木について Belief Propagation などの手法を用いて推論をおこない、周辺確率を更新していくという手続きを繰り返すことで近似的に周辺確率を求めるアルゴリズムである。

## 3 実験

### 3.1 訓練・評価用データ

2005 年 10 月 29 日時点の Wikipedia の記事から、ランダムに選択した日本語の記事約 2300 内のリスト (<LI>タグ) に含まれるアンカー (<A>タグ) に対して、開根らによって提案された階層的な固有表現分類である拡張固有表現階層 [8] の対応するクラスを手手で付与したデータを用いる。ここで、アンカーのリンク先の記事が存在しない場合についても、記事が存在する場合と同様にクラス付与をおこない分類対象とした。また、依存関係を考慮した分類の有効性を調査するため、Sibling, Cousin, Relative の関係にあるアンカーによって構成される連結グラフのノード数が 2 以下となるような事例については分類対象から除外した。その結果、全体で 16136 アンカーが分類対象となった。このうち、固有表現は 14285 である。

**Sibling**  $E_S = \{(v_i^T, v_j^T) | v_i^T, v_j^T \in V^T, d(v_i^T, v_j^T) = d(v_j^T, v_i^T) = 1, cpa(v_i^T, v_j^T) = 1, v_j^T = ch(pa(v_i^T, 1), k), v_i^T = ch(pa(v_j^T, 1), \max\{|l| < k\})\}$   
**Cousin**  $E_C = \{(v_i^T, v_j^T) | v_i^T, v_j^T \in V^T, d(v_i^T, v_j^T) = d(v_j^T, v_i^T) \geq 2, cpa(v_i^T, v_j^T) \geq 2, v_i^T = ch(pa(v_i^T, k), v_j^T) = ch(pa(v_j^T, k), pa(v_i^T, d(v_j^T, cpa(v_i^T, v_j^T)) - 1) = ch(pa(v_j^T, d(v_j^T, cpa(v_i^T, v_j^T))), k), pa(v_i^T, d(v_i^T, cpa(v_i^T, v_j^T)) - 1) = ch(pa(v_i^T, cpa(v_i^T, v_j^T)), \max\{|l| < k\})\}$   
**Relative**  $E_R = \{(v_i^T, v_j^T) | v_i^T, v_j^T \in V^T, d(v_i^T, v_j^T) = 1, d(v_j^T, cpa(v_i^T, v_j^T)) = 3, pa(v_j^T, 2) = ch(pa(v_j^T, 3), k), v_i^T = ch(pa(v_i^T, 1), \max\{|l| < k\})\}$

図 2: Sibling, Cousin, Relative の各クリークの定義.

素性	SVMs	CRFs
記事の定義文 (bag-of-words)	✓	✓ (V)
記事の見出し (全体)	✓	✓ (V)
記事の見出し (形態素)	✓	✓ (V)
記事のカテゴリ (全体)	✓	✓ (V)
記事のカテゴリ (形態素)	✓	✓ (V)
リンク元のアンカーテキスト (全体)	✓	✓ (V)
リンク元のアンカーテキスト (形態素)	✓	✓ (V)
アンカーテキストの親タグ	✓	✓ (V)
アンカー直前のヘッダのテキスト (全体)	✓	✓ (V)
アンカー直前のヘッダのテキスト (形態素)	✓	✓ (V)
ラベル間素性		✓ (S, C, R)
前ラベル	✓	

表 2: 分類に用いる素性. "✓" は該当する素性を与えたこと, CRFs の V はノードに対して, S は Sibling, C は Cousin, R は Relative クリークに対して素性を与えたことを意味している.

開根の拡張固有表現階層のクラス数は 200 以上であるが, 本稿ではラベル数を限定し, 拡張固有表現階層の深さ 2 までに限定したラベルセットを使用した. また, 訓練・テストデータに存在する数が極端に少ない固有表現クラスについては, 一つのクラスに統合した. その結果, 訓練・評価用データに含まれるクラス数は, 全体で 13 クラスとなった. 拡張固有表現階層のクラスと Wikipedia 記事数の対応を表 1 に示す.

### 3.2 実験方法・評価

本稿で提案した手法の有効性を示すために, ベースラインの手法として汎化性能に優れ, 文書分類など多くの分類問題において高い精度を示している Support Vector Machines (SVMs) [12] を使い, CRFs との精度を比較する. SVMs の多クラス問題への適用のため, one-versus-rest 法を用いる. CRFs, SVMs で用いた素性を表 2 に示す.

CRFs のグラフを構成するために定義した 3 種類のクリークのうち, どのクリークが精度向上に寄与するかを調査するため, Sibling(S), Cousin(C), Relative(R) の各クリークを含むか含まないかの全組合せについて実験をおこなう (SCR 全て, SC, SR, CR, S のみ, C のみ, R のみ, 個別に分類 (I とした) の 8 通り). 各設定によって構成されるグラフの例を図 3 に示す. 各クリークの辺を除いた結果, グラフが全体で非連結となるものについては, 部分連結グラフごとに分類をおこなう.

SVMs による識別では, アンカーを個別に分類する手法 (以後 I), SCR によって構成されるノード集合に対応するアンカー  $\mathcal{A} = \{a_1, a_2, \dots, a_N\}$  のうち, 出現順序の早いもの

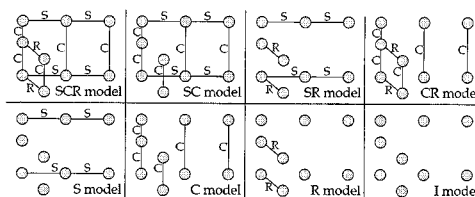


図 3: Sibling, Cousin, Relative の組合せによって構成されるグラフの例.

から順番に識別する手法 (以後 P) の 2 種類の手法を適用した. P ではアンカー  $a_j$  の識別に, アンカー  $a_{j-1}$  の識別結果を素性として用いた.

評価は Wikipedia データを 5 分割し, 訓練 4, テスト 1 の比率で交差検定によりおこなう. CRFs の学習および実行には, GRaphical Models in Mallet (GRMM) [9] を使い, パラメータ推定および最適解の導出には Tree-based Reparameterization (TRP) を, モデルの各素性に対する重みの事前分布には Gaussian Prior を使い, 分散は  $\sigma^2 = 10$  に設定した. SVMs の学習および実行には, TinySVM \*2 を使い, カーネルは線形カーネルを用いた. また, 形態素の素性を得るため, 形態素解析器として MeCab \*3 を用いた.

### 3.3 実験結果・考察

表 3 は, アンカーのリンク先に文書が存在するもの, しないものの双方を含めた Wikipedia データに対する, SVMs と CRFs の全体の分類結果 (F 値) を示している. 全体の傾向として, イベント名, 単位名など全体数の少ない固有表現については, CRFs と比較して SVMs の精度が高い傾向となっているが, 人名, 地名, 製品名など出現する頻度の高い固有表現については CRFs が SVMs を上回る結果となり, 全体として CRFs が高い精度を示す結果となった. これは, CRFs においてノード間の依存関係を捉えた分類をおこなったことで, 全体として良い結果が得られたものと考えられる.

CRFs の Sibling, Cousin, Relative の各クリークを含めた場合と含めなかった場合の全ての組合せの中では, Cousin と Relative を含む場合 (CR) に最も高い F 値が得られ, 以降 SC, SCR, C, SR, S, R, I の順となっている. これは, 個別に分類する場合 (I) と比較し, 依存関係を考慮した分類をおこなうことで精度向上が可能であることを示している.

\*2 <http://www.chasen.org/~taku/software/TinySVM/>

\*3 <http://mecab.sourceforge.net/>

NE CLASS	N	CRFs								SVMs	
		C	CR	I	R	S	SC	SCR	SR	I	P
人名	3315	.7419	.7429	.7453	.7458	.7507	.7533	<b>.7981</b>	.7515	.7383	.7386
時間・数値表現	2749	.9936	<b>.9944</b>	.9940	.9936	.9938	.9931	.9933	.9940	.9933	.9935
施設名	2449	.8546	.8541	.8540	.8516	.8500	.8530	.8495	.8495	.8504	<b>.8560</b>
製品名	1664	.7414	<b>.7540</b>	.7164	.7208	.7130	.7371	.7418	.7187	.7154	.7135
地名	1480	<b>.7265</b>	.7239	.6989	.7048	.6974	.7210	.7232	.7033	.7022	.7132
自然物名	1132	.3333	.3422	.3476	.3513	.3547	.3294	.3304	.3316	<b>.3670</b>	.3326
組織名	991	.7122	.7160	.7100	.7073	.7122	.6961	.5580	.7109	.7141	<b>.7180</b>
職業名	303	.9088	.9050	.9075	.9059	.9150	.9122	.9100	<b>.9186</b>	.9091	.9069
イベント名	121	.2740	.2345	.2533	.2667	.2800	.2740	.2759	.2667	.3418	<b>.3500</b>
称号名	42	.1702	.0889	.2800	.2800	<b>.3462</b>	.2083	.1277	<b>.3462</b>	.2593	.2642
名前・その他	24	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	<b>.0690</b>	.0000
単位名	15	.2353	.1250	.2353	.2353	.2353	.1250	.1250	.2353	<b>.3333</b>	.3158
ALL	14285	.7846	<b>.7862</b>	.7806	.7814	.7817	.7856	.7854	.7823	.7790	.7798

表 3: CRFs と SVMs の分類精度 (全体, F 値).

NE CLASS	N	CRFs								SVMs	
		C	CR	I	R	S	SC	SCR	SR	I	P
施設名	1080	.9363	.9369	.9343	.9351	.9273	<b>.9382</b>	.9347	.9324	.7602	.7789
人名	1068	.6709	.6732	.6480	.6492	.6507	.6741	<b>.6855</b>	.6441	.5485	.5572
自然物名	623	.0799	.0795	.0828	.0828	<b>.0858</b>	.0284	.0222	.0253	.0712	.0712
地名	552	.5306	.5303	.4631	.4758	.4540	<b>.5313</b>	.5236	.4593	.3157	.3121
製品名	359	.6894	<b>.6991</b>	.6089	.6133	.6169	.6839	.6851	.6034	.5867	.6066
組織名	174	.2120	.2353	.2342	.2192	.3108	.2648	.2132	.2996	<b>.3165</b>	.2932
イベント名	21	.1000	.0909	.1000	.1000	.0000	.0000	.0000	.0000	<b>.1739</b>	.1739
名前・その他	8	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
時間・数値表現	7	.8571	.8571	.6667	.6667	.8571	.8571	.8571	.8571	.5455	.6000
職業名	4	.5714	.5714	.0000	.0000	.5714	.5714	.5714	.5714	.4000	.4000
単位名	1	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
称号名	1	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
ALL	3898	.6469	<b>.6487</b>	.6231	.6258	.6245	.6463	.6440	.6182	.5278	.5386

表 4: CRFs と SVMs の分類精度 (リンク先なしのみ, F 値).

また、定義した 3 種類のクリークのうち、Cousin が最も精度向上に寄与している。これは、Cousin の関係にあるアンカーの組は出現頻度が高く、また他の関係と比較して同じカテゴリに属しやすいという強い依存関係があることから、精度に影響を与えているものと考えられる。

各固有表現クラス個別の分類精度に関して、まず人名については Sibling, Cousin を含むもの (SC, SCR) にて高い F 値が得られた。データ中の Sibling の関係にあるラベルの組で双方のアンカーのラベルが人名である事例は 4925 中 435、Cousin の関係にあるラベルの組で双方のアンカーのラベルが人名である事例は 13125 中 2557 であり、それぞれ高い頻度で出現していたため、ラベル間素性が有効に働き、高い精度で分類できたものと考えられる。製品名、地名については Cousin を含むもの (C, CR, SC, SCR) にて高い F 値が得られた。Cousin の関係にあるラベルの組で双方のアンカーのラベルが製品名である事例は 1072、地名である事例は 738 と高頻度で出現していたため、高い頻度で分類できたと考えられる。

また、表 4 は、表 3 の実験結果のうちリンク先が存在しないアンカーのみについての SVMs と CRFs の全体での分類結果 (F 値) である。リンク先が存在しない場合、見出し語、定義文、Wikipedia のカテゴリの情報が欠落するため分類が難しくなる。そのため、CRFs, SVMs の双方、F 値が大

幅に低下しているが、CRFs の精度低下の幅は SVMs と比較して小さく、SVMs との差が顕著に現れる結果となった。これは、CRFs においてノード間の依存関係を捉えた分類をおこなったことで、全体として良い結果が得られたものと考えられる。

また、より高精度な分類を期待して、[3] において提案されている手法で、クリークに対して特定の観測素性を結びつけるような素性の与え方 (すなわち、 $\Phi_{SCR} = \exp(\sum_k \lambda_k f_k(y_i, y_j, x_i))$ ,  $x_i \in \mathbf{x}$ ) を試みたが、良い結果は得られなかった。その理由として、提案モデルではクリークにおける素性の数  $|F|$  がラベルの組合せの数  $|F| = |\mathcal{L}| \times |\mathcal{L}|$  であるのに対し、 $f_k(y_i, y_j, x_i)$  の素性の数は、ラベルの全組合せの数と観測素性の数の積  $|F| = |\mathcal{L}| \times |\mathcal{L}| \times |\mathcal{F}_o|$  (観測素性) となる。ラベルの組によっては、データ中での出現数が僅かであるものも存在するため、観測素性を結びつけることで素性空間がスパースとなり、パフォーマンスの低下につながったものと考えられる。

本稿の手法により抽出した固有表現候補から固有表現辞書を構築する場合、少ない人手のコストでおこなえることが望ましい。そこで、CRFs の結果のうち一番良い結果が得られた CR model の実験結果について、モデルが出力した各ラベル割り当ての周辺確率  $p(y_j|x) = \sum_{y \setminus y_j} p(y|x)$  を特定の閾値  $\tau$  でフィルタリングをおこなった。図 4 は、

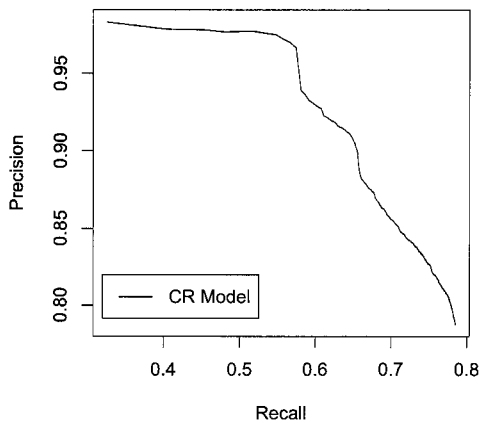


図 4: 分類精度の Precision-Recall 曲線 (CR model)

CRFs (CR model) の実験結果について、各ラベル割り当ての周辺確率を閾値  $\tau$  でフィルタリングした時の Precision と Recall の関係をプロットしたものである。曲線は、Recall 0.57 付近にピークがあり、その点における Precision はおよそ 0.97 である。これは、データ中に存在する固有表現全体のうち 5.7 割を、精度 97%、つまり 30 固有表現候補中誤りが 1 程度と非常に高い精度で獲得できることを示している。適当な閾値  $\tau$  を決め、 $p(y_i|x) > \tau$  であるような固有表現候補のみを手で確認することで、少ないコストで効率的に固有表現を獲得することができる。

#### 4 まとめ

本稿では、Web 上の多言語百科事典である Wikipedia から精度よく固有表現を獲得し分類する手法を提案した。具体的には、Wikipedia の文書の DOM 構造が持つ依存関係を、3 種類のクリークを定義することによってグラフ構造に反映させ、グラフを構成するそれらのクリークに対して依存関係を考慮したポテンシャル関数を導入することで相互の依存関係をとらえた上で全体に対する最適なラベル割り当てを求める CRFs のモデルを提案した。評価実験にて、SVMs と比較して高い精度で固有表現の分類が可能であることを示した。

今後の課題としては、付与する固有表現クラスの粒度の細分化が挙げられる。分類した固有表現を質問応答や関係抽出などの応用に用いることを考えた場合、固有表現クラスの粒度の細分化は非常に重要である。例えば、質問応答システムへの応用を考えた場合、固有表現が細分化されていることで回答候補を少数に絞ることが可能となり、回答の精度向上が期待できる。しかし、一般的に付与するラベル集合が大きい場合、各ラベルごとの事例数は少なくなるため、統計的手法による分類は難しくなる。その問題をどう解消するかは今後の課題である。

また、本稿で使用した固有表現分類である拡張固有表現階層は階層構造をなしているが、この階層構造を分類に利用することでさらなる精度向上が見込める。階層構造を考慮した分類手法としては、Max Margin Markov Network[10]

に基づき、階層構造と同様の木構造を持つグラフ (Markov Tree) 上の各ノードに対してラベルを割り当てることで階層的な分類をおこなう手法 [7] などがあるが、我々は、本稿で扱った依存関係とラベルの階層構造とを同時に扱うことが可能なモデルの設計に現在取り組んでいるところである。

#### 参考文献

- [1] Chen, S. F. and Rosenfeld, R.: A gaussian prior for smoothing maximum entropy models, Technical report, Carnegie Mellon University (1999).
- [2] Getoor, L., Segal, E., Taskar, B. and Koller, D.: Probabilistic Models of Text and Link Structure for Hypertext Classification, *IJCAI Workshop on Text Learning: Beyond Supervision, 2001*. (2001).
- [3] Ghamrawi, N. and McCallum, A.: Collective Multi-Label Classification, *Fourteenth Conference on Information and Knowledge Management CIKM, 2005* (2005).
- [4] Lafferty, J., McCallum, A. and Pereira, F.: Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data, *Proceedings of the 18th International Conference on Machine Learning* (2001).
- [5] Liu, D. C. and Nocedal, J.: The Limited Memory BFGS Methods for Large Scale Optimization, *Mathematical Programming* 45 (1989).
- [6] Lu, Q. and Getoor, L.: Link-based Text Classification, *Proceedings of the International Joint Conference on Artificial Intelligence* (2003).
- [7] Rousu, J., Saunders, C., Szedmak, S. and Shawe-Taylor, J.: Kernel-Based Learning of Hierarchical Multilabel Classification Models, *Journal of Machine Learning Research*, Vol. 7, pp. 1601–1626 (2006).
- [8] Sekine, S., Sudo, K. and Nobata, C.: Extended Named Entity Hierarchy, *Proceedings of the LREC 2002*. (2002).
- [9] Sutton, C.: GRMM: A Graphical Models Toolkit (2006). <http://mallet.cs.umass.edu>.
- [10] Taskar, B., Guestrin, C. and Koller, D.: Max-Margin Markov Networks, *Advances in Neural Information Processing Systems (NIPS 2003)*, Vancouver, Canada (2004). Winner of the Best Student Paper Award.
- [11] Taskar, B., Abbeel, P. and Koller, D.: Discriminative Probabilistic Models for Relational Data, *Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann (2002).
- [12] Vapnik, V.: *Statistical Learning Theory*, Wiley Interscience (1998).
- [13] Wainwright, M., Jaakkola, T. and Willsky, A.: Tree-based reparameterization framework for analysis of sum-product and related algorithms, *IEEE Transactions on Information Theory*, Vol. 45, No. 9, pp. 1120–1146 (2003).