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

2008－NL－186（13）
2008 / 7 / 18

リッチアノテーション：固有表現に焦点をあてた知識抽出の試み

菊井玄一郎† 松尾義博† 小林のぞみ† 平野徹† 浅野久子†

† NTT サイバースペース研究所 〒239-0847 神奈川県横須賀市光の丘 1-1
E-mail: †{kikui.genichiro,matsuo.yoshihiro,kobayashi.nozomi,hirano.tohru,asano.hisako}@lab.ntt.co.jp

あらまし 本稿ではテキスト中の固有表現に関する意味情報を抽出するシステムについて述べる．抽出する意味情報は大きく２種類であり，１つ目は固有表現の参照物，２つ目は固有表現間の意味的関係である．前者については周囲に出現する単語や固有表現，参照物候補の特性を用いて地名を緯度経度に，人名を Wikipedia のエントリーに対応づける手法を実現した．また，後者についてはテキスト中の２つの固有表現の意味的関係性の有無を Salient Reference List に基づく文脈的情報を素性として含む教師あり学習によって判別している．これらによりウェブの情報を表形式データベースのような構造知識として扱うことを可能にした．

キーワード 固有表現，情報抽出，意味解析，ウェブテキストマイニング

Rich Annotation: Knowledge Extraction focusing on Named Entities

Genichiro KIKUI†, Yoshihiro MATSUO†, Nozomi KOBAYASHI†,
Toru HIRANO†, and Hisako ASANO†

† NTT Cyber Space Laboratories,
Hikarino-oka 1-1, Yokosuka-shi, Kanagawa, 239-0847 Japan
E-mail: †{kikui.genichiro,matsuo.yoshihiro,kobayashi.nozomi,hirano.tohru,asano.hisako}@lab.ntt.co.jp

Abstract This paper describes a general purpose information extraction system. The system has two main functions. The first function associates a named entity (NE) expression with its "referent", which is defined in external databases such as Wikipedia and a geographic database. The second function extracts semantically related pairs of NEs. The system can find an NE pair across sentence boundaries by using a machine learning method that refers to contextual information. The paper also presents an example of extracted information in a structuralized format.

Key words named entity, information extraction, web text mining

1. はじめに

インターネット上に存在する膨大なテキストは広い分野をカバーしており，大規模な知識源と考えることができる．しかし，生のテキストをそのまま知識源として利用することは次のような理由で難しい．第一に，言語表現と意味（情報）との対応関係は１対１でなく，一つの表現が文脈に依存して別の意味に対応したり（多義性），表層上異なった言語表現がほぼ同じ意味に対応したりする（ゆらぎ・冗長性）という問題がある．第二に，テキストは単語の一次元列であり情報の構造（要素間の関係）が明示的に表現されていないという問題がある．

すなわち，インターネット上のテキスト集合を知識源として活用するためには，個々のテキストに含まれる情報を抽出して曖昧性や冗長性のない明確なセマンティックスを持つ構造化データに変換しなければならない．この分野では 90 年代に MUC(Message Understanding Conference) を中心に特定の分野に関する深い知識を意味フレームのインスタンス生成という形で抽出する研究が行なわれたが，提案された手法は分野ごとに人手で抽出規則を作成しなければならず，汎用性の点で大きな問題があった．その後，MUC の後期から最近の ACE(Automatic Content Extraction (注1)) では，複数の固有表現の参照物の同一性を判定したり，意味的関係を持つ固有表現の組を抽出したりといった汎用性の高い課題で研究が進められている．また，関根 [1] らは与えられたトピックに特徴的な固有表現(注2)や固有表現間の関係を事前の人手作業なしに抽出して構造化する手法を提案している．これらの研究において，

(注1)：http://projects.ldc.upenn.edu/ace
(注2)：IREX における固有表現の定義を拡張した「拡張固有表現」を用いている

固有表現の参照物同定は冒頭で述べた第一の課題，固有表現間の関係抽出は第二の課題の解決を目指していると考えられる．

我々の試みている情報抽出も冒頭の２つの課題の解決を目指しているという点でこれらの研究と同じ方向にある．まず，第一の課題に関してはテキスト中の固有表現に対して実世界における参照物（例えば外部データベースのエントリー）との対応づけを行なう．また，第二の課題に対してはテキストから固有表現間の意味関係を抽出して表形式データベースの形式に構造化する．いずれも特定のトピックに依存することなく，テキストの現象を広く正規化，構造化することで，より高次の意味処理やウェブマイニングを行なう際の基本データを生成することを目指している．この点でトピックに内在する情報の構造化 [1] 等の処理の前段階と考えられる．

以下では，まず，2. 節において知識抽出システムの概要を述べたあと，3. 節，および 4. 節においてそれぞれ固有表現の参照物同定と固有表現間の関係抽出について説明する．5. 節では抽出された知識の例を示す．

2. 全体構成

システムへの入力はウェブ等から得られたテキストである．これに対して，まず，html タグの除去などの前処理を行ない，次に，形態素解析 [2]，固有表現抽出 [3]，係り受け解析 [4] を適用する．この結果に対して「固有表現の参照物同定」および「固有表現間の意味関係抽出」をそれぞれ独立に適用する．(注3)．

ここで，固有表現抽出 [3] では形態素列を入力として，CRF(Conditional Random Fields) による系列ラベリングを用いて，IREX で規定された 8 種類（人名，地名，組織名等）の固有表現を抽出する．

固有表現の参照物同定では各固有表現に対して外部データベースにおける指示対象（参照物）を同定し，同定された指示対象の（当該データベースにおける）ID を付与 (annotate) する．

固有表現の意味関係抽出では意味的な関係を持つ二つの固有表現の組を見つけ，これらの固有表現 (mention) のＩＤ関係を示すラベルを加えた３つ組としてデータベースに登録する．たとえば，固有表現 1 と固有表現 2 が rel という関係を持つ場合には

(rel, 固有表現 1, 固有表現 2)

という形で登録する．

以下，参照物同定と意味的関係性抽出について説明する．

3. 固有表現の参照物同定

3.1 タスク

固有表現とは実世界において何らかの参照物が想定できる言語表現である．固有表現の参照物同定とはテキスト中の固有表現をこの参照物と対応づけるタスクである．ここで，実世界の存在をそのまま計算機に入れることはできないので，何らかの方法で**モデル化**し，このモデルにおけるデータ（記号や数値）を固有表現の参照物とせざるを得ない．実世界全体を統一的な枠組みでモデル化することは人工知能などの分野で様々な検討がなされてきたが，ここではそのような方法は取らず，固有表現のタイプごとに実用的な見地から必要最小限のシンプルなモデルを考えることにした．

具体的には，地名については緯度経度で示される物理的な位置，人名，組織名については人名や企業名，店舗名などのデータベースにおけるエントリーを参照先とした．また，時間，数量，日時については単位等を正規化した形式とした．いわゆる固有物 (artifact) は統一化が難しいので細分化して，たとえば，商品名であれば商品名データベースのエントリーに対応づけることとした．

3.2 解決すべき課題

固有表現の参照物同定は一般に次の二つのステップに分けられる．

候補生成　固有表現単独で見た場合の参照物の候補を生成する

多義解消　固有表現が複数の参照物候補を持つ場合に，周囲のコンテクストから尤もらしいものを選ぶ

前者の候補生成は別の見方をすると，各実体（参照先になりうるもの）がどのような名前（固有表現）で指され得るかを明らかにするという異表記獲得（抽出）の問題と言える．特にブログなどの UGC(User Generated Contents) を対象とした場合，ニックネームや省略形などが次々に出現するため，ウェブテキストを用いて収集することが必要である．また，地名などは住所がどのように省略されて出現するかを考慮する必要がある．

後者の多義解消はいわゆる語義の曖昧性解消のタスクと基本的には共通であり，次のような手がかりを用いて最尤の候補を選択する．

（1）　固有表現の出現するテキスト側の文脈における妥当性

（2）　参照物（先）の組み合わせの妥当性

これらを使った多義解消はたとえば Naïve Bayes をはじめとする教師あり学習の枠組みで解くことができるが，固有表現は異なり数やバリエーションが多く学習データを準備するのが容易ではないという問題がある．従って，外部の資源をうまく活用して教師データを作成することや，ヒューリスティックな尺度によって候補のランキングを行うことが必要である．

以下では地名と人名についてこれらの課題の具体的な解決法について説明する．

3.3 地　　名

文書中の地名は「日本橋」のように都道府県や市区町村名が省略され，その実世界での位置を一意に決定できないことが多い．ここではこのような地名から位置情報（完全な住所，および，緯度・経度の情報）を得る手法について述べる．

ここで，文書中の地名に対して実世界での位置を特定する従来手法の多くは次の３ステップから構成されている [5]〜[8]．

Step1　入力文書中の各地名に対して実世界における位置の候

(注3)：実際は参照物同定の過程で固有表現抽出結果の修正が一部可能であるため，将来的には参照物同定を行ったのちに関係抽出を行うと全体性能が向上すると思われる．

補を住所 DB から取得する.

Step2 (入力文書中に2つ以上の地名が存在する場合) 各地名に対する位置候補の組み合わせのうち, "距離" が最短となるものを選ぶ. "距離" としては, 緯度・経度に基づく地理的距離や住所の階層距離などが用いられ, 3つ以上の地名が存在する場合, 例えば, 2つの組み合わせの中で距離最短なものをまず選び, これらを起点として残りの地名に対する位置を決めるといった手法が取られる. なお, 最短距離が一定の閾値を越える場合には候補選択を行わない.

Step3 前記の処理で候補が一意に絞られていない地名については, "有名度" (各候補がどの程度多くの人に想起されうるかを示したもの) が最大となる候補を選ぶ. "有名度" として, 住所階層や人口数などの情報から算出されたスコアが用いられている.

この3つのステップにおいて Step1 が前述の候補生成, Step 2および3が多義解消に対応する.

3.3.1 提案手法

提案手法は既存手法と比べて, 多義解消における, 有名度の算出方法, および, 距離と有名度の組み合わせ方法が異なる.

a) 有名度の算出方法

従来手法の住所階層や人口数を用いた有名度は, 地名の全ての候補が同スコアになり, 機能しないことがある. 例えば, 住所階層を用いた有名度は, 上位階層の候補の方が高スコアだが, 階層ごとにスコアが定まるため, 同じ階層の候補は同スコアになる. また, 人口数を用いた有名度は, 人口数の多い候補の方が高スコアだが, 一般に利用可能な人口数情報は都道府県と市区町村に限られており, 人口数情報の不明な候補は同スコアになる. 例えば,「日本橋」の候補である「大阪府大阪市中央区日本橋」と「東京都中央区日本橋」は, 人口数情報がわからないため, 同スコアになる.

そこで, 我々は,「有名な場所=店の多い場所」と考え, 各住所候補に対して店舗DBを用いて店舗数を計算し, これを有名度のスコアとする.

例えば, ある店舗 DB を用いると,「大阪府大阪市中央区日本橋」と「東京都中央区日本橋」にはそれぞれ 60 店舗および 215 店舗が存在し, 従来手法では同スコアだった候補に対しても有名度よる評価ができる.

b) 距離と有名度の組み合わせ方法

従来手法では, 距離に基づく処理の後に有名度に基づく処理を行なうため, 常に距離に基づく尺度が優先される. このことにより, 非常に多くの人に想起されうる有名な候補が正しい場合でも, 他の候補を選んでしまうことがある.

そこで, 距離に基づく処理を常に優先するのではなく, 有名度が他の候補に比べ突出している場合には, 有名度に基づく処理を優先することによりこの問題を回避する. 具体的には下記の **Step a**,**Step b** を先に述べた従来手法の Step1 の直後に挿入する.

Step a 各地名について有名度が突出している参照先 (位置情報) の候補があれば この候補を参照先とする. 有名度が突出している候補とは, 有名度が最大であり, かつ, その値を (当該地名に対する) 全候補の有名度の和で割った値がある閾値より大きいものを言う.

Step b 参照先が特定されていない地名 W_j について, 同一文書内に存在し, 既に参照先が決まっている他の地名 W_h があれば, その参照先 G_h との距離が最短となる候補を地名 W_j の実世界での位置 G_j とする. もし, W_h が複数存在する場合は (最短距離の候補を選んだ場合の) 距離が最短のものを利用する. なお, 最短距離が一定の距離以上の W_j については候補の選択を行なわない.

3.3.2 評価

評価実験には, gooブログ(注4)でカテゴリに各都道府県名もしくは「食べ歩き」と設定された文書に, 人手で地名とその実世界での位置を付与した 1,908 文書を用いた. この中には 3,872 個の地名が存在し, その実世界での位置候補は平均 17.34 個である. なお, 候補が 1 つのみの地名は 1,284 個あり, 最も候補が多かったのは「上野」の 355 個である.

地名の実世界での位置候補を取得するために, 国土交通省の「街区レベル位置参照情報」(13,045,497 件) を住所 DB として,「国土数値情報 (鉄道データ)」(8,918 件) を駅 DB として用いた. さらに, 提案した有名度算出手法で用いる店舗 DB には goo 地域(注5)のデータ (約 250,000 件) を利用した.

また, 各ステップでの閾値として, Step a の有名度では 0.9, Step b の地理的距離 (十進経緯度) では 0.2 を与えた. 閾値はデベロップメントセットを別途用意して決定した.

評価実験の結果, 提案手法は 92.7%の精度で地名に対する実世界の位置を決定できた. これは先に述べた従来手法の精度 67.5%に比べて大幅な改善であるといえる.

3.4 人名

人名の参照物同定とは, 1 つのエントリが実世界における特定の人物に対応するような「人物データベース」を想定して, テキスト中の固有表現をこのデータベースのいずれかのエントリーに対応づけるタスクである (対応先がない場合は「対応なし」とする). 人物は文書中において姓や名, ニックネームなど様々な表層形で出現する. また, 同姓同名の人物の存在を考えるとフルネームで記述されていても人物を特定できるとは限らない. 従って, 人名についても, 先に述べた「候補生成=異表記獲得」と「多義解消」の二つの課題を解決する必要がある.

以下では実体データベースとして wikipedia(注6)を想定した参照物同定の実現方法について述べる. wikipedia はユーザが自由にエントリを作成・編集できるため, 歴史上の人物をはじめ, 通常の辞書にはない実在の有名人のエントリも多く存在する. また, wikipedia のエントリはタイトル名と一対一対応にあり, 同姓同名の人物の場合でも個別のエントリとして曖昧性解消された状態になっているため (例えば, 電電太郎→電電太郎 (政治家), 電電太郎 (サッカー選手)), 他のデータベースへの対応付けも容易であり, 我々の目的にかなうデータベースといえる.

(注4) : http://blog.goo.ne.jp/
(注5) : http://local.goo.ne.jp/
(注6) : http://ja.wikipedia.org/

3.4.1 出現表記の獲得

コーパスに出現しうる異表記についてはウェブを用いて獲得する研究が行なわれている ([9], [10], [11] など). また, 人名に特化した研究として「○○こと××」のようなパターンを用いる方法が提案されている [12].

我々も現状は単純なパターンマッチの方法を使って wikipedia から獲得する手法を実装している. wikipedia の各ページを見ると, 人物の場合の多くは「安藤 美姫 (...) は, 日本の女性フィギュアスケート選手.」のように, エントリ名の姓 名に空白が入った状態で書かれている. この情報を利用し, 名字と名前をその人物の出現表記として収集した. また,「愛称は〜」などのパタンを用いて人物の愛称を取得した. なお, 今後は高橋ら [13] の手法を利用して獲得範囲 (再現率) を拡大する予定である.

3.4.2 多義解消

人名の参照物の多義解消については, 同姓同名の人物を対象とした, 関根らの WePS タスク [14], 小野らの名寄せシステム [15] など, 人名の出現する文書の集合を同じ人物が同一のクラスになるようにクラスタリングするというアプローチがメインである. これらに対して, 我々の目的は文書中に出現する人名がデータベース中のどの人物 (エントリー) と対応するかを同定することであるから, クラスタリングというより, 以前より研究されているいわゆる語義の曖昧性解消の問題に近い.

ここでは通常の語義の曖昧性解消と同様に, 出現表記の文脈を現す特徴語集合 X と, 各人物候補 i に対する特徴語集合 Y_i との類似度を次の式により求めて類似度の一番高い候補 (=人物) を選ぶ. なお, 類似度が閾値より低ければ参照先はないものとする.

$sim(X, Y_i) = \sum_{x \in X \cap Y_i} w_i(x)$

ただし, $w_i(x)$ は i における特徴語 x の重みとする.

ここで, 入力テキスト側の人名に対する特徴語集合 X としてはその人名の周辺に出現する一般名詞と固有表現 (人名, 組織名, 固有物名) を用いる. このとき, 固有表現はその人物を特徴づける語として一般名詞よりも強い制約になると考えられるため, 文書中の出現位置に限らず全て特徴語として使用する. 一方, 一般名詞は固有表現ほど強い制約にはならない場合が多いため, 出現した人名の近く (例えば一文前まで) に出現した名詞のみを特徴語として抽出する.

参照先候補 (=人物) に対する特徴語集合 Y_i についても同様に, wikipedia の各人物のページを形態素解析し, 名詞と固有表現 (人名, 組織名, 固有物名) を抽出して特徴語とした. 特徴語の重み付けは tfidf(下記の式) で与えた.

$tfidf(PSN, t) = tf(t) \times log(N/pf(t))$

ここで, tf(t): 単語 t の人物 (PSN) での頻度, N: 人物の総数, df(t): 単語 t が出現した人物ページの数である.

4. 意味的関係の抽出

固有表現の意味関係抽出とはテキスト中で意味的な関係を持つ二つの固有表現の組を見つけることである. ここで,「意味的関係」を広く捉えると同一テキスト中に出現する固有表現の組は (文脈を共有していることから) 全て何らかの意味的関係にあるともいえる. しかし, 本研究では同一テキスト中に出現する二つの固有表現が次のいずれかである場合に限定する.

- 一方が他方の属性値になっている場合
- 二つの固有表現が 同一意味フレームの直接の要素 (埋め込みは除外する) である場合

固有表現の意味関係抽出を我々は次の 2 つのステップに分けて行なう.

(1) テキストから何らかの意味的関係を有する固有表現の組を取り出す

(2) 取り出された組に対してこれらがどういう関係にあるかを推定する

前者についてはテキストから全ての固有表現の組を候補として取り出し, これらの間に関係性があるかどうかを判定する (関係性判定) という手法を取る. 一般に遠く離れた二つの固有表現は直接の意味的な関係を持たないことも多いが, たとえば, 冒頭に 1 回主題として出現した固有表現がずっと後の文の固有表現と関係性を持つ現象など長距離の関係もあることからここでは上記のような戦略を取る.

後者については「二つの固有表現が同一単位文に出現する場合に係り先述語を関係表現とする」などのヒューリスティックルールで暫定的に実装しているが, 本格的な手法は現在研究中であり本稿では扱わない. なお ACE の Relation Type のような非常に一般的なラベルであれば二つの固有表現のタイプの組み合わせから決定できる (例えば「人名-人名」で「person-social」など).

4.1 固有表現間の意味的関係性判定

固有表現間の関係性判定の従来研究は, 文構造を素性として用いた教師ありの機械学習によるものが多い [16]〜[19]. 例えば, Kambhatla らの研究 [18] では, 二つの固有表現の関係の有無を判断するのに, 係り受け木における二つの固有表現の最短パスを素性として利用した手法を提案している.

しかし, 我々の調査によれば実データ中に存在する直接的な意味的関係のある固有表現の組みのうち, 異なる文に出現する場合は全体の約 35 %を占め, 従来研究のように統語構造などの文に閉じた素性だけを用いた手法では不十分である.

そこで, 二つの固有表現が異なる文に出現する場合に有用だと考えられる文脈情報などの素性を併用することにした. 以下ではこの手法について説明する(注7).

4.1.1 基本的な考え方

従来手法と同じく我々の手法も教師ありの学習型分類技術を用いる. 訓練データはテキスト中において意味的関連性のある固有表現間に正解フラグを付与したものである. また, 分類アルゴリズムは構造を持つ素性を許す BACT [21] (注8) を使用し, 次のような素性を用いた

NE 判定対象の 2 つの固有表現そのものに関する素性

DEP 判定対象の固有表現の間の, 係り受け構造における最短

(注7) : 詳しくは文献 [20] を参照されたい.

(注8) : http://chasen.org/~taku/software/bact/

パス.

WORD　判定対象の固有表現の間に出現する単語や品詞

CT1　先行する固有表現 (NE1) が後続する固表有現 (NE2) が出現する場所において文脈的に Salient な要素であるかどうか

CT2　後続する固有表現 (NE2) が出現する場所における Salient Reference List(SRL) [22] の内容

これらのうち，我々が新たに提案したのが次に説明する CT1 と CT2 という二つの文脈的素性である．なお，WORD という素性は文境界を考慮せずに求めるため，文単位を超えた「文脈的」素性であるがここでは文脈的素性には含めない．

4.1.2　文脈的顕現性に基づく素性

文境界を越えて出現する二つの固有表現が関係性を持つとき，後述する固有表現 (NE2) の出現位置（あるいは出現する文）において，先行する固有表現 (NE1) は参照されやすいと考えられる．NE2 の出現する文において NE1 がゼロ代名詞になっていると考えても良い．このことから，文境界を越えた二つの固有表現の関係性の判定は，ゼロ代名詞が全て補完できれば文内の処理で行える可能性がある．しかしながら，任意格要素まで全て考慮して補完することは難しい上に無駄が多い．そこで本研究ではある文において，先行して出現する名詞句のうちのどれが文脈的に影響を持っているか，という顕現性 (salience) の情報は使うが，明示的な省略補完は行わないことにした．

具体的には，テキスト中のある位置において文脈的に Salient な順に名詞句をリストした Salient Reference List から得られる情報を判別のための素性として用いる．素性は２つであり，１つ目の素性 (CT1) は NE2 が出現した位置における SRL の第１位の要素と NE1 が一致したら True そうでなければ False となる２値の素性であり，もう一つの素性 (CT2) は前記 SRL そのものである．CT1 は CT2 と NE の情報があれば計算できるという点で冗長な情報であるが，判別において重要であることが明らかなため追加している．

4.1.3　評　　価

テキスト中の固有表現の組に人手で意味的関係の有無を判定した日本語の新聞記事 1,400 記事とブログ 4,800 記事の計 6,200 記事を用意した，なお，判定対象の固有表現の組は次の通りであり総数は 537,411 このうち意味的関係ありと判定されたものは 24,329 個であった．

［人名⇔地名］，［人名⇔組織名］，［組織名⇔地名］，［人名⇔人名］，［組織名⇔組織名］，［地名⇔地名］

このデータを用いて５分割交差検定により本手法を評価したところ，精度 0.804，再現率 0.650 となり，文脈的素性を含まない従来手法と比べて精度が約 0.113，再現率が約 0.142 向上することが確認できた(注9)．

5.　抽出サンプル

本稿で述べた手法によって抽出された「知識」は表形式データベースの形で格納されている．具体的には，各レコード（行）が意味関係を持つ二つの固有表現間の組（のテキストにおける出現）に対応し，フィールド（カラム，列）は，各固有表現の出現形，固有表現タイプ（人名，地名，など），参照先 ID，固有表現間の意味関係を表す言語表現（関係表現），抽出元のテキストＩＤなどとなっている．

たとえば，次のようなテキストからは表１のような　情報が抽出される．

「松坂が池袋で目撃された．ボストンから...」

なお，「関係表現」は４節で述べたように，いくつかのヒューリスティックルールによって暫定的に求めている．

以上のデータベースを用いて検索を行なったものを図１　に示す．この例は NE1 のタイプとして人名，NE2 のタイプとして地名，関係表現として「コンサート」と前方一致するもの，という制約によって検索したものである．この結果において固有表現は参照物によって「名寄せ」されている．このように，一定の関係にある固有表現を参照物によって統一化して一覧することが可能になるほか，たとえば，アーティストのデータベースと組み合わせるなど様々な利用が可能となる．

6.　おわりに

本稿ではテキストから固有表現に関する知識を抽出する試みについて述べた．実装した知識抽出システムの機能としては，１）固有表現（人名，地名，数量，日時）に対してその参照物を同定すること，および，２）テキスト中で意味的に関係のある固有表現の組を抽出することの２つである．

残された課題として，参照物同定の範囲を固有物名 (artifact) などに広げること，固有表現間の関係に対するラベル付けを行うことなどがある．前者に関しては，たとえば「VAIO TYPE-G」を商品名データベースのどこに対応づけたらよいのかなど，固有表現のサブタイプによって参照先の確定が難しい場　合があることから，参照先をどのようにモデル化するかというオントロジー的な検討を含めて行う必要がある．また後者については二つの固有表現の間の「述語」を見つけ，さらにそれらを同値類に分類するということが必要である．この方向については [1], [23] などの手法が参考になると考えている．

(注9)：詳しい評価結果は [20] に記載している

文　　献

[1] 関根：“オンデマンド情報抽出”，言語処理学会　第 14 回年次大会，pp. 927–930 (2008).

[2] 渕，松岡，高木：“保守性を考慮した形態素解析システム”，情報処理学会自然言語処理研究会報告，**97**, 4, pp. 59–66 (1997).

[3] 齋藤，今村，鈴木：“CRF を用いたブログからの固有表現抽出”，言語処理学会　第 13 回年次大会 (2007).

[4] 今村：“系列ラベリングによる準話し言葉の日本語係り受け解析”，言語処理学会　第 13 回年次大会，pp. 518–521 (2007).

[5] H. Li, R. K. Srihari, C. Niu and W. Li: “InfoXtract location normalization: a hybrid approach to geographic references in information extraction”, In Proceedings of the HLT-NAACL 2003 workshop Analysis of geographic references, pp. 39–44 (2003).

[6] Y. Li, A. Moffat, N. Stokes and L. Cavedon: “Exploring probabilistic toponym resolution for geographical information retrieval”, Workshop on Geographic Information Retrieval (2006).

表 1　データベースのレコードの例

内容	NE1:出現形	NE1:タイプ	NE1:ID	NE2:出現形	NE2:タイプ	ID	関係表現	TEXTID
値	松坂	PSN	松坂大輔-001	池袋	LOC	東京都豊島区池袋	目撃された	blog-20070102789

図 1　抽出サンプル

[7] E. Rauch, M. Bukatin and K. Baker: "A confidence-based framework for disambiguating geographic terms", In Proceedings of the HLT-NAACL 2003 workshop Analysis of geographic references, pp. 50–54 (2003).

[8] D. A. Smith and G. Crane: "Disambiguating geographic names in a historical digital library", In Proceedings of the 5th European Conference on Research and Advanced Technology for Digital Libraries, pp. 127–136 (2001).

[9] 関, 嶋田, 遠藤：“表の構造を利用した類義語抽出”, 言語処理学会　第 11 回年次大会, pp. C1–6 (2005).

[10] 本間, D. Bollegala, 松尾：“語の共起ネットワークを用いたエンティティの別名抽出”, 人工知能学会　第 22 回全国大会 (2008).

[11] 村山, 奥村：“Noisy-channel model を用いた略語自動生成”, 言語処理学会　第 12 回年次大会, pp. 763–766 (2006).

[12] 外間, 北川：“Web データを用いた人物の呼称抽出”, DBSJ Letters, **5**, 2, pp. 49–52 (2006).

[13] 高橋, 浅野, 松尾, 菊井：“単語正規化による固有表現の同義性判定”, 言語処理学会　第 14 回年次大会, pp. 821–824 (2008).

[14] 関根：“Web 検索における人名の曖昧性解消技術の動向”, 情報処理, **49**, 5, pp. 573–578 (2008).

[15] 小野, 佐藤, 吉田, 中川：“重要語抽出を用いた web 文書上の同姓同名の曖昧さ解消”, 第 19 回データ工学ワークショップ (DEWS) (2008).

[16] E. Agichtein and L. Gravano: "Snowball: Extracting relations from large plain-text collections", 5th ACM International Conference on Digital Libraries, pp. 85–94 (2000).

[17] A. Culotta and J. Sorensen: "Dependency tree kernels for relation extraction", Annual Meeting of Association of Computational Linguistics, pp. 423–429 (2004).

[18] N. Kambhatla: "Combining lexical, syntactic, and semantic features with maximum entropy models for information extraction", Annual Meeting of Association of Computational Linguistics, pp. 178–181 (2004).

[19] D. Zelenko, C. Aone, and A. Richardella: "Kernel methods for relation extraction", Journal of Machine Learning Research, pp. 3:1083–1106 (2003).

[20] 平野, 松尾, 菊井：“文脈的素性を用いた固有表現間の関係性判定”, 自然言語処理, p. to appear (2008).

[21] 工藤, 松本：“半構造化テキストの分類のためのブースティングアルゴリズム”, 情報処理学会論文誌, **45**, 9, pp. 2146–2156 (2004).

[22] S. Nariyama: "Grammar for ellipsis resolution in japanese", 9th International Conference on Theoretical and Methodological Isssues in Machine Translation, pp. 135–145 (2002).

[23] Y. Shinyama and S. Sekine: "Preemptive information extraction using unrestricted relation discovery", Human Lanuage Technology Conference of the North American Chapter of the ACL, pp. 304–311 (2006).