共起語を手がかりとした 固有表現とデータベースレコードの対応付け

小林 のぞみ　　松尾 義博　　菊井 玄一郎
日本電信電話株式会社 NTT サイバースペース研究所
{kobayashi.nozomi,matsuo.yoshihiro,kikui.genichiro}@lab.ntt.co.jp

1 はじめに

テキストから抽出した情報を集約・整理する上で，①ある実体を指す表現が複数あるという表記ゆれの問題，②テキスト中の表記が文脈によって指す実体が異なるという曖昧性の問題は重要な課題である．たとえば，首相の「麻生太郎」に関する情報を集約する場合，同姓同名の別人物の情報は除外し，別の表記（たとえば「麻生」）のうち，麻生太郎首相を指す情報のみを集約する必要がある．

テキスト中の固有表現が指す実体の曖昧性を解消する研究では，与えられた文書集合を同じ実体を指す表現ごとにクラスタリングするという問題設定が多い（[1, 5],etc）．これに対して我々は，2 節で述べるように，テキストに出現した各固有表現（以下，**出現表記**）をデータベースのレコードに対応づけるという，語義曖昧性解消に近い問題設定とする．これにより，blog に書かれた商品や店舗などと外部のデータベースを連携でき，さまざまな Web サービスでの活用が期待できる．

以下，本稿ではまずタスク設定を述べ，3 節で関連研究と解くべき課題を述べる．4 節で対応付け手法を説明し，5 節で評価結果を示し，最後に 6 節でまとめる．

2 タスク設定

我々は「実体＝外部データベースのレコード」と考え,「入力文書中の固有表現を与えられたデータベース中の対応するレコードと対応づける」ことを目指す．図 1 の例では,「麻生」を「麻生太郎（ID PSN-0924）」に,「秋葉原」を「秋葉原（緯度 35.698，経度 139.77）」にそれぞれ対応づける．

ここで，実体データベース (以下，実体 DB) は外部から与えられるものとする．ただし，実体 DB は実体の名称（以下，**登録名称**) とその ID に相当する情報を必ず持つとし，たとえば店舗であれば，店舗の名前と電話番号もしくは住所が含まれているとする．基本的には，市販もしくは公開されているデータベース，たとえば住所であれば国土交通省の「街区レベル位置参照情報」[1]，店舗であればタウンページ [2] などを想定している．また，近年では Semantic Web の分野で”Linked data”という，インスタンスを URI で記述する試みが注目されており [2]，これらの研究で得られた成果も実体 DB として今後使用できる可能性がある．

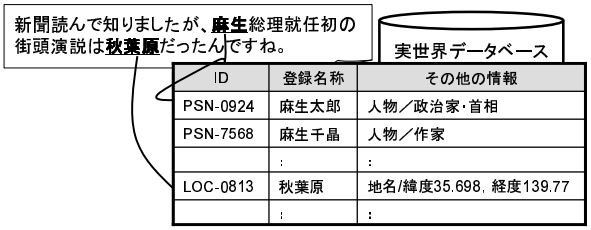

図 1: 指示対象同定問題のタスク設定

3 関連研究

今回の問題設定は，あらかじめ実体の集合が与えられた上で，テキスト中の固有表現がどの実体に対応するかを同定する問題であり，古くから取り組まれてきた語義曖昧性解消の問題設定と非常に近い．語義曖昧性解消では，多義語に対して共起語を手がかりに文脈の類似度を測り語義を決定するアプローチが取られている．我々もこの手法と同様に，共起語を使用した類似度ベースの手法でアプローチする．ただし，語義曖昧性解消では語義を広く網羅した辞書がすでに電子的に存在するのに対し，実体全てを網羅した DB を仮定することは難しい．そのため，対応先が実体 DB にない場合にいかに棄却するかが一つの課題である．

また，各語義がどういう文脈で使用されるかについては，あらかじめ語義タグつきのコーパスを作成し，教師あり学習などにより手がかりとなる共起語を自動抽出する手法がよく使われているが（[4] など），実体 DB に登録された実体全てを網羅したコーパスを作成することは現実的ではない．そのため人手を介さずにどう

[1] http://nlftp.mlit.go.jp/isj/
[2] http://www.nttbis.co.jp/townpage/townpage_top.shtml

共起語を獲得するかが課題となる．

問題設定が非常に近い研究として Cucerzan [3] がある．彼は出現表記をオンラインの事典である wikipedia [3] の各ページと対応づける手法を提案している．この手法では，wikipedia からあらかじめ実体とその実体を指し得る表現を獲得しているが，wikipedia が仮定できない商品や店舗ではその表現をあらかじめ得ておくことはできない．また，この手法は wikipedia に特化したカテゴリの情報などを取り入れており，wikipedia がカバーしていない分野での適用は難しい．

4 固有表現と DB レコードの対応付け手法

3 節で述べた課題に対して，本稿では，手がかりとなる共起語は事典や Web 文書から自動的に獲得し，実体 DB の網羅性については，選択された実体候補を解とするか否かの判定を導入することで解決を試みる．対応付け手法は以下の 3 ステップからなる．

1. 各出現表記に対し，実体の候補集合を作成する
2. 候補の中からもっともらしい実体候補を選択する
3. 選択された実体候補を解とするか解なしとするか判定する

以下，それぞれの処理について詳しく述べる．

4.1 候補生成

実体を言及する表現には，その実体の正式名称（たとえば「株式会社 エヌ・ティ・ティ・ドコモ」）のほかにもさまざまな種類があるが，多くの場合は愛称や通称（たとえば「NTT ドコモ」）か，英語，読みなどの表記ゆれ（たとえば「docomo」），もしくは名称の一部（たとえば「ドコモ」）であると考えることができる．

愛称や読みの表記ゆれは，既存手法（たとえば [6]）を用いてあらかじめ辞書として獲得し，辞書引きで候補を生成する．名称の一部に対しては出現表記を部分文字列として持つ登録名称を検索して候補とする．

4.2 曖昧性解消

曖昧性解消処理では，まず文章内に候補のいずれかと一意に決定できる手がかり（たとえば，店舗であれば電話番号）が存在すればその候補を解として出力し，そうでなければ共起語に基づいて類似度を測り，もっとも類似度の高い候補を選択する．

4.2.1 出現表記に対する文脈の獲得

文書中の各出現表記に対し共起語を獲得して文脈情報とする．共起語には一般名詞（形態素解析器が未知語と出力する語も含む），固有表現を使用する．

[3] http://ja.wikipedia.org/

文脈とする範囲は，固有表現は広く文章全体，一般の名詞（未知語含む）はウィンドウ内とする．これは，固有表現がトピック的に使われることが多く広い範囲に影響すると考えられるのに対して，一般名詞はローカルに影響することが多いと考えたためである．実際，予備実験で文書全体から獲得するよりもよい結果が得られたため，今回は固有表現は文章全体から抽出し，一般名詞はウィンドウ幅内に存在する場合のみ文脈として使用した．

4.2.2 実体の共起語の獲得

実体の共起語は，wikipedia のような実体について記述された事典や Web ページから獲得する．たとえば，店舗であれば電話番号をクエリとして各店舗について書かれた Web ページを獲得可能であり，人物や企業であれば wikipedia からそれぞれに関する記述を得ることが可能である．

語の重み付けには「多くの実体で共通して使用される語は弁別能力が低い」という考えに基づき，以下に示すような一実体を一文書と考えた idf に相当する重みを使用した．

$$ief(t) = log(N/ef(t))$$

ここで，t は単語 t を，$ef(t)$ は単語 t がいくつの実体で出現したか，N は実体の総数をそれぞれ表すとする．得られた語のうち ief の低い語は除外して各実体の特徴語とした．

4.2.3 類似度の計算

上記の手法で生成された二つのベクトルの類似度を計算する．ベクトル間の類似度を測る尺度にはさまざまなものが考えられるが，ここではよく使用される cosine 類似度を使用する．

4.3 信頼度による判定

ここでは，システムが選択した候補に対し，そのまま解として出力すべきか対応先が実体 DB にないと返すべきかを，信頼度がある閾値以上か否かで判定する．各候補 c の信頼度 $confidence$ は以下の式で求める．

$$confidence(c) = strsim(m, e) \times support(c)$$

ここで，m は出現表記，e は登録名称である．

第一項目の $strsim(m, e)$ は表記の一致率である．今回の手法では，候補は出現表記を含む登録名称であるため，出現表記よりも長い登録名称も候補として出力される．たとえば先ほどの「ふじの」という出現表記に対し,「ふじの」が候補として得られるが「ふじの里」という店舗名も候補となる．この場合に「ふじの」のほうが後者の店舗よりもより信頼できると考えられる．

この考えに基づき，下記に示すように編集距離に基づいて表記の一致率を求める．

$$strsim(m,e) \quad = \quad 1 - \frac{EditDistance(m,e)}{length(e)}$$

ただし，辞書引きにより得られた候補の場合，表記一致率は 1 とする．

第二項目として,「共通して出現した共起語の重みの和」を考える．これは，その候補であると支持した語が多ければ信頼できるという考えに基づくものである．その式は以下になる．

$$support(c) = \sum_{t \in X \cap C} tf(t) \times ief(t)$$

X は出現表記の共起語，C は各候補の共起語ベクトルであり，$tf(t)$ は単語 t の文脈での出現頻度，$ief(t)$ は実体の共起語の重みである．

5 評価・考察

導入した信頼度の有効性と，問題がどの程度解けるかを確認するために店舗分野を対象に評価実験を行った．

5.1 必要な資源

実体 DB，実体の共起語についてその作成法を述べる．

データベース 店舗 DB としてタウンページ[3]を採用した．今回使用したタウンページには，関東と関西の飲食店約 18 万件が収録されている．

実体の共起語の作成 実体を特徴づける共起語を獲得するために，blog 記事 約 4.3 億文書を対象に上記タウンページ中の電話番号を含むページを収集した．その結果，18 万店舗のうち 2 割にあたる 32,000 店舗に関する blog 記事を獲得できた[4]．獲得した各 blog 記事から 4.2.2 で述べた方法で共起語を抽出し，重み付けした．

5.2 評価データ

blog 約 2000 記事を対象に，まず店舗にタグを付与し，その後店舗の電話番号を検索エンジンを使って調査して ID として付与する作業を行った．評価データの一例を以下に示す．

> 山下公園に車を停めて、東門近くの ⟨shop ID=”045-641-0779”⟩ 謝甜記 ⟨/shop⟩ に行くと「準備中」... 諦めて ⟨shop ID=”?”⟩ 山東飯店 ⟨/shop⟩ にいくと、こちらは行列ができている。

タグが付与された店舗数は 3,291 で，うち 911 がデータベース中に対応先がある店舗となっており，被覆率は 0.27 と低い．ただしこれは今回使用した店舗 DB が限定された地域であるためで，関東・関西の店舗に絞った場合の被覆率は 0.63 であった．

5.3 評価

評価尺度として精度と再現率を使用する．

$$\text{精度} \quad = \quad \frac{\text{DB レコードに正しく対応付けできた数}}{\text{システムが DB レコードと対応付けた数}},$$

$$\text{再現率} \quad = \quad \frac{\text{DB レコードに正しく対応付けできた数}}{\text{DB に対応先がある店舗の数}}$$

システムの出力を信頼度の高い順にソートして描いた精度－再現率曲線 (confidence) を図 2 に示す．これにより，信頼度の高い場合にどの程度の精度と再現率が得られるかを見た．グラフの左端は，ある出現表記の候補に，文章中に出現した電話番号を ID として持つ候補がいればそれを解とした場合の結果であり，再現率 0.1 で精度は 1 であった．

4.1 で述べた方法で候補を取得した場合の候補の被覆率は約 6 割でありこの値が再現率の上限値になる．被覆率を下げている主要な原因は，外国語の店舗名とカタカナ表記との対応がとれないことであった．この問題については今後音訳の技術などを導入して対処したい．

比較のため，信頼度ではなく cosine 類似度の高い順にソートして各点での精度と再現率をプロットした精度－再現率曲線 (cosine) も図 2 に示す．両者が対応付けたレコードはまったく同じであり，信頼のできるものをどれだけスコア上位にもってこられたかという評価である．図 2 から，低い再現率のときの精度はいずれの点でも信頼度を導入した結果のほうが高く，cosine 類似度をそのまま足切りに使用するよりも，よい結果が得られるといえる．この結果から導入した信頼度は有効に働いたと考えている．

次に，信頼度で使用したそれぞれの項がどの程度効果があったのかを調べた．図中の”w/o string similarity” は，$support(c)$ のみを信頼度として用いた場合であり，その場合も cosine 類似度よりよい結果を得ており，表記の一致率を考慮することでさらに精度が向上していることがわかる．

5.3.1 誤り分析

上記の評価で，システムが誤って対応づけた事例のうち，システムの出力した信頼度が高かった事例から上位 40 件を人手で分析し，何が問題となっているかを調査した．その結果，あるチェーン店を指していた場合に店舗 DB に対応先がなく，別のチェーン店を出力したという誤りが半数を占めていることがわかった．その一例を示す．

[4] これは評価データに出現した店舗 DB に対応先がある店舗の 8 割を被覆する

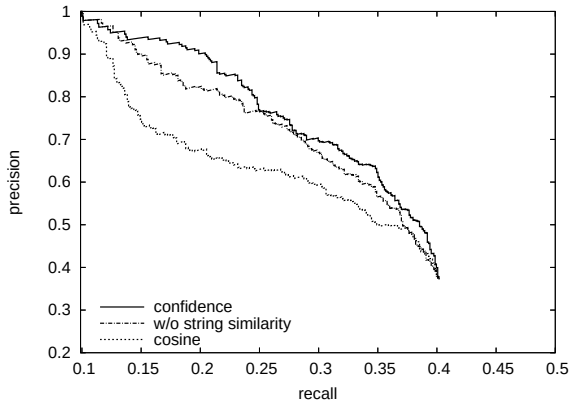

図 2: 信頼度と cosine 類似度の比較

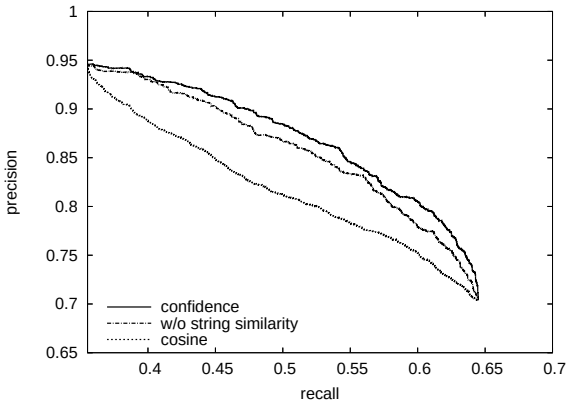

図 3: 人物ドメインでの評価

永坂更科は麻布十番に本店がある老舗チェーン店ですね。

この例では**永坂更科**は永坂更科という店舗全体を指しているため，対応先はないとして棄却すべきところを，近くに「麻布十番」という本店を指す有力な共起語があったために,「永坂更科本店」に誤って対応付けられていた．これらの問題を解くには，共起語を獲得する範囲を統語情報を利用して範囲を絞りこむなど，文脈の獲得方法についてさらに検討する必要がある．

5.3.2 提案手法の別分野での評価

上では手法の店舗分野における性能について述べた．さらに手法の非ドメイン依存性を調査するために，人物についても同じ手法で評価した．

人物の実体 DB は，wikipedia 中の人物ページを抽出して作成し，約 7 万人が登録されている．人物ページか否かは，各ページのカテゴリ情報に「〜年生・〜年没」「〜の人物」などが含まれているか否かで判断した．また，実体の共起語は wikipedia の各ページから獲得し，その際，愛称も「愛称は〜」などのパタンを用いて獲得した．評価データ作成のため，新たに blog 記事 1000 件に人名タグを付与し，フルネームと ID を付与した．タグが付与された人名は 15561 人で，そのうち約 65%はデータベースに対応先があった．

結果を図 3 に示す．4.1 で述べた方法で候補を取得した場合の候補の被覆率は約 9 割であり，店舗よりも再現率の上限値が高い．左端は再現率 0.36 のときの精度 0.94 の点で，これは文章内にフルネームが存在した場合に，ある出現表記の候補にそのフルネームを持つ ID が一人であれば解とした場合の精度・再現率である．結果から，cosine 類似度よりも今回の信頼度がよく，表記一致率の寄与についても店舗と同様の結果が得られた．この結果と店舗の結果を踏まえて， 今回の手法は異なるドメインにおいて有効に働いたと考えられる．

6 おわりに

本稿では，テキスト中の固有表現が実際に指す実体を同定する問題を，固有表現とデータベースレコードの対応づけタスクとして設定し，信頼度による棄却を導入した共起語の類似度による手法を検討した．店舗と人物でこの手法を評価し導入した信頼度の効果を確認した．今後の課題はおおきく 2 つある．まず，今回の手法は個々の固有表現を独立に対応付けしているが，実際には固有表現が互いに依存して決まることも多いため，このような依存性を考慮した手法を今後検討する必要がある．

もう一つは実体をどう考えるべきかについての検討である．今回は，個別の実体を指していてかつデータベースにある場合以外を全て対応先なしとしているが，少なくとも個別の実体を指す場合と集合を指す場合（たとえばグループとしての NTT）の二つが存在している．今後はこれらの問題を切り分けつつ，フラットなデータベースだけでなく階層関係を持ったデータベースを考えた問題設定も検討していきたいと考えている．

参考文献

[1] Javier Artiles, Julio Gonzalo, and Satoshi Sekine. The SemEval-2007 WePS evaluation: Establishing a benchmark for the web people search task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pp. 64–69, 2007.

[2] Chris Bizer, Tom Heath, Kingsley Idehen, and Tim Berners-Lee. Linked data on the web. In *Proceedings of the Workshop on Linked Data on the Web (LDOW)*, 2008.

[3] Silviu Cucerzan. Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pp. 708–716, 2007.

[4] Christopher D. Manning and Hinrich Schütze. *Foundations of statistical natural language processing, Chapter 7 Word sense disambiguation.* MIT press, 1999.

[5] 小野真吾, 吉田稔, 中川裕志. Web における名寄せシステム. 言語処理学会第 12 回年次大会発表論文集, 2006.

[6] 高橋いづみ, 浅野久子, 松尾義博, 菊井玄一郎. 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 回年次大会発表論文集, pp. 821–824, 2008.