重要語抽出を用いたWeb文書上の同姓同名の曖昧さ解消

小野　真吾† 　佐藤　一誠† 　吉田　稔†† 　中川　裕志††

† 東京大学大学院 情報理工学系研究科　〒113–0033 東京都文京区区本郷 7–3–1
†† 東京大学 情報基盤センター
E-mail: †{ono,sato,mino}@r.dl.itc.u-tokyo.ac.jp, ††nakagawa@dl.itc.u-tokyo.ac.jp

あらまし　Web 上には人物に関する多くの情報があり，近年では検索エンジンを用いることで，その情報を容易に得ることができる．しかし，人名には同姓同名が存在するように曖昧性があり，1 つの名前についての検索結果が実際には異なる数人についての記述を集めてしまう場合が多々ある．この問題に対し，我々は同姓同名の曖昧さ解消を目的としたシステムを提案する．このシステムは，2 つのアルゴリズムを用いて曖昧さ解消を行う．1 つは，対象となる人名と文書上で共起する固有表現を見つけることにより抽出されるソーシャルネットワーク情報を用いる方法であり，もう 1 つは各文書中から抽出される重要語に基づく文書間の類似度計算による方法である．実際の Web 上にある文書をデータセットとして用いた実験により，これらのアルゴリズムが有効であることを示した．

キーワード　同姓同名の曖昧さ解消，Web マイニング，クラスタリング，重要語抽出

Person Name Disambiguation on the Web Using Keyword Extraction

Shingo ONO†, Issei SATO†, Minoru YOSHIDA††, and Hiroshi NAKAGAWA††

† Graduate School of Information Science and Technology, The University of Tokyo.　Hongo 7–3–1, Bunkyo-ku, Tokyo, 113–0033 Japan
†† Information Technology Center, The University of Tokyo
E-mail: †{ono,sato,mino}@r.dl.itc.u-tokyo.ac.jp, ††nakagawa@dl.itc.u-tokyo.ac.jp

Abstract　The World Wide Web (WWW) provides lots of information about persons, and in recent years WWW search engines have been common tools to learn about persons. However, the ambiguities of person names (i.e., there are many persons with the same name,) typically causes the search results of one person name to result in web pages about several different persons. We propose a novel framework of the person name disambiguation. Our framework employ the following two components: extracting social network information by finding co-occurrences of named entities, and measuring document similarities based on occurrences of key compound words. In experiments using the actual web document dataset, the result of our framework is promising.

Key words　person name disambiguation, Web mining, clustering, keyword extraction

1. はじめに

近年，Web には経歴や業績などをはじめとする人物に関連する情報が豊富に存在し，検索エンジンを用いることにより，それらの情報に容易にアクセスすることが可能である．しかし，検索エンジンを利用する際，よくある名前を検索すると、その結果の曖昧さに困る場合がある．例えば，元総理大臣ではない “小泉純一郎” という人について情報を集めたい場合，検索結果には元総理に関するページが並び，目的のページを探すことは大変困難である．状況にもよるが，検索を行うユーザは検索エンジンを利用した後に検索結果の中のどこに目的の情報が書かれているか丹念に探す必要がある．この問題は，異なる人物が同じ名前を持つために起こる問題である．目的のページは見つかるが，そのためには大変多くの手間と時間が必要になるおそれがある．

本論文において，人名とは文書中に出現する，実世界における人物（実体）をただ一つのみ参照する文字列とする．上記の問題は，異なる人物が共通する人名を持つために起こる問題である．本論文では，この問題の解決を目的とした，我々の開発したシステムについて説明する．このシステムの内部で用いられる手法は以下の 2 つの考えに基づいている．1 つは人が現実世界で何らかのソーシャルネットワークを持つということ，もう 1 つは人を特徴付けるキーワードが存在するということである．これらを手がかりに，同姓同名を見分けることを狙う．

我々のシステムは，Web 上の文書に対し，固有表現抽出により人名や地名，組織名といった固有表現の共起を探し，それを利用してソーシャルネットワーク情報を抽出する．また，その情報とは別に，複合名詞の統計値に基づく重要語抽出を行い，文書中のキーワードを抽出する．これらの情報を利用し，同じ実体を参照する人名をクラスタリングすることで，同姓同名の曖昧さ解消を行っている．評価実験の結果，これらの方法は文書中の単語情報をそのまま利用する手法に比べ，F-measure で 0.20 以上の性能の向上が見られた．

本論文の構成は以下のとおりである．まず，2 章において本論文が扱う問題について詳細に定義する．続いて，3 章では，我々の提案する手法について説明する．また，システムの概要についても紹介する．4 章で我々が行った性能評価とその結果を紹介する．5 章にまとめと今後の課題を述べる．

2. 同姓同名の曖昧さ解消の問題について

我々の目的は，人名の持つ曖昧さ解消である．この問題を以下のように定義する．クエリを q とする．本論文において，クエリとは，すなわち曖昧さ解消の対象とする人名である．検索エンジンにクエリ q を与えることにより得られる Web 上の文書を $\mathcal{D} = \{d_1, d_2, \cdots, d_k\}$ とする．なお，以下では簡単のために，Web 上の文書のことを Web ページ，あるいは単にページと書く．各 Web ページにはクエリ q が少なくとも 1 回は出現している．そこで，Web ページ d_i において，クエリ q が j 回目に出現した文字列を s_{ij} とし，s_{ij} の集合を $\mathcal{S}$ とする．それぞれの文字列 s_{ij} は，実世界において名前 q を持つ人物の集合 $\mathcal{E} = \{e_1, e_2, \cdots, e_n\}$ 中のただ 1 つの要素のみを指し示す．人名から実体への参照を表す関数 Φ を，$\Phi : \mathcal{S} \rightarrow \mathcal{E}$ と定義する．この関数 Φ を求めることにより，同姓同名の曖昧さ解消が行われることになる．

上記のモデル化では 1 つの文書中に出てくる複数のクエリ q がそれぞれ異なる人物を参照することも考慮可能である．しかし，今回は簡単のために，同一の文書中に出現するクエリ q はすべて同一の人物を参照すると仮定する．すなわち，ある文書 d_i 中に出現する s_{ij} について，$\forall j, \Phi(s_{ij}) = e_m$ となる．この仮定により，求める関数は $\Phi' : \mathcal{D} \rightarrow \mathcal{E}$ となる．我々のこの論文における目的は，この関数 Φ' を推定することである．

なお，関数 Φ' の推定において，用いることのできる前提知識は入力されたクエリ q のみとし，対象とする人名を持つ人物に関する実世界での情報は用いないものとする．

この問題の解は，Web ページをクラスタリングした形で得られることになる．クラスタリングされた結果，Web ページの集合 $\mathcal{D}$ の中に出現する同姓同名の異なる人物の人数と同じ数のクラスタが作られ，各クラスタにおいて同一の人物を参照する Web ページが漏れなくまとまっていれば，同姓同名の曖昧さ解消が完全に成功したことになる．

この問題において難しい点のひとつとして，クエリ q に対して異なる実体が実際に何個存在するか，ということを事前に知ることはできないところにある．そのため，クラスタの数を指定したクラスタリングの方法などは用いることができない．

この問題に関する先行研究として，以下のようなものがある．Bagga ら [2] は，この問題に対し，文書に出現する単語を要素とする文書ベクトルを作るにより文書間の類似度を計算し，クラスタリングを行った．出現する単語に加え，文書中から人物に関する個人情報を抽出し，それらの情報を考慮してクラスタリングする試みとして, [5] が挙げられる．佐藤ら [9] は文書中に出現する人名に注目し，共起する人名を用いて Web ページ間のリンクを作成し，それをグループ分けすることで同姓同名の分離を行っている．Wan ら [11] は，高い頻度で検索される人名をクエリとした場合の検索結果をクラスタリングするシステムを提案している．Wan らのシステムにおいては，例えば苗字や名前だけがクエリとして与えられる場合も想定されている．

Bekkerman ら [3] は，実世界において同一コミュニティに属す複数の人物に関するページを集め，それらの中でのリンク解析や階層併合・分割ダブルクラスタリングを行うことで，同一コミュニティに属す人物を同時にそれぞれの同姓同名についての記述と分離する方法を提案した．この方法では同じコミュニティに属す人物が結果として前提知識として与えられた上でクラスタリングを行っていることになる．

3. 提案するシステム

本章では，我々の提案するシステムの内部で用いられているアルゴリズムと，システム全体の概要について説明する．

3.1 クラスタリングの方法

我々が同姓同名を見分けることを手作業で行う場合，何か手がかりとなる人物の情報（性別，年齢，肩書きや所属といった個人情報や，その人物の属すコミュニティの情報など）を元に，それぞれの文書に出現する対象の人名が同一人物かどうかを判定していく．その際，人物に関する情報は時間の経過とともに変化していくことが予想される．そのため，これらの情報が文書間で食い違うだけでは，出現する人名が異なる実体を参照するとは判定できない．一方，上で挙げたような情報が多くの点で一致するならば，同一の人物を参照していると考えられる．すなわち，複数の文書に渡って同じ人名が出現した場合，同一の人物を参照しているという証拠を元に，同じ人物を参照する文書を判定していくのである．我々はこの手作業による同定の方法に近いクラスタリングの方法を採用する．

まず，すべての 2 文書 d_x, d_y 間について何らかの基準で類似度 $\mathrm{sim}(d_x, d_y)$ を計算する．そして，その類似度が閾値 θ を超える文書ペアについては，同一の実体を参照する，すなわち $\Phi'(d_x) = \Phi'(d_y)$ と判定する．この条件によって次のようにクラスタリングが行われる．$G(V, E)$ を無向グラフとする．各頂点 $v_i \in V$ は Web ページ d_i に対応している．また，上記のアルゴリズムにおいて $\Phi'(d_i) = \Phi'(d_j)$ ならば $e_{ij} \in E$ とする．これより，枝集合 E が得られることになる．ここで，$G = \langle V, E \rangle$ はいくつかの連結成分を持つことになる．それぞれの連結成分は，同じ実体を参照している Web ページの集合を表すことになる．この連結成分をクラスタとみなすことで，クラスタリングを行う．

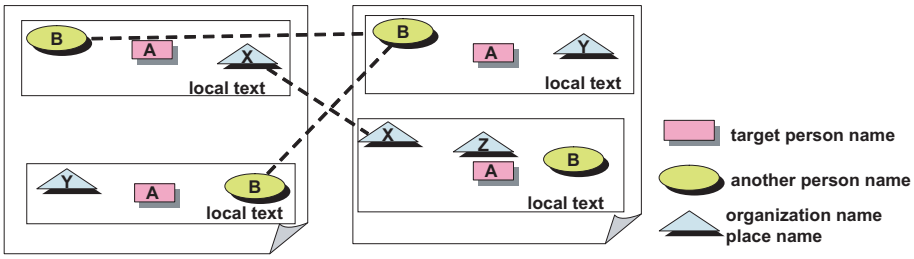

図 1　人名や地名，組織名の異なる文書間での出現

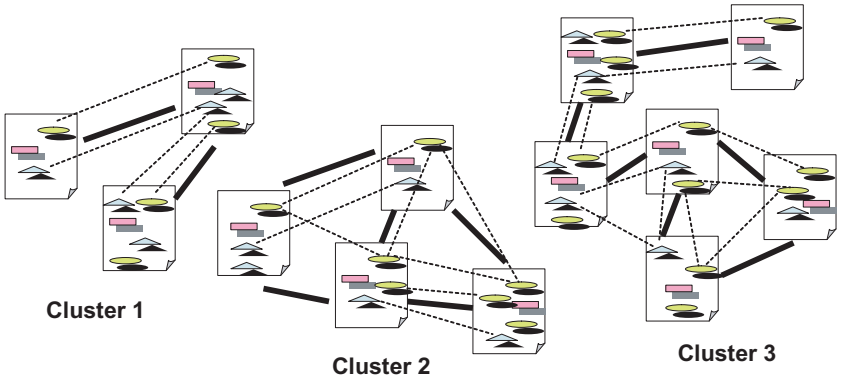

図 2　ソーシャルネットワーク情報によるクラスタリング

3.2　ソーシャルネットワーク情報によるクラスタリング

固有表現（Named Entity; 以下 NE）[6] とは人名や組織名など事物に特有の表現であり，例えば [6] によると，日本語の NE としては，組織名，人名，地名，固有物名，日付表現，時間表現，金額表現，割合表現という 8 つの固有表現の種類が定義されている．NE は一般的な語と比較して，より文書を特徴付けていると考えられる [10]．すなわち，NE は文書を非常に強く特徴付けたり，文書中の中心となるトピックを与えるものであるため，NE を正しく用いることができれば，非常に強力な同一の実体の判定方法となることが期待できる．我々は特に人名，地名，組織名といった NE に着目することで，その人物に関するソーシャルネットワークの情報が効率よく抽出できることを狙いとしている．

人物の同定を行う作業を考えた場合，

- 対象とは別の特定の人物と，複数の文書に渡って共に現れる場合，それらはすべて同一人物である
- 人物の名前の近くに出現する組織名は，その人物の所属を表すと考えられ，複数の文書でそのような状況ならば，その人物は同一人物である

といった NE に着目したヒューリスティクスに基づく方法が考えられる．図 1 にこの方法のイメージ図を示す．点線は同じ人名，地名あるいは組織名を表している．このように文書中に異なる人名や組織名などが共通して出現するのが見られた場合，必ずしも同一の実体を参照すると断定はできないが，同一の実体を参照している確率が非常に高いと言える．

このように，関連する人名や組織名といった NE の形で文書中に出現するソーシャルネットワーク情報に基づいて，複数の文書に渡って出現する同名の参照先が同一実体であるかどうかを判定する方法を，ソーシャルネットワーク情報によるクラスタリングと呼ぶ．

ソーシャルネットワーク情報の抽出によるクラスタリングのアルゴリズムを以下に示す．

procedure ソーシャルネットワークによるクラスタリング

1. クエリ q を含むすべての文書 d_j $(1 \leqq j \leqq k)$ について，NE タガーによってタグ付けされた人名，地名および組織名を抽出する．なお，人名に関しては姓名が揃って出現するものに限る．
2. 文書間のソーシャルネットワーク類似度（以下，SN 類似度）$\mathrm{sim}_{\mathrm{SN}}(d_x, d_y)$ を次のように計算する．
   $\mathrm{sim}_{\mathrm{SN}}(d_x, d_y)$
   $= \alpha * (d_x, d_y$に共通して出現した人名の数）
   $+\beta * (d_x, d_y$に共通して出現した人名以外の NE の数）
3. 2 文書 d_x, d_y において，$\mathrm{sim}_{\mathrm{SN}}(d_x, d_y) \geqq \theta_{\mathrm{SN}}$ であるならば，$\Phi'(d_x) = \Phi'(d_y)$.

なお，θ_{SN} はあらかじめ定めた閾値，α, β は重み付けのパラメータである．図 2 にソーシャルネットワーク情報によるクラスタリングのイメージ図を示す．図 2 中の太い実線は 2 文書間の SN 類似度が閾値よりも高く，グラフの頂点間（すなわち，Web ページ間）に枝が張られた状態を表している．

なお，今回は固有表現抽出を行う NE タガーとして Sen [12] の出力結果を利用している．Sen は Java で実装された形態素解析器であるが，固有表現（人名，地名，組織名）に関してもある程度タグを出力するので，そのタグを NE の判定に用いた．

3.3　文書中の重要語を用いたクラスタリング

本節では，文書を特徴づける重要語を文書中から見つける方法を説明する．一般的な固有表現抽出を行うツールでは，固有

名詞的な表現の抽出に比べ，普通名詞の複合語からなる特徴的な語（例えば，電子情報通信学会など）については十分に抽出できない場合が多い．また，辞書にない地名や組織名には対応しきれない場合がある．そこで，固有表現のかわりに，文書を特徴づける重要語を見つけ，それをその文書のキーワードとした上で，キーワードの重複性の高い文書どうしを同じ人物を参照すると判定する方法を考える．

今回は，重要語抽出の方法として中川らが開発した言選 Web [7] で使用されているアルゴリズムを用いた．言選 Web では，それぞれの文書中で単語の頻度情報と連接情報の計算を行うことで，専門用語候補を発見することができる．

言選 Web で用いられている重要語抽出のアルゴリズムを以下に示す．まず，文書中の複合名詞の重要度が次のようにして計算される．単名詞 N の左方の連接頻度を $FL(N)$，右方の連接頻度を $FR(N)$ とする．単名詞 $N_1, N_2, \cdots, N_L$ がこの順で連接した複合名詞を CN とする．CN の連節度を考慮したスコアは，次のように定義される．

$$LR(CN) = \left(\prod_{i=1}^{L} (FL(N_i)+1)(FR(N_i)+1) \right)^{\frac{1}{2L}}$$

この式により，複合名詞と同時に単名詞のスコア付けも行われている．複合名詞の重要度は，この $LR(CN)$ に文書中での複合名詞の出現回数 $f(CN)$ を掛け合わせることにより定義される．すなわち，CN の重要度 $FLR(CN)$ は，

$$FLR(CN) = f(CN) \times LR(CN)$$

である．なお，$f(CN)$ は，語 CN が単独で出現した場合の頻度であり，他の複合名詞の中に包含されていた場合はカウントされない．

例えばある文書中から “情報” または “システム” を含む語句を抽出したときに，以下の語句が抽出されたとする．

情報 通信，情報，　災害 情報，　電子 情報 通信，
情報 抽出，情報 通信，　情報，　災害 情報 システム，
情報 システム

これを，バイグラムごとに頻度をまとめると，以下のようになる．

災害 情報 (2)	情報 通信 (3)
電子 情報 (1)	情報 抽出 (1)
	情報 システム (2)

この場合において，単語 “情報” の重みを計算すると，LR(情報) $= \sqrt{(3+1)(6+1)} \simeq 5.29$ であり，情報は単独で 2 回出現していることから，FLR(情報) $= 2 \times \sqrt{(3+1)(6+1)} \simeq 10.58$ となる．同様にして，複合語 “情報システム” の重みは，FLR(情報システム) $= 2 \times ((3+1)(6+1) \times (2+1)(0+1))^{\frac{1}{4}} \simeq 6.05$ となる．

この言選 Web のアルゴリズムは日本語以外の言語においても品詞タガーさえあれば適用することができる．すなわち，前述の固有表現抽出を用いてソーシャルネットワーク情報を抽出する場合に比べて，言語依存性が小さいと言える．

言選 Web の重要語抽出の手法により，各文書 d_i に対し文書中の重要語とその重みが計算される．抽出された重要語とその重みを要素とする文書ベクトルを $\boldsymbol{w_i}$ をとすると，以下のようにして重要語によるクラスタリングが行われる．

procedure 重要語によるクラスタリング

1. クエリ q を含むすべての文書 d_i $(1 \leqq i \leqq k)$ について，重要語抽出アルゴリズムに従って，重要語とその重みを計算し，文書ベクトル $\boldsymbol{w_i}$ を計算する．
2. 重要語に関しての類似度 $\mathrm{sim_{CKW}}(d_x, d_y)$ を次のように計算する．
   $\mathrm{sim_{CKW}}(d_x, d_y) = (\boldsymbol{w_x}, \boldsymbol{w_y})$
3. 2 文書 d_x, d_y において，$\mathrm{sim_{CKW}}(d_x, d_y) \geqq \theta_{\mathrm{CKW}}$ であるならば，$\Phi'(d_x) = \Phi'(d_y)$.

なお，θ_{CKW} はあらかじめ定めた閾値である．先に述べたとおり，$\Phi'(d_x) = \Phi'(d_y)$ の条件により，クラスタリングが行われることになる．

3.4 同姓同名の曖昧さ解消を行うシステムの概要

前節までで紹介した手法を用いて，我々が開発した同姓同名の曖昧さ解消を行うシステムの概要を図 3 に示す．このシステムはクエリが与えられる度に検索結果の Web ページをすべて実際に取得する．システムを動作させる際，Web ページを取得するところが最も時間を要する．そこで，ページのダウンロードと文書の前処理を行う部分では並列処理を行って，実行時間が短縮されるようにしている．

我々が開発したシステムは以下のようにして動作する．

まず，システムはユーザから検索対象とするクエリ（人名）を受け取る．システムは受け取ったクエリを検索エンジンに与え，そのクエリに関する検索結果上位 k 件の URL を得る．今回は検索エンジンとして Yahoo! JAPAN の検索用の API [13] を用いている．その後，検索結果の k 件の Web ページをすべてダウンロードする．

ダウンロードした Web ページに対し，以下の前処理を行う．まず，HTML のタグを取り除き，タイトル（title タグで囲まれた中身）と本文（body タグで囲まれた中身）を取り出す．以下，この取り出した部分を文書と呼ぶ．文書を抽出した後，文書中にクエリが含まれているかを確認する．これは，Web ページの更新と検索エンジンのインデックス作成には時間差があるため，検索エンジンでヒットした文書であっても，更新が行われ対象とする人名を含まない文書になっている場合や，その Web ページが削除されて場合が考えられるためである．そのため，クエリが含まれていない文書については，これ以降は無視することにする．クエリが含まれることが確認された文書については，さらにクエリの前後 n 単語のみの文脈を抽出し，この文脈に対し Sen [12] を用いて形態素解析を行い，その結果を元に固有表現や重要語を抽出する．

前処理により，各文書中の固有表現や重要語が抽出されるので，それを利用して文書間の類似度計算およびクラスタリング

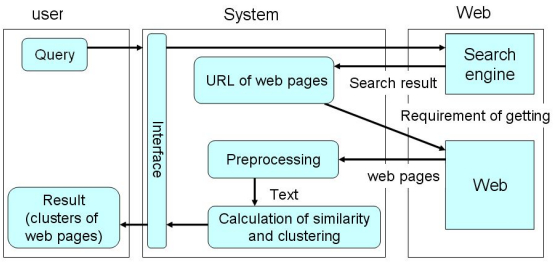

図 3　システムの概要

を行う．システムはこのクラスタリングの結果を出力して終了する．

なお，ソーシャルネットワークの情報によるクラスタリングで得られるグラフの枝集合を E_{SN}，重要語抽出によるクラスタリングで得られるグラフの枝集合を E_{CKW} としたとき，この 2 つの手法を併用した場合のクラスタリング結果は，枝集合 $E = E_{\mathrm{SN}} \cup E_{\mathrm{CKW}}$ により与えられる．これは，この 2 つのクラスタリング方法がそれぞれ異なる情報源によりクラスタリングを行っていることが期待されるからである．

4. 評価実験

本章では，我々が行ったこのシステムの性能評価とその結果について紹介する．

4.1 評価指標

我々は，評価指標として Precision（精度），Recall（再現率），F-measure を用いた．複数の文書についてクラスタリングを行った場合の各指標の計算方法は以下のような方法で行う [4]．

正解集合を $\mathcal{C} = \{C_1, C_2, \cdots, C_n\}$，クラスタリングを行った結果の集合を $\mathcal{D} = \{D_1, D_2, \cdots, D_m\}$ とする．ここで，C_i や D_j は文書の集合（クラスタ）である．正解集合の各クラスタ $C_i (1 \leqq i \leqq n)$ について，Precision，Recall，F-measure の値 P_{ij}，R_{ij}，F_{ij} をすべての結果のクラスタ $D_j (1 \leqq j \leqq m)$ に対して計算する．

$$\mathrm{P}_{ij} = \frac{|C_i \cap D_j|}{|D_j|},\ \mathrm{R}_{ij} = \frac{|C_i \cap D_j|}{|C_i|},\ \mathrm{F}_{ij} = \frac{2\mathrm{P}_{ij}\mathrm{R}_{ij}}{\mathrm{P}_{ij} + \mathrm{R}_{ij}}$$

各 C_i について，もっとも F-measure が高くなるような D_j を計算し，それを正解集合の C_i における F-measure とする．すなわち，$\mathrm{F}_i = \max_j \mathrm{F}_{ij}$ である．また，このとき，P_i，R_i については，$j' = \mathrm{argmax}_j \mathrm{F}_{ij}$ を用いて，$\mathrm{P}_i = \mathrm{P}_{ij'}$，$\mathrm{R}_i = \mathrm{R}_{ij'}$ として求める．

システムとしての総合評価は，クラスタの要素数を重みとした重みつき平均により与える．よって，システムとしての F-measure(F) は，$\mathrm{F} = \sum_{i=1}^{n} |C_i|\mathrm{F}_i / |\mathcal{C}|$ となる．ただし，$|\mathcal{C}| = \sum_{i=1}^{n} |C_i|$ である．システム全体の Precision(P) および Recall(R) についても同様に計算する．

各指標はそれぞれ，次のような意味を持つ．Precision はどれだけ正解集合の要素のみでクラスタを作られているかを表している．そのため，クラスタの大きさに関わらず，正解の対象となる 1 つの実体のみを指すクラスタでは必ず P_i=1 となる．一方，Recall はどれだけ正解集合の要素を含んでいるかを表している．クラスタに他の実体を指す文書がどれだけ混在しても，正解の対象をすべて含んでいれば，R_i=1 となる．なお，クラスタリングをまったく行わず，1 つの Web ページが 1 つのクラスタとなる状態では Precision は 1 となること（ただし，この場合には Recall はかなり低くなる）や，すべての文書をまとめて，1 つのクラスタとみなすと Recall は 1 となる（逆に Precision がかなり低くなる）ことに注意する．Precision や Recall は上記のような性質を持つので，2 つの指標の調和平均の値である F-measure が総合的な指標として用いられる．

表 1　実験結果（人名 38 クエリの平均）

適用した手法	F	P	R
No cluster	0.2996	1.0000	0.2536
Baseline	0.5409	0.6668	0.6950
SN	0.7163	0.9000	0.6692
CKW	0.6974	0.8195	0.7050
SN ∪ CKW	**0.7443**	**0.8328**	**0.7683**

4.2 データセット

実験は，日本語で書かれた Web ページを対象に行った．まず，人名をクエリとして検索エンジンで検索し，その結果の上位のページ 100〜200 件程度を各クエリについて収集した．その中から同名を持つ 2 つ以上の実体を参照する Web ページや，クエリの文字列を含まないページを取り除き，その結果，38 クエリについて計 5015 の Web ページを収集した．収集した各 Web ページについて人手で正解付けを行った上で，それぞれの手法を適用し，各手法の性能を評価した．

なお，正解付けにおいても，できるだけ前提知識は用いず，基本的に収集された Web ページから各ページの参照する人物かどうかを判断するようにしている．これは例えば言語学者であり政治学者でもあるノーム・チョムスキー (Noam Chomsky) に関する文書をクラスタリングする際に，チョムスキーが言語学と政治学の両方に携わっているという言及が収集した文書中になければ，結果として言語学者としてのチョムスキーと政治学者としてのチョムスキーの 2 つのクラスタができるのが正解となる．

今回実験を行うソーシャルネットワークの情報を用いてクラスタリングや，重要語を用いたクラスタリングでは閾値 $\theta_{\mathrm{SN}}, \theta_{\mathrm{CKW}}$ をそれぞれ決める必要がある．そのため，クエリをランダムに 5 グループに分けて交差検定を行った．すなわち，5 グループの中から，1 つのグループを選び，これをテストセットとする．閾値 $\theta_{\mathrm{SN}}, \theta_{\mathrm{CKW}}$ をそれぞれ，テストセットを除く 4 グループ（トレーニングセット）において F-measure が最大になるように設定する．その閾値を用いてテストセットの評価を行う．テストセットとトレーニングセットの組合せを変えて，5 回行うことで，全体の評価を求めている．

4.3 実験結果

実験結果を表 1 に示す．また，人名ごとの詳細な結果を図 4 に示す．図 4 において，横軸はそれぞれ 1 つの人名に対応する id を並べたものになっている．表 1 および図 4 において，No cluster はクラスタを作らない，すなわち検索エンジンの結果を

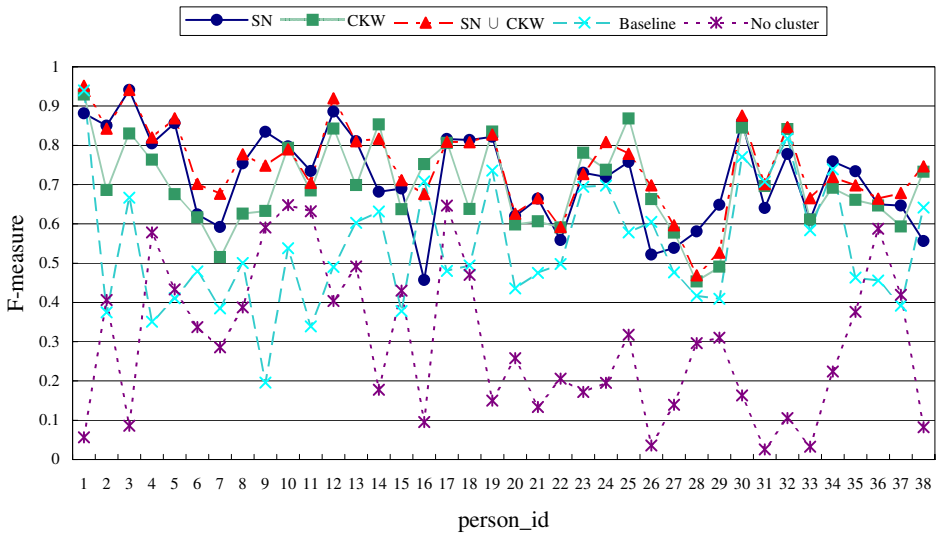

図 4　人名ごとの結果

クラスタリングせずにそのまま評価した場合の指標である．前節で述べたとおり，精度は必ず 1 になるが，再現率が非常に低く，そのために F-measure も非常に低い．また，Baseline は文書中に出現する語を出現頻度により重み付けした文書ベクトルを計算し，文書ベクトルの類似度を元にクラスタリングを行う単純な手法の場合の結果である．SN，CKW はそれぞれソーシャルネットワーク情報を用いたクラスタリング，文書中の重要語を用いたクラスタリングであり，SN ∪ CKW は 2 つの手法を併用した場合の結果である．

実験結果より，単純な手法に比べ，SN，CKW とも大いに性能の向上が見られている．手法単体では，SN，すなわち固有表現を抽出することによりソーシャルネットワーク情報を利用するのが効果的であり，精度も非常に高いが，この方法には固有表現抽出のためのツールが必要であるという欠点がつきまとう．一方，CKW，すなわち重要語抽出によるクラスタリングにおいては，SN に近い結果を言語依存性のより小さいアルゴリズムを用いることで達成できている．

また，2 つの手法においては，違う種類の情報を元にクラスタリングを行っていると考えられ，2 つの手法を同時に適用した場合，さらに F-measure の向上が見られる．図 4 で見られるように，多くの人名において SN∪CKW を用いた場合が最も F-measure が高くなった．

5. おわりに

我々は同姓同名の曖昧さ解消を目的として，ソーシャルネットワーク情報によるクラスタリングと文書中の重要語抽出によるクラスタリングの 2 つの方法を提案した．さらに，それらを用いて同姓同名の曖昧さ解消を行うシステムを開発した．性能評価の結果，我々の提案した 2 つの手法を併用した場合，F-measure で 0.74 を超える高い性能を示すことがわかった．

同姓同名の曖昧さ解消の研究は近年さかんになってきており，実験についてのデータセットの整備も行われてきている．昨年には Web 上での人物検索に関するワークショップ WePS [1] が行われ，英語で書かれた Web ページからなるテスト用のデータセットが公開された．現在，このシステムは日本語で書かれた文書のみを対象に扱っているが，英語で書かれた文書についても解析を行えるようにし，共通のデータセットでの結果を求めることが今後の課題として挙げられる．

文　献

[1] J. Artiles, J. Gonzalo and S. 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] A. Bagga and B. Baldwin: Entity-Based Cross-Document Coreferencing Using the Vector Space Model. In Proceedings of COLING-ACL 1998, pp. 79–85, 1998.

[3] R. Bekkerman and A. McCallum: Disambiguating Web Appearances of People in a Social Network. In Proceedings of WWW2005, pp. 463–470, 2005.

[4] B. Larsen and C. Aone: Fast and effective text mining using linear-time document clustering. Proceedings of the 5th ACM SIGKDD. pp. 16–22, 1999.

[5] G. S. Mann and D. Yarowsky: Unsupervised Personal Name Disambiguation. In Proceedings of CoNLL2003, pp. 33–40, 2003.

[6] 宮本昌幸, 松尾衛, 森辰則: IREX NE タスク 結果分析. In Proceedings of the IREX Workshop, pp. 107–114, 1999.

[7] H. Nakagawa and T. Mori: Automatic Term Recognition based on Statistics of Compound Nouns and their Components, Terminology, Vol 9, No 2, pp.201–219, 2003.

[8] S. Ono, M. Yoshida and H. Nakagawa: NAYOSE: A System for Reference Disambiguation of Proper Nouns Appearing on Web Pages. Proceedings of AIRS2006(Asia Information Retrieval Symposium), pp. 338–349, 2006.

[9] 佐藤進也, 風間一洋, 福田健介, 村上健一郎: 実世界指向 Web マイニングによる同姓同名人物の分離, 情報処理学会論文誌：データベース, Vol. 46, No. SIG8 (TOD26), pp. 26–36, 2005.

[10] 関根聡: テキストからの情報抽出. 情報処理学会論文誌, Vol. 40, No. 4, pp. 370–375, 1999.

[11] X. Wan, J. Gao, M. Li, and B. Ding: Person Resolution in Person Search Results: WebHawk. Proceedings of CIKM2005, pp. 163–170, 2005.

[12] Sen. `https://sen.dev.java.net/`

[13] Yahoo! JAPAN Web サービス `http://developer.yahoo.co.jp/`