同姓同名人物情報の分類に関する一考察

吉田　康浩† 　鍛治　伸裕† 　喜連川　優†

† 東京大学生産技術研究所
E-mail: †{yoshida,kaji,kitsure}@tkl.iis.u-tokyo.ac.jp

あらまし　本論文では wikipedia という新しい言語資源に注目して、これがウェブ空間における同姓同名問題に対して有効か検証する実験を行なった。このタスクは、自然言語処理において広く議論されている多義性解消の一種であると考えることができる。実験では wikipedia に記載されている人物情報テキストの特徴単語を用いてページの分類を行なった。１７の人物名を用いて，１７３６件のウェブ文書に対する実験を行った結果、wikipedia に記載されている情報は同姓同名問題の解決に有効であることを確認した。

キーワード　Web データマイニング, 同姓同名問題

A Study of Person Name Disambiguation

Yasuhiro YOSHIDA†, Nobuhiro KAJI†, and Masaru KITSUREGAWA†

† University of Tokyo Institute of Industrial Science
E-mail: †{yoshida,kaji,kitsure}@tkl.iis.u-tokyo.ac.jp

Abstract　This paper presents the efficiency of wikipedia content for solving the problem of Person Name Disambiguation on the World Wide Web. Person name disambiguation problem is similar to word sense disambiguation problem in Natural Language Task. We utilize the word of the wikipedia page for classifing web pages. To evaluate our method, we callected and hand-labeled a dataset of 1736 web pages on 17 personal names. On this dataset, the content of wikipedia is effective in solving person name disambiguation problem.

Key words　Web data mining, Person Name Disambiguation,

1. はじめに

情報検索における問題のひとつに同姓同名問題が存在する。ある人物に関する情報を探すため、その人物名を検索語として与えた場合を考える。検索語と文書の間で単純に文字列照合を行ったのでは、目的の人物について記述している文書のほかに、その同姓同名人物に関する文書まで検索されてしまう。これに対する単純な解決方法は、検索された文書を別人ごとにクラスタリングして提示することである。

このタスクは、自然言語処理において広く議論さている多義性解消 (Word Sense Disambiguation) の一種であると考えることができる。これまで自然言語処理の分野では、タグ付きコーパスを利用した教師有り手法や、国語辞典の定義文に基づく手法が多義性解消のために提案されている [1]。しかし、こうした手法は同姓同名問題には直接適用できない。なぜなら，検索されるような人物名を網羅したようなタグ付きコーパスや国語辞典は存在しないからである。

そこで我々は wikipedia という辞書に着目した。wikipedia に記載される情報は不特定多数のユーザが協力して更新されていて、既存の辞書とは異なり即時性や網羅性が高い。そのため、検索されるような人物名に関する記述も豊富であることが期待できる。

本論文では wikipedia という新しい言語資源の同姓同名問題における有効性を検証する実験を行なったのでその報告を行なう。多義性解消の手法には、wikipedia に記載されているテキストの特徴単語を用いるという、ベースライン的な手法を用いた。１７の人物名を用いて，１７３６件のウェブ文書に対する実験を行った結果、wikipedia に記載されている情報は同姓同名問題に有効であることが確認できた。

以下、まず 2. 章で同姓同名問題の関連研究について紹介する。3. 章では wikipedia の概要について述べる。4. 章で wikipedia の有効性確認のための実験手法の説明をする。5. 章では有効性確認のための実験とその結果について報告する。6. 章では実験結果に対する考察を与える。7. 章で本論文のまとめと今後の課題について論じる。

2. 関連研究

この章では同姓同名問題に関する関連研究を紹介する。

同姓同名問題ウェブマイニング以外の分野でも議論されている。ウェブ以外では論文の著者や引用における同姓同名問題の

解消を行なう方法が盛んに議論されている。

まず人物の伝記的情報を特徴量として利用する手法がある。Deepak [2] らは人物情報として生年月日や職業に注目し、大量のコーパスからこれらの値を抽出するテンプレートを自動的に作成した。Gideon [3] らはこのテンプレート作成法を用いて類別対象のウェブページから人物情報を抽出し、ページに含まれた人物情報をベクトルの成分としてウェブページをクラスタリングする手法を提案した。

次に人物名の含まれるページ同士の類似度を URL などから設定しクラスタリングする手法がある。Bekkerman [4] らは同姓同名な人物名を含むウェブページ同士の距離をリンクと URL の類似度で定義した。URL の類似とはドメインの重複する部分が大きいほど近いページとみなす事である。これらの距離の定義を用いて階層的クラスタリングを行なう手法を提案した。Al-Kamha [5] らは名前をクエリとして与えた検索サイトからの出力ページに、URL 類似度とページに含まれる連語の種類で特徴量を与え、これを同一人物でグループ化する手法を提案した。

そして、人物名と共起する他の情報によりその人物を判別する手法がある。他の情報としてよく使われるのが他の人物名や組織である。Fleischman [6] らはタグ付きデータから人物名と共起する単語のペアを学習し、人名と単語のペアの生起確率を事前に計算しておく事で、与えられた文章が指す人物を特定する手法を提案した。佐藤 [7] らは人物の所属する組織や友人関係といった、実際の世界の人間関係がウェブ空間にも反映されると考え、ある人物名と共起する人名や組織名から、ページに含まれる人物名が指す人物を特定する手法を提案した。

また、同姓同名である人物をクラスタリングし、ユーザーに分かりやすく表示する試みも行なわれている。Xiaojun [8] らは人物の電話番号や住所などを元に、同姓同名と思われる人物のページをクラスタリングしてユーザーに提示するシステムを作成した。

3. Wikipedia について

wikipedia とはインターネット上で百科辞典を作成しようとするプロジェクトである。米国 wikimedia 財団によって運営されるプロジェクトであり、GFDL ライセンスに基づいて公開されている。

フリーのウェブサービスを提供する事で、辞書の記述がインターネット上の任意の人物により自由に加筆訂正できるシステムである。wiki の特徴であるウェブブラウザ上から容易に内容を変更できる機能を利用する事で、参加者が誰でも自由に内容を編集する事ができる。

そのため内容が多岐に渡り、従来の辞書にない専門性の高い用語新言葉や新しい事象が記載されている。これが一般の辞書との大きな違いである。wikipedia は辞書の形状を意識して作られており、一般的な wiki に存在するユーザーのコメントや議論中の話題は別ページに移されている。そのためフォーマットに統一性があり機械処理に向いている。

wikipedia には人物名の項目も存在する。人物名の項目には

図 1　wikipedia のページ例

略歴や職業などの、人物を特徴付ける情報が記述されている事が多い。もし同姓同名の人物が存在する場合には、各個人に個別の項目が与えられる。例えば高橋英樹という人物名には、野球の高橋英樹と俳優の高橋英樹の二つの項目が用意されている。このような人物名の項目を利用して同姓同名人物を判定しようと言うのが、我々の基本的な手法である。

4. 提案手法

4.1 問題設定

本論文で解決しようとしてる同姓同名問題は次のように定義される。

ある人物名 N に対して同姓同名である人物が複数存在したとする。このとき、N という文字列を含むウェブページ集合を $P = \{p_1, p_2, \ldots\}$ と置く。これらを人物名 N の人物情報ページと呼ぶ。これら人物情報ページは、検索語 N を用いてウェブ検索を行ったときに得られる検索結果に相当する。

wikpedia の人物名 N に関する項目ページ集合を $W = \{w_1, w_2, \ldots, w_m\}$ と置く。N には同姓同名人物が複数存在するため、一般には $1 < m$ である。これらを N の人物説明ページと呼ぶ。また、wikipedia に項目ページを持つ人物を項目保持者と呼ぶ。

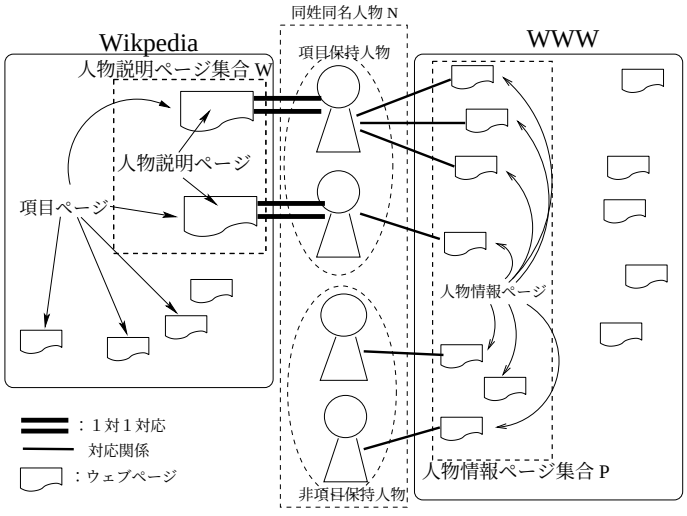

図 2　用語の説明

本論文で解くべき問題は

> 人物情報ページ p_i に含まれる人物名 N が、m 個のどの人物説明ページに対応する人物であるか、または適切な人物説明ページが存在しないかを判定する。

これは人物情報ページ p_i の m+1 値分類問題となる。

用語について図 2 に示した。

4.2 ウェブページ間の類似度にもとづく同姓同名問題の解法

本論文では人物情報ページ $p_{i=\{1,\ldots\}}$ と人物説明ページ $w_{j=\{1,\ldots,m\}}$ の間の類似度にもとづいて、同姓同名問題を解く。ある人物情報ページ p_k を分類する手順は以下のとおりである。

（1） 前処理：人物情報ページ p_k と、m 個の人物説明ページ w_j の全てを特徴単語ベクトルに変換する。

（2） 項目保持者分類：ある p_k に対して最も類似度 sim の高い人物説明ページを w_l とする。この時、p_k に含まれる人物名 N は人物説明ページ w_l に対応すると判断する。

（3） 非項目保持者判定：ある人物情報ページについて全ての人物説明ページと類似度を比較する。類似度は特徴単語ベクトルのコサイン距離で与えられ、$sim(p_k, w_j)$ と表す。もし全ての sim がある閾値 t を超えていなければ $(max\{sim(p_k, w_j)|j = 1, \ldots, m\} < t)$、$p_k$ に適切な人物説明ページは W 中に存在しないと判断し、p_k に含まれる人物名 N は非項目保持者であるとする。

4.3 ウェブページの特徴単語ベクトルの作成

本節では人物情報ページに出現する単語の種類と出現数を利用してページの特徴単語ベクトルを作成する手法を論じる。

wikipedia 中に含まれる単語には、人物説明ページが指す人物を特徴づける単語とそうではない単語がある。例えば「所属」という単語はあまり特徴がない単語である。ほぼ全ての有名人はなんらかの組織に所属している旨が記述されているためである。逆に「野球」や「テレビ」という単語は野球選手や俳優のページに特有の単語であり、学者や政治家のページには含まれにくい。結局、人物を特徴付ける単語とは「他の人物の説明ではあまり用いられない単語」と言う事ができる。

そこで人物を特徴付ける単語に重みをつけ、人物説明ページの特徴単語ベクトルを作成するため、人物情報ページ間の単語の偏りに着目する。例えば二つの wikipedia ページ w_A、w_B において、名詞 n_1、n_2、$n3$ が表 1 の個数出たとする。この場合、

表 1　wikipedia 中における単語の出現回数の例

	w_A	w_B
n_1	1	4
n_2	2	0
n_3	6	8

n_1 は w_B に、n_2 は w_A に偏って出現していることになる。このような名詞は先の例の「野球」や「テレビ」のような単語であり、それぞれの人物説明ページ特有の単語であると言う事ができる。一方、n_3 は双方ともに最も出現回数が多いが、特にどちらかの人物説明ページに偏って出ているわけではない。これは先の例の「所属」という単語に当たり、どちらかに特有の単語とは言えない。

このことから、人物説明ページにおいて各単語の出現数を特徴単語ベクトル成分の重みとして置く事で、各人物情報ページによって異なるベクトルを作成する事ができると言える。出現数に偏りがある単語はベクトル空間において異なる方向を示す役割を果たし、ともに出現数が大きい単語はベクトル空間で似たような方向を示すからである。結局、各人物説明ページは実世界の各人物に対応するため、各人物そのものを特徴づけると言う事ができる。よって本手法の各人物説明ページに含まれる名詞を特徴単語ベクトルの成分、その個数を成分の値として用いる。

特徴単語ベクトルの成分を決定したので、これに基づいて各人物説明ページの特徴単語ベクトルを作成する。先に述べたように、人物説明ページの特徴単語ベクトルの値は名詞の出現回数である。よって人物説明ページ中より特徴単語ベクトルの各成分の出現回数を調べ特徴単語ベクトルの値とする。

次に人物情報ページについて述べる。人物情報ページは、検索サイトなどから URL を得る事でユーザーが入手するものである。これらのページには人物名の文字列と人物名に関する文章が書かれていると考えられる。よってこちらについても wikipedia ページと同様に特徴単語ベクトルの成分である単語の出現回数を調べ、ベクトルの値とする。

5. 実　　験

5.1 実験データ

我々は同姓同名問題の実験のために同姓同名の人物が存在し、かつ wikipedia に項目ページを持つ人物名を１７種類選択した。これらの内訳は wikipedia に項目ページを２つ持つ人物名が１４種と、３つ持つ人物名３種、合計１７種類である。実験対象とする人物名は、以下の項目を重視して選択した。

- 名前の文字列が完全に一致する（必須）
- 表記にブレがある人物名は使わない
- 似たような分野で働いている人物を優先的に採り入れる

そして、これらの人物名を持つウェブページを実験用ページセットとした。この実験用ウェブページは検索エンジンにクエリとして人名のみを与えた結果から取得した。全てのページは人手により内容を吟味し、ページ中の人物名が指す実世界の人物のタグ付けが済んでいる。また、検索サイトの結果でしばしば見受けられる、時系列の違いによる同一ページの重複も人手によって除外されている。

実験用ページセットについて、人物名や各人物毎のページ数量など詳しい内容を表 2 に示す。表 2 における識別情報とは、人物名と組で用いる事で各人物を区別するものであり、wikipedia に習い本人の職業名を利用している。番号とは識別情報に対応した番号で、以降の表では簡略化のために識別情報ではなくこの番号を記載している。その他という項目には、wikipedia に項目ページを持たない非項目保持者のウェブページが含まれる。これは検索サイトから得られた結果ページにタグ付けをしていた際に、項目ページ以外の人物のウェブページであった際に逐

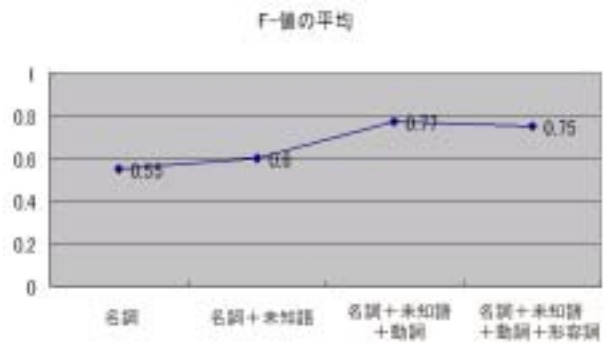

図 3　F-値の平均値の比較

次保存したページで、この中に複数の人物情報ページが含まれる。また、テストページは手動で集めているため、有名人であるほどウェブ中に存在するページの個数が増え、テストページを集めやすかった。そのためページ数は人物が有名であるほど大きくなる傾向がある。ウェブにおける人物の有名度のちょっとした参考にしていただきたい。

表 5.1 には各人物の人物説明ページから作成された特徴ベクトルの成分中から、各人物を特徴付けていると思われる要素の単語を挙げた。

5.2　実験１ 項目保持者の特定

実験１では 4.2 節における「手順 (2)：項目保持者分類」の評価を行なう。実験１では予めに非項目保持者を除外したテストデータを用意して、手順 (2) の作業を実行した。実験はそれぞれの人物名毎に行なった。

また、特徴ベクトルの要素として用いられる単語の種類が結果にどの程度影響するかも同時に考察するために、以下の種類の単語の組合せを用いてそれぞれについて実験を行なった。

- 名詞
- 名詞＋未知語
- 名詞＋未知語＋動詞
- 名詞＋未知語＋動詞＋形容詞

ただし未知語とは、文を形態素分解する際に形態素分解器が持つ辞書に登録されていない単語である。これらの単語は例えばサッカーチームの名前などの固有名詞である事が多い。

表 4 に実験１のうち、名詞＋未知語＋動詞を特徴ベクトルとして利用した場合の結果を示す。その他の単語の組合せも値の違いはあれど基本的に同様の傾向を示す。表は次のような構成になっている。列 (a) は入力ページ総数であり、計算機に与えられた実験対象の人物情報ページの総数である。列 (b) は各人物の人物情報ページの数である。列 (c) は計算機がその列の人物のページと判定して出力した結果である。列 P、R、F は当該列の人物に対する Precision、Recall、F-measure を示している。

また、図 3 に各単語の組における F-measure の平均値を示す。これは各単語の組の表における F-measure に該当する値を平均したものである。

5.3　実験２ 非項目保持者の判別

この節では 4.2 節における手順「(3)：非項目保持者判定」

表 2　実験対象となるデータセット

名前	識別情報	番号	ページ数	特徴
江川卓	文学者	0	53	ロシア文学者
	野球	1	50	野球選手、解説者
	その他	2	5	
岩崎良美	歌手	0	49	女優業も行なう
	アナウンサー	1	24	福井 TV
	その他	2	9	
増田俊朗	作曲家	0	31	ゲーム曲作成
	ラジオ	1	23	関西で放送
	その他	2	52	
小川知子	アナウンサー	0	50	TBS 所属
	女優	1	50	TV ドラマ出演
	その他	2	13	
佐々木正洋	青森放送	0	11	アナウンサー
	TV 朝日	1	52	アナウンサー
	その他	2	23	
鈴木俊一	都知事	0	38	都庁を移転
	衆議院議員	1	51	岩手県出身
	その他	2	38	
石川賢	漫画家	0	36	ゲッターロボ作者
	野球大洋	1	2	投手、引退
	野球中日	2	22	投手
	その他	3	0	
中田浩二	声優	0	33	俳優活動も行なう
	サッカー	1	49	日本代表選手
	その他	2	12	
小野大輔	フットサル	0	50	日本代表選手
	声優	1	55	出演作品多数
	その他	2	7	
白井貴子	歌手	0	51	ロック系
	バレー選手	1	48	オリンピック出場
	その他	2	0	
高橋英樹	俳優	0	48	時代劇役者
	投手	1	4	広島所属
	その他	2	18	
吉田恵	アナウンサー	0	47	めざましテレビ
	サッカー	1	26	J リーグ選手
	その他	2	5	
伊藤博文	首相	0	52	趣味が将棋
	棋士	1	47	将棋教室開催
	その他	2	2	
前田愛	文芸評論家	0	40	都市小説論
	声優	1	47	歌手活動も行なう
	女優	2	53	声優にも挑戦
	その他	3	40	
坂本太郎	歴史家	0	58	坂本太郎著作集
	特撮監督	1	48	アニメ監督も
	その他	2	25	
鈴木健	アナウンサー	0	8	スポーツ実況
	情報系	1	21	仮想通貨
	内野手	2	34	ヤクルト所属
	その他	3	61	
田村亮	俳優	0	52	兄弟も俳優
	お笑い	1	48	ロンドンブーツ
	その他	2	12	
合計			1736	

表 3　人物名のベクトルにおける特徴的な名詞の例

名前 （ベクトル次元）	番号	特徴的なベクトル成分
江川卓	0	社、潮、文学、 ロシア、文庫、教授
(703)	1	江川、野球、こと 投手、巨人、阪神
岩崎良美	0	年、賞、音楽 新人、祭、タッチ
(353)	1	アナウンサー、福井
増田俊朗	0	音楽、番組、作曲
(227)	1	木曜、ラジオ、水曜
小川知子	0	アナウンサー、ニュース
(314)	1	歌手、女優、映画 テレビ、デビュー
佐々木正洋	0	朝日放送、青森、八
(191)	1	アナウンサー、朝日
鈴木俊一	0	知事、東京、推薦
(397)	1	善幸、堤、岩手
石川賢	0	作品、石川、原作 永井、魔
(694)	1	山梨、大洋、球団
	2	中日、野球、選手
中田浩二	0	アニメ、江戸、探偵
(474)	1	出場、選手、鹿島 日本、移籍、監督
小野大輔	0	選手、移籍、サッカー
(383)	1	編集、声優、出演
白井貴子	0	出演、音楽、本人
(366)	1	バレーボール、リーグ
高橋英樹	0	俳優、役、映画
(316)	1	野球、選手、広島
吉田恵	0	大学、テレビ
(277)	1	得点、試合、選手
伊藤博文	0	伊藤、年、博文 明治、韓国、内閣
(596)	1	クラス、段、将棋
前田愛	0	文学、評論、文芸 都市、柳、北
(582)	1	アニメ、歌、声優 主題、ラジオ
	2	共演、姉妹
坂本太郎	0	歴史、研究、国史
(267)	1	戦、隊、監督 レンジャー、テレビ
鈴木健	0	テレビ、スポーツ
(335)	1	エンジニア、伝播
	2	打率、本塁
田村亮	0	ドラマ、兄弟、京都
(421)	1	淳、吉本、殿堂

表 4　実験 1 の結果（名詞＋未知語＋動詞）

江川卓	0	103	53	0.83	0.72	0.77
	1		50	0.74	0.84	0.79
岩崎良美	0	73	49	0.83	0.39	0.53
	1		24	0.40	0.83	0.54
増田俊朗	0	54	31	0.81	0.42	0.55
	1		23	0.53	0.87	0.66
小川知子	0	100	50	0.59	0.80	0.68
	1		50	0.69	0.44	0.54
佐々木正洋	0	63	11	0.35	0.55	0.43
	1		52	0.89	0.79	0.84
鈴木俊一	0	89	38	0.67	0.87	0.76
	1		51	0.88	0.69	0.77
石川賢	0	60	36	0.84	0.72	0.78
	1		2	0.07	0.5	0.12
	2		22	0.62	0.82	0.71
中田浩二	0	82	33	0.90	0.88	0.89
	1		49	0.92	0.94	0.93
小野大輔	0	105	50	0.82	1	0.90
	1		55	1	0.80	0.89
白井貴子	0	99	51	0.80	0.96	0.88
	1		48	0.95	0.76	0.84
高橋英樹	0	52	48	0.94	0.71	0.81
	1		4	0.95	0.76	0.84
吉田恵	0	73	47	0.79	0.96	0.88
	1		26	0.93	0.54	0.68
伊藤博文	0	99	52	0.84	0.88	0.86
	1		47	0.86	0.81	0.84
前田愛	0	140	40	0.39	0.90	0.55
	1		47	0.40	0.55	0.46
	2		53	0.42	0.34	0.38
坂本太郎	0	106	58	0.88	0.98	0.93
	1		48	0.98	0.83	0.90
鈴木健	0	63	8	0.23	0.88	0.37
	1		21	0.46	0.86	0.54
	2		34	0.78	0.41	0.54
田村亮	0	100	52	0.88	0.40	0.55
	1		48	0.59	0.93	0.73

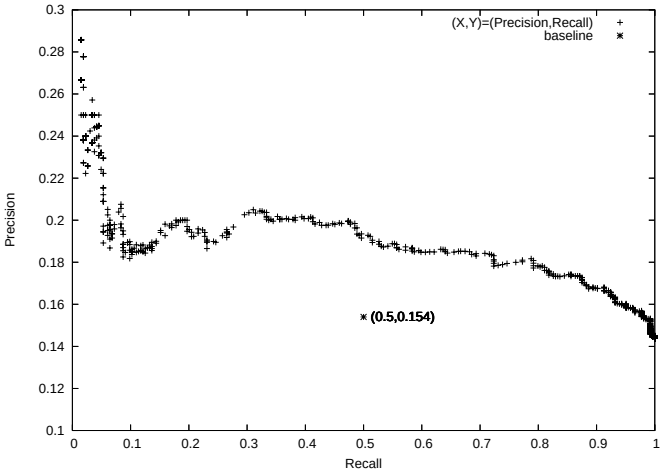

図 4　実験 2 結果: Precesion,Recall の関係

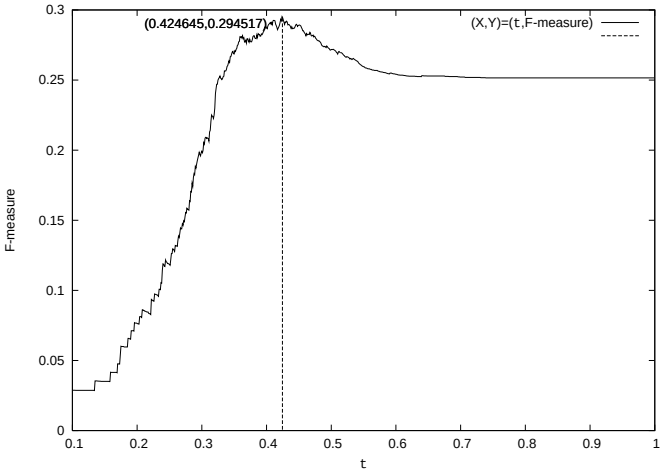

図 5　実験 2 結果: 閾値 t と F-measure の関係

の実験と評価をする。手順 (3) では閾値 t を利用するがこの値は実験的に与える必要がある。この実験では t の値を変えながら、実験用ページセットを非項目保持者と項目保持者という二つのクラスに分類し、各値毎に非項目保持者に対する Precison、Recall、F-measure を求める。θ の値は 0 から 1 の範囲で、0.001 刻みに変動させ、1000 個の値を算出した。

Precision を X 軸、Recall を Y 軸にとったグラフを図 4 に示す。また、baseline として全くランダムに分類を行なった場合の Precision と Recall の値を図中に示した。この場合の値は $(\mathrm{Precesion}, \mathrm{Recall}) = (0.5, 0.154)$ である。θ を X 軸、F-measure を Y 軸にとったグラフを図 5 に示す。

6. 考　　察

6.1　実験 1 に対する考察

全体として結果は良好といえる。これは wikipedia 中の人物説明ページに含まれている情報の質が高い事を示す。人物の職業が違う組合せの場合には高い値を示している。「江川卓」（文学者、スポーツキャスター）、「小野大輔」（フットサル、声優）、「中田浩二」（声優、サッカー）などである。これらの人物は有名であるため、人物説明ページのみならず人物情報ページのほうも各個人の事をよく理解して記述されている。そのためページに出現する単語の一致率が高いのが理由である。一方、「鈴木健」のような正解率が低いものもある。鈴木健-1 は wikipedia では職業：エンジニアとなっている。しかし、実際の活動はベンチャー起業家であり、学者である。このように、wikipedia には内容にミスがあることがある。これが正式な辞書ではないデメリットである。

さて、本実験でも他の同姓同名問題の手法と同様に、同名でかつ同一の職業の人物が居る場合に正解率が下がるという問題が発生している。これは対象とする人物の背景情報が似てくる事に起因する問題である。しかし本実験ではこのような条件が重なっても、大幅に正解率が下がる事はない傾向にある。

「佐々木正洋」を例に挙げる。彼ら二人はともにアナウンサーである。佐々木正洋-0 は、過去に青森朝日放送、現在は八峯テレビに所属している。まだ若いため情報説明ページにはアナウンサーである事と所属局の履歴程度しか書かれていない。佐々木正洋-1 はテレビ朝日のアナウンサーであり、かなりのベテランである。そのため情報説明ページがとても充実している。特徴量的に見ると、佐々木正洋-0 は「青森」「八峯」という単語を持っているが、それ以外はこれといった単語を持っていない。そして単語のほとんどが佐々木正洋-1 のページに包含されている。ところが、ここまで類似性がある人物にもかかわらず佐々木正洋-0 の分類が全く出来ていないわけではない。これは先に挙げた二つの単語が、二人の間の違いを極めて的確に捉えているからである。この事は、wikipedia の情報が似通った背景情報を持つ人物を特徴付けるためにも有効である事を示している。

この事はさらに厳しい条件を持つ「前田愛」についても同様である。前田愛-1 は声優でありアニメやゲームに出演している。前田愛-2 は女優でありテレビに出演している。一般的に「女優」と「声優」という職業は、共にテレビに出演することなどから人物情報ページに含まれる単語が似通っている。そのためこの職業のペアを区別するのは通常より少し難易度が高い。そして「前田愛」の場合は更に以下の条件がついている。前田愛-1 は女優活動もすることがある。一方の前田愛-2 は、最近有名なアニメ映画の声優に挑戦することになり、ニュースサイトや個人の blog などでかなり注目を集める事になった。この状況では、ウェブページを人間が見てもすぐに分類する事は難しい。実際、本論文の実験データは自分で作成したが、その際この人物名のテストデータ収集に最も苦労した。ところが「前田愛」の識別もある程度成功している。なぜなら、「前田愛」特徴単語ベクトルには、ゲームソフトの名前の一部や、アニメのキャラクターの名前が含まれており、これが前田愛-1 を特徴付ける単語として働いたからである。このような通常の辞書からは手に入らないようなマニアックな情報が取得できるのも wikipedia の特徴であるといえる。

図 3 からは、単語の組合せによる精度の違いがあることが分かる。これは単語の種類による違いではなく、単語の種類が増えたために起こる特徴ベクトルの次元の増大による精度の向上

表 5　複数のベクトルで大きな値を持つ名詞のリスト

名詞	出現時のベクトル成分値の例	(h)
年	29、6、5、12	37
月	4、29、6	16
日	4、4、2	19
先	5、2、2	2
編集	10、5、3、20、12	23
リンク	5、4、5、3	19
更新	5、3	10
項目	4、6	9

と考えられる。また、最後の形容詞の部分で値が減少する理由は、同じような形容詞が大抵どの人物のページにも使われており、人物を特徴付ける単語として向いておらず、むしろノイズになっているからだと考えられる。

6.2　実験２に対する考察

実験結果の図 4 について。ランダムな分類は上回っているため、ある程度分類の成果は出ていると言える。しかし Precision の値が全体的に低い。また Recall 値の大部分で Precesion 値が変わらない。つまり閾値 t の値を変更しても分類精度にあまり変化が見られないと言う事である。よって現状の手法では、閾値を与えて分類を行なう事は現実的ではない。

非項目保持者のページを人手で分類する事は可能なため、これらのページの持つ内容が人物説明ページと比べて違いがあるのは明らかである。よって、分類精度が悪いのは、ベクトル成分が非項目保有者の特徴を捉えていないため、不適切な特徴単語ベクトルが算出されているからである。

我々の手法で非項目保持者の特徴を捉えられない理由として考えられるのは、wikipedia 中の人物説明ページに含まれるありふれた単語への対策を施していないからである。現在の手法では、ベクトル成分に単語の出現数をそのまま与えている。これでは人物によらず出現頻度の高い単語が出てきた場合にも、区別する事無く高い値を与えてしまう。4.3 節でも述べたように、このような単語は特徴を表す単語として有用ではなく、ベクトル間の類似度を適正以上に大きくする原因となる。

実際に人物情報ページのベクトル成分の中から、複数の人物の人物情報ページベクトルで、比較的大きな値を持っているものを抜き出した。(表 5) ただし表中の (h) は、この成分が自身のベクトル成分のうち上位 20 番目以内に入っている人物の数である。表 5 には二つのグループに分けられる。まず上から４番目までの時間を表す単語である。これらの名詞は人物の生年月日などが wikipedia の文章中に大量に記述されるため値が高くなっている。特に「年」は全ての人物で最高値に近い部分にある。残りの単語は wikipedia のフォーマットに含まれるものである。値の最も大きな「編集」について。今回はアニメの監督などで「編集」という言葉が重要な成分になると考えたため除外する事をしなかった。人物説明ページの特徴単語ベクトルにおける成分の値の分布の例を示した（図 6)。これは１７人のうちの４人について、人物情報ページに与えられた特徴単語ベクトルの成分の値を昇順に並べて表示したものである。図 6 から分かるように、ベクトル成分の大部分は 3 以下である。そのため表 5 に挙げた単語は他のベクトル成分に比べ非常に高い値を示しているといえる。

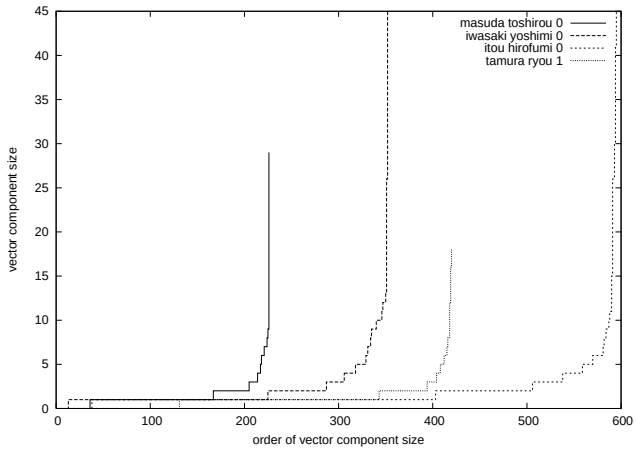

図 6　特徴単語ベクトルの成分のサイズの分布の例

「年」、「月」などの単語は人物を扱うページなら大抵入っているありふれた単語である。これは wikipedia にも共通して言う事ができる。よって、wikipedia の人物説明ページに書かれた情報をうまく利用するためには、このような単語に tfidf などの単語の出現頻度による重み付けを与えてベクトルにおける成分値を低く押え、影響を少なくすればよいと考えられる。

6.3　全体に対する考察

最後に各人物説明ページの特徴単語ベクトル間のコサイン距離の表 6 を載せる。この表より、wikipedia ページベクトル同士の特徴単語ベクトル自体も離れている事が分かる。これはそれぞれのページの内容が独特のものである事を示す。逆に F 値の低い「石川賢」や「鈴木健」について同姓同名人物の間での特徴単語ベクトルの距離が近い事が分かる。

7.　まとめと今後の課題

本論文ではウェブ空間における同姓同名問題解決の手法として、wikipedia に含まれる有名人の項目ページに着目した。我々は同姓同名な人物名に対する特徴単語ベクトルを作る手法として、人物説明ページに含まれる単語に注目し、単語の出現数を特徴単語ベクトルとした。そしてベクトル間のコサイン距離を使って分類をする事にした。

また、評価実験のために人手でタグ付けした１７３６ページのテストページを用意した。

そして wikipedia に含まれる情報が人名の多義性解消に有効であるか検証するために、二つの実験を行なった。実験１ではページの特徴単語ベクトルを使って、同姓同名な項目保持者のそれぞれをクラスとし、テストページを分類を行なった。その結果、wikipedia は項目ページを持つ人物同士なら同姓同名問題の解決に有効であると分かった。特に同一職業や、背景情報が似通った人物の分類にも役たつ事が分かった。実験２ではページの特徴単語ベクトルを使って、テストページを項目保持者と非項目保持者の二つのクラスに分類する実験を行なった。その結果、単純な単語出現数だけではテストページを項目保

持者と非項目保持者に分割できない事が分かった。だがこれは wikipedia の問題ではなく、ベクトルの作成方法に問題があると考えられる。

以上より、本論文では wikipedia は同姓同名人物の分類に対して有効な情報源であることを実験で確認した。しかしながら、特徴量の作成方法に一部問題があるなど、wikipedia に含まれる情報を完全に利用したとは言いがたい。

したがって、今後はより wikipedia に含まれる情報を引き出せるような手法を研究していきたいと考えている。まずは、その他の人々と wikipedia に含まれる人々の分類を成功させたい。そのために、今後は単語以外の要素をページの特徴ベクトル成分に含むんだ上で再実験する事を検討している。

また wikipedia 中のテキスト情報以外の利用も考えている。現在、本研究ではテキスト情報のみを利用して、リンク情報は用いていない。しかし、先行研究ではリンク情報が有効であると示されている。そのためリンク情報を採り入れる事も今後の課題としたい。

表 6　人物説明ページ間のコサイン距離

		1	2
江川卓	0	0.17	
岩崎良美	0	0.26	
増田俊朗	0	0.38	
小川知子	0	0.24	
佐々木正洋	0	0.52	
鈴木俊一	0	0.37	
石川賢	0	0.20	0.19
	1	*	0.52
中田浩二	0	0.22	
小野大輔	0	0.39	
白井貴子	0	0.22	
高橋英樹	0	0.33	
吉田恵	0	0.16	
伊藤博文	0	0.22	
前田愛	0	0.26	0.26
	1	*	0.19
坂本太郎	0	0.18	
鈴木健	0	0.53	0.31
	1	*	0.29
田村亮	0	0.34	

文　献

[1] Christopher D. Manning and Hinrich Schütze, “Foundations of Statistical Natural Language Processing”,The MIT Press.

[2] Deepak Ravichandran and Eduard Hovy, “Learning Surface Text for a Question Answering System”, In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 2001.

[3] Gideons S. Mann and David Yarowsky, “Unsupervised Personal Name Disambiguation”, In *Proceedings of CoNLL ’03*, Edmonton, Canada, 2003.

[4] Ron Bekkerman and Andrew McCallum, “Disambiguating Web Appearances of People in a Social Network”, In *Proceedings of WWW ’05*, Chiba, Japan, 2005.

[5] Reema Al-Kamha and David W. Embley, “Grouping Search-Engine Returned Citations for Person-Name Queries”, In *Proceedings of WIDM ’04*, Washington, DC, USA, 2004.

[6] Michael Ben Fleischman and Eduard Hovy, “Multi-Document Person Name Resolution”, In *Proceedings of ACL-42,Refarence Resolution Workshop*,2004.

[7] 佐藤進也, 風間一洋, 福田健介, 村上健一郎, “実世界 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, Vol.46. No.SIG8(TOD 26),2005.

[8] Xiaojun Wan, Jianfeng Gao, Mu Li, Binggong Ding, “Person Resolution in Person Search Results:Web Hawk”, In *Proceedings of CIKM’05*, Bremen, Germany, 2005.

[9] Hongkun Zhao et al., “Fully Automatic Wrapper Generation For Search Engines”, WWW-05, 2005.