修士論文

Wikipediaを用いた人物名の曖昧性解消

平成19年2月2日 提出

指導教員： 喜連川 優 教授
情報理工学系研究科 電子情報学専攻
56441 吉田 康浩

目 次

図目次

概要

本論文ではウェブ空間における人物名の曖昧性解消、即ち同姓同名問題の解決を行なうために、wikipediaという新しい言語資源に注目し、それが有効か検証する実験を行なった。

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 Disambig uation) 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。

第1章　はじめに

この章では、まず人物名の曖昧性解消問題、即ち同姓同名問題について説明する。次にウェブ空間における人物名の曖昧性と、計算機による曖昧性の解消が何故必要であるかを論じる。そして解決するための手法として我々が提案するwikipediaを用いた手法について簡単に述べ、最後に本論文の残りの章について概略を与える。

1.1　人物名の曖昧性解消とは

本論文で論じる人物名の曖昧性とは、いわゆる同姓同名の事を指す。同姓同名とは、ある人物と名前が同じ異なる人物が居る状態のことであり、同じ名前の人物の事を同姓同名な人物という。一般的に同姓同名という場合には、名前を文字に記した場合のつづりの一致や、名前を発音した時の音の一致が考えられる。また会話や文章では名前を全て書かず、一部のみを書く事が多い。例えば論文などで引用を記述する際に、論文の筆者の名前のみを書く事が多い。この時名前が一致する他の人物が文中に出てくるなら、これらの人物の間には同姓同名な関係があると言える。

次に同姓同名問題について説明する。生活を送る際や事務処理をする際に、同姓同名な人物がいる場合と様々な不具合がある。最も分かりやすい例は、学校での点呼の際に同じクラスに同姓同名な人物が居る場合である。この場合名前を呼ぶだけでは、どの人物を呼んだのか特定できない。また病院でのカルテの整理など事務処理の際に、名前だけを書類のインデックスにしている場合に書類を取り違えるという不具合が起きる。これらの不具合は、人物に対してユニークであることが期待されている名前がユニークでない事に起因する。この不具合を解決するためには、それぞれの人物名が指し示す人物を周りの状況から推定する必要がある。このような人物名と実際の人物の不一致と、それを解消する手法を与えるというタスクを、総括して同姓同名問題という。よって同姓同名問題を解決するということは、人物名の曖昧性を解消するという事に他ならない。

同姓同名は日本人に限らず、他の言語でも起きる現象である。むしろ名前に使用する文字の種類や、名前のバリエーションが少ない英語やロシア語など、ラテン語系の言語を使用する人々の間でよく起きる。また英語では”Joseph”

を"Joe" と呼ぶといった、正式な名前を愛称で記述する場合があり、これがさらに名前のバリエーションを減らす要因となっている。そのため同姓同名問題はラテン語圏で古くから注目されてきた。

同姓同名問題は自然言語処理の分野で研究されてきた。この分野では同姓同名問題のひとつとして、同じ著者名を持つ論文が同一人物によって書かれたものか否かを判別するタスクが論じられている。またこの問題の発展系として、論文に含まれる人物名が実世界のどの人物を指すか推定する、というタスクがある。人物名を単語として考えると、人物名が指す実際の人物はその単語の意味であると言える。そして同姓同名問題の解決とは、ある単語が複数の意味を持つ時に、対象の語がどの意味で使われているかを判別するタスクであると言える。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。自然言語処理の分野では、通常の単語に対しての多義性解消をするために、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提案されている [1]。しかし、こうした手法を同姓同名問題の解決に適用することはできない。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。これらの手法は多義性解消の対象となる単語が、既知の物であるとしている。そして単語を網羅したタグ付きコーパスや国語辞典の定義を利用して問題解決をする。しかし名前の場合、その名前を持つ人物は無限に存在し、また人物は時間を追う毎に増えていくからである。そのため同姓同名問題の解決には、多義性解消とは少し違ったアプローチを取ることになる。

1.2 ウェブ空間における人物名の曖昧性

まずウェブ空間という単語の意味と、ウェブページの構成要素について説明する。ウェブ空間とは、インターネット上に存在するウェブページの集合の事である。各ページに含まれるコンテンツは一つの文章と考える事ができる。また各ページ間のリンクも構成要素の一つである。つまりウェブ空間は、文章とその間に張られたリンクから構成されているものである。ウェブページの各構成要素とそれらの要素を利用した曖昧性解消法の既存研究について、詳しくは 2 章で述べる。

次にウェブページ上における人物名の曖昧性について述べる。あるウェブページ中の文章中に単語として人物名が出てくる場合を考える。例えば、研究者の発表論文紹介サイト中に出現する引用文献の著者名や、ブログの映画レビュー中に含まれる出演俳優の名前である。このような記述の場合、人物名は現実世界のある人間ただ一人を指している。ところが人物名に対して同姓同名な人物

が複数居る場合、人物名だけでは現実世界のどの人物を指すか分からなくなる。これがウェブ空間における人物名の曖昧性である。

さて、人間がウェブページを閲覧する際に、曖昧性を含む人物名が混乱を引き起こす事は稀である。なぜなら人間は文章中に出現した人物名の周囲の情報から、人物名が実際のどの人物を指すか推定する事ができるからである。例えば、タレントの木村拓哉について記述されたページ中に「木村」という人物名が出てきたとする。「木村」という名字を持つ人物は木村拓哉以外にも居るが、この場合読者は「木村」は間違い無くタレントの木村拓哉を指しているものと考える。このような場合、ページの読者は無意識のうちに文脈など回りの状況や、周囲の単語から連想される情報から人物名が現実世界のどの人物を指すか判定していると言える。一方、曖昧性判定を計算機に行なわせる事は容易ではない。何故なら、計算機で文脈を完全に把握したり、ページ中に出現する単語全ての知識を事前に学習しておく事が難しいからである。

それでもなお機械による人物名の曖昧性解消が求められている理由の一つに、検索サイトの利用にまつわる問題がある。現在ウェブを利用する際には、まず検索サイトを用いて目的のページを捜し出すのが一般的である。検索サイトでは、ユーザーは自身が閲覧したいと思う情報に関連する単語を検索サイトにクエリとして送信する。検索サイトは独自のアルゴリズムを用いて、クエリとして与えられた単語に関係すると思われるページのURLをユーザーに提示する。ここで問題となってくるのが、ユーザーに提示されるURLの量の増大である。近年インターネット上に存在するウェブページは、増大の一途を辿っている。よって、これに同調してクエリとして入力された単語と、関連があるとされるページも増えることになる。大抵の場合、ユーザーの閲覧したいページは検索結果うちのごく一部である。そのため、ユーザーは自分の知りたい情報を含むページを取得するために、検索結果として与えられたURLの内容を逐次調べて目的のページを探す必要がある。

ここである人物に関係するウェブページを検索した場合を考える。この時、検索サイトはキーワードである人物名を含むページが関連性が高いと考える特徴があるため、対象の人物名を含んだサイトを出力する。人物名に同姓同名な人物が居た場合、検索サイトは人物名の一致を最優先するため、ユーザーが調べたい人物とは違う人物のページも出力してしまう。このような人物を検索した場合の検索結果の増大は、ウェブページに含まれる人物名と現実世界の人物の間の対応関係が曖昧な事が原因である。ここで検索結果のウェブページに出現する名前の曖昧性を解消し、現実世界の人物と自動的に対応付ける事ができれば、ユーザーの手間を省く事ができる。このように現在ウェブ空間における人物名の曖昧性を解消する事が求められている。

1.3 ウェブページに出現する人物名の曖昧性解消手法

そこで我々はwikipediaという辞書を利用して、ウェブページに出現する人物名の曖昧性解消を行ない、検索結果における人物名の曖昧性による問題の緩和を行なおうと考えた。ある人物名がWikipediaに記載された同姓同名の有名人であるかどうか検証し、そうならばどの人物か判別し、そうでないならば有名人以外であるとする。これにより検索結果を各有名人とそれ以外の人物に分類する事ができる。ユーザーが有名人のウェブページを知りたい場合は、そのまま対象の有名人に分類されたページを閲覧すれば良く、それ以外の人物のウェブページを知りたい場合にも、有名人のサイトが削除されて個数が少なくなった検索結果からユーザーが捜し出せばよい。

Wikipediaとはインターネット上で作成された百科辞典である。Wikipediaはオンライン操作により、インターネット上の誰もが内容を編集する事が出来る辞書である。そのため、Wikipedia中には通常の辞書にはないような専門性の高い単語や、新たな事象が記載されている。これらの中には有名人の情報も含まれており、有名人の名前そのものが辞書の項目になっている。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。Wikipediaの詳細な説明は3章で与える。

Wikipediaという辞書を用いた曖昧性解消の手法の概要は次のようになっている。ウェブページに含まれる人物名と、ページ中の人物名以外の部分の文章は関連があると考えられる。我々は、特に文章中に出現する単語と、その出現数が人物名と深い関係にあると考えた。例えば、映画のレビューをするページ中で俳優の名前が出てくる場合を考える。俳優が出演する映画のため、映画の名前や配給会社の名前が俳優に関連する単語であると言える。一方、Wikipediaに含まれる各人物の項目ページも一種の文章である。そのため、項目ページに含まれる単語も項目の人物と関係があると考えられる。そして、Wikipediaの項目は現実の人物と1対1対応である事が保証されている。そこでWikipediaの項目ページに出現する単語と、ウェブページの文章中に出現する単語分布を比較する事で、ウェブページがWikipediaの項目ページに最も近いか判別する。ウェブページに含まれる人物名は文章と関係がある事が多いため、結局この作業はウェブページに含まれる人物名がWikipediaの項目ページの人物の誰に最も近いかを判別していることになる。この作業により、ウェブページに出現する人物名が有名人である場合は、どの人物であるか判別できる。また、それ以外の人物である場合も判別する。手法の詳細は4章で与える。

そして、我々は手法の有効性を確認するために、１７の人物名を用いて、１７３６件のウェブページに対する幾つかの実験を行った。その結果、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。実験の詳細は5章で与える。

1.4 論文の流れ

以下に本論文の流れを記す。まず2章で同姓同名問題を扱った関連研究について紹介する。この章ではウェブページではなく、通常の文章も対象とした論文についても扱う。3章ではwikipediaの概要について述べる。この章では、Wikipediaが構築されるシステムや、含まれる情報、また人物を説明した項目ページの特徴について述べる。4章でwikipediaの有効性確認のための実験手法の説明をする。この章では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。

第2章　関連研究

この章では同姓同名問題を取り扱った関連研究を紹介する。先にも述べたように、同姓同名問題は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。したがって紹介する関連研究には、ウェブページに含まれる人物名だけではなく、論文や新聞記事等、文章に含まれる人物名を対象とした手法が含まれる。ここで紹介する関連研究の手法に共通している事は、判別対象を特徴づける特徴量の設定が重要であるという事である。

一般的な同姓同名問題の解決手法は、以下のような構成になっている。

1. 同姓同名な人物に関する特徴量の抽出

2. 特徴量に基づいて対象を分類、もしくはクラスタリング

そのため同姓同名問題の研究は、このどちらかに重点を置いている場合がほとんどである。ここで後者に関しては、クラスタリングの新たな手法の具体的利用法の一例として同姓同名問題の解決を扱っている場合などが多いため、同姓同名問題の研究とは直接関係の無い場合が多いと考えられる。そこでこの章では、前者に関連する研究を取り上げる事にした。

さて、同姓同名問題の研究では、２番目の分類の段階で分類する対象が文章、文章中に出現する名前など、問題設定が論文によってまちまである。そのため人物名の曖昧性解消問題を紹介するには、まずこの部分をはっきりさせる事が重要である。そこで以下に人物名と、それが含まれる文章、ウェブページの関係を整理する (図 2.1)。まず文章における同姓同名問題の発生の必須条件として、文章は必ず対象の人物名を含む。図の場合、二つの文章は共に「吉田」という人物名を含んでいる。そしてこの人物名が同姓同名問題を引き起こしている。問題設定には以下の二種類がある。

- 人物名そのものの曖昧性解消を行なうという問題設定
  これは吉田 A、B、b の三つの人物名がそれぞれ同一人物を指しているのか否かを判定することである。

- 人物名が含まれている文章を分類するという問題設定
  この場合、各文章はそれぞれ一人の人物の話題のみを扱っていると考える。図の左では文章内に「吉田 A」以外に「田中」という人物名を含ん

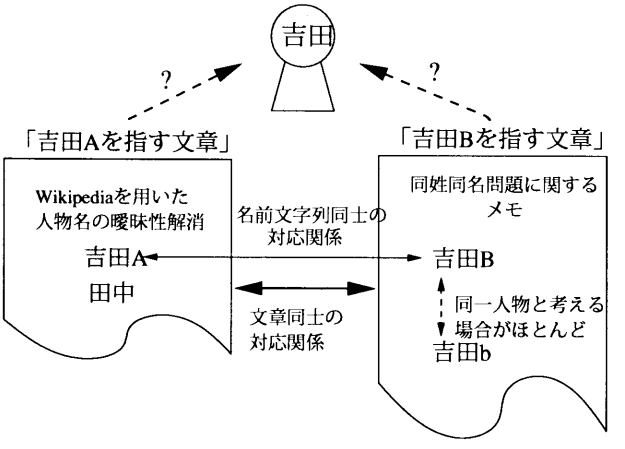


図 2.1: 人物名と文章の関係

でいるが、この文章は「吉田 A」に関する話題を扱った物で、「田中」はその話題の一つに過ぎないと考える。また右では「吉田 B」、「吉田 b」と二つの同じ人物名が登場する。この場合、この二人は同一人物であると考える。そのため、文章と人物は一対一対応となり、人物名が指す人物の判別は、ページが指す人物の判別をすればよいことになる。

大部分の同姓同名問題は、問題設定として後者を採用している。今回紹介する関連研究においても、Fleischman[6] ら以外の研究では後者を採用している。

2.1 人物の伝記的特徴を利用した手法

まず人物の伝記的 (biographical) 情報を特徴量として利用する手法がある。このような伝記的情報を用いた手法は、同姓同名問題の解決手法としては最も古典的な部類である。ここでは伝記的情報を利用した手法の例として、伝記的情報の抽出手法を提案した Deepak[2] らの研究と、それを応用して同姓同名問題の解決手法を提案した Gideon[3] らの研究を紹介する。

伝記的情報とは人物の生年月日や職業、生まれた場所などの情報である。これらは歴史の教科書で、歴史上の人物の伝記と共に出現するような情報である事から伝記的な情報と呼ばれるようである。例えば、物理学者のアルバート・アインシュタインの伝記とは以下のようなものである。

```
Albert Einstein (1879 - 1955) was a German-born theoretical
physicist who is widely considered one of the greatest
physicists of all time. While best known for the theory
of relativity (以下省略)
```

伝記における人物情報の記述で特徴的なのは、

"Albert Einstein (March 14, 1879 - April 18, 1955)"

"Einstein (1879)"

"Albert Einstein (1879-1955)"

"Albert Einstein, March 14, 1879"

"Albert Einstein (German)"

のように、名前の前後に人物にまつわる情報を記す事である。Deepak らはこれらの伝記的情報を文章中から抽出するために、括弧書きを用いた記述のテンプレートを自動的に作成する手法を提案した。

彼らは上に示したような伝記的情報の記述法には一定の法則があると考えた。そこで彼らは生年月日が既知である歴史上の有名な人物について、人物に関する文章から人物名と生年月日の周囲に現れる記述を抽出した。そこで、記述のテンプレートのみを取得するため、文章中で単語がマッチした部分全てを一旦取得し、そこから逆に人物固有の情報を削除する。

例としてアインシュタインを考える。アインシュタインは、<人物名>="Alvert Einstein"、<生年>="1879"、<没年>="1955" である。文章中でこれらの単語が同時に出てきた場合、その記述はアインシュタインの伝記的記述であると考えられる。例えば文章中に

"**Albert Einstein (1879 - 1955)** was a German-born ...."

のように単語が同時に出現したとする。この場合、単語も含めた周囲のフォーマットの候補として考えられるのが

"<人物名> (<生年> - <没年>)"

"<人物名> (<生年> - <没年>) was"

"<人物名> (<生年> - <没年>) was a"

などである。

さて、このような伝記的情報の記述法は人物によってあまり差異が無いと考えられる。そのため複数の人物の伝記的情報において、テンプレート候補を抽出し、特に出現頻度の高いテンプレート候補を、人物情報を抽出するためのテンプレートとした。この例では、一番上のテンプレート候補の出現頻度が最も高いため、このテンプレート候補をテンプレートとした。そして、このようなテンプレートを生年月日以外に、国籍、所属組織、職業などについて作成した。

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、これを人物を特徴付ける情報として用いることで、ウェブページ中に出現する人物ごとにページをクラスタリングする手法を提案した。

彼らは、ウェブページの特徴をベクトルとして表現するために、ベクトルの要素として、生年月日、国籍、配偶者などのそれぞれの情報の有無を用いた。二つのウェブページ間でこれらの数値の一部、もしくは全部が一致すれば二つのページの人物には相応の関係があると言える。そしてこれらのベクトルで構

成されたベクトル空間を用いて、同姓同名人物を持つ人物名を含むウェブページ集合のクラスタリングを行ない、ウェブページを人物ごとに分類することに成功した。

2.2 人物名と共起する情報を利用した手法

次に、生年月日など人物固有の情報だけでなく、文章中で人物名と同時に現れる普通の単語にも注目して、同姓同名な人物の判別を行なう手法がある。ここでは、共起する単語を利用した例として Fleischman[6] らの研究を、共起する人物に注目した例として佐藤 [7] らの研究を挙げる。

判別ための情報としてしばしば使用されるのが、同姓同名な人物名と同時に出現する他の人物名や組織である。Fleischman[6] らは分類済みな学習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計算する事で、文中に出現する同姓同名な人物のクラスタリングをする手法を提案した。

彼らは文章中に人物名が出現する際、人物名の近傍に人物の所属する組織や役職などの単語が出現すし、それらの組は同じ人物ならば同じであると考えた。例えば、元首相の小泉純一郎氏の名前の一部である「小泉」について考える。この名前はニュース記事中では、「首相」という単語と同時に出現する可能性が多い。そのため、未知の文章中に「小泉」と「首相」という単語が同時に出現した場合、この文章は元首相の小泉純一郎氏に関連する話題を扱った文章であると推定できる。そこで彼らは、このような役職、組織名と名前のペアの出現率を新聞記事のコーパスから抽出し、各ペアの出現回数を求めた。

さて、ありふれた名前のほうがコーパス中における出現回数が多いのは自明である。例えば「佐藤」という名前を持つ人は多数居るため、どの職業にも存在すると考えられるため、色々な単語と一緒に出現する事になる。そのため、本来相応しくないペアの出現回数も増加させることになる。そこで彼らは名前の特有性を加味することで、ありふれた名前に左右されないペアの出現頻度を作成する事にした。これは単語と名前のペアの出現回数が少ない場合でも、特有な名前であれば出現回数を高く計算する方法である。名前の特有性はウェブ検索にて名前をクエリとして与えた場合の結果の数と、ACL データセットにおける出現数が用いられている。これにより人物名と単語のペアの出現頻度が与えられる。表 2.2 に人物名と単語、それに出現頻度のペアの例を示す。佐藤が複数含まれるのは、これらが文章中に出てきた同姓同名人物の疑いのある人物名ということである。

同姓同名な人物が存在する名前について、複数のペアが存在する場合、同じ名前を持つペアでも、この名前が違う人物を指す場合がある。そこでこのペアを辺、名前と単語を頂点としたグラフを作成した。各辺の重みは、先に求めた出現頻度を与える。このグラフのクラスタリングを行なう事で各人物の分離を

名前	単語	ペアの出現頻度
佐藤 1	学生	1.0
佐藤 1	会社員	0.6
佐藤 2	学生	0.7
佐藤 2	会社員	0.8
佐藤 3	会社員	0.3
佐藤 3	自営業	0.9
吉田	自営業	0.5
田中	自営業	0.2
⋮	⋮	⋮

図 2.2: 名前と単語のペアの例

行なう。表 2.2 の佐藤に関して作成したグラフと、そのクラスタリング後のグラフを図 2.3 に示す。図 2.3 左図が表を元に作成したグラフであり、右図が辺重みが 0.3 より上の辺で繋がれている頂点をクラスタリングした結果である。この例では、最終的に佐藤は 1 、 2 が同一人物であり、 3 が別人であるという

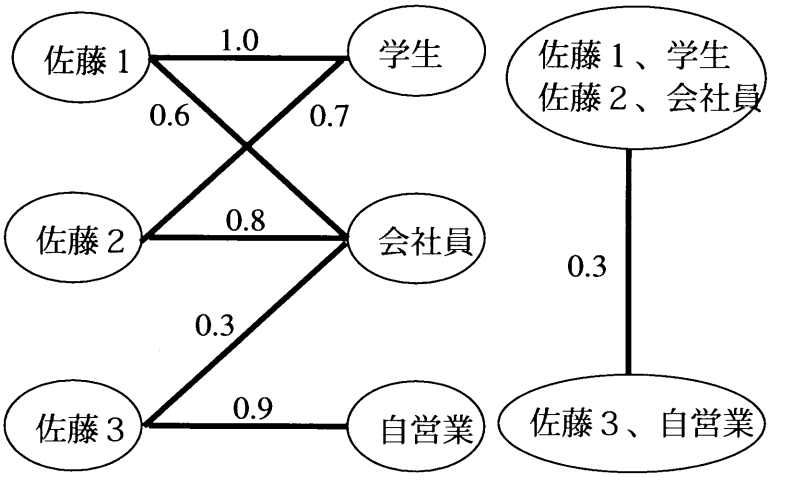


図 2.3: グラフ例

結果になっている。

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た。そこで同一のウェブページや同ドメインのページで、ある人物名と同時に出現する他の人物は、その人物と関係があると考えた。そして、この関係をグラフにした上でクラスタリングを行なう事で、同姓同名な人物名を含むウェブページのクラスタリングを行ない、各

人物毎に分離する事に成功した。

佐藤らは実世界での人間関係においてクラスタが生じていると考えた。例えば、同じ大学に通っている人物の間には互いに人間関係があり、この関係を人物を頂点、関係を辺としたグラフに起こすと、グラフは密である。一方、大学の外部との関係は、各人物が個々に関係を持っているが内部と比較すると疎である。ここで、大学の各研究室のウェブページにこれらの人物の名前が掲載されている場合、各ページのURLは違えど、大学のドメイン単位で見れば、ドメインには実際の人物関係が反映されていると言える。そこで佐藤らは同姓同名な人物名を含むウェブページ、もしくはそのページと同一なドメイン中に共起する他の人物は、その人物名と関係があるとした(図2.4)。

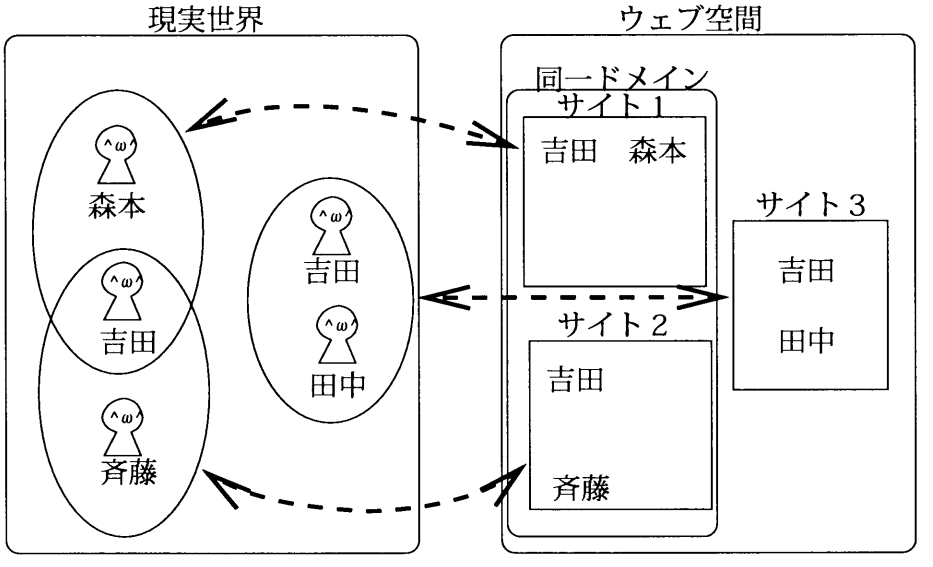


図 2.4: 佐藤らの考えた実世界とウェブ空間の対応関係

そして、人物名を頂点、共起関係を辺としたグラフを作成し、これをクラスタリングすることにより同姓同名な人物の分離を行なった。

2.3　リンク、URL間の距離を利用した手法

最後にウェブページ特有の情報として、ページ同士の類似度をURL、リンクの有無などを利用して同姓同名人物する手法がある。

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ページ間に存在するリンクパスの有無で定義した。そしてリンクパスの有無をベクトルの要素のひとつとして与え、階層的クラスタリングにより、同姓同名な人物を扱うウェブページの分類を行なう手法を提案した。

リンクパスとは、直接、または間接的にリンクでページ間が繋がっている場合に、その経路の事を指す (図 2.5)。一般的にウェブページにリンクが張られ

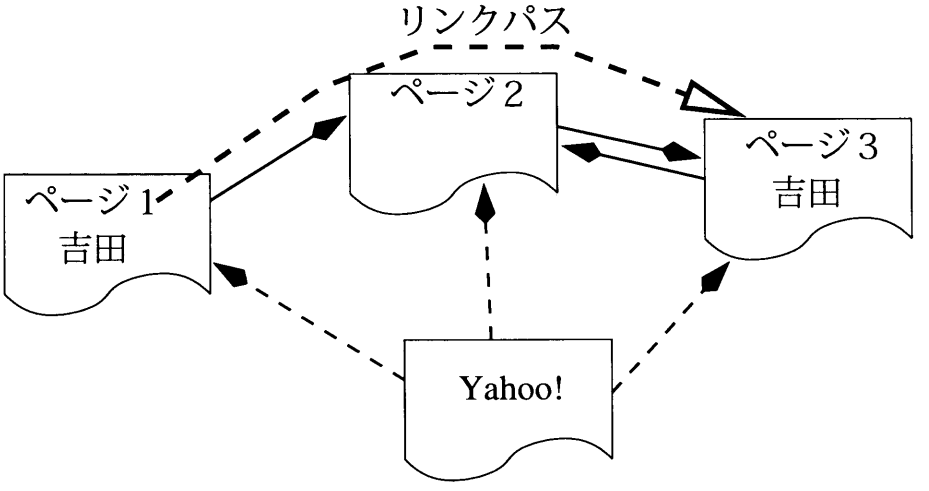


図 2.5: リンクパスの例と、その例外

る場合、リンク先のページの内容が元のページの扱う話題と関係しているためと考えられる。この関係がが再帰的に起きると考えられるため、リンクパスで繋がっているページの話題には類似性があると考えられる。そこで Bekkerman らは、ページが同一人物の話題を扱っている場合にはその間にリンクパスができると考えた。ただし、リンクパスを構成する要素の例外として、Bekkerman らは有名サイトを挙げている。有名サイトとは、検索エンジンや、リンク集などのページは自身とは関係ないページへ大量にリンクを張っているサイトの事である。このようなサイトのうち悪意のある物はリンクスパムと呼ばれる。また善意の物でも、極端に有名なサイトには無関係なページからリンクを張られる事が多い。例えば、ある会社の無料ブログサービスにおいて、各ユーザーのページにはユーザーの意志とは無関係にその会社のトップページへのリンクが張られている。Bekkerman らはウェブページの有名度を計る指標として、ウェブページ URL を検索サイトで検索した際の出力数を考え、この値が一定以下のページのみをリンクパスを考える上での対象とした。

同様の研究として、Al-Kamha[5] らのものがある。Al-Kamha[5] らは URL 類似度とページに含まれる連語の種類で特徴量を定義し、ページを同一人物でグループ化する手法を提案した。

2.4 その他の関連研究

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。表示システムは、ユーザーによる分類精度の評価を行

ないやすくする狙いがある。この評価をフィードバックすることで、システムは分類精度を向上する事が出来る。

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、人物毎にページをまとめてユーザーに提示するシステムを作成した。

第3章　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。

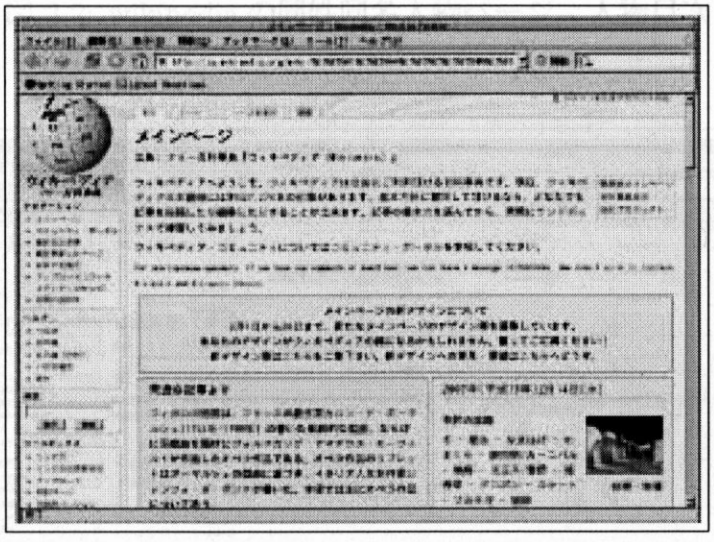


図 3.1: wikipedia のページ例

wikipedia には人物名の項目も存在する。人物名の項目には略歴や職業などの、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。Wikipedia 中で同姓同名な人物を検索した場合、まず図 3.2 のような同姓同名な人物をリストにしたページが出力される。これは、Wikipedia 中における項目名の独自性を保証するためである。Wikipedia 中ではほとんどの場合、名前の後ろに職業名をつける事で同姓同名な人物を区別している。そして、このページから各人物の項目に移動する事になる（図 3.3）。

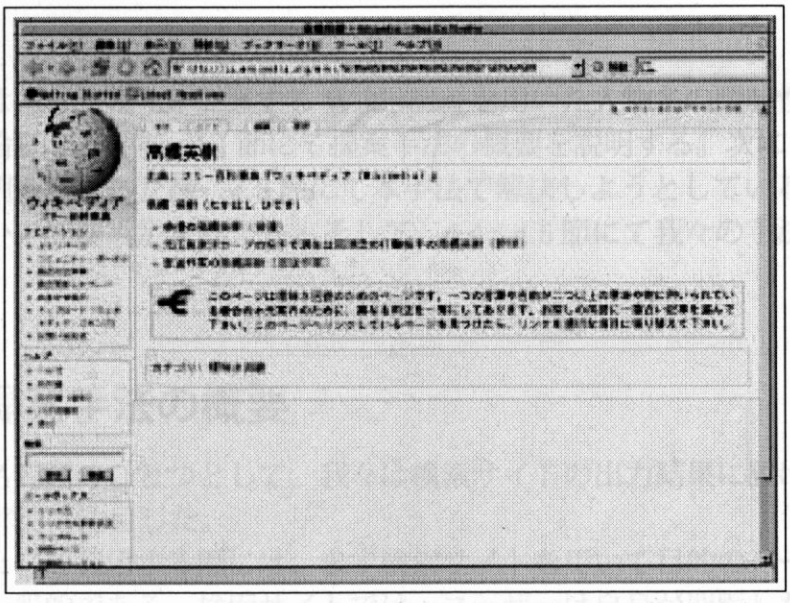


図 3.2: wikipedia の同姓同名人物ページ - 人物目次

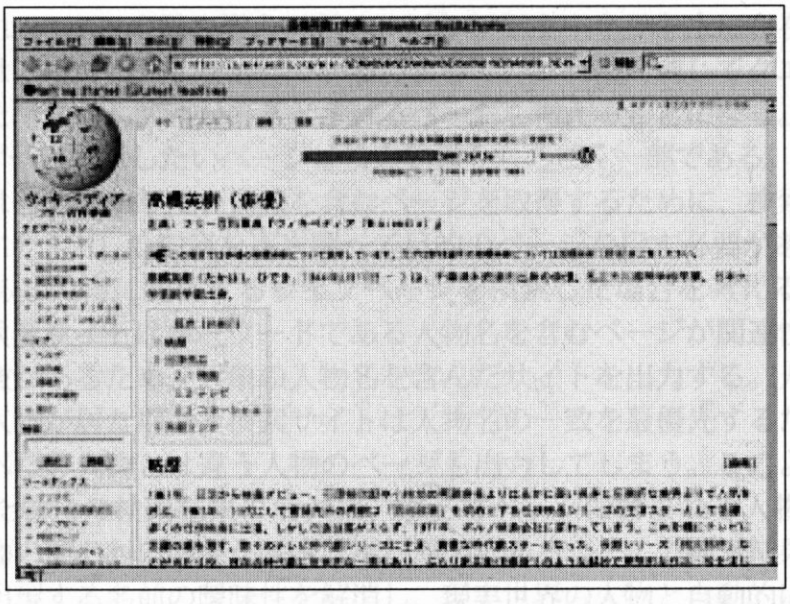


図 3.3: wikipedia の同姓同名人物ページ - 各人物ページ

第4章　提案手法

この章では、今回我々が提案するWikipediaを用いた人物名の曖昧性解消手法について論じる。まず、4.1節にて提案手法の概要を説明する。次に4.2節にて用語の説明を行なった後、4.3節にて本手法で解決しようとしている同姓同名問題について問題設定を与える。そして、4.4、4.5節にて我々の手法について説明する。

4.1　提案手法の概要

同姓同名問題のひとつとして、我々は検索サイトの出力結果に置ける人物名の曖昧性解消に注目した。

現在ウェブを利用する際には、まず検索サイトを用いて目的のページを捜し出すのが一般的である。検索サイトでは、ユーザーは自身が閲覧したいと思う情報に関連する単語を検索サイトにクエリとして送信する。検索サイトは独自のアルゴリズムを用いて、クエリとして与えられた単語に関係すると思われるページのURLをユーザーに提示する。ここで問題となってくるのが、ユーザーに提示されるURLの量の増大である。近年インターネット上に存在するウェブページは、増大の一途を辿っている。よって、これに同調してクエリとして入力された単語と、関連があるとされるページも増えることになる。大抵の場合、ユーザーの閲覧したいページは検索結果うちのごく一部である。そのため、ユーザーは自分の知りたい情報を含むページを取得するために、検索結果として与えられたURLの内容を逐次調べて目的のページを探す必要がある。

ここである人物に関係するウェブページを検索した場合を考える(図4.1)。この時、検索サイトはキーワードである人物名を含むページが関連性が高いと考える特徴があるため、対象の人物名を含んだサイトを出力する。人物名に同姓同名な人物が居た場合、検索サイトは人物名の一致を最優先するため、ユーザーが調べたい人物とは違う人物のページも出力してしまう。このような人物を検索した場合の検索結果の増大は、ウェブページに含まれる人物名と現実世界の人物の間の対応関係が曖昧な事が原因である。ここで検索結果のウェブページに出現する名前の曖昧性を解消し、現実世界の人物と自動的に対応付ける事ができれば、ユーザーの手間を省く事ができる。このように現在ウェブ空間における人物名の曖昧性を解消する事が求められている。

図 4.1: 同姓同名問題が生じている検索結果の例

そこで我々はwikipediaという辞書を利用して、ウェブページに出現する人物名の曖昧性解消を行ない、検索結果における人物名の曖昧性による問題の緩和を行なおうと考えた。以下に問題解決の流れを示す。ある人物名に対して、

1. ある人物名がWikipediaに記載された同姓同名の人物であるかどうか判別する
   (非項目保持者判定)

2. Wikipediaに記載ががある人物を対象に、Wikipedia中のどの人物であるか判別する
   (項目保持者判定)

これにより検索結果をWikipediaに項目がある各人物と、それ以外の人物に分類する事ができる。ユーザーがWikipediaに項目がある人物のウェブページを発見したい場合は、そのまま分類されたページを閲覧すれば良く、それ以外の人物のウェブページを知りたい場合にも、Wikipediaに項目がある人物のサイトが削除されて個数が少なくなった検索結果からユーザーが捜し出せばよい。

Wikipediaという辞書が持つ情報を利用する手法は次のようになっている。ウェブページに含まれる人物名と、ページ中の人物名以外の部分の文章は関連があると考えられる。我々は、特に文章中に出現する単語と、その出現数が人物名と深い関係にあると考えた。例えば、映画のレビューをするページ中で俳優の名前が出てくる場合を考える。俳優が出演する映画のため、映画の名前や配給会社の名前が俳優に関連する単語であると言える。一方、Wikipediaに含まれる各人物の項目ページも一種の文章である。そのため、項目ページに含まれる単語も項目の人物と関係があると考えられる。そして、Wikipediaの項目は現実の人物と1対1対応である事が保証されている。そこでWikipediaの項目ページに出現する単語と、ウェブページの文章中に出現する単語分布を比較する事で、ウェブページがWikipediaの項目ページに最も近いか判別する。ウェブページに含まれる人物名は文章と関係がある事が多いため、結局この作業はウェブページに含まれる人物名がWikipediaの項目ページの人物の誰に最も近いかを判別していることになる。

4.2 用語説明

以降、主張をより的確に伝えるために使用する用語の説明を行なう (図4.2)。

- 「同姓同名人物」
  同じ名前を持つ人物の集合。さらに以下のように二つに分けられる。
  - 「項目保持人物」
    Wikipedia中に自身を説明するページを持つ人物

– 「非項目保持人物」
Wikipedia 中に自身を説明するページを持たない人物

- 「人物情報ページ」
ウェブ空間中に存在するページのうち、人物を扱っているページである。これらは必ず現実世界の人物のうちの誰かを指す。一人の人物に対して複数のページがある場合がある。

- 「項目ページ」
Wikipedia 中に含まれる辞書の各項目に該当するページである。各人物に対応するページはこの中に含まれる。

  – 「人物説明ページ」
  Wikipedia 中に存在するページのうち、人物を扱っているページである。これらも必ず現実世界の人物のうちの誰かを指す。人物説明ページの場合は、一人の人物に対して一つのページしかない。

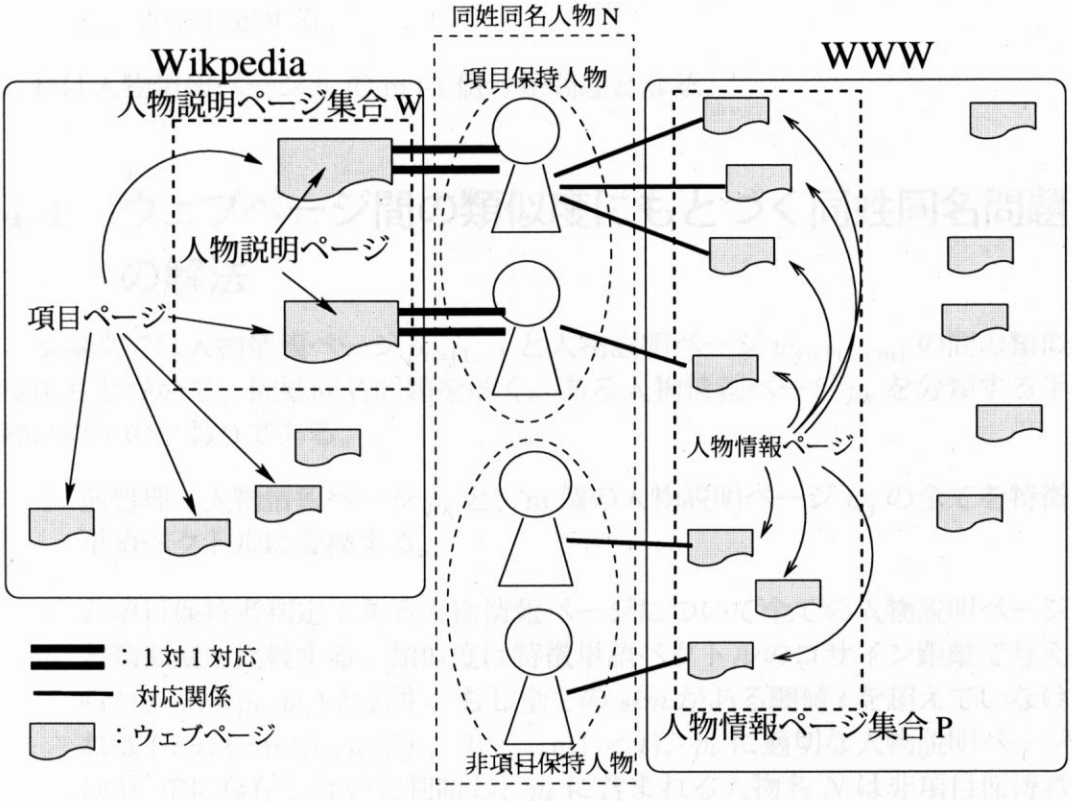


図 4.2: 用語の説明

4.3 問題定義

我々が行なおうとしている作業は、ウェブ空間に存在するページを各人物に対応付ける事である。そこで、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。

結局、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。

4.4 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. 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。

3. 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。

図 4.3 にそれぞれの判定の具体例を示した。この図では、Wikipedia 中の人物項目ページはそれぞれ、A、B 二人の人物、分類対象となる人物情報ページ X、Y はそれぞれ A、C 二人の人物を指すページである。人物 C は人物項目ページを持たないので、非項目保持者である。また、図の矢印の横に示されている数値は、各ページ間の類似度である。

まず、項目保持者判定について例を挙げて説明する。項目保持者判定では、分類対象の人物情報ページと全ての人物項目ページの間の類似度を比較する。そして、類似度が最も高い人物項目ページの人物を人物情報ページが指していると判定する。この例では、ページ X の各人物項目ページに対する類似度は 0.8、0.3 である。よって、類似度の高い A の人物情報ページと同一人物のページ、つまり A のページであると言える。

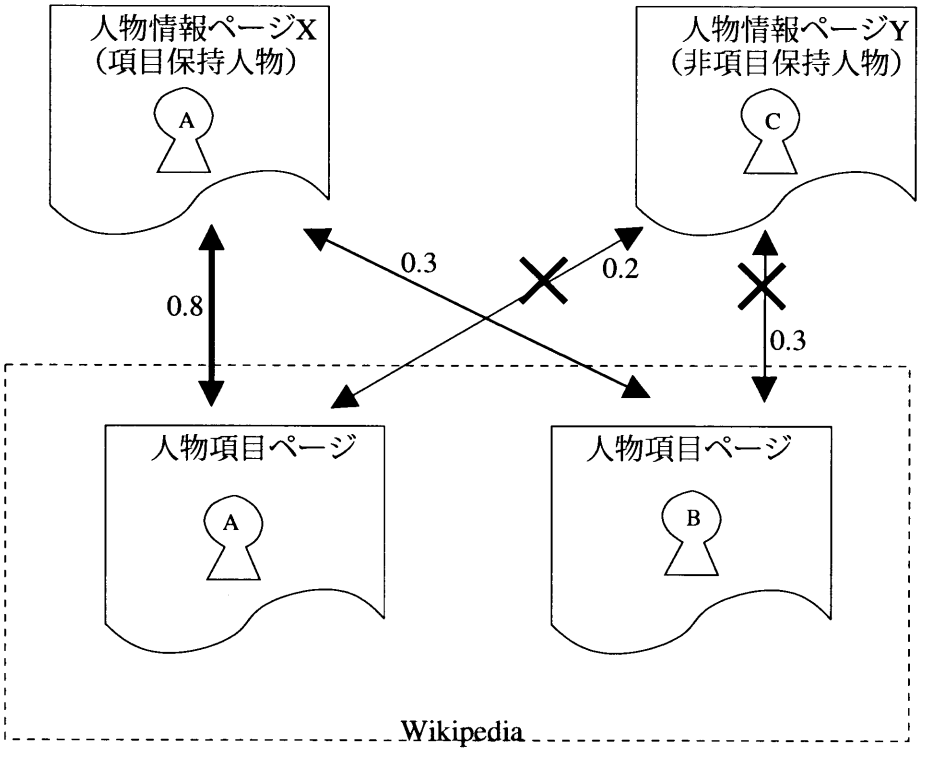


図 4.3: 項目保持者判定、非項目保持者判定の説明

4.5 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名詞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 4.1 の個数出たとする。この場合、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

表 4.1: wikipedia 中における名詞の出現回数の例

	w_A	w_B
n_1	1	4
n_2	2	0
n_3	6	8

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。

第5章　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5.2に示す。表5.2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参考にしていただきたい。

表 5.1: 実験対象となるデータセット (1)

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	

表 5.2: 実験対象となるデータセット (2)

名前	識別情報	番号	ページ数	特徴
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 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。

表 5.3: 人物名のベクトルにおける特徴的な名詞の例 (1)

名前 (ベクトル次元)	番号	特徴的なベクトル成分
江川卓 (703)	0	社、潮、文学、 ロシア、文庫、教授
	1	江川、野球、こと 投手、巨人、阪神
岩崎良美 (353)	0	年、賞、音楽 新人、祭、タッチ
	1	アナウンサー、福井
増田俊朗 (227)	0	音楽、番組、作曲
	1	木曜、ラジオ、水曜
小川知子 (314)	0	アナウンサー、ニュース
	1	歌手、女優、映画 テレビ、デビュー
佐々木正洋 (191)	0	朝日放送、青森、八
	1	アナウンサー、朝日
鈴木俊一 (397)	0	知事、東京、推薦
	1	善幸、堤、岩手
石川賢 (694)	0	作品、石川、原作 永井、魔
	1	山梨、大洋、球団
	2	中日、野球、選手
中田浩二 (474)	0	アニメ、江戸、探偵
	1	出場、選手、鹿島 日本、移籍、監督
小野大輔 (383)	0	選手、移籍、サッカー
	1	編集、声優、出演
白井貴子 (366)	0	出演、音楽、本人
	1	バレーボール、リーグ

表 5.4: 人物名のベクトルにおける特徴的な名詞の例 (2)

名前 (ベクトル次元)	番号	特徴的なベクトル成分
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

5.2 実験 1 項目保持者の特定

実験 1 では 4.4 節における「手順 (3)：項目保持者分類」の評価を行なう。実際の手順 (3) では、(2) より与えられるテストデータに非項目保持者が含まれない。

よって同様の条件で実験を行なうために、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 (3) 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。

表 5.8 に実験 1 の結果を示す。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 P、R、F は当該列の人物に対する Precision、Recall、F-measure を示している。

ただし、各値のそれぞれの計算式は以下のようになっている。ある人物名 N を持つ二人の人物 A、B が居て、それぞれの人物のウェブページがある場合、人物 A に対する Precision、Recall、F-measure は

a = 真の A のページのうち、計算機が A のページであるとして出力したウェブページ数

b = 真の A のページのうち、計算機が B のページであるとして出力したウェブページ数

d = 真の B のページのうち、計算機が A のページであるとして出力したウェブページ数

としたときに、以下のように表される。

$$\text{Precision} = \frac{a}{a+d}$$

$$\text{Recall} = \frac{a}{a+b}$$

$$\text{F-measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

表 5.5: 実験 1 の結果 (名詞)

名前	番号	(a)	(b)	P	R	F
江川卓	0	103	53	0.84	0.92	0.88
	1		50	0.91	0.82	0.86
岩崎良美	0	73	49	0.81	0.36	0.51
	1		24	0.39	0.83	0.53
増田俊朗	0	54	31	0.82	0.45	0.58
	1		23	0.54	0.87	0.67
小川知子	0	100	50	0.59	0.84	0.69
	1		50	0.72	0.42	0.53
佐々木正洋	0	63	11	0.36	0.73	0.48
	1		52	0.93	0.74	0.82
鈴木俊一	0	89	38	0.75	0.87	0.80
	1		51	0.89	0.78	0.83
石川賢	0	60	36	0.82	0.64	0.72
	1		2	0.06	0.5	0.1
	2		22	0.56	0.82	0.67
中田浩二	0	82	33	0.89	0.97	0.93
	1		49	0.98	0.92	0.95
小野大輔	0	105	50	0.79	1	0.88
	1		55	1	0.76	0.87
白井貴子	0	99	51	0.91	0.96	0.93
	1		48	0.96	0.90	0.93
高橋英樹	0	52	48	0.94	0.60	0.73
	1		4	0.10	0.5	0.16
吉田恵	0	73	47	0.78	0.96	0.86
	1		26	0.87	0.50	0.63
伊藤博文	0	99	52	0.89	0.81	0.85
	1		47	0.81	0.89	0.85
前田愛	0	140	40	0.38	0.90	0.53
	1		47	0.38	0.55	0.45
	2		53	0.38	0.28	0.32
坂本太郎	0	106	58	0.86	0.98	0.91
	1		48	0.98	0.81	0.89
鈴木健	0	63	8	0.21	0.88	0.34
	1		21	0.42	0.71	0.53
	2		34	0.67	0.44	0.51
田村亮	0	100	52	0.82	0.44	0.58
	1		48	0.60	0.90	0.72

表 5.6: 実験 1 の結果 (名詞+未定義語)

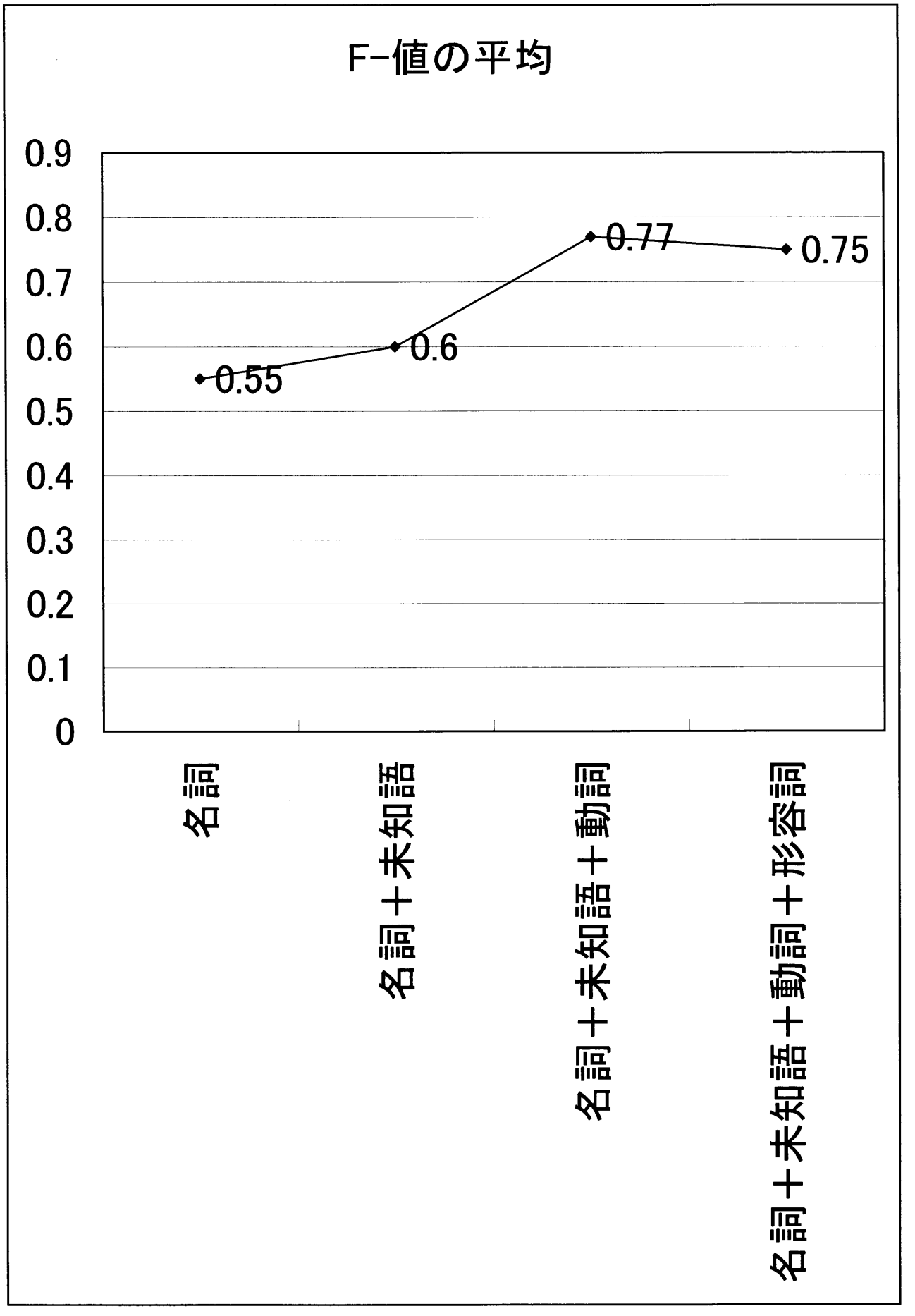
名前	番号	(a)	(b)	P	R	F
江川卓	0	103	53	0.82	0.91	0.86
	1		50	0.88	0.80	0.84
岩崎良美	0	73	49	0.89	0.34	0.50
	1		24	0.41	0.92	0.56
増田俊朗	0	54	31	0.75	0.39	0.51
	1		23	0.50	0.83	0.62
小川知子	0	100	50	0.56	0.74	0.64
	1		50	0.63	0.44	0.52
佐々木正洋	0	63	11	0.37	0.64	0.47
	1		52	0.91	0.77	0.84
鈴木俊一	0	89	38	0.72	0.87	0.79
	1		51	0.88	0.75	0.81
石川賢	0	60	36	0.80	0.56	0.66
	1		2	0.05	0.50	0.09
	2		22	0.51	0.82	0.63
中田浩二	0	82	33	0.88	0.85	0.86
	1		49	0.90	0.92	0.91
小野大輔	0	105	50	0.74	1	0.85
	1		55	1	0.67	0.80
白井貴子	0	99	51	0.87	0.94	0.91
	1		48	0.93	0.86	0.89
高橋英樹	0	52	48	0.93	0.58	0.72
	1		4	0.09	0.5	0.15
吉田恵	0	73	47	0.80	0.96	0.87
	1		26	0.88	0.58	0.70
伊藤博文	0	99	52	0.89	0.77	0.85
	1		47	0.77	0.89	0.83
前田愛	0	140	40	0.38	0.90	0.54
	1		47	0.37	0.45	0.40
	2		53	0.41	0.40	0.40
坂本太郎	0	106	58	0.86	0.98	0.91
	1		48	0.98	0.81	0.89
鈴木健	0	63	8	0.18	0.63	0.52
	1		21	0.42	0.67	0.52
	2		34	0.64	0.53	0.58
田村亮	0	100	52	0.84	0.40	0.55
	1		48	0.59	0.92	0.72

表 5.7: 実験 1 の結果 (名詞+未定義語+動詞)

名前	番号	(a)	(b)	P	R	F
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

表 5.8: 実験 1 の結果 (名詞+未定義語+動詞+形容詞)

名前	番号	(a)	(b)	P	R	F
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	0.70	0.46	0.55
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	0.83	1	0.91
	1		55	1	0.82	0.90
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	0.87
	1		26	0.88	0.54	0.67
伊藤博文	0	99	52	0.82	0.90	0.86
	1		47	0.88	0.79	0.83
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	0.42	0.57
	1		48	0.60	0.94	0.73


F−値の平均
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.55
0.6
0.77
0.75
名詞
名詞＋未知語
名詞＋未知語＋動詞
名詞＋未知語＋動詞＋形容詞

5.3 実験2 非項目保持者の判別

この節では4.4節における手順「(2)：非項目保持者判定」の実験と評価をする。手順(2)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5.1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$(\text{Precesion}, \text{Recall}) = (0.5, 0.154)$である。$\theta$をX軸、F-measureをY軸にとったグラフを図5.2に示す。

ただし、各値のそれぞれの計算式は以下のようになっている。分類対象のウェブページが「その他の人物」かそうでないかを判定するとして、

$$\text{Precision} = \frac{\begin{array}{c}\text{計算機が「その他の人物」と出力したページ中で} \\ \text{真にその他の人物のページであるものの数}\end{array}}{\text{計算機が「その他の人物」としたページ数}}$$

$$\text{Recall} = \frac{\begin{array}{c}\text{計算機が「その他の人物」と出力したページ中で} \\ \text{真にその他の人物のページであるものの数}\end{array}}{\text{真にその他の人物のページであるものの数}}$$

$$\text{F-measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

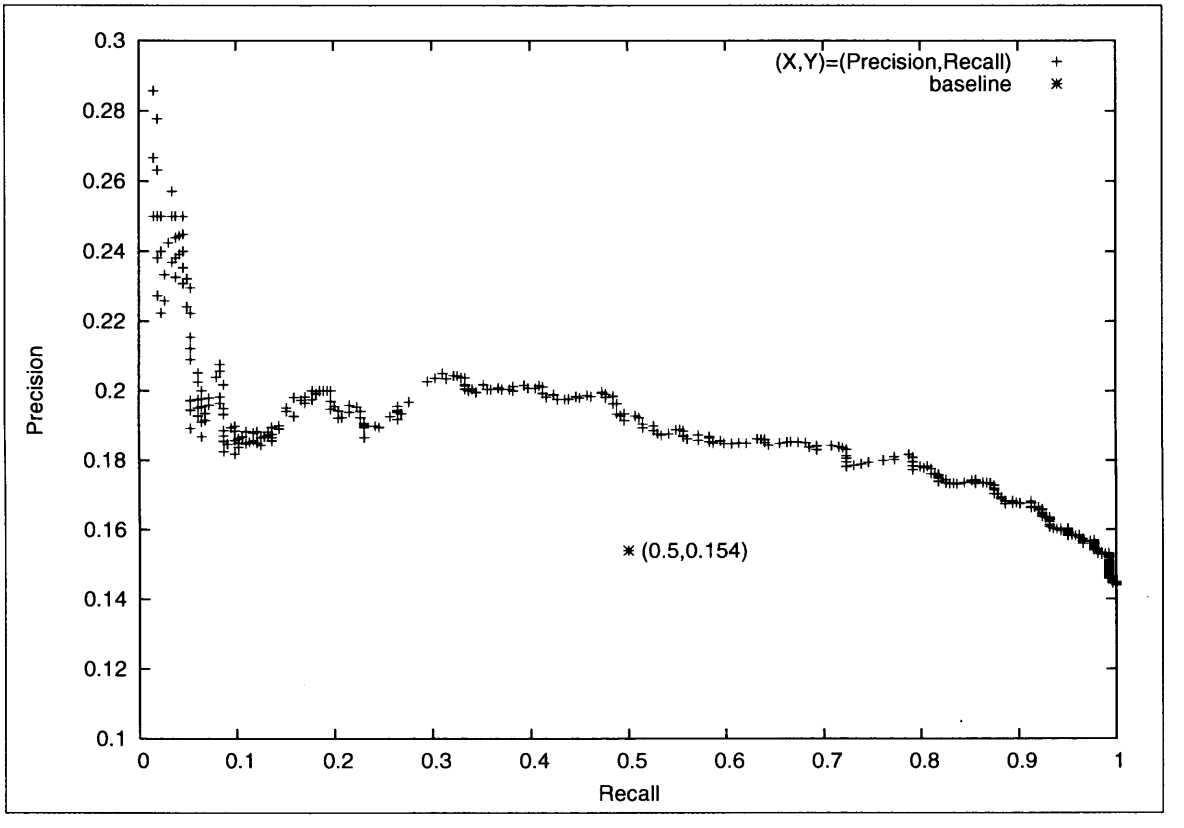


図 5.1: 実験２結果: Precesion,Recall の関係

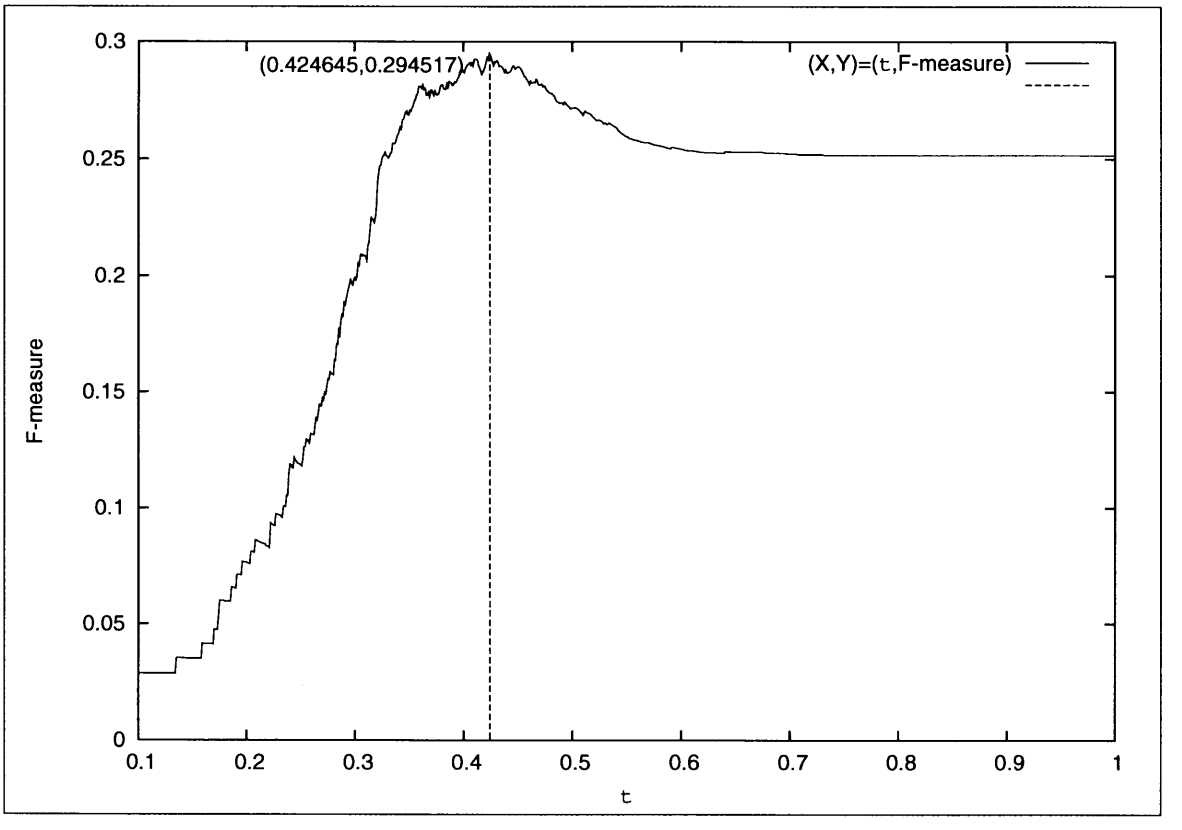


図 5.2: 実験 2 結果: 閾値 t と F-measure の関係

第6章　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。

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6.1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。

6.2 実験2に対する考察

実験結果の図5.1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。4.5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6.2) 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6.2には二つのグループに分けられる。まず上から

表 6.1: 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	

表 6.2: 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

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6.1)。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6.1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6.2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。

「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。

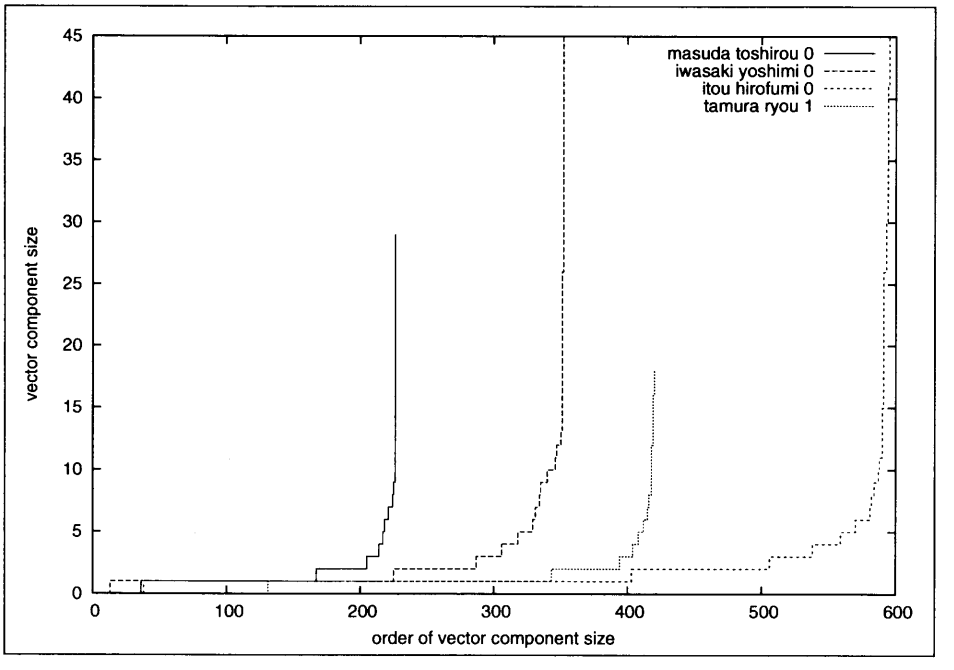


図 6.1: 特徴単語ベクトルの成分のサイズの分布の例

第7章　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、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。実験2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、テストー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。

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。

今回の実験で顕著だったのは、分類対象ページに特徴量として指定した単語が含まれていない場合が多いと言う事だ。例えば、サッカー選手の話題である、サッカーチームの鹿島アントラーズに関するページで、サッカーと言う単語をあまり使用しないでアントラーズに関する話題を論じているページが存在した。このようなページの場合、ベクトル中でサッカー選手を判別するための重要な要素の一つであるサッカーと言う単語の出現数が低くなってしまうため、分類が困難である。ところが人間の場合、「鹿島アントラーズ＝サッカーチーム」という知識を有しており、このような連想がすぐに働く。これにより、人間がこ

のページを見るとすぐさまサッカー選手の話題であると言う事が分かる。そこで、今後の研究ではこのような単語の連想関係の知識を採り入れる事を考えている。連想に関する研究は多々あるため、今後はこれらの研究についても調査していきたい。

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。

謝辞

本研究を進める上でお世話になりました多くの方々に、この場を借りてお礼申し上げます。

まず、本研究室の指導教員である喜連川優教授には、研究テーマや研究への取り組みに関して多くの助言を頂き、大変感謝しております。

また、秘書の中野恵理さん、小笠原薫さん、井崎葉子さん、松島恵里さんは、事務の面で研究室を支えてくださり、快適な研究を行うことができました。ありがとうございます。

鍛治 伸裕氏は、自然言語処理分野での研究を活かし、経験の少ない私に知識やアドバイスを与えてくださいました。本当に感謝しております。

更に、研究室の先輩方、研究員の方々、同期の修論生の皆様にも、日常的な相談から技術的な指導まで、幅広く私の研究生活を支えて頂きました。

本論文を書き終えることができたのも、ひとえに皆様のおかげです。どうもありがとうございました。

参考文献

[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.