Web 検索を利用した人物関係抽出とその評価

鈴鹿工業高等専門学校 専攻科 電子機械工学専攻 三谷 亮介
鈴鹿工業高等専門学校 電子情報工学科 田添 丈博
愛知工業大学 情報科学部 情報科学科 椎野 努

1. 背景・目的

世界におけるインターネット利用者数は日々増加し続けている。日本でも、その利用者数は 9,091 万人(2008 年末)、総人口における利用率は 75.3%と、非常に多くの人々がインターネットを利用していることがわかる[1]。また、利用者数の増加に伴って、facebook や mixi といったソーシャルネットワーキングサービスや、個人のブログが流行し、その中では実際の友人だけでなく、同じ趣味・嗜好を持つ様々な人物と交流を深めている。

近年 SocialGraph という、ブログやソーシャルネットワーキングサービスの情報を用いて、ネットユーザの交友関係をあらわすグラフが話題となっている。その中でも有名な Social Graph API(Google)は、Web 上にあらかじめ埋め込まれた XFN（XHTML Friends Network）や FOAF（Friend Of A Friend）の情報からネットユーザの交友関係を取得し、その関係をグラフとして表現することができる[2]。SocialGraph から得られる交友関係の情報は、ビジネスへの導入や新たなサービスの展開など大きな可能性をみせている。

本研究では、人物関係を抽出する際に Web 検索を利用する。そのため、あらかじめ XFN や FOAF が埋め込まれていないページをも検索の対象にすることができる。検索にヒットした Web 上の文章から形態素解析により人名を抽出し、Jaccard 係数を用いた関連度計算を用いて、人物関係を抽出できることがわかった。また、抽出された人物関係を explicit(直接的)な人物関係と呼び、それにレコメンドエンジンの考え方を取り入れることにより、implicit(暗喩的)な人物関係を導出する手法について研究を行っている。

本論文の構成を示す。1 章では、本研究の背景および目的を示す。2 章では、実際に Web 上の文章から人物関係を抽出するアルゴリズムを示す。3 章では、人物関係の取得にあたって問題となる人名抽出の方法を示す。4 章では、実験を行い人物関係の取得を行った結果とその評価について示す。5 章では、Jaccard 係数による関連度計算では取得できなかった implicit な人物関係を取得する方法と実装する際のアルゴリズムを示す。6 章では、研究のまとめと今後の計画を示す。

2. 人物関係を抽出するアルゴリズム

図 1 のアルゴリズムを用いることで、Web 検索による人物関係の抽出を行うことができる。

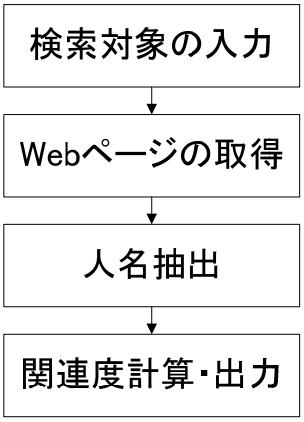

図 1 人物関係抽出アルゴリズム

まず、ユーザが入力した人物名を用いて Web 検索を行う。ヒットした Web ページを取得し、その文章から形態素解析ソフト用いて人名を抽出する。

次に、再び検索エンジンを用いて、抽出した人名とユーザが入力した人物名の AND 検索と OR 検索を行う。その検索件数より、式 1 の Jaccard 係数による関連度を算出する。

$$\text{Jaccard 係数} = \frac{|X \cap Y|}{|X \cup Y|} \qquad \text{(式 1)}$$

この関連度が大きいほど、二者の間に強い関係性があると推測することができる。

3. 人名抽出の方法

3.1. 問題点

本システムにおける一番の課題は人名抽出である。形態素解析には形態素解析ソフト ChaSen[3]を用いた。

ChaSen による形態素解析は「名詞-固有名詞-人名-姓」「名詞-固有名詞-人名-名」といった細かい形で解析することができる。抽出した文章が正しく解析されるなら、「姓」「名」と続いたものを人名として抽出すればよい。だが現在の ChaSen による解析の精度は完全ではなく、解析がうまくいかない場合がある。その理由としては、ChaSen の辞書データに含まれる人名が少ない（姓 1.2 万件、名 1.8 万件）点と、型にはまらないウェブ上の文章は形態素解析自体が難しいことが考えられる。これらの対策として、ChaSen の辞書データの増強と、より確実に人名を抽出するための法則を検討する。

3.2. 辞書データの増強

辞書については、ChaSen に関しては、未知語を前後関係などから推測する機能はないため、登録済みの別の語に分割されたり、「未知語」として扱われたりする。このことから辞書データの増強は不可欠であるが、増強（特に名）に関して２つ問題点がある。

一つ目は、適切な形態素生起コストの設定が困難であることである。ChaSen の辞書データでは、単語ごとに形態素生起コストというものを指定する。このコストの設定が適切でない場合、他の単語と競合してしまい適切な解析が行えなくなる。たとえば「山口」という語は、「名詞-固有名詞-地域-一般」と「名詞-固有名詞-人名-姓」両方の形態素で登録されている。コストの設定が悪ければ、文によっては姓の「山口」まで地名の「山口」として解析されたりする。しかし膨大な人名データに対し個々に適切なコストを指定することは困難である。

二つ目は、辞書への登録数の多さである。その数は、姓に関して言えば、ある程度の範囲に絞られる。読みや表記の違いを全て考慮すると日本には 30 万件の姓がある。読みの違いを考慮しなければ 10 数万件と言われている[4]。

だが、名は姓と比べると、自由度が高く、数が非常に多い。100 万件の名をもつ命名辞典があるが、さらに毎年新しい名が登場している。これら全てを ChaSen の辞書に登録して対応するのは、非常に困難である。

上記を踏まえて、本研究では、Web サイト

「同姓同名辞典」
(http://www.douseidoumei.net/)
「平成名前辞典」
(http://www.namaejiten.com/)

これら二つのサイトから姓 13 万件と重複を除いた名 60 万件のデータを利用して ChaSen の辞書データの増強をおこなった。

辞書増強方法
姓 13 万件・名 60 万件
一律コスト 3000

3.3. 人名抽出の法則

人名抽出の法則に関しては ChaSen により正しく解析された場合の抽出方法を考える。

一般にウェブ上において人名は

①鳩山由紀夫　　([姓][名]の区切りなし)
②鳩山 由紀夫　　([姓]＋半角スペース＋[名])
③鳩山　由紀夫　　([姓]＋全角スペース＋[名])

のように書かれることが多い。

これらの解析結果は

①鳩山[姓]＋由紀夫[名]
②鳩山[姓]＋由紀夫[名]
＊ ChaSen は半角スペースを単語として扱わない。
③鳩山[姓]＋[＿]＋由紀夫[名]

となる。出力結果として①と②は同じである。

よって人名抽出の基本法則として、次の二つを適用することとする。

人名抽出法則
「姓」＋「名」
「姓」＋「＿」＋「名」

4. 実験と考察

4.1. 形態素解析による人名抽出

3 章の辞書データの増強方法と人名抽出法則を用いて実験を行う。Yahoo!の WebAPI を利用して「田添丈博」を検索して、ヒットした上位 3 件の Web ページに関して形態素解析と人名の抽出を行った結果を表 1 に示す。

表 1　人名抽出結果(23 個)

渥美清隆	○	三谷亮介	○	田中亘	○
伊藤八十四	○	出口祐輝	○	樋口健太	○
伊藤明	○	信田龍哉	○	平野武範	○
井瀬潔	○	森育子	○	本郷ダイヤ	×
宮地洋太	○	人工知能	×	箕浦弘人	○
桑原裕史	○	青山俊弘	○	濱崎達也	○
高橋勲	○	斉藤正美	○	眞野裕也	○
砂川未佳	○	長嶋孝好	○		

文章中には、24 個の名前があったが、抽出数は 23 個、そのうち正解数が 21 個、抽出ミスは「人工知能」と「本郷ダイヤ」の 2 個であった。

4.2. Jaccard 係数による関連度計算

次に、Jaccard 係数の有効性を確かめるため、同様に「田添丈博」を検索して得られる文書 47 件から人名抽出を行い、抽出された人名に対し Jaccard 係数を使った関連度計算を行った。

この結果の全 1709 名中、上位 20 名を本人が

○「相互に知っている」
△「一方的に知っている」
×「知らない」

の三段階でそれぞれ検証した(表 2)。その結果、20 名中 18 名が実際に知り合いであった。

この結果から、人物間の関連度計算において Jaccard 係数が有効であることがわかった。また、人名抽出ミスについては、人名でないものは Jaccard 係数による関連度計算の際、ふるい落とされ上位にはあがってこないことがわかった。

表 2　人物関係の抽出結果

	名前	関連度	
1	井瀬潔	0.209	○
2	長嶋孝好	0.185	○
3	箕浦弘人	0.165	○
4	平野武範	0.146	○
5	伊藤八十四	0.133	○
6	青山俊弘	0.114	○
7	椎野努	0.107	○
8	桑原裕史	0.093	○
9	渡辺千亜季	0.090	○
10	河出康宏	0.058	○
11	出口祐輝	0.058	○
12	中林雄介	0.054	○
13	石原茂宏	0.054	○
14	河合啓文	0.052	○
15	眞野裕也	0.049	○
16	渥美清隆	0.048	○
17	井沢味奈子	0.045	×
18	川岸将実	0.045	×
19	中根孝司	0.044	○
20	下古谷博司	0.044	○

しかし、同姓同名の多い人名は Jaccard 係数の値が低くなってしまうため、同姓同名も考慮した関連度計算の方法について研究を進めている。

5. 隠された人物関係の抽出

5.1. explicit と implicit な関係

本研究は人と人のつながりを調べようというものである。一般に人間はなんらかのつながりの中にいる。「田添丈博」であれば「鈴鹿高専」のつながりに所属している。つながりに属する人たちは相互に結びついているため、鈴鹿高専の教員を誰から検索しても他の教員の名が抽出される。

このことから、Jaccard 係数による関連度で上位に来た人たちからさらに検索を行い、その中でよく抽出される名は、Jaccard 係数による関連度が低い場合でも関連性が高いのではないかと推測することができる。

ここで人間関係抽出結果に表れた人物とユーザが入力した人物や、図2中Aとabcd、Bとbcdのように、はっきりと表現されている関係をexplicit(直接的)な関係と呼ぶ(実線)。

レコメンドエンジンにおけるレコメンド手法におけるアイテム同士の関連[5]に着目すると、bcdのようにAとBどちらにも関係があるものが多数ある場合、図中には(実線では)示されていないAとBの間にも何かの関係があるのではないかと推測することができる、この関係をimplicit(暗喩的)な関係と呼ぶ(破線)。

explicitな情報より、レコメンドエンジンの手法を用いることによって implicit な情報を導くことができると思われる。

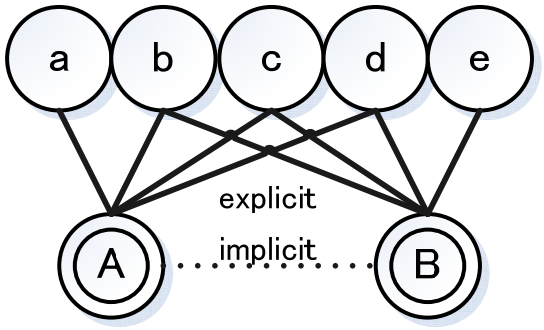

図2 アイテム同士の関連

5.2.アルゴリズムの拡張

Jaccard係数を使った関連度計算より、人物間のexplicit な関係、implicit な関係を判定するアルゴリズムを提案する(図3)。

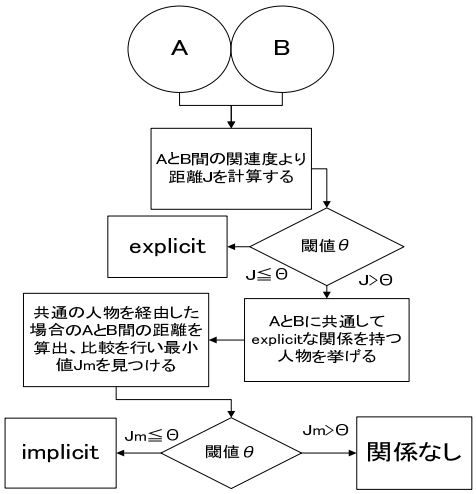

図3 AB間の関係を判定するアルゴリズム

ユーザが指定した人物Aと、人物関連の有無を調べたい人物Bを入力として与える。まず、2者間の関連度を計算し、その値から距離Jを求める。その値が閾値θ以下であるならば、その2者間には explicit な関係が存在することになる。閾値θより大きい場合、AとBの両方にexplicitな関係を持つ人物を挙げる。その人物を経由した場合の A と B の間接的な距離を計算する。共通して explicit な関係を持つ人物の数だけ距離を計算し、その最小距離を見つける。その最小距離を閾値θと比較し、θ以下であれば implicit な関係が存在することになる。また、最小距離を用いても、θより大きい場合、二者の間には関係が存在しないということがわかる。

6.まとめ

本研究では、Web検索を用いて人物関係を抽出する研究を行った。XFNやFOAFを使わずに、Web検索を利用するため、あらかじめ情報を埋め込んでいないページからも人物関係を取得することができる。手順は、まずユーザが指定した名前を Web 検索で検索し、ヒットしたページから形態素解析を用いて人名を抽出する。次に、抽出した名前と入力の関連度を計算し出力する。人名抽出の際、問題であった形態素解析辞書と名前取得の法則を改良した。また、人物間の関連度はJaccard係数による関連度計算が有効なことがわかった。今後は、2者間の関連度から計算できる距離と共通して explicit(明示的)な関係をもつ人物を用いて、2者間の関係を判定するプログラムを作成し、有効性を検証する。

参考文献

[1]総務省 平成20年 通信利用動向調査
[2]Google Social Graph APIを徹底解剖
http://japan.cnet.com/
[3]形態素解析ソフト ChaSen
http://chasen.naist.jp/hiki/ChaSen/
[4]丹羽基二著 日本苗字大辞典 芳文館出版部
[5]Software Design 2007年8月-2008年3月
レコメンドエンジン開発室