社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2003－DBS－130 (3)
2003－ＦＩ－ 71 (3)
2003／5／22

Web上のキーパーソンの発見と関係の可視化

原田 昌紀　佐藤 進也　風間 一洋
NTT 未来ねっと研究所
東京都武蔵野市緑町 3-9-11

我々は文書検索と固有表現抽出を組み合わせた情報検索手法 NEXAS(Named Entity eXtraction and Association Search) を提案する．NEXAS は，検索質問と適合する文書だけでなく，それらと関連する実世界のエンティティを発見する．これにより利用者は文書集合内の情報に加え，実世界に関する知識を得ることができる．本稿では提案手法の実際的な適用例として，与えられたトピックに関するキーパーソンを Web 上から発見するシステムについて述べる．また，キーパーソン間の関係を無向グラフとして可視化する方法を説明する．

Finding Key People and Visualizing their Relationships on the Web

Masanori HARADA, Shin-ya SATO and Kazuhiro KAZAMA
NTT Network Innovation Laboratories
3-9-11 Midori-cho, Musashino-shi, Tokyo

We propose NEXAS(Named Entity eXtraction and Association Search), an information retrieval method that combines document retrieval with named entity extraction. NEXAS finds not only documents relevant to a query but also real world entities related to them. This helps users to get real world knowledge as well as information in the document collection. As a practical application of the proposed method, this paper presents a system to find key people on the Web for a given topic. We also describe a visualization method of relationships of key people as undirected graphs.

1. はじめに

今日の情報検索システムの多くは，検索結果を適合度順に表示するランキング検索方式を採用することで，利用者が目的とする文書を効率的に検索できるようにしている．しかし，ランキング検索の精度が高かったとしても，利用者が一つ一つの文書を閲覧して初めて情報が得られることには変わりがない．その意味で，これまでの情報検索システムは文書検索システムに過ぎなかったともいえる．

そこで我々は文書検索と固有表現抽出を組み合

わせることで，検索質問と適合する文書だけでなく，それらと関連する実世界のエンティティを検索する手法である NEXAS(Named Entity eXtraction and Association Search) を提案する．これはたとえば，検索質問と適合する Web ページに加えて，それらと関連する書籍や人物を発見する Web サーチエンジンを実現しようというアプローチである．

これまでの情報検索やテキストマイニングの研究は，利用者が文書集合から情報を得ることを目的としていたため，語句の出現頻度や共起関係といった文書集合内に閉じた要素の利用に主眼が置かれていた．本手法ではこれらに加えて文書中の固有表現を利用することで，文書集合を実世界のエンティティと関連づける．その目的は，利用者が文書集合のみならず，実世界から新たな知識を獲得できるようにすること，さらには，利用者が各自持っている実世界に関する背景知識を活用し，文書集合を多様な視点から検索できるようにすることである．

本稿では NEXAS の有効性を示す例として，与えられたトピックと関連する人物を Web 上から発見するシステム NEXAS//KeyPerson について述べる．本システムは，Web における人名の共起関係を利用して，発見した人物間のネットワークを無向グラフとして可視化することもできる．

以下，第 2 節では NEXAS の提案をおこない，その一般的な枠組を述べる．第 3 節では，Web 上の人名の抽出方法を説明した後，Web サーチエンジンの検索結果からキーパーソンを発見する方法について述べる．続いて第 4 節では，発見された人物間の関連の可視化方法を説明する．第 5 節では，関連研究を簡潔に紹介する．最後に第 6 節において，まとめと今後の課題を述べる．

2. NEXAS

2.1 エンティティの定義

本稿では**エンティティ**を人物や組織，書籍，楽曲などの固有の名前を持った対象と定義する．エンティティは必ずしも物理的に実在する必要はないが（例：組織），固有の名前を持っているため，同一性を有した存在として認識される．

文書と関連するエンティティを発見するもっとも基本的な方法は，文書に含まれる**固有表現**を抽出することである．[2] によれば，固有表現とは，固有名詞（組織名・人名・地名・固有物名），時間表現（日付・時刻），数値表現（金額・割合）といった情報抽出のキー要素のことをいう．ただし，時間表現や数値表現は，ここでいうエンティティを示すものではないので，NEXAS では扱わない．一方，メールアドレスやドメイン名はエンティティの正式な名前ではないが，単独でエンティティを特定する手がかりとなるため，広義の固有表現として扱う．表 1 にエンティティと，対応する固有表現の例を示す．なお，複合名詞の扱いなど，固有表現の定義はしばしば問題になるが，NEXAS では固有表現そのものではなく，エンティティの発見を目的とするため，ここでは定義の問題には立ち入らないことにする．

現実には一つのエンティティが複数の名前を持つ場合や，一つの名前が異なるエンティティを指すこともあり，エンティティと固有表現が一対一に対応しないこともある．しかし，本稿では煩雑な表現を避けるため，これ以降，エンティティと固有表現を区別せずに述べることがある．

表 1: エンティティの例

エンティティ	文書中の表現
人物	姓名
	姓
	メールアドレス
企業	企業名
	ドメイン名
書籍	書名
	ISBN

2.2 一般的な手順

NEXAS による検索は，文書検索，適合文書からの固有表現抽出，エンティティの関連度の計算という 3 つのステップからなる．

まず，全文検索システムなどを用いて，検索質問と適合する文書群を求める．文書検索の方法は任意だが，検索精度が低ければ，検索結果から求

められるエンティティも検索質問と無関係なものになってしまう．従来の文書検索では，利用者が閲覧する可能性の高い上位数十件の精度に関心が払われていたが，本手法を用いる場合，上位数百〜数千程度の文書が検索質問とある程度適合していることが望まれる．

続いて，高い適合度を得た文書群から固有表現を抽出し，それらをエンティティ単位に正規化して列挙する．実際には処理を高速におこなうために，あらかじめすべての文書から固有表現の抽出をおこない，文書とエンティティの関係を索引づけしておく．

最後に抽出されたエンティティと検索結果文書群との関連の大きさを示す**関連度**を計算し，関連度の大きいエンティティから順に出力する．適合度の計算と同様，関連度の計算には様々なモデルが考えられるが，一般的には適合度の高い文書における出現頻度が高く，文書集合全体での出現頻度が低いエンティティほど関連度を大きくする．

後述するように大規模な情報検索システムでは，単にそれぞれのエンティティが出現した適合文書数を関連度とすれば十分であることも多い．ただし，そのためには最初のステップの文書検索の精度が十分に高い必要がある．

3. Web 上のキーパーソンの発見

3.1 ねらい

今日の Web は社会と深く結び付いており，Web 文書には多くの人々の行動や考えが反映されていると考えられる．そこで大量の Web 文書から人名を抽出して分析すれば，各分野で誰がキーパーソンとして認知されているか，また，人物間にどのようなつながりがあるかを調べることができると期待できる．

本節では NEXAS の実際的な例として，Web から与えられたトピックに関する人物を検索するシステムである NEXAS//KeyPerson について述べる．まず，テキストから日本人の人名を抽出する方法を説明し，大量の Web 文書から人名を抽出した結果を報告する．その上で，Web サーチエンジンの検索結果から，トピックと関連する人物を発見する手法を述べる．最後にいくつかの例を挙げる．

3.2 形態素解析による人名の抽出

日本語テキストから人名を抽出する方法として，人名を含む文の表記に見られるパターンを利用する方法が提案されている [2][3][4]．日本語では，人名の後ろには「さん」「氏」のような接尾語，あるいは「社長」のような役職名が多く，人名の前には「漫画家」のような職業名が多い．このようなパターンを規則化することで人名を抽出できる．

しかし，本システムでは単にテキストを形態素解析し，品詞が人名として同定された形態素の並びを人名として抽出する．これは，本システムは Web 上の多様で統制されていないテキストを対象としており，新聞記事等を対象とする場合とは異なり，あらかじめ人名が出現する典型的なパターンを調べ上げることが難しいためである．また,「さん」「氏」などの接尾語程度であれば，形態素解析の品詞同定にも反映される．

表 2: 人名抽出に用いた形態素辞書のエントリ数

	姓	名	一般
標準	17,877	12,130	2,160
追加後	21,141	40,836	19,675

具体的には形態素解析器 MeCab バージョン 0.7[5] と形態素辞書 IPADIC バージョン 2.5.0 を用い，HTML テキストからタグやコメントを取り除いたテキストを形態素解析して，次のような品詞の形態素の並びを人名として抽出する．形態素解析の結果は最適解のみを用いる．

1. (名詞-固有名詞-人名-姓)(名詞-固有名詞-人名-名)
2. (名詞-固有名詞-人名-姓)(記号-空白)(名詞-固有名詞-人名-名)
3. (名詞-固有名詞-人名-一般)

ここで示す品詞名は IPA 品詞体系のもので，(名詞-固有名詞-人名-一般) は,「ビートたけし」「二葉亭四迷」のように姓と名の分割に適さない人名に用いられる．すなわち，本システムでは現在のところ，姓，名の順で書かれた日本人のフルネームだ

けを抽出しており，アルファベット表記の人名や，姓のみの表現は対象としていない．

この方法の最大の問題は，形態素辞書に含まれていない姓や名を持つ人名が抽出できないことである．そこで，仮名漢字変換用として公開されているフリーソフトウェアの人名辞書をいくつか収集し，固有名詞として追加した．形態素辞書の登録語数を表 2 に示す．タレントや作家の名前は姓と名に分割できるものであっても，(名詞-固有名詞-人名-名) として追加した．

3.3 抽出性能の評価

形態素解析のみを用いた人名抽出方法の性能を評価するため，本研究会のプログラムの Web ページ*を対象として，それに含まれる日本人の人名（正解 68 個）を抽出する簡単な実験をおこなった．

結果を表 3 に示す．ここで精度，再現率，F 値は次のように定義される．

$$\text{精度 } P = \frac{\text{正しく抽出された人名数}}{\text{抽出された人名数}}$$

$$\text{再現率 } R = \frac{\text{正しく抽出された人名数}}{\text{文書中の人名数}}$$

$$\text{F 値} = \frac{2PR}{P+R}$$

文書一つのみの簡単な評価ではあるが，追加後の形態素辞書を用いた場合には，実用的な精度・再現率が達成された．ただし，追加後も，誤抽出・抽出もれの原因の大半は未登録の姓あるいは名によるものであった．

表 3: 抽出精度の評価

	抽出数	精度 (誤抽出)	再現率 (抽出もれ)	F 値
標準	65	0.784 (14)	0.750 (8)	0.767
追加後	62	0.935 (4)	0.853 (6)	0.892

3.4 Web 上に存在する人名数

上述の方法を用いて，2001 年 12 月に JP ドメインを中心に収集した Web ページ約 4,300 万ページから人名の抽出をおこなった結果を表 4 に示す．2 割強の文書から人名が抽出されており，抽出された人名数は文書数を上回っている．このように高い頻度で人名が出現していることは提案手法によってキーパーソンを発見できる一つの根拠となっている．

表 4: Web 文書集合から抽出された人名数

文書数	43,090,336
人名が抽出された文書数	10,028,348
抽出された人名の数	66,860,851
ユニークな人名数	4,242,519

この文書集合から，最も高い頻度で抽出された人名を表 5 に示す．これは Web における知名度ランキングといえるが，その解釈にはいくつかの注意が必要である．

まず，この表では抽出された数を出現数として示しているが，実際には抽出もれがあるため，より高い頻度で出現している人名がある可能性がある．また，一戸建（いちのへ けん）のような誤抽出も含まれている．

単純な出現回数の他に文書単位，ディレクトリ単位，サーバ単位の出現頻度を示しているのは，機械的に生成されたテキストの影響を明らかにするためである．たとえば，ライターの塩田紳二氏，元麻布春男氏の出現回数が大きいのは，ある Web サイトのほぼ全てのページに彼らの書いたコラムへのリンクがあったためである．

また，近年は Web サーチエンジンで検索される確率を高めるために，有名タレントの名前を機械的に羅列するなどのスパム行為をおこなう Web ページも多く，それらの影響を受けている可能性もある．出現サーバ数はこうした問題の影響を受けにくいが，大規模なサーバと小規模なサーバを同列に扱っているため，正確に知名度を表すものとは言いがたい．

3.5 検索結果からのキーパーソンの発見

本システムにおけるキーパーソンの発見手順は次の通りである．

まず Web サーチエンジン ODIN を用いて全文検索をおこない，適合度上位最大 1,000 件の Web ページを求める．続いて，それらの文書から上述

*http://www.ipsj.or.jp/katsudou/sig/kaikoku/DBS130FI71.html (2003 年 4 月 15 日現在)

表 5: 出現頻度の高い人名のランキング

出現回数		出現文書数		出現ディレクトリ数		出現サーバ数	
123,327	浜崎あゆみ	69,367	浜崎あゆみ	43,109	浜崎あゆみ	6,768	浜崎あゆみ
93,581	宇多田ヒカル	54,862	宇多田ヒカル	36,989	宇多田ヒカル	6,185	宇多田ヒカル
72,766	塩田紳二	41,912	一戸建	21,582	松浦亜弥	5,622	徳川家康
71,053	手塚治虫	35,258	手塚治虫	20,492	平井堅	5,271	手塚治虫
53,766	椎名林檎	33,170	椎名林檎	20,447	矢井田瞳	5,163	豊臣秀吉
53,370	広末涼子	31,623	倉木麻衣	19,517	椎名林檎	5,105	織田信長
51,390	一戸建	28,503	松浦亜弥	19,088	河村隆一	4,843	椎名林檎
50,324	宮沢賢治	28,295	中田英寿	18,360	手塚治虫	4,823	宮沢賢治
45,351	倉木麻衣	27,894	広末涼子	17,521	藤木直人	4,689	夏目漱石
36,832	松浦亜弥	25,919	元麻布春男	17,332	上田寛	4,282	司馬遼太郎

の方法で抽出された人名をキーパーソンの候補として列挙する．現在のところ，抽出された人名をエンティティ（人物）として扱っている．最後にそれぞれのエンティティの関連度を計算し，関連度が大きい順にキーパーソンとして出力する．

もっとも簡単な関連度の計算方法として，適合度上位 1,000 件の文書集合における文書頻度（以下，適合文書頻度）を用いる方法がある．この方法は単純ではあるが，多くの検索質問に対して良好な結果が得られる．特に検索結果文書数が大きいときには，その上位 1,000 件は検索質問と適合したものになりやすいので，それらの多くに出現する人物はトピックと強く関連していると期待できる．

しかし，この方法では文書集合全体での出現頻度を考慮していないため，トピックとの関連がそれほど大きくない有名人が出力されやすい．そこで，以下ではある文書が検索結果になることと，その文書に人名が出現することに関連があるかを統計的な方法で調べる方法を説明する．ここでは関連度として，対数尤度比検定で用いられる G スコアを用いる [6].

G スコアの計算方法は次の通りである．

$$G = 2 \times \left(a \log \frac{aN}{(a+b)(a+c)} + b \log \frac{bN}{(a+b)(b+d)} + c \log \frac{cN}{(a+c)(c+d)} + d \log \frac{bN}{(b+d)(c+d)} \right)$$

ここで N は検索対象となる文書集合のサイズ（文書の総数）であり，a，b，c，d は，検索結果文書の集合 D と，人名を含むすべての文書の集合 K の重なりを示す次のような表の要素である．

	K	$\overline{K}$	計
D	a	b	$a+b$
$\overline{D}$	c	d	$c+d$
計	$a+c$	$b+d$	N

つまり，a，b，c，d は次のような関係から簡単に計算できる．

$$\begin{aligned} N &= a+b+c+d \\ |D \cap K| &= a \\ |D| &= a+b \\ |K| &= a+c \end{aligned}$$

$|K|$ および $|D \cap K|$ の値は全文検索によって求めることもできるが，計算を高速化するために，人名の出現位置の索引を別途用意している．

4 つのトピックについてキーパーソンを発見した例を表 6 に示す.「マラソン」では有名選手と指導者が,「ノーベル賞」については 2001 年末時点での受賞者が的確に発見されている．マラソンの検索結果文書数が比較的大きいのに対して，ノーベル賞の検索結果文書数は中程度という違いはあるが，いずれの場合も検索結果の上位 1,000 件はトピックと適合しているように見えた．このような場合には，単に適合文書頻度を関連度としても，ほぼ同様の結果が得られる．

一方,「情報検索」で検索される Web ページには，研究分野としての情報検索に関連しないものも多く，様々な検索サービスの Web ページが含まれている．また,「java」の場合は検索結果の上位に英語の Web ページや，技術文書が多く，日本人

表 6: 発見されたキーパーソンの例（左から人名，適合文書頻度，G．括弧内は検索結果文書数）

マラソン (136,495)			ノーベル賞 (23,421)			情報検索 (78,550)			java(503,861)		
高橋尚子	119	92331.9	白川英樹	222	12298.9	上田修一	4	1379.1	高木浩光	6	9515.0
有森裕子	27	17872.9	湯川秀樹	117	9222.1	野口悠紀雄	3	985.6	風間一洋	8	4353.9
小出義雄	24	12325.3	大江健三郎	120	8769.9	中川裕志	9	942.6	萩本順三	6	4013.4
山口衛	32	11913.1	野依良治	166	8013.5	原田昌紀	6	812.5	中川真実	3	3701.9
市橋有里	29	9820.3	利根川進	111	6102.2	吉川正俊	2	750.5	結城浩	2	3350.6
弘山晴美	18	8850.8	江崎玲於奈	122	5635.3	井佐原均	2	739.9	新居雅行	19	3085.9
谷川真理	20	8116.0	福井謙一	93	4476.3	徳永健伸	3	737.4	白根雅彦	8	3079.4
渋井陽子	37	7571.7	朝永振一郎	82	4226.5	長谷川豊	2	733.8	戸松豊和	7	2628.4
藤田敦史	15	7375.0	川端康成	67	3878.4	山名早人	6	731.7	首藤一幸	2	2489.4
増田明美	10	6412.9	金大中	45	2827.0	松本裕治	1	722.6	元麻布春男	6	2109.5

の人名があまり含まれていなかった．こうしたトピックでは適合文書頻度は一様に小さくなり，差がつきにくい．そのために，一般的な出現頻度が高い有名人や，無名な人物が選ばれやすくなってしまう．しかし，G スコアを使い，文書集合全体での関連を見ることで，トピックと関連のあるキーパーソンを多く発見できている．

ただし，対数尤度比は個々の文書の適合度を考慮せずに，多数の文書を使って計算されるため，機械的に生成されたテキストを含む Web ページの影響を受けやすい面も見られた．java と強い関係があるとは思えない元麻布春男氏がキーパーソンとして発見されているのはこのためである．

これら以外にもさまざまな種類の関連度の定義が考えられる．発見精度の向上と，詳細な評価は今後の課題としたい．

4. 人物関係の可視化

4.1 人物間のつながりの強さ

前節で述べた方法によって発見された人々は，共通のトピックに関連しているという意味で，互いに関係している．しかし，それはいわば間接的な関係である．そこで本節では発見された人物間の直接的なつながりの強さを数値化し，それによって人々のネットワークを可視化する方法を説明する．

人物間の直接的なつながりの強さは，文書における人名の共起の頻度から推定できると考えられる．人名が一度共起しただけでは，それらの人物間につながりがあるとは限らないが，共起の回数が大きければ，人物間に強いつながりがある可能性が高い．

本システムでは次のように定義される**共起度**を用いる．ここで，ある文書 d 内で人名 a が i 番目に抽出され，人名 b が j 番目に抽出されたとき，$|i-j|$ が定数 R 以下であれば，a と b は距離 $|i-j|$ で共起するという．ただし，重複を除くため，i と j の間で抽出される人名は a，b のいずれでもないものとする．このとき，人名 a と人名 b の共起度 $C(a,b)$ は，文書集合内のすべての文書における a と b のすべての共起について，距離の逆数を足し合わせた値として定義する．

人物間のつながりの強さを示す指標として，Jaccard 係数を用いる方法もある [7]．この場合，Jaccard 係数は 2 つの人名の両方を含む文書の数を，いずれかを含む文書の数で除した値になる．しかし，Jaccard 係数は出現文書数に大きな差があると小さくなるため，有名な人物はそうでない人物とあまり関連しないと見なされることになる．これは有名な人物は他の多くの人物と関連するという直感に反している．

4.2 無向グラフによる可視化

人名の組と共起度を表示するだけでは，発見された人々の関係を直感的に理解することは難しい．そこで，本システムでは人物間のネットワークを無向グラフとして可視化する．すなわち，人物ををグラフのノードとして表示し，共起関係にある人物間にエッジを表示する．

ただし，共起する人物間をすべてエッジとして表示すると，グラフの密度が高くなり，ネットワークの構造を読み取ることが難しくなることが多い

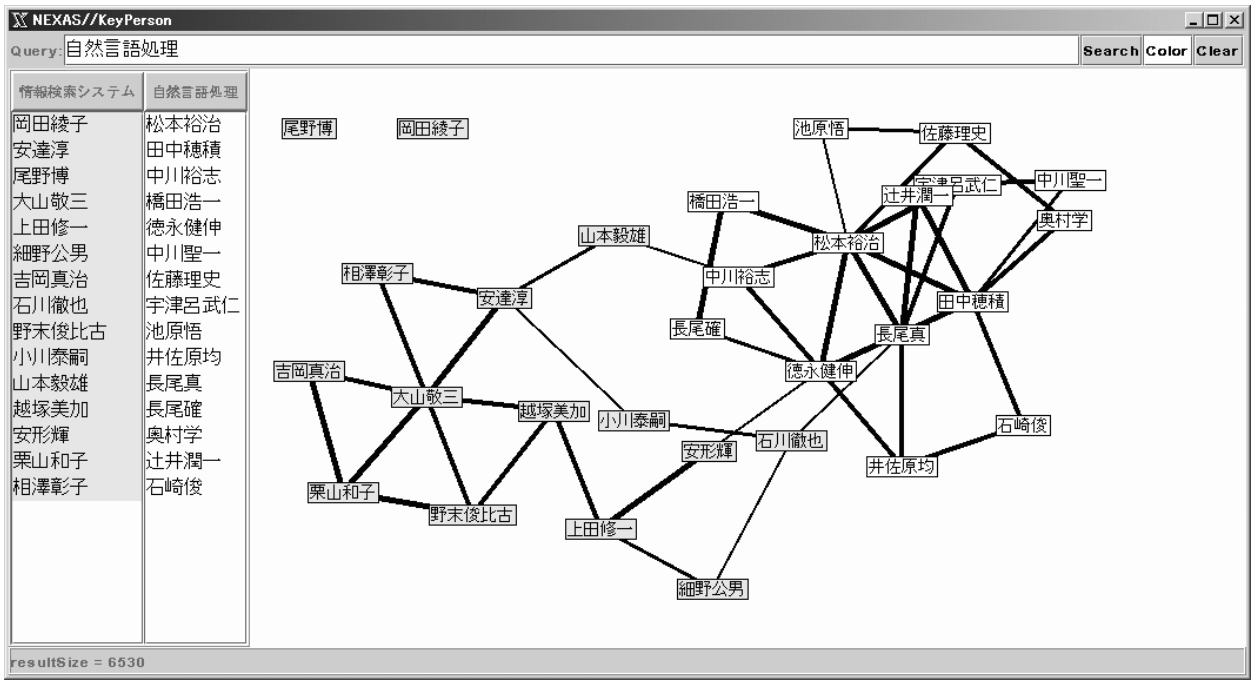

図 1: 無向グラフによる可視化の例

ため，表示するエッジを選択する必要がある．もっとも単純な方法は，ある閾値より高い共起度を持つエッジだけを残すことである．しかし，それでは一部の集団にのみにエッジが集中し，それ以外の人物はすべて孤立したノードになることが多い．これは，後に述べるように複数のコミュニティを同時に表示する際に特に問題になる．

そこで，本システムではそれぞれのノードごとに，共起度の高い相手ノードを最大 2 つ選び，それらの間のエッジを表示する．このようにしても，他の多くの人物と強いつながりを持っている人物のノードには多くのエッジが集まり，中心的な人物であることがわかる．

例として「情報検索システム」「自然言語処理」という 2 つのトピックでそれぞれ 15 名ずつキーパーソンを発見し，そのつながりを可視化した様子を図 1 に示す．それぞれの人物がどちらのトピックで検索されたかはノードの色によって示される．複数のトピックで同じ人物が発見されたときには中間色が用いられる．共起度の大きいエッジは太い線で表示している．共起の最大距離 R は 3 とした．

本システムでは，このように 2 つのトピックで発見された人々を同時に表示することで，両方のトピックと関連した人物の存在を明らかにすることができる．こうした人物は，2 つのコミュニティの間を繋ぐ重要な人物である可能性が高いと考えられる．

5. 関連研究

Web 上の人名を収集し人間関係を可視化するシステムとして Kautz らによる REFERRAL WEB がある [7]．REFERRAL WEB はまず，シードとして与えられた人名の出現する Web ページを Web サーチエンジンを用いて収集し，それらに出現する人名を抽出する．そして，それらの中でシードと強い共起関係にある人名に対して同様の処理を繰り返すことで，人物間のネットワークを構築していく．REFERRAL WEB は小規模なコミュニティのネットワークを事前に構築し，その上で特定の専門用語と関連する人物の検索などをおこなう．一方，我々のシステムは大規模な文書集合から任意のトピックに対して関連する人々を検索し，それらの人物間のネットワークを可視化する．

Ogata らによる SocialPathFinder は Web ロボッ

トを用いて，起点として与えられた Web ページの周辺から個人の Web ページを収集し，人物間の関係を抽出する [8]．そのため，本システムのように広範囲に散在した人物情報を高速に検索することはできない．

山本らは「政治家」などの職業名を入力として，Web サーチエンジンとハイパーリンクを利用して，特定の職業の人物情報を網羅的に収集する方法を提案している [9]．彼女らの方法はターゲットとなる職業に関して，表形式で書かれた人名録が存在することを前提にしており，我々のシステムのように任意のトピックを扱うことはできない．

6. まとめ

本稿では，固有表現を抽出することで，大量の文書と実世界のエンティティを関連づける情報検索手法である NEXAS を提案した．その実例として，Web 上から与えられたトピックにおけるキーパーソンを発見し，それらの人物間のネットワークを可視化するシステム NEXAS//KeyPerson について述べた．

本システムに用いた人名抽出方法は非常に単純であり，精度・再現率共に向上の余地は大きい．また，関連度の計算方法も，文書の大きさや検索語の出現位置を考慮しない単純なものである．しかし，簡単な方法の組み合わせで実現された現在のシステムでも，多くのトピックで良好な結果が得られている．このことは提案手法の妥当さを示している．

提案手法の定量的な評価は今後の課題である．他の関連度の計算方法も検討し，どのような方法が有効か明らかにしていきたい．また，人物だけでなく，組織や書籍など，他の方法の発見も試みていきたい．

参考文献

[1] 関根 聡:“テキストからの情報抽出,” 情報処理, Vol.40, No.4, pp.370–373, 1999.

[2] 竹元義美，福島俊一，山田洋志: “辞書およびパターンマッチルールの増強と品質強化に基づく日本語固有表現抽出,” 情報処理学会論文誌, Vol.42, No.6, pp.1580–1591, 2001.

[3] 久光 徹, 丹羽芳樹: “辞書と共起情報を用いた新聞記事からの人名獲得,” 情報処理学会研究報告, NL118-1, pp.1–6, 1997.

[4] 西野文人, 落谷亮: “新聞記事からの人物・企業情報の抽出,” 情報処理学会研究報告，NL127-17，pp.125-132, 1998.

[5] 工藤 拓: “MeCab: Yet Another Part-of-Speech and Morphological Analyzer,” Version 0.7b, 2002.

[6] Ted E. Dunning, “Accurate Methods for the Statistics of Surprise and Coincidence,” Computational Linguistics, Vol. 19, No. 1, pp. 61–74, 1993.

[7] Kautz, H., Selman, B. and Shah, M.: “The Hidden Web,” AI Magazine, Vol.18, No.2, pp.27–36, 1997.

[8] Ogata, H., Fukui, T. and Yano, Y.: “SocialPathFinder: Computer Supported Exploration of Social Networks on WWW”, ICCE 99, Vol.2, pp.768–771, 1999.

[9] 山本あゆみ, 佐藤理史: “ワールドワイドウェブからの人物情報の自動収集,” 情報処理学会研究報告, 2000-ICS-119-24, pp.173–180, 2000.