Web 情報を用いた人物の愛称抽出

Extracting Nicknames corresponding to a Formal Name using the Web

若木 裕美† 藤井 寛子†
福井 美佳† 住田 一男†

Hiromi WAKAKI Hiroko FUJII
Mika FUKUI Kazuo SUMITA

近年，音声認識技術を用いたヒューマンインタフェースの実用化が進んでおり，多様な入力表現への対応が求められている．本稿ではWeb情報を用いて，人名に対応した愛称を自動的に抽出する手法を提案する．まず愛称生成ルールを学習しそのルールから愛称候補を自動生成する．加えて，愛称を明示的に示す典型表現“○○こと(名称)”などからの文字列抽出を行う．これらの2つの愛称候補群を組み合わせて選別し，愛称抽出結果とする．提案手法と先行研究の手法に対し，本稿で正解と定義したデータを用いて比較評価を行った結果，提案手法は MAP 評価で 18%の精度向上がみられた．

Recently, speech recognition technique has been put to practical use in human interface. Therefore, it is required to be able to deal with various expressions. In this paper, we propose a method to extract nicknames using the Web. We first generated two kinds of nickname lists from a formal name and merged them into one nickname list. The nicknames on one of the lists were generated by using nickname generation rules learned by given pairs of formal names and his/her nicknames. The others were extracted on the Web by using a pattern phrase in Japanese. We also conducted an experiment to compare two existing methods with our proposed method.

1. はじめに

近年，音声認識技術を用いたヒューマンインタフェースの実用化が進んでおり，今後は音声対話や音声による情報検索が行われることが想定される．音声による情報検索処理では，テキスト検索に比べ，多様な入力表現への対応が求められる．特に人物名は，愛称で呼ばれる可能性がある．愛称とは，大辞林[1]によれば“本名以外の，親しみをこめて呼ぶ呼び名．”であり，ユーザの自然な発話を促進するために，愛称を人物の名称に対応付けて認識できることが必要である．そこで，本稿ではWeb情報を用いた，人名に対応する愛称の自動抽出手法を提案する．

愛称には，大きく分けて２種類あると考える．１つは，名前由来の造語，もう１つは，名前とは無関係の表現である．名前由来には，芸能人の及川光博を“ミッチー”と呼ぶように，元表記や読みの一部を利用し，元の文字にはない文字を挿入した愛称がある．また，木村拓哉を“キムタク”と呼ぶように，元表記や読みの省略形で作られる愛称もある．一方，名前由来でない愛称とは，斎藤祐樹選手を“ハンカチ王子”と呼ぶような場合である．いずれにせよ愛称が造語である場合，形態素解析辞書に単語がないことが想定され，愛称文字列抽出は困難である．加えて，造語であるかを問わず，人物の愛称を発見し対応付けを行うのは難しい．

また，愛称は自然発生的に呼ばれるようになるため，愛称の正解定義は難しく評価実験を行いにくい．既存研究で人物の呼称抽出[2]や別名抽出[3]があるが，呼称や別名の明確な定義づけがなされていない．また，“愛称”以外にも，通称，別名，異名，呼称，ニックネーム，あだ名等の人物の呼び名を示す表現がある．本稿ではこれらの違いは区別せず，人物名の言い換え表現のうち一般的に通用するものを愛称とする．一般に通用している愛称としてWikipedia[1]の文章中から愛称等の記載を抜粋し，正解データの作成を行った．

本研究では，まず，人物の名前に由来して人間が愛称を付けるルールを学習し愛称抽出を行う．すなわち人名とその愛称の組の学習データから愛称生成ルールを学習し，学習した愛称生成ルールと愛称を知りたい人名から愛称候補を生成する．次に名前由来でない愛称の抽出も行う．愛称と人名の対応付けを明示的に表す表現パターン“○○こと(名称)”を利用して，Web上で「こと」の前に現れる文字列から愛称候補を抽出する．上記２つの愛称候補リストを組み合わせて愛称抽出結果とする．評価実験では，２つの愛称候補リストを組み合わせることで単独の精度よりも向上することを示す．

2. 関連研究

人物の愛称抽出には，２つの技術的な課題が関わっている．１つは，名称と愛称の対応付けである．もう１つは，単語の境界が不明確な場合の愛称文字列抽出である．

名称と愛称の対応付けのように，２つの語の関係性から対応付けを抽出する方法には，手や辞書で抽出ルールを作成するほかに機械学習の手法が応用されている．例えば，シェークスピアとハムレットをシードにして，著者名と本の題名のような特定の関係にある表現をコーパスから取り出す方法がある[4][5]．また，略語とその原語の対応付けでは，略語のすべての文字が含まれる表現を原語とするルールを用いて対応付ける方法がある[6]．略語が生成される幾つかのルールやスコア関数を学習で統合する方法もある[7]．日本語の略語生成規則は英語に比べ複雑であることから，略語候補の自動生成を行いWeb上の情報を利用して略語候補の絞込みを行う手法もある[8]．その他の対応付けには，言語的な手がかり句を使って知識を集めるという研究がある．例えば，Hearst は“A such as B”という表現を利用して上位下位概念の単語を収集する[9]．また，外間らは人物名の別の呼び名を明示する表現「“○○こと(名称)”」を利用し，「こと」の前の単語を抽出することで呼称表現の抽出をする[2]．

一方，日本語のように単語の境界が不明確な言語の場合，愛称を表す文字列の範囲を特定する必要がある．例えば，外間ら[2]の呼称表現抽出では形態素解析を用いるが，形態素解析辞書の項目に愛称が含まれないとき，「こと」の直前の文字列抽出で失敗する場合がある．そこで，本間ら[3]は，その文字列を重ね合わせた木構造を作り，分岐点での相対頻度によって木構造中の分岐を選択し，文字列を抽出している．

†株式会社 東芝 研究開発センター {hiromi.wakaki, hiroko.fujii, mika.fukui, kazuo.sumita}@toshiba.co.jp

1 http://ja.wikipedia.org/

3. 愛称の分析

本稿では，次のように愛称を定義する．

人物名の言い換え表現のうち一般的に通用するものを愛称とする．

実験のため，一般に通用されているかどうかは，Wikipedia中に愛称等の人物に対する他の呼び名が記載されているかどうかで判断することとした．その呼び名を愛称とする．

まず，調査用に愛称リストを次のようにして作成した．はてな[2]のニックネームリスト3種類[3]に愛称がある有名人として記載されていた136人の人名を収集した．このうち，Wikipedia上に愛称等が記載され，かつ，正式名称が姓名で構成される108人を対象としてWikipedia中の記載から人手で愛称を抽出した．その結果，1人に対して複数の愛称がある場合があり，対象とした108名に対して257個の愛称が取得できた．ただし，芸能人の本名としての別名は除外した．

ここで，取得した愛称が名前由来であるかを調べた．その結果，257個の愛称のうち，名前由来の愛称が86%(222個)，名前由来でない愛称が14%(35個)であった．また，名前由来の愛称のうち[4]，正式名称由来が30%(70個)，読みのひらがなが47%(110個)，読みのカタカナが23%(55個)であった．名前由来の愛称である場合には，例えば，

・接尾辞が付く場合（○○ちゃん，○○君，○○たんなど）
・省略される場合（キムタク，まつじゅんなど）
・加工される場合（ユーミン，まっすーなど）

のように幾つかのルールがある．このような愛称は，名前の一部を使って加工した単語であり造語となる．また，姓名のいずれから由来したかの割合を調べたところ，222個の名前由来の愛称のうち，姓由来が35%(78個)，名由来が58%(129個)，姓名から由来が7%(15個)であった．一方，名前由来でない場合には，名前とは別の事柄から付く．既存の単語やその組み合わせで表現されることが多い．

4. 提案手法

4.1 方針

名前由来の愛称取得のための愛称生成を行う．まず，名前とその愛称の組を学習データとして与え，人間が愛称を付けるルールを学習させる(4.3節)．次に人名を入力に与え，愛称生成ルールから愛称候補を生成する(4.4節)．

しかしこの方法だけでは，名前由来でない愛称が取得できない．そこで，本間ら[3]の手法を応用して用いる．“(愛称)こと(名称)”という表現があることを想定して，“こと(名称)”という検索語で検索を行い，「こと」の前の文字列から愛称として1語を形成すると思われる部分を抽出する．こうして名前由来でない愛称も含めて文字列の取得を行う(4.5節)．

この 2 手法による愛称候補の組み合わせを愛称抽出結果とする(4.6節)．

4.2 前処理

正式名称で記載された文書中では，プロフィール紹介や言い換えで愛称が使用されることがあるため，愛称と共起することが想定される．そこで，前処理として Web 上で正式名称を検索語とする検索結果上位 n 件の文書をダウンロードしておき，愛称候補の絞り込みに利用する．ここで，正式名称の姓名を *fullname*，ダウンロードした文書を *Page* とおく．

4.3 愛称生成ルールの学習

名前とその愛称の組を学習データとして与え愛称生成ルールを学習する方法について述べる．1)正式名称（姓/名），2)その読みのひらがな，3)その読みのカタカナ，4)愛称の4つを入力として，愛称を生成するルールを学習させる．

入力された名称に対し，次に示すルールで番号を与え，1文字ごとに3桁の数字へ置換する．まず，100の位の数値は文字種を表し，「1」が名称，「2」がひらがな表記，「3」がカタカナ表記に対応する．10 の位の数値は名称を構成する単語の位置を表し，人名ならば「1」が姓，「2」が名に対応する．1の位の数値は各単語中での先頭からの位置を表す．

例えば，正式名称が`滝沢秀明"で，愛称が“タッキー”のとき，次のようにしてルールを学習する（図1参照）．名称，読みのひらがな，読みのカタカナのそれぞれと愛称文字列を比較し，一致する文字を探す．すると，愛称に含まれる文字として“タ”と“キ” が見つかる．それぞれ記号に変換すると 311 と 312 なので，愛称の文字のうち名称と合致した部分をこの数字に置き換え，“311 ッ 312 ー”という愛称生成ルールを学習する．こうして与えた名称や読みと愛称から，愛称生成ルールを学習し保存する．

・正式名称の「滝沢 秀明」の場合

	文字種	姓名	文字位置	記号
滝	1	1	1	111
沢	1	1	2	112
秀	1	2	1	121
明	1	2	2	122

・カタカナ表記の「タキザワ ヒデアキ」の場合

	文字種	姓名	文字位置	記号
タ	3	1	1	311
キ	3	1	2	312
:	:	:	:	:
ア	3	2	3	323
キ	3	2	4	324

図1 人名「滝沢秀明」の場合の記号への変換方法．

Fig.1 Converting characters of the name into symbols in the case of the celebrity, “Hideaki Takizawa”.

4.4 愛称生成ルールを用いた愛称抽出

4.3 節の学習により得られた愛称生成ルールを元に，新たな人名に対応する愛称候補を生成する方法を述べる．

1) 4.3 節の学習により得た愛称生成ルールを用いて，新たな人名(正式名称，ひらがな表記，カタカナ表記)に対し愛称候補を生成し，これを *cand* とする．例えば，愛称生成ルールとして、“311 ッ 312 ー”や“121 ち ゃ ん”，“111 112 ちゃん”を保持するとき，新たな人名“槇原敬之”（槇原/敬之，まきはら/のりゆき，マキハラ/ノリユキ）を与えると，“マッキー”，“敬ちゃん”，“槇原ちゃん”を生成する．

2) 次に，候補 *cand* を次のようにして選別する．

2-a) 前処理で取得した *Page* 中で *fullname* の前後 *s* 文字を取得し，それらをスニペット *Snippet* とする．

2-b) 各愛称候補 *cand* で1回以上 *Snippet* 中に出現した愛称候補 *nick* を得る．

2-c) Web 上で“*nick* こと *fullname*”の連語を検索語として検索を行い，検索結果数が0の *nick* は削除する．

---

[2] http://www.hatena.ne.jp/

[3] リスト::ジャニーズメンバーのニックネーム，リスト::ハロプロメンバーのニックネーム，リスト::アーティストのニックネームの3種類

[4] 正式名称がひらがな表記等の場合があるため，計 235 個

また，*nick* は検索結果数の降順で並べる．

2-d) *nick* のうち，末尾が共通する部分文字列になっている場合，部分文字列になっている *nick* を削除する．例えば，*nick* 中に，“さとちゃん”と“とちゃん”があった場合，“とちゃん”は“さとちゃん”と末尾が共通する部分文字列なので削除する．

4.5 “こと *fullname*”を用いた愛称候補抽出

“こと *fullname*”の前の文字列から下記のように部分文字列を抽出し愛称候補とする．“単語内部では後続する文字の種類（分岐数）が少なく，単語が終了すると分岐数が増加に転じる”という性質[10]があると考えられる．そこで，分岐数が少ないところは単語であり，分岐数が増えるところが単語の境目であると考え，以下に示すように愛称を抽出する．

ただし，文書集合 D 中において，ある文字列 S(例えば“大ちゃん”)の直前の 1 文字が何であるかをチェックし，その 1 文字の種類数を，D 中での S の直前の異なり数と呼ぶことにする(図 3 参照)．また，S の先頭の 1 文字を取り除いた文字列(例えば“ちゃん”)の直前の異なり数を，D 中での S の先頭の異なり数と呼ぶことにする．図 3 では，“大ちゃん”の直前の異なり数は「は」「の」で 2，“大ちゃん”の先頭の異なり数は「大」「お」「ー」で 3 と数える．異なり数を数えるときは，カタカナはひらがな表記に変換して数える．

1) “こと *fullname*”で検索をし，上位 n 件のスニペットを取得し，“こと *fullname*”の前の文字列 t 文字を集める（この文字列集合を *Str* とする）．
2) 2)の各文字列から全接尾辞を得る．(図 2 参照．)
3) 各接尾辞について，*Page* と *Str* 中での直前の異なり数と先頭の異なり数を数える．ただし，文字列長が 1 の文字列の場合は，*Page* 中はチェックしない．
4) 各接尾辞について，*直前の異なり数>1* かつ *先頭の異なり数=1* のとき，その接尾辞を *Data* として登録する．また，各接尾辞について，*直前の異なり数=1* かつ *先頭の異なり数>1* のとき，その接尾辞の先頭の 1 文字を除いた残りの文字列を *Data* として登録する．
5) *Data* に含まれる文字列のうち，*Page* 中で一回も出現しない文字列は *Data* から削除する．
6) *Data* 中の文字列のうち，文字列 str_i が文字列 str_j の末尾から共通する部分文字列であるとき，*Str* 中でのそれぞれの頻度を $freq_i$ と $freq_j$ とし，差分 $\Delta_{ij}=freq_i - freq_j$ を計算する．そして，$\Delta_{ij} < freq_i * q$ のとき str_i を *Data* から削除する(図 4 中(a))．また，$\Delta_{ij} < freq_j$ のとき str_j を *Data* から削除する(図 4 中(b))．
7) *Data* に含まれる文字列のうち，文字列長が r 文字以上の文字列を愛称候補 *nick* とする．
8) Web 上で“*nick* こと *fullname*”の連語を検索語として検索を行い，検索結果数が 0 の *nick* は削除する．また，*nick* は検索結果数の降順で並べる．

4.6 2抽出手法の出力組み合わせ

4.4 と 4.5 の手法による出力を組み合わせる．4.4 から得られた $nick_i$ と 4.5 から得られた $nick_j$ があるとき，$nick_i$ の順位付けされたリストを *List* とする．そして，$nick_j$ の上位から順に検索結果数に応じた順位に *List* 中に追加する．ただし，$nick_j$ が *List* 中のいずれかの文字列を，末尾の共通する部分文字列とするか，または部分文字列とされる場合には追加しない．例えば，$nick_i$ に“大ちゃん”があり，$nick_j$ に“ちゃん”があった場合，“ちゃん”は“大ちゃん”の末尾の共通する部分文字なので追加しない．

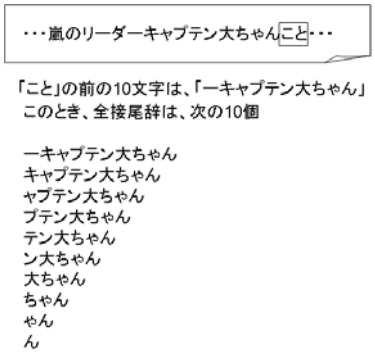

図 2 接尾辞の作り方の例.

Fig.2 An example of suffixes extracted from a text.

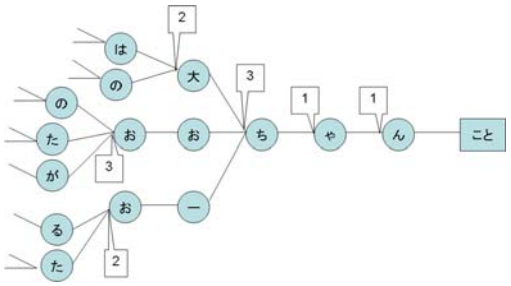

図 3 1 文字前の異なり数の数え方の例.

Fig.3 Counting types in previous strings.

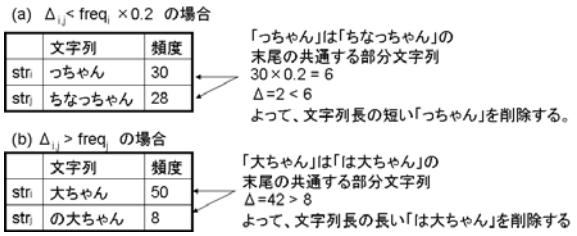

図 4 文字列の削除.

Fig.4 Deletion of strings.

5. 評価実験

5.1 実験の目的

提案手法の愛称抽出の精度評価を行う．また，Wikipedia データベース[11]から自動抽出した愛称や，先行研究である外間らの手法[2]や本間らの手法[3]を用いて抽出できた愛称と，提案手法で抽出できた愛称の比較を行う．

5.2 評価用データ

3 章で収集した 108 名の愛称 257 個を評価用の正解データとして用いる．108 名の姓名とその読みのひらがなとカタカナを入力データとして与える．

5.3 評価指標

愛称候補は順位付きで出力されることが想定されるため，情報検索の精度測定方法を利用する．TREC[5]やNTCIR[6]等の

[5] http://trec.nist.gov/

[6] http://research.nii.ac.jp/ntcir/index-ja.html

情報検索システムの精度評価を行うワークショップでは，複数の検索質問に対する平均の性能評価方法として，平均適合率の平均(Mean Average Precision; MAP)や平均逆順位(Mean Reciprocal Rank; MRR)等が用いられる[12][13]．MRRと異なり，MAPは精度と再現率の両方を重視し複数個の正解を考慮するため，本稿ではMAPを用いる．

MAP とは，各再現率ポイントでの適合率に対する平均(平均適合率，AP)を算出し，その平均適合率について全質問での平均を求めたものをさす．各再現率ポイントでの適合率とは，対象人物の正解愛称が出現した各順位において，その順位以上に含まれる正解愛称の割合をさす．ただし，愛称候補中に現れなかった正解愛称についての適合率は 0 とみなす．このとき MAP は次のような式で表される．

$$\Pr ec_f(k) = \frac{|C_f(k) \wedge \mathrm{Re}l_f|}{k}$$

$$AP_f = \frac{1}{|\mathrm{Re}l_f|}\sum_k I(k)\Pr ec_f(k)$$

$$MAP = \frac{1}{|D|}\sum_{f \in D} AP_f$$

ただし，fという人物に対して得られた愛称候補について，第 k 位での適合率を $Prec_f(k)$，平均適合率を AP_fとする．また，Dを評価用データにある人名集合，Rel_fを人物 fの正解愛称集合，$C_f(k)$を人物 fに対する第 k 位目までの愛称候補集合，$I(k)$を第 k 位目で正否（正解のとき 1，不正解のとき 0 を取る値）を表すとする．MAP は 0 から 1 の間をとる値である．愛称候補の上位 10 位までを対象として評価を行った．

また，システムの出力した愛称候補の信頼性を測るため，$MPrec_k$を次のように定義し，評価に用いた．これは，上位 k 位(ここでは $k=5$ とした)の k 個の愛称候補の適合率を，評価用データ全体について平均をとる．

$$M\ \Pr ec_k = \frac{1}{|D^*|}\sum_{f \in D} \Pr ec^*{}_f(k)$$

ただし，$j<k$ である第 j 番目までしかシステムが愛称候補を出力しない場合は，$Prec^*_k(f)$は j 位までの適合率 $Prec_j(f)$とする．また，1 つも愛称候補が出力されなかった人物については評価対象から外す．つまり，システムから愛称候補の出力が 1 つ以上得られた人物の集合を D^*とするとき，この D^*における適合率の平均を算出する．

加えて，情報検索で用いられる R-Precision でも評価を行った．R-Precision は，各質問に用意された正解の数の順位までの適合率を見るものである．そこで，各人物の正解とする愛称の数までの適合率を測り，全人物での平均を測った値を MRPrec とおく．MRPrec は 0 から 1 の間の値をとる．

5.4 提案手法の適用

5.4.1 学習用データの自動生成

学習用データの自動生成のために，Wikipedia で提供されるデータベース[11]を利用する．以下，Wikipedia から 1)愛称抽出，2)姓名分割，3)姓名の読み抽出をする方法を述べる．

Wikipedia では，各人の情報としてタグ中({{}}で囲まれた部分)に，愛称（または Alias）が記載されている(“愛称=”の部分．以下，愛称タグと呼ぶ)．例えば，小倉優子の場合，「ゆうこりん」が愛称タグ中に記載されている(図 5 参照)．この愛称タグを利用して，人名に対応する愛称を抽出できる．

また，Wikipedia では各人の解説の始まりに姓名がスペースで区切られて記載され，その次に括弧書きで姓名の読みが記載されることがある(図 5 参照)．そこでこの形式を利用し，Wikipedia から人名の姓名分割と，姓名の読みを抽出する．

次に，学習用データの素となる人名リストを作成する．タレント，俳優，政治家，スポーツ選手などの人名を人手で集め 37490 人の人名リスト（名称のみ）を作成した．この人名リストを対象に，上記の方法で Wikipedia 中から抽出した，姓名に分割された名称と，その読みのひらがな，カタカナ，愛称の 4 つを学習用データとして使用する．

37490 人の人名リストに対して Wikipedia から自動抽出できた愛称が 1662 人分(愛称 2587 個)であり，このうち名称と読みの両方が姓名に分割取得できた人物の愛称が 921 人分であった．ここから評価用データに含まれる 108 人を除いた 874 名分(愛称：1306 個)を 4.3 節の愛称生成ルールの学習に用いた．学習された愛称生成ルールは 32327 個であった．

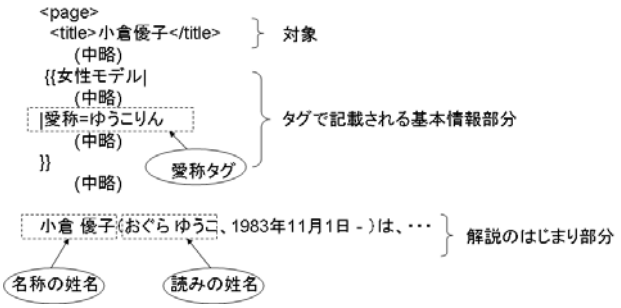

図 5 Wikipedia データベース中の記載例.

Fig.5 An example of a data in Wikipedia database.

5.4.2 パラメータ設定

4.2～4.5 節でのべた提案手法中の各パラメータは，$n=100$, $s=100$, $t=10$, $r=2$, $q=0.2$ とした．また，検索結果の取得や検索結果数の取得にはYahoo! JAPAN[7]の提供するYahoo! JAPAN Webサービス[8]を使用した．また，4.3 節の処理の後，ルールの追加([14]参照)を行った．

5.5 比較手法

以下の 3 手法と提案手法の比較を行う．先行研究[2][3]では，正解と想定する呼び名は本研究と異なる可能性があるが，比較のために本研究で正解とみなす愛称データで評価を行った．検索には Yahoo! JAPAN Web サービスを利用し，検索結果の上位 100 文書を対象とした．

5.5.1 Wikipedia からの自動抽出

Wikipedia中の愛称タグから評価用データ中の人名に対して愛称を自動抽出した(5.4.1節の方法)．評価用データ108人中，56人分（96個の愛称）が取得できた．

5.5.2 外間ら[2]による人物の呼称抽出手法

外間らによる人物の呼称抽出手法(先行1とする)を用い，5.2節で述べた評価用データを対象に実験を行った．各パラメータは論文中の記載と同じ設定にした．ただし，文献[2]で同姓同名を区別するためとして与えている$relobj$の計算には人手を介す必要があるため，本実験では対象とする名称と同姓同名はいないものと仮定し$relobj$は与えなかった．形態素解析にはMeCab[15]を使用した．

5.5.3 本間ら[3]による人物の別名抽出手法

本間らは独自の複数の方法について比較を行っている．このうち２つの手法（先行2-1，先行2-2と呼ぶ）を本稿の比較

[7] http://www.yahoo.co.jp/

[8] http://developer.yahoo.co.jp/search/web/V1/webSearch.html

対象とし実験を行った．先行2-1は，表現パターン“○○こと(名称)”を使った手法をさす．また，先行2-2は，6つの表現パターンから抽出した別名候補をSVM(Linear)により統合しランキングする手法をさす．ランキングSVMでは学習が必要なため，9分割交差検定(評価用データ108人について96人分で学習,12人分でテスト)で評価した.閾値は,0.7とした.

表１．実験結果の比較.
Table 1 Comparison our method with other methods.

	Wikipedia (5.5.1)	先行１ (5.5.2)	先行 2-1 (5.5.3)	先行 2-2 (5.5.3)	提案手法 (4.6)
MAP	0.379	0.179	0.444	0.464	**0.549**
MPrec	**0.845**	0.122	0.203	0.205	0.343
MRPrec	0.382	0.149	0.424	0.448	**0.522**
出力人数	56	106	107	108	108

表 2. 提案手法とその部分手法の結果の比較.
Table 2 Comparison our method with its sub-methods.

	a)4.5 “こと(名称)”	b)4.4 名称由来	c)提案手法 a)+b) (4.6)
MAP	0.441	0.456	0.549
MPrec	0.379	0.337	0.343
MRPrec	0.441	0.446	0.522
出力人数	105	102	108

表３．人名「坂本龍一」の愛称出力結果.（正解は,「教授」.）
Table 3 Comparison between nicknames extracted by our method and by others in the case of “Ryuichi Sakamoto”.

順位	提案手法	先行１	先行 2-2
1	教授	坂本	教授
2	サカモト	教授	という
3	坂本	時代の	敬
4	龍さん	SAKAMOTO	世界の坂本
5	坂本龍一	坂本教授	世界や着メロでおなじ

5.6 実験の結果

Wikipediaからの自動抽出，先行研究の手法(先行1，先行2-1,先行2-2)と,提案手法の結果の比較を行った.MAP,MPrec,MRPrecの評価結果と出力人数(愛称候補出力が1個以上得られた人物の数．最大108人．)をまとめたのが表1である．表1を見るとMAP,MRPrecの2つで提案手法がもっとも良い性能を示した．また，2番目に良かった先行2-2に比べ，MAPでは18%向上，MRPrecでは16%向上がみられた．このことから提案手法は他手法に比べ，システム出力の上位の方に正しい愛称が多く見られ，正解データの数に合わせた再現性・適合性が高いと言える.一方,システム出力の信頼性を表すMPrecでは,Wikipediaからの自動抽出がもっとも良い結果であった.

さらに，提案手法の分析を行った．提案手法は，名前由来の愛称を対象とする4.4節の手法と，“こと(名称)”の直前の文字列を抽出する4.5節の手法の組み合わせである．そこで，4.4節と4.5節の各結果と，組み合わせ後の結果の比較を行った．MAP，MPrec，MRPrecの評価結果と出力人数をまとめたのが表2である．4.4節の手法に比べ，MAPでは20%，MRPrecでは17%向上していた．また，4.5節の手法に比べ，MAPでは24%，MRPrecでは18%向上していた．愛称には名前由来の愛称と名前由来でない愛称があるという知見から，各タイプの愛称に特化した抽出手法を組み合わせることで，各々単独の性能に比べ飛躍的に性能向上ができることがわかった．

出力結果の例を表3，表4に載せた．表3は，先行研究の各文献で出力例が掲載されていた“坂本龍一”について，先行1，先行2-2，提案手法による出力結果を示した．各文献中に記載の出力例と完全一致はしなかったが，似た単語が得られたことを確認するものである．表4は評価用データに対する提案手法の愛称候補結果の一部とその正否を示す．各人名において正解とした愛称も併記した．この結果から，提案手法により得られた愛称候補は，不正解となる愛称であっても，正解の愛称の表記ゆれである候補が多いことが分かる．

6. 有名人の愛称抽出

6.1 愛称抽出

実データに対応した愛称抽出を行う目的で，5.4.1節で述べた37490人の人名リストを対象に愛称抽出実験を行った．愛称抽出は，4章で述べた方法による．愛称生成ルールの学習には，5.4.1節で得た愛称(921人分)を使った．また，愛称生成ルールは，4.3節の手順で作成した796個のルール用いた．その結果，37490人に対し，愛称が１個でも付与された人数は6401人であった.また,抽出した愛称は13429個であった.

6.2 評価データの作成

有名人(37940人)に対し,次の方法で100人を抜粋し新たな正解データを作成し，評価を行った．まず，Web上のヒット件数の多い順に並べ，上位から姓名に分かれる人物名を100人選んだ．職種は，俳優・女優・スポーツ選手・政治家・タレントなど多岐に渡り，国籍は日本人95人のほか，韓国人3人,アメリカ人2人である.5.2との重複は18人あった.次に,上記100人について,Wikipedia中から愛称を人手で収集した.見つかった愛称は60人分で，これを評価用データとした.

6.3 愛称抽出結果と考察

60人の評価用データで提案手法と先行2-2について評価実験を行った結果が表5である．ただし，先行2-2のSVMの学習には，5.2節で作成した108人の愛称(256個)を利用した.

表1の結果に比べ，両手法とも精度が下がっていた．5.2節のデータに比べ様々な業種の人物が加わったことで愛称抽出特性が変わったと考えられる.また,提案手法は先行2-2に比べ，精度の低下が少なかった．名称由来の愛称を積極的に取得する愛称生成ルールを用いることで，愛称が取得しにくい対象人物に対しても頑強な手法になると考えられる.

6401人分の愛称抽出結果を見ると,愛称抽出が間違っている場合(失敗事例)には次のようなものがあった.

1)「こと」の前の文字列抽出で失敗している
　例) タン，チャン，ッチ
2)配役名を取得している
　例) 岡田准一⇒ぶっさん，仲間由紀恵⇒ヤンクミ
3)関連するが愛称以外の語を取得している
　例) タモリ⇒ユンケル，橋下徹⇒弁護士，阿部寛⇒結婚できない男
4)1)～3)とは別の文脈で「こと」に続く文字列を抽出
　例) 思った(こと)，した(こと)，にしおかすみ(こと)

失敗事例の2～4は，「こと(人名)」が愛称以外の文脈でも使われることに起因する．また，失敗事例の1は，「こと」の前の文字列抽出の範囲を決めるのが難しいことに起因する．今後の改良点として，「(愛称)こと(人名)」をランキングに使わないで愛称らしさを推定する方法が考えられる.

表４．提案手法で得られた上位５位の愛称候補と用意した正解と正否の例．（正解を○，不正解を×で表記．）
Table 4 Examples of the top five nicknames extracted by our method and correct nicknames prepared in advance.

順位	若槻千夏	正否	倖田來未	正否	保田圭	正否	辻希美	正否	大野智	正否
1	チナッティー	○	くぅちゃん	○	圭ちゃん	○	のんちゃん	○	キャプテン	○
2	チナッティ	×	くうちゃん	×	けいちゃん	×	ののたん	○	リーダー	○
3	ブログの女王	×	くーちゃん	×	やすす	×	つじじ	×	大ちゃん	○
4	ちぃちゃん	○	歌姫	×	圭さん	×	辻ちゃん	○	大チャン	×
5	ちなっちゃん	○	くぅチャン	×	ケメコ	×	ののちゃん	×	さとやん	×
正解	ちぃ ちっち ちぃちゃん ちなっちゃん チナッティー 若槻！		くぅちゃん くぅたん		圭ちゃん ケメ子		のの のん 辻ちゃん ののたん のんちゃん つーじー		大ちゃん おおちゃん リーダー キャプテン	

表５ 実験結果の比較．
Table 5 Experimental results.

	提案手法	先行 2-2
MAP	0.433	0.300
MPrec	0.245	0.139
MRPrec	0.392	0.287
出力人数	55/60	59/60

7. まとめ

愛称抽出手法として，愛称生成ルールに基づく愛称生成手法と，“（愛称）こと（名称）”という手がかり句を利用した文字列抽出手法との組み合わせによる手法を提案した．また，愛称の定義をし，既存手法との数値的な比較評価を行った．

愛称抽出精度は，情報検索で用いられる評価方法を用い評価を行った．提案手法による愛称抽出では既存手法に比べMAP で 18%以上の精度向上ができた．また，提案手法の構成要素である名前由来の愛称を抽出する手法と，名前由来でない愛称も含めて抽出する手法のそれぞれの MAP に比べて，提案手法では 20～24%の精度向上がみられた．すなわち，名前由来である愛称と名前由来でない愛称のそれぞれに特化して愛称を抽出しその結果を組み合わせることで，それぞれでの精度よりも上回る精度が得られることが分かった．

本研究では愛称抽出後に “（愛称）こと（名称）”での検索結果数によって順位付けを行っている．今後は，愛称らしさの推定方法を検討し，愛称候補の順位付けに反映したい．

[文献]

[1] 大辞林 第二版，三省堂, http://dictionary.goo.ne.jp/

[2] 外間智子，北川博之: “Web データを用いた人物の呼称抽出”, DBSJ Letters, Vol.5, No.2, pp.49-52 (2006).

[3] 本間大輝, Danushka Bollegala, 松尾豊，石塚 満: “Web を用いた人物の別名抽出”, NLP 若手の会第 2 回シンポジウム，発表 12 (2007).

[4] Brin, S.: “Extracting Patterns and Relations from the World Wide Web.”, EDBT'98, pp.172-183 (1998).

[5] Lin、D. and Pantel, P.: “Concept Discovery from Text”, COLING'02, pp.577-583 (2002).

[6] Schwartz, A. S. and Hearst, M. A.:“A Simple Algorithm For Identifying Abbreviation Definitions in Biomedical”, PSB2003, pp.451-462 (2003).

[7] Nadeau, D. and Turney, P. D.: “A Supervised Learning Approach to Acronym Identification”, AI-2005, pp.319-329 (2005).

[8] 村山紀文，奥村学: “Noisy-channel model を用いた略語自動推定”, NLP2006, pp. 763-766 (2006).

[9] Hearst, M. A.: “Automatic Acquisition of Hyponyms from Large Text Corpora”, COLING'92, pp.539-545 (1992).

[10] Tanaka, I. K., Yamamoto, M. and Nakagawa, M.: “Kiwi: a multilingual usage consultation tool based on internet searching”, the Interactive Posters/Demonstrations, ACL-03, pp.105-108 (2003).

[11] Wikipedia データベース http://download.wikimedia.org/jawiki/20071013/.

[12] 藤井敦: “Web 検索におけるアンカーテキストのモデル化と質問の自動分類”, IC2007, pp.87-96 (2007).

[13] 酒井哲也: “情報検索テストコレクションと評価指標”, SIGNL-183, pp.1-8 (2008).

[14] 若木裕美，藤井寛子，福井美佳，住田一男: “Web 情報を用いた人物の愛称抽出手法”, DEWS2008, 7A-2.

[15] MeCab http://mecab.sourceforge.net/.

若木 裕美 Hiromi WAKAKI

(株)東芝・研究開発センター所属．2007 東京大学大学院情報理工学系研究科電子情報学専攻博士課程修了．同年(株)東芝入社．博士(情報理工学)．情報検索，テキストマイニングの研究に従事．情報処理学会会員．

藤井 寛子 Hiroko FUJII

(株)東芝・研究開発センター・研究主務．1994 慶応義塾大学大学院理工学研究科計算機科学専攻修士課程修了．同年 (株)東芝入社．マルチメディア検索，ナレッジマネジメントシステム，音声対話等の研究開発に従事．日本ソフトウエア科学会会員．

福井 美佳 Mika FUKUI

(株)東芝・研究開発センター・主任研究員．1986 横浜国立大学工学部情報工学科卒業． 同年(株)東芝入社． ヒューマンインタフェースの研究・開発に従事. 1996 年情報処理学会山下記念賞受賞． 情報処理学会会員．

住田 一男 Kazuo SUMITA

(株)東芝・研究開発センター・研究主幹．1982 東京工業大学大学院総合理工学研究科修士課程修了，1982 東京芝浦電気(株)(現(株)東芝)入社，博士(工学)．自然言語処理，音声翻訳，音声対話等の研究・開発に従事．情報処理学会，人工知能学会，電子情報通信学会，言語処理学会各会員．